arXiv:2402.18771v2 [cs.CV] 16 Apr 2024

NARUTO: Neural Active Reconstruction from Uncertain Target Observations

Ziyue Feng [*†,1,2] Huangying Zhan [*‡,1] Zheng Chen [†,1,3] Qingan Yan [1]
Xiangyu Xu [1] Changjiang Cai [1] Bing Li [2] Qilun Zhu [2] Yi Xu [1]

[1] OPPO US Research Center [2] Clemson University [3] Indiana University

Abstract

We present NARUTO, a neural active reconstruction system that combines a hybrid neural representation with uncertainty learning, enabling high-fidelity surface reconstruction. Our approach leverages a multi-resolution hash-grid as the mapping backbone, chosen for its exceptional convergence speed and capacity to capture high-frequency local features. The centerpiece of our work is the incorporation of an uncertainty learning module that dynamically quantifies reconstruction uncertainty while actively reconstructing the environment. By harnessing learned uncertainty, we propose a novel uncertainty aggregation strategy for goal searching and efficient path planning. Our system autonomously explores by targeting uncertain observations and reconstructs environments with remarkable completeness and fidelity. We also demonstrate the utility of this uncertainty-aware approach by enhancing SOTA neural SLAM systems through an active ray sampling strategy. Extensive evaluations of NARUTO in various environments, using an indoor scene simulator, confirm its superior performance and state-of-the-art status in active reconstruction, as evidenced by its impressive results on benchmark datasets like Replica and MP3D. Project page: oppo-us-research.github.io/NARUTO-website/

1. Introduction

In the realm of computer vision research, one of the most notable advancements is the ability to generate detailed 3D reconstructions from an array of 2D images or scene videos. This intricate process, executed in real-time, involves progressive 3D modeling as additional visual data is assimilated, predominantly through the use of Simultaneous Localization and Mapping (SLAM). In many robotic applications, SLAM systems are instrumental for tasks such as planning and navigation. This integration of localization, mapping, planning, and navigation tasks forms the essence of what is known as Active SLAM. Our paper specifically addresses a subset of Active SLAM, termed Active Reconstruction, under the assumption that localization is already established. We venture into an innovative exploration of Active Reconstruction by adopting a sophisticated, learned hybrid neural representation . In this work, we devise methodologies capable of meticulously planning and maneuvering camera trajectories to enhance the completeness and quality of the scene's reconstruction.

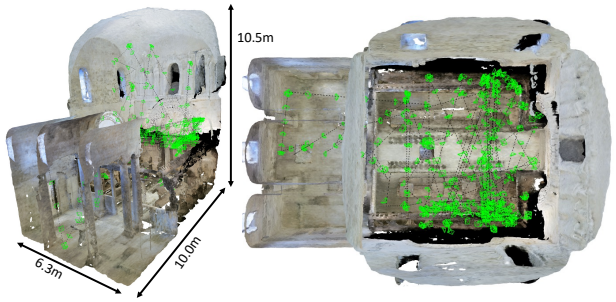

Figure 1. We introduce a neural active reconstruction system, named *NARUTO*, which is guided by learned uncertainty. NARUTO enables an agent to identify areas of uncertainty and proactively investigate these regions to minimize reconstruction ambiguity. Consequently, this approach facilitates the incremental completion of the entire scene's reconstruction. *NARUTO* represents the first neural active Reconstruction system capable of functioning in large-scale environments with unrestricted movement.

Neural representations, particularly implicit Neural Radiance Fields (NeRFs), have recently been applied in diverse geometric applications, such as 3D object reconstruction [50], novel view rendering [44, 54, 81, 85], surface reconstruction [2, 37], and generative models [48, 62]. While many of these methods focus on posed cameras, recent efforts have expanded to broader tasks like structure from motion [13, 38, 76] and SLAM [68, 73, 86, 87]. Despite the impressive capabilities of NeRFs, their processing speed remains a challenge. To address this, more efficient hybrid neural representations have been developed [46, 69].

Integrating these representations into active vision applications continues to pose significant challenges. Existing research utilizing neural representations for path planning is limited [1], and only a handful of recent studies

*Equal contribution
†Work done as an intern at OPPO US Research Center
‡Corresponding author (zhanhuangying.work@gmail.com)

have explored active reconstruction with neural representations [36, 49, 56, 79, 84]. These approaches, while innovative, often suffer from the inherent slow speeds of NeRFs [36, 49, 56]. Moreover, they typically constrain the movement of agents to a lower degree-of-freedom (DoF) within restricted areas, such as specific locations [36, 49], within a hemisphere [56, 84], or on a 2D plane [79].

To overcome the aforementioned limitations, we introduce *NARUTO*, a groundbreaking neural active reconstruction system. *NARUTO* unites a hybrid neural representation with a novel uncertainty-aware planning module, excelling in high-fidelity surface reconstruction and proactive planning, shown in Fig. 1. Our key contributions are as follows:

- The *first* neural active reconstruction system operating with 6DoF movement in unrestricted spaces.
- An uncertainty learning module quantifies reconstruction uncertainty in real-time.
- A novel uncertainty-aware planning features a meticulously designed uncertainty aggregation for goal searching, and efficient path planning.
- Active ray sampling strategy enhances the performance and stability of mapping modules across various tasks.
- Achieving exceptional active reconstruction performance, advancing state-of-the-art in reconstruction completeness from 73% to 90%.

2. Related Work

Active Reconstruction In autonomous robotics, essential capabilities include localization, mapping, planning, and motion control [64]. These elements have led to research areas like visual odometry [60, 83], monocular depth estimation [3, 20, 23, 24, 82], multi-view stereo [6, 11, 28, 40, 63, 70, 80], structure-from-motion (SfM) [61], path planning [22, 27, 34, 35], and SLAM [5, 16, 19, 71, 75]. Active SLAM, which combines these approaches for autonomous localization, mapping, and planning, minimizes uncertainties in environmental modeling [15]. We refer readers to the survey papers [5, 41, 53] for a comprehensive discussion regarding the development of active SLAM. Our focus is on active reconstruction, often investigated as exploration problems [4, 21, 42, 47, 65, 66, 72]. a problem that seeks optimal movements for accurate environmental representations [14], primarily for scene and object reconstruction from multiple viewpoints [17, 29, 33, 43, 51, 52].

Neural Representaitons NeRFs [44] use multi-layer perceptrons (MLPs) to represent scenes as continuous neural radiance fields. NeRF's potential has been demonstrated in a range of applications, from novel view rendering [44, 54, 81, 85] to object [44, 50] and surface reconstruction [2, 37], as well as in generative models [48, 62], Structure-from-Motion [13, 38, 76]. NeRFs are trained by comparing rendered images with accurately posed ones. However, the volume rendering process [30], which involves querying numerous sample points for image rendering, makes training NeRFs time-intensive, often requiring about a day for simple scenes. While efforts have been made to accelerate NeRFs [12, 18, 39, 57], these methods still fall short of real-time application speeds. Recent work [10, 46, 58, 69] have achieved fast speed through hybrid representations, combining implicit and explicit elements for light and density fields, respectively. The advancement in hybrid representations has been instrumental in meeting the real-time requirements of SLAM challenges [73, 86, 87]. Despite these advancements, applying neural representations in active vision problems is still an underexplored area.

Neural Active Vision Our research builds upon prior works that have explored the use of NeRFs for path planning [1] and active reconstruction [36, 49, 56]. [1] derives optimal paths for navigation from the NeRF-based scene representation. Recent studies [36, 49, 56] have focused on active mapping, optimizing NeRFs with next-best-view selection strategies. However, these approaches are constrained by the inherent slow speed of NeRFs, limiting their real-time application in robotics. [84] proposes an efficient framework using hybrid representations to address these speed concerns. Meanwhile, works like [9, 25, 79] have expanded the scope from object-centric reconstruction [36, 49, 56, 84] to larger indoor environments. However, these methods still restrict camera motion to a hemisphere or a 2D plane. In contrast, *NARUTO* enables 6DoF exploration in unrestricted spaces.

3. NARUTO: Neural Active Reconstruction

In this section, we introduce *NARUTO* (Fig. 2), a pioneering neural framework in active reconstruction with uncertainty-aware planning. Our approach begins with the neural 3D mapping module, utilizing a hybrid representation for real-time, high-fidelity surface reconstruction. We incorporate Co-SLAM [73] as the mapping backbone, as discussed in Sec. 3.1, laying the groundwork for 3D reconstruction using hybrid neural representation. Building upon this, Sec. 3.2 delves into the framework's core, illustrating the joint optimization method that fuses bundle adjustment with uncertainty learning. In Sec. 3.3, we present the uncertainty-aware planning module for goal searching and path planning. Sec. 3.4 introduces a versatile active ray sampling module. This module, leveraging the learned uncertainty, is designed for seamless integration into existing neural mapping methodologies. Concluding this section, we summarize the procedure of active reconstruction in Sec. 3.5.

3.1. Neural 3D Mapping

Implicit Neural Mapping Recent advancements have established neural implicit representations as notably expressive and compact, effectively encoding scenes' appearance

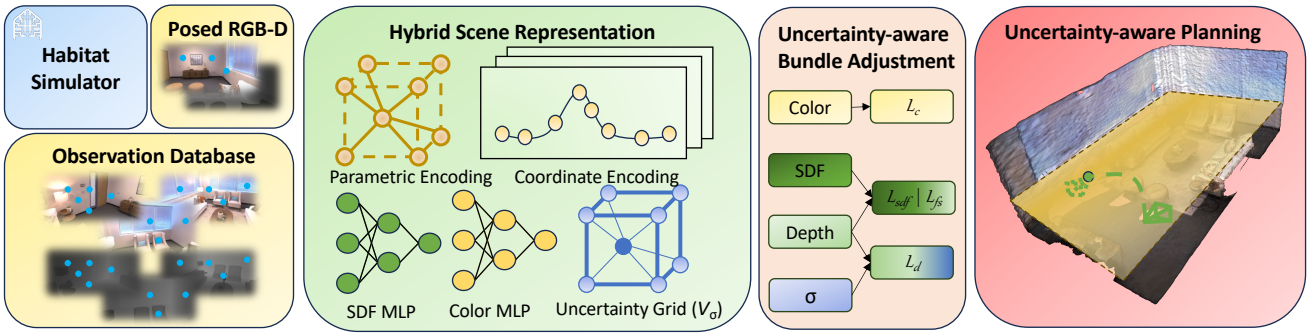

Figure 2. **NARUTO framework** Upon reaching a keyframe step, HabitatSim [59] generates posed RGB-D images. A select number of pixels from these images are sampled and stored in the observation database. Utilizing a mixed ray sampling strategy (combining Random and Active methods), a subset of rays is selected from the current keyframe and the database. These rays are then processed through the Hybrid Scene Representation (Map) to deduce the corresponding color, Signed Distance Function (SDF), depth, and uncertainty values. The predictions derived from this process facilitate uncertainty-aware bundle adjustment, updating both the scene's geometry and reconstruction uncertainty. Subsequently, the Map is refreshed, and our novel uncertainty-aware planning algorithm is employed to determine a goal and trajectory based on the SDFs and uncertainties. The agent then executes the planned action.

and 3D geometry. A series of prior works, including [37, 68, 73, 86, 87], have demonstrated the applicability of neural representation in 3D reconstruction. Given a stream of RGB-D images, dense mapping with representations, such as radiance fields and truncated signed distance fields (TSDF), can be achieved by optimizing a neural representation via rendering supervision. TSDF, in particular, is widely used for neural surface reconstruction. Coordinate-based neural representations are often employed to map world coordinates $\mathbf{x}$ to color $\mathbf{c}$ and TSDF value s.

Hybrid Representation MLPs are widely utilized as coordinate-based implicit representations for high-fidelity scene reconstruction, owing to their coherence and smoothness. However, they are not without drawbacks, such as slow convergence and catastrophic forgetting in continual learning scenarios, as identified in [7, 78]. To address these challenges, we apply several innovative solutions introduced by Co-SLAM [73]. Among these is a joint coordinate and parametric encoding, designed to enhance fidelity while expediting training processes. The incorporation of one-blob coordinate encoding $\gamma(\mathbf{x})$ [45] with a multi-resolution hash-based feature grid achieves rapid querying speeds, efficient memory usage, and a notable hole-filling capability. In this setup, the feature vector $V_\alpha(\mathbf{x})$ at each sampled point $\mathbf{x}$ is obtained through trilinear interpolation on the feature grid. The geometry decoder f_τ predicts an SDF value s and a feature vector $\mathbf{h}$. Additionally, the color MLP, denoted as f_ϕ, calculates the color value.

$$f_\tau(\gamma(\mathbf{x}), V_\alpha(\mathbf{x})) \mapsto (\mathbf{h}, s)\ ;\ f_\phi(\gamma(\mathbf{x}), \mathbf{h}) \mapsto \mathbf{c}, \quad (1)$$

where $\{\alpha, \phi, \tau\}$ represents the learnable parameters that can be optimized in the bundle adjustment.

Bundle Adjustment Bundle Adjustment (BA) in neural SLAM typically employs volumetric rendering optimization [68, 73, 86]. Instead of storing full images, we execute BA on sparse samples from the keyframes, enabling more frequent keyframe insertions and a larger keyframe collection. For this process, given a camera origin $\mathbf{o}$ and a ray direction $\mathbf{r}$, 3D points are sampled along the ray, based on predefined depths d_i: $\mathbf{x}_i = \mathbf{o} + d_i\mathbf{r}$. The color $\hat{\mathbf{c}}$ and depth $\hat{\mathbf{d}}$ can be rendered:

$$\hat{\mathbf{c}} = \frac{1}{\sum_{i=1}^{M} w_i} \sum_{i=1}^{M} w_i \mathbf{c}_i\ ,\ \hat{d} = \frac{1}{\sum_{i=1}^{M} w_i} \sum_{i=1}^{M} w_i d_i, \quad (2)$$

where $w_i = \varphi(\frac{s_i}{tr})\varphi(-\frac{s_i}{tr})$ represents the weights computed along the ray, obtained by applying Sigmoid functions $\varphi(.)$ to the predicted SDF s_i within a truncated range $tr = 10$cm.

Post rendering, a multi-objective function is minimized to execute bundle adjustment, incorporating color and depth rendering losses. These losses are calculated between the rendered results ($\hat{\mathbf{c}}$, $\hat{d}$) and the observed values ($\mathbf{c}^o$,D):

$$\mathcal{L}_c = \frac{1}{N} \sum_{i=1}^{N} (\hat{\mathbf{c}}_i - \mathbf{c}_i^o)^2\ ,\ \mathcal{L}_d = \frac{1}{|R_d|} \sum_{r \in R_d} (\hat{d}_r - D_r)^2 \quad (3)$$

where $N = 2148$, R_d denotes the set of rays with valid depths, and D_r corresponds to the pixel on the image plane.

Following [73], we apply additional regularizations to enhance reconstruction quality. For samples within the truncation region S_r^{tr}, SDF loss is approximated by the distance between the sampled point and its observed depth value. Conversely, for points outside the truncation region S_r^{fs}, a free-space loss ensures SDF predictions equal to tr:

$$\mathcal{L}_{sdf} = \frac{1}{|R_d|} \sum_{r \in R_d} \frac{1}{|S_r^{tr}|} \sum_{p \in S_r^{tr}} (s_p - (D_p - d))^2 \quad (4)$$

$$\mathcal{L}_{fs} = \frac{1}{|R_d|} \sum_{r \in R_d} \frac{1}{|S_r^{fs}|} \sum_{p \in S_r^{fs}} (s_p - tr)^2. \quad (5)$$

To ensure smooth reconstructions in unobserved free-space regions, we apply a feature smoothness regularization on the interpolated features $V_\alpha(\mathbf{x})$:

$$\mathcal{L}_{smooth} = \frac{1}{|\mathcal{G}|} \sum_{\mathbf{x} \in \mathcal{G}} \Delta_x^2 + \Delta_y^2 + \Delta_z^2, \tag{6}$$

where $\Delta_{x,y,z} = V_\alpha(\mathbf{x} + \epsilon_{x,yz}) - V_\alpha(\mathbf{x})$ is the feature difference of some sampled vertices.

