# VI-NeRF-SLAM: A Real-time Visual-Inertial SLAM with NeRF Mapping

DaoQing Liao
Wei Ai ( aiwei@scut.edu.cn )

Research Article

**Keywords:** NeRF, SLAM, Intelligent Map, Real- time Online Algorithm

**Posted Date:** December 8th, 2023

**DOI:** https://doi.org/10.21203/rs.3.rs-3710160/v1



**Additional Declarations:** No competing interests reported.

# VI-NeRF-SLAM: A Real-time Visual-Inertial SLAM with NeRF Mapping

**DaoQing Liao**[1], **Wei Ai**[1]

**Abstract** In numerous robotic and autonomous driving tasks, traditional visual SLAM algorithms estimate the camera's position in a scene through sparse feature points and express the map by estimating the depth of sparse point clouds. However,practical applications require SLAM to create dense maps in real-time, overcoming the sparsity and occlusion issues of point clouds. Furthermore,it is advantageous for SLAM map to possess an auto-completion capability, where the map can automatically infer and complete the remaining 20% when the camera observes only 80% of an object. Therefore, a more dense and intelligent map representation is needed. In this paper, we propose a Visual-Inertial SLAM with Neural Radiance Fields reconstruction to address the aforementioned challenges. We integrate traditional rule-based optimization with NeRF. This approach allows for the real-time update of NeRF local functions by rapidly estimating camera motion and sparse feature point depths to reconstruct 3D scenes. To achieve better camera poses and globally consistent map, we address the issue of IMU noise spikes resulting from rapid motion changes, along with handling pose adjustments due to loop closure fusion. Specifically, we employ a form of widening the static noise covariance to refit the dynamic noise covariance. During loop closure fusion, we treat the pose adjustment between pre- and post- loop closure as a spatiotemporal transformation, migrating NeRF parameters from pre- to post- to expedite loop closure adjustments in NeRF mapping. Moreover, we extend this method to scenarios with only grayscale images. By expanding the color channels of grayscale images and conducting linear spatial mapping, we can rapidly reconstruct 3D scenes with only grayscale images. We demonstrate the precision and speed advantages of our method in both RGB and grayscale scenes.

**Keywords** NeRF · SLAM · Intelligent Map · Real-time Online Algorithm

DaoQing Liao
School of Automation Science and Engineering, South China University of Technology, guangzhou, China
E-mail: ldq4399@163.com

Wei Ai
School of Automation Science and Engineering, South China University of Technology, guangzhou, China
E-mail: aiwei@scut.edu.cn

## 1 Introduction

Reconstructing 3D scenes and camera localization is currently one of the more challenging computer vision tasks. At the same time, localization and map perception are also fundamental modules for most robot movements, such as autonomous driving, home service robots and VR games. Therefore, for a SLAM algorithm, the following requirements should be met: first, the system must be real-time, and the algorithm should be measurable or estimated in absolute scale or need to know the ratio of the constructed map to the real scene. On the other hand, the algorithm should have strong generalization, so the SLAM task needs to be carried out without pre-training. In traditional non-learning-dense SLAM algorithms, RGBD cameras or stereo cameras are generally used to directly measure the depth values of points [10, 15, 31, 40]. However, RGBD are susceptible to light interference and stereo require a longer baseline to reduce uncertainty. In learning-based SLAM algorithms, they rely too much on network pre-training or prior conditions [34, 36, 43].Therefore, this article aims to develop a real-time and dense Visual or Visual-Inertial SLAM system that combines the advantages of Nerf. In Visual dense reconstruction, previous

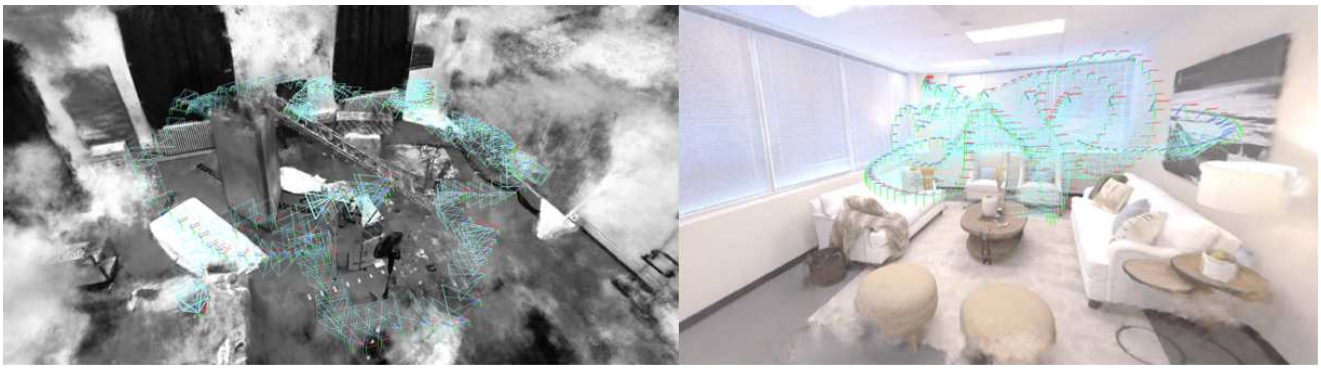

**Fig. 1** We have developed a SLAM system that can perform real-time sparse tracking and dense reconstruction.The system estimates the pose based on feature point reprojection constraints and utilizes NeRF mapping for dense reconstruction.The left image shows the reconstruction results in a gray scene,the right image shows the results in an RGB scene.our model is real-time and pre-training-free.

