

Neural 3D Reconstruction in the Wild

Jiaming Sun
jiaming@idr.ai
Image Derivative Inc.
China

Xi Chen
cxi@zju.edu.cn
Zhejiang University
China

Qianqian Wang
qw246@cornell.edu
Cornell Tech, Cornell University
USA

Zhengqi Li*
zl548@g.cornell.edu
Cornell Tech, Cornell University
USA

Hadar Averbuch-Elor
hadarelor@cornell.edu
Cornell Tech, Cornell University
USA

Xiaowei Zhou†
xwzhou@zju.edu.cn
Zhejiang University
China

Noah Snavely
snavey@cornell.edu
Cornell Tech, Cornell University
USA

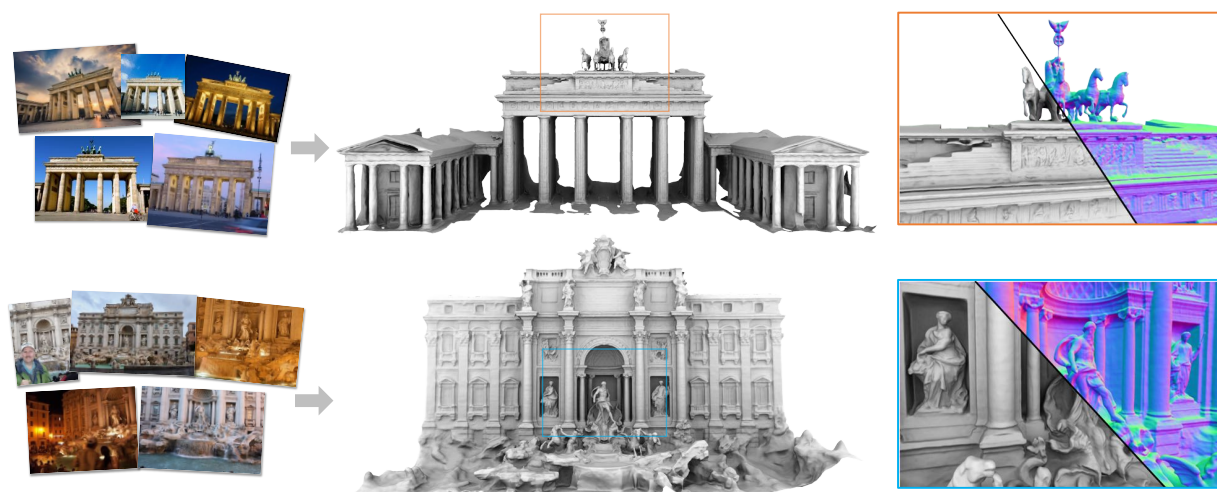


Figure 1: Neural 3D reconstruction in the wild. Given a large number of Internet photos capturing popular tourist attractions (left), our approach learns to produce high quality 3D surface reconstruction, efficiently modeling entire 3D scenes with a novel hybrid neural implicit representation (right). Colors indicate surface normals. Please zoom in to see details in the geometry. Photos by Flickr users rickz, Infinite Ache, jon collier, Modes Rodríguez, Ashwin Kumar, Richard Cyganiak, David Lebech, Matt Drobnik, Scott, Naval S. under CC-BY.

ABSTRACT

We are witnessing an explosion of neural implicit representations in computer vision and graphics. Their applicability has recently expanded beyond tasks such as shape generation and image-based

rendering to the fundamental problem of image-based 3D reconstruction. However, existing methods typically assume constrained 3D environments with constant illumination captured by a small set of roughly uniformly distributed cameras. We introduce a new method that enables efficient and accurate surface reconstruction from Internet photo collections in the presence of varying illumination. To achieve this, we propose a hybrid voxel- and surface-guided sampling technique that allows for more efficient ray sampling around surfaces and leads to significant improvements in reconstruction quality. Further, we present a new benchmark and protocol for evaluating reconstruction performance on such in-the-wild scenes. We perform extensive experiments, demonstrating that our approach surpasses both classical and neural reconstruction methods on a wide variety of metrics. Code and data will be made available at <https://zju3dv.github.io/neuralrecon-w>.

*Current address: Google Research.

†Corresponding author. Affiliated with State Key Lab of CAD&CG, Zhejiang Univ.

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 ACM 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.

SIGGRAPH '22 Conference Proceedings, August 7–11, 2022, Vancouver, BC, Canada

© 2022 Association for Computing Machinery.

ACM ISBN 978-1-4503-9337-9/22/08...\$15.00

<https://doi.org/10.1145/3528233.3530718>

CCS CONCEPTS

• **Computing methodologies** → **Reconstruction**; *Mesh geometry models*.

KEYWORDS

3D Reconstruction, Novel View Synthesis, Neural Rendering

ACM Reference Format:

Jiaming Sun, Xi Chen, Qianqian Wang, Zhengqi Li, Hadar Averbuch-Elor, Xiaowei Zhou, and Noah Snavely. 2022. Neural 3D Reconstruction in the Wild. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings (SIGGRAPH '22 Conference Proceedings)*, August 7–11, 2022, Vancouver, BC, Canada. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3528233.3530718>

1 INTRODUCTION

Reconstructing a 3D mesh from a collection of 2D images is a long-standing goal in computer vision and graphics. Neural field-based 3D reconstruction methods (e.g., [Oechsle et al. 2021; Wang et al. 2021; Yariv et al. 2021]) have recently gained traction as they can reconstruct high-fidelity meshes for both objects and scenes, surpassing the quality of traditional reconstruction pipelines [Cernea 2020; Schönberger et al. 2016]. But these methods are often demonstrated in controlled capture settings. Consider the Internet photo collections of the Brandenburg Gate and Trevi Fountain shown in Fig. 1. Can neural 3D reconstruction techniques apply to real-world, unconstrained Internet datasets like these? Handling such data requires both scalability and robustness to highly diverse imagery.

Unlike standard 3D reconstruction datasets that typically contain tens of images (e.g., the DTU dataset [Jensen et al. 2014] provides 49 or 64 images per scene), Internet datasets can contain hundreds or thousands of images. Neural 3D reconstruction techniques must process such image collections efficiently, without sacrificing accuracy in complex scenes featuring geometric detail of varying granularity. Beyond scalability to large scenes and image collections, existing reconstruction methods typically assume constant illumination and leverage photometric consistency across the input images. In contrast, for in-the-wild scenes, robustness to appearance variation is another key requirement.