3.2. Reconstruction Uncertainty Learning

Recent studies [26, 49, 56, 79, 84] have investigated various approaches for quantifying uncertainty in implicit representations. [49, 56] propose implicitly learning uncertainty through an MLP network. This uncertainty MLP predicts point uncertainties for each sampled point along selected rays. These point uncertainties are then integrated to calculate the photometric uncertainty of each pixel, employing the volume rendering technique described in Sec. 3.1. However, this form of uncertainty, as noted in [32], does not strongly correlate with geometric uncertainty. Alternatively, [84] opts for explicit and efficient computation of geometric uncertainty, represented as a 3D volume, from predicted densities. Notably, the methods mentioned above are either RGB-based, lacking depth sensing, or do not incorporate depth measurements in uncertainty learning. This omission is significant, as depth information is essential for accurate uncertainty quantification. In our work, we integrate the uncertainty learning process with depth rendering, as outlined in Eq. (3), within the bundle adjustment framework. This integration follows the strategy proposed in [31], effectively combining depth data with uncertainty.

$$\mathcal{L}_d = \frac{1}{|R_d|} \sum_{r \in R_d} \left(\frac{1}{2\hat{\sigma}_r^2} (\hat{d}_r - D_r)^2 + \frac{1}{2} \log \hat{\sigma}_r^2 \right), \tag{7}$$

$$\text{where } \hat{\sigma}_r^2 = \frac{1}{\sum_{i=1}^{M} w_i} \sum_{i=1}^{M} w_i \sigma_i^2 \tag{8}$$

This study delves into two distinct methodologies for representing reconstruction uncertainty: implicit and explicit representations. For the implicit approach, we employ an MLP to estimate point uncertainty, $f_\sigma(\gamma(\mathbf{x}), \mathbf{h}) \mapsto V_\sigma(\mathbf{x})$. However, our observations highlight a notable drawback of this implicit uncertainty representation. Due to the reliance on the UncertaintyNet for predictions, any parameter update within the MLP results in alterations to uncertainty values across all regions, including those yet to be observed, *i.e.* regions that lack observations are expected to exhibit high uncertainty; however, these areas often show random uncertainty levels instead. In response to this challenge, we develop a learnable uncertainty volume, V_σ, designed to represent surface reconstruction uncertainty efficiently. This volume enables rapid querying of uncertainties via trilinear interpolation, $\sigma_i^2 = f_\rho(V_\sigma(\mathbf{x}_i))$, followed by a non-linear softplus activation function $f_\rho(.)$. We initially set the volume with high uncertainty. Significantly, as this volume is updated during bundle adjustment through uncertainty-aware depth rendering, only the uncertainties in regions that have been observed are modified. This feature is vital for the effectiveness of active vision tasks. The comparative advantages of our explicit representation over implicit methods are further detailed in Sec. 4.3.

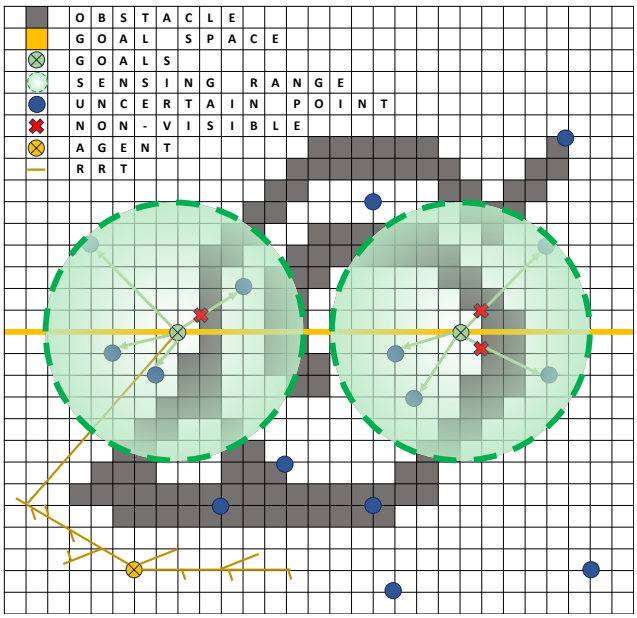

Figure 3. **Uncertainty-aware Planning Illustration**. The top-k uncertain points are accumulated within the sensing range at each potential goal location. The goal with the greatest level of uncertainty is subsequently selected as the provisional target location. Efficient RRT planning effectively identifies a viable trajectory from the agent's current position to the designated goal.

3.3. Uncertainty-aware Planning

In this section, we elaborate on the application of learned uncertainty and geometry in active planning, aiming to achieve comprehensive and high-quality reconstruction. The planning module comprises two primary components: Goal Searching and Path Planning. Utilizing the up-to-date SDF map that incorporates the learned geometric uncertainty, our primary goal is to pinpoint the most effective goal location for reducing overall map uncertainty. To this end, we introduce an innovative uncertainty aggregation strategy, which facilitates the creation of an uncertainty-aware goal space. Following the identification of the optimal observation location, we proceed with executing efficient path planning to establish a trajectory toward the chosen goal. A 2D illustration of this approach is depicted in Fig. 3.

Uncertainty Aggregation for Goal Search Utilizing the most recent mapping model, denoted as $\mathbf{M}$, we undertake

Algorithm 1 NARUTO: Neural Active Reconstruction

Initialization Mapping Model $\mathbf{M}$ with $[V_s; V_\sigma]$; Agent State $\mathbf{s}_t = \mathbf{s}_0$; Goal Space $\mathbf{S}_g$; Observations $\{O\}_{i=0}^{0}$; **PLAN_REQUIRED** = True
for $t \leftarrow 0$ to T **do**
  if PLAN_REQUIRED then
    # Search a new goal from Goal Space if needed
    GoalSearch($\mathbf{M}_t$, $\mathbf{s}_t$) $\rightarrow \mathbf{s}_g \in \mathbf{S}_g$
    # Plan a feasible path based on M_t towards $\mathbf{s}_g$
    PathPlanning($\mathbf{M}_t$, $\mathbf{s}_t$, $\mathbf{s}_g$) $\rightarrow \{\mathbf{s}_j\}_{j=t}^{g}$
    # Set **PLAN_REQUIRED** to False
    PLAN_REQUIRED $\leftarrow$ False
  end if
  # Execute to follow planned path
  Action $\mathbf{s}_t \leftarrow \{\mathbf{s}_j\}_{j=t}^{g}$
  # Update Database in keyframe steps
  Observation: acquire a new observation O_t
  Update database: $\{O\}_{i=0}^{t} \leftarrow \{O\}_{i=0}^{t-1}$
  # Update Mapping Model
  Mapping Optimization: Update $\mathbf{M}_t \leftarrow \mathbf{M}_t$
  # Replanning if detected collision or reached goal
  CheckPlanRequired: update **PLAN_REQUIRED**
end for

two key constructions. First, we generate an SDF volume, $V_s \in \mathbb{R}^{H\times W\times D}$, through uniform querying $\mathbf{M}$ across the space. Second, we establish an uncertainty volume, $V_\sigma \in \mathbb{R}^{H\times W\times D}$, which encapsulates the geometric uncertainty of the reconstruction space. The foremost goal of this process is to determine the optimal observation location. This location is characterized as the point from which the most substantial regions of high uncertainty can be observed. To effectively identify such a location, we have developed a novel *uncertainty aggregation* strategy.

Initially, we set up a multi-level Goal Space, denoted as $\mathbf{S}_g \in \mathbb{R}^{H\times W\times N}$, comprising layers that are distributed at different heights within the space. The arrangement is such that each layer is approximately 1 meter apart from its adjacent layers, providing a structured vertical distribution throughout the space. Rather than aggregating uncertainties at every vertex within V_σ onto the Goal Space, our method focuses on a set of vertices with the top-k uncertainty, denoted as $\{\mathbf{x}_\sigma\}^k$, where $k = 300$. For each point $\mathbf{x}_g$ sampled within the Goal Space, we accumulate the uncertainty of all visible $\{\mathbf{x}_\sigma\}^k$ points, provided they fall within the optimal observation range of $[0.5, 2]m$. Visibility is ascertained by examining the SDF values between $\mathbf{x}_g$ and $\mathbf{x}_\sigma$. Upon completing this aggregation process, the goal with the highest aggregated value is subsequently selected as the provisional target location. The goal state $\mathbf{s}_g$ is defined as the goal location looking at its most uncertain region.

Efficient RRT Path Planning Upon pinpointing the goal location, our path planning module is activated to devise a viable path linking the current state, $\mathbf{s}_t$, with the goal state, $\mathbf{s}_g$. For this purpose, we adopt a sampling-based path planning methodology akin to the Rapid-exploration Random Tree (RRT) [35], utilizing the SDF map V_s as a basis. Notably, executing the conventional RRT within a large-scale 3D environment proves to be considerably time-consuming. To mitigate this challenge, we implement an efficient planning approach inspired by [34]. Our strategy enhances the traditional RRT by not only iterating through random point sampling but also consistently seeking direct, feasible lines connecting these sampled points with the goal state. Such augmentation significantly expedites the planning process, thereby making RRT practical and efficient even in expansive scenes. Note that occasionally, the identified goal state $\mathbf{s}_g$ may be situated in a location that, while lying within the predefined 3D bounding box, is actually outside the navigable space. In such instances, RRT typically fails to find a valid or feasible path, as shown by reaching the maximum sampling number. To address this issue, we assess the reachability of all V_σ vertices by querying RRT. If a vertex is determined to be unreachable — specifically, if it lies at a minimum distance beyond the agent's step size — it is then excluded from the uncertainty aggregation process.

Action Execution In our system, the agent is capable of performing several actions under various events:

- ***Move***: The agent moves towards the target, looking at the 3D point with the highest uncertainty.
- ***Observe***: Upon reaching $\mathbf{s}_g$, the agent sequentially observes the top-*10* uncertain points within the sensing range via rotational motion.
- ***Stay***: The agent remains stationary either upon reaching the goal location or when collisions are detected.

Note that Goal Space and the RRT space can be tailored to suit the specific dimensions of the scene as well as the type of agent involved, whether it be a ground robot or an aerial robot. To demonstrate the generalization of our system, we model the agent as a free-moving entity with a spherical body, which has a radius of 5cm. The agent's motion is constrained to translations $\leq$ 10cm and rotations $\leq 10^\circ$.

3.4. Active Ray Sampling

In the process of mapping optimization, Co-SLAM [73] employs a strategy of sampling N rays from both the database and the most recent keyframe. While this random sampling technique facilitates optimization across various regions, it occasionally leads to inconsistent results. Moreover, this approach does not ensure that regions characterized by subpar reconstruction quality are adequately sampled. By incorporating the learned uncertainty, we introduce a more targeted ray sampling method. This approach

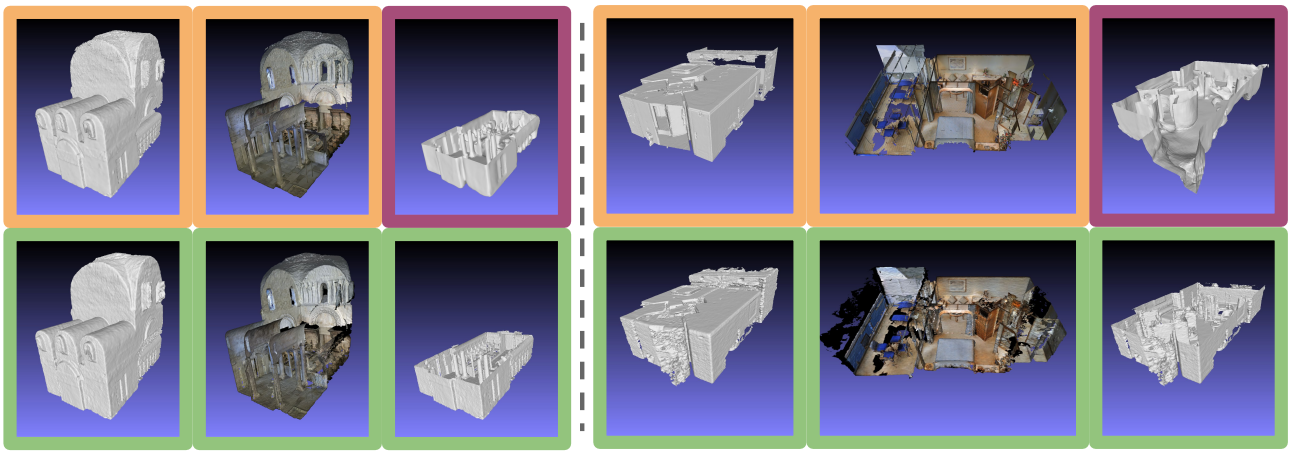

Figure 4. **Matterport3D Results** Two scenes (Left: pLe4; Right: HxpK) are presented here. The results are distinguished by border colors: [Ground Truth , ANM[79], Ours]. In our results, notably in the second and fifth columns, black regions signify incomplete GT mesh, illustrating the extrapolation capacity of our neural mapping module. Results in columns 3 and 6 are trimmed for better comparison.

retains the diversity of the original sampling strategy but enhances it by substituting N' rays from the random sample with the top-N' rays, selected based on their uncertainty. This active ray sampling technique improves the consistency and quality of the system's output across different iterations, as presented in Sec. 4.3.

3.5. Active Reconstruction

Integrating the mapping module outlined in Sec. 3.1 and Sec. 3.2, with the planning module from Sec. 3.3, we establish a comprehensive neural active reconstruction system, as detailed in Algorithm 1 and illustrated in Fig. 2. Leveraging an up-to-date neural mapping model, this system employs the planning module to perform goal searching and path planning. Subsequent to each action executed for acquiring a new RGB-D frame, a selection of rays from the keyframes is stored in a database to facilitate mapping optimization. This storage occurs at a fixed interval of every *5* steps. Replanning is triggered under two conditions: either after the completion of the ***Observe*** action at the goal location or upon detection of a collision.

4. Experiments and Results

4.1. Experimental Setup

Simulator and Dataset Our experiments utilize the Habitat simulator [59] and are evaluated on two photorealistic datasets: Replica [67] and Matterport3D (MP3D) [8]. Specifically, we select 8 scenes from Replica [68] and 5 scenes from MP3D [79] for our analysis. The experiments are designed to run for 2000 steps in Replica and 5000 steps in MP3D, reflecting the larger scene sizes in MP3D that necessitate more steps for thorough exploration. In these experiments, our system processes posed RGB-D images at a resolution of 680×1200, with the field of view settings at 60° vertically and 90° horizontally. We use 10cm as the voxel size for all experiments when generating 3D volume.

This work represents a departure from previous neural active reconstruction efforts, which typically involve action spaces constrained to teleporting between discrete locations [49, 56], moving within limited areas such as a hemisphere [84], or navigating the local vicinity on a 2D plane [9, 79]. In contrast, we introduce the *first* neural active reconstruction system operating with 6DoF movement in unrestricted 3D spaces. Given the inherent randomness in the methods, we conduct each experiment five times to ensure reliability and present the average outcomes. For experiments with active planning, the agent's starting position is randomly initialized within the traversable space for each trial.

Metrics We evaluate the reconstruction using *Accuracy* (cm), *Completion* (cm), *Completion ratio* (%) with a threshold of 5cm. We also compute the mean absolute distance, *MAD* (cm), between the estimated SDF distance on all vertices from the ground truth mesh. In line with methodologies employed in previous studies [73, 74], we refine the predicted mesh by removing unobserved regions and noisy points that are within the camera frustum but external to the target scene, utilizing a mesh culling technique. Refer to [73] for a detailed explanation of the mesh culling process.

4.2. Evaluation

To our knowledge, this is the *first* study to address the challenge of active surface reconstruction in large-scale indoor scenes with the provision for 6DoF movements in 3D space. Previous studies that allow for 6DoF motions, such as [29, 33, 36, 56, 84], have primarily focused on object-centric scenarios. In contrast, earlier works targeting

	MAD (cm) ↓	Acc. (cm) ↓	Comp. (cm) ↓	Comp. Ratio (%) ↑
FBE [77]	/	/	9.78	71.18
UPEN [25]	/	/	10.60	69.06
OccAnt [55]	/	/	9.40	71.72
ANM [79]	4.29	7.80	9.11	73.15
Ours	**1.44**	**6.31**	**3.00**	**90.18**

Table 1. **MP3D Results** Our method shows superior performance with better reconstruction quality and completeness.

large-scale indoor scenes have generally been categorized under the *active exploration* task. These studies, including [9, 25, 79], often employ reinforcement learning-based planners and restrict agent movement to a 2D plane. Notably, ANM [79] is among the closest to our work; it also utilizes neural implicit representation for mapping in large-scale indoor environments. Averaged results are presented in this section, while a comprehensive evaluation of individual scenes is included in the supplementary material.