methods generally used voxel representations of TSDF [15, 27, 28]. Tandem not only surpasses other SOTA SLAM methods in camera tracking but also demonstrates SOTA 3D reconstruction performance. However, its heavy reliance on depth estimation networks means that the reconstruction results are influenced by monocular depth estimation. Moreover, in terms of generalization, depth estimation often exhibits suboptimal performance.

In the latest research in computer vision, NeRF have achieved great success in new view synthesis [1,2,24,26]. After training from various scene perspectives, NeRF can freely render observed images from any viewpoint, making the fusion of NeRF Mapping and rule-base-SLAM algorithms a potential research direction [34,36, 43]. In iMap [36], encoding scene features in an MLP network to update is inefficient, and the singularity of spatial partitioning leads to forgetfulness and reconstruction details missing problems. NICE-SLAM [43] improved the iMap algorithm and proposed a multi-scale spatial partitioning and feature grid to store scene feature encoding, updating only the feature encoding under the camera view to avoid network forgetfulness. Therefore, NICE-SLAM expands the application of iMap algorithms to large-scale scenes and can reconstruct more details. However, the above NeRF-based SLAMs requires depth image. They need to estimate the pose of the new image using a previously converged scene and use the estimated pose and depth map to supervise the convergence of the neural radiance field model. Lack of depth or replacing RGB with grayscale image will lead to catastrophic failure. Moreover, due to the indiscriminate grid feature encoding partitioning, the algorithm requires a large amount of memory and convergence is slow, making it difficult to achieve real-time performance. NeRF-SLAM [34] separates the pose estimation and NeRF mapping, uses Droid-SLAM [38] to estimate dense optical flow of the image to obtain the initial pose and local depth of the image to supervise the training of the Instant-NGP. Because the scene model is updated using hash coding, the scene convergence can achieve real-time performance. However, the heavy reliance on the Droid-SLAM network for depth supervision makes the whole model very large, and Droid-SLAM also requires long training time to use, making it difficult to achieve real-time performance and meet the requirements of pre-training-free scene models. To address the aforementioned issues, we developed a SLAM algorithm called VI-NeRF-SLAM (Fig. 1) that can perform dense 3D scene reconstruction in real-time online without the need for pre-training, by combining the advantages of non-learning SLAM methods [4, 12, 32] and new view synthesis. It supports monocular and monocular-inertial fusion.For image processing, we utilize SuperPoint [11] and LightGlue [21] for local feature extraction and matching. Simultaneously, we incorporate inertial measurements to rapidly estimate the camera's motion state and the depth of sparse point clouds. This information is used to supervise the training of a neural radiance field based on multi-resolution hash encoding. Sparse depth not only facilitates depth supervision but also helps maintain sampled points near the zero-level set of objects, enabling faster learning of object surfaces. During the NeRF optimization stage, we concurrently optimize the camera and scene representations. In summary, our contributions are as follows:

– We propose VI-NeRF-SLAM,a SLAM system that combines non-learning methods for pose tracking with NeRF mapping. It does not require pre-training and can be applied to real-time 3D dense reconstruction in most scenarios. Additionally, we extend the sensor input to not only support monocular dense

reconstruction but also incorporate IMU input, enabling the recovery of absolute scale for scene modeling.
- We introduce a robust IMU noise handling module, which effectively deals with the uncertainty and catastrophic failures caused by long-term integration when visual frame rates are low and IMU performance is poor.
- We propose a method to handle loop closure adjustments in NeRF mapping. In rule-based optimization-based SLAM systems, loop closure fusion can address the issue of cumulative errors. However, it introduces global pose adjustments, leading to overall spatial changes. We treat the changes before and after loop closure as dynamic transformations of the scene, estimating these transformations to remap sample points back to their original radiance fields for better reuse of previous optimization information.
- In challenging dataset tests,extensive validation against SOTA neural rendering-based SLAM algorithms, as well as evaluations of grayscale scene reconstruction, demonstrate the superior performance of our algorithm.

## 2 Related Work

We will review two different lines of work and explore their joint efforts.

### 2.1 Visual And Visual-Inertial State Estimator

In recent years, SLAM methods based on visual or visual-inertial fusion have gradually gained favor among researchers due to their low cost and small sensor systems compared to other sensor systems-based SLAM methods. Monocular SLAM methods often have the characteristics of high robustness and good positioning accuracy. However, their shortcomings are also evident, namely, they cannot recