MegaSurf: Learnable Sampling Guided Neural Surface Reconstruction Framework of Scalable Large Scene

Anonymous Authors

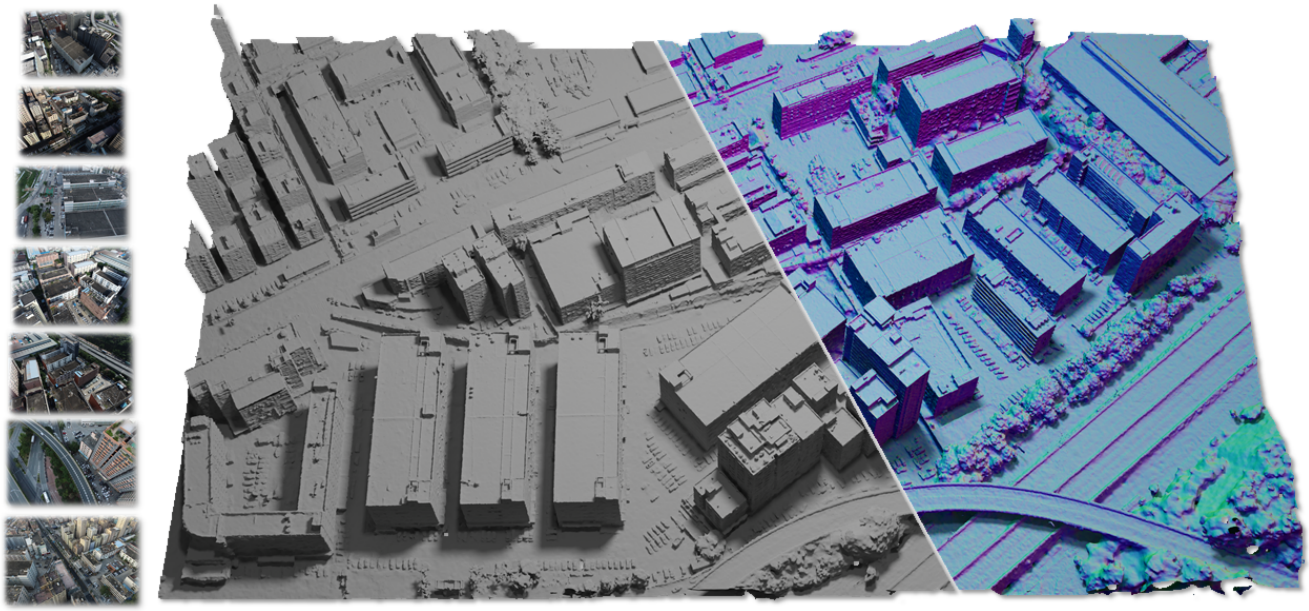


Figure 1: We present MegaSurf, an efficient and robust neural surface reconstruction framework to reconstruct the 3D large-scale scene from thousands of input RGB images collected by the drone. MegaSurf has both the robustness of the stereo matching and the high-fidelity details of the rendering-based reconstruction methods.

ABSTRACT

We present MegaSurf, a Neural Surface Reconstruction (NSR) framework designed to reconstruct 3D models of large scenes from aerial images. Many methods utilize geometry cues to overcome the shape-radiance ambiguity, which would produce large geometric errors. However, directly using inevitable imprecise geometric cues would lead to degradation in the reconstruction results, especially on large-scale scenes. To address this phenomenon, we propose a Learnable Geometric Guider (LG Guider) to learn a sampling field from reliable geometric cues. The LG Guider decides which position should fit the input radiance and can be continuously refined by rendering loss. Our MegaSurf uses a Divide-and-Conquer training strategy to address the synchronization issue between the Guider and the lagging NSR's radiance field. This strategy enables the Guider to transmit the information it carried to the radiance field without being disrupted by the gradients back-propagated from the lagging rendering loss at the early stage of training. Furthermore, we propose a Fast PatchMatch MVS module to derive the geometric cues in the planer regions that help overcome ambiguity. Experiments on several aerial datasets show that MegaSurf can overcome ambiguity while preserving high-fidelity details. Compared to SOTA methods, MegaSurf achieves superior reconstruction accuracy of large scenes and boosts the acquisition of geometric cues more than four times.

CCS CONCEPTS

• **Computing methodologies** → **Reconstruction**.

KEYWORDS

Neural Surface Reconstruction, Large Scale Scenes, Multiview Reconstruction

1 INTRODUCTION

Recently, Neural Surface Reconstruction (NSR), derived from neural radiance field (NeRF)[18, 28, 42, 49], not only excels in high-fidelity novel view synthesis, but also enables accurate geometric acquisition. Accurate 3D surface model is an essential and basic element to create immersive experiences in the game engines and VR experiences. Although NSR has achieved good results in small-scale scenes, there is limited research on its effectiveness in large-scale

Unpublished working draft. Not for distribution.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACM MM, 2024, Melbourne, Australia
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

scenes. Besides, there are many works [10, 26, 27, 34, 37, 39] on the novel view synthesis of large scenes,but little research on the 3D reconstruction directly using images without the aid of LiDAR [5, 21]. Utilizing drones for image acquisition and employing NSR technology can efficiently digitize and vividly recreate cities and historical sites for preservation, while also supporting wide-spreading of AR/VR applications.

However, NSR often encounters geometric errors due to shape-radiance ambiguity, as NSR uses rendering loss to optimize geometries for SDF network [2] implicitly. This problem becomes worse when NSR needs to render aerial images captured for large and complex scenes. Therefore, existing works [2, 32, 48] introduce geometric cues from multi-view stereo into NSR, imposing additional geometric constraints on the rendering, thereby improving the accuracy of the NSR methods.

Methods like MonoSDF [32, 48] add a geometric loss by rendering depth from network to compare with the geometric cues provided by depth estimation. Some other methods like GeoNeuS [2, 29] use a multiview photometric consistency loss derived from the implicit surface as the geometric loss without explicitly deriving geometries from MVS. However, geometric errors and noise persist in the radiance field due to strong constraints caused by geometry loss. This prevents the implicit surface from fitting the real geometry accurately and leads to the degradation of details. NerfingMVS [35] employs confidence of geometric cues to define the sampling range around the prior to deal with noisy geometric cues. However, the confidence of geometric cues is difficult to assess, and manually set the threshold to the sampling ranges is too rigid to be applied to different and variable datasets.

