Multi-scale Hash Encoding based Neural Geometry Representation

Zhi Deng[1] Haiyao Xiao[1] Yining Lang Hao Feng[2] Juyong Zhang[1*]
[1]University of Science and Technology of China [2]Alibaba Group
[1]Hefei, China [2]Hangzhou China
{zhideng,xhy1999512}@mail.ustc.edu.cn {yuanning.fh, louis.lyn}@alibaba-inc.com
juyong@ustc.edu.cn

Abstract

Recently, neural implicit function-based representation has attracted more and more attention, and has been widely used to represent surfaces together with differentiable neural networks. However, existing neural geometry representations still suffer from slow computation speed and insufficient reconstruction accuracy when applied to surface reconstructions from point clouds and multi-view images. To alleviate these issues, we propose a multi-scale hash encoding-based neural geometry representation to effectively and efficiently optimize the surface represented as a signed distance field. To this end, a novel neural network structure is proposed by carefully combining low-frequency Fourier position encoding with multi-scale hash encoding. Accordingly, the initialization of the geometry network and geometry features of the rendering module is redesigned. Extensive experiments demonstrate that our proposed representation achieves at least 10 times speedup on the task of million-level point cloud reconstruction, and significantly improves the efficiency and accuracy on the multi-view reconstruction task. Our code and models will be available at https://github.com/Dengzhi-USTC/Neural-Geometry-Reconstruction.

Keywords: Neural Geometry Representation, Hash Encoding, Point Cloud Reconstruction, Multi-view Reconstruction

1. Introduction

3D geometry shape is fundamental to many problems in computer graphics, computer vision, and robotics due to the fact that our physical world is in the 3D space. Unlike the image, which is usually represented as a regular matrix in the digital world, 3D geometry shapes have various representations according to different applications. Conventional geometry representations such as the polygon mesh, point cloud, and voxel grid can directly model 3D objects, but suffer from excessive storage to represent high-precision geometries. While the parametric geometry representation describes 3D objects via a series of basis functions, it is limited by the expressive ability of low-dimensional parametric space. Recently, the MLP-based neural implicit representations have demonstrated great success in the aspects of effectiveness and compactness. The coordinate-based MLP models the 3D space as a continuous implicit function by mapping a given point to its corresponding scalar attribute, such as the occupancy and (un) signed distance value. Then the geometric surface can be extracted from a specified level-set via Marching Cubes [32].

This work is interested in inferring the signed distance field from an unorganized input point cloud or calibrated multi-view images. In other words, we aim to learn a coordinate-based implicit function $\Phi(\theta, \mathbf{x})$ with learnable parameters $\theta \in \mathbb{R}^d$, which satisfies the eikonal equation:

$$\|\nabla_{\mathbf{x}}\Phi\| \equiv 1, \quad s.t \quad \{G_k(\Phi(\theta, \mathbf{x}_i)), \mathbf{x}_i \in \partial\Omega\}_{i,k}, \quad (1)$$

where $\mathbf{x} \in \mathbb{R}^3$ is a 3D point, Ω is a well-behaved open set with boundary $\partial\Omega$, and $G_k(\cdot)$ is a non-linear constraint for geometry representation Φ. MLP-based approaches [20, 68] firstly introduce the eikonal equation constraint to enforce the neural implicit function as a signed distance field under the input of point cloud and multi-view images, respectively. However, as analyzed in [44, 35], simple coordinate-based MLPs with ReLU activation have limited representation ability due to the "spectral bias" of neural networks. In addition, this geometry representation often takes a long time to solve the geometry reconstruction problem.

Many works have been proposed to improve the capability of neural geometry representation. Some works try to design more powerful activation functions, like SIREN [46]. Some other works focus on positional encoding, which encodes the spatial location into a high-dimensional space via a given set of sinusoidal functions or spline functions [50, 21, 56]. These methods aim to represent the high-frequency details of a given surface shape, but still, fail to reconstruct geometric details accurately and efficiently. To tackle this problem,

*Corresponding author

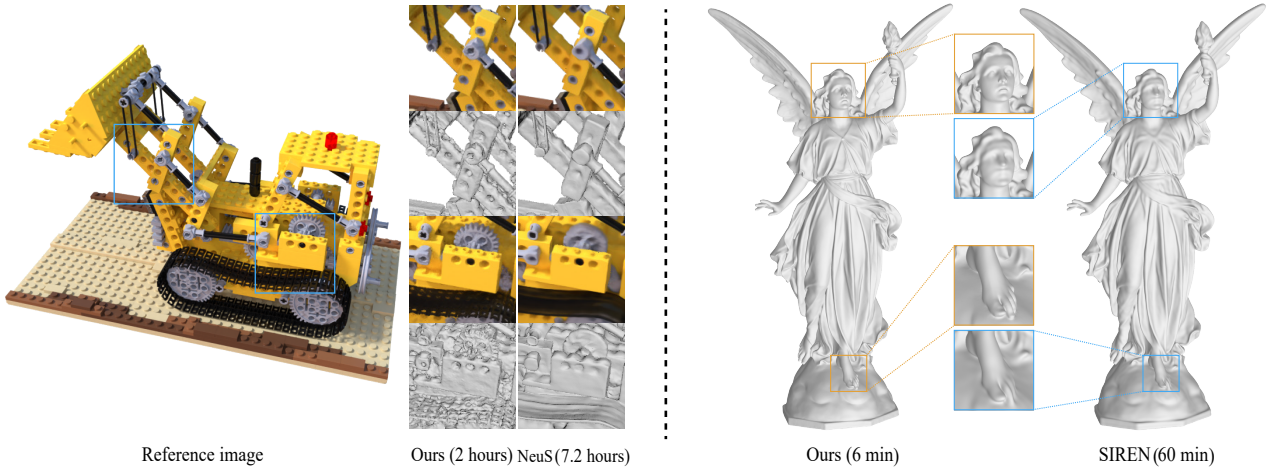

Figure 1. The left and right show our multi-view and point cloud reconstruction results, respectively. Compared with the state-of-the-art method NeuS [55], our method is 3.6 times faster, and the reconstruction accuracy is significantly improved. Compared with the state-of-the-art method SIREN [46], our method also considerably improves the optimization speed and reconstruction accuracy on the point cloud reconstruction.

learnable positional encoding has been introduced to further encode the local geometric details of a given point based on a predefined voxel grid. In particular, the learnable multi-scale hash encoding [36] could efficiently obtain the multi-scale geometric information. However, reconstruction artifacts would appear by directly applying this encoding strategy to neural geometry reconstruction tasks. The possible reasons include the following two aspects. First, it does not satisfy eikonal constraints well under weak supervision, and explicitly discrete grids-based neural geometry representation has poor gradient continuity, resulting in a poor approximation of the signed distance field. Second, it requires reasonable initialization to help network optimization.

We propose a novel geometry representation based on multi-scale hash encoding to address these issues. Specifically, we move the hash encoding to the hidden layer as part of the input of the connected layer, and introduce the Fourier position encoding as the input of the first layer to encode consecutive spatial locations, enhancing the continuity of gradients of geometry representations. We also initialize the geometry network with the modified version of SAL [1], where the optimization of the geometry module is started from an approximate sphere.

To further verify the effectiveness of our geometry representation on multi-view reconstruction tasks, we use NeuS [55] as the baseline framework, and validate the reconstruction accuracy and efficiency. In the previous volume rendering frameworks, the geometry features of any point in 3D space are extracted from the last layer of the geometry network. Thus, hash encoding of geometry representation encodes geometry features. This causes the reconstructed geometry to be inconsistent with the multi-view image rendered by the light field, due to the fast learning ability of the hash encoding operator. Thereby, we change the feature extraction position in the geometry network to the connection layer, such that the multi-scale hash encoding only represents the geometry.

Extensive experiments and ablation studies demonstrate that our neural geometry representation outperforms the state-of-the-art neural geometry representations in terms of computation efficiency and accuracy of reconstructions from point clouds and multi-view images. Compared with existing neural implicit function-based point cloud reconstruction methods, our method achieves at least 10 times speedup and significantly improves the reconstruction accuracy. In the multi-view image reconstruction task, benefiting from the modified rendering framework, our approach can recover fined geometry details and the reconstruction speed is at least 3.6 times faster than the state-of-the-art.

2. Related works

Neural Geometry Representation. Recently, the coordinate-based neural networks $(\Phi(\theta, \mathbf{x}), \mathbf{x} \in \mathbb{R}^3)$, which represent a 3D object as a continuous geometry shape, have attracted a lot of attention. [40, 31, 13] utilizes a neural network to represent the 3D shape as a (sign) distance field, and [34, 11] represent it as an occupancy field. Our work is more related to IGR [20], which uses simple coordinate-based MLPs to recover the SDF of 3D shapes. However, the implicit representation based on coordinate-based MLPs cannot represent high-frequency details well due to its limited representation ability [50]. Xiao *et al.* [60] give a more detailed description to representation. Many methods have been proposed to address this issue.

SIREN [46] uses Sine as the activation function and proposes a proper initialization method for optimizing. Some works [43, 25, 7, 19, 30] divide complex shapes or large-

scale scenes into regular subregions and replace the global MLP with local MLPs, so as to improve the geometry representation. It has been proved that the sinusoidal positional encoding can improve the performance of MLPs with ReLU in radiance fields fitting [35]. One following work, Hertz *et al*. [21] proposes a spatially adaptive progressive encoding (SAPE) scheme based on sinusoidal positional encoding, which makes MLP-based representation better fit the target signals with complex frequencies. Except for the sinusoidal function, uniform parametric spline basis functions are also utilized as position encoding to improve local and high-frequency geometry information [56].