MP3D In Tab. 1, we provide a quantitative comparison of our system against previous studies on MP3D. Our approach significantly surpasses prior work across all evaluation metrics. The MAD metric reflects the precision of the learned 3D neural distance field in our model. Furthermore, both the Completion and Completion Ratio metrics, which assess the extent of active exploration coverage in 3D space, indicate that our method achieves remarkably high completeness. This success is attributable to our effective method of goal identification combined with the agent's unrestricted movement capabilities, as shown in Fig. 1.

It is important to note that the Accuracy metric is calculated by computing the mean nearest distance (with respect to the prediction) between the predicted vertices and the ground-truth vertices. However, a challenge arises with the MP3D scenes due to their real-world capture; the ground-truth mesh often exhibits incompleteness resulting from incomplete scanning. In scenarios where neural implicit reconstruction is applied, the neural networks' extrapolation capacity can fill in these missing regions. While this might be beneficial in some contexts, it poses a disadvantage for the Accuracy evaluation. This effect is exemplified in Fig. 4, where the discrepancy due to neural network extrapolation is evident. In Fig. 4, it is evident that our method yields a more comprehensive and high-fidelity reconstruction, underscoring the effectiveness of our approach.

4.3. Ablation Studies

Replica features photorealistic 3D indoor scenes, spanning both room and building scales. Each scene in this dataset is represented by a dense mesh, which typically exhibits greater completeness compared to the MP3D scenes. Given this higher level of completeness, we primarily conduct our ablation studies on the Replica dataset to ensure more representative and robust results.

Method	Acc. (cm)		Comp. (cm)		Comp. Ratio (%)	
	μ	$\sigma^2(10^{-3})$	μ	$\sigma^2(10^{-3})$	μ	$\sigma^2(10^{-2})$
Neural SLAM						
iMAP [68]	3.62	/	4.93	/	80.50	/
NICE-SLAM [86]	2.37	/	2.63	/	91.13	/
Co-SLAM [73]	**2.30**	34.56	**2.35**	29.51	**92.74**	72.90
[73] **w/ ActRay**	**2.30**	**26.10**	**2.35**	**15.06**	92.70	**11.77**
Neural Mapping: Tracking is disabled						
Co-SLAM [73]	**1.96**	3.02	2.00	0.86	93.79	2.16
[73] **w/ ActRay**	**1.96**	**2.88**	**1.98**	**0.50**	**93.90**	**1.88**
Neural Active Mapping						
w/o ActiveRay	1.67		1.76		96.89	
Uncertainty Net	1.69		2.05		94.62	
Full	**1.61**		**1.66**		**97.20**	

Table 2. Evaluation and Ablation Studies on Replica.

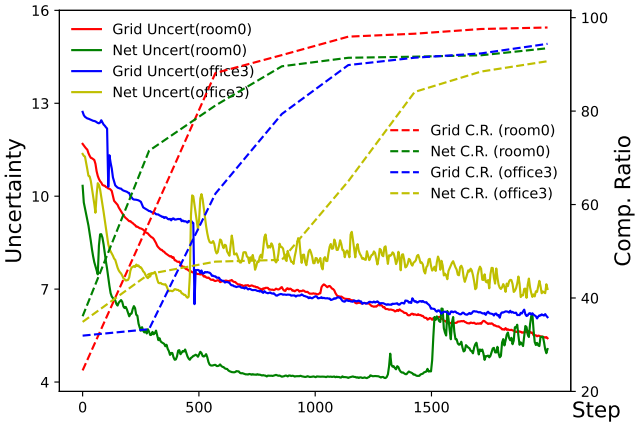

Figure 5. Evolution of Uncertainty and Completion Using Explicit Grid and Implicit Net. The abrupt decrease in *Grid Uncert(office3)* correlates with the implementation of the reachability filtering strategy, as outlined in Sec. 3.3.

Active Ray Sampling In this section, we assess the efficacy of the Active Ray Sampling strategy (ActiveRay), as detailed in Sec. 3.4. We tested the strategy across three distinct tasks, presenting the results in Tab. 2. Leveraging our learned uncertainty, the Active Ray Sampling module acts as a versatile plug-and-play enhancement for existing neural mapping methods, leading to improved reconstruction outcomes. Focusing on the Neural SLAM task, we integrate our learned uncertainty and the Active Ray Sampling strategy into Co-SLAM [73]. Our results demonstrate reconstruction quality comparable to the original Co-SLAM. More importantly, multiple trials reveal that the inclusion of Active Ray Sampling yields more consistent results with reduced variance. The Neural SLAM task, which involves estimating camera poses, introduces an additional complexity to the optimization process. In the second task, we concentrate on mapping capabilities, deactivating the tracking function in Co-SLAM [73]. Without the instability introduced by the tracking thread, our method exhibits improved reconstruction quality compared to Co-SLAM. A key advantage of this approach in both tasks is the enhancement of result stability, evidenced by reduced variance. In the

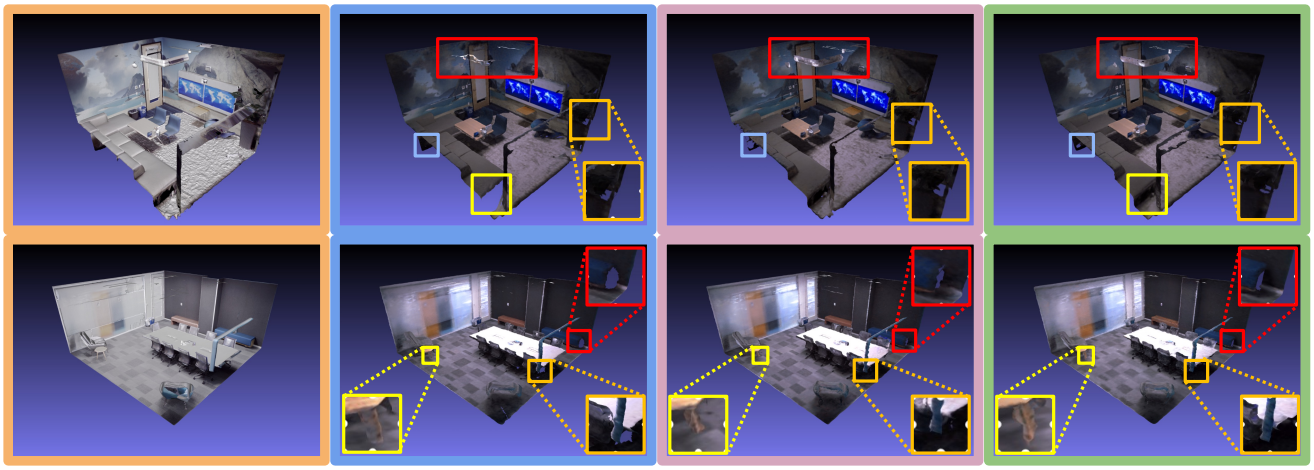

Figure 6. **Replica Results** Two scenes (office0, office3) are shown in the first and second rows, respectively. The results represent [Ground Truth, Uncertainty Net, w/o ActiveRay, Full]. Our Full method shows a better completeness and quality on the highlighted regions. Note that the GT visualization uses view-dependent rendering, unlike our mapping backbone, resulting in color differences in the visualizations.

third task, focusing on Active Neural Mapping, we demonstrate that ActiveRay is a crucial element of our system. We surmise that this effectiveness stems from our system's deliberate focus on accruing more observations from regions of uncertainty. Consequently, this leads to an increase in the number of valid rays, especially those marked by uncertainty, making them prime candidates for selection by ActiveRay. We provide a qualitative comparison in Fig. 6, contrasting results obtained using our complete method with those achieved without ActiveRay. The full implementation of our method, employing ActiveRay, demonstrates enhanced completeness and finer detail in thin structures.

Explicit Grid v.s. Implicit Net We discuss the use of explicit and implicit representation in Sec. 3.2. It was noted that utilizing an implicit representation (Uncertainty Net) for learning uncertainty presents stability challenges. The optimization process employing Uncertainty Net is depicted in Fig. 5, where it is juxtaposed with our proposed Uncertainty Grid for comparative analysis. Two principal observations emerge from this comparison: Firstly, both Uncertainty Net and Uncertainty Grid demonstrate rapid convergence, underscoring the efficacy of our uncertainty-aware planning approach. Secondly, as previously discussed in Sec. 3.2, Uncertainty Net tends to produce fluctuating uncertainty values during the optimization phase due to continuous updates in network parameters. This instability is also illustrated in Fig. 5, where we include $\log(\sum_{\mathbf{x}_i} V_\sigma(\mathbf{x}_i))$ and the completion ratios, highlighting the comparative stability offered by Uncertainty Grid. In Uncertainty Grid, a clear correlation is observed: the completion ratio increases as uncertainty decreases. Conversely, in Uncertainty Net, these two metrics do not exhibit a strong correlation. In Fig. 6, we present a qualitative comparison demonstrating that using Uncertainty Grid results in higher reconstruction completeness than Uncertainty Net.

5. Discussion

In summary, NARUTO represents a significant advancement in the field of neural active reconstruction. By integrating a hybrid neural representation with uncertainty learning, and a novel uncertainty-aware planning module, we present the *first* neural active reconstruction system that enables agents to execute 6DoF movement in unrestricted space. Furthermore, the enhancement of state-of-the-art neural mapping methods through our active ray sampling strategy underscores the versatility and practicality of NARUTO. Rigorous evaluation in diverse environments using an indoor scene simulator demonstrates our system's superior performance, outperforming existing methods on benchmark datasets such as Replica and MP3D, setting a new standard in active reconstruction.

While NARUTO exhibits outstanding performance, future research directions are identified to advance the field. Firstly, the current assumption of known localization and perfect action execution, which might not hold in real-world scenarios, suggests the need for a robust planning and localization module to enhance real-world applicability. Secondly, the agent's motion constraints, vital in practical applications, should be considered to refine the system's general movement solution. Lastly, the use of a single-resolution uncertainty grid, primarily focusing on scene completeness, could be evolved into a multi-resolution uncertainty representation to meet diverse requirements. These future explorations aim to augment NARUTO's practicality and adaptability in real-world settings, pushing the boundaries of autonomous robotic systems.

References

[1] Michal Adamkiewicz, Timothy Chen, Adam Caccavale, Rachel Gardner, Preston Culbertson, Jeannette Bohg, and Mac Schwager. Vision-only robot navigation in a neural radiance world. *IEEE Robotics and Automation Letters*, 7(2):4606–4613, 2022. 1, 2

[2] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6290–6301, 2022. 1, 2, 3

[3] Jiawang Bian, Zhichao Li, Naiyan Wang, Huangying Zhan, Chunhua Shen, Ming-Ming Cheng, and Ian Reid. Unsupervised scale-consistent depth and ego-motion learning from monocular video. *Advances in neural information processing systems*, 32, 2019. 2

[4] Frederic Bourgault, Alexei A Makarenko, Stefan B Williams, Ben Grocholsky, and Hugh F Durrant-Whyte. Information based adaptive robotic exploration. In *IEEE/RSJ international conference on intelligent robots and systems*, pages 540–545. IEEE, 2002. 2

[5] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6): 1309–1332, 2016. 2

[6] Changjiang Cai, Pan Ji, Qingan Yan, and Yi Xu. Riav-mvs: Recurrent-indexing an asymmetric volume for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 919–928, 2023. 2

[7] Zhipeng Cai and Matthias Müller. Clnerf: Continual learning meets nerf. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23185–23194, 2023. 3

[8] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017. 6, 3, 4

[9] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural slam. *arXiv preprint arXiv:2004.05155*, 2020. 2, 6, 7

[10] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[11] Liyan Chen, Weihan Wang, and Philippos Mordohai. Learning the distribution of errors in stereo matching for joint disparity and uncertainty estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17235–17244, 2023. 2

[12] Zhang Chen, Zhong Li, Liangchen Song, Lele Chen, Jingyi Yu, Junsong Yuan, and Yi Xu. Neurbf: A neural fields representation with adaptive radial basis functions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4182–4194, 2023. 2

[13] Shin-Fang Chng, Sameera Ramasinghe, Jamie Sherrah, and Simon Lucey. Gaussian activated neural radiance fields for high fidelity reconstruction and pose estimation. In *European Conference on Computer Vision*, pages 264–280. Springer, 2022. 1, 2

[14] Cl Connolly. The determination of next best views. In *Proceedings. 1985 IEEE international conference on robotics and automation*, pages 432–435. IEEE, 1985. 2

[15] Andrew J Davison and David W. Murray. Simultaneous localization and map-building using active vision. *IEEE transactions on pattern analysis and machine intelligence*, 24(7):865–880, 2002. 2

[16] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1052–1067, 2007. 2

[17] Jeffrey Delmerico, Stefan Isler, Reza Sabzevari, and Davide Scaramuzza. A comparison of volumetric information gain metrics for active 3d object reconstruction. *Autonomous Robots*, 42(2):197–208, 2018. 2

[18] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022. 2

[19] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006. 2

[20] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014. 2

[21] Hans Jacob S Feder, John J Leonard, and Christopher M Smith. Adaptive mobile robot navigation and mapping. *The International Journal of Robotics Research*, 18(7):650–668, 1999. 2

[22] Ziyue Feng, Shitao Chen, Yu Chen, and Nanning Zheng. Model-based decision making with imagination for autonomous parking. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 2216–2223. IEEE, 2018. 2

[23] Ziyue Feng, Longlong Jing, Peng Yin, Yingli Tian, and Bing Li. Advancing self-supervised monocular depth learning with sparse lidar. In *Conference on Robot Learning*, pages 685–694. PMLR, 2022. 2

[24] Ziyue Feng, Liang Yang, Longlong Jing, Haiyan Wang, YingLi Tian, and Bing Li. Disentangling object motion and occlusion for unsupervised multi-frame monocular depth. In *European Conference on Computer Vision*, pages 228–244. Springer, 2022. 2

[25] Georgios Georgakis, Bernadette Bucher, Anton Arapin, Karl Schmeckpeper, Nikolai Matni, and Kostas Daniilidis. Uncertainty-driven planner for exploration and navigation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 11295–11302. IEEE, 2022. 2, 7

[26] Lily Goli, Cody Reading, Silvia Selllán, Alec Jacobson, and Andrea Tagliasacchi. Bayes' rays: Uncertainty quantification for neural radiance fields. *arXiv preprint arXiv:2309.03185*, 2023. 4

[27] Peter Hart, Nils Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. 2

[28] Heiko Hirschmuller. 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)*, pages 807–814. IEEE, 2005. 2

[29] Stefan Isler, Reza Sabzevari, Jeffrey Delmerico, and Davide Scaramuzza. An information gain formulation for active volumetric 3d reconstruction. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3477–3484. IEEE, 2016. 2, 6

[30] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH computer graphics*, 18(3):165–174, 1984. 2

[31] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017. 4

[32] Maria Klodt and Andrea Vedaldi. Supervising the new with the old: learning sfm from sfm. In *ECCV 2018*, pages 698–713, 2018. 4

[33] Simon Kriegel, Christian Rink, Tim Bodenmüller, and Michael Suppa. Efficient next-best-scan planning for autonomous 3d surface reconstruction of unknown objects. *Journal of Real-Time Image Processing*, 10(4): 611–631, 2015. 2, 6

[34] James J Kuffner and Steven M LaValle. Rrt-connect: An efficient approach to single-query path planning. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, pages 995–1001. IEEE, 2000. 2, 5, 1, 3

[35] Steven M LaValle, James J Kuffner, BR Donald, et al. Rapidly-exploring random trees: Progress and prospects. *Algorithmic and computational robotics: new directions*, 5:293–308, 2001. 2, 5, 1

[36] Soomin Lee, Le Chen, Jiahao Wang, Alexander Liniger, Suryansh Kumar, and Fisher Yu. Uncertainty guided policy for active robotic 3d reconstruction using neural radiance fields. *IEEE Robotics and Automation Letters*, 2022. 2, 6

[37] Kejie Li, Yansong Tang, Victor Adrian Prisacariu, and Philip HS Torr. Bnv-fusion: Dense 3d reconstruction using bi-level neural volume fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 3

[38] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5741–5751, 2021. 1, 2

[39] Celong Liu, Zhong Li, Junsong Yuan, and Yi Xu. Neulf: Efficient novel view synthesis with neural 4d light field. *arXiv preprint arXiv:2105.07112*, 2021. 2