We propose a Learnable Geometric Guider (LG Guider) which firstly distill the geometric cues to the sampling network to avoid sampling on the ambiguous regions, and also can be continuously refined by rendering to overcome the missing details due to noises of geometric cues. If the LG Guider is used to guide an unoptimized radiance field directly, the learned geometric information carried by the Guider will be damaged and causing ambiguity again (Fig. 3). Therefore, we propose the Divide-and-Conquer training strategy as shown in Fig. 2. Firstly, we train the LG Guider with geometry net with geometry cues, to distill the prior geometric cues to prior knowledge of sampling and SDF field. Then we freeze the LG Guider, and train render net. The purpose is to use the distilled sampling retrain the the radiance field falling into ambiguous regions. Finally we train the full network, to refine the sampling and geometry by rendering loss for recovering geometric details from noisy geometric cues. To be noticed, geometry cues is only introduced in the first stage to avoid its continuous noise effects to the final results.

In additional, we find that ambiguities in NSR often occur in the large planar geometries in the region of low texture and shadows, while the complex geometries often easier to be reconstructed due to their rich and distinct color. We propose a fast PatchMatch MVS module to efficiently reconstruct the large planar geometries. A novel local propagation strategy is designed which progressively propagate geometries with similar plane with the SFM points, with only one step of PatchMatch operation performed per pixel to speed up the MVS process.

In summary, our main contributions are the following:

- We introduce a Learnable Geometric Guider to distill the geometric cues to overcome shape-radiance ambiguities and can be continuously refined by rendering to recover details from geometric noises.
- We propose a Divide-and-Conquer training strategy to improve the guidance of learning of shape and radiance field using the Learnable Geometric Guider.
- We present a fast MVS module to efficiently obtain high confidence planar geometric priors over 4× improvement in speed where large shape-radiance ambiguities often occur.
- On the several aerial photography datasets, our algorithm achieved the best results of quantitative and qualitative results. To our knowledge, we are the first to extend accurate NSR to large scale aerial scene.

2 RELATED WORK

Multiview stereo matching. Multiview Stereo (MVS) [24] aims to recover 3D geometric model of the real scene from input images. The key idea of image based multiview reconstruction is photo-consistency matching [3, 23, 41]. However, the performance of local photo-consistency matching is easily reduced in regions with low textures, shadows, and non-Lambertian materials. Therefore, several global matching aggregation methods are applied to improve the quality, including semi-global optimization [6], PatchMatch [23], and 3D convolution regularization [43]. Even though the learning-based MVS methods [4, 9, 13, 17, 38, 43, 50] show their advantages of reconstruction in difficult regions, their application on large-scale aerial datasets is limited due to the lack of various 3D training datasets, which are often expensive to acquire. Patchmatch-based MVS methods [23, 40], with their efficient parallelization structure and robust performance, are more suitable and already widely applied for large-scale scene reconstruction. However, common PatchMatch MVS requires performing PatchMatch operations several times through all pixels globally from random initialization. These intensive computation especially on large scale datasets introduce an unnegligible overhead when using them as geometric cues for NSR.

Neural surface reconstruction. Recently, rendering-based neural surface reconstruction methods [25, 30, 44, 45] have become a promising way to promote the development of 3D reconstruction due to their high-quality reconstruction results, especially for fine structures [11, 33] and training speed [22, 31, 36]. The multi-resolution hash encoding [19] provides a compact high-resolution feature representation which shows its potential for high-fidelity reconstruction for large scenes. Li et al [11] introduce a progressive training strategy on the multi-resolution hash encoding representation, and a numerical calculation of normals, firstly extending the high reconstruction accuracy to large outdoor scenes. However, on large-scale aerial scenes, large geometrical errors often occur due to the more server shape-radiance ambiguity in the complicate large scenes as show by Neuralangelo [11] results in Figure 6.

Neural surface reconstruction with geometry cues. Many works incorporate geometry cues into NSR reconstruction to address the ambiguity problem. Several of these utilize the geometry cues as a geometry loss to ensure that the geometry reconstructed from NSR are consistent with the geometric cues. However, noisy

