MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

FreBIS: Frequency-Based Stratification for Neural Implicit Surface Representations

Sawada, Naoko; Miraldo, Pedro; Lohit, Suhas; Marks, Tim K.; Chatterjee, Moitreya

TR2025-074 June 07, 2025

Abstract

Neural implicit surface representation techniques are in high demand for advancing technologies in augmented reality/virtual reality, digital twins, autonomous navigation, and many other fields. With their ability to model object surfaces in a scene as a continuous function, such techniques have made remarkable strides recently, especially over classical 3D surface reconstruction methods, such as those that use voxels or point clouds. However, these methods struggle with scenes that have varied and complex surfaces principally because they model any given scene with a single encoder network that is tasked to capture all of low through high-surface frequency information in the scene simultaneously. In this work, we propose a novel, neural implicit surface representation approach called FreBIS to overcome this challenge. FreBIS works by stratifying the scene based on the frequency of surfaces into multiple frequency levels, with each level (or a group of levels) encoded by a dedicated encoder. Moreover, FreBIS encourages these encoders to capture complementary information by promoting mutual dissimilarity of the encoded features via a novel, redundancy-aware weighting module. Empirical evaluations on the challenging BlendedMVS dataset indicate that replacing the standard encoder in an off-the-shelf neural surface reconstruction method with our frequency-stratified encoders yields significant improvements. These enhancements are evident both in the quality of the reconstructed 3D surfaces and in the fidelity of their renderings from any viewpoint.

IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPR) 2025

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

FreBIS: Frequency-Based Stratification for Neural Implicit Surface Representations

Naoko Sawada[1,2] Pedro Miraldo[1] Suhas Lohit[1] Tim K. Marks[1] Moitreya Chatterjee[1]

[1]Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA

[2]Information Technology R&D Center, Mitsubishi Electric Corporation, Kanagawa, Japan

Sawada.Naoko@df.MitsubishiElectric.co.jp,{miraldo,slohit,tmarks,chatterjee}@merl.com

Abstract

Neural implicit surface representation techniques are in high demand for advancing technologies in augmented reality/virtual reality, digital twins, autonomous navigation, and many other fields. With their ability to model object surfaces in a scene as a continuous function, such techniques have made remarkable strides recently, especially over classical 3D surface reconstruction methods, such as those that use voxels or point clouds. However, these methods struggle with scenes that have varied and complex surfaces principally because they model any given scene with a single encoder network that is tasked to capture all of low through high-surface frequency information in the scene simultaneously. In this work, we propose a novel, neural implicit surface representation approach called FreBIS to overcome this challenge. FreBIS works by stratifying the scene based on the frequency of surfaces into multiple frequency levels, with each level (or a group of levels) encoded by a dedicated encoder. Moreover, FreBIS encourages these encoders to capture complementary information by promoting mutual dissimilarity of the encoded features via a novel, redundancy-aware weighting module. Empirical evaluations on the challenging BlendedMVS dataset indicate that replacing the standard encoder in an off-the-shelf neural surface reconstruction method with our frequency-stratified encoders yields significant improvements. These enhancements are evident both in the quality of the reconstructed 3D surfaces and in the fidelity of their renderings from any viewpoint.

1. Introduction

While a picture is worth a thousand words, yet 2D image understanding methods miss out on critical details, including depth cues and occluded structures, driving research into techniques for reconstructing complete 3D surfaces from images. Approaches for reconstructing 3D surfaces find wide use in a broad swathe of applications, including Augmented Reality (AR), Virtual Reality (VR), robotics, archaeology and allow users to easily create 3D content.

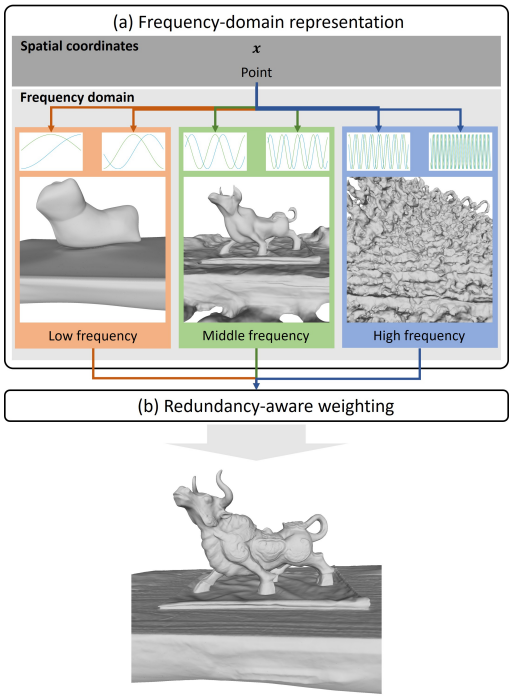


Figure 1. Overview of FreBIS: (a) *Frequency-domain Representation:* FreBIS works by mapping the input point coordinate to the frequency domain and encoding it via three frequency-band encoders – one each for low, middle, and high. (b) *Redundancy-aware Weighting*: This module computes weights that indicate the importance of the three encoded features according to the dissimilarity of each to the other two. These weights are then used to combine the encoded features. The 3D surface is reconstructed by decoding the combined feature into a SDF value.

Conventional methods for the task of 3D scene recon-

struction leverage explicit representations, such as voxels [3, 4, 38] and point clouds [2, 11, 12, 37], where the granularity of the voxels or the 3D points determines the resolution of the reconstructed mesh, thereby limiting the quality of the reconstruction. Neural implicit surface representation methods overcome this challenge by learning continuous functions to model the 3D surfaces, including signed distance functions (SDF) [43, 52] and occupancy fields [31]. These implicit representations can encode 3D geometries at infinite resolution and reduce memory requirements, thereby realizing high-fidelity 3D surface reconstruction from 2D images.

Prior works on neural implicit surface representation [43, 52] and their variants can reconstruct 3D surfaces with high details. However, their ability to simultaneously represent the correct shape of complex surfaces, while capturing their fine details is limited. This is primarily because they employ a single encoder network that attempts to capture all the various surface frequencies present in the scene (possibly from a very low to a very high one) simultaneously.

In this paper, we propose *Frequency-Based Stratification for Neural Implicit Surface Representation* (FreBIS) – a novel approach to neural implicit surface representation, where multiple encoder networks are specialized to encode different frequency bands so that each encoder can capture complementary information about the scene, allowing FreBIS to effectively learn low– through high–frequency information simultaneously. In practice, FreBIS employs three encoders dedicated to capturing information in the low–, middle–, and high–frequency bands, respectively, from the scene which is then assimilated and decoded by a single decoder network to estimate the SDF value and a RGB feature vector which encodes the color information, as shown in Fig. 1 (a). Thus, instead of a unified latent feature encoding, features corresponding to different frequency bands can be derived from three different encoders. To effectively combine the disparate information learned by the different encoders, FreBIS introduces a novel *redundancy-aware weighting* module, as shown in Fig. 1 (b). Given the different feature encodings, this module estimates normalized importance scores for each of them and uses them as weights to combine the encodings to derive a unified representation. Subsequently, a decoder module decodes this unified representation to predict the SDF value and RGB feature, corresponding to a 3D point in the scene.

FreBIS makes it possible to recover high-quality surfaces of 3D scenes that contain various levels of detail. Additionally, it provides a flexible mechanism to combine the stratified encoders with any off-the-shelf decoder backbones. Empirical evaluations on the challenging BlendedMVS [50] dataset show that our strategy of frequency-based stratification results in improved reconstruction of 3D surfaces while better preserving the fidelity of their renderings from any given viewpoint.

In summary, the key contributions of our work are as follows:

- A novel, frequency-based 3D surface representation method (called FreBIS) that works by stratifying the scene into non-overlapping frequency bands.
- FreBIS employs a *redundancy-aware weighting* module that encourages the stratified encoders to capture complementary information by promoting mutual dissimilarity of the encoded features.
- Empirical evaluations demonstrate the effectiveness of FreBIS on the challenging BlendedMVS [50] dataset.

2. Related Work

Early multi-view surface reconstruction methods: Multi-view stereo (MVS) technologies have traditionally been used to recover 3D shapes from multiple RGB images of a scene. Classical MVS approaches can be classified into voxel-based [3, 4, 9, 17, 21, 38], point-cloud-based [2, 11, 13, 32, 37], and mesh-based [8, 22, 40] methods. While promising, these methods suffer from quantization artifacts, and noisy or disconnected reconstructed points. Moreover, the quality of the recovered surfaces is voxel/point-resolution-dependent. We, on the other hand, learn an implicit, continuous function, resulting in smoother, more detailed, and robust reconstructions.

Neural implicit surface representation approaches: Neural implicit surface representation techniques represent a 3D surface as a continuous function defined by a neural network, such as SDF or occupancy function. Early methods [30, 51] achieved 3D surface reconstruction from multi-view images by leveraging object mask priors. The advent of NeRF [29] heralded a paradigm shift in this field, integrating implicit surface representation methods with radiance-field-based approaches. For instance, VolSDF [52] and NeuS [43] transform SDF into a differentiable volume density, enabling 3D surface reconstruction solely from 2D images while also permitting a rendering of the mesh from any viewpoint. UNISURF [31] formulates occupancy-based implicit surface representation and radiance field in a unified framework. Different from previous approaches [30, 51], they eliminate the need for object masks. These methods have paved the way for newer neural implicit surface representation methods. Several variants built upon VolSDF [52] and NeuS [43] enhance the input feature encoder, venturing beyond a simple Multilayer Perceptron (MLP), to be capable of capturing the fine details of the scene [14, 44, 46]. NeuralWarp [7] and GeoNeuS [10] add explicit multi-view geometry constraints to enforce photo consistency and depth consistency across views. Other approaches [1, 15, 27, 33, 35, 41, 42, 54] try to enhance the robustness and details of the representation by integrating priors, such as monocular depth and

normal estimates, in addition to RGB images. Recent works [23, 26, 45] have leveraged multi-resolution grid structures to accelerate training and boost the accuracy of the reconstructed surfaces. Some extensions of these approaches [25, 48, 49] adapt neural implicit surface representations to object-compositional scenes. Despite the noteworthy strides made by prior methods, to the best of our knowledge, none have looked at the efficacy of stratifying the scene based on surface frequencies as a cue towards achieving improved 3D surface reconstruction and rendering. Additionally, our approach is complementary to many of these approaches and can be integrated with them for possibly additive performance gains.