Another strategy is decomposing the learnable feature or domain into different components based on an explicit 3D data structure. EG3D [8] encodes the 3D position of geometry rendering features by projecting it into a triplane with learnable features to accelerate the training speed. ACORN [33] applies a tree subdivision to the domain, wherein a large learnable auxiliary coordinate encoder neural network is trained to output dense feature grids. Features of these dense grids are used to represent the position encoding of any point in space. NGLOD [49] represents neural implicit function using an octree-based position encoding, which adaptively fits shapes with multiple discrete levels of detail (LOD).

Hash Encoding-based Methods. As an efficient encoding tool, hash encoding is also widely used in geometry reconstruction. Voxel hashing is utilized in [38] to reconstruct an online system for large and fine-scale volumetric reconstruction. A dynamic spatially-hashed truncated signed distance field is applied in [28] to contribute to a real-time house scale dense 3D reconstruction system. Recently, a learnable multi-resolution hash encoding framework [36] has been proposed to encode the 3D position, and it has been successfully applied to tasks for fast training of neural radiation fields and SDF fitting. The work of [18] learns a series of neural radiance fields as facial expression basis by hash encoding to enable semantic control over personalized semantic NeRF.

Point Cloud Reconstruction. Given a point cloud (possibly with normals), reconstructing its corresponding 3D geometry shape is a classical problem in digital geometry processing. The parametric RBFs representation is utilized to reconstruct the surface by fitting point cloud [6, 52]. Besides, a widely used approach is the Poisson surface reconstruction [27], which solves a Poisson equation on a discrete volume based on the given points and normals. For more related works, the readers can refer to the survey [4].

Recently, surface reconstruction based on coordinate-based neural implicit representations has achieved great progress. DeepSDF [40] utilizes a neural implicit function to decode the SDF of 3D position in a bounding volume. Points2Surf [15] decomposes the neural geometry representation into a global sign function and local absolute distance function. Based on the eikonal equation, IGR [20] proposes a new paradigm for computing high-fidelity implicit neural representations directly from raw 3D points.

SALD [2] advocates a novel sign agnostic regression loss, which incorporates both point-wise values and gradients of the unsigned distance function. Neural-Pull [3] uses the predicted signed distance value and the gradient at query locations to train a high-quality neural geometry representation. To solve problems not limited to a specific topology or type of input 3D signal, Chen *et al*. [10] proposes a new data-driven approach for mesh reconstruction based on dual contouring.

Multi-view Image Reconstruction. Traditional MVS algorithms focus on neighbor view selection algorithms and photometric error measures. Robust neighbor view selection and visibility consistency algorithm are discussed in depth in the [17] and [29], respectively. A current popular MVS system, COLMAP [45], jointly estimates depth and surface normal, leverages photometric and geometric priors for pixel-wise view selection, and uses geometric consistency for simultaneous refinement. We refer readers to [16], a comprehensive overview of classical multi-view stereo reconstruction algorithms. Classical learning-based MVS methods attempt to replace some components of the traditional MVS pipeline. Some works learn to match 2D features across views [26, 53, 9] or infer depth maps from multi-view images based on the data-driving framework [63, 65, 42]. There are also some works [61, 64, 12, 54] discuss in depth the memory reduction of 3D convolution and speeding up the inference speed of the model.

Recently, inverse rendering-based approaches have achieved great success in multi-view reconstruction. DVR [37] proposes a differentiable rendering approach to directly optimize the shape and texture of the input RGB images. IDR [68] utilizes the neural implicit function to simultaneously learn geometry, camera parameters, and a neural render approximates the light reflected towards the camera. Wang *et al*. [57] using the prior information to extend the IDR to 3D head reconstruction. NeRF [35] proposes a novel view synthesis framework, which optimizes an underlying continuous volumetric scene function using multi-view images. VolSDF [67] attaches volume rendering techniques to IDR and eliminates the need for mask information. UNISURF [39] proposes a new multi-view framework where implicit surface models and radiation fields can be formulated in a unified way. Thus they can achieve surface and volume rendering using the same model. NerfingMVS [59] proposes a multi-view framework with learning-based priors to guide the optimization process of NeRF. NeuralRecon [48] offers a neural network to directly reconstruct local surfaces represented as sparse TSDF volumes for each video fragment sequentially. MVSDF [69] jointly optimizes an SDF and a surface light field appearance model, which are

directly supervised by geometry from stereo matching, and refined by the multi-view feature consistency and the fidelity of rendered images. DI-Fusion [22] proposed a local implicit function based framework for online 3D reconstruction with a commodity RGB-D camera. NeuS [55] proposes an unbias density representation to recover high-quality surface shape with the help of differentiable volume rendering.

3. Method

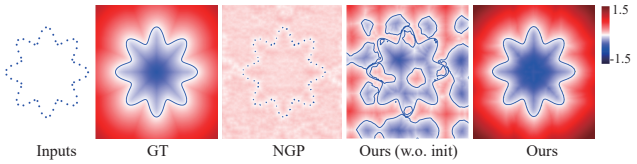

Figure 2. Illustration of our exploration of neural geometry representation abilities for the 2D point cloud (sampled from a snowflake curve) reconstruction task. Left to right: input (2D point cloud with normals), GT, the baseline of hash encoding (NGP), Ours (w.o.init), Ours. The blue curves are the extracted zero level set via Marching Cubes. The color of the images represents the numerical value of the reconstructed implicit function. The values of the eikonal constraints D_{eik} are 0.028, 0.0093, and 0.0041 from the third to the fifth column. And the values of the grad-error $D_{\text{grad-error}}$ are 0.29, 0.09, 0.04 from the third to the fifth column.

3.1. Neural Geometry Representation

3.1.1 Background

The popular coordinate-based geometry representation is to represent the 3D geometry shape as SDF: $\Phi : \mathbb{R}^3 \rightarrow \mathbb{R}$, $SDF(\mathbf{x}) = \Phi(\theta, \mathbf{x})$, where θ is learnable parameters. Based on this continuously differentiable geometry representation, the traditional geometry reconstruction can be directly solved by an end-to-end optimization framework. e.g., reconstruction from point clouds or multi-view images. The two most important validation metrics are the reconstruction accuracy and efficiency for geometry reconstruction problems. Recently, NGP [36] utilizes a learnable multi-scale hash encoding $\text{enc}(\theta_h, \mathbf{x})$, which encodes the 3D position of the space into the learnable features θ_h of hash table. The $\text{enc}(\theta_h, \mathbf{x})$ improves the expressive ability of geometry representations and convergence speed for geometry fitting tasks. e.g., SDF fitting and neural radiance field fitting. Thus, we introduce it to the geometry reconstruction tasks under weak supervision information.

Our geometry representation utilizes the multi-scale hash encoding $\text{enc}(\theta_h, \mathbf{x})$, which derives from NGP. For more specific details, we first construct hash tables arranged in L levels, with each level l containing up to T_l learnable features of dimension F. Each level independently and conceptually stores feature vectors at the vertices of a grid of a given resolution. Similar to NGP, the resolution of each level is progressively set to a value between the coarsest and finest resolutions $[N_{\text{min}}, N_{\text{max}}]$:

$$N_l = \lfloor N_{\text{min}} * b^l \rfloor.$$

Here b is the per level scale $2^{(\frac{1}{L}\log_2(\frac{N_{\text{max}}}{N_{\text{min}}}))}$ and N_{max} is a predefined target resolution. We set N_{max} and N_{min} to be 2048 and 16 respectively in our experiments.

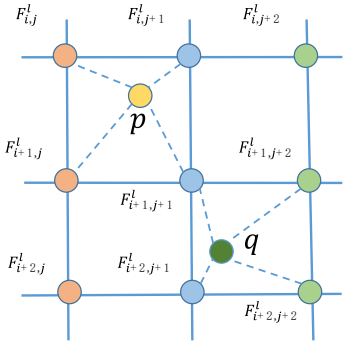

Figure 4. It shows two spatially adjacent points p and q obtain interpolation features on the l-level grid. The features of p and q are also quite different, due to the discontinuity of features on the explicit grids.

Then, for any point $\mathbf{p} \in \mathbb{R}^3$ in the space, we compute the $\text{enc}(\theta_{\text{h}}, \mathbf{p})$ by finding the corresponding position on the grid of different levels through the predefined mapping firstly and then extract the learnable features from features stored in hash tables at different resolutions. Specifically, considering a single level l hash table, point $\mathbf{p}$ is firstly scaled by the level's grid resolution and then rounded down and up $\lfloor \mathbf{p}_l \rfloor = \lfloor \mathbf{p} * N_l \rfloor$, $\lceil \mathbf{p}_l \rceil = \lceil \mathbf{p} * N_l \rceil$. $\lfloor \mathbf{p}_l \rfloor$ and $\lceil \mathbf{p}_l \rceil$ span a voxel grid with 2^3 integer vertices in $\mathbb{Z}^d$, then, the $\lfloor \mathbf{p}_l \rfloor$ and $\lceil \mathbf{p}_l \rceil$ are mapped into the hash-table by mapping function $h : \mathbb{Z}^3 \rightarrow \mathbb{Z}_{T_l}$. In the mapping function h, we map each point into the feature grid according to a one-to-one mapping , when $N_l^3 \leq T_l$. For the fine level, due to storage limitations, we use a hash function to determine its position in the feature hash table, similar to the strategy in NGP.

Finally, the feature vectors of the point $\mathbf{p}$ at each level are obtained via the tri-linear interpolation according to the relative position of $\mathbf{p}$ within its hypercube of the hash table. To improve the continuity of high-level features, we use a second-order continuous interpolation weight function $d(x) = 6x^5 - 15x^4 + 10x^3$.