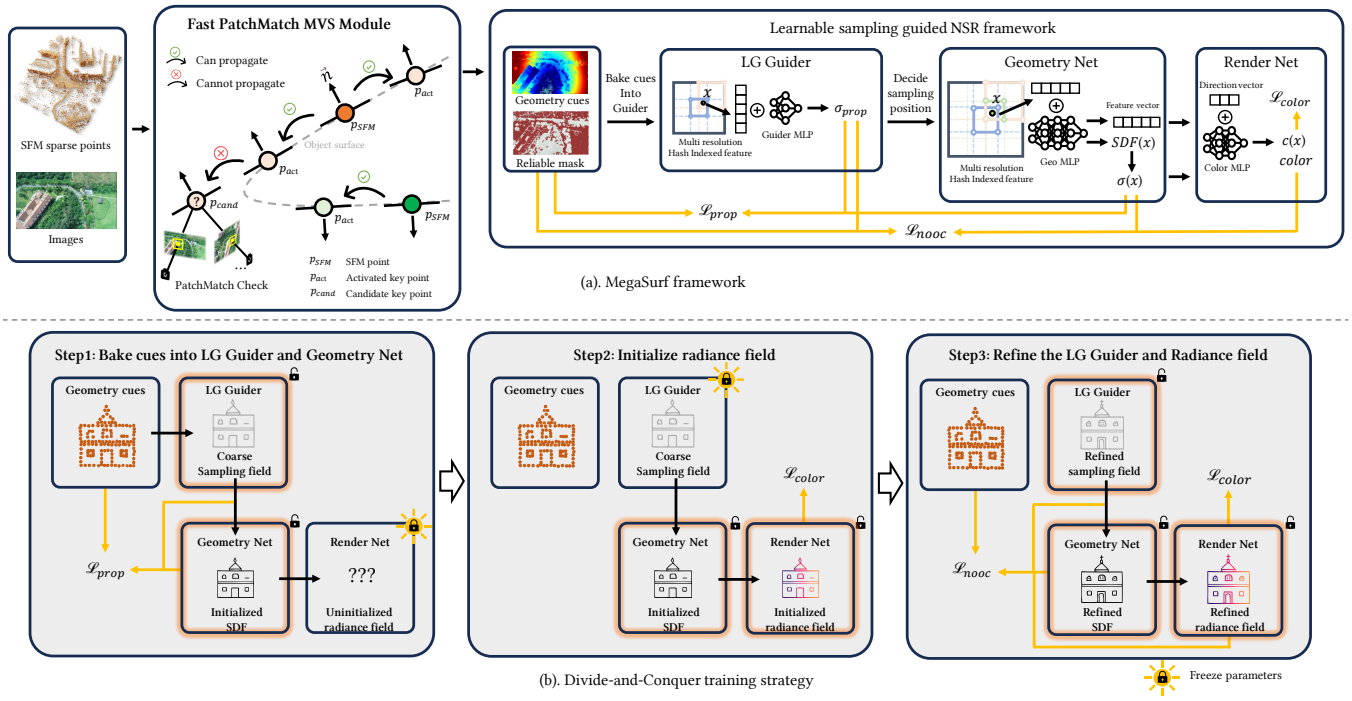


Figure 2: Method overview. (a) We propose a Fast PatchMatch MVS Module (Section. 3.4) to rapidly propagate SFM points to obtain high-confidence geometry cues. Then, we use these cues to train our Learnable Geometric Guider (LG Guider, Section 3.2). The LG Guider which position should be used to fit the input radiance and can be continuously refined by rendering loss. To address the synchronization issue between the Guider and the lagging radiance field, we propose a three steps Divide-and-Conquer training strategy (Section. 3.3). This strategy enables the Guider to efficiently guide the radiance field training while preserving its learned geometric information from being disrupted by the rendering. (b) The detail of our Divide-and-Conquer training strategy. We bake the geometry cues into LG Guider and Geometry net in Step 1, then freeze the LG Guider parameters and initialize the whole radiance field in Step 2. In Step 3, we use rendering loss to refine the radiance field and our LG Guider and propose L_{prop} to preserve the geometry information the Guider carries from being impaired.

geometry cues persistently contribute to the loss, resulting in over-smooth effects on the detailed structures. To avoid the intensive computation of global optimization [6, 23] of MVS, some other works directly use the photo consistency measurement, Normalized Cross-Correlation (NCC) as geometry cues. However, NCC, a highly localized geometric measurement, often fails to give reliable geometry in the ambiguous areas, resulting in a worse reconstruction in the large scenes(see Supplementary Materials for details). Another approach retrains the sampling points around the geometric cues to deal with noisy geometric cues. Wei et al employ the confidence of geometric cues to define the sampling range around the prior to deal with noises. However, the confidence of geometric cues is difficult to assess and manually set the threshold to the sampling ranges cannot be applied to different and variable datasets.

3 METHOD

As shown in Figure 2, MegaSurf proposes a Fast PatchMatch MVS Module to efficiently obtain the geometry cues (Section 3.4). Then, we propose a Learnable Geometric Guider (Section 3.2) to learn these reliable geometry cues. Next, MegaSurf employs a Divide-and-Conquer training strategy (Section 3.3) to train the radiance field.

3.1 Preliminary

Neural radiance field. NeRF [18] represents a complex 3D scene as a learned function that maps each 3D point and corresponding ray direction to a color and density. It integrates the color of sampled points along the ray to render each pixel:

$$\mathrm{C}(r) = \sum_i \omega_i \mathbf{c}_i, \quad \omega_i = \boldsymbol{T}_i \alpha_i, \quad \alpha_i = 1 - \exp\left(-\sigma_i \delta_i\right) \tag{1}$$

where α_i is the opacity of the ith sample point along the ray, σ_i is the corresponding density, which is also the learned function's output. $\delta_i = t_i - t_{i-1}$, is the distance between two sample points, t is the distance to the ray center. $\boldsymbol{T}_i = \prod_{j=1}^{i-1}\left(1 - \sigma_j\right)$ is the accumulated transmittance. As the geometry of NeRF is represented by density, extracting surfaces from densities often leads to noisy results.

Neural surface reconstruction. Most rendering based NSR methods take NeRF as the backbone and use signed distance function (SDF) as the geometric representation instead of density in NeRFs. The surface can be represented by the zero-level set of the SDF, $S = \{\mathbf{x} : f(\mathbf{x}) = 0\}$, where x is a 3D position. To use volume rendering, VolSDF [44] defines the volume density function τ to map the signed distance $f(x)$ to volume density σ:

$$\tau(\mathbf{x}) = \beta^{-1}\Psi_\beta\left(f(\mathbf{x})\right), \tag{2}$$

where $\beta > 0$ is a scheduling parameters and approaches 0 during optimization,$\tau(\mathbf{x})$ is the cumulative distribution funcion (CDF) of the zero-mean Laplace distribution with scale β. Manually controlling the β allows different reconstructed cases to have the same β, so that the surface details of different cases are consistent.

Neuralangelo. Recently, multi-resolution hash encoding proposed by Muller et al. [19] is a compact feature representation that can represent large-scale scenes in unprecedented detail. Neuralangelo [11] designs a coarse-to-fine optimization scheme to reconstruct the surfaces with progressive levels of detail:

$$\gamma_l = [F_0, F_1, ..., F_{start+l}], \quad l_{start} < l < l_{max}, \tag{3}$$

where γ represents the features from hash grids, F is the features of each level of hash grid, and the coarse to fine resolution spans from level l_{start} to level l_{max}. Another important contribution is the design of a numerical gradient computation to distribute the back-propagation updates to wider neighboring hash grids to improve the smoothness of surface reconstruction:

$$\nabla_x f(x) = \frac{f\left(\gamma\left(x+\epsilon_x\right)\right) - f\left(\gamma\left(x-\epsilon_x\right)\right)}{2\epsilon}, \tag{4}$$

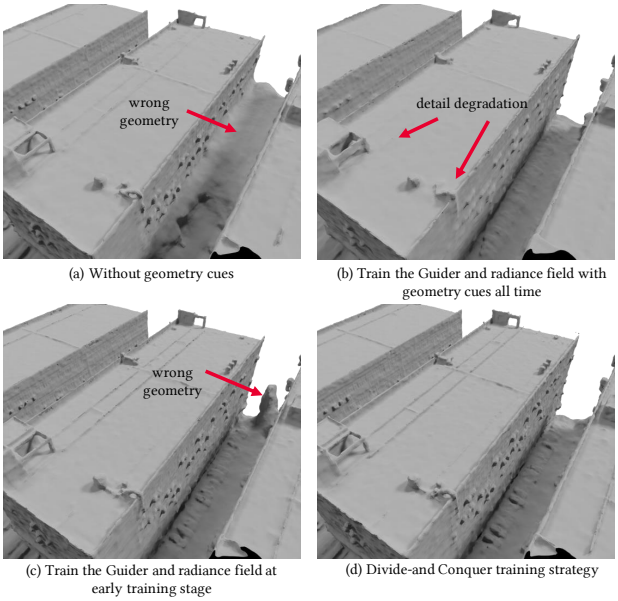
where ϵ is the step size away from x for sampling points to calculate gradient numerically.

However, when applying it to large-scale aerial datasets, severe shape radiance often happens in the areas of heavy shadows, low textures, and illumination variations.