In this work, we present an approach that can efficiently reconstruct surface geometry for large-scale scenes in the presence of varying illumination. Inspired by Neural Radiance Fields in the Wild (NeRF-W) [Martin-Brualla et al. 2021], we model appearance variation using appearance embeddings, but seek meshes as output, rather than radiance fields as in that work. Meshes, unlike raw radiance fields, provide a direct representation of the scene’s geometry and can be readily imported into standard graphics pipelines. To reconstruct such surface geometry, we leverage volume rendering methods as in NeuS [Wang et al. 2021], coupling a neural surface representation with volumetric rendering. However, a straightforward integration of the surface representation presented in NeuS with a volumetric radiance field that models appearance variations involves huge compute demands for large-scale Internet collections, and is intractable in settings with limited access to high-end GPUs. For each scene we consider, training using this integrated framework on 32 GPUs converges after roughly ten days.

We therefore propose a hybrid voxel- and surface-guided sampling technique. We observe that the standard ray sampling strategy for optimizing neural radiance fields is highly redundant (Figure 2a). To reduce redundant training samples, we first leverage the sparse point clouds from structure-from-motion (SfM) to initialize a sparse volume from which samples are generated (Figure 2b). We then combine this voxel-guided strategy with a surface-guided sampling technique which generates samples based on the current state of optimization (Figure 2c). Our key insight here is to not only use the SfM point clouds, but also our surface approximation, yielding new samples that are centered around the true surface. This strategy guides the network to explain the rendered color with near-surface samples, leading to more accurate geometric fitting.

Finally, while established benchmarks and evaluation schemes exist for controlled datasets, such benchmarks with ground truth geometry do not exist for Internet collections. Therefore, we introduce *Heritage-Recon*, a new benchmark dataset derived from the public catalog of free-licensed LiDAR data available in Open Heritage 3D, a repository of open 3D cultural heritage assets.¹ We pair this unique data source with Internet-derived image collections and SfM models from the MegaDepth dataset [Li and Snavely 2018], performing additional processing steps such as model alignment and visibility checking. We also carefully design an evaluation protocol suited for such large-scale scenes with incomplete ground truth (as even LiDAR scans may not cover the entirety of a scene visible from imagery). Evaluating on Heritage-Recon, we demonstrate that our approach surpasses classical and neural reconstruction methods in terms of efficiency and accuracy.

2 RELATED WORK

Image-based 3D reconstruction aims at estimating the most likely 3D shape (and possibly appearance) of an object or a scene given a set of captured 2D photos. In this section, we summarize prior work ranging from classical to modern approaches, highlighting work most closely related to our own.

Multi-view reconstruction. Multi-view 3D reconstruction methods take images and estimate geometry with a variety of representations, including point clouds, depth maps, meshes, or volumetric implicit functions [Furukawa and Hernández 2015]. Many classical multi-view stereo (MVS) methods reconstruct geometry by estimating a depth map for each image followed by depth fusion to obtain dense point clouds [Furukawa and Ponce 2009; Goesele et al. 2006; Hedman et al. 2017; Schönberger et al. 2016]. Surface reconstruction algorithms such as Poisson reconstruction [Kazhdan et al. 2006; Kazhdan and Hoppe 2013] and Delaunay triangulation [Labatut et al. 2009] can then be applied to these point clouds to produce meshes. Recently, learning-based multiview depth estimation methods have achieved state-of-the-art performance on numerous benchmarks by taking advantage of data-driven priors and physical constraints [Darmon et al. 2021; Gu et al. 2020; Huang et al. 2018; Liu et al. 2019; Yao et al. 2018, 2019]. Since these methods perform depth estimation, point cloud fusion and mesh extraction stages separately, they are sensitive to outliers or inconsistencies in depth maps and can yield noisy or incomplete reconstructions. In

¹ <https://openheritage3d.org/>

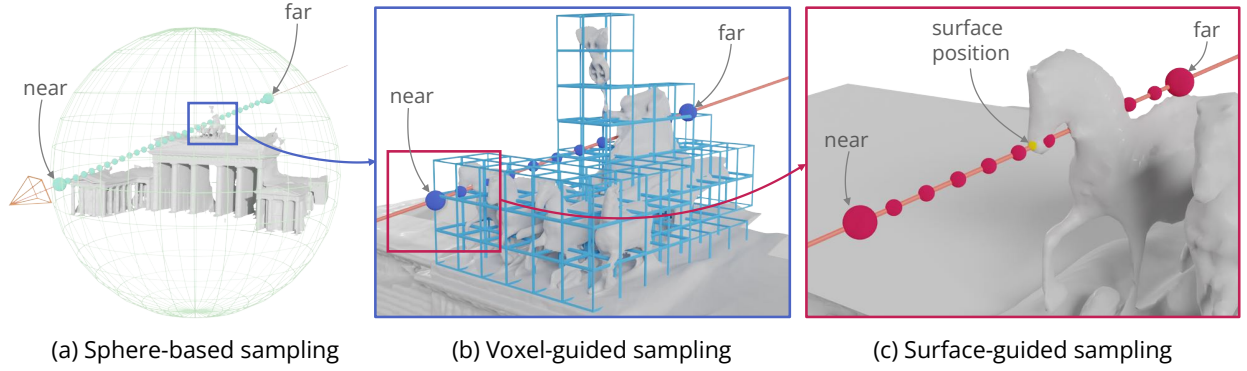


Figure 2: Comparison between sphere-based sampling and our proposed sampling strategy. Sphere-based sampling (a), used in NeuS, generates samples scattered throughout the unit sphere and spanning the whole scene, with the result that most samples lie in empty regions and are hence unnecessary. We propose voxel-guided sampling (b) to avoid unnecessary samples by sampling only within a sparse voxel volume around surfaces estimated from SfM point clouds (only a subset of voxels are shown for clarity). To further increase the sampling density around surfaces, we additionally propose a surface-guided sampling strategy (c), where we store SDF values from previous training iterations in the sparse voxels, and generate samples within a smaller range centered around the estimated surface positions. Note that each successive region of the volume considered by each sampling strategy from (a) to (b) to (c) is progressively smaller as suggested by the 2D blue and red bounding boxes.

contrast, our approach models the full scene using a global representation and optimizes both appearance and geometry end-to-end via neural rendering. Other methods also directly estimate the 3D surfaces [Božić et al. 2021; Murez et al. 2020; Sun et al. 2021b], but require ground-truth 3D reconstructions for training and cannot generalize beyond the training data (e.g., from indoor scenes to outdoor scenes) due to their data-dependent nature.