3.1.2 Baseline Geometry Representation (NGP)

Fig. 2 provides a 2D example of our geometry representation exploration process. We validate the baseline network structure in NGP for the point cloud reconstruction task. It takes a hash encoding of a 3D point as input and outputs a 1-dimensional scalar value with several hidden layers. However, it results in non-smooth reconstruction results, and the gradient of the geometry representation has very poor continuity, as shown in Table 3. The reconstructed snowflake curve is wrong, resulting in a large difference between the

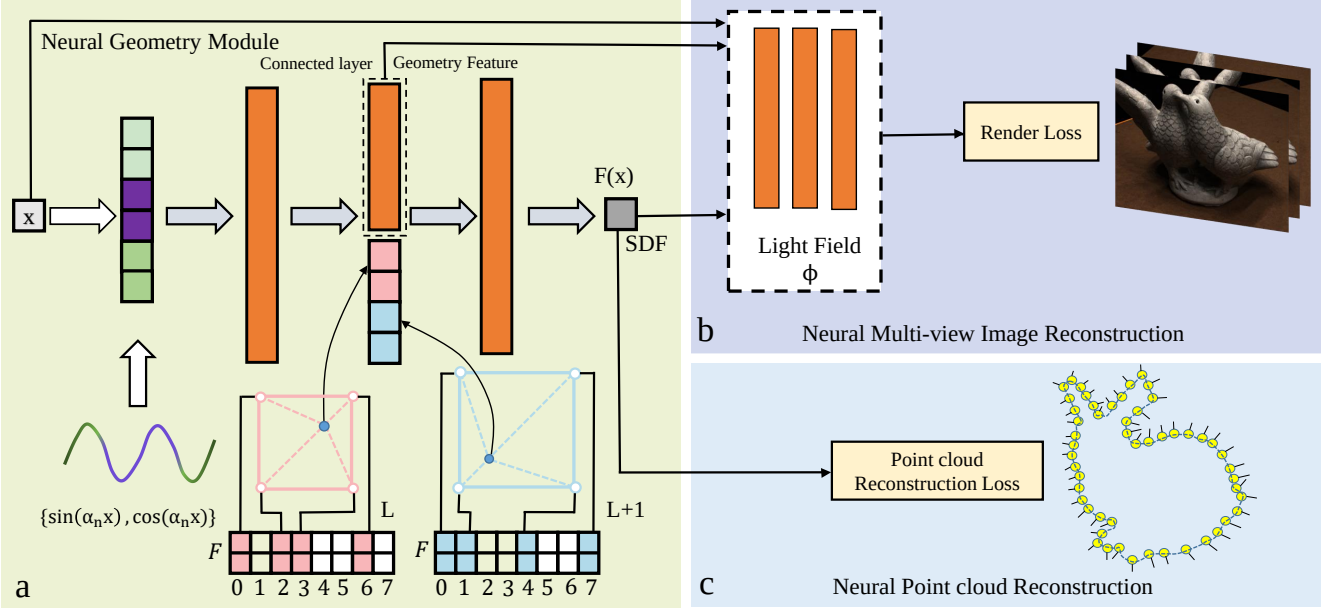

Figure 3. The left and right are our geometry representation and its application to geometry reconstruction tasks. We propose to combine Fourier position encoding and multi-scale hash encoding for neural geometry representation (a). Furthermore, regarding the geometry features of the multi-view rendering framework, we extract them from the connected layer (inside the dashed box), such that hash encoding is well avoided for ambiguity geometry learning. (The description diagram of hash encoding borrowed from NGP [36].)

predicted SDF and ground truth; the straight reason is that our supervision information is much weaker than that of NGP, turning ground truth supervision into weakly supervised. This phenomenon also has a great impact on the multi-view image reconstruction task, as shown in Fig. 10. What we observed, the more substantial reason is mainly two aspects. On the one hand, only using the multi-scale hash encoding neural network structure is not easy to minimize the eikonal constraint well under the constraint of weak supervision information. On the other hand, the neural geometry representation based on the explicit grid has discontinuities. The high compression with hash map h makes gradient discontinuities in geometry representation worse. As shown in Fig. 4, point p and point q are adjacent in space, but due to the discontinuity of the features on the grid, they are far away in the feature space. One straightforward solution is to increase the number of sampling points for computing the eikonal equation constraint, but it still can not solve this problem well.

3.1.3 Our Neural Geometry Representation

As shown in Fig. 3.a, based on the baseline network (NGP), we first introduce the connected layer, a hidden layer in the network used to connect features between different layers. Next, we move the hash-encoding $\mathrm{enc}(\theta_h, \mathbf{x})$ to the middle layer of the MLP as the input of the connected layer. Then the low-dimensional Fourier position encoding $\mathrm{enc}_{\mathrm{ff}}(\mathbf{x})$ is added as the input of the first layer to reduce the discontinuity of learnable features brought by hash encoding. Finally, the modified initialization of SAL [1] is utilized to initialize our geometry network. As shown in Fig. 5, it initializes our network parameters into an approximately spherical shape. Our final result is shown in Fig. 2 and Table 3, where the eikonal constraints are better satisfied, and the reconstruction results get smoother. Our final neural geometry representation is expressed as: $\Phi(\theta, \mathrm{enc}_{\mathrm{ff}}(\mathbf{x}), \mathrm{enc}(\theta_h, \mathbf{x}))$.

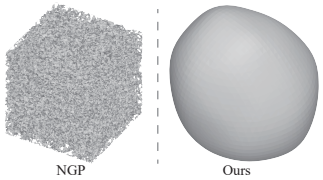

Figure 5. The initial results of the NGP and our geometry network with default initialization.

3.1.4 Modification of Geometry Features

A neural rendering-based multi-view reconstruction framework usually consists of a geometry network and a color network, as discussed in Sec. 3.2.2. The color network predicts the RGB value of a given point with its geometry features, which are previously extracted from the last layer of the geometry network. In our geometry representation, due to the fast learning ability of hash encoding, the hash encoding of geometric features may encode rendering properties. As a result, the geometry module and the color module cannot be well decoupled. Thus, the reconstructed geometry would be inconsistent with the rendered image, as analyzed in Sec. 5.3. To avoid this problem, we directly extract the geometry features from the connected layer in the geometry network, thus ensuring that the hash encoding only encodes the geometry

(SDF) of the object, as shown in Fig. 3.a.

3.2. Applications of Our Geometry Representation

3.2.1 Neural Point cloud Reconstruction

Similar with [46, 40], we embed our neural geometry representation $SDF(\mathbf{x}) = \Phi(\theta, \text{enc}_{\text{ff}}(\mathbf{x}), \text{enc}(\theta_h, \mathbf{x}))$ into the classical point cloud reconstruction task. Given an input point cloud $\{\mathbf{p}_i | \mathbf{p}_i \in \mathbb{R}^3\}$ and normal $\{\mathbf{n}_i | \mathbf{n}_i \in \mathbb{R}^3\}$ of the underlying surface S, the neural point cloud reconstruction task aims to infer a neural signed distance function of the surface S under a bounded volume Ω ($\{\mathbf{p}_i\} \subset \Omega$). As indicated in Eq. (1), the constraints $\{G_k(\cdot)\}$ encourages $\{\mathbf{x}_i\}$ to be on the surface, and the gradient of the implicit surface at $\{\mathbf{x}_i\}$ should be identical with the given normal $\{\mathbf{n}_i\}$. Specifically, we optimize our neural geometry representation via the following loss terms:

$$L = L_{\text{data}} + \lambda_1 L_{\text{eikonal}} + \lambda_2 L_{\text{off}}, \tag{2}$$

where

$$L_{\text{data}} = \int_{\Omega_0} |\Phi(\mathbf{x})| + \lambda_3 (|1 - < \nabla_{\mathbf{x}} \Phi(\mathbf{x}), \mathbf{n}(\mathbf{x})) > |) d\mathbf{x},$$

$$L_{\text{eikonal}} = \int_{\Omega} (\|\nabla_{\mathbf{x}} \Phi(\mathbf{x})\| - 1)^2 d\mathbf{x},$$

$$L_{\text{off}} = \int_{\Omega \setminus \Omega_0} \psi(\Phi(\mathbf{x}), \beta_0) d\mathbf{x}.$$

Here, Ω_0 represents the input point set $\{\mathbf{p}_i\}$, and the function $\psi(x, \beta_0)$ can be formulated as $\exp(-\beta_0 * |x|), \beta_0 \gg 1$, $\{\lambda_i\}_{i=1}^4$ are balance weights. Among them, the data term L_{data} constraints the implicit function Φ by using oriented points sampled on the point cloud. The off-surface term L_{off} encourages the value of non-surface points to be nonzero.

3.2.2 Neural Multi-view Image Reconstruction

Given calibrated multi-view images, neural multi-view reconstruction decouples geometry and appearance from them. Its geometry and appearance are represented by implicit signed distance function (SDF) and light field, respectively. In our work, the SDF is represented by our neural geometry representation $SDF(\mathbf{x}) = \Phi(\theta, \text{enc}_{\text{ff}}(\mathbf{x}), \text{enc}(\theta_h, \mathbf{x}))$. To optimize the parameters of our geometry representation, we utilize the state-of-the-art volume rendering framework NeuS [55] to render the 2D images based on neural implicit SDF and light field, and then minimize the difference between the rendered images and the inputs. It needs to be noted that the volume rendering framework contains a color network $c : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}^3$, which encodes the color associated with geometry properties of a point $\mathbf{x} \in \mathbb{R}^3$ and view direction $\mathbf{v} \in \mathbb{S}^2$.