3.2 Learnable sampling guided NSR

Our learnable geometric guider borrows the sampling proposal network to distill the geometric clues to restrain samplings in the ambiguity areas. The proposal net adopts a two-level, (coarse level: $Prop_0$ and fine level: $Prop_1$), coarse to fine hierarchical sampling procedure [18]. Each level consists of a small multi-resolution hash grid and a tiny MLP to learn the importance of sampling to propose informative samples to subsequent geometry and render net to learn the SDF and radiance field as shown in Figure 2.

A naive solution is using geometric cues to train the LG Guider while simultaneously training the whole network. This approach is similar to [35] which constrains the sampling during training, but our sampler is learnable. As the geometric noises affect the training process all the time, this solution would also inevitably introduce noisy geometry cues to the final reconstruction, leading to missing details as shown in Figure 3b. Another approach is to bake the geometric cues by training LG Guider and the whole network at the beginning of the training and then letting the network refine the noisy geometric guider without using geometric cues. However, when introducing the render net into the baking step, the ambiguity problem of rendering will directly affect the baking results, leading to some remaining geometric errors in the final reconstruction, as shown in Figure 3c.



(a) Without geometry cues (b) Train the Guider and radiance field with geometry cues all time

(c) Train the Guider and radiance field at early training stage (d) Divide-and Conquer training strategy

Figure 3: The illustration of the impact of different training strategies using geometry cues. (a) Training without geometry cues. (b) Train the LG Guider and radiance field simultaneously using geometry cues all the time. (c) Only use geometry cues to train the LG Guider and the radiance field at early training stage. (d) Our Divide-and-Conquer training strategy showing in Figure 2.

3.3 Divide-and-Conquer training strategy

We propose a Divide-and-Conquer training strategy to distill the geometric information into the LG Guider and enable it to specify specific positions in the radiance field for optimization. The Divide-and-Conquer training strategy consists of three steps: 1) Baking geometry cues into LG Guider, 2) Initializing the radiance field, 3) Refining training.

Step1: Bake cues into LG Guider. In baking stage, we train the LG Guider with geometry net with geometry cues, to distill the prior geometric cues to prior knowledge of sampling and SDF field while leaving render net untrained as shown in Figure 2 bottom left. The reason for not training the render net at this stage is that when rendering loss is introduced, the introduced ambiguity problem can affect the distillation of prior geometric knowledge to both the LG Guider and the geometry net. This reduces the effectiveness of LG Guider to resolving ambiguities as shown in bottom left in Figure 3

To be specific, we maximize the sampling weights ω^h given by coarse level $Prop_0$ and fine level $Prop_1$ and the radiance field weight ω^{geo} given by geometry net within the range $[t_{prior} - \epsilon, t_{prior} + \epsilon]$ around the geometry cues t_{prior} :

$$\begin{aligned} L_{prop} =& 1 - \sum_{i\in\Lambda}(\omega_i^h + \omega_i^{geo}), \\ \Lambda : \{i : t_{prior} - \epsilon < t_i <& t_{prior} + \epsilon\}, h \in Prop_0, Prop_1. \end{aligned} \tag{5}$$

where t_{prior} is the distance between the camera center to the 3D point corresponding to the depth cue. The computation of ω_i^h follows the Eqn. 1, we replace the geometry net output σ with the LG Guider output σ_{prop}.

We further add a Curvature loss to improve the smoothness of sampling field and the geometry to address the noise and incompleteness of the geometric cues:

$$\mathcal{L}_{\text{curv}} = \frac{1}{N}\sum_{i=1}^{N}\left|\nabla^2 f(\mathbf{x}_i)\right|, \tag{6}$$

The overall loss of step 1 is:

$$L_{step1} = L_{prop} + L_{curv}. \tag{7}$$

In this way, step 1 completes the training of the LG Guider and also finishes the initialization of the geometry net. Next, we need to initialize the entire radiance field by adding color information to the geometry represented by the geometry net.

Step2: Initialize radiance field. Since our geometry net and render net share a multi-resolution hash grid to provide hash indexed feature vectors, modifications to the render net will impact the geometry net. However, the structure of the LG Guider is independent of the geometry net and render net. To prevent the learned information of the LG Guider from being compromised by the uninitialized render net, we freeze the parameters of the LG Guider.

Since the parameters of the LG Guider are fixed, the sampling positions outputted by the LG Guider are also fixed. The radiance field will only prioritize using these positions to fit the input colors, which will help overcome shape-radiance ambiguity.

The step 2 training loss is defined as:

$$L_{step2} = L_{color} + L_{curv} + L_{eikonal}. \tag{8}$$

We take rendering loss L_{color} as primary loss, and take Curvature loss L_{curv} and Eikonal loss $L_{eikonal}$ as regularization terms.

Step3: Refine the LG Guider and radiance field. In this step, we unfreeze all parameters for training, aiming to use rendering loss to refine the LG Guider which is affected by prior noisy geometry. During this process, we further employs the prior geometry cues to avoid the rendering to step back into ambiguity regions, hence we introduce Non occupancy loss L_{nocc}:

$$\begin{aligned} L_{nocc} &= || \sum_{i\in\Gamma} \omega_i c_i ||_1, \\ \Gamma &:\{i : t_i < t_{prior} - \epsilon\}, \end{aligned} \tag{9}$$

where ω and c is given by geometry net and render net. L_{nocc} is used to ensure that no new surfaces appear between the camera center and the surfaces corresponding to reliable cues. Thus that the accumulated color should be nearly black color and L_{nocc} should be close to 0. As the LG Guider decides the radiance field's sampling positions, the rendering loss can be backpropagated to LG Guider, which makes the sampling more precise.

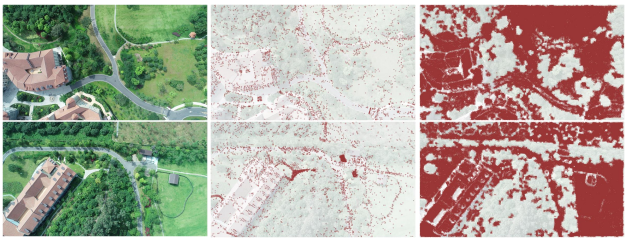
we add a Non occupancy loss into the loss of step 2:

$$L_{step3} = L_{color} + L_{nocc} + L_{curv} + L_{eikonal}. \tag{10}$$

.

After training is completed, we utilize Marching Cube [16] to extract the zero level set from the signed distance function (SDF) represented by the geometry net as the final reconstructed mesh.

3.4 Fast PatchMatch MVS Module



(a) Images (b) SFM points (c) Confidence mask

Figure 4: The illustration of the high-confidence region acquired by our Fast PatchMatch MVS module. (a) The input images. (b) Sparse SFM points. (c) The high-confidence position which we used as the geometric prior during our NSR training.