Reconstruction in the wild. Internet photo collections, especially those capturing tourist attractions around the world, are a popular data source in 3D vision and graphics. Because of their abundance and diversity of viewpoints, appearance, and geometry, prior research has used such data in a range of problems, including SfM [Agarwal et al. 2011; Schonberger and Frahm 2016; Snavely et al. 2006], MVS [Frahm et al. 2010; Furukawa et al. 2010; Goesele et al. 2007; Schönberger et al. 2016], time-lapse reconstruction [Martin-Brualla et al. 2015a,b; Matzen and Snavely 2014] and appearance modeling [Kim et al. 2016; Li et al. 2020; Martin-Brualla et al. 2021; Meshry et al. 2019; Shan et al. 2013]. In particular, NeRF-W [Martin-Brualla et al. 2021] models the scene with neural radiance fields, which are suitable for synthesizing novel-view images but cannot produce high-quality surface reconstructions. In contrast, our approach models the scene with a surface-based representation and directly produces smooth and accurate 3D meshes.

Neural implicit representations. Neural implicit representations have recently shown great promise for 3D modeling due to their intrinsic global consistency and continuous nature. These properties allow for efficient representation of scene appearance and geometry with a high degree of detail. These representations have been applied to a variety of applications including shape generation and completion [Chabra et al. 2020; Mescheder et al. 2019; Park et al. 2019; Peng et al. 2020], novel view synthesis [Li et al. 2021; Mildenhall et al. 2020; Park et al. 2021; Wizarwongsa et al. 2021],

camera pose estimation [Lin et al. 2021; Yen-Chen et al. 2021], and intrinsic decomposition [Boss et al. 2021a,b; Zhang et al. 2021b,a].

Most neural implicit representations are optimized from 2D images using differentiable rendering, and can be roughly categorized into two types: surface rendering (e.g., [Niemeyer et al. 2020; Yariv et al. 2020]) and volume rendering (e.g., [Kondo et al. 2021; Martin-Brualla et al. 2021; Mildenhall et al. 2020], see [Dellaert and Yen-Chen 2020] for a comprehensive survey). While surface rendering methods allow for more accurate modeling of geometry, most prior methods require additional constraints, such as ground truth masks. On the other hand, volume rendering techniques have shown impressive results for image-based rendering of complex scenes, but due to their soft volumetric properties it is hard to extract accurate surface geometry from such representations. More recently, several methods unify surface and volumetric representation, enabling reconstruction of accurate surfaces without requiring masks [Oechsle et al. 2021; Wang et al. 2021; Yariv et al. 2021]. In our work, we extend the representation proposed in NeuS [Wang et al. 2021] such that it can accommodate unconstrained Internet photo collections.

In contrast to recent work on landmark- or city-scale neural rendering [Martin-Brualla et al. 2021; Rematas et al. 2022; Xiangli et al. 2021], which mainly addresses novel view synthesis, our work focuses on modeling geometry from unstructured Internet photos. We demonstrate that our approach enables efficient, accurate, and highly detailed surface reconstructions of landmarks.

3 METHOD

To model the shape and appearance of a 3D scene, we propose an approach inspired by recent work on neural radiance fields that can reconstruct a 3D scene as the weights of a neural network by optimizing for image reconstruction losses [Mildenhall et al. 2020]. In particular, we use the latent appearance modeling introduced in NeRF in the Wild (NeRF-W) [Martin-Brualla et al. 2021] to model

3D scenes from unconstrained Internet collections with varying lighting. Furthermore, to model accurate surface geometry, we extend the scene representation proposed in NeuS [Wang et al. 2021] and represent the scene using two neural implicit functions d and c_i encoded by MLPs. Given a point $\mathbf{x} \in \mathbb{R}^3$ in the scene, a viewing direction $\mathbf{v} \in \mathbb{S}^2$ and an image index i , we have:

$$d = \text{MLP}_{\text{SDF}}(\mathbf{x}), \quad (1)$$

$$c_i = \text{MLP}_{\text{COLOR}}(\mathbf{x}, \mathbf{v}, \mathbf{e}_i), \quad (2)$$

where $\{\mathbf{e}_i\}_{i=1}^N$ are appearance embeddings corresponding to each input photo, optimized alongside the parameters of MLPs.

We use the function d to approximate the signed distance to the true surface S , represented as the zero level set of this function:

$$S = \{\mathbf{x} | d(\mathbf{x}) = 0\}. \quad (3)$$

The function c_i models the appearance of 3D point \mathbf{x} as it appears in a given image i , allowing for the varying appearance of each input image. The MLP parameters and appearance embeddings are learned by optimizing color consistency between real photos and rendered images via a volume rendering scheme: Given a ray, $\{\mathbf{r}(t) = \mathbf{o} + t\mathbf{v} | t \geq 0\}$ with \mathbf{o} denoting the camera center, we can render that ray’s expected color $\hat{C}_i(\mathbf{r})$ corresponding to image i as:

$$\hat{C}_i(\mathbf{r}) = \int_0^{+\infty} w(t) c_i(\mathbf{r}(t), \mathbf{v}, \mathbf{e}_i) dt, \quad (4)$$

where $w(t)$ is an unbiased and occlusion-aware weight function, as further detailed in [Wang et al. 2021].

Note that dynamic objects, which are prominent in Internet collections, can significantly impact model performance. The model proposed in NeRF-W [Martin-Brualla et al. 2021], for instance, incorporates a transient head to distinguish between static and dynamic parts of the scene. To reconstruct accurate geometry, a different approach is required, as the transient head dominates the rendered color, leading to all scene structures modeled as view-dependent transient effects rather than geometry. We discuss our solution and additional design choices in Section 3.2.

3.1 Efficient Sampling during Training

NeuS uses a hierarchical importance sampling strategy to generate sample points on each ray during the optimization phase. For each scene, NeuS defines a unit bounding sphere to separate the background and foreground parts of the scene. The coarse sample points are sampled regularly along a ray between the two intersection points of the ray and the bounding sphere. Fine-level samples are iteratively generated based on samples from the previous iteration.