[40] Jiachen Liu, Pan Ji, Nitin Bansal, Changjiang Cai, Qingan Yan, Xiaolei Huang, and Yi Xu. Planemvs: 3d plane reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8665–8675, 2022. 2

[41] Iker Lluvia, Elena Lazkano, and Ander Ansuategi. Active mapping and robot exploration: A survey. *Sensors*, 21(7):2445, 2021. 2

[42] Alexei A Makarenko, Stefan B Williams, Frederic Bourgault, and Hugh F Durrant-Whyte. An experiment in integrated exploration. In *IEEE/RSJ international conference on intelligent robots and systems*, pages 534–539. IEEE, 2002. 2

[43] Jasna Maver and Ruzena Bajcsy. Occlusions as a guide for planning the next view. *IEEE transactions on pattern analysis and machine intelligence*, 15(5): 417–433, 1993. 2

[44] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1): 99–106, 2021. 1, 2

[45] Thomas Müller, Brian McWilliams, Fabrice Rousselle, Markus Gross, and Jan Novák. Neural importance sampling. *ACM Transactions on Graphics (ToG)*, 38(5):1–19, 2019. 3

[46] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives

with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, 2022. 1, 2

[47] Paul Newman, Michael Bosse, and John Leonard. Autonomous feature-based exploration. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, pages 1234–1240. IEEE, 2003. 2

[48] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11453–11464, 2021. 1, 2

[49] Xuran Pan, Zihang Lai, Shiji Song, and Gao Huang. Activenerf: Learning where to see with uncertainty estimation. In *European Conference on Computer Vision*, pages 230–246. Springer, 2022. 2, 4, 6

[50] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 1, 2

[51] Daryl Peralta, Joel Casimiro, Aldrin Michael Nilles, Justine Aletta Aguilar, Rowel Atienza, and Rhandley Cajote. Next-best view policy for 3d reconstruction. In *European Conference on Computer Vision*, pages 558–573. Springer, 2020. 2

[52] Richard Pito. A solution to the next best view problem for automated surface acquisition. *IEEE Transactions on pattern analysis and machine intelligence*, 21(10): 1016–1030, 1999. 2

[53] Julio A Placed, Jared Strader, Henry Carrillo, Nikolay Atanasov, Vadim Indelman, Luca Carlone, and José A Castellanos. A survey on active simultaneous localization and mapping: State of the art and new frontiers. *arXiv preprint arXiv:2207.00254*, 2022. 2

[54] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 1, 2

[55] Santhosh K Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. Occupancy anticipation for efficient exploration and navigation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 400–418. Springer, 2020. 7

[56] Yunlong Ran, Jing Zeng, Shibo He, Lincheng Li, Yingfeng Chen, Gimhee Lee, Jiming Chen, and Qi Ye. Neurar: Neural uncertainty for autonomous 3d reconstruction. *arXiv preprint arXiv:2207.10985*, 2022. 2, 4, 6

[57] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[58] Sara Fridovich-Keil and Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *CVPR*, 2022. 2

[59] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019. 3, 6

[60] Davide Scaramuzza and Friedrich Fraundorfer. Visual odometry [tutorial]. *IEEE robotics & automation magazine*, 18(4):80–92, 2011. 2

[61] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 2

[62] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *Advances in Neural Information Processing Systems*, 33:20154–20166, 2020. 1, 2

[63] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, pages 519–528. IEEE, 2006. 2

[64] Roland Siegwart, Illah Reza Nourbakhsh, and Davide Scaramuzza. *Introduction to autonomous mobile robots*. MIT press, 2011. 2

[65] Cyrill Stachniss. *Robotic mapping and exploration*. Springer, 2009. 2

[66] Cyrill Stachniss, Dirk Hahnel, and Wolfram Burgard. Exploration with active loop-closing for fast-slam. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, pages 1505–1510. IEEE, 2004. 2

[67] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 6, 3, 4, 5

[68] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J Davison. imap: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6229–6238, 2021. 1, 3, 6, 7

[69] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *CVPR*, 2022. 1, 2

[70] Jian Sun, Nan-Ning Zheng, and Heung-Yeung Shum. Stereo matching using belief propagation. *IEEE Transactions on pattern analysis and machine intelligence*, 25(7):787–800, 2003. 2

[71] Sebastian Thrun. Probabilistic robotics. *Communications of the ACM*, 45(3):52–57, 2002. 2

[72] Sebastian B Thrun and Knut Möller. Active exploration in dynamic environments. *Advances in neural information processing systems*, 4, 1991. 2

[73] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13293–13302, 2023. 1, 2, 3, 5, 6, 7, 4, 16

[74] Jingwen Wang, Tymoteusz Bleja, and Lourdes Agapito. Go-surf: Neural feature grid optimization for fast, high-fidelity rgb-d surface reconstruction. In *2022 International Conference on 3D Vision (3DV)*, pages 433–442. IEEE, 2022. 6, 3

[75] Weihan Wang, Jiani Li, Yuhang Ming, and Philippos Mordohai. Edi: Eskf-based disjoint initialization for visual-inertial slam systems. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1466–1472. IEEE, 2023. 2

[76] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 1, 2

[77] Brian Yamauchi. A frontier-based approach for autonomous exploration. In *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation'*, pages 146–151. IEEE, 1997. 7

[78] Zike Yan, Yuxin Tian, Xuesong Shi, Ping Guo, Peng Wang, and Hongbin Zha. Continual neural mapping: Learning an implicit scene representation from sequential observations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15782–15792, 2021. 3

[79] Zike Yan, Haoxiang Yang, and Hongbin Zha. Active neural mapping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10981–10992, 2023. 2, 4, 6, 7, 3, 5, 15

[80] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018. 2

[81] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 1, 2

[82] Huangying Zhan, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian Reid. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 340–349, 2018. 2

[83] Huangying Zhan, Chamara Saroj Weerasekera, Jia-Wang Bian, and Ian Reid. Visual odometry revisited: What should be learnt? In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4203–4210. IEEE, 2020. 2

[84] Huangying Zhan, Jiyang Zheng, Yi Xu, Ian Reid, and Hamid Rezatofighi. Activermap: Radiance field for active mapping and planning. *arXiv preprint arXiv:2211.12656*, 2022. 2, 4, 6

[85] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 1, 2

[86] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12786–12796, 2022. 1, 2, 3, 7

[87] Zihan Zhu, Songyou Peng, Viktor Larsson, Zhaopeng Cui, Martin R Oswald, Andreas Geiger, and Marc Pollefeys. Nicer-slam: Neural implicit scene encoding for rgb slam. *arXiv preprint arXiv:2302.03594*, 2023. 1, 2, 3

NARUTO: Neural Active Reconstruction from Uncertain Target Observations

Supplementary Material

6. Overview

In this supplementary material, we provide a detailed outline structured as follows: Sec. 7 delves into additional implementation specifics of NARUTO. Sec. 8 examines the computation costs associated with each module. Complementing the results in Sec. 4, Sec. 9 extends our analysis with per-scene evaluations for MP3D and Replica.

7. Implementation Details

Hardware Details We run the experiments on a desktop PC with a 2.2GHz Intel Xeon E5-2698 CPU and NVIDIA V100 GPU.

Memory requirement Memory consumption varies depending on the scene size. As a reference, in a $120m^3$ scene, the corresponding GPU memory and RAM are 8.1GB and 8.6GB respectively. The consumption can be further reduced with a more efficient implementation as our current implementation involves intensive exchanges between RAM and GPU memories.

7.1. Neural Mapping Details

We adopt Co-SLAM [73] as the foundational mapping framework for our system, adhering to the hyperparameter configurations established therein. For details pertaining to the hyperparameters specific to the mapping component, we direct readers to [73] for comprehensive information.

7.2. Efficient RRT Details

Path planning in three-dimensional spaces presents significant computational challenges, particularly when employing standard 3D RRT algorithms [35]. In our approach, we introduce an accelerated version of RRT, dubbed E-RRT (Efficient RRT), which incorporates several optimizations for improved performance.

The primary innovation in E-RRT, drawing inspiration from RRT-Connect [34], is its strategy to first attempt direct connections from the growing tree to the goal at each iteration. While this does not ensure the shortest path, it significantly enhances the efficiency of finding a viable path.

Furthermore, E-RRT enhances the process of node expansion. Instead of adding a single node, our method integrates a series of feasible points uniformly distributed between a randomly generated node and its nearest neighbor in the tree, based on a predefined step size, for instance, 10 cm, up the distance of $M \times$ step size. Here M equals to 10. This modification substantially accelerates the expansion of the tree, especially in the initial growth stages.

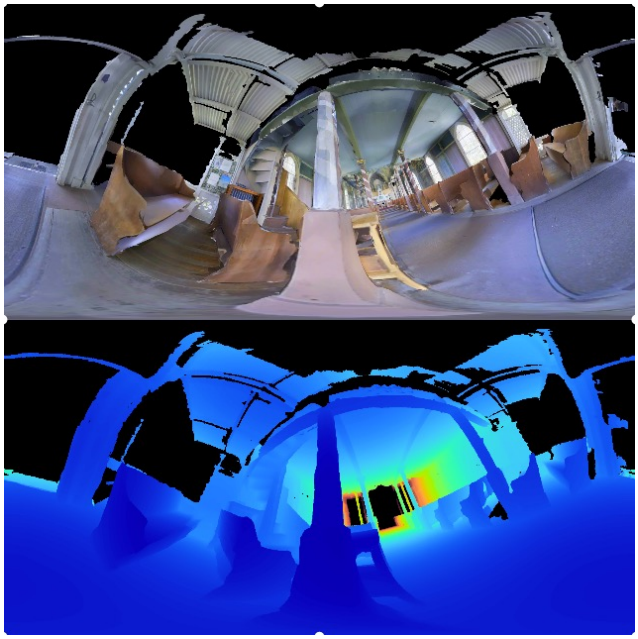

Figure 7. **Equirectangular RGB-D Example** Black regions refer to the invalid regions with zero depth measurement. The ratio of black regions increases significantly when the agent leaves the building. This is used as a signal for collision detection.

Lastly, we address the increasing computational load associated with nearest-neighbor searches as the tree expands. By leveraging parallel processing on a GPU, E-RRT achieves a consistently high search speed, thus mitigating the computational costs that typically escalate with tree complexity.

7.3. Collision Detection

We have tailored two distinct collision detection methodologies to align with the nuances of the Replica and Matterport3D datasets.

For experiments conducted within the Replica dataset, collision detection is facilitated through an SDF map derived from our hybrid scene representation. We assess potential collisions by sampling points at 2 cm intervals between consecutive states and querying the SDF map at these points. A collision is inferred when the SDF value at any point falls below the 5 cm threshold, consistent with our model of the agent as a sphere with a 5 cm radius.

This protocol effectively prevents the agent from intersecting with wall surfaces during simulations. Nonetheless, it cannot preclude the agent from exiting the scene through non-watertight boundaries. In contrast, the Matterport3D

dataset, reflecting real-world environments, presents unique challenges with regions devoid of geometry—artifacts of incomplete depth data during dataset construction. These gaps in the environment can erroneously permit the agent to traverse through "walls" or exit buildings. To counteract this, in addition to the SDF-based collision detection, we have developed a specialized collision detection system that assesses equirectangular depth measurements (*e.g.* Fig. 7) at prospective states, calculating the proportion of invalid regions. An increase in this proportion signals potential egress from the building, and by establishing a threshold ratio, we can determine the validity of the next state, thereby preventing unintended departure from the environment.

7.4. Rotation Planning

As delineated in Sec. 3.3, when the agent arrives at a designated goal state $\mathbf{s}_g$, it proceeds to sequentially observe the top-10 points of uncertainty within its sensing radius through a series of rotational movements. In an effort to reduce the number of steps necessary to cover all ten of these uncertain perspectives, we have devised a straightforward rotational planning algorithm. This method involves identifying the subsequent viewpoint that can be reached with the least rotational effort and then executing the transition using a Spherical Linear Interpolation (SLERP) strategy.

7.5. Active Ray Sampling Details

In the context of mapping optimization within Co-SLAM[73], the conventional approach entails the random selection of 2048 pixels from the database, supplemented by a minimum of 100 pixels from the current viewpoint. Our Active Ray Sampling strategy introduces a refinement to this process. Specifically, we quadruple the count of randomly sampled pixels, thus drawing 8192 pixels from the database and ensuring at least 400 pixels from the current viewpoint. Within this augmented sample set, we then identify and prioritize the 500 most uncertain pixels. The remaining 1548 pixels are selected from the database, in addition to a minimum of 100 random points from the current viewpoint. This hybrid sampling method effectively combines the breadth of random sampling with the targeted insight of Active Ray Sampling, thereby capturing a broad yet informative snapshot of the environment.

8. Runtime Analysis

8.1. System Runtime

In this section, we present a detailed runtime analysis of the three major modules in NARUTO, as illustrated in Fig. 8. The first module is a simulator for data generation. The second is a mapping module optimized for a hybrid scene representation. Lastly, we have an uncertainty-aware planning module. For data generation, HabitatSim requires, on average, 24.4ms to generate 680×1200 RGB-D data per iteration. The Mapping module, although taking about 300ms per iteration, averages 60.5ms since it is activated only every five keyframes. The Active Planning module averages 2.1ms, which includes 0.3ms for collision detection per iteration. Additionally, Active Planning encompasses two modules that are triggered occasionally when the 'PLAN_REQUIRED' condition is met. These are the uncertainty-aware goal searching, averaging 6.8ms, and RRT path planning, averaging 5.77 ms. In conclusion, our analysis demonstrates that NARUTO offers real-time capabilities, particularly due to its efficient planning module.

Method	Time (ms)	Node Num.	Step Num.
RRT	19×10^3	19×10^3	28×10^3
w/o direct line	17×10^3	20×10^3	21×10^3
w/o fast tree	16.00	44.17	2.56
Ours (E-RRT)	5.77	16.70	1.19

Table 3. **RRT runtime analysis on Replica-room0.** We conducted a runtime analysis of RRT variants, revealing that our optimized RRT implementation significantly outpaces traditional RRT in planning speed, achieving real-time planning capabilities.

8.2. RRT Runtime Analysis

In this section, we delve deeper into our optimized RRT implementation, as outlined in Sec. 7.2. We have engineered a customized version of RRT that enhances planning speed through several strategies:

- Direct Line: Actively identifying straight paths that link the RRT tree to the goal.
- Fast Tree: Speeding up the expansion of the tree.
- Parallel Computing: Utilizing GPU processing for increased efficiency.

These innovations significantly reduce the time required for path planning, making our RRT variant highly suitable for real-time applications. We present an ablation study on the runtime performance of our RRT approach in Tab. 3. To maintain consistency, all experiments were conducted using parallel processing for nearest-neighbor searches during tree expansion.

Evaluation Our evaluation of the methods encompasses three key metrics: the average time taken for each path planning request, the average number of nodes generated within the RRT tree, and the average number of steps taken in the RRT process.