Prelimiary of heavy PatchMatch MVS module. Commonly used PatchMatch MVS module starts from randomly initializing geometry on each pixel, and every pixel uses PatchMatch optimization to select its best geometric candidate with the smallest photo-consistency loss $\boldsymbol{E}_{NCC}$ from all the candidates propagated from its neighboring pixels[23, 40]. In a nutshell, PatchMatch operation is to chose the best geometric candidate propagated from neighborhoods for each pixel. Every pixel will continuously update its geometry through PatchMatch until it receives its accurate geometry. Due to the random initialization, pixels often require several (4 times in [40]) global PatchMatch optimizations to get the accurate candidate to converge, which are the major computation cost contribute to MVS.

Fast local propagation from SFM points. Instead, we start from high confident SFM points in each image as activate key points p_{act} to propagate the information to surrounding neighbors. We randomly select eight neighboring pixels for each p_{act} within a 11*11 pixel area as candidate key points p_{cand}. Next, we perform PatchMatch operation on the p_{cand}. The p_{cand} become a new p_{act} when they satisfy that the distance of the p_{cand} to the corresponding p_{act} is less than the given reconstruction accuracy.

In this way, if the p_{cand} is in the similar plane with the p_{act}, it immediately receive the its accurate candidate geometry from p_{act} which will be most likely selected from one-step PatchMatch operation with a minimal photo-consistency loss comparing to other neighboring geometric candidates.

When the activated key point is determined, we design a skip propagation strategy to further propagation by generating a neighbor mask from the activated key as shown in Figure 5. No new key point would be sampled within this mask. This is to mitigate the incorrect propagation to the outside of the plane across the boundary. When no p_{act} exists, we perform the PatchMatch operation for all pixels that are not sampled.

Our propagation strategy ensures every pixel to perform PatchMatch operation once to speed up MVS to more than 4 times. As shown in Figure 4, Our reconstruction aims to reconstruction high confident large planar geometries from SFM points where large shape-radiance ambiguities more likely occur, and leaves non-planar geometries, such as trees and fine details, where NSR methods can reconstruct better.

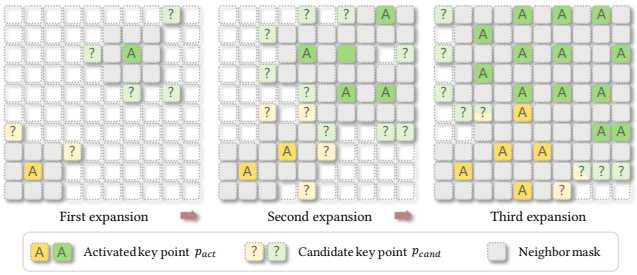


Figure 5: The propagation strategy of our Fast PatchMatch MVS module. The high-confidence geometric information is progressively propagated to its surrounding area.

4 EXPERIMENTS

4.1 Experimental Setup

Baselines. Our experiments are conducted on Urbanscene3D [12], Mill19 [27] and Songshanhu which is collected by our drone. Their areas are between $60000m^2(300m \times 200m)$ and $150000m^2(300m \times 500m)$. We divided the whole scenes into several blocks and each block covers a 150m x 150m ground region. We compare MegaSurf with ACMH [40], a traditional reconstruction method, and two NSR methods: Bakedangelo [46] and Monoangelo. Bakedangelo combines BakedSDF [45] with Neuralangelo [11] settings and has a better background modeling, which is more efficient than Neuralangelo. We migrate the key ideas of MonoSDF [48] to Bakedangelo which called Monoangelo, as the results obtained by MonoSDF are generally oversmooth.

We train MegaSurf for 200k iterations per block (step1: 10k, step2 10k, step3: 180k). The memory consumption is about 22G. The efficiency is basically the same as Bakedangelo [46]. The weights of Curvature loss, Eikonal loss, and L_{nooc} are all 1e-3; the others are all 1. After NSR training, we extract the mesh from the SDF by Marching Cube [16]. We compared the reconstruction results of SciArt and Polytech with the LiDAR ground truth following the official evaluation protocol.

4.2 Comparisons

We developed our Fast PatchMatch MVS module on ACMH software [40], which claims the better quality, and three time speed than another popular open source software, COLMAP [23]. We project the high-confidence geometric cues obtained by our Fast PatchMatch MVS module to the 3D space to form a point cloud and compare it with ACMH.

Table 1: Quantitative results of generating the priors of our Fast PatchMatch MVS module vs ACMH.

Method	$Acc_{50}\downarrow$	$Comp_{50}\downarrow$	$Overall_{50}\downarrow$	$Acc_{95}\downarrow$	$Comp_{95}\downarrow$	$Overall_{95}\downarrow$
Artsci						
ACMH	0.1566	0.1432	0.1499	0.2035	0.3663	0.2849
Ours	0.1629	0.1320	0.1475	0.2010	0.3832	0.2921
Polytech						
ACMH	0.1021	0.1043	0.1032	0.1701	0.2300	0.2000
Ours	0.1227	0.1218	0.1222	0.1937	0.2704	0.2320

Table 1 shows that our module is comparable to ACMH. We report evaluation results for the top 50% accuracy and 95% accuracy points to reduce the influence of noise. Only in the PolyTech dataset, our method may has less detailed structures. However, our method is more than four times faster than ACMH when only the PatchMatch step is counted (2). This matches the configuration of ACMH, which applies four times PatchMatch global sweeps on each pixel. Furthermore, ACMH requires a depth fusion step to filter noisy geometries for the final geometric cues. This step is extremely slow when a large number of images are applied due to their naive implementation, which is not counted in our table. Note that we do not need this fusion step and can also get comparable geometries with reliable masks.

Table 2: The time consumption for generating the cues of our Fast PatchMatch MVS module vs ACMH.

	Residence	SciArt	PolyTech	Songshanhu
Image number	2581	3091	2508	738
Image size	1216×912	1216×912	1500×1000	1368×768
ACMH PatchMatch time	5891s	7274s	7641s	1200s
Ours	1133s	1357s	1460s	291s

We provide qualitative and quantitative comparisons to evaluate the performance of our method. Fig 6 and Table 3 shows the results respectively. We achieved the best results in terms of the Chamfer Distance (*CD*) and Overall score (*Overall*).

Bakedangelo can generate realistic details but suffers inherent shape-radiance ambiguity due to the lack of geometric constraints, often leading to incorrect geometry. Traditional methods such as ACMH are stable in large scene reconstruction. However, due to the large amount of noise in point clouds, the triangulation may incorrectly connect the points and cause over-smoothing. Monoangelo takes depth priors as a regular term to guide the NSR optimization. The depth provided by MVS can help Monoangelo overcome shape-radiance ambiguity, but the noise in priors makes it difficult to reconstruct the fine geometric details. Our MegaSurf utilizes the LG Guider to learn accurate geometric knowledge and prioritize fitting input colors to guide certain regions of the radiance field, thereby overcoming the shape-radiance ambiguity present in Bakeangelo. Additionally, since MegaSurf does not directly employ inaccurate geometry loss and the LG Guider can continuously self-optimize based on rendering loss during training, the given sampling positions become more precise, resulting in finer details than Monoangelo.