This simple strategy works well on lab-captured datasets like DTU, where the camera views are distributed uniformly on a hemisphere. However, it is extremely inefficient for “in-the-wild” scenarios with large-scale scenes, where the camera views are distributed non-uniformly and are often front-facing. To give an example, if we were to follow NeRF-W, which uses 1024 samples for both the coarse and fine levels per ray for training, then using the same number of samples for training the NeuS model would result in an estimated ~10 days of training with 32 GPUs. Instead, we introduce a hybrid voxel- and surface-guided sampling strategy to improve training efficiency, as detailed in the following sections. A visualization of different sampling strategies is shown in Fig. 2.

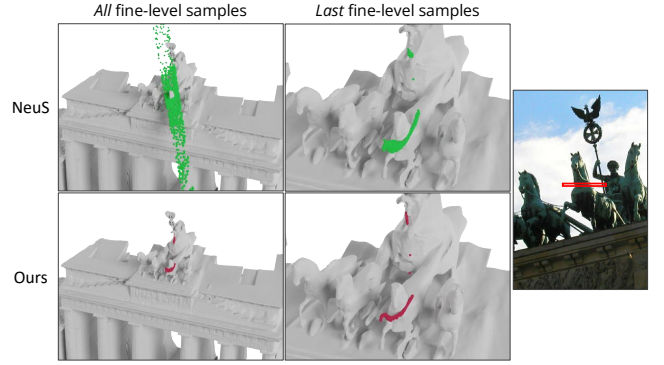


Figure 3: Comparison of fine-level samples from our surface-guided sampling and the sphere-based hierarchical sampling used in NeuS. We visualize samples from rays corresponding to pixels in the red box shown in the image on the right. The hierarchical sampling in NeuS (upper left) uses a redundant set of fine-level samples (1024 in total), while surface-guided sampling (lower left) uses much fewer samples (24 in total). At the last iteration of fine-level sampling, samples from surface-guided sampling (lower right) are denser and closer to the surface than those from NeuS (upper right), guiding the network to fit the surface geometry accurately with more details. The sampled points are generated from trained models. The mesh is shown for clarity only.

Voxel-guided sampling. To speed up training, we first remove unnecessary training samples by reducing the search space from the entire unit sphere to a smaller spatial envelope that contains the true surface position. Specifically, we observe that rough initial surface estimates are provided by the sparse point cloud that SfM produces alongside the estimated camera poses. Therefore, at the start of training we generate a sparse volume \mathbf{V}_{sfm} from the sparse SfM point cloud. This sparse volume is expanded via a 3D dilation operation to ensure that most of the visible surfaces are encompassed by this volume. The sampling range of a given ray can then be reduced to the in- and out-intersection points between each ray and \mathbf{V}_{sfm} , and n_v points are sampled during this stage. We call this sampling technique *voxel-guided sampling*. Related sampling strategies have been explored in recent work [Liu et al. 2020], but rather than pruning from a dense voxel grid, our method makes use of the already available 3D information from SfM as more explicit guidance for reducing the search space for point sampling. Moreover, we found that the constructed sparse voxels provides a rough separation of the scene into foreground and background regions, and by removing rays that do not intersect the sparse voxels (e.g., background rays in the sky), the number of required training rays can often be reduced by over 30%.

Surface-guided sampling. In order to train the geometry MLP d to accurately fit the 3D surface, it is beneficial to generate as many samples around the true surface as possible. NeuS achieves a high sampling density through multiple iterations of fine-level importance sampling, which gradually guides samples towards the surface position. This strategy is time-consuming, since a large number of unnecessary samples must be generated by passing through

the geometry MLP d for multiple iterations. A visual illustration can be found in Fig. 3.

Therefore, we propose a surface-guided sampling strategy that further increases sample density around the true surface. In particular, after the training is bootstrapped by voxel-guided sampling, we leverage the surface position estimates from the previous training iteration to generate new samples. To achieve this, we cache the SDF predictions from previous iterations inside sparse voxels V_{cache} , and query the surface position from this cache at each training iteration. V_{cache} is an octree built upon V_{sfm} with a depth level of ℓ . With the queried surface position \hat{x} , we query a number n_s of samples within a narrower range ($\hat{x} - t_s, \hat{x} + t_s$) around the surface position. V_{cache} is updated periodically during training to ensure that the stored SDF values are up-to-date.

The cached surface positions provide a good approximation of the true surface position, leading the network to improve upon previous estimations. Surface-guided sampling guides the network to explain the rendered color with samples around the true surface position, allowing the network to fit geometry more accurately. As shown in ablation studies, without surface-guided sampling, training is unable to converge to the same degree of accuracy even given sufficient time.

Hybrid sampling. Using only surface-guided sampling will result in artifacts around voxel borders since there is insufficient supervision for empty space. To avoid this problem, we use a hybrid of voxel- and surface-guided sampling. Note that the voxel-guided samples are much sparser than the surface-guided samples since they are generated within a much larger search range. We use another iteration of importance sampling after surface-guided sampling to ensure a good sampling density, bringing the total number of samples along each ray to $n_v + 2 \times n_s$.

3.2 Additional Details

Handling transient objects. We empirically found that, if we directly use the dynamic object modeling strategy proposed in NeRF-W (i.e., a transient NeRF head), the transient NeRF will dominate the rendered color. As a result, all scene structures will be modeled as view-dependent transient effects by NeRF instead of by the geometry MLP d , since d converges more slowly compared to NeRF. We instead use segmentation masks to remove rays belonging to dynamic objects during training.

Supervision signals and handling the textureless sky. Following NeuS, we use an \mathcal{L}_1 loss to supervise the rendered color images ($\mathcal{L}_{\text{COLOR}}$) and use an eikonal term \mathcal{L}_{REG} to regularize the SDF. Since the textureless sky lacks motion parallax, directly using a background NeRF for foreground-background separation as in NeuS will lead to reconstructions contained in spherical shells. The remaining background rays in V_{sfm} (mostly sky) are labeled with semantic masks and penalized as free space with $\mathcal{L}_{\text{MASK}}$. Since the semantic masks for the background are not perfect and often contain foreground scene structures, we only apply $\mathcal{L}_{\text{MASK}}$ with a small weight, which we found empirically can remove the sky while keeping foreground structures intact. Please refer to NeuS for further details on $\mathcal{L}_{\text{COLOR}}$, \mathcal{L}_{REG} and $\mathcal{L}_{\text{MASK}}$.



Figure 4: Alignment quality of the LiDAR scan on Brandenburg Gate. We render the LiDAR-scanned point clouds as depth maps by projecting the points to a set of camera views in the aligned SfM coordinate frame. The rendered depth maps are color-coded by depth (warmer colors are closer) and overlaid with the corresponding images. The accuracy of the alignment can be observed in, for instance, the agreement of image and depth edges.