Rendering. We render the proposed geometry representation with the corresponding light field to 2D images via a volume rendering framework and then measure the difference between the rendered images and the input images for network supervision. Specifically, given a pixel from the input image I_t, we denote the ray along the center of the camera through this pixel as $\{\mathbf{r}(s) = \mathbf{o} + s\mathbf{v} | s \geq 0\}$, where $\mathbf{o}$ is the center of the camera and $\mathbf{v}$ is the unit direction vector of the ray. We integrate the color along the ray as:

$$\hat{C}(\mathbf{r}) = \int_{s_n}^{s_f} T(t)\sigma(\mathbf{r}(t))c(\mathbf{r}(t), \mathbf{v})\, \mathrm{d}t, \tag{3}$$

$$\hat{M}(\mathbf{r}) = \int_{s_n}^{s_f} T(t)\sigma(\mathbf{r}(t))\, \mathrm{d}t, \tag{4}$$

where $\hat{C}(\mathbf{r})$ is the output color of this pixel, $\hat{M}(\mathbf{r})$ is the sum of the transmittance weights along the camera ray, s_n and s_f represent near and far bounds of the ray $\mathbf{r}$. $T(t) = \exp(-\int_{s_n}^{t} \sigma(\mathbf{r}(u))\, \mathrm{d}u)$ denotes the accumulated transmittance along the ray, and $c(\mathbf{r}(t), \mathbf{v})$ is the color at the point $\mathbf{r}(t)$ along with the viewing direction $\mathbf{v}$. We set $\sigma(\mathbf{r}(t))$ as an unbiased and occlusion-aware function, defined in NeuS [55]. Finally, we use the following loss functions to optimize the network parameters of the geometry module and the color module.

$$L = L_{\text{color}} + \alpha L_{\text{eikonal}} + \beta L_{\text{mask}} + \gamma L_{\text{off}}, \tag{5}$$

where α, β and γ are balance weights, and the color term L_{color} is defined as

$$L_{\text{color}} = \frac{1}{\#\mathcal{R}} \sum_{\mathbf{r} \in \mathcal{R}} \|M(\mathbf{r})(\hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r}))\|_1,$$

where

$$\mathcal{R} = \mathcal{R}(\{\mathcal{K}_i\}, \{\mathcal{T}_i\}),$$

$$\#\mathcal{R} = \sum_{\mathbf{r} \in \mathcal{R}} M(\mathbf{r}).$$

Here, $\mathcal{R}(\{\mathcal{K}_i\}, \{\mathcal{T}_i\})$ represents ray set constructed based on the pixels of all images, and $\{\mathcal{K}_i\}, \{\mathcal{T}_i\}$ are the intrinsic and extrinsic parameters of the camera, respectively. $C(\mathbf{r}) \in \mathbb{R}^3$ and $M(\mathbf{r}) \in \{0, 1\}$ are ground truth color and the object mask value of the ray $\mathbf{r}$, respectively.

The eikonal term, which regularizes the geometry representation $\Phi(\theta, \text{enc}_{\text{ff}}(\mathbf{x}), \text{enc}(\theta_h, \mathbf{x}))$ as SDF, is defined as

$$L_{\text{eikonal}} = \frac{1}{\#\mathcal{X}} \sum_{\mathbf{p} \in \mathcal{X}} (\|\nabla_{\mathbf{x}} \Phi(\mathbf{p})\|_2 - 1)^2,$$

where $\mathcal{X}$ is the sample point set on rays of set $\mathcal{R}(\{\mathcal{K}_i\}, \{\mathcal{T}_i\})$, and $\#\mathcal{X}$ is the number of points in $\mathcal{X}$.

The mask term L_{mask} is optional, and defined as

$$L_{\text{mask}} = \frac{1}{\#\mathcal{R}} \sum_{\mathbf{r} \in \mathcal{R}} \text{BCE}(\hat{M}(\mathbf{r}), M(\mathbf{r})),$$

where BCE is the binary cross entropy loss.

The off-surface loss L_{off} is defined as

$$L_{\text{off}} = \frac{1}{\sum_{\mathbf{x}\in\Omega} 1} \sum_{\mathbf{x}\in\Omega} \psi(\Phi(x), \beta_0),$$

where Ω is the bounding volume of the object. We uniformly sample 500 points per iteration in Ω, ψ is defined in Sec. 3.2.1, and the β_0 is 100.

4. Datasets and Implement Details

4.1. Datasets Description

For the neural point cloud reconstruction task, we evaluate our approach and baseline methods on the public "FAMOUS" dataset released by Points2Surf [15] and other cases from the Standard 3D Scanning Repository[1] and the online 3D data library[2], the total is 19 models. To show high-quality point cloud reconstruction results, we preprocess the above dataset by subdividing each original mesh into millions of points with normals. In addition, we normalize these point clouds into $[-1, 1]^3$.

For the neural multi-view reconstruction task, same as [68, 55, 39], we evaluate our approach and baseline methods on 15 scenes from the DTU dataset [23]. Each scene contains 49 or 64 images from different perspectives with corresponding extrinsic parameters and a foreground mask provided by IDR [68]. The resolution of each image is 1200×1600, and the intrinsic camera parameters of each scene are known. It is noted that the above dataset is particularly challenging for reconstruction algorithms due to its diverse materials, appearances, geometry, Non-Lambertian appearance, and thin structure effect. In addition, we conduct experiments and evaluations on some challenging scenes from the low-res set of the BlendedMVS dataset [66], which is a large-scale dataset containing multi-view images with respective camera extrinsic and intrinsic parameters. These cases have 31 to 143 images with a resolution of 768 × 576 and corresponding masks. Finally, some scenes from a large-scale multi-view face image dataset, i.e., FaceScape dataset [62], were evaluated. For each scene, we select 32 images with the 900 × 600 resolution and corresponding camera parameters, and a manually annotated rough foreground mask.

4.2. Evaluation Metrics

For the neural point cloud reconstruction task, to extract the fine geometry, we set the volume resolution as 2048^3 in the Marching Cubes algorithm [32]. And for each scene, we evaluate the quality of 3D surface reconstruction result by calculating the Chamfer-L_2 distance between the 10^7 uniformly sampled points on the reconstructed surface and the ground truth point cloud.

$$D_{\text{scd}}(P, Q) = \frac{1}{\#P} \sum_{\mathbf{p}\in P} \min_{\mathbf{q}_t\in Q} \|\mathbf{p} - \mathbf{q}_t\|_2^2, \tag{6}$$

$$D_{\text{cd}}(P, Q) = D_{\text{scd}}(P, Q) + D_{\text{scd}}(Q, P), \tag{7}$$

where P and Q are two point clouds, respectively,

$$D_{\text{lap}}(\{V, E\}) = \frac{\sum_{\mathbf{v}\in V} \| \sum_{\mathbf{u}\in\mathcal{N}(\mathbf{v})} \omega_{\mathbf{v},\mathbf{u}}\mathbf{u} - \mathbf{v}\|_2^2}{\text{ave-edge }(\{\text{V, E}\})}, \tag{8}$$

where $\{V, E\}$ represents a triangle mesh, V and E represent the whole vertices and egdes of the triangle mesh $\{V, E\}$ respectively, non-scale laplace $D_{\text{lap}}(\{V, E\})$ measures the non-scale smoothness of the surface ($\{V, E\}$), ave-edge ($\{\text{V, E}\}$) represents the average edge length of the triangle mesh $\{V, E\}$, $\mathcal{N}(\mathbf{v})$ represents the set of neighbors of $\mathbf{v}$, and $\omega_{\mathbf{v},\mathbf{u}}$ represents area weights of the discrete laplace operator as defined in the [58].

$$D_{\text{eik}}(\Phi) = \frac{1}{\#\mathcal{X}} \sum_{\mathbf{p}\in\mathcal{X}} (\|\nabla_{\mathbf{x}}\Phi(\mathbf{p})\|_2 - 1)^2, \tag{9}$$

$$D_{\text{grad}}(\Phi) = \frac{1}{\#\mathcal{X}} \sum_{\mathbf{p}\in\mathcal{X}} \|\nabla_{\mathbf{x}}\Phi(\mathbf{p}) - \nabla_{\mathbf{x}}\Phi(\mathbf{p} + \delta)\|_2, \tag{10}$$

where $D_{\text{eik}}(\Phi)$ and $D_{\text{grad}}(\Phi)$ measure the satisfaction of the eikonal constraint and the continuity of the gradient of geometry representation Φ, respectively, $\mathcal{X}$ is the uniformly sampling point set in the bounding volume Ω, we set $\#\mathcal{X}$ 20000 in our experiments, Φ represents neural function, and $\delta = \{10^{-3}, 10^{-3}, 10^{-3}\}$ represents a displacement.

For the neural multi-view image reconstruction task, similar to [68, 55, 39, 69], we choose 512^3 as the resolution of volume in the Marching Cubes algorithm to extract the final geometry shape. We use the formal "surface" evaluation script from the DTU dataset [23] to evaluate our 3D surface reconstruction results. Besides, we choose a higher resolution 2048^3 to show the details of reconstructed results. Furthermore, we synthesize the novel view images by performing the volume or surface rendering to the reconstructed geometry under the given novel view. And for all methods, we report the PSNR using pixels located in the predefined masks between the rendered images and the reference images to measure the reconstruction quality of the light field.