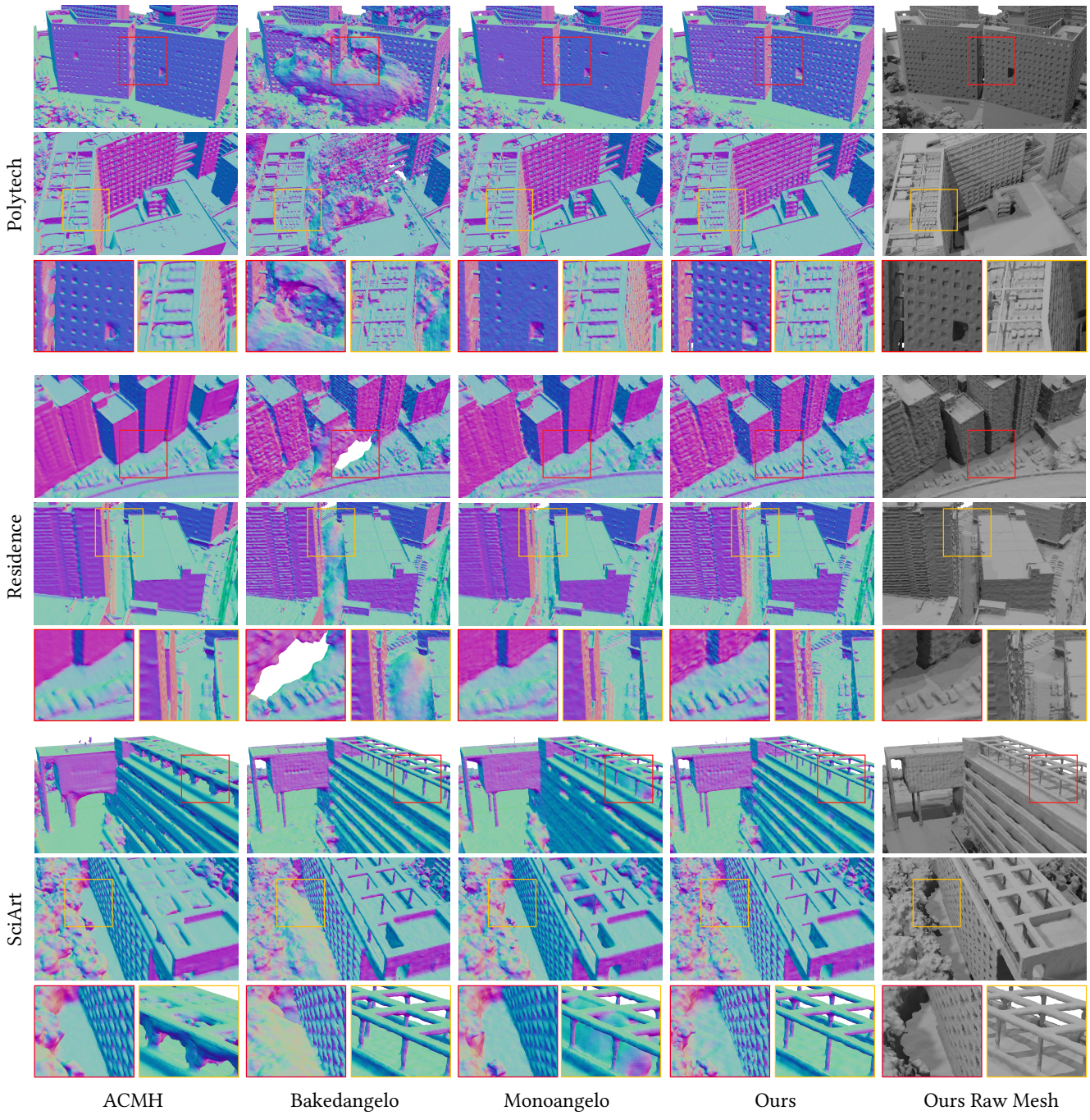


Figure 6: Qualitative results on the Urbanscene3D dataset. MegaSurf both have the robustness to the severe shape-radiance ambiguity and preserve high-fidelity details. The first four columns show the Normal of the corresponding mesh. (Polytech: 12 blocks, Residence: 16 blocks, SciArt: 12 blocks)

4.3 Ablations

We perform ablation experiments over several MegaSurf training strategies. The experiment was conducted on Urbanscene3D. The qualitative and quantitative evaluation results are shown in Fig 7 and Table 3, respectively.

LG Guider. We freeze the parameters of the LG Guider (Freeze *Prop*) after step2 training, the sampling position given by LG Guider can no longer vary. When the parameters of the LG Guider are not

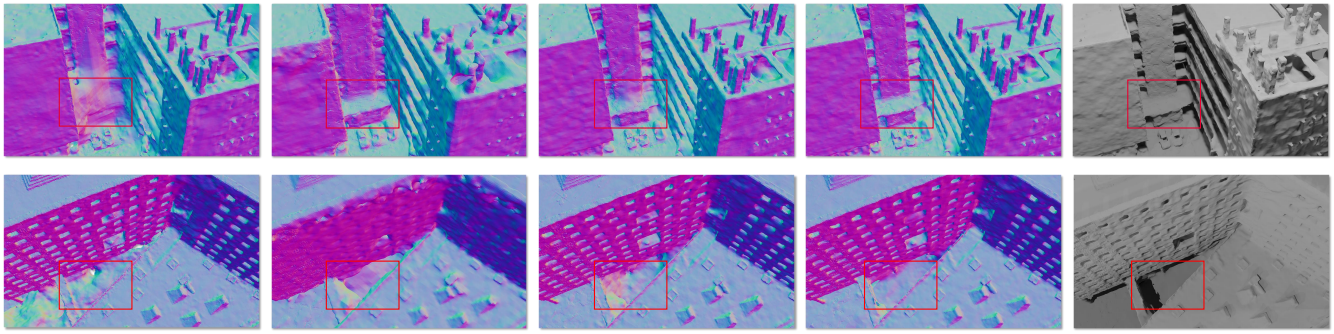


Figure 7: Visualization results of the ablation study.

Table 3: Quantitative evaluation of reconstruction with existing methods on the Urbanscene3D dataset. MegaSurf achieves the best surface reconstruction performance.

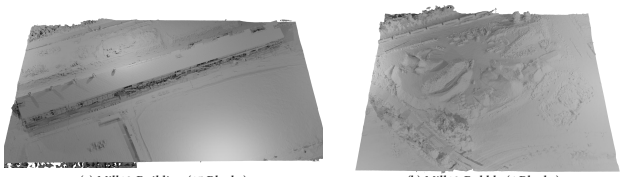
Method	CD ↓	Acc_{95} ↓	$Comp_{95}$ ↓	$Overall_{95}$ ↓
Artsci				
ACMH	1.1675	0.2958	0.5136	0.4047
Bakedangelo	1.3938	0.3319	0.5813	0.4566
Monoangelo	1.4142	0.3778	0.6152	0.4965
Ours	1.0574	0.2990	0.4138	0.3564
Polytech				
ACMH	0.6913	0.1588	0.2499	0.2044
Bakedangelo	1.1029	0.2989	0.3969	0.3479
Monoangelo	0.7414	0.1810	0.2472	0.2141
Ours	0.6593	0.1763	0.2086	0.1925

affected by the rendering loss of step 3, we found that the ambiguity is somewhat alleviated. This because the LG Guider has already learned the geometric information at step1. However, LG Guider loss its ability to refine its sampling field during training, it is difficult to focus around the real surface for finer reconstruction resulting in over-smoothing.

Non occupancy loss L_{nooc}. L_{nocc} is designed to prevent the new surface from appearing in areas where σ should be smaller according to the reliable geometric information when we take rendering loss at optimization step 3. When L_{nooc} is removed, we can see that the scene has some raised surfaces at some corner regions which is easily suffers the ambiguity.

Table 4: Quantitative results of the ablation study on the Urbanscene3D dataset.

Method	CD ↓	Acc_{95} ↓	$Comp_{95}$ ↓	$Overall_{95}$ ↓
Artsci				
Base	1.3938	0.3319	0.5813	0.4566
Freeze *Prop*	1.4585	0.3736	0.6982	0.5359
No L_{nocc}	1.1322	0.2980	0.4649	0.3815
Full	1.0574	0.2990	0.4138	0.3564
Polytech				
Base	1.1029	0.2989	0.3969	0.3479
Freeze *Prop*	0.8350	0.2218	0.3182	0.2700
No L_{nocc}	0.6782	0.1749	0.2120	0.1935
Full	0.6593	0.1763	0.2086	0.1925



(a) Mill19-Building (15 Blocks) (b) Mill19-Rubble (6 Blocks)

Figure 8: The reconstruction results of Mill19 by MegaSurf. For more cases, please refer to the Supplementary.

5 LIMITATIONS AND FUTURE WORK

Due to the high reconstruction accuracy of MegaSurf, seams are usually imperceptible when assembling all the blocks together. However, in some cases, seams may still be perceptible. Applying mesh refinement to the assembled model can effectively overcome this issue. Furthermore, we also need to fine-tune the parameters of MegaSurf or consider alternatives such as 3D Gaussian [7, 8, 47] to replace the neural radiance field, in order to improve the efficiency. For extremely thin surfaces in large scenes, NSR struggles to represent the rapidly changing SDF field. Transforming the SDF into a UDF [14, 15] or using rendered depth by NSR followed by depth fusion [1, 20] might improve detailed reconstruction in large scenes.

6 CONCLUSION

We introduce MegaSurf, a Learnable Sampling Guided surface reconstruction approach for reconstructing large-scale scenes. We propose a Fast PatchMatch MVS module to progressively propagate the SFM information to its surrounding area to obtain high-confidence geometric cues, which is proven to be more than four times faster than the SOTA method. Then, we propose a Learnable Geometric Guider (LG Guider) to learn a sampling field from reliable geometric cues and can be continuously refined by the rendering loss. And we also propose a Divide-and-Conquer training strategy to synchronize the LG Guider and the radiance field, which make the NSR efficiently overcome the shape-radiance ambiguity while preserving the high-fidelity details. Experiments on large-scale scene datasets show our SOTA performance.

REFERENCES

[1] Angela Dai, Matthias Nießner, Michael Zollöfer, Shahram Izadi, and Christian Theobalt. 2017. BundleFusion: Real-time Globally Consistent 3D Reconstruction using On-the-fly Surface Re-integration. *ACM Transactions on Graphics 2017 (TOG)* (2017).

[2] Qiancheng Fu, Qingshan Xu, Yew Soon Ong, and Wenbing Tao. 2022. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *Advances in Neural Information Processing Systems* 35 (2022), 3403–3416.

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

[4] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. 2020. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2495–2504.

[5] Jianfei Guo, Nianchen Deng, Xinyang Li, Yeqi Bai, Botian Shi, Chiyu Wang, Chenjing Ding, Dongliang Wang, and Yikang Li. 2023. StreetSurf: Extending Multi-view Implicit Surface Reconstruction to Street Views. *arXiv preprint arXiv:2306.04988* (2023).

[6] Heiko Hirschmuller. 2005. Accurate and efficient stereo processing by semi-global matching and mutual information. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 2. IEEE, 807–814.

[7] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2024. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. *arXiv preprint arXiv:2403.17888* (2024).

[8] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics* 42, 4 (2023), 1–14.

[9] Jingliang Li, Zhengda Lu, Yiqun Wang, Ying Wang, and Jun Xiao. 2022. Ds-mvsnet: Unsupervised multi-view stereo via depth synthesis. In *Proceedings of the 30th ACM International Conference on Multimedia*. 5593–5601.

[10] Zhuopeng Li, Lu Li, and Jianke Zhu. 2023. Read: Large-scale neural scene rendering for autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 1522–1529.

[11] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. 2023. Neuralangelo: High-Fidelity Neural Surface Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8456–8465.

[12] Liqiang Lin, Yilin Liu, Yue Hu, Xingguang Yan, Ke Xie, and Hui Huang. 2022. Capturing, Reconstructing, and Simulating: the UrbanScene3D Dataset. In *ECCV*. 93–109.

[13] Tianqi Liu, Xinyi Ye, Weiyue Zhao, Zhiyu Pan, Min Shi, and Zhiguo Cao. 2023. When epipolar constraint meets non-local operators in multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 18088–18097.

[14] Yu-Tao Liu, Li Wang, Jie Yang, Weikai Chen, Xiaoxu Meng, Bo Yang, and Lin Gao. 2023. Neudf: Leaning neural unsigned distance fields with volume rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 237–247.

[15] Xiaoxiao Long, Cheng Lin, Lingjie Liu, Yuan Liu, Peng Wang, Christian Theobalt, Taku Komura, and Wenping Wang. 2023. Neuraludf: Learning unsigned distance fields for multi-view reconstruction of surfaces with arbitrary topologies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 20834–20843.

[16] William E Lorensen and Harvey E Cline. 1987. Marching cubes: A high resolution 3D surface construction algorithm. *ACM siggraph computer graphics* 21, 4 (1987), 163–169.

[17] Zeyu Ma, Zachary Teed, and Jia Deng. 2022. Multiview stereo with cascaded epipolar raft. In *European Conference on Computer Vision*. Springer, 734–750.

[18] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2020. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*. Springer, 405–421.

[19] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)* 41, 4 (2022), 1–15.

[20] Christian Reiser, Stephan Garbin, Pratul P Srinivasan, Dor Verbin, Richard Szeliski, Ben Mildenhall, Jonathan T Barron, Peter Hedman, and Andreas Geiger. 2024. Binary Opacity Grids: Capturing Fine Geometric Detail for Mesh-Based View Synthesis. *arXiv preprint arXiv:2402.12377* (2024).

[21] Konstantinos Rematas, Andrew Liu, Pratul P. Srinivasan, Jonathan T. Barron, Andrea Tagliasacchi, Tom Funkhouser, and Vittorio Ferrari. 2022. Urban Radiance Fields. *CVPR* (2022).

[22] Radu Alexandru Rosu and Sven Behnke. 2023. Permutosdf: Fast multi-view reconstruction with implicit surfaces using permutohedral lattices. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8466–8475.

[23] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. 2016. Pixelwise view selection for unstructured multi-view stereo. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*. Springer, 501–518.

[24] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. 2006. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, Vol. 1. IEEE, 519–528.

[25] Jiaming Sun, Xi Chen, Qianqian Wang, Zhengqi Li, Hadar Averbuch-Elor, Xiaowei Zhou, and Noah Snavely. 2022. Neural 3d reconstruction in the wild. In *ACM SIGGRAPH 2022 Conference Proceedings*. 1–9.

[26] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. 2022. Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8248–8258.

[27] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. 2022. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12922–12931.

[28] Chen Wang, Xian Wu, Yuan-Chen Guo, Song-Hai Zhang, Yu-Wing Tai, and Shi-Min Hu. 2022. NeRF-SR: High-Quality Neural Radiance Fields using Super-sampling. In *Proceedings of the 30th ACM International Conference on Multimedia*. 6445–6454.

[29] Jiepeng Wang, Peng Wang, Xiaoxiao Long, Christian Theobalt, Taku Komura, Lingjie Liu, and Wenping Wang. 2022. Neuris: Neural reconstruction of indoor scenes using normal priors. In *European Conference on Computer Vision*. Springer, 139–155.

[30] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689* (2021).

[31] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. 2023. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3295–3306.

[32] Yusen Wang, Zongcheng Li, Yu Jiang, Kaixuan Zhou, Tuo Cao, Yanping Fu, and Chunxia Xiao. 2022. NeuralRoom: Geometry-Constrained Neural Implicit Surfaces for Indoor Scene Reconstruction. *ACM Transactions on Graphics (TOG)* 41, 6 (2022), 1–15.

[33] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. 2022. Hf-neus: Improved surface reconstruction using high-frequency details. *Advances in Neural Information Processing Systems* 35 (2022), 1966–1978.

[34] Zian Wang, Tianchang Shen, Jun Gao, Shengyu Huang, Jacob Munkberg, Jon Hasselgren, Zan Gojcic, Wenzheng Chen, and Sanja Fidler. 2023. Neural Fields meet Explicit Geometric Representations for Inverse Rendering of Urban Scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8370–8380.

[35] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. 2021. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5610–5619.

[36] Tong Wu, Jiaqi Wang, Xingang Pan, XU Xudong, Christian Theobalt, Ziwei Liu, and Dahua Lin. 2022. Voxurf: Voxel-based Efficient and Accurate Neural Surface Reconstruction. In *The Eleventh International Conference on Learning Representations*.

[37] Yuanbo Xiangli, Linning Xu, Xingang Pan, Nanxuan Zhao, Anyi Rao, Christian Theobalt, Bo Dai, and Dahua Lin. 2022. BungeeNeRF: Progressive Neural Radiance Field for Extreme Multi-scale Scene Rendering. In *The European Conference on Computer Vision (ECCV)*.

[38] Luoyuan Xu, Tao Guan, Yuesong Wang, Yawei Luo, Zhuo Chen, Wenkai Liu, and Wei Yang. 2022. Self-supervised multi-view stereo via adjacent geometry guided volume completion. In *Proceedings of the 30th ACM International Conference on Multimedia*. 2202–2210.

[39] Linning Xu, Yuanbo Xiangli, Sida Peng, Xingang Pan, Nanxuan Zhao, Christian Theobalt, Bo Dai, and Dahua Lin. 2023. Grid-guided Neural Radiance Fields for Large Urban Scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8296–8306.

[40] Qingshan Xu, Weihang Kong, Wenbing Tao, and Marc Pollefeys. 2022. Multi-scale geometric consistency guided and planar prior assisted multi-view stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 4 (2022), 4945–4963.

[41] Qingshan Xu and Wenbing Tao. 2019. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5483–5492.

[42] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. 2022. Point-NeRF: Point-based Neural Radiance Fields. *arXiv preprint arXiv:2201.08845* (2022).

[43] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. 2018. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 767–783.

[44] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. 2021. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems* 34 (2021), 4805–4815.

[45] Lior Yariv, Peter Hedman, Christian Reiser, Dor Verbin, Pratul P. Srinivasan, Richard Szeliski, Jonathan T. Barron, and Ben Mildenhall. 2023. BakedSDF: Meshing Neural SDFs for Real-Time View Synthesis. *arXiv* (2023).

[46] Zehao Yu, Anpei Chen, Bozidar Antic, Songyou Peng, Apratim Bhattacharyya, Michael Niemeyer, Siyu Tang, Torsten Sattler, and Andreas Geiger. 2022. SDFStudio: A Unified Framework for Surface Reconstruction. https://github.com/autonomousvision/sdfstudio

[47] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. 2023. Mip-Splatting: Alias-free 3D Gaussian Splatting. *arXiv:2311.16493* (2023).

[48] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. 2022. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in neural information processing systems* 35 (2022), 25018–25032.

[49] Junyi Zeng, Chong Bao, Rui Chen, Zilong Dong, Guofeng Zhang, Hujun Bao, and Zhaopeng Cui. 2023. Mirror-NeRF: Learning Neural Radiance Fields for Mirrors with Whitted-Style Ray Tracing. In *Proceedings of the 31st ACM International Conference on Multimedia*. 4606–4615.

[50] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. 2020. Visibility-aware multi-view stereo network. *arXiv preprint arXiv:2008.07928* (2020).