4 THE HERITAGE-RECON BENCHMARK

To evaluate our method we need ground truth 3D geometry. However, to the best of our knowledge, there is no existing dataset pairing Internet photo collections with ground truth 3D. Therefore, we introduce *Heritage-Recon*, a new benchmark dataset, derived from Open Heritage 3D. We first describe how we constructed the dataset, including how the data was collected and processed (e.g., alignment to the SfM sparse point clouds), and later present the metrics and evaluation protocol used in our experiments.

Data collection and processing. We obtained 3D LiDAR data from Open Heritage 3D, which provides public, freely-licensed 3D scans for hundreds of cultural heritage sites. We select four landmarks, namely *Brandenburg Gate (BG)*², *Pantheon Exterior (PE)*³, *Lincoln Memorial (LM)*⁴ and *Palacio de Bellas Artes (PBA)*⁵ for the dataset. These landmarks were selected because they can be easily paired with Internet image collections. The corresponding images for *BG*, *PE* and *LM* were gathered from the MegaDepth dataset [Li and Snavely 2018]. We collect images for *PBA* from Flickr, following a similar procedure as described in prior works for the other landmarks. The images for each scene are registered using SfM [Schonberger and Frahm 2016] to obtain camera poses and a sparse point clouds. For *BG* and *PBA*, the original LiDAR scans are very dense and are over 100GB in size. These point clouds are downsampled to a density of 2cm. A bounding box is manually selected for each scan as the Region of Interest (ROI) to further reduce the size of the point cloud.

Coordinate alignment. Since the SfM reconstructions and LiDAR scans have different coordinate frames, they must be aligned before evaluation. To align them, we first filter sparse points obtained

²<https://openheritage3d.org/project.php?id=d51v-fq77>

³<https://openheritage3d.org/project.php?id=t9sj-mf53>

⁴<https://openheritage3d.org/project.php?id=90yg-1054>

⁵<https://openheritage3d.org/project.php?id=vdae-mr89>

from SfM by their track length and reprojection error, and align the resulting point cloud to the LiDAR scan using ICP [Rusinkiewicz and Levoy 2001] with carefully tuned parameters. The alignment quality can be visually inspected in Fig. 4. We quantitatively check the alignment quality by reprojecting a set of feature tracks from SfM using depth maps rendered from the LiDAR scan. We observe that the resulting reprojection error is less than one pixel across all the scenes, an accuracy level comparable to SfM.

Visibility check. The LiDAR scans and the images cover different portions of the scene. Only the regions of the scan visible to the input images should be used for evaluation. We derive visibility information for the LiDAR scans from the SfM point cloud, which is guaranteed to be observable by the images. To maximize the coverage of the true visible region, we use LoFTR [Sun et al. 2021a] to run SfM and generate semi-dense point clouds. We filter the LiDAR points by generating voxels around the SfM point clouds with a relatively large voxel size.

5 EXPERIMENTS

5.1 Implementation Details

Training is first bootstrapped by voxel-guided sampling for 5000 iterations, after which surface-guided sampling is added. We use 8 layers with 512 hidden units for the geometry MLP and 4 layers with 256 hidden units for the color MLP. The voxel size s of V_{sfm} for each scene are 2.8, 5.9, 2.0 and 1.0m for BG, LM, PE and PBA respectively. The depth level ℓ of octree V_{cache} is 10 for all scenes. The sampling radius t_s for each scene is defined as $16/2^\ell$ times of the voxel size s . We use $n_v = 8$ and $n_s = 8$ in all experiments. We use 8 NVIDIA A100 GPUs for all the experiments. For the final output mesh, we only extract a mesh within V_{sfm} .

5.2 Baselines

We compare our approach to state-of-the-art classical and learning-based MVS algorithms in terms of reconstruction quality and efficiency. For classical methods, we compare against the COLMAP dense reconstruction system [Schönberger et al. 2016], which is based on patch-match stereo and Poisson surface reconstruction. We use two different octree depths (11 and 13) in the Poisson reconstruction for comparisons, which we found to have the best numerical accuracy and visual quality, respectively (see Fig. 7). For learning-based approaches, we compare to Vis-MVSNet [Zhang et al. 2020], which achieves state-of-the-art performance on MVS benchmarks. We reconstruct meshes by fusing the depth maps using COLMAP’s point fusion algorithm, followed by Poisson surface reconstruction with octree depth=13. For completeness, we also compare to NeRF-W [Martin-Brualla et al. 2021]. For each scene, we first train a NeRF-W model, then create a pre-defined camera path and render color and depth at each viewpoint using the NeRF-W models. We feed the resulting RGB-D sequence into KinectFusion [Izadi et al. 2011] from Open3D [Zhou et al. 2018] for TSDF fusion to obtain a mesh.

5.3 Evaluation

Metrics and evaluation protocol. We quantitatively evaluate the reconstruction quality by measuring accuracy and completeness, and

Table 1: Reconstruction Evaluation. We report precision (P), recall (R) and F1 scores over the different datasets in Heritage-Recon. We compare our reconstruction performance to NeRF-W [Martin-Brualla et al. 2021], Vis-MVS [Zhang et al. 2020] and two variants of COLMAP, as detailed in the text. For our method, “Ours” measures performance after training converges and “Ours \star ” is an earlier checkpoint, selected using the F1 score (taking the model that yields 95% of the final value). We report performance over three different thresholds (“Low”, “Medium”, “High”) and also report an AUC metric that integrates performance over all thresholds (“All (AUC)”). Best results are in bold, and second best are underlined. As illustrated above, our method obtains the best or second best performance in nearly all cases.