Neural radiance field (NeRF): Some prior works extract explicit 3D surfaces from radiance field representations of 3D scenes obtained via Neural Radiance Field (NeRF) [29]. MobileNeRF [5], NeRF2Mesh [39], NeRFMeshing [36], and BakedSDF [53] extract an explicit textured mesh from a trained NeRF model, by having a separate network (in addition to the NeRF model) which predicts the SDF value of a point, given a feature encoding of the point and the viewing direction obtained from the NeRF network. However, these methods require a fully trained NeRF to begin with, which can be prohibitively slow to train.

Gaussian splatting (GS): 3D Gaussian Splatting (3DGS) [19] has emerged as a fast and accurate novel view synthesis method, where scenes are modeled as sets of 3D Gaussians, which are splatted in any novel viewing direction to obtain the color. To leverage 3DGS for 3D surface representation, SuGaR [16] introduces a new regularization term to encourage Gaussians to scatter on surfaces, while Gaussian Surfels [6] and 2DGS [18] flatten 3D Gaussians into 2D ellipses. SplatSDF [24] and 3DGSR [28] fuse SDF and 3DGS to achieve both high accuracy and efficiency. While these methods offer fast training and rendering, and some of them achieve surface reconstructions that are comparable in quality to the best implicit methods, however, they result in high memory consumption. Additionally, some of these methods are sensitive to noise and thereby lack robustness.

3. Background

3.1. Positional Encoding

Positional encodings have assumed a critical role in neural implicit models, such as NeRF [29] or VolSDF [52]. In these models, positional encoding is used to map the input coordinates into vectors in the frequency domain. Such a transformation injects ordering information into the input and enables the encoder network to capture the scene frequency information. Eq. 1 shows a prototypical definition of the positional encoding, as used in neural implicit networks, such as VolSDF [52].

$$\gamma(\boldsymbol{x}) = (\sin(2^0\boldsymbol{x}), \cos(2^0\boldsymbol{x}), \cdots, \sin(2^{N-1}\boldsymbol{x}), \cos(2^{N-1}\boldsymbol{x})), \quad (1)$$

where $\boldsymbol{x} \in \mathbb{R}^3$ denotes the coordinate of the input point, while a total of N frequencies are used for the encoding.

3.2. Neural Volume Rendering

Neural volume rendering approaches, such as NeRF [29], have achieved tremendous success at the task of novel view rendering of 3D scenes. These models learn an implicit representation of the scene via a mapping from any 3D point $\boldsymbol{x}$ in the scene, encoded using positional encodings, to a volume density $\sigma(\boldsymbol{x}) \in [0, 1]$ and a RGB color $\boldsymbol{c}(\boldsymbol{x}) \in \mathbb{R}^3$, given a viewing direction $\boldsymbol{v}$. Such a mapping is typically implemented via a MLP network. The novel view rendering of the scene is generated pixel by pixel by casting a ray ($\boldsymbol{r}(t) = \boldsymbol{o} + t\boldsymbol{v}, t \geq 0, t \in \mathbb{R}$) emanating from the position of the camera center $\boldsymbol{o}$ in the viewing direction $\boldsymbol{v}$.

Using volume rendering, each pixel color $\hat{\boldsymbol{C}}_{\boldsymbol{p}}(\boldsymbol{r})$ at pixel $\boldsymbol{p}$ is calculated as the accumulation of all color contributions along the ray $\boldsymbol{r}$, weighed by the accumulated transmittance $T(t)$ from the near bound t_{near} upto t, where the transmittance is defined as: $T(t) = \exp(\int_{t_{\text{near}}}^{t} \sigma(\boldsymbol{r}(s))ds)$ and opacity of the point being captured by the density $\sigma(\boldsymbol{r}(t)) \in [0, 1]$). More formally, the pixel color $\hat{\boldsymbol{C}}_{\boldsymbol{p}}(\boldsymbol{r})$ is given by the following equation:

$$\hat{\boldsymbol{C}}_{\boldsymbol{p}}(\boldsymbol{r}) = \int_{t_{\text{near}}}^{t_{\text{far}}} T(t)\sigma(\boldsymbol{r}(t))\boldsymbol{c}(\boldsymbol{r}(t))dt, \quad (2)$$

where $t_{\text{near}}, t_{\text{far}}$ denote the nearest and farthest points that could be sampled along the ray $\boldsymbol{r}$.

3.3. Signed Distance Function (SDF)

Signed Distance Function (SDF) has recently emerged as a very effective tool for representing 3D surfaces [43, 52]. An SDF is a continuous function that denotes the distance of any point in 3D to the closest surface in the scene. The zero-level set of an SDF implicitly represents the scene's outer surface, points inside objects in the scene have a negative SDF value, while those that are outside have a positive SDF value. In practice, the SDF network is often instantiated by a MLP [43, 52]. To train the SDF network without ground truth 3D mesh information, prior works, such as VolSDF [52] and NeuS [43], convert SDF values into a density field and use it to synthesize RGB images from the viewing direction of the training views, via volume rendering. Such a design allows for the SDF model to derive a training signal by comparing ground truth RGB images with those estimated by the volume rendering step.

More concretely, given a scene $\Omega \subset \mathbb{R}^3$, the volume density at a point $\boldsymbol{x}$ is derived from its SDF value $d_\Omega(\mathbf{x}) \in$

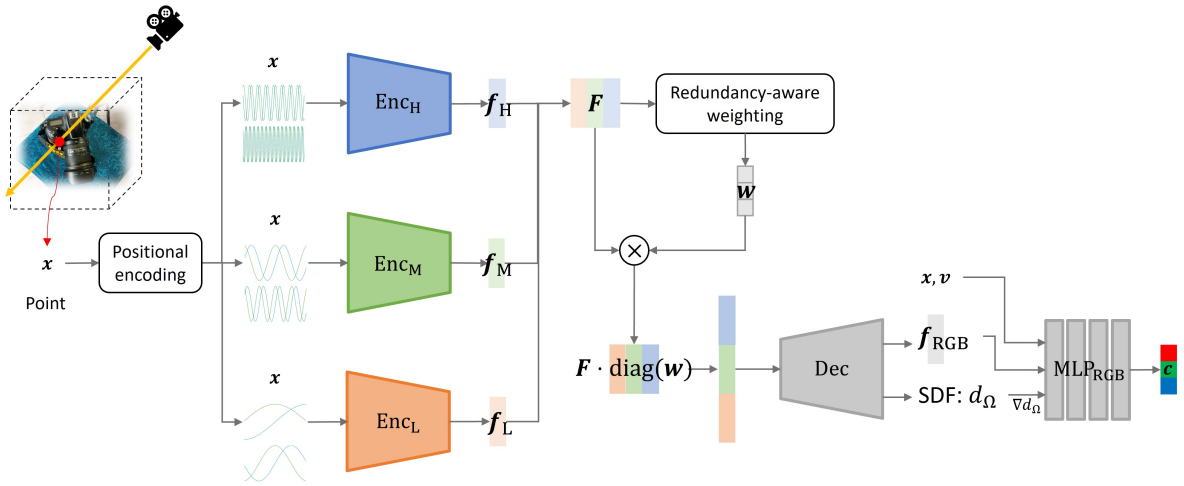


Figure 2. FreBIS framework: Given an input 3D point $\boldsymbol{x}$, positional encoding maps it to the frequency domain. The output of the positional encoding is then encoded into latent feature vectors corresponding to low–, middle–, and high–frequencies, respectively ($\boldsymbol{f}_{\mathrm{L}}, \boldsymbol{f}_{\mathrm{M}}, \boldsymbol{f}_{\mathrm{H}}$) by leveraging our frequency-stratified encoders $\mathrm{Enc}_{\mathrm{L}}, \mathrm{Enc}_{\mathrm{M}}$, and $\mathrm{Enc}_{\mathrm{H}}$. The *redundancy-aware weighting* module takes the concatenated feature encodings ($\boldsymbol{F} = [\boldsymbol{f}_{\mathrm{L}}, \boldsymbol{f}_{\mathrm{M}}, \boldsymbol{f}_{\mathrm{H}}]$) and decides on the relative importance of these features according to the dissimilarity of each to the other two, estimating a normalized weight vector ($\boldsymbol{w}$). Finally, the weighted features ($\boldsymbol{F} \cdot \mathrm{diag}(\boldsymbol{w})$) are passed to a decoder Dec to extract a SDF value d_Ω and an appearance feature $\boldsymbol{f}_{\mathrm{RGB}}$ for the point $\boldsymbol{x}$. $\mathrm{MLP}_{\mathrm{RGB}}$ predicts $\boldsymbol{x}$'s color given the appearance feature, point position $\boldsymbol{x}$, view direction $\boldsymbol{v}$, and point normal ∇d_Ω.

