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  
snavely@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 Rodriguez, 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.
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.

1 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.

¹ https://openheritage3d.org/
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; Wizadwongsa et al. 2021], camera pose estimation [Lin et al. 2021; Yen-Chen et al. 2021], and intrinsic decomposition [Boss et al. 2021a; 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 $x \in \mathbb{R}^3$ in the scene, a viewing direction $v \in S^2$ and an image index $i$, we have:

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

$$c_i = \text{MLP}_{\text{COLOR}}(x, v, e_i),$$

where $(e_i)_{i=1}^N$ 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:

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

The function $c_i$ models the appearance of 3D point $x$ 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, $[r(t) = o + tv | t \geq 0]$ with $o$ denoting the camera center, we can render that ray’s expected color $C_i(r)$ corresponding to image $i$ as:

$$\hat{C}_i(r) = \int_0^{+\infty} w(t)e_i(r(t), v, e_i)dt,$$

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 $V_{\text{sfm}}$ 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 $V_{\text{sfm}}$, and $n_{\text{vox}}$ 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 voxel sets provide 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 $V_{\text{cache}}$, and query the surface position from this cache at each training iteration. $V_{\text{cache}}$ is an octree built upon $V_{\text{sfm}}$ with a depth level of $\ell$. 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. $V_{\text{cache}}$ 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 $n_d + 2 \times n_s$.

### 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 given sufficient time.

**Supervision signals and handling the textureless sky.** Following NeuS, we use an $L_1$ loss to supervise the rendered color images ($L_{\text{COLOR}}$) and use an eikonal term $L_{\text{REG}}$ 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 $V_{\text{sfm}}$ (mostly sky) are labeled with a semantic mask and penalized as free space with $L_{\text{MASK}}$. Since the semantic masks for the background are not perfect and often contain foreground scene structures, we only apply $L_{\text{MASK}}$ 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 $L_{\text{COLOR}}$, $L_{\text{REG}}$ and $L_{\text{MASK}}$.

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* ($PE$), *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 Open Heritage 3D. 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 from 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.
from SM 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 SM 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 SM.

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 SM 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 SM and generate semi-dense point clouds. We filter the LiDAR points by generating voxels around the SM 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 $V_{SM}$ for each scene is 2.8, 5.9, 2.0 and 1.0m for BG, LM, PE and PBA respectively. The depth level $\ell$ 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 $V_{SM}$.

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 [Schoenberger 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-Brual 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 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 $\theta_{max}$ to compute the AUC (F1) metric, which integrates the F1 scores from 0 to $\theta_{max}$. 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 and $\theta_{max}$. 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.

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-Brual 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 $4^\ell$” 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  |     |     |
| NeRF-W | 1.8 | 1.1 | 1.4 | 8.0 |
| Vis-MVS| 48.2| 6.6 | 11.6| 23.4|
| colmap1 | 61.2| 40.6| 48.8| 29.0|
| colmap13 | 79.1| 41.6| 54.3| 25.8|
| Ours   | 69.9| 46.8| 52.9| 33.8|
| Ours $4^\ell$ | 69.9| 46.8| 52.9| 33.8|

| Method | Medium | Low | High |
|--------|--------|-----|------|
| $P$    | R      | F1  | All (AUC) |
| NeRF-W | 7.6    | 11.5| 9.2 |
| Vis-MVS| 85.6   | 14.5| 24.8|
| colmap1 | 81.4  | 59.0| 68.4|
| colmap13 | 94.1| 58.8| 72.4|
| Ours   | 85.9 | 73.7| 79.3|
| Ours $4^\ell$ | 85.9| 73.7| 79.3|

| Method | $P$    | R      | F1   | All (AUC) |
|--------|--------|--------|------|-----------|
| NeRF-W | 2.0    | 2.8    | 2.9  | 2.1        |
| Vis-MVS| 26.3   | 4.2    | 7.2  | 4.24       |
| colmap1 | 27.4  | 18.8  | 22.1 | 2.8        |
| colmap13 | 33.3 | 19.1  | 24.1 | 3.2        |
| Ours   | 28.2  | 22.9  | 25.1 | 3.3        |
| Ours $4^\ell$ | 28.2| 22.9| 25.1| 3.3|
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 colmap outperforms our method on PE and PBA, our visual quality is significantly higher than colmap, 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.

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.