	Method	BG			LM			PE			PBA		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Low	NeRF-W	1.8	1.1	1.4	28.0	13.4	18.1	11.0	3.4	5.2	39.6	12.7	19.2
	Vis-MVS	48.2	6.6	11.6	23.4	27.0	25.1	36.1	22.9	28.1	<u>81.0</u>	42.5	55.7
	colmap ¹¹	61.2	40.6	48.8	20.4	23.6	21.9	<u>48.7</u>	27.8	35.4	91.6	49.7	64.5
	colmap ¹³	79.1	41.6	54.5	25.8	32.6	28.8	54.4	71.0	61.6	75.5	28.9	41.8
	Ours \star	60.9	<u>46.8</u>	52.9	33.8	35.9	34.8	44.7	45.9	45.3	77.0	<u>56.4</u>	<u>65.1</u>
	Ours	<u>62.6</u>	47.9	<u>54.3</u>	<u>32.8</u>	<u>35.6</u>	<u>34.1</u>	48.5	<u>50.8</u>	<u>49.6</u>	78.1	57.7	66.4
Medium	NeRF-W	4.1	5.0	4.5	51.9	24.8	33.6	21.9	6.2	9.7	64.3	28.8	39.8
	Vis-MVS	72.4	12.1	20.8	46.3	42.7	44.4	57.8	34.4	43.1	<u>95.1</u>	57.8	71.9
	colmap ¹¹	73.7	53.1	61.8	50.1	55.1	52.4	66.6	56.5	61.2	98.3	57.9	72.9
	colmap ¹³	89.3	53.1	66.6	57.7	60.6	59.1	74.8	79.7	77.1	86.8	71.5	78.4
	Ours \star	77.6	<u>64.9</u>	<u>70.7</u>	<u>65.9</u>	<u>65.8</u>	<u>65.9</u>	68.3	65.8	67.1	87.9	68.9	77.3
	Ours	78.7	65.7	71.6	67.7	68.7	68.2	71.7	71.1	71.4	88.3	69.7	77.9
High	NeRF-W	7.6	11.5	9.2	67.3	34.2	45.3	28.8	8.3	12.9	79.5	40.5	53.7
	Vis-MVS	85.6	14.5	24.8	63.8	53.0	57.9	68.2	42.3	52.2	<u>98.0</u>	64.4	77.7
	colmap ¹¹	81.4	59.0	68.4	70.7	74.0	72.3	73.8	69.4	71.5	99.2	63.7	77.6
	colmap ¹³	94.1	58.8	72.4	78.0	75.9	76.9	81.6	83.6	82.6	91.3	80.8	85.7
	Ours \star	85.3	<u>73.2</u>	<u>78.8</u>	<u>80.8</u>	<u>80.2</u>	<u>80.5</u>	77.1	73.2	75.1	92.8	76.6	83.9
	Ours	85.9	73.7	79.3	82.1	82.1	82.1	<u>79.9</u>	<u>77.9</u>	<u>78.9</u>	93.0	77.1	84.3
All (AUC)	NeRF-W	2.0	2.8	2.3	2.9	1.5	2.0	6.1	1.9	2.9	46.4	21.4	29.1
	Vis-MVS	26.3	4.2	7.2	2.7	2.4	2.5	12.4	8.3	10.0	<u>66.6</u>	40.7	50.4
	colmap ¹¹	27.4	18.8	22.1	2.8	3.0	2.9	13.5	12.4	12.9	71.9	43.3	54.0
	colmap ¹³	33.3	19.1	24.1	3.2	3.2	3.2	14.6	15.7	14.9	62.9	45.5	50.5
	Ours \star	28.2	<u>22.9</u>	<u>25.1</u>	<u>3.3</u>	<u>3.3</u>	<u>3.3</u>	13.7	13.3	13.5	63.9	50.4	<u>56.3</u>
	Ours	<u>28.6</u>	23.2	25.5	3.4	3.4	3.4	<u>14.1</u>	<u>14.0</u>	<u>14.0</u>	64.4	51.1	56.9

evaluate reconstruction efficiency by measuring training/optimization time. Note that compared with object scenarios such as those in DTU, our “in-the-wild” cases exhibit much larger reconstruction errors and thus need multiple thresholds to reflect the reconstruction quality at different scales. Therefore, we use three different thresholds (Low, Medium, High) to measure per-scene precision, recall, and F1 scores. These thresholds are selected as follows: We first select a maximal threshold θ_{max} to compute the AUC (F1) metric, which integrates the F1 scores from 0 to θ_{max} . The maximal threshold is selected as the first threshold that reaches a F1-score of 80 on each scene. The Low, Medium and High thresholds are then evenly sampled between $(0, \theta_{\text{max}})$. To better indicate robustness and overall accuracy of each approach, we additionally compute the area under the curve (AUC) of precision, recall and F1 score curves by sweeping the error thresholds. More details on the evaluation protocol is presented in the supplemental material.

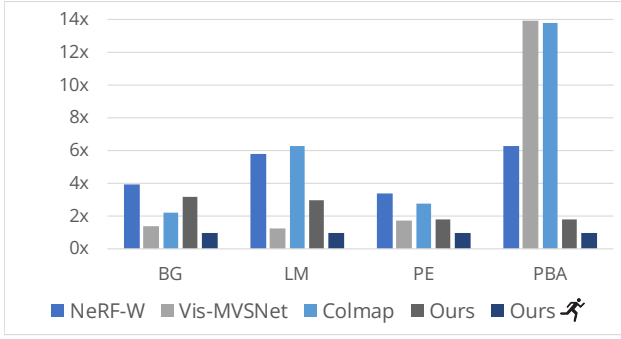



Figure 5: Training speed in comparison to “Ours”. As illustrated above, for larger scenes (e.g., PBA), the baselines can be up to 14× slower compared to our model that is optimized for training speed. In contrast, our best model yields a slowdown of up to 3×.

Comparisons. The quantitative and qualitative comparisons with baselines are shown in Table 1 and Fig 7, respectively. For our proposed approach, we report two results: (i) the model after full convergence (ours) and (ii) fast model with early stopping (ours ) . Our approach achieves the best or second best quantitative performance in nearly all scenes and across all thresholds. NeRF-W yields poor reconstruction performance as rendered depth maps are inconsistent across different views (while relatively accurate on individual views). The inconsistent back-projected point clouds will cancel each other’s contribution to the surface position during TSDF fusion, leading to poor reconstruction quality. Even though colmap¹³ outperforms our method on PE and PBA, our visual quality is significantly higher than colmap¹³, as shown in Fig. 7. In addition, our method achieves the overall best completeness as evidenced by its higher recall scores. Our representation and learning objective can even yield filled-in geometry in regions with insufficient observations, while the baselines fail to do so. To summarize, our method achieves significantly better visual quality than the baselines while being competitive numerically.

Reconstruction time. We compare the reconstruction time of different methods in Fig. 5. As the figure illustrates, our proposed approach can optimize scenes significantly faster, in most cases also if we consider our models obtained after full convergence (“Ours”). The point fusion in COLMAP operates on the CPU and takes the largest portion of the total time, highlighting the advantages of our method as an end-to-end surface reconstruction method.