$[-1, 1]$ (estimated from a neural network), using the following equation [52]:

$$\sigma(\boldsymbol{x}) = \begin{cases} \frac{\alpha}{2}\exp(\frac{\mathrm{d}_\Omega(\boldsymbol{x})}{\beta}) & \text{if } d_\Omega(\boldsymbol{x}) \leq 0, \\ \alpha\left(1 - \frac{1}{2}\exp(\frac{-\mathrm{d}_\Omega(\boldsymbol{x})}{\beta})\right) & \text{if } d_\Omega(\boldsymbol{x}) > 0, \end{cases} \tag{3}$$

where $\alpha, \beta > 0$ are learnable parameters. Volume rendering can then be used to render a novel view image by using this volume density to weigh the color (RGB) value at the point $\boldsymbol{x}$, as estimated by a separate color prediction network.

4. Proposed Approach

In this section, we introduce FreBIS, our novel approach for neural implicit surface representation. FreBIS reconstructs the 3D surface of a scene and can render it from any viewpoint, given a series of posed 2D images of the 3D scene. FreBIS leverages our novel, frequency-stratified encoders to encode an input point in 3D space and decode it, using any (off-the-shelf [52]) decoder, to obtain the SDF value of the point as well as a feature, encoding its appearance. This appearance feature can then be decoded to obtain the view-dependent color, given the desired viewing direction. Fig. 2 shows an overview of our proposed approach.

4.1. Frequency-domain Representation

Prior approaches for neural implicit surface representation struggle to simultaneously represent the correct shape of complex surfaces while capturing their fine details. This is primarily because they employ a single encoder network for the input point that attempts to capture all the various surface frequencies present in the scene (possibly from a very low to a very high one) simultaneously. This typically leads to a bias towards capturing the low–frequencies while ignoring the high-frequency details. In our framework, we overcome this challenge by employing three encoders (low–frequency encoder ($\mathrm{Enc}_{\mathrm{L}}$), middle–frequency encoder ($\mathrm{Enc}_{\mathrm{M}}$), and high–frequency encoder ($\mathrm{Enc}_{\mathrm{H}}$)) that convert the input to features corresponding to different frequency bands, instead of a single encoding, to make the model more expressive and capable of representing surfaces with a wide variety of frequencies.

To transform the spatial coordinates into the frequency domain, we encode the input point using positional encodings (see Sec 3.1) and route it to the appropriate frequency encoder based on its associated frequency. For instance, to distribute 6 frequency levels (i.e., $N = 6$) equally among the three encoders, we assign the lowest frequencies $\{2^0, 2^1\}$ to $\mathrm{Enc}_{\mathrm{L}}$, the middle frequencies $\{2^2, 2^3\}$ to $\mathrm{Enc}_{\mathrm{M}}$, while those for two highest frequencies $\{2^4, 2^5\}$ are routed to $\mathrm{Enc}_{\mathrm{H}}$.

Each encoder converts the positional encodings into corresponding 256-D latent feature vectors ($\boldsymbol{f}_{\mathrm{L}}, \boldsymbol{f}_{\mathrm{M}}, \boldsymbol{f}_{\mathrm{H}}$). Such stratification of the frequency representation bolsters the model's capability to model the shape of the surface of the scene while capturing its details.

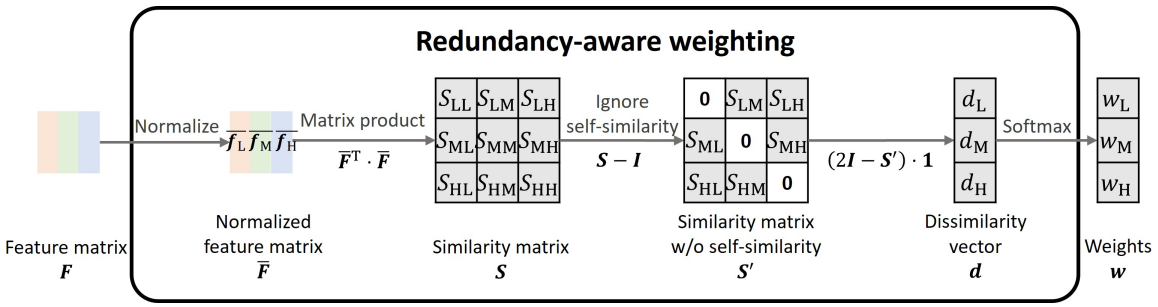


Figure 3. Redundancy-aware weighting module: The redundancy-aware weighting module takes the encoded frequency features and predicts a normalized importance score, following the pipeline shown in the figure, assigning a higher weight to the frequency encoding that is least similar to the other two and vice-versa.

4.2. Redundancy-aware Weighting

For the encoder capacity to be maximally utilized, encouraging dissimilarity between the learned representations of the three encoders is essential. To promote such behavior and effectively combine the complementary information learned by the different encoders, we propose a novel, *redundancy-aware weighting* module, as shown in Fig. 3. This module estimates normalized importance scores for each of the three different feature encodings and uses them as weights to combine the encodings to derive a unified representation. A higher score is assigned to the feature encoding which is the most dissimilar to the other two and vice-versa, promoting the learning of complementary feature encodings between the encoders.

At the outset, the module concatenates features from the three encoders into a matrix, which we denote as $\boldsymbol{F} = [\boldsymbol{f}_{\mathrm{L}}, \boldsymbol{f}_{\mathrm{M}}, \boldsymbol{f}_{\mathrm{H}}] \in \mathbb{R}^{256\times 3}$, which is then normalized per column, based on the L_2 norm, denoted by $\bar{\boldsymbol{F}}$. Next, a similarity matrix $\boldsymbol{S}$ is computed by taking the matrix product of $\bar{\boldsymbol{F}}^T$ and $\bar{\boldsymbol{F}}$, as shown in Eq. 4.

$$\boldsymbol{S} = \bar{\boldsymbol{F}}^T \cdot \bar{\boldsymbol{F}} = \begin{pmatrix} S_{\mathrm{LL}} & S_{\mathrm{LM}} & S_{\mathrm{LH}} \\ S_{\mathrm{ML}} & S_{\mathrm{MM}} & S_{\mathrm{MH}} \\ S_{\mathrm{HL}} & S_{\mathrm{HM}} & S_{\mathrm{HH}} \end{pmatrix}, \tag{4}$$

where each entry, $\in [-1, 1]$. To compute the dissimilarity information from $\boldsymbol{S}$, we remove the diagonal entries, which capture the self-similarity, as shown:

$$\boldsymbol{S}' = \boldsymbol{S} - \boldsymbol{I} = \begin{pmatrix} 0 & S_{\mathrm{LM}} & S_{\mathrm{LH}} \\ S_{\mathrm{ML}} & 0 & S_{\mathrm{MH}} \\ S_{\mathrm{HL}} & S_{\mathrm{HM}} & 0 \end{pmatrix}, \tag{5}$$

where $\boldsymbol{I}$ denotes the 3×3 identity matrix. Next, a dissimilarity vector $\boldsymbol{d}$ is computed, as shown in Eq. 6:

$$\boldsymbol{d} = (2\boldsymbol{I} - \boldsymbol{S}') \cdot \mathbf{1}, \tag{6}$$

where $\mathbf{1}$ is $[1, 1, 1]^T$. Finally, the weight vector $\boldsymbol{w}$ for $\boldsymbol{F}$ is given by Eq. 7.

$$\boldsymbol{w} = \mathrm{Softmax}\left(\frac{\boldsymbol{d}}{\tau}\right), \tag{7}$$

where the $\mathrm{Softmax}(\cdot)$ function rescales elements in a vector to be in the range $[0, 1]$ and sum to 1, and τ is a temperature parameter that controls the smoothness of the softmax distribution. The default value of τ is set to 0.5. The redundancy-aware encoder features are then computed by: $\boldsymbol{F} \cdot \mathrm{diag}(\boldsymbol{w})$.

4.3. Decoder

The redundancy-weighted encoder features can be decoded to obtain the SDF value of the point and its appearance feature. This is undertaken via a decoder (Dec), often instantiated by a MLP network, which takes the flattened redundancy-weighted feature vector as an input and estimates the SDF value and an appearance feature vector ($\boldsymbol{f}_{\mathrm{RGB}}$) as an output. $\boldsymbol{f}_{\mathrm{RGB}}$ is then used to derive the view-dependent RGB color for the point $\boldsymbol{x}$. The final RGB-color value of the point is obtained by feeding $\boldsymbol{f}_{RGB}$, the point coordinates, and the viewing direction to the color prediction network $\mathrm{MLP}_{\mathrm{RGB}}$, akin to volume rendering methods discussed in Sec. 3.2.

4.4. Loss Function

We train FreBIS using the following set of losses: (i) the photometric loss $\mathcal{L}_{\mathrm{RGB}}$ and (ii) the Eikonal loss $\mathcal{L}_{\mathrm{Eikonal}}$. The final loss is given by:

$$\mathcal{L} = \mathcal{L}_{\mathrm{RGB}} + \lambda \mathcal{L}_{\mathrm{Eikonal}}, \tag{8}$$

where $\lambda \in \mathbb{R}, \lambda > 0$. $\mathcal{L}_{\mathrm{RGB}}$ and $\mathcal{L}_{\mathrm{Eikonal}}$ in Eq. 8 are defined as follows:

$$\mathcal{L}_{\mathrm{RGB}} = ||\boldsymbol{C}_{\boldsymbol{p}} - \hat{\boldsymbol{C}}_{\boldsymbol{p}}(\boldsymbol{r})||_1, \tag{9}$$

$$\mathcal{L}_{\mathrm{Eikonal}} = (||\nabla d_{\Omega}(\boldsymbol{z})|| - 1)^2. \tag{10}$$

In Eq. 9, $\boldsymbol{C}_{\boldsymbol{p}}$ is the ground truth color at pixel $\boldsymbol{p}$, and $\hat{\boldsymbol{C}}_{\boldsymbol{p}}(\boldsymbol{r})$ is the rendered color (obtained using Eq. 2). In Eq. 10, $d_{\Omega}(\boldsymbol{z})$ is an approximated SDF value for the sampled point $\boldsymbol{z}$.