4.3. Implement Details

Comparison methods. For the point cloud reconstruction task, we compare our approach with several state-of-the-art neural point cloud reconstruction methods, including IGR [50], SIREN [46], SplinePE [56], EG3D [8] and

[1] http://graphics.stanford.edu/data/3Dscanrep/

[2] https://www.turbosquid.com/

SAPE [21]. And we conduct all experiments with Pytorch [41] on the GeForce RTX 3090 (24 GB memory) graphics except SplinePE [56], whose official implementation conflicts with the GeForce RTX 3090 graphics. Thus, we conduct the experiments of the SplinePE on the Tesla V100 (32 GB memory) graphics. For SAPE [21], we refer to the point cloud reconstruction framework used in IGR [20], and then reproduce it for the point cloud reconstruction task. For EG3D [8], we reproduce the second-order gradient of the tri-plane position encoding operator on the Pytorch and embed it as a learnable position encoding in the framework of the geometry reconstruction task. To obtain a fair comparison with IGR [20], we add a Fourier-position encoding [50] layer with encoding dimension 6 into their geometry network in the official implementation. For simplicity, we denote the three reproduced methods as EG3D*, SAPE*, and IGR(PE), respectively.

For the multi-view image reconstruction task, we compare our approach with IDR [68], NeuS [55], UNISURF [39], VolSDF [67], and NeRF [35]. Similarly, we conducted all the experiments on the GeForce RTX 3090 graphics based on their official implementations.

Our approach. The hash encoding $\text{enc}(\theta_h, \mathbf{x})$ in our work has 16 layers, and the dimension of each grid feature in each layer is 2. The frequency domain dimension of Fourier position encoding $\text{enc}_{\text{ff}}(\mathbf{x})$ is 6. In addition, we modify the network initialization strategy in [1] and then apply it to initialize our geometry network. Following NeuS [55], we use the hierarchical sampling strategy to sample points on the rays in the multi-view image reconstruction task. Then we use the mean of the SDF of sampling points as a threshold to eliminate some invalid sampled points on each ray.

5. Experiments

5.1. Neural Point Cloud Reconstruction

Architecture. We use a 4-layer MLP with Softplus activation functions to represent the geometry network in all neural point cloud reconstruction experiments. Each hidden layer contains 128 units, and the parameters of the Softplus activation functions are set as $\beta = 100$. Specifically, the input of the first layer is the Fourier-position encoding of spatial location ($\text{enc}_{\text{ff}}(\mathbf{x})$), and the input of the third layer is the concatenation of the hash encoding of spatial location ($\text{enc}(\theta_h, \mathbf{x})$) and the output of the second hidden layer.

Hyperparameter. We train our neural network for 1500 iterations with reconstruction loss Eq. (2). Following IGR [20], for each iteration, we sample 65536 points from the input unorganized 3D point cloud and 65536 points from the bounding volume uniformly to optimize our network, respectively. In our objective loss functions in the Eq. (2), we set β_0 and balance parameters λ_1, λ_2, λ_3 as 100, 0.1, 0.05, and 1, respectively.

Table 1. Comparisons of reconstruction quality and computational cost of different methods for point cloud reconstruction. The CD in the first row is defined as the Chamfer-L_2 Distance (Eq. (7)).

	IGR (PE)	SPAE*	SplinePE	SIREN	EG3D*	Ours
CD($\times 10^{-6}$)	4.41	1.97	2.04	1.48	1.02	**0.59**
Time(min)	18	50	1200	60	150	**6**
#Iter($\times 10^2$)	500	500	200	400	500	**15**
Mem(GB)	6.6	11.31	6.8	13	6.2	**6.0**

Comparison of point cloud reconstruction. We measure the reconstruction quality with the Chamfer distance metric and record the required training time for each method. As shown in Table 1, in these selected challenging cases, our approach achieves great accuracy improvement and a 10 times increase in training speed. In addition, we conduct the qualitative comparisons on the "Thai statut" case in Fig. 6. For IGR and SIREN, modifying the activation function and the spatial position encoding based on the ReLU-MLPs can improve the accuracy to a certain extent. But for IGR, the limited expressiveness of sinusoid functions results in a generally smooth reconstruction result with fewer details and a slow convergence speed. SAPE and SplinePE adopt a novel progressive learning strategy from low-frequency information to high-frequency information, which can reconstruct more details but take a lot of time for training. EG3D introduces great adaptability for the reconstructed objects to high-frequency and low-frequency information via the learnable position encoding, so its reconstruction results have rich details. However, as shown in the blue rectangle of Fig. 6, EG3D is prone to noise in high-frequency details due to a lack of proper initialization and low continuity of the explicit discrete representation. Like EG3D, our hash-encoded geometry representation employs a learnable positional encoding with multiple resolution layers, which is then compressed into a hash table with learnable features. Due to our geometry representation and initialization design, our approach achieves better performance.

5.2. Neural Multi-view Image Reconstruction

Architecture. In the neural multi-view image reconstruction task, our geometry network architecture is roughly the same as described in Sec. 3.2.1, except for the number of MLP layers, here is 6. In addition, our light field c for color prediction is modeled by an MLP with 4 hidden layers, each containing 256 units. And the inputs of the light field are the Fourier position encoding of view direction $\mathbf{v}$, gradient $\mathbf{n}$, and the geometry feature vector output from the connected layer of the geometry network.

Hyperparameter. In our experiments, we set balance parameters α, β and γ in Eq. (5) as 0.1, 0.1 and 5×10^{-4}.

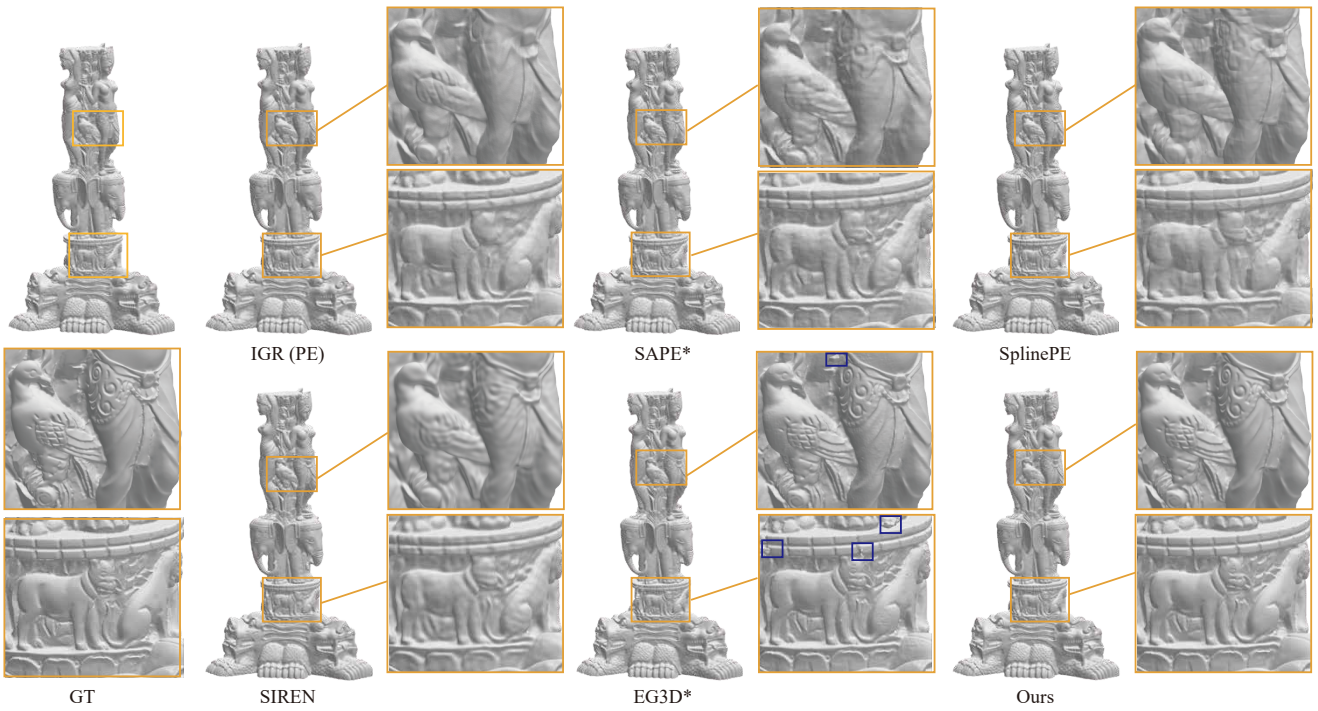

Figure 6. Comparisons of different geometry representations on the neural point cloud reconstruction. The method with $*$ in the figure is our implementation with the same settings in SIREN [46]. The yellow box is an enlarged area of details, which can be more directly observed. The accuracy of the reconstruction of our method has a significant advantage. The blue boxes are some anomalous results of the reconstruction.

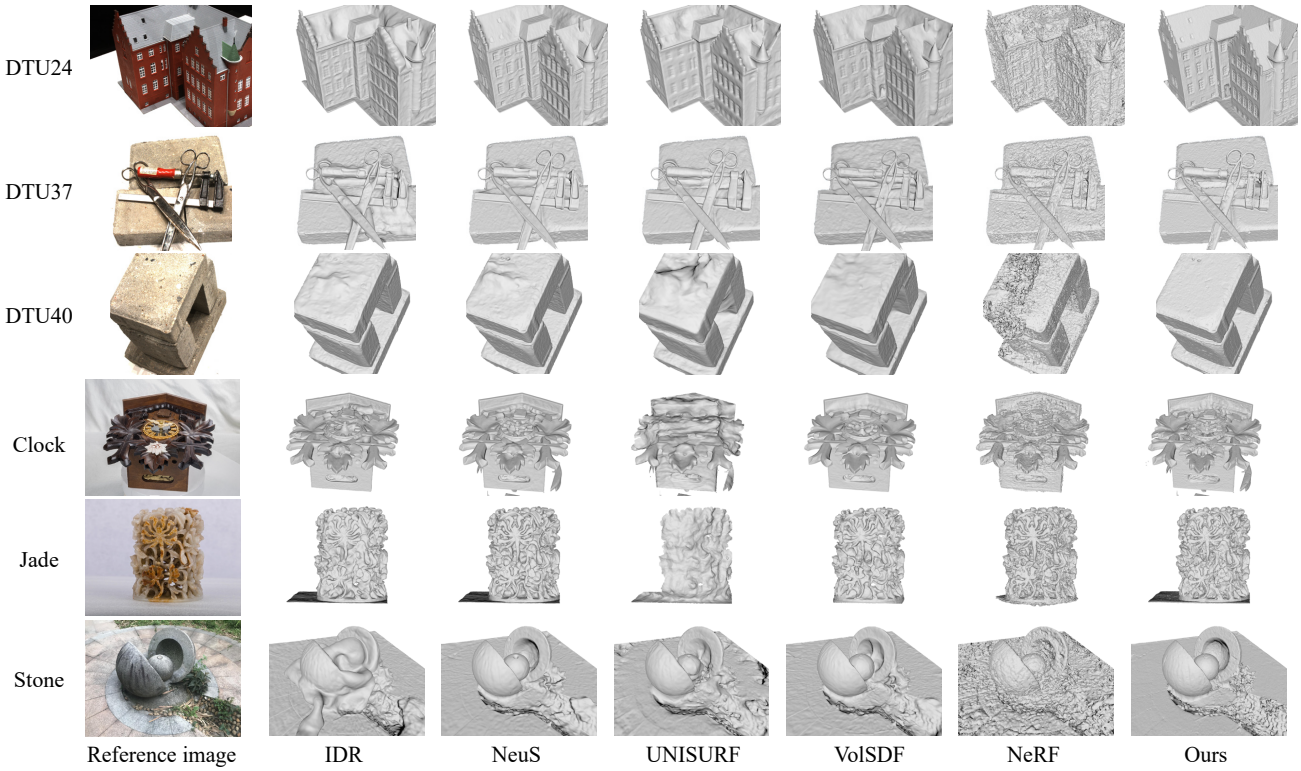

Figure 7. Comparison of different methods for multi-view image reconstruction on DTU and MVS-blender dataset.

Table 2. Comparison results of multi-view reconstruction on the DTU dataset.

ScanID	IDR		NeuS		UNISURF		VolSDF		NeRF		Ours	
	CD ↓	PSNR ↑	CD ↓	PSNR ↑	CD ↓	PSNR ↑	CD ↓	PSNR ↑	CD ↓	PSNR ↑	CD ↓	PSNR ↑
scan24	1.63	23.29	0.83	26.73	1.32	25.51	1.14	24.16	1.90	26.02	**0.64**	**29.67**
scan37	1.87	21.36	0.98	23.42	1.36	23.26	1.26	21.29	1.60	**24.78**	**0.90**	24.23
scan40	0.63	24.39	0.56	26.32	1.72	25.79	0.81	24.93	1.85	27.83	**0.40**	**28.86**
scan55	0.48	22.96	0.37	24.92	0.44	25.53	0.49	22.78	0.58	26.36	**0.35**	**29.86**
scan63	**1.04**	23.22	1.13	30.49	1.35	28.12	1.25	28.99	2.28	**31.48**	**1.04**	30.97
scan65	0.79	23.94	**0.59**	32.55	0.79	30.38	0.70	28.68	1.27	31.92	0.72	**32.99**
scan69	0.77	20.34	**0.60**	29.03	0.80	28.78	0.72	27.67	1.47	**30.46**	0.71	28.53
scan83	1.33	21.87	1.45	**33.51**	1.49	30.78	**1.29**	31.50	1.67	33.31	1.39	33.45
scan97	1.16	22.95	0.95	**27.65**	1.37	25.93	1.18	22.57	2.05	26.43	**0.90**	27.49
scan105	0.76	22.71	0.78	31.20	0.89	30.83	**0.70**	30.56	1.07	31.07	0.76	**31.63**
scan106	0.67	22.81	0.52	32.13	0.59	30.68	0.66	29.50	0.88	32.26	**0.47**	**33.53**
scan110	**0.90**	21.26	1.43	28.85	1.47	29.03	1.08	27.11	2.53	28.19	1.01	**29.77**
scan114	0.42	25.35	0.36	28.42	0.46	28.06	0.42	26.60	1.06	29.08	**0.36**	**29.40**
scan118	0.51	23.54	**0.45**	34.97	0.59	32.31	0.61	28.60	1.15	34.86	0.49	**36.58**
scan122	0.53	27.98	**0.45**	34.81	0.62	33.03	0.55	31.60	0.96	32.95	0.57	**35.91**
mean	0.90	23.20	0.77	29.66	1.02	28.53	0.86	27.11	1.49	29.8	**0.72**	**30.85**

5.2.1 Comparison of Multi-view Image Reconstruction

On the multi-view image reconstruction task, for each scene, IDR and NeuS reconstruct the foreground object only with a given mask, while NeRF, UNISURF, and VolSDF reconstruct the entire 3D scene. We evaluate the geometry reconstruction quality with the Chamfer distance metric on the DTU dataset. And we refer directly to the existing results of IDR, NeuS, NeRF, UNISURF, and VolSDF, which were reported in the original paper [55] and [67], respectively. The corresponding scores are reported in Table 2. The results show that our approach outperforms other baseline methods in these selected scenes. In addition, we fairly and comprehensively compare the time and memory consumption in training between our approach and baseline methods in Table 4. As shown in Table 4, our approach requires less memory and training faster.

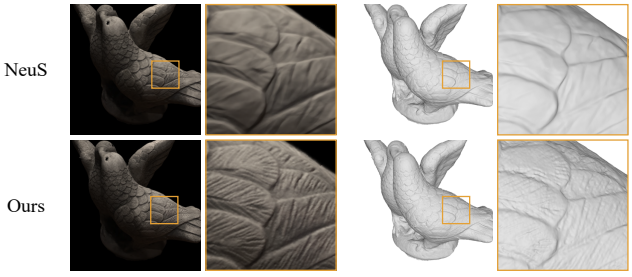

Figure 8. Comparison with NeuS on DTU106. Left and right are renderings of the reconstructed light field and geometry.

We also conduct qualitative comparisons on the DTU and BlendedMVS datasets in Fig. 7, respectively. As Fig. 7 illustrates, NeuS, IDR, VolSDF, and UNISURF perform poorly in textureless areas of scenes DTU40. And because

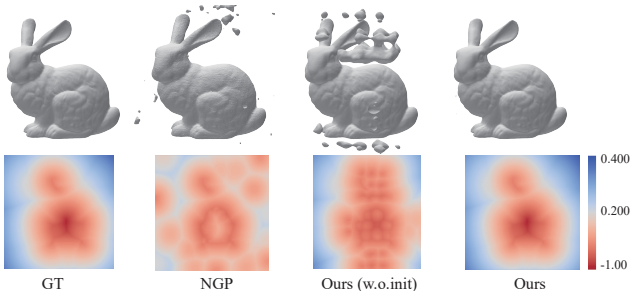

Figure 9. Ablation study of geometry network designs on point cloud reconstruction task. Left to right: GT, the baseline of hash encoding (NGP), Added with Fourier position encoding, and the geometry network is properly initialized (Ours). First row: the extracted zero level set via Marching Cubes. Second row: a slice view of SDFs. The values of the eikonal constraints D_{eik} are 0.012, 0.006, and 0.00091 from the second column to the fourth column. The values of D_{grad} are 0.22, 0.10, 0.005 from second column to fourth column. In addition, the inside of the NGP's result has a double-layer structure.

Figure 10. Ablation study of geometry network designs on neural multi-view image reconstruction task.

Table 3. Ablation of the network structure of geometry representation on our processed point cloud datasets (contained 19 original models before upsampling). "S" and "T" represent the original point cloud and the reconstructed mesh, respectively. "S"2"T" and "T"2"S" represents the D_{scd}("S", "T") and D_{scd}("T", "S"), respectively. The values in the "S"2"T" and "T"2"S" are multiplied by 10^{-7} and 10^{-4}, respectively.

	"S"2"T"↓	"T"2"S" ↓	D_{lap}	D_{eik} ↓	D_{grad} ↓
NGP	4.06	1.64	0.166	0.023	0.27
Our (w.o.init)	3.962	1.63	0.145	0.009	0.11
Our (w.init)	**3.242**	**0.758**	**0.129**	**0.002**	**0.06**

of the lack of direct constraints on the volume density, the geometry reconstructed by NeRF is relatively rough and with obvious noise. Compared with other baselines, our approach has the ability to reconstruct more geometry details, which is evident in the result of scene DTU24 and bmvs-clock (tick value) and bmvs-stone (pebbles and flowers) in Fig. 7. We further compare our approach with NeuS [55] on scene DTU106. Focusing on the bird's detailed feathers in Fig. 8, our approach reconstructs high-frequency details consistent with the multi-view images.

Figure 11. Illustration of the location of geometric features in the rendering module of the neural multi-view framework. The middle is the result of the usual feature extraction location, and the right is the result by changing the extraction location to the connected layer.

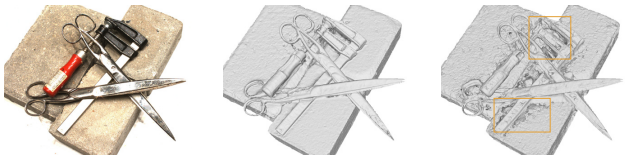

Figure 12. Ablation study of the initialization strategy on the multi-view reconstruction.

Table 4. Comparisons of the computational cost of different methods for multi-view image reconstruction.