Ablation studies. We conduct ablation studies to validate the effectiveness of our proposed sampling techniques. Fig. 6 shows a comparison of hybrid sampling to sphere-based sampling and pure voxel-guided sampling. Hybrid sampling consistently achieves the best F1 AUC score across almost all time steps, and remains noticeably better than the baselines even after substantial training time. To achieve an F1 AUC of ~ 3.2 , pure voxel-guided sampling requires 2~3× longer training compared to hybrid sampling, and sphere-based sampling is incapable of achieving this accuracy in a reasonable time. Additional ablation studies are presented in the supplemental material.

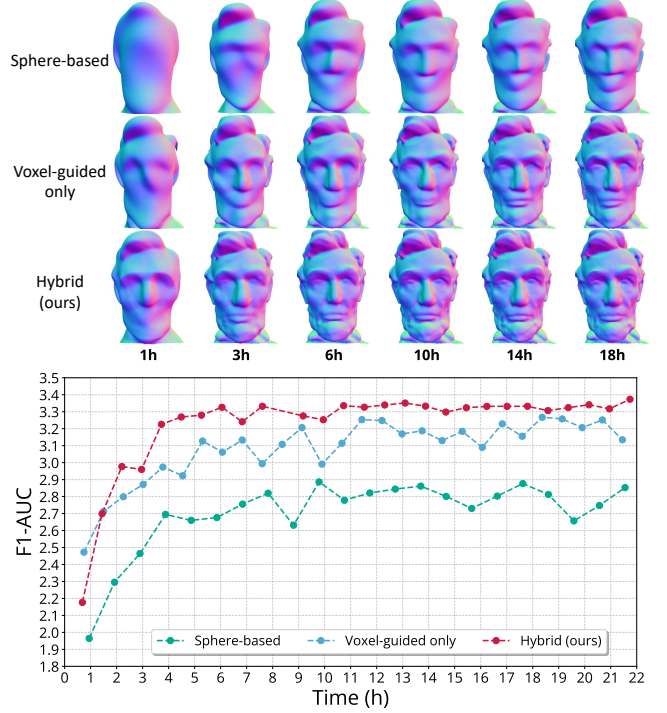


Figure 6: Ablation studies. We visualize the reconstructed meshes (zoomed in to the head region) and plot the curves of performance w.r.t. training time for sphere-based sampling, voxel-guided sampling and hybrid sampling on the Lincoln Memorial (LM). Hybrid sampling leads to significantly better training speed while consistently achieving the best F1-AUC accuracy and visual quality.

6 LIMITATIONS AND CONCLUSION

Limitations. Our approach inherits limitations from NeRF-like methods. For example, inaccurate camera registration can affect final reconstruction quality. In addition, since our model only learns surface locations from known image observations without imposing domain-specific priors, it can fail to produce accurate geometry in unseen regions.

Conclusion. We presented a new neural method for high-quality 3D surface reconstruction from Internet photo collections. To efficiently learn accurate surface locations of complex scenes, we introduce a hybrid voxel-surface guided sampling technique that significantly improves training time over baseline methods. In the future, we envision a full inverse rendering approach, as well as the ability to model scene dynamics across different time scales.

REFERENCES

- Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. 2011. Building Rome in a day. *Commun. ACM* 54, 10 (2011).
- Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. 2021a. NeRD: Neural reflectance decomposition from image collections. *ICCV* (2021), 12684–12694.
- Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan Barron, and Hendrik Lensch. 2021b. Neural-PIL: Neural pre-integrated lighting for reflectance decomposition. *NeurIPS* (2021).



Figure 7: Qualitative results on the Heritage-Recon dataset. As illustrated in the full rendered meshes and the zoomed-in selected regions (in orange), our models yield more complete and higher quality meshes. Zoom in for details.