Analysis Compared to traditional RRT, our efficient RRT implementation is markedly faster, both in average planning time and iteration count. It also generates fewer nodes and uses less memory, as shown by the reduced average number of nodes required per planning request. The ablation

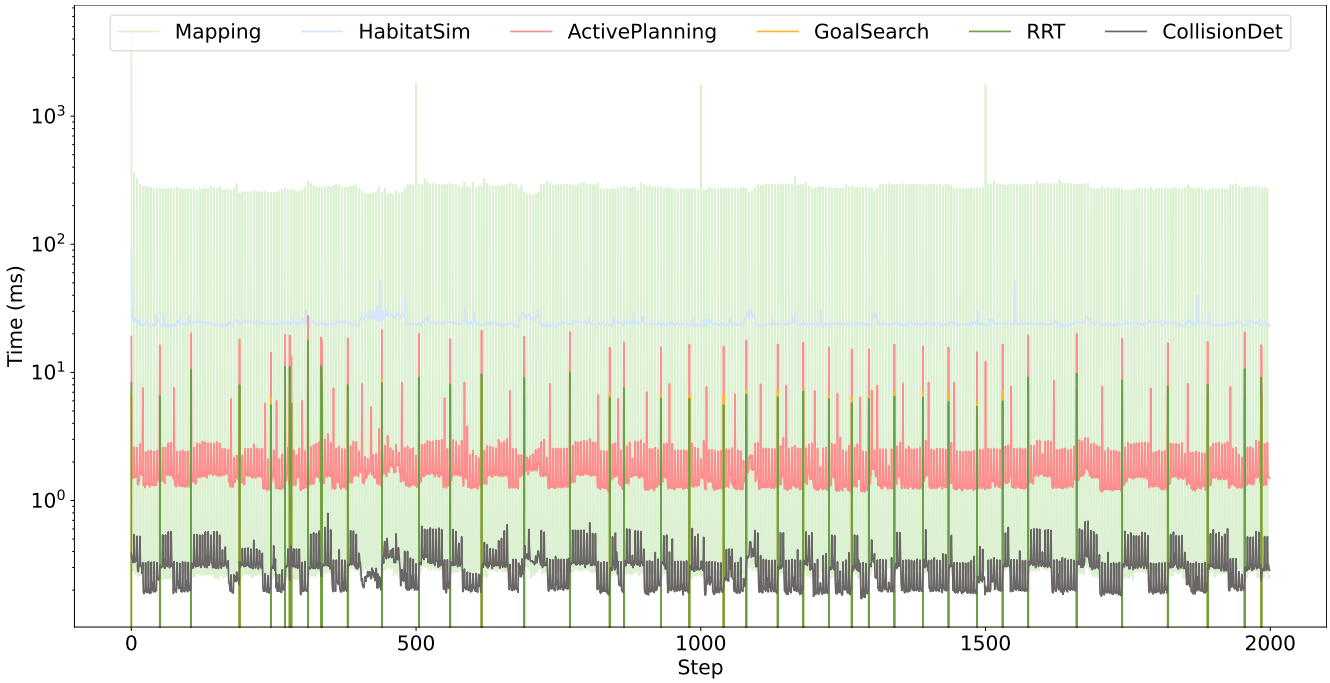

Figure 8. **Runtime Analysis in the Replica-room0 Environment** This figure illustrates the runtime analysis of each module within the *Replica-room0* environment. A notable runtime impulse is observed during goal-searching iterations. The analysis encompasses three principal modules: Habitat Simulator for data generation, Active Planning for path planning, and Mapping for mapping optimization. In the Active Planning module, further runtime analysis includes its submodules: Uncertainty-aware Goal Searching, RRT Path Planning, and Collision Detection.

study detailed in Tab. 3 highlights that our primary strategy for improvement involves identifying potential straight paths, drawing inspiration from RRT-Connect [34]. This approach, along with quicker tree growth, not only accelerates the planning process but also decreases memory usage.

9. Additional Experimental Results

9.1. Detailed results on MP3D and Replica

In this section, we present more comprehensive results for the various scenes included in the Matterport3D [8] and Replica dataset [67]. Detailed, scene-specific quantitative results are provided in Tab. 5 and Tab. 4. For the qualitative visualization, the reconstructed meshes undergo a culling process as delineated in Neural RGB-D [2] and GoSURF [74], ensuring that only the most relevant data is presented.

MP3D In Tab. 5, we present a comparative analysis of our method against the state-of-the-art Active Neural Mapping (ANM) [79]. The results demonstrate that our method outperforms ANM across all evaluated metrics. Most notably, our method exhibits a significant advancement in terms of reconstruction quality and completeness, surpassing the existing benchmarks set by previous art. This consistent superiority in performance underscores the effectiveness of our approach in challenging reconstruction scenarios.

In Fig. 9, we conduct a qualitative evaluation of our 3D reconstruction method against the ground truth for various scenes in the Matterport3D dataset. Ground truth meshes are presented in the odd-numbered rows, while the even-numbered rows showcase our method's reconstructed meshes. Each scene is identified by a unique code (*e.g.*, "Gdvg", "gZ6f") on the left. We offer a tripartite comparison for each: the first and second columns depict the exterior surfaces; the third and fourth columns reveal the interior surfaces; and the final two columns provide close-up views of the intricate internal reconstructions. This format delineates a comprehensive visual assessment, contrasting both the textural and geometric dimensions of the meshes.

In Fig. 11 through Fig. 15, we present per-scene trajectory visualizations on the Matterport3D dataset. For enhanced visual clarity, we focus exclusively on illustrating the trajectory formed by keyframe camera poses and the reconstructed texture mesh. To provide a thorough perspective of each scene, we include a bird's eye view alongside two distinct side views. This tri-view presentation facilitates a comprehensive understanding of the spatial dynamics in each scene. It is important to note that the "black regions" visible in the mesh represent areas lacking ground truth data, which were consequently excluded from the

Method	Metrics	office0	office1	office2	office3	office4	room0	room1	room2	Avg.
Neural SLAM										
Co-SLAM [73]	Acc. [cm] ↓	1.68	1.46	2.98	3.07	2.44	2.14	2.64	2.02	**2.30**
	Comp. [cm] ↓	1.68	1.82	2.70	2.83	2.64	2.25	2.84	2.02	**2.35**
	Comp. Ratio ↑	96.25	94.44	89.80	90.82	91.59	94.61	90.32	94.09	**92.74**
[73] **w/ ActRay**	Acc. (cm) ↓	1.61	1.48	2.96	3.12	2.43	2.17	2.58	2.00	**2.30**
	Comp. (cm) ↓	1.61	1.85	2.67	2.96	2.67	2.26	2.78	2.03	**2.35**
	Comp. Ratio ↑	96.24	94.44	90.61	89.85	91.51	94.66	90.23	94.08	92.70
Neural Mapping: Tracking is disabled.										
Co-SLAM [73]	Acc. [cm] ↓	1.50	1.28	2.56	2.69	2.25	2.01	1.55	1.87	**1.96**
	Comp. [cm] ↓	1.48	1.61	2.17	2.52	2.47	2.13	1.71	1.88	2.00
	Comp. Ratio ↑	96.33	94.65	92.47	91.43	91.34	94.67	95.45	93.95	93.79
[73] **w/ ActRay**	Acc. (cm) ↓	1.47	1.27	2.55	2.71	2.26	2.02	1.57	1.87	**1.96**
	Comp. (cm) ↓	1.47	1.59	2.13	2.55	2.49	2.07	1.71	1.85	**1.98**
	Comp. Ratio ↑	96.44	94.80	92.90	91.32	91.32	94.92	95.40	94.12	**93.90**
Neural Active Mapping										
w/o ActiveRay	Acc. (cm) ↓	1.29	1.05	2.17	2.86	1.72	1.56	1.24	1.46	1.67
	Comp. (cm) ↓	1.40	1.50	1.66	3.14	1.76	1.67	1.45	1.47	1.76
	Comp. Ratio ↑	97.92	95.87	98.04	90.68	98.09	98.31	97.62	98.55	96.89
Uncertainty Net	Acc. (cm) ↓	1.32	1.05	2.04	3.13	1.70	1.58	1.26	1.45	1.69
	Comp. (cm) ↓	2.12	2.01	2.73	2.50	2.07	1.90	1.58	1.56	2.06
	Comp. Ratio ↑	94.21	93.22	92.62	92.12	94.24	96.36	96.65	97.54	94.62
Full	Acc. (cm) ↓	1.30	1.03	2.25	2.29	1.75	1.56	1.25	1.47	**1.61**
	Comp. (cm) ↓	1.39	1.53	1.69	2.27	1.79	1.68	1.43	1.48	**1.66**
	Comp. Ratio ↑	98.17	95.26	97.54	93.91	97.93	98.28	98.04	98.47	**97.20**

Table 4. Per-scene quantitative results on Replica[67] dataset

Method	Metric	Gdvg	gZ6f	HxpK	pLe4	YmJk	**Avg.**
ANM [79]	MAD (cm) ↓	3.77	3.18	7.03	3.25	4.22	4.29
	Acc. (cm) ↓	5.09	4.15	15.60	5.56	8.61	7.80
	Comp. (cm) ↓	5.69	7.43	15.96	8.03	8.46	9.11
	Comp. Ratio ↑	80.99	80.68	48.34	76.41	79.35	73.15
Ours	MAD (cm) ↓	1.60	1.23	1.53	1.37	1.45	**1.44**
	Acc. (cm) ↓	3.78	3.36	9.24	5.15	10.04	**6.31**
	Comp. (cm) ↓	2.91	2.31	2.67	3.24	3.86	**3.00**
	Comp. Ratio ↑	91.15	95.63	91.62	87.76	84.74	**90.18**

Table 5. **Per-scene quantitative results on Matterport3D [8] dataset**. Our method achieves consistently better reconstruction than the state-of-the-art method ANM [79].

mapping optimization process. Our observations indicate that while our method demonstrates high completeness in fully exploring the environment, it tends to allocate a considerable number of steps to survey these "black regions". This behavior can be attributed to our selective exclusion of these regions during mapping optimization, which in turn, prevents effective reduction of uncertainty in these areas. Our method, prioritizing observation of uncertain regions, thus allocates more attention to these parts. This phenomenon is a reflection of the challenges posed by the imperfect simulation of real-world environments.

Replica We present per-scene ablation studies on Replica in Tab. 4. These results demonstrate that Active Ray Sampling enhances the performance of CoSLAM [73], particularly in scenarios where tracking is disabled. Additionally, our ablation studies reveal that employing the Uncertainty Grid (Full) approach yields superior results compared to the Uncertainty Net across most scenes.

In Fig. 10, we conduct a qualitative evaluation of our 3D reconstruction method against the ground truth for various scenes in the Replica dataset. Ground truth meshes are presented in the odd-numbered rows, while the even-numbered rows showcase our method's reconstructed meshes. Our results show a high level of quality and completeness, closely mirroring the ground truths.

In Fig. 16 - Fig. 23, we present trajectory visualization for each scene. Given that five trials were conducted for each scene, we selectively showcase the most illustrative visualization result for demonstration purposes. In our qualitative analysis, we present two key elements for each scene: the texture mesh visualization and the corresponding planned trajectory. Similarly, we only illustrate the trajectory formed by keyframe camera poses and the reconstructed texture mesh for better clarity.

Method	Metrics	office0	office1	office2	office3	office4	room0	room1	room2
CoSLAM (no tracking)	Comp. Ratio ↑	96.33	94.65	92.47	91.43	91.34	94.67	95.45	93.95
	Traj. (m) ↑	18.20	11.56	23.16	29.16	25.22	24.69	16.21	23.07
Ours	Comp. Ratio ↑	98.17	95.26	97.54	93.91	97.93	98.28	98.04	98.47
	Traj. (m) ↑	81.27	30.02	90.20	88.59	96.36	73.91	96.99	41.31

Table 6. Per-scene trajectory length evaluation on Replica[67] dataset

9.2. More qualitative comparison on MP3D

For the completeness of the study, we provide more comparison between ground truth, ANM baseline [79], and our method in Matterport3D dataset, as shown in Fig. 24. We trim the meshes for a better visualization purpose.

9.3. Comparison against passive mapping methods

In traditional mapping methods, typically involving environments scanned by human-operated sensing devices, the trajectory of scanning significantly impacts the reconstruction's quality and completeness. Such approaches are termed passive mapping methods, characterized by the absence of a planning or guidance module. In Tab. 2, we present a quantitative comparison between Passive Neural Mapping and Active Neural Mapping, utilizing Co-SLAM as the backbone. Here, we aim to offer additional qualitative comparisons in Fig. 25 to highlight differences in reconstruction details more vividly. In passive Co-SLAM (with tracking disabled), regions may be missed or poorly reconstructed if not adequately covered by the scanning trajectory. Conversely, our active reconstruction method ensures a more comprehensive and accurate reconstruction, effectively addressing these limitations.

We compared the trajectory lengths of passive versus active scanning on the Replica dataset, with the results detailed in Tab. 6. Under the same conditions (2000 frames with 400 keyframes), passive scanning may result in redundant observations due to the lack of guided scanning. Active scanning, on the other hand, enables more extensive coverage and yields superior reconstruction quality. However, this approach typically results in longer trajectories, as the agent continuously moves to ensure comprehensive scanning of the environment.

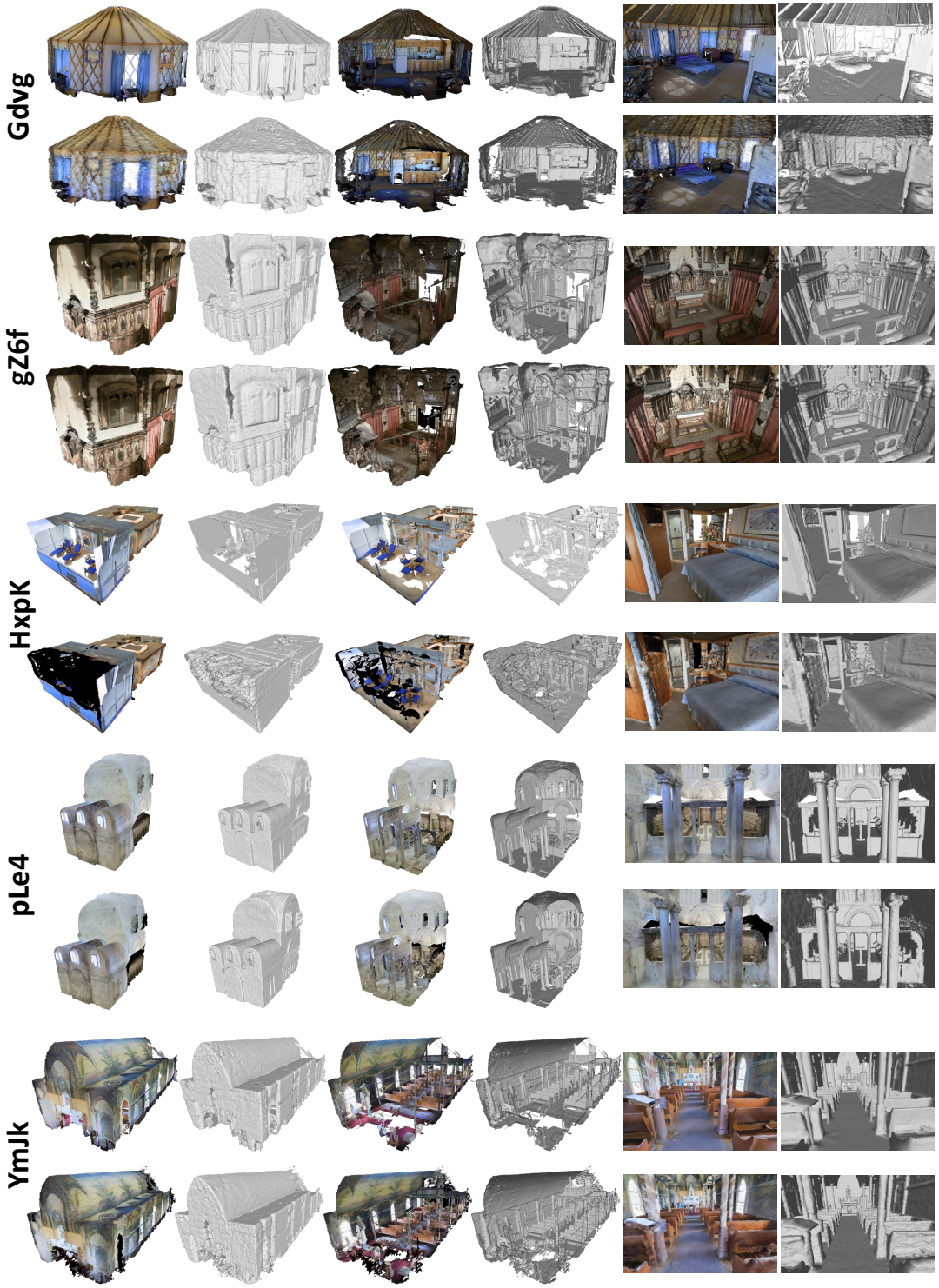

Figure 9. **MP3D Reconstruction Results** This presents a side-by-side comparison of the reconstruction results with the Matterport3D dataset. The odd-numbered rows display the ground truth meshes, while the even-numbered rows feature the meshes reconstructed by our method. Our results show a high level of quality and completeness, closely mirroring the ground truths. This alignment underscores the efficacy of our method in accurately exploring and reconstructing complex spatial geometries.