	Method (no. of parameters)	Doll	Egg	Head	Angel	Bull	Robot	Dog	Bread	Camera	**Mean**
PSNR(↑)	VolSDF [52] (0.5M)	25.43	27.23	26.94	30.28	26.18	26.39	28.44	**31.18**	22.96	27.23
	Scaled-up VolSDF (1.4M)	26.07	27.15	26.62	30.37	26.08	25.07	28.32	29.44	23.02	26.90
	Ours (1.4M)	**26.22**	**27.48**	**27.29**	**30.52**	**26.33**	**26.69**	**28.56**	30.22	**23.08**	**27.38**
SSIM(↑)	VolSDF [52] (0.5M)	0.911	0.943	0.959	0.989	0.970	0.957	0.950	**0.988**	0.928	0.955
	Scaled-up VolSDF (1.4M)	0.925	0.943	0.956	**0.990**	0.970	0.946	0.949	0.980	0.929	0.954
	Ours (1.4M)	**0.928**	**0.946**	**0.961**	**0.990**	**0.971**	**0.962**	**0.952**	0.983	**0.930**	**0.958**
LPIPS(↓)	VolSDF [52] (0.5M)	0.041	0.032	0.017	0.007	0.021	0.032	0.027	**0.006**	0.045	0.025
	Scaled-up VolSDF (1.4M)	**0.035**	0.032	0.018	**0.006**	0.021	0.043	0.028	0.011	0.045	0.027
	Ours (1.4M)	**0.035**	**0.030**	**0.015**	**0.006**	**0.020**	**0.030**	**0.026**	0.009	**0.044**	**0.024**

Table 1. Quantitative results for 9 scenes from the BlendedMVS dataset. The best score in each scene is shown in **bold**

.

N_L, N_M, N_H	Doll	Egg	Head	Angel	Bull	Robot	Dog	Bread	Camera
2,2,2	26.22	27.48	**27.29**	30.52	26.33	26.69	28.56	30.22	23.08
1,2,3	26.01	27.38	27.00	30.37	26.35	26.58	28.47	30.85	22.95
1,3,2	**26.23**	27.44	27.07	30.44	26.25	26.76	28.42	29.73	**23.11**
2,1,3	25.96	**27.50**	26.95	29.97	26.28	26.67	28.50	29.46	22.92
2,3,1	26.00	27.03	27.15	30.35	26.04	26.51	28.75	**31.80**	23.09
3,1,2	26.02	27.34	27.05	30.50	**26.38**	**26.84**	28.67	30.17	23.01
3,2,1	26.18	26.85	27.16	**30.56**	26.14	26.61	**28.82**	31.73	23.01

Table 2. Quantitative comparison of scene rendering performance of various assignments of frequencies to each encoder, in terms of PSNR. **Bold** texts denote the best score in each scene.

5. Experiments

We evaluate the performance of FreBIS for the tasks of viewpoint-based scene rendering and 3D surface reconstruction across various complex, real-world scenes, comparing it against appropriate baselines.

5.1. Experimental Setup

Implementation details: We implemented FreBIS in Pytorch [34] and performed experiments on an NVIDIA A40 GPU with 48GB RAM. All three encoders of FreBIS have 6 layers with 256 dimensions per layer, while the decoder has 2 layers with 256 dimensions per layer. We set the frequency level $N = 6$ and distribute it evenly to each encoder, i.e., each encoder deals with two frequencies. Additionally, we also concatenate the original input point coordinates to the input of each frequency encoder. The training loss (Eq. 4) is computed with $\lambda = 0.1$. We set the initial learning rate to 0.005 for all the parameters in the model, which are optimized by the Adam optimizer [20]. The color network $\mathrm{MLP}_{\mathrm{RGB}}$ has 4 layers with 256 dimensions.

Dataset: We evaluate our method quantitatively and qualitatively on the BlendedMVS dataset [50], which consists of various object-centric real-world scenes with backgrounds. Following the protocol of prior work [52], we selected the same 9 scenes for evaluation. Each scene is composed of 31 to 144 multi-view images with a resolution of 768×576.

Evaluation metrics: We evaluate the performance of competing methods for the task of view-dependent scene rendering using standard metrics such as peak signal-to-noise ratio (PSNR) measured in dB, structural similarity index measure (SSIM) [47], and learned perceptual image patch similarity (LPIPS) [55]. Besides, we also qualitatively evaluate the quality of the reconstructed 3D mesh (since the ground truth mesh is not available for this dataset).

Baselines: FreBIS is flexible in design and can work with any off-the-shelf decoder. For our experiments, we use the popular VolSDF decoder [52]. Given this setup, to evaluate the effectiveness of our method, we compare our approach against VolSDF [52] and a customized, challenging baseline called Scaled-up VolSDF. The Scaled-up VolSDF is an adaptation of VolSDF, where the number of parameters is increased to be roughly the same as Ours, for fair comparison. This baseline has a surface encoding network with 8 layers with 427 dimensions each, instead of the typical 256 dimensions in VolSDF.

5.2. Results

Table 1 summarizes the results of quantitative comparisons of our method against VolSDF and Scaled-up VolSDF. FreBIS achieves the highest PSNR and SSIM and the lowest LPIPS score for all scenes in the dataset, except for the simpler, less textured scene – the *Bread* scene, registering gains of up to 2% on SSIM over the Scaled-up VolSDF baseline on an overall assessment.

Qualitative comparisons of rendered images on the *Doll*, *Bull*, and *Robot* scenes are presented in Fig. 4. As illustrated in these images, FreBIS considerably improves rendering quality, especially the fine details of objects. Qualitative comparisons on the reconstructed meshes are presented in Fig. 5. In particular, the reconstructed surfaces by FreBIS have higher fidelity and are better at preserving the details, e.g., bands on the *Doll*'s cloth, the *Bull*'s saddle, the *Robot*'s gun and face. Moreover, we also notice that the eyeballs of the *Doll* are inappropriately reconstructed as concave surfaces by VolSDF and Scaled-up VolSDF, while FreBIS does

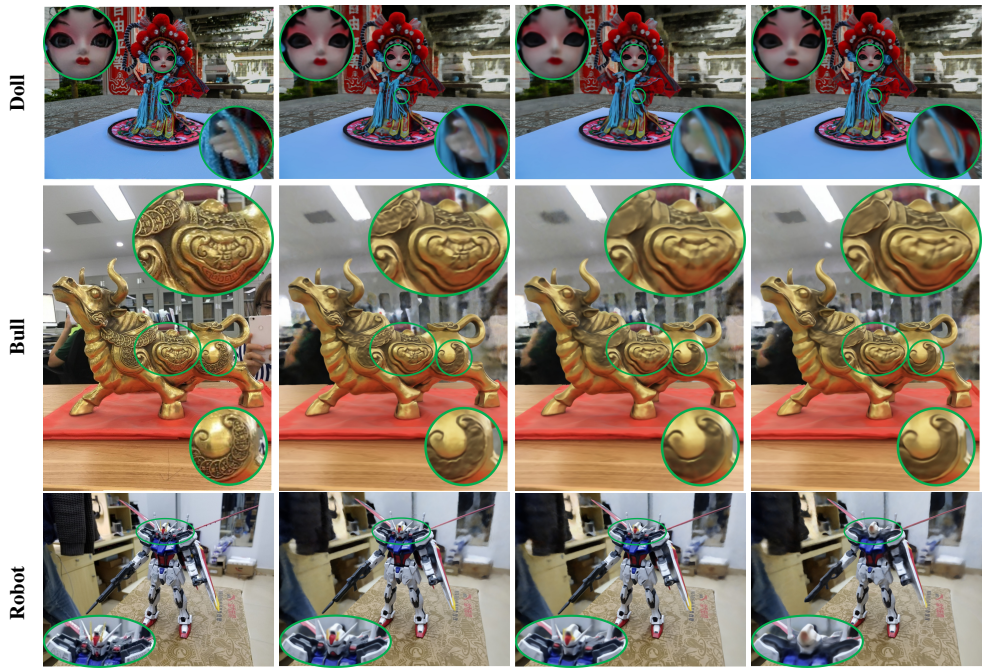


Figure 4. Qualitative comparison of viewpoint-based scene rendering on the BlendedMVS dataset.