	IDR	NeuS	UNISURF	VolSDF	NeRF	Ours
Time(h)	5.2	7.2	21	9	9.1	**1.8**
#Rays	2048	512	1024	1024	1024	**512**
#Iter ($\times 10^3$)	128	300	400	128	200	**120**
Mem(GB)	6.5	7	6	13.8	8	**5**

5.2.2 Comparison of Novel View Synthesis Results

The novel view synthesis task is a direct application of our neural multi-view image reconstruction framework. After using the existing neural volume rendering technique under a given novel view, we can obtain the synthesis image in this view. Thus, we also measure the PSNR between the reference images and the synthesis images rendered from the reconstructed light field (only focusing on the foreground mask region) on the DTU in Table 2. It indicate that the quality of our reconstructed images achieves significant improvements compared with other neural multi-view image reconstruction methods. We also perform better than the results with the state-of-the-art method of novel view synthesis, i.e., NeRF.

Focusing on the novel view synthetic task, we compare our approach with NeuS, NeRF on scenes DTU55 ("rabbit") qualitatively in Fig. 13. From Fig. 13, we observe that, compared to the state-of-the-art geometry reconstruction method NeuS, our novel view synthetic results are more realistic and detailed, achieving better results than NeRF, as shown in the fined texture of "rabbit". Even in some places, such as the yellow light of the background of "rabbits", NeRF has evident new novel synthesis artifacts due to incorrect geometry learned, but our approach still performs well. In addition, our training time is five times shorter than the original NeRF.

5.3. Ablation Study

Network structure of geometry representation. In this section, we conduct a thorough ablation analysis for the effectiveness of our geometry representation. As we analyzed in the Sec. 3.1, the geometry representation, which only uses learnable hash features as a positional encoding (NGP), is unable to satisfy the constraints of the eikonal equation well under weak supervision, and it is prone to falling into a non-smooth solution in the point cloud reconstruction and multi-view image reconstruction tasks. The Table 3 shows our comparative numerical results on network structure design, indicates that the fitting accuracy of NGP on the point cloud is very high, but the reconstructed surface is very rough. Specially, the non-scale Laplace metric $D_{\text{lap}}(\{V_{\text{ori}}, E_{\text{ori}}\})$ of original mesh is 0.132, is closest to the result reconstructed by our method. In particular, with regard to the continuous metric of the neural implicit representation in the whole space, the eikonal constraint D_{eik} and the spatial gradient continuity D_{grad} of the NGP are both poor. The reason for the discontinuity of the gradient of the geometry representation in the space is that the features bound on these explicit grids are discontinuous, as shown in Fig. 4. To alleviate the above issue, we first introduce a Fourier position encoding [50] as input to the first layer of the geometry network to encode 3D position information in space, improving the continuity of geometry representations. We then use the

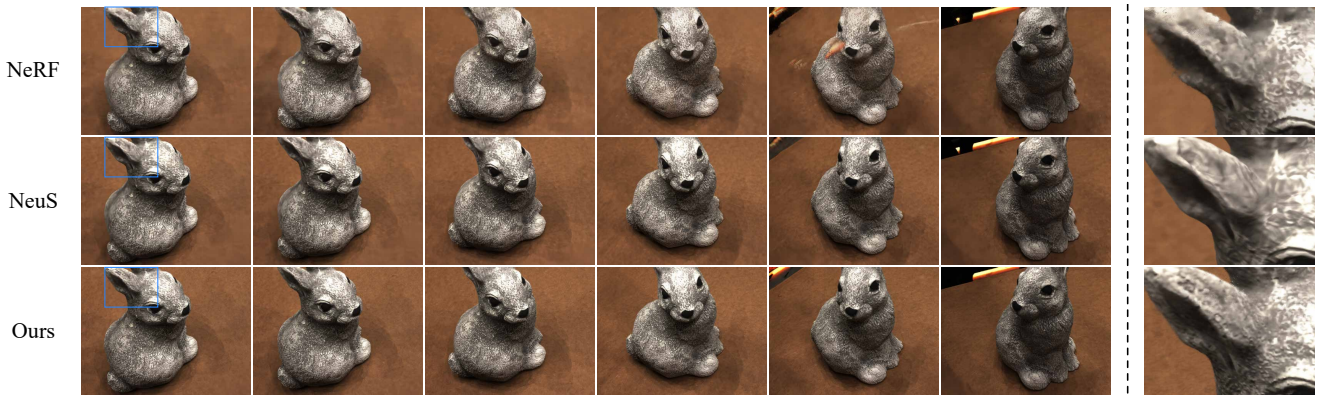

Figure 13. Comparison of different methods on novel view synthesis. It shows the continuous interpolation of randomly selected two views to synthesize an unknown view image for DTU55. From the first column to the sixth column are the synthesis image of different novel views. The last column is a partial enlargement of the blue box in the first column of images.

Table 5. Ablation study of whether hash encoding only encodes geometry (SDF) of multi-view reconstruction on the DTU dataset. w.o. M Fea and w. M Fea represents the geometry representation of the extracted position of the geometry feature without modification and modification, respectively.

	NGP		Ours (w.o. M Fea)		Ours (w. M Fea)	
	CD ↓	PSNR ↑	CD ↓	PSNR ↑	CD ↓	PSNR ↑
mean	0.94	29.45	0.86	29.89	**0.72**	**30.85**

network initialization of the SAL [1] into our network to better satisfy eikonal constraints. Fig. 9 and Fig. 10 show our geometry representation with the Fourier positional encoding and the learnable hash positional encoding together results in smoother reconstruction results.

Framework of neural volume rendering. In most existing neural rendering frameworks, such as [68, 35, 55], estimating the light field of each input 3D point $\mathbf{p} \in \mathbb{R}^3$ requires the geometry feature and density, which are generated by the geometry network. Usually, the geometry feature is a part of the output of the last network layer. However, it is easy for the hash encoding to participate in learning color attributes in the multi-view geometry reconstruction task, which makes the optimization of geometry density ambiguous. The main reasons are its strong expressiveness and faster-to-learn properties. As indicated in Table 5, encoding only geometric properties can improve geometry reconstruction and quality rendering. Thus, in our geometry representation, we extract the geometry feature from the connected layer of our geometry network, where the hash encoding is exclusively used to represent the geometry (SDF) of objects. It avoids the expression ambiguity and achieves more consistent geometry results with multi-view images, so our reconstructed geometry is more consistent with the rendered image, as shown in Fig. 11.

Initialization of the geometry network. Regarding the problem of neural geometry reconstruction, reasonable initialization plays a critical role in the optimization of the network, as discussed in the NeuS and IDR. The Table 3 shows the initialization of the geometry representation is a crucial module. And the Fig. 12 shows that the geometry network without a proper initialization performs poorly in some complex areas, such as somewhere with highlights or rapidly changing geometry.

6. Discussion

Although our method moderately improves the reconstruction accuracy of textureless regions for the neural multi-view reconstruction tasks, it still needs to be improved to solve the problem of reconstructions with shadows and highlights. This is mainly due to the fact that the rendering representation cannot be perfectly decoupled from the geometry representation. Thus, it is crucial to design a more powerful rendering representation, and decoupling method to solve these problems, such as [51, 47]. In addition, it is a topic worth exploring to introduce it to the problems of dynamic geometry reconstruction and single-image based geometry reconstruction to improve speed and precision, such as [5, 24, 14].

7. Conclusion

In this work, we have proposed a hash encoding-based neural geometry representation, and applied this representation to recover the surface's signed distance function from the input point clouds or multi-view images. In the geometry network, we further combine our method with the low-dimensional Fourier positional encoding and network initialization in SAL [1]. Meanwhile, in the multi-view reconstruction task, we redesign the extraction way of geometry features to avoid the singularity between geometries and color values. Extensive experimental results also have demonstrated that our method can achieve at least 10 times speedup in the point cloud-based surface reconstruction task, and significantly improve the accuracy and efficiency of multi-view reconstruction.

Acknowledgements

This research was partially supported by the National Natural Science Foundation of China (No.62122071, No.62272433), the Fundamental Research Funds for the Central Universities (No. WK3470000021), and Alibaba Group through Alibaba Innovation Research Program (AIR). The authors thank Peng Wang (HKU) for providing the script for evaluation of multi-view reconstruction, and Xueying Wang (USTC) and Yuxin Yao (USTC) for their helps on the paper writing.