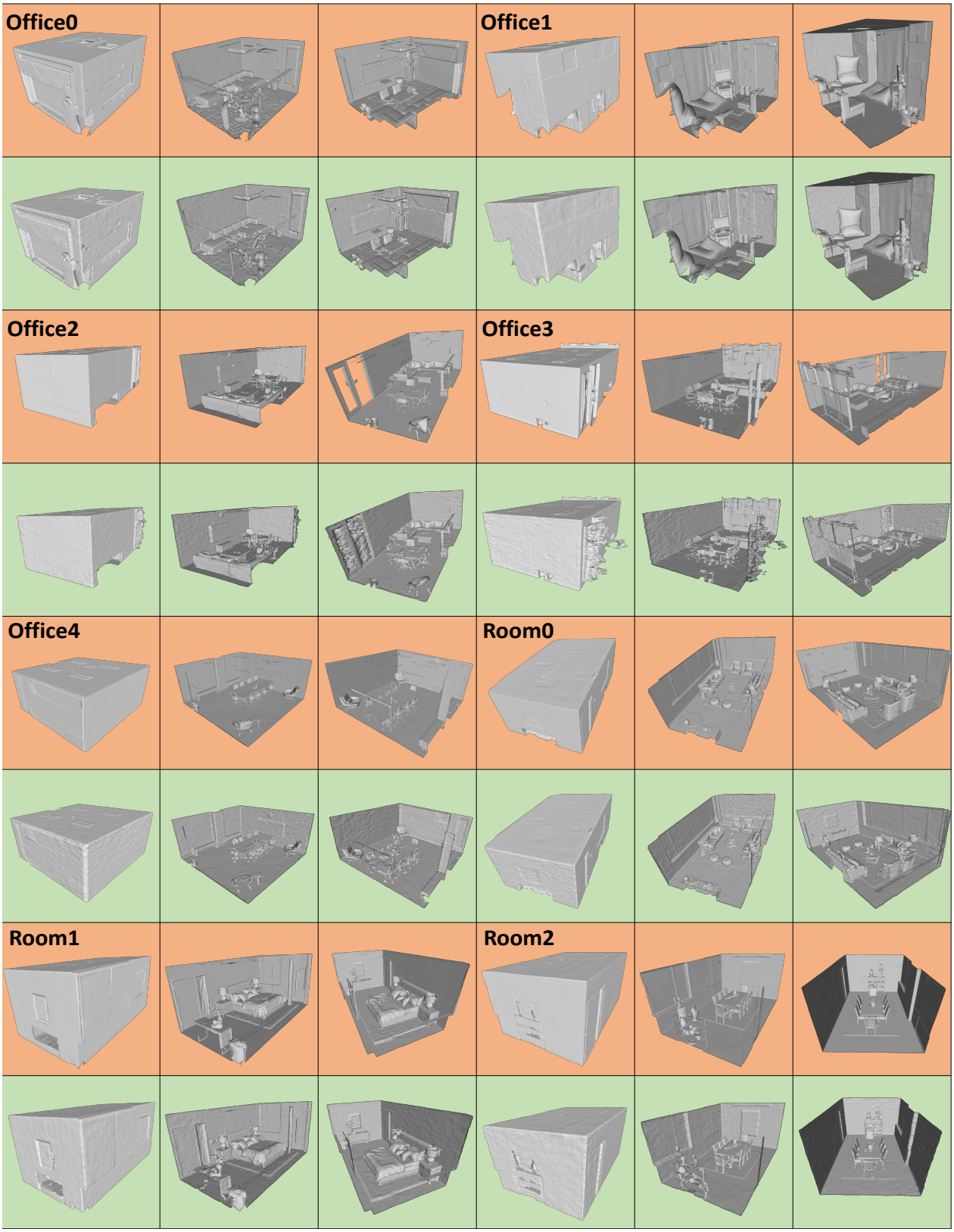

Figure 10. **Replica Reconstruction Results** This presents a side-by-side comparison of the reconstruction results with the Replica dataset. The odd-numbered rows display the ground truth meshes, while the even-numbered rows feature the meshes reconstructed by our method. Our results show a high level of quality and completeness, closely mirroring the ground truths.

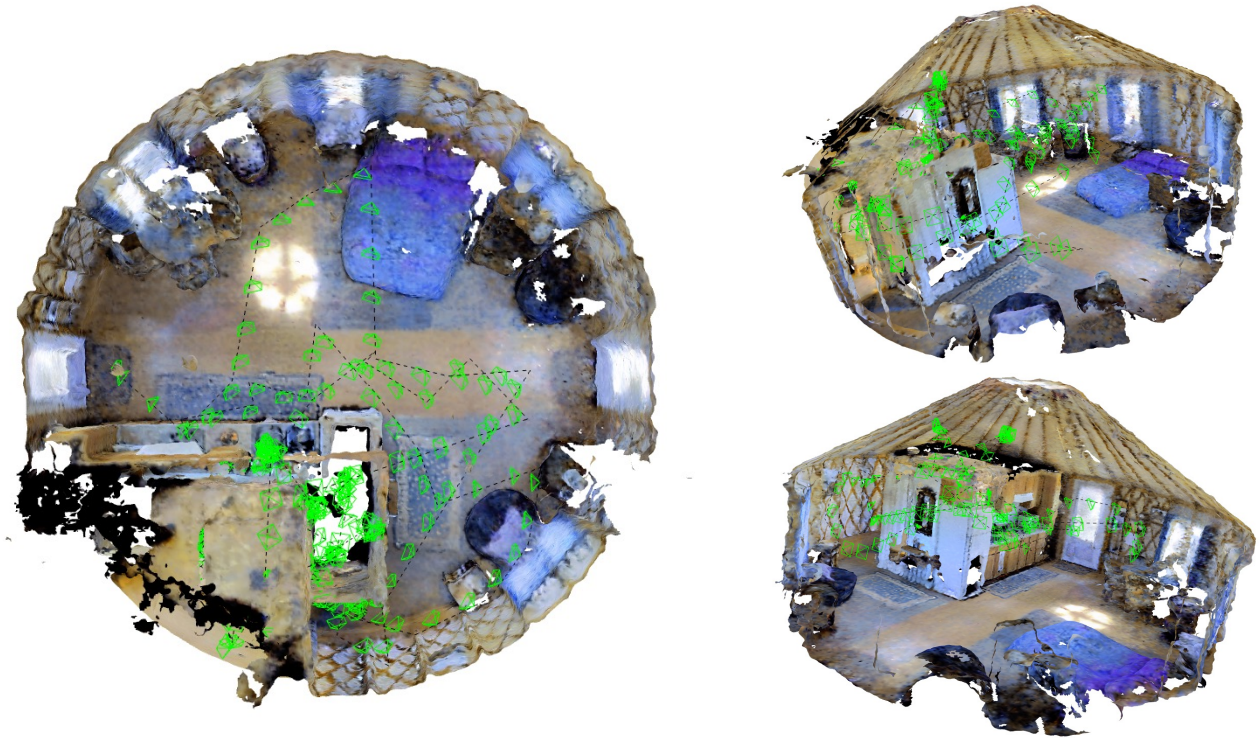

Figure 11. **Matterport3D (Gdvg)** Reconstructed Mesh and planned trajectory.

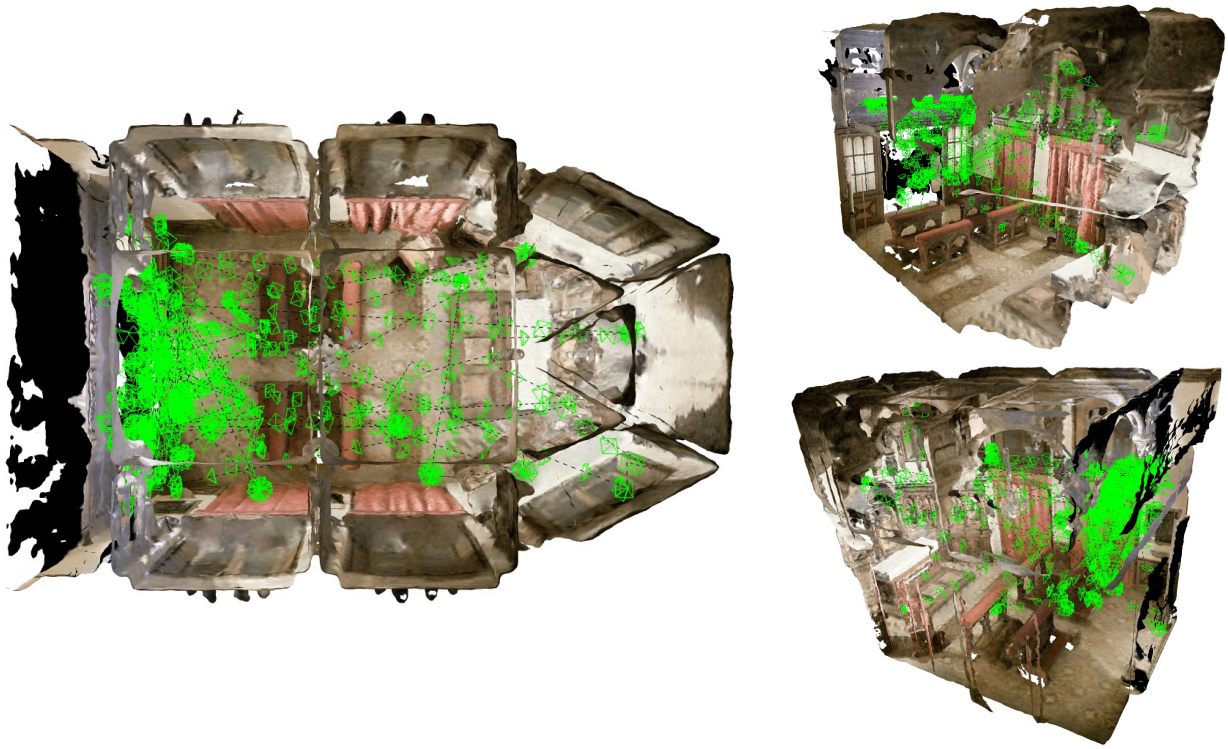

Figure 12. **Matterport3D (gZ6f)** Reconstructed Mesh and planned trajectory.

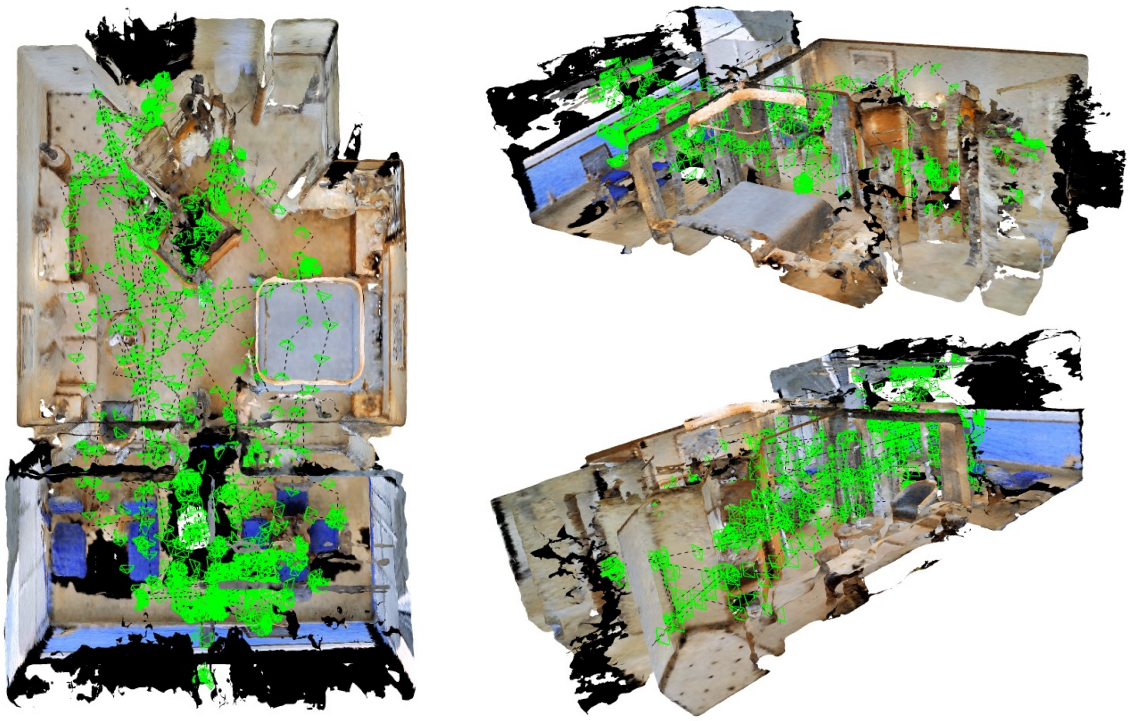

Figure 13. **Matterport3D (HxpK)** Reconstructed Mesh and planned trajectory.

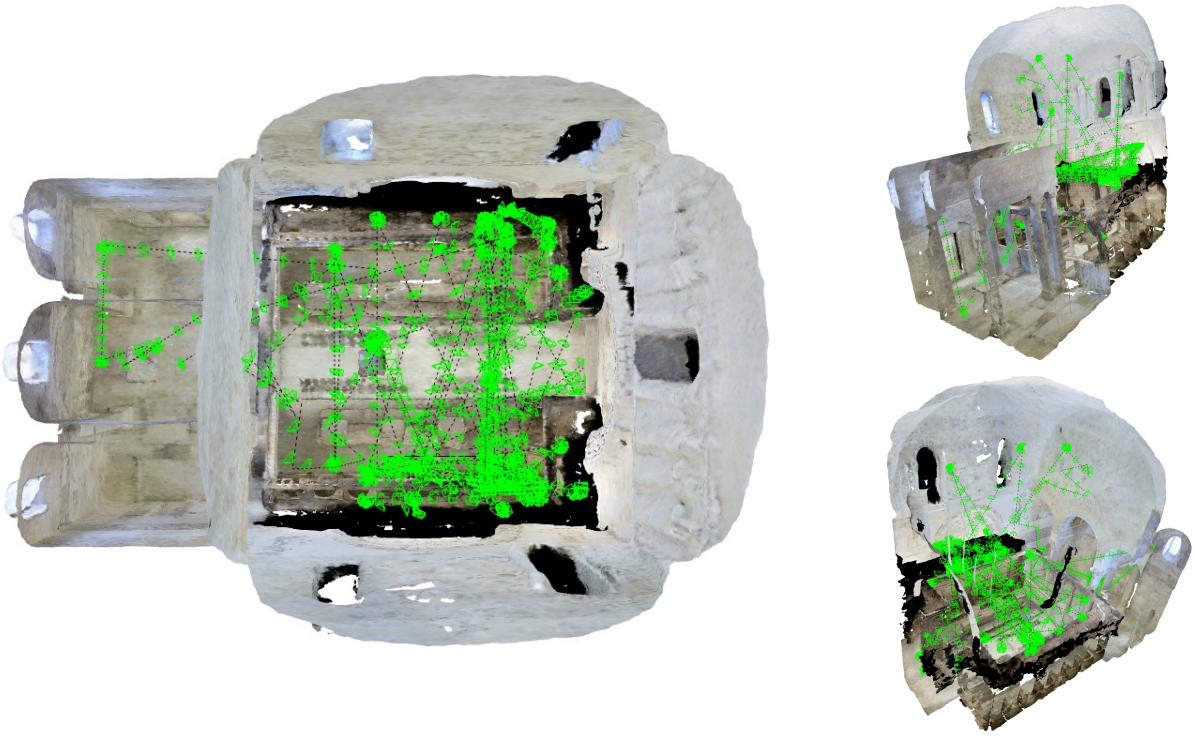

Figure 14. **Matterport3D (pLe4)** Reconstructed Mesh and planned trajectory.