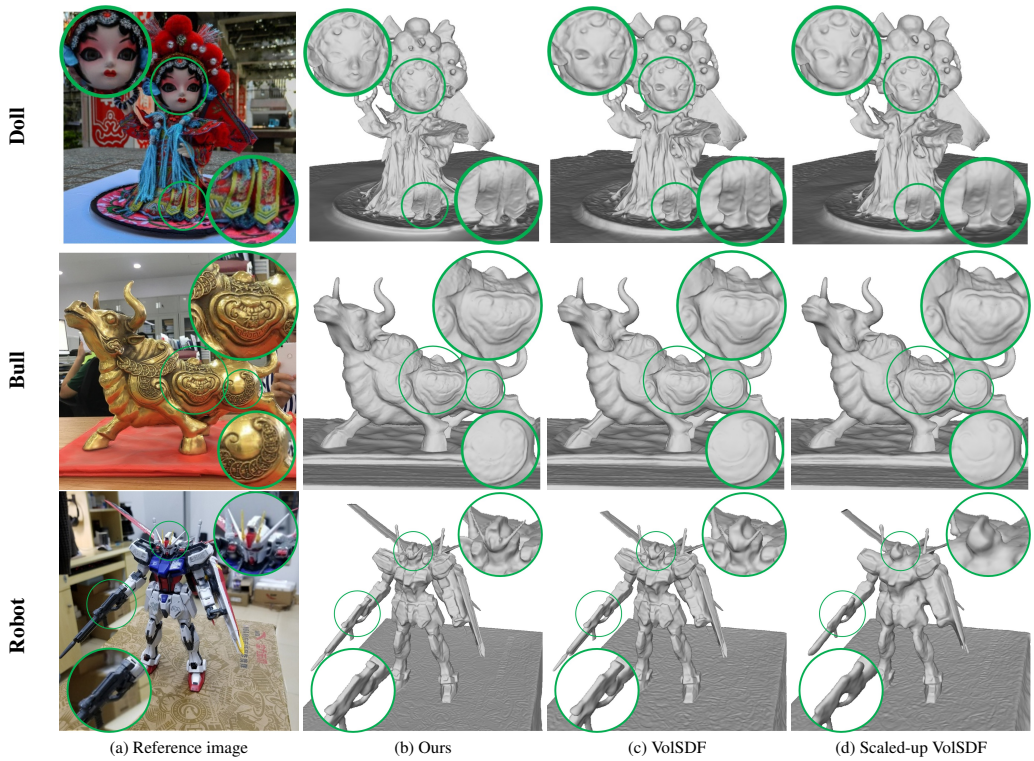


Figure 5. Qualitative comparison of surface reconstruction quality for the BlendedMVS dataset.

a better job of the reconstruction. We see that FreBIS outperforms VolSDF and Scaled-up VolSDF both in terms of scene rendering and surface reconstruction quality. These results attest to the effectiveness of our method and show

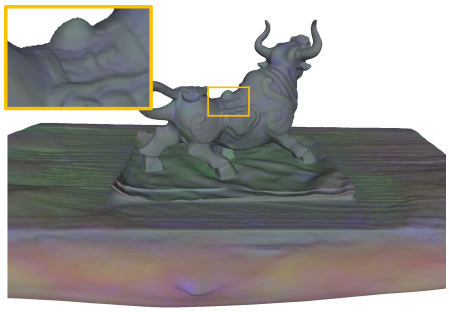
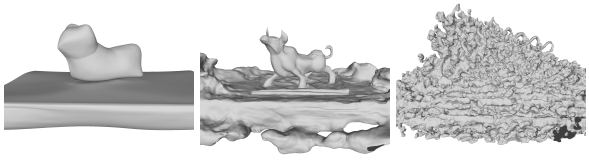


Figure 6. Visualization of norms of weighted feature vectors, $\boldsymbol{F} \cdot \mathrm{diag}(\boldsymbol{w})$. The norms of low–, middle–, and high–frequency features are visualized as red, green, and blue channels, respectively.



(a) Low frequency (b) Middle frequency (c) High frequency

Figure 7. Reconstructed meshes for each frequency band.

that the gain in performance cannot simply be attributed to scaling up the number of parameters. More visualizations are present in the supplementary.

To verify that appropriate frequency bands are used in each region and that the encoders learn complementary features, we visualize the norms of weighted features ($\boldsymbol{F} \cdot \mathrm{diag}(\boldsymbol{w})$), that are redundancy-aware, for each frequency band and the quality of meshes obtained for each frequency band.

Norms of weighted features for each frequency-band: Fig. 6 shows the reconstructed mesh of the *Bull* scene, where the vertex color denotes the norm of weighted features. For this visualization, the low–, middle–, and high–frequency features are mapped to red, green, and blue channels, respectively. Note also that the norm is scaled to $[0.4, 1.0]$ for visibility. We see that high–frequency information (blue) is more dominant in regions with finer details, e.g., decorative carving, whereas low–frequency information (red) is mainly used for unobserved and interpolated areas where details are missing. Our encoders successfully distinguish between smooth and rough surface regions and model them with different frequency bands.

Surface reconstructions for each frequency domain: To examine whether each encoder learns complementary features, we decode the output of each frequency encoder independently and visualize the results. Figs. 7a, 7b, and 7c are meshes reconstructed from feature vectors $\boldsymbol{f}_{\mathrm{L}}$, $\boldsymbol{f}_{\mathrm{M}}$, $\boldsymbol{f}_{\mathrm{H}}$, respectively, for the *Bull* scene. As shown in Fig. 7, the low–frequency mesh captures the global structure of the scene well, the middle–frequency mesh gets the rough shape of objects and some details, while the high–frequency mesh captures the fine details. These results show that the encoders successfully learn complementary, frequency-dependent features.

	(a) Scaled-up VolSDF	(b) Ours w/o redundancy-aware weighting	(c) Ours
PSNR (↑)	28.32	28.31	**28.56**
SSIM (↑)	0.949	0.950	**0.952**
LPIPS (↓)	0.028	0.027	**0.026**

Table 3. Ablation of the redundancy-aware weighting module: We show quantitative results for the *Dog* scene using the Scaled-up VolSDF, Ours without redundancy-aware weighting, and Ours.

5.3. Ablation Study

Ablation of the redundancy-aware weighting: To evaluate the effect of our redundancy-aware weighting module, we take the average of features from the different encoders instead of applying the redundancy-aware weighting. As seen in Table 3, the Scaled-up VolSDF and Ours without redundancy-aware weighting perform worse than our proposed FreBIS, attesting to its efficacy.

Assignment of frequency-bands to each encoder: Unlike the experiments in Sec. 5.2, we construct variants of our model where we assign frequency levels to the encoders unevenly. Note that the total number of frequency levels N is set to 6. Table 2 shows quantitative results for different configurations, under this setup. Though the optimal assignment of frequency domains seems to vary depending on the scene, the even distribution ($(N_{\mathrm{L}}, N_{\mathrm{M}}, N_{\mathrm{H}}) = (2, 2, 2)$) performs most stably across various scenes.

6. Conclusions

In this work, we proposed FreBIS, a novel approach for neural implicit surface representation. FreBIS stratifies the scene into multiple frequency levels according to the surface frequencies and leverages a novel *redundancy-aware weighting* module to effectively capture complementary information by promoting mutual dissimilarity of the encoded features. Empirical results show that coupling FreBIS encoders with the VolSDF decoder improves the qualities of reconstructed mesh as well as their viewpoint-based surface renderings.

Going forward, we plan to evaluate FreBIS on other datasets and backbones. Combining FreBIS with object-compositional frameworks, such as ObjectSDF [48] and RICO [25], should allow us to reconstruct more complex scenes with multiple objects, which can be leveraged for higher fidelity complex 3D simulation, and 3D content generation for AR/VR.

References

[1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural RGB-D surface reconstruction, 2022. 2

[2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3), 2009. 2

[3] Jeremy S. De Bonet. Poxels: Probabilistic voxelized volume reconstruction. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1999. 2

[4] A. Broadhurst, T.W. Drummond, and R. Cipolla. A probabilistic framework for space carving. In *Proceedings of IEEE International Conference on Computer Vision*, pages 388–393 vol.1, 2001. 2

[5] Zhiqin Chen, Thomas Funkhouser, Peter Hedman, and Andrea Tagliasacchi. Mobilenerf: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures. In *Proceedings of The Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[6] Pinxuan Dai, Jiamin Xu, Wenxiang Xie, Xinguo Liu, Huamin Wang, and Weiwei Xu. High-quality surface reconstruction using gaussian surfels. In *Proceedings of ACM SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024. 3

[7] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *Proceedings of The Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[8] O. Faugeras and R. Keriven. Variational principles, surface evolution, pdes, level set methods, and the stereo problem. *IEEE Transactions on Image Processing*, 7(3):336–344, 1998. 2

[9] Olivier D. Faugeras and Renaud Keriven. Complete dense stereovision using level set methods. In *Proceedings of the 5th European Conference on Computer Vision-Volume I - Volume I*, page 379–393, Berlin, Heidelberg, 1998. Springer-Verlag. 2

[10] Qiancheng Fu, Qingshan Xu, Yew-Soon Ong, and Wenbing Tao. Geo-Neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction, 2022. 2

[11] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, 2010. 2

[12] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 873–881, 2015. 2

[13] Michael Goesele, Noah Snavely, Brian Curless, Hugues Hoppe, and Steven M. Seitz. Multi-view stereo for community photo collections. In *Proceedings of IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007. 2

[14] Xiaodong Gu, Weihao Yuan, Heng Li, Zilong Dong, and Ping Tan. HIVE: HIerarchical Volume Encoding for Neural Implicit Surface Reconstruction, 2024. arXiv:2408.01677 [cs]. 2

[15] Yi Guo, Che Sun, Yunde Jia, and Yuwei Wu. Neural 3D Scene Reconstruction from Multiple 2D Images without 3D Supervision, 2023. arXiv:2306.17643 [cs]. 2

[16] Antoine Guédon and Vincent Lepetit. SuGaR: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *Proceedings of The Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv, 2023. arXiv:2311.12775 [cs]. 3

[17] Alexander Hornung and Leif Kobbelt. Hierarchical volumetric multi-view stereo reconstruction of manifold surfaces based on dual graph embedding. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, page 503–510, USA, 2006. IEEE Computer Society. 2

[18] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *Proceedings of SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024. 3

[19] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 3

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 6

[21] Kiriakos N. Kutulakos and Steven M. Seitz. A theory of shape by space carving. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 38(3):199–218, 2000. 2

[22] Patrick Labatut, Jean-Philippe Pons, and Renaud Keriven. Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In *Proceedings of IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007. 2

[23] Hai Li, Xingrui Yang, Hongjia Zhai, Yuqian Liu, Hujun Bao, and Guofeng Zhang. Vox-surf: Voxel-based implicit surface representation. *IEEE Transactions on Visualization and Computer Graphics*, 30(3):1743–1755, 2024. 3

[24] Runfa Blark Li, Keito Suzuki, Bang Du, Ki Myung Brian Lee, Nikolay Atanasov, and Truong Nguyen. Splatsdf: Boosting neural implicit sdf via gaussian splatting fusion, 2024. 3

[25] Zizhang Li, Xiaoyang Lyu, Yuanyuan Ding, Mengmeng Wang, Yiyi Liao, and Yong Liu. RICO: Regularizing the Unobservable for Indoor Compositional Reconstruction. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. arXiv, 2023. arXiv:2303.08605 [cs]. 3, 8

[26] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H. Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-Fidelity Neural Surface Reconstruction. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv, 2023. arXiv:2306.03092 [cs]. 3

[27] Zhihao Liang, Zhangjin Huang, Changxing Ding, and Kui Jia. HelixSurf: A robust and efficient neural implicit surface learning of indoor scenes with iterative intertwined regularization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv, 2023. arXiv:2302.14340 [cs]. 2

[28] Xiaoyang Lyu, Yang-Tian Sun, Yi-Hua Huang, Xiuzhe Wu, Ziyi Yang, Yilun Chen, Jiangmiao Pang, and Xiaojuan Qi. 3DGSR: Implicit surface reconstruction with 3d gaussian splatting, 2024. 3

[29] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of The European Conference on Computer Vision (ECCV)*, 2020. 2, 3

[30] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable Volumetric Rendering: Learning Implicit 3D Representations without 3D Supervision. In *Proceedings of The Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv, 2020. arXiv:1912.07372 [cs]. 2

[31] Michael Oechsle, Songyou Peng, and Andreas Geiger. UNISURF: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *International Conference on Computer Vision (ICCV)*. arXiv, 2021. arXiv:2104.10078 [cs]. 2

[32] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-dense similarity learning for multiple object tracking. In *Proceeding of the Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[33] Minyoung Park, Mirae Do, YeonJae Shin, Jaeseok Yoo, Jongkwang Hong, Joongrock Kim, and Chul Lee. H2O-SDF: Two-phase Learning for 3D Indoor Reconstruction using Object Surface Fields. In *Proceedings of The International Conference on Learning Representations*. arXiv, 2024. arXiv:2402.08138 [cs]. 2

[34] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019. 6

[35] Aarya Patel, Hamid Laga, and Ojaswa Sharma. Normal-guided Detail-Preserving Neural Implicit Functions for High-Fidelity 3D Surface Reconstruction, 2024. arXiv:2406.04861 [cs]. 2

[36] Marie-Julie Rakotosaona, Fabian Manhardt, Diego Martin Arroyo, Michael Niemeyer, Abhijit Kundu, and Federico Tombari. NeRFMeshing: Distilling Neural Radiance Fields into Geometrically-Accurate 3D Meshes, 2023. arXiv:2303.09431 [cs]. 3

[37] Johannes L. Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of European Conference on Computer Vision*, pages 501–518, Cham, 2016. Springer International Publishing. 2

[38] S.M. Seitz and C.R. Dyer. Photorealistic scene reconstruction by voxel coloring. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1067–1073, 1997. 2

[39] Jiaxiang Tang, Hang Zhou, Xiaokang Chen, Tianshu Hu, Errui Ding, Jingdong Wang, and Gang Zeng. Delicate textured mesh recovery from nerf via adaptive surface refinement. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2023. 3

[40] Hoang-Hiep Vu, Patrick Labatut, Jean-Philippe Pons, and Renaud Keriven. High accuracy and visibility-consistent dense multiview stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):889–901, 2012. 2

[41] Jingwen Wang, Tymoteusz Bleja, and Lourdes Agapito. GO-Surf: Neural Feature Grid Optimization for Fast, High-Fidelity RGB-D Surface Reconstruction. In *Proceedings of International Conference on 3D Vision (3DV)*. arXiv, 2022. arXiv:2206.14735 [cs]. 2

[42] Jiepeng Wang, Peng Wang, Xiaoxiao Long, Christian Theobalt, Taku Komura, Lingjie Liu, and Wenping Wang. NeuRIS: Neural Reconstruction of Indoor Scenes Using Normal Priors. In *Proceedings of European Conference on Computer Vision*. arXiv, 2022. arXiv:2206.13597 [cs]. 2

[43] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. In *Proceedings of 35th Conference on Neural Information Processing Systems*. arXiv, 2021. arXiv:2106.10689 [cs]. 2, 3

[44] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. HF-NeuS: Improved Surface Reconstruction Using High-Frequency Details. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*. arXiv, 2022. arXiv:2206.07850 [cs]. 2

[45] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. NeuS2: Fast Learning of Neural Implicit Surfaces for Multi-view Reconstruction, 2023. arXiv:2212.05231 [cs]. 3

[46] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. PET-NeuS: Positional Encoding Tri-Planes for Neural Surfaces. In *Proceedings of The Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv, 2023. arXiv:2305.05594 [cs]. 2

[47] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 6

[48] Qianyi Wu, Xian Liu, Yuedong Chen, Kejie Li, Chuanxia Zheng, Jianfei Cai, and Jianmin Zheng. Object-Compositional Neural Implicit Surfaces. In *Proceedings of European Conference on Computer Vision*. arXiv, 2022. arXiv:2207.09686 [cs]. 3, 8

[49] Qianyi Wu, Kaisiyuan Wang, Kejie Li, Jianmin Zheng, and Jianfei Cai. ObjectSDF++: Improved object-compositional neural implicit surfaces. In *Proceedings of the IEEE/CVF*

International Conference on Computer Vision. arXiv, 2023. arXiv:2308.07868 [cs]. 3

[50] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. BlendedMVS: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of The Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv, 2020. arXiv:1911.10127 [cs]. 2, 6

[51] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Ronen Basri, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 2020. arXiv:2003.09852 [cs]. 2

[52] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume Rendering of Neural Implicit Surfaces. In *Proceedings of The Conference on Neural Information Processing Systems*. arXiv, 2021. arXiv:2106.12052 [cs]. 2, 3, 4, 6

[53] Lior Yariv, Peter Hedman, Christian Reiser, Dor Verbin, Pratul P. Srinivasan, Richard Szeliski, Jonathan T. Barron, and Ben Mildenhall. Bakedsdf: Meshing neural sdfs for real-time view synthesis, 2023. 3

[54] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. MonoSDF: Exploring Monocular Geometric Cues for Neural Implicit Surface Reconstruction. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. arXiv:2206.00665 [cs]. 2

[55] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of The Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 6

FreBIS: Frequency-Based Stratification for Neural Implicit Surface Representations

—Supplementary Material—

Naoko Sawada[1,2] Pedro Miraldo[1] Suhas Lohit[1] Tim K. Marks[1] Moitreya Chatterjee[1]

[1]Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA

[2]Information Technology R&D Center, Mitsubishi Electric Corporation, Kanagawa, Japan

Sawada.Naoko@df.MitsubishiElectric.co.jp,{miraldo,slohit,tmarks,chatterjee}@merl.com

The following summarizes the supplementary materials we present:

1. Ablation study of the *redundancy-aware weighting* module.
2. Comparative study of the number of frequency levels.
3. Comparative study of encoder architecture variants.

1. Ablation study of the redundancy-aware weighting module

A key innovation of FreBIS is the *rendundancy-aware weighting* module which combines the complementary information from the different encoders by promoting mutual dissimilarity. Table 1 shows quantitative comparison results of the FreBIS with and without this module. The results show that our model with this module outperforms the variant without it, where a simple averaging of the encoder features is performed, clearly bringing out its effectiveness.

2. Comparative study of the number of frequency levels

We conduct experiments to study the effect of the choice of frequency levels N for both FreBIS and Scaled-up VolSDF [1]. As shown in Table 2 and Fig. 1, the Scaled-up VolSDF is sensitive to the choice of frequency levels and has particular difficulty in dealing with higher frequency encodings. In particular, the Scaled-up VolSDF with $N = 9$ results in a reconstructed mesh with too many bumps, while that with $N = 12$ results in a mesh that is hard to interpret. On the other hand, FreBIS is capable of processing higher–frequency information without sacrificing information gleaned from the low–frequency bands. Fig. 2 and 3 show the qualitative comparisons of rendered images and reconstructed meshes with $N = 6, 9, 12$ using FreBIS on the *Doll*, *Bull*, and *Robot* scenes.

3. Comparative study of encoder architecture variants

In order to design the encoders of FreBIS optimally, we study the effect of varying the number of layers of each of the three encoders of FreBIS and compare their performances. As seen from the results in Table 3 as well as Fig. 4, and Fig. 5. FreBIS performs comparably irrespective of the choice of encoder architecture, maintaining a good performance throughout. Based on this analysis and in order to stay consistent with the baseline VolSDF [1] architecture, we choose the 6–layer architecture for each encoder, with each layer having 256 dimensions.

References

[1] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume Rendering of Neural Implicit Surfaces. In *Proceedings of The Conference on Neural Information Processing Systems*. arXiv, 2021. arXiv:2106.12052 [cs]. 1

	(a) Scaled-up VolSDF	(b) Ours w/o redundancy-aware weighting	(c) Ours
PSNR (↑)	28.32	28.31	**28.56**
SSIM (↑)	0.949	0.950	**0.952**
LPIPS (↓)	0.028	0.027	**0.026**

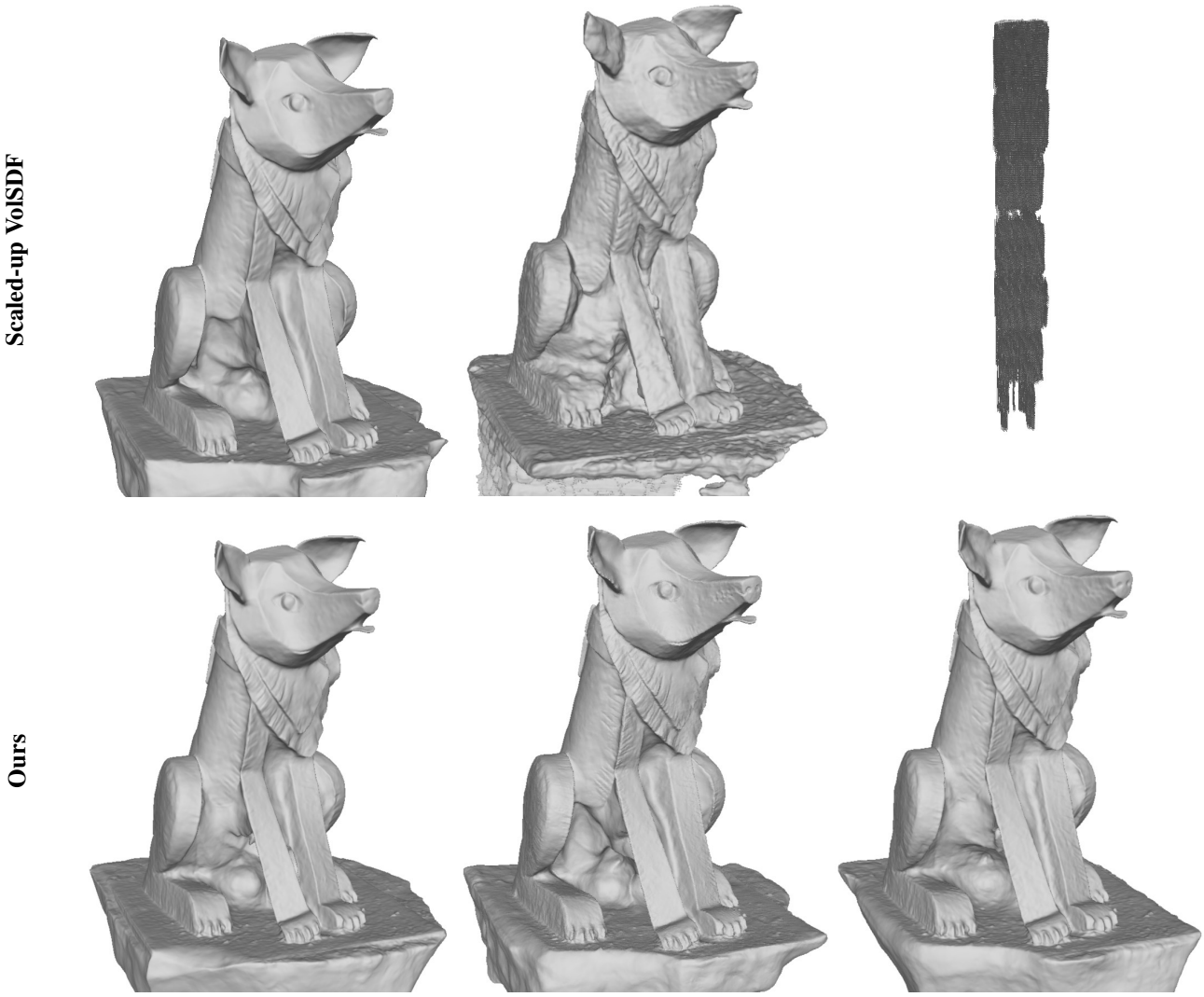
Table 1. Ablation of the redundancy-aware weighting module: We show quantitative results for the *Dog* scene using the Scaled-up VolSDF, Ours without redundancy-aware weighting, and Ours.

Method	Frequency level (N)	Doll	Egg	Head	Angel	Bull	Robot	Dog	Bread	Camera	**Mean**
Scaled-up VolSDF	6	26.07	27.15	26.62	30.37	26.08	25.07	28.32	29.44	23.02	26.90
Ours	6	26.22	27.48	27.29	30.52	26.33	26.69	28.56	30.22	23.08	27.38
Scaled-up VolSDF	9	25.69	26.66	26.94	28.59	26.02	22.67	26.78	32.62	23.45	26.60
Ours	9	26.10	27.47	27.24	30.56	25.78	26.85	28.88	30.08	23.28	27.36
Scaled-up VolSDF	12	–	–	–	–	–	–	24.86	–	19.59	–
Ours	12	26.02	27.54	25.81	30.56	26.89	26.66	28.62	30.18	30.26	27.21

Table 2. Comparison of viewpoint-based rendering performance with a varying number of frequencies, as measured by PSNR. – denotes that the method failed to construct a mesh during training.

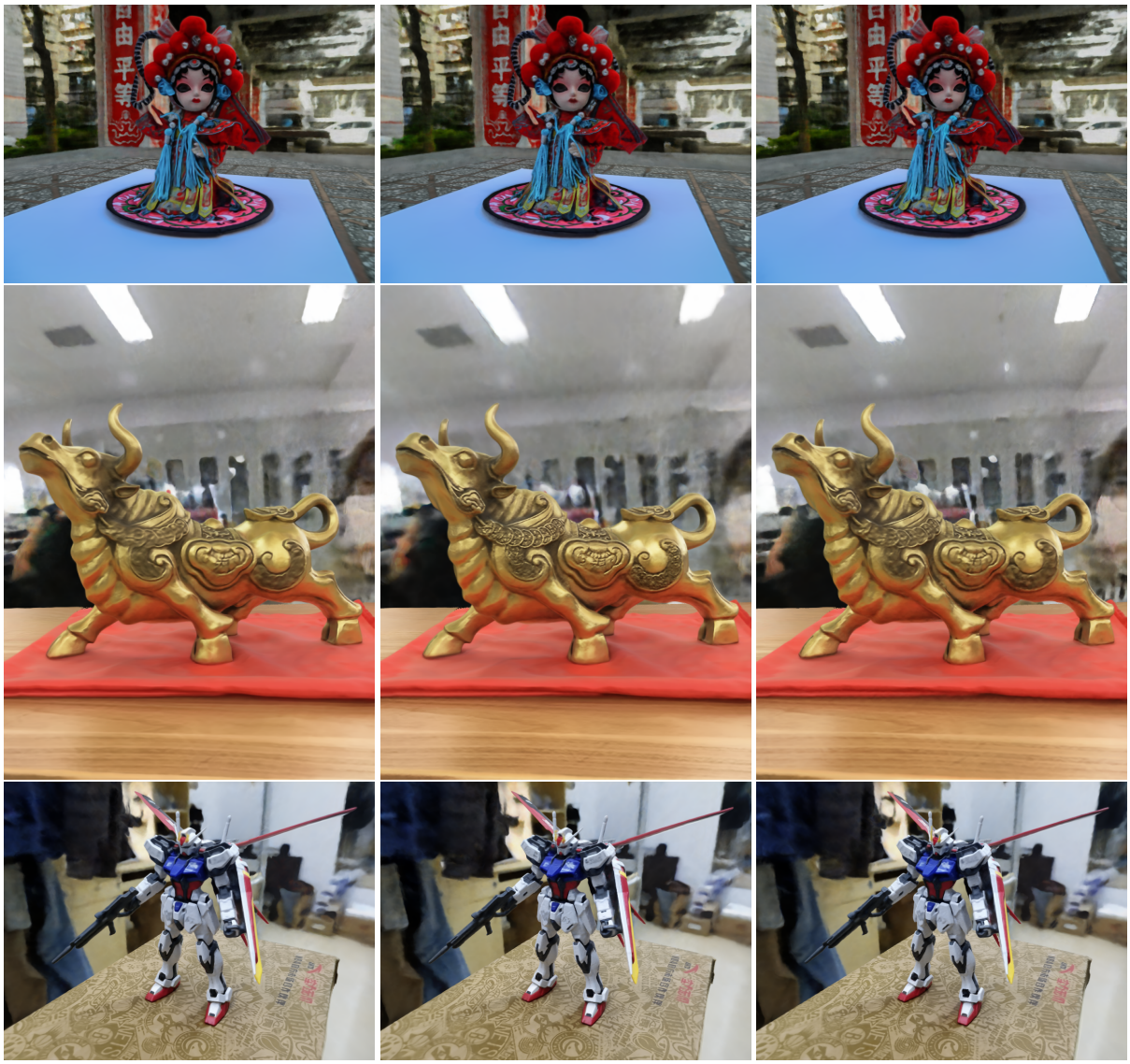
$N_{\mathrm{L}}, N_{\mathrm{M}}, N_{\mathrm{H}}$	Doll	Egg	Head	Angel	Bull	Robot	Dog	Bread	Camera	**Mean**
$6, 6, 6$	26.22	27.48	27.29	30.52	26.33	26.69	28.56	30.22	23.08	27.38
$5, 5, 5$	26.18	27.47	27.14	30.42	26.37	26.62	28.55	30.20	23.10	27.34
$4, 4, 4$	26.25	27.51	26.96	30.49	26.37	26.51	28.18	31.12	23.17	27.39
$4, 5, 6$	26.18	27.45	27.13	30.50	26.38	26.64	28.60	30.16	23.19	27.36
$2, 4, 6$	26.26	27.47	24.45	30.44	25.95	26.67	28.74	31.60	23.21	27.19

Table 3. Performance comparison of variants of FreBIS with varying number of encoder layers, as measured by PSNR.