- Aljaž Božič, Pablo Palafox, Justus Thies, Angela Dai, and Matthias Nießner. 2021. TransformerFusion: Monocular RGB Scene Reconstruction using Transformers. *NeurIPS* (2021).
- Dan Cerna. 2020. OpenMVS: Multi-View Stereo Reconstruction Library. (2020). <https://cdseacave.github.io/openMVS>
- Rohan Chabra, Jan E Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. 2020. Deep local shapes: Learning local sdf priors for detailed 3D reconstruction. *ECCV* (2020), 608–625.
- François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. 2021. Deep Multi-View Stereo gone wild. *3DV* (2021).
- Frank Dellaert and Lin Yen-Chen. 2020. Neural Volume Rendering: NeRF And Beyond. *arXiv preprint arXiv:2101.05204* (2020).
- Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. 2010. Building rome on a cloudless day. *ECCV* (2010), 368–381.
- Yasutaka Furukawa, Brian Curless, Steven M Seitz, and Richard Szeliski. 2010. Towards internet-scale multi-view stereo. *CVPR* (2010).
- Yasutaka Furukawa and Carlos Hernández. 2015. Multi-View Stereo: A Tutorial. *Found. Trends Comput. Graph. Vis.* 9 (2015), 1–148.
- Yasutaka Furukawa and Jean Ponce. 2009. Accurate, dense, and robust multiview stereopsis. *PAMI* 32, 8 (2009), 1362–1376.
- Michael Goesele, Brian Curless, and Steven M Seitz. 2006. Multi-view stereo revisited. *CVPR* (2006), 2402–2409.
- Michael Goesele, Noah Snavely, Brian Curless, Hugues Hoppe, and Steven M Seitz. 2007. Multi-view stereo for community photo collections. *ICCV* (2007).
- Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuo Zhuo Dai, Feitong Tan, and Ping Tan. 2020. Cascade cost volume for high-resolution multi-view stereo and stereo matching. *CVPR* (2020), 2495–2504.
- Peter Hedman, Suhil Alisan, Richard Szeliski, and Johannes Kopf. 2017. Casual 3D photography. *ACM Trans. Graph.* 36, 6 (2017), 1–15.
- Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. 2018. DeepMVS: Learning multi-view stereopsis. *CVPR* (2018), 2821–2830.
- Shahram Izadi, David Kim, Otmar Hilliges, David Molyneux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, and Andrew Fitzgibbon. 2011. KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera. In *UIST*. ACM.
- Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. 2014. Large scale multi-view stereopsis evaluation. *CVPR* (2014), 406–413.
- Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. 2006. Poisson surface reconstruction. *Proceedings of the fourth Eurographics symposium on Geometry processing* 7 (2006).
- Michael Kazhdan and Hugues Hoppe. 2013. Screened poisson surface reconstruction. *ACM Trans. Graph.* 32, 3 (2013), 1–13.
- Kichang Kim, Akihiko Torii, and Masatoshi Okutomi. 2016. Multi-view inverse rendering under arbitrary illumination and albedo. *ECCV* (2016), 750–767.
- Naruya Kondo, Yuya Ikeda, Andrea Tagliasacchi, Yutaka Matsuo, Yoichi Ochiai, and Shixiang Shane Gu. 2021. VaxNeRF: Revisiting the Classic for Voxel-Accelerated Neural Radiance Field. *arXiv preprint arXiv:2111.13112* (2021).
- Patrick Labatut, J-P Pons, and Renaud Keriven. 2009. Robust and efficient surface reconstruction from range data. *Computer graphics forum* 28, 8 (2009), 2275–2290.
- Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. 2021. Neural scene flow fields for space-time view synthesis of dynamic scenes. *CVPR* (2021).
- Zhengqi Li and Noah Snavely. 2018. MegaDepth: Learning Single-View Depth Prediction from Internet Photos. *CVPR* (2018).
- Zhengqi Li, Wenqi Xian, Abe Davis, and Noah Snavely. 2020. Crowdsampling the plenoptic function. *ECCV* (2020), 178–196.
- Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. 2021. BARF: Bundle-Adjusting Neural Radiance Fields. *ICCV* (2021).
- Chao Liu, Jinwei Gu, Kihwan Kim, Srinivasa G. Narasimhan, and Jan Kautz. 2019. Neural RGB-D Sensing: Depth and Uncertainty From a Video Camera. *CVPR* (2019), 10978–10987.
- Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. 2020. Neural sparse voxel fields. *NeurIPS* (2020).
- Ricardo Martin-Brualla, David Gallup, and Steven M Seitz. 2015a. 3D time-lapse reconstruction from internet photos. *ICCV* (2015), 1332–1340.
- Ricardo Martin-Brualla, David Gallup, and Steven M Seitz. 2015b. Time-lapse mining from internet photos. *ACM Trans. Graph.* 34, 4 (2015), 1–8.
- Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. 2021. NeRF in the Wild: Neural radiance fields for unconstrained photo collections. *CVPR* (2021), 7210–7219.
- Kevin Matzen and Noah Snavely. 2014. Scene chronology. *ECCV* (2014), 615–630.
- Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. 2019. Occupancy networks: Learning 3D reconstruction in function space. *CVPR* (2019).
- Moustafa Meshry, Dan B Goldman, Sameh Khamis, Hugues Hoppe, Rohit Pandey, Noah Snavely, and Ricardo Martin-Brualla. 2019. Neural rerendering in the wild. *CVPR* (2019), 6878–6887.
- 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. *ECCV* (2020), 405–421.
- Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. 2020. Atlas: End-to-end 3D scene reconstruction from posed images. *ECCV* (2020), 414–431.
- Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. 2020. Differentiable volumetric rendering: Learning implicit 3D representations without 3D supervision. *CVPR* (2020), 3504–3515.
- Michael Oechsle, Songyou Peng, and Andreas Geiger. 2021. UNISURF: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. *ICCV* (2021).
- Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. 2019. DeepSDF: Learning continuous signed distance functions for shape representation. *CVPR* (2019).
- Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. 2021. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *ACM Trans. Graph.* (2021).
- Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. 2020. Convolutional occupancy networks. *ECCV* (2020), 523–540.
- Konstantinos Rematas, Andrew Liu, Pratul P. Srinivasan, Jonathan T. Barron, Andrea Tagliasacchi, Tom Funkhouser, and Vittorio Ferrari. 2022. Urban Radiance Fields. *CVPR* (2022).
- Szymon Rusinkiewicz and Marc Levoy. 2001. Efficient variants of the ICP algorithm. *3DIM* (2001), 145–152.
- Johannes L Schönberger and Jan-Michael Frahm. 2016. Structure-from-motion revisited. *CVPR* (2016).
- Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. 2016. Pixelwise view selection for unstructured multi-view stereo. *ECCV* (2016).
- Qi Shan, Riley Adams, Brian Curless, Yasutaka Furukawa, and Steven M Seitz. 2013. The visual Turing test for scene reconstruction. *3DV* (2013), 25–32.
- Noah Snavely, Steven M Seitz, and Richard Szeliski. 2006. Photo Tourism: Exploring photo collections in 3D. *ACM Trans. Graph.* (2006).
- Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. 2021a. LoFTR: Detector-Free Local Feature Matching with Transformers. *CVPR* (2021).
- Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. 2021b. NeuralRecon: Real-Time Coherent 3D Reconstruction from Monocular Video. *CVPR* (2021), 15598–15607.
- 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. *NeurIPS* (2021).
- Suttisak Wizatwongsa, Pakkapon Phongthawee, Jiraphon Yenphraphai, and Supasorn Suwajanakorn. 2021. NeX: Real-time view synthesis with neural basis expansion. *CVPR* (2021).
- Yuanbo Xiangli, Linning Xu, Xingang Pan, Nanxuan Zhao, Anyi Rao, Christian Theobalt, Bo Dai, and Dahua Lin. 2021. CityNeRF: Building NeRF at City Scale. *arXiv preprint arXiv:2112.05504* (2021).
- Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. 2018. MVSNet: Depth inference for unstructured multi-view stereo. *ECCV* (2018), 767–783.
- Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. 2019. Recurrent mvsnet for high-resolution multi-view stereo depth inference. *CVPR* (2019), 5525–5534.
- Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. 2021. Volume rendering of neural implicit surfaces. *NeurIPS* (2021).
- Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. 2020. Multiview Neural Surface Reconstruction by Disentangling Geometry and Appearance. *NeurIPS* (2020).
- Lin Yen-Chen, Pete Florence, Jonathan T. Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. 2021. iNeRF: Inverting Neural Radiance Fields for Pose Estimation. *IROS* (2021).
- Jason Zhang, Gengshan Yang, Shubham Tulsiani, and Deva Ramanan. 2021b. NeRS: Neural reflectance surfaces for sparse-view 3D reconstruction in the wild. *NeurIPS* (2021).
- Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. 2020. Visibility-aware Multi-view Stereo Network. *Bmvc* (2020).
- Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. 2021a. NeRFactor: Neural Factorization of Shape and Reflectance Under an Unknown Illumination. *ACM Trans. Graph.* (2021).
- Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. 2018. Open3D: A Modern Library for 3D Data Processing. *arXiv:1801.09847* (2018).