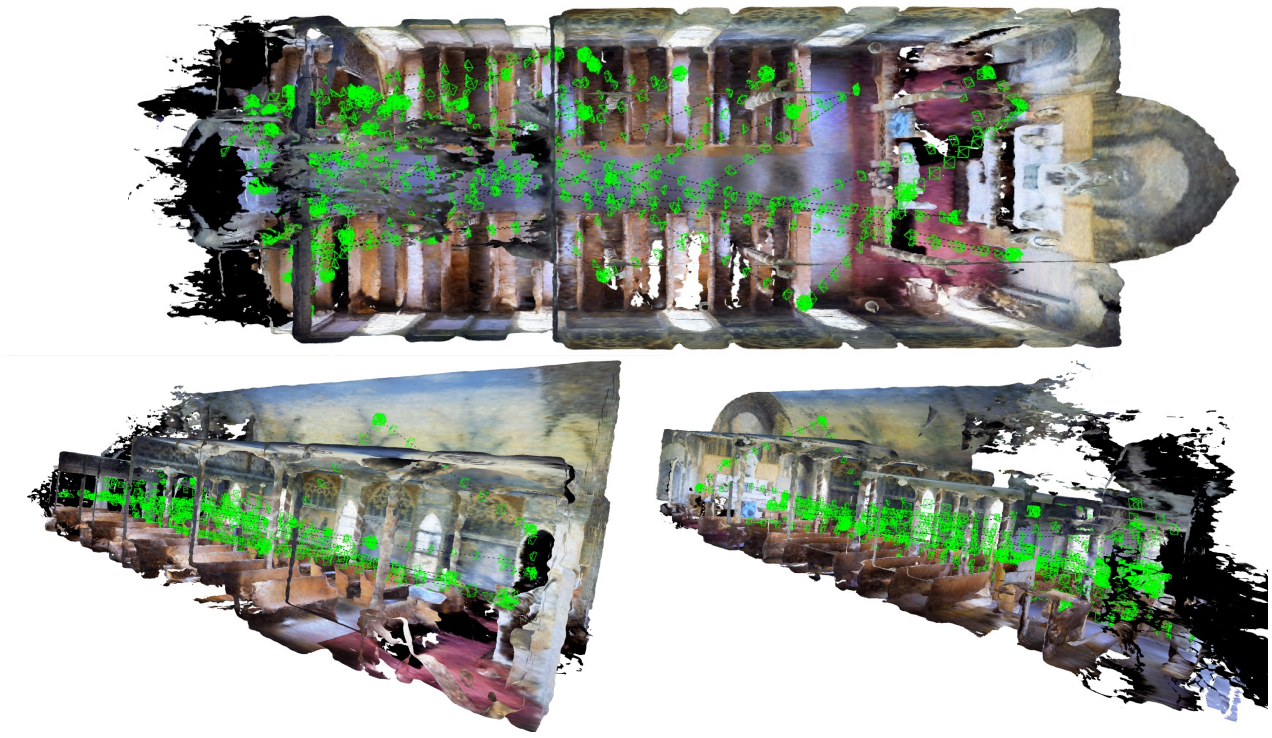

Figure 15. **Matterport3D (YmJk)** Reconstructed Mesh and planned trajectory.

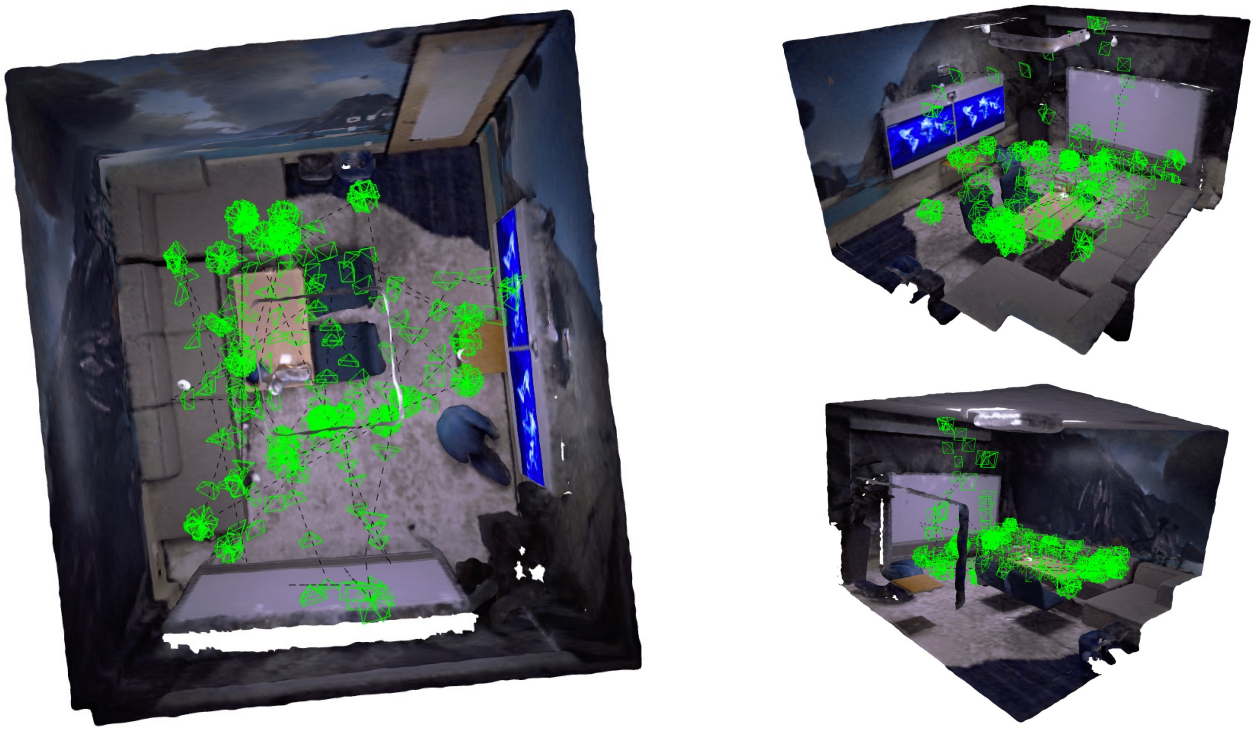

Figure 16. **Replica (office0)** Reconstructed Mesh and planned trajectory.

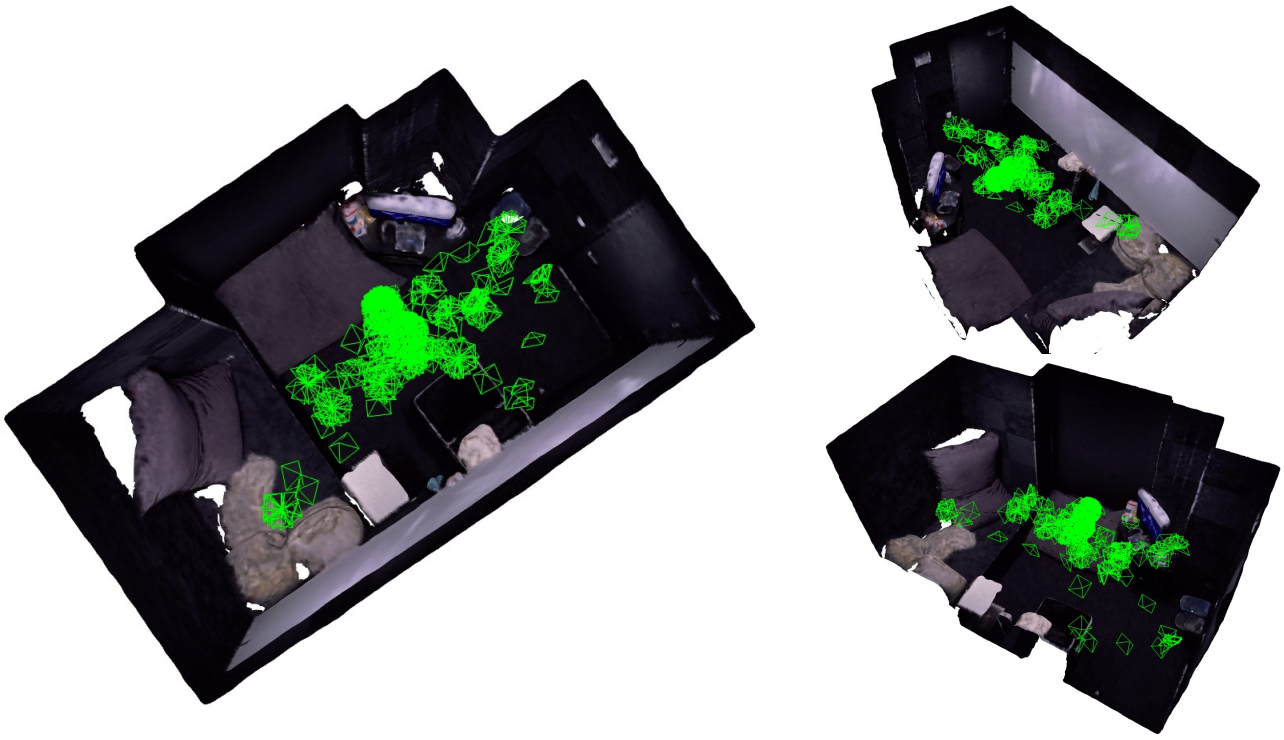

Figure 17. **Replica (office1)** Reconstructed Mesh and planned trajectory.

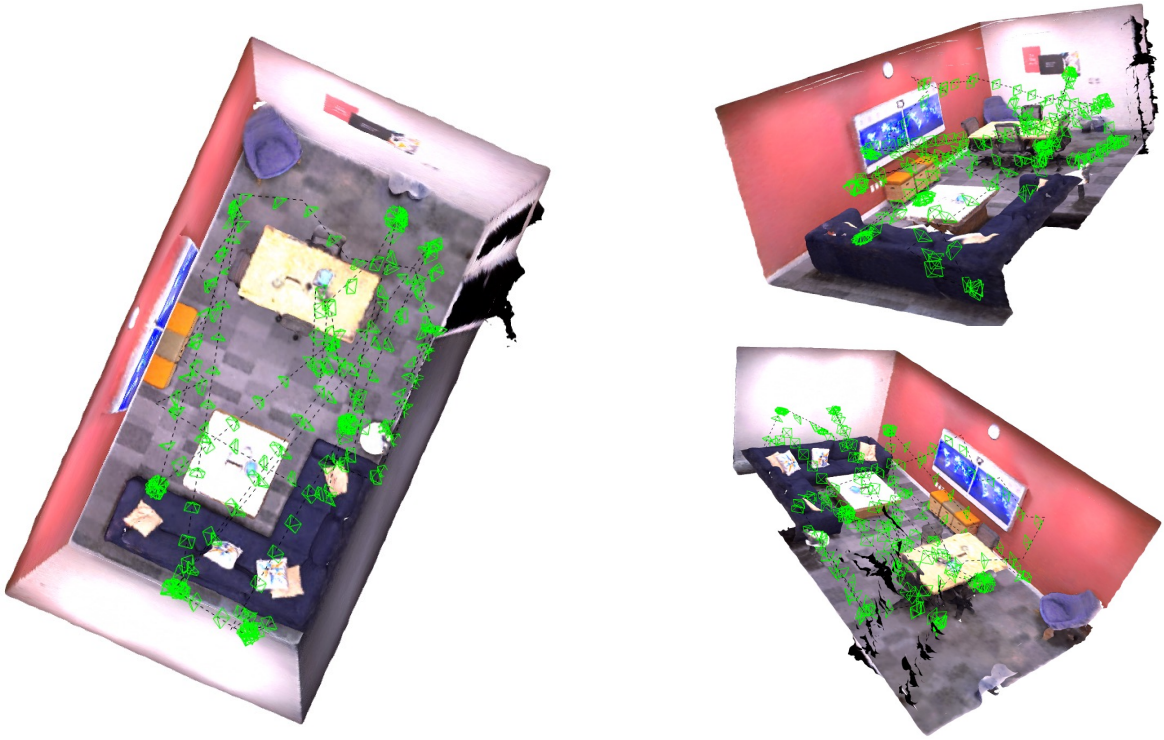

Figure 18. **Replica (office2)** Reconstructed Mesh and planned trajectory.

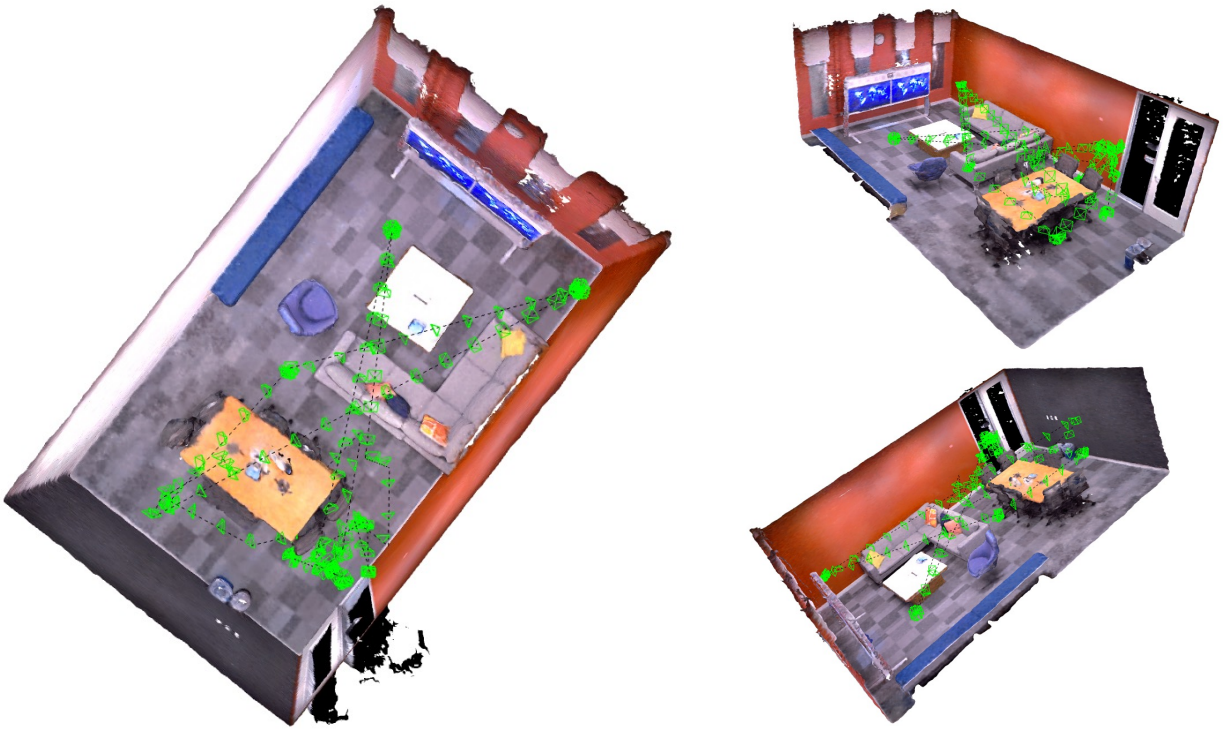

Figure 19. **Replica (office3)** Reconstructed Mesh and planned trajectory.

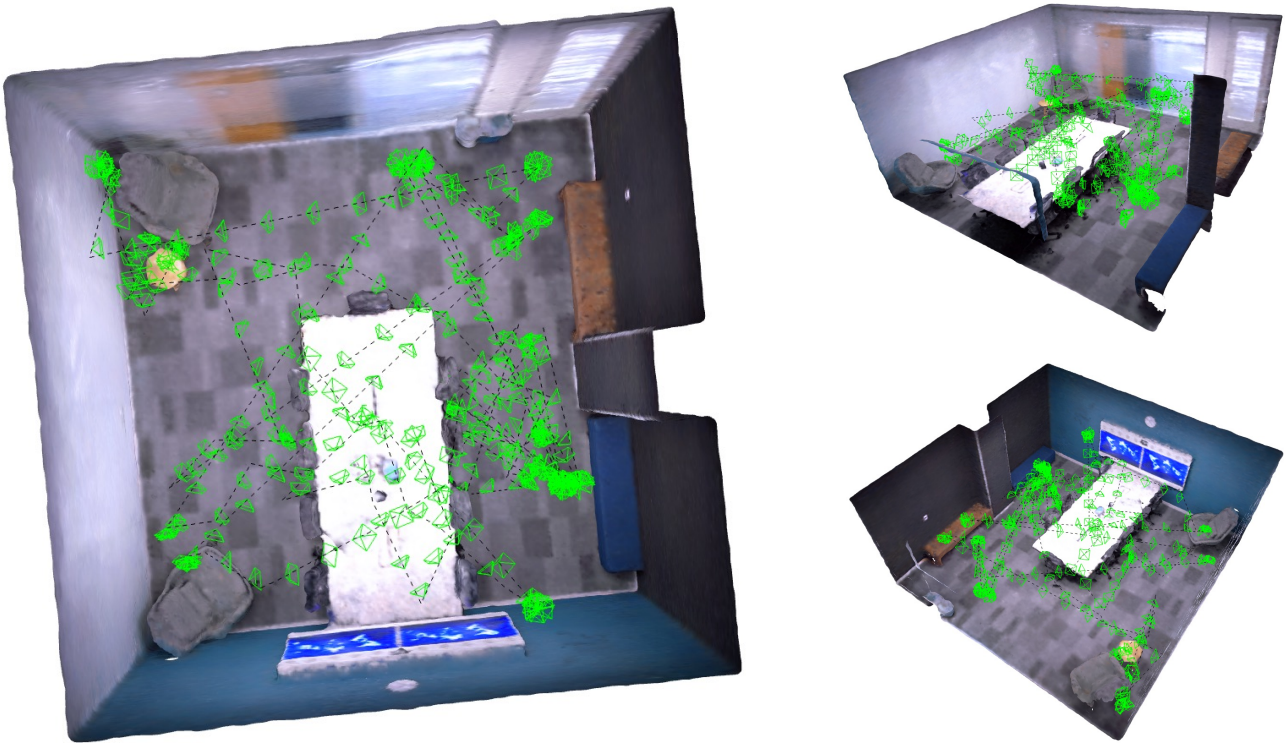

Figure 20. **Replica (office4)** Reconstructed Mesh and planned trajectory.