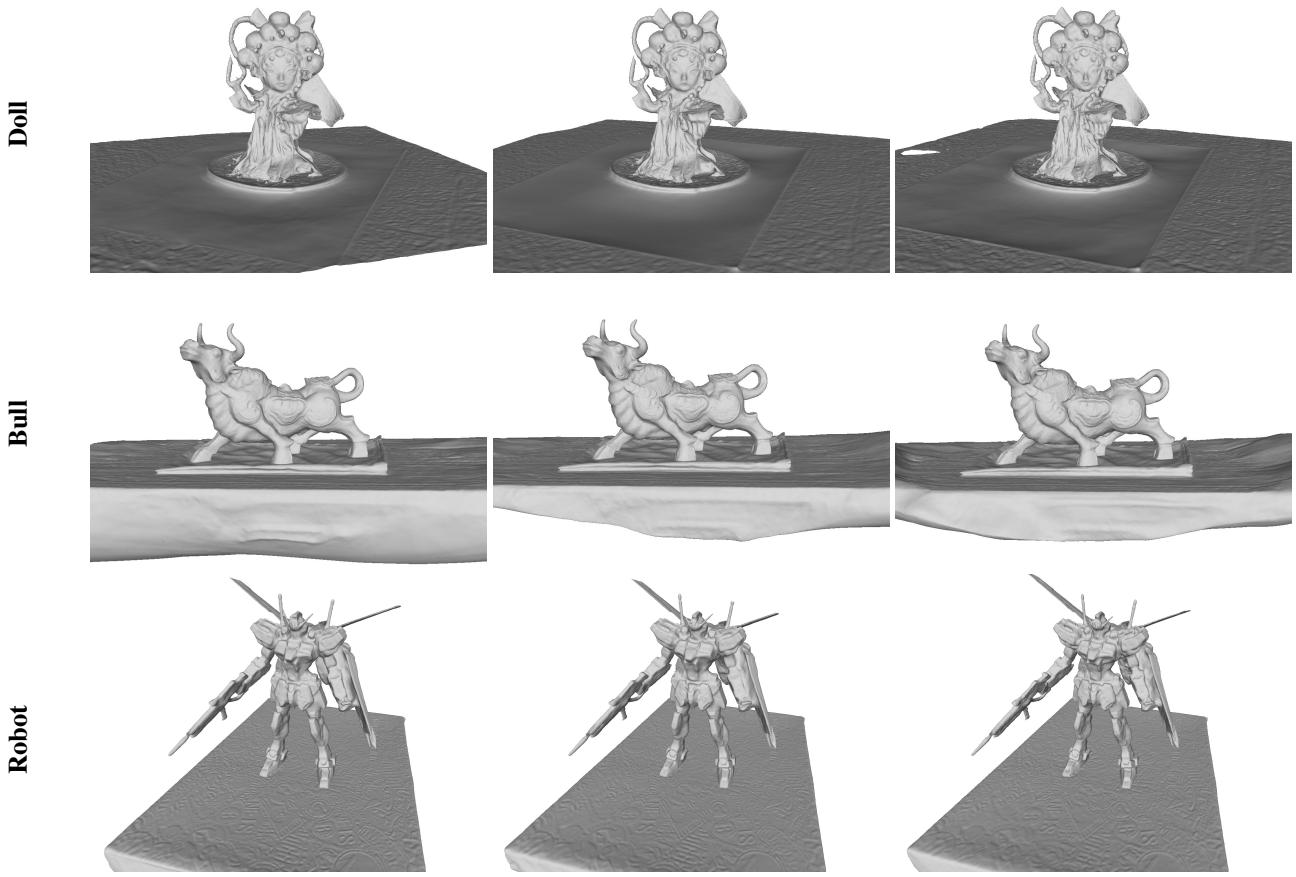
(a) $N = 6$ (b) $N = 9$ (c) $N = 12$

Figure 1. Qualitative comparison on the capability to deal with higher frequencies.



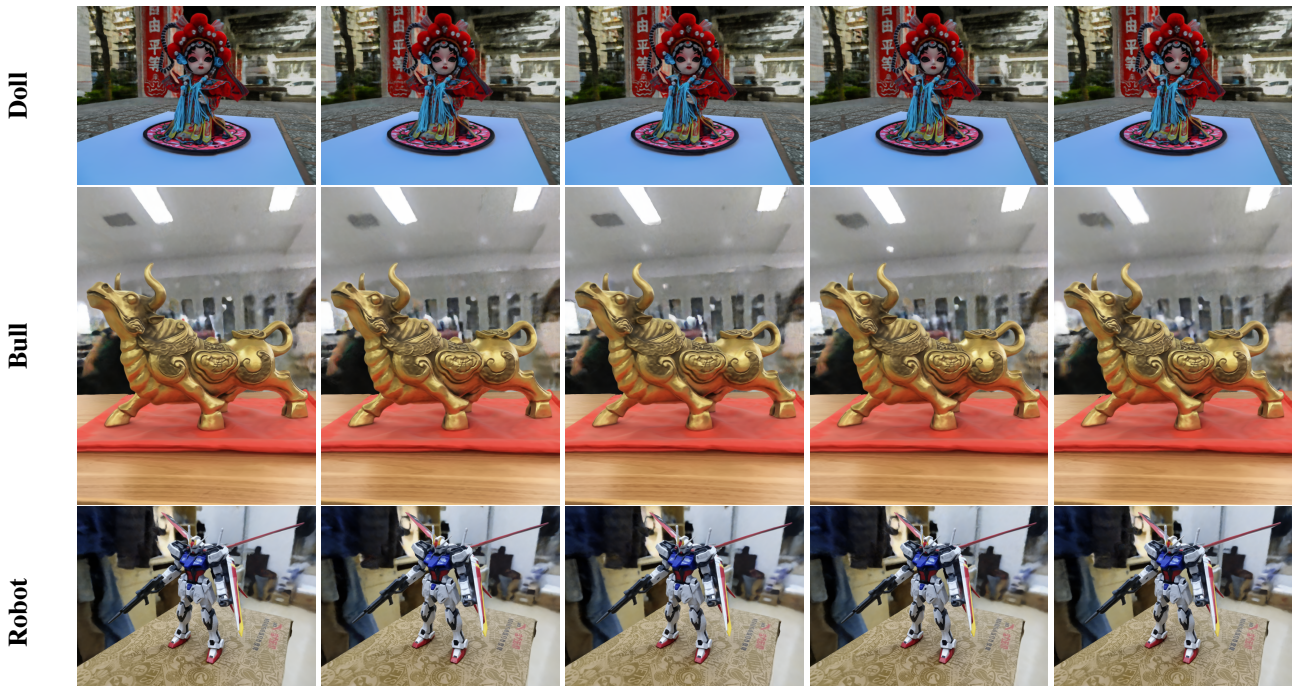
(a) $N = 6$ (b) $N = 9$ (c) $N = 12$

Figure 2. Qualitative comparison of viewpoint-based scene rendering with varying number of frequencies.



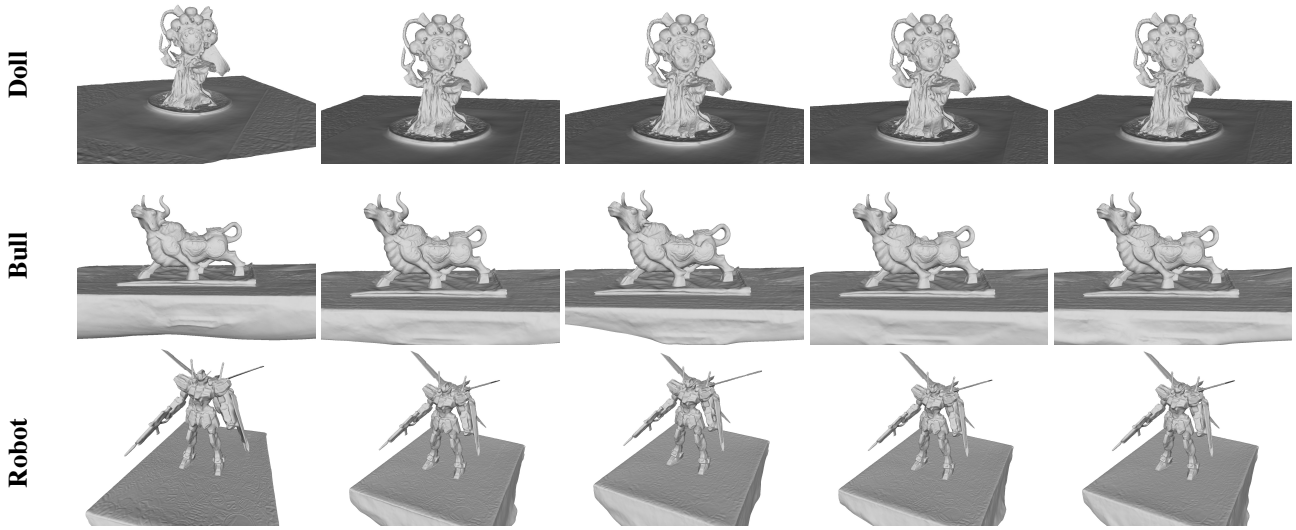
(a) $N = 6$ (b) $N = 9$ (c) $N = 12$

Figure 3. Qualitative comparison on surface reconstruction with a different number of frequencies.



(a) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (6, 6, 6)$ (b) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (5, 5, 5)$ (c) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (4, 4, 4)$ (d) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (4, 5, 6)$ (e) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (2, 4, 6)$

Figure 4. Qualitative comparison on viewpoint-based scene rendering using FreBIS, obtained by varying the number of encoder layers.



(a) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (6, 6, 6)$ (b) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (5, 5, 5)$ (c) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (4, 4, 4)$ (d) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (4, 5, 6)$ (e) $(\mathrm{L}, \mathrm{M}, \mathrm{H}) = (2, 4, 6)$

Figure 5. Qualitative comparison based on 3D surface reconstruction using FreBIS, obtained by varying the number of encoder layers.