References

[1] M. Atzmon and Y. Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 2565–2574, 2020. 2, 5, 8, 12

[2] M. Atzmon and Y. Lipman. SALD: sign agnostic learning with derivatives. In *9th International Conference on Learning Representations, ICLR 2021*, pages 1–14, 2021. 3

[3] M. Baorui, H. Zhizhong, L. Yu-Shen, and Z. Matthias. Neural-pull: Learning signed distance functions from point clouds by learning to pull space onto surfaces. In *International Conference on Machine Learning (ICML)*, volume 139, pages 7246–7257, 2021. 3

[4] M. Berger, A. Tagliasacchi, L. M. Seversky, P. Alliez, G. Guennebaud, J. A. Levine, A. Sharf, and C. T. Silva. A survey of surface reconstruction from point clouds. *Comput. Graph. Forum*, pages 301–329, 2017. 3

[5] H. Cai, W. Feng, X. Feng, Y. Wang, and J. Zhang. Neural surface reconstruction of dynamic scenes with monocular rgb-d camera. In *Thirty-sixth Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 12

[6] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3d objects with radial basis functions. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 67–76, 2001. 3

[7] R. Chabra, J. E. Lenssen, E. Ilg, T. Schmidt, J. Straub, S. Lovegrove, and R. Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 608–625, 2020. 2

[8] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. De Mello, O. Gallo, L. J. Guibas, J. Tremblay, S. Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16123–16133, 2022. 3, 7, 8

[9] R. Chen, S. Han, J. Xu, and H. Su. Point-based multi-view stereo network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1538–1547, 2019. 3

[10] Z. Chen, A. Tagliasacchi, T. Funkhouser, and H. Zhang. Neural dual contouring. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 3

[11] Z. Chen and H. Zhang. Learning implicit fields for generative shape modeling. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5939–5948, 2019. 2

[12] X. Cheng, Y. Zhong, M. Harandi, Y. Dai, X. Chang, H. Li, T. Drummond, and Z. Ge. Hierarchical neural architecture search for deep stereo matching. In *Advances in Neural Information Processing Systems*, volume 33, pages 22158–22169, 2020. 3

[13] J. Chibane, A. Mir, and G. Pons-Moll. Neural unsigned distance fields for implicit function learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 21638–21652, 2020. 2

[14] Z. Deng, Y. Liu, H. Pan, W. Jabi, J. Zhang, and B. Deng. Sketch2pq: Freeform planar quadrilateral mesh design via a single sketch. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 12

[15] P. Erler, P. Guerrero, S. Ohrhallinger, N. J. Mitra, and M. Wimmer. Points2Surf: Learning implicit surfaces from point clouds. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 108–124, 2020. 3, 7

[16] Y. Furukawa and C. Hernández. Multi-view stereo: A tutorial. *Found. Trends Comput. Graph. Vis.*, 9(1-2):1–148, 2015. 3

[17] Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, 2010. 3

[18] X. Gao, C. Zhong, J. Xiang, Y. Hong, Y. Guo, and J. Zhang. Reconstructing personalized semantic facial nerf models from monocular video. *ACM Transactions on Graphics (TOG)*, 41(6):1–12, 2022. 3

[19] K. Genova, F. Cole, A. Sud, A. Sarna, and T. Funkhouser. Local deep implicit functions for 3d shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4857–4866, 2020. 2

[20] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman. Implicit geometric regularization for learning shapes. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, pages 3789–3799, 2020. 1, 2, 3, 8

[21] A. Hertz, O. Perel, R. Giryes, O. Sorkine-Hornung, and D. Cohen-Or. Sape: Spatially-adaptive progressive encoding for neural optimization. *Advances in Neural Information Processing Systems*, 34:8820–8832, 2021. 1, 3, 8

[22] J. Huang, S.-S. Huang, H. Song, and S.-M. Hu. Di-fusion: Online implicit 3d reconstruction with deep priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8932–8941, 2021. 4

[23] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs. Large scale multi-view stereopsis evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413, 2014. 7

[24] B. Jiang, Y. Hong, H. Bao, and J. Zhang. Selfrecon: Self reconstruction your digital avatar from monocular video. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 5595–5605, 2022. 12

[25] C. M. Jiang, A. Sud, A. Makadia, J. Huang, M. Nießner, and T. Funkhouser. Local implicit grid representations for

3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6000–6009, 2020. 2

[26] A. Kar, C. Häne, and J. Malik. Learning a multi-view stereo machine. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 365–376, 2017. 3

[27] M. M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, volume 256 of *SGP '06*, pages 61–70, 2006. 3

[28] M. Klingensmith, I. Dryanovski, S. S. Srinivasa, and J. Xiao. Chisel: Real time large scale 3d reconstruction onboard a mobile device using spatially hashed signed distance fields. In *Robotics: science and systems*, pages 0–14, 2015. 3

[29] F. Langguth, K. Sunkavalli, S. Hadap, and M. Goesele. Shading-aware multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 469–485, 2016. 3

[30] R. Liang, H. Sun, and N. Vijaykumar. Coordx: Accelerating implicit neural representation with a split MLP architecture. *arXiv preprint arXiv:2201.12425*, 2022. 2

[31] S. Liu, H. Guo, H. Pan, P. Wang, X. Tong, and Y. Liu. Deep implicit moving least squares functions for 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1788–1797, 2021. 2

[32] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *ACM Trans. Graph. (SIGGRAPH).*, pages 163–169, 1987. 1, 7

[33] J. N. P. Martel, D. B. Lindell, C. Z. Lin, E. R. Chan, M. Monteiro, and G. Wetzstein. Acorn: Adaptive coordinate networks for neural scene representation. *ACM Trans. Graph. (SIGGRAPH)*, pages 58–72, 2021. 3

[34] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4460–4470, 2019. 2

[35] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, volume 12346, pages 405–421, 2020. 1, 3, 8, 12

[36] T. Müller, A. Evans, C. Schied, and A. Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022. 2, 3, 4, 5

[37] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3501–3512, 2020. 3

[38] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger. Real-time 3d reconstruction at scale using voxel hashing. *ACM Trans. Graph.*, 32(6):169:1–169:11, 2013. 3

[39] M. Oechsle, S. Peng, and A. Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 5569–5579, 2021. 3, 7, 8

[40] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 165–174, 2019. 2, 3, 6

[41] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 8024–8035, 2019. 8

[42] R. Peng, R. Wang, Z. Wang, Y. Lai, and R. Wang. Rethinking depth estimation for multi-view stereo: A unified representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8635–8644, 2022. 3

[43] S. Peng, M. Niemeyer, L. Mescheder, M. Pollefeys, and A. Geiger. Convolutional occupancy networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 523–540, 2020. 2

[44] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. A. Hamprecht, Y. Bengio, and A. C. Courville. On the spectral bias of neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 5301–5310, 2019. 1

[45] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm. Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 501–518, 2016. 3

[46] V. Sitzmann, J. Martel, A. Bergman, D. Lindell, and G. Wetzstein. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems*, 33:7462–7473, 2020. 1, 2, 6, 7, 9

[47] M. Suhail, C. Esteves, L. Sigal, and A. Makadia. Light field neural rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 8259–8269, 2022. 12

[48] J. Sun, Y. Xie, L. Chen, X. Zhou, and H. Bao. NeuralRecon: Real-time coherent 3D reconstruction from monocular video. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15598–15607, 2021. 3

[49] T. Takikawa, J. Litalien, K. Yin, K. Kreis, C. Loop, D. Nowrouzezahrai, A. Jacobson, M. McGuire, and S. Fidler. Neural geometric level of detail: Real-time rendering with implicit 3D shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11358–11367, 2021. 3

[50] M. Tancik, P. P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. T. Barron, and R. Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:7537–7547, 2020. 1, 2, 7, 8, 11

[51] K. Tiwary, T. Klinghoffer, and R. Raskar. Towards learning neural representations from shadows. In *Computer Vision - ECCV 2022 - 17th European Conference*, pages 300–316, 2022. 12

[52] C. Walder, B. Schölkopf, and O. Chapelle. Implicit surfaces with globally regularised and compactly supported basis functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 273–280, 2006. 3

[53] F. Wang, S. Galliani, C. Vogel, P. Speciale, and M. Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 14194–14203, 2021. 3

[54] F. Wang, S. Galliani, C. Vogel, P. Speciale, and M. Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14194–14203, June 2021. 3

[55] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 27171–27183, 2021. 2, 4, 6, 7, 8, 10, 11, 12

[56] P. Wang, Y. Liu, Y. Yang, and X. Tong. Spline positional encoding for learning 3d implicit signed distance fields. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 1091–1097, 2021. 1, 3, 7, 8

[57] X. Wang, Y. Guo, Z. Yang, and J. Zhang. Prior-guided multi-view 3d head reconstruction. *IEEE Trans. Multim.*, 24:4028–4040, 2022. 3

[58] M. Wardetzky, S. Mathur, F. Kälberer, and E. Grinspun. Discrete laplace operators: no free lunch. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing, Barcelona, Spain, July 4-6, 2007*, volume 257, pages 33–37, 2007. 7

[59] Y. Wei, S. Liu, Y. Rao, W. Zhao, J. Lu, and J. Zhou. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 5590–5599, 2021. 3

[60] Y.-P. Xiao, Y.-K. Lai, F.-L. Zhang, C. Li, and L. Gao. A survey on deep geometry learning: From a representation perspective. *Computational Visual Media*, pages 113–133, 2020. 2

[61] H. Xu and J. Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1959–1968, 2020. 3

[62] H. Yang, H. Zhu, Y. Wang, M. Huang, Q. Shen, R. Yang, and X. Cao. Facescape: a large-scale high quality 3d face dataset and detailed riggable 3d face prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 598–607, 2020. 7

[63] J. Yang, W. Mao, J. M. Alvarez, and M. Liu. Cost volume pyramid based depth inference for multi-view stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 4748–4760, 2022. 3

[64] Z. Yang, Z. Ren, Q. Shan, and Q. Huang. Mvs2d: Efficient multi-view stereo via attention-driven 2d convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8574–8584, June 2022. 3

[65] Y. Yao, Z. Luo, S. Li, T. Shen, T. Fang, and L. Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5525–5534, 2019. 3

[66] Y. Yao, Z. Luo, S. Li, J. Zhang, Y. Ren, L. Zhou, T. Fang, and L. Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1787–1796, 2020. 7

[67] L. Yariv, J. Gu, Y. Kasten, and Y. Lipman. Volume rendering of neural implicit surfaces. In *Thirty-Fifth Conference on Neural Information Processing Systems*, pages 4805–4815, 2021. 3, 8, 10

[68] L. Yariv, Y. Kasten, D. Moran, M. Galun, M. Atzmon, B. Ronen, and Y. Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:2492–2502, 2020. 1, 3, 7, 8, 12

[69] J. Zhang, Y. Yao, and L. Quan. Learning signed distance field for multi-view surface reconstruction. *The IEEE International Conference on Computer Vision (ICCV)*, pages 6505–6514, 2021. 3, 7