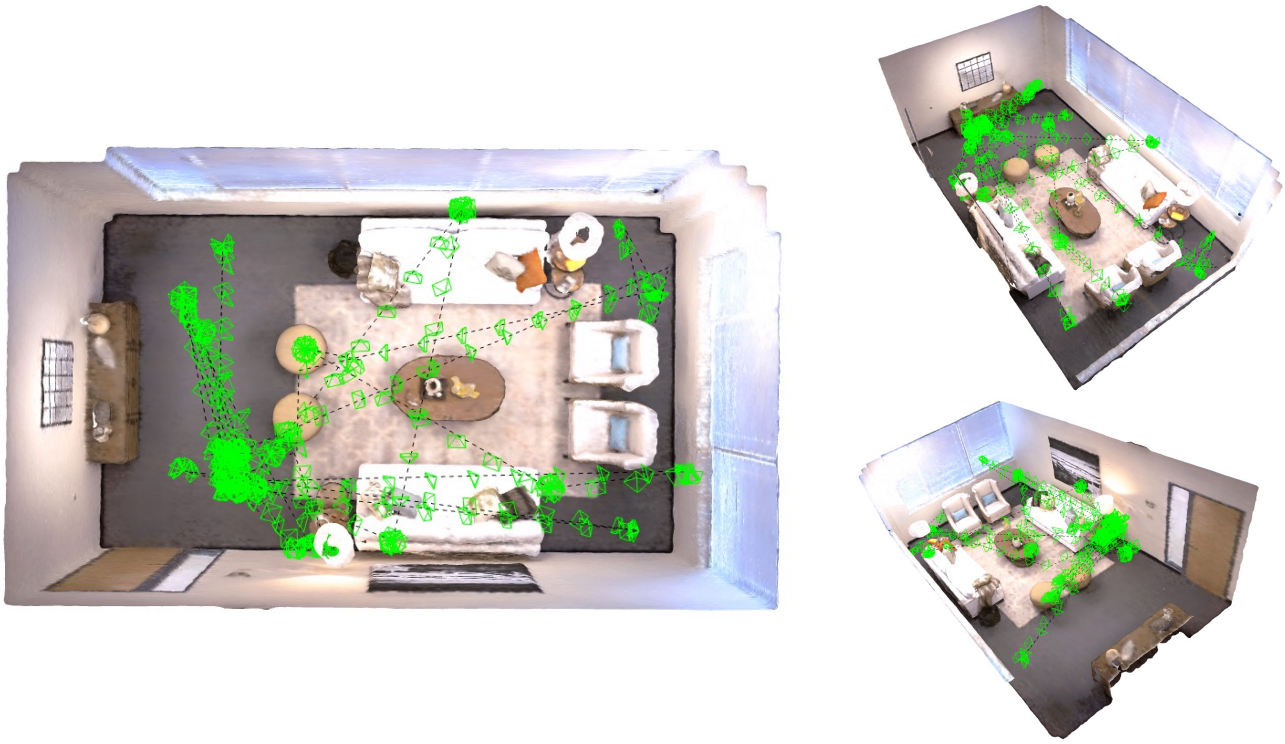

Figure 21. **Replica (room0)** Reconstructed Mesh and planned trajectory.

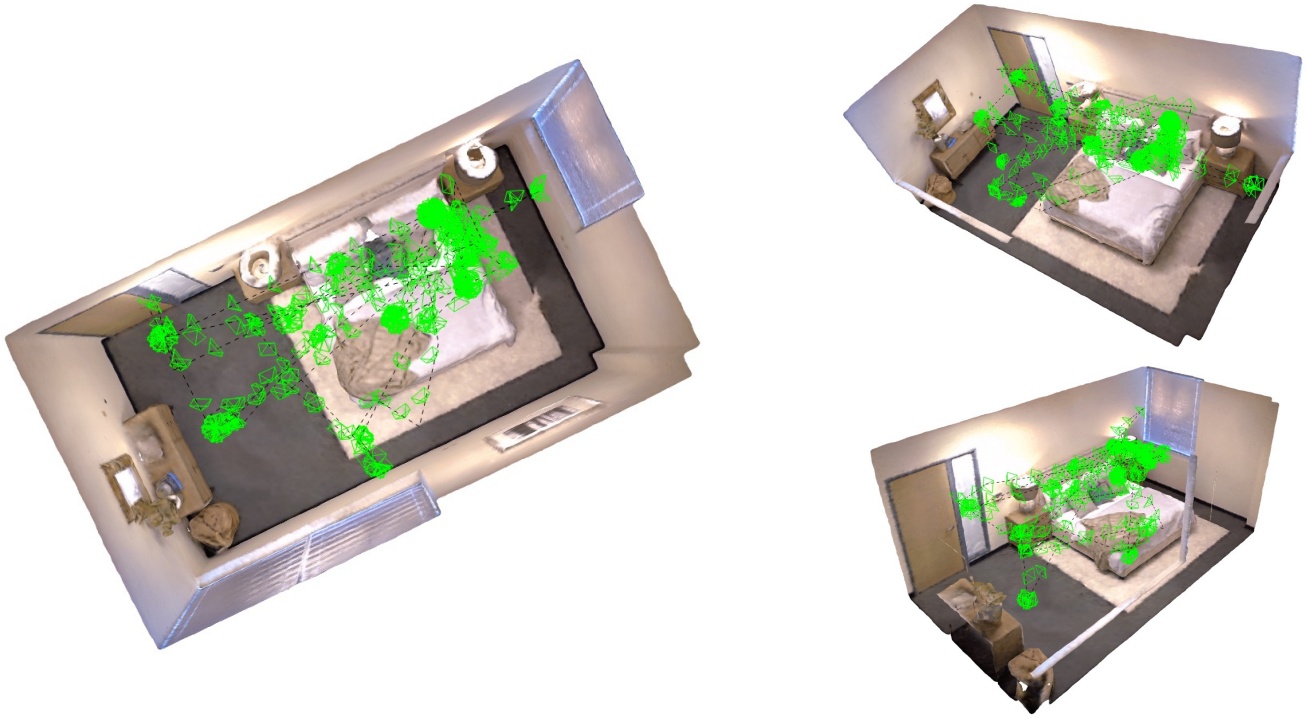

Figure 22. **Replica (room1)** Reconstructed Mesh and planned trajectory.

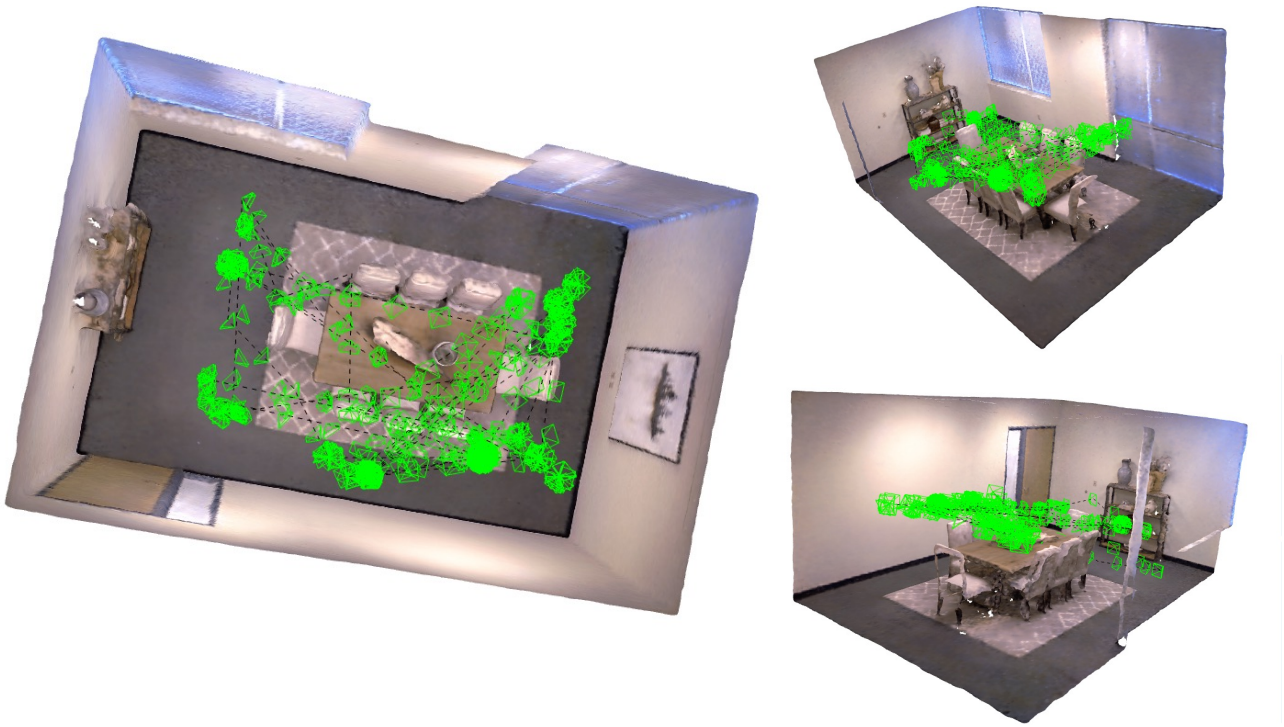

Figure 23. **Replica (room2)** Reconstructed Mesh and planned trajectory.

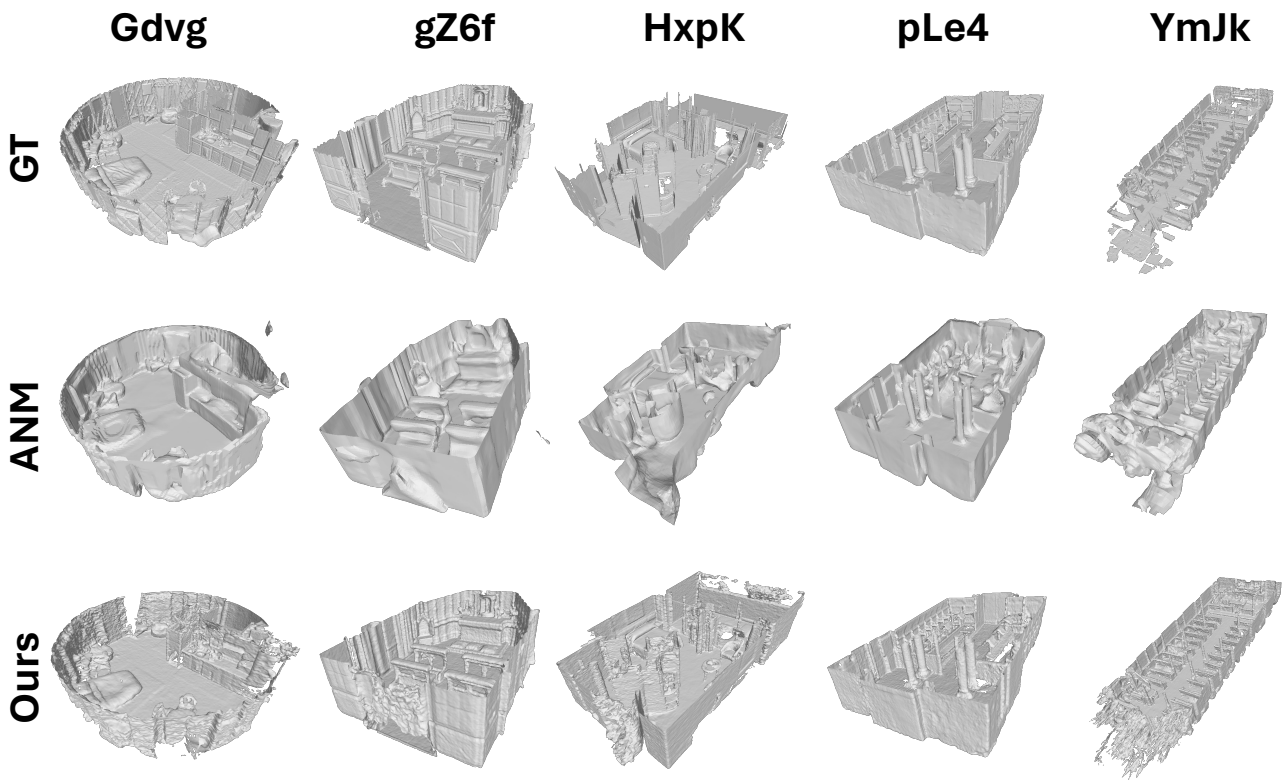

Figure 24. **More Matterport3D results** We trim the reconstruction results for a better comparison. Compared to the baseline method, ANM [79], our method shows more precise and complete reconstructions.

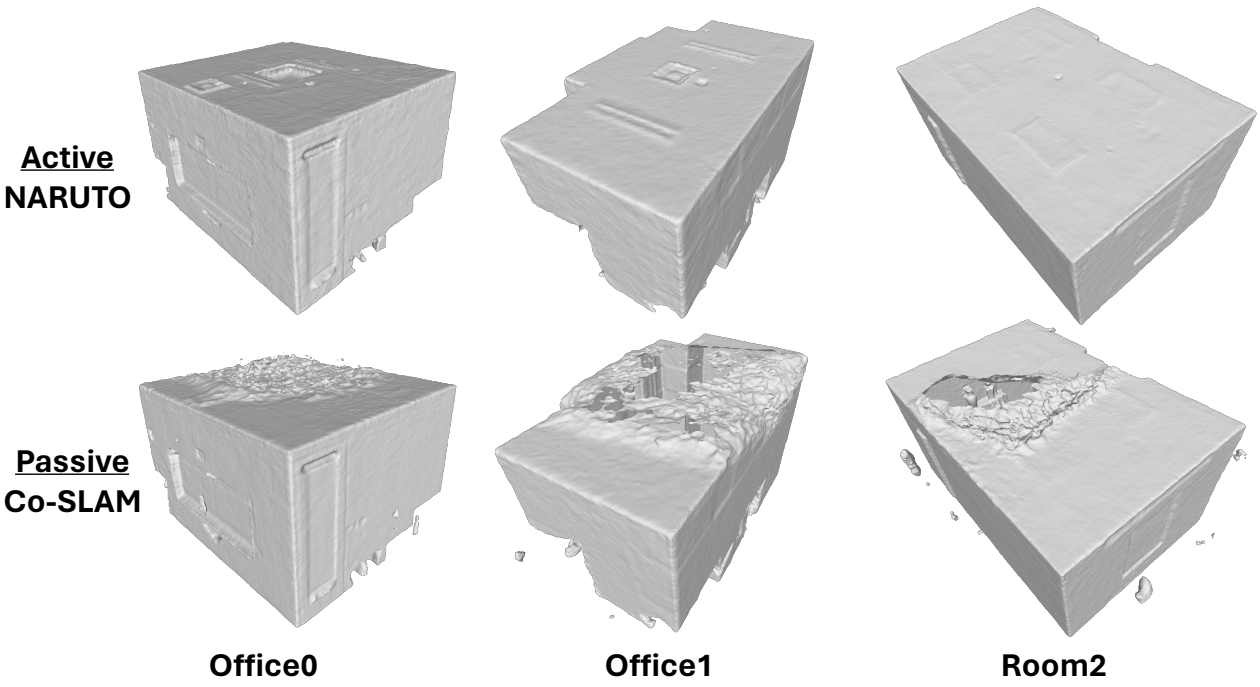

Figure 25. **Qualitative comparison between active and passive mapping methods.** For Co-SLAM [73], we disable the tracking thread and run the reconstruction using a pre-defined trajectory. Active NARUTO shows a more complete and precise reconstruction, especially for the regions that have not been adequately covered by the passive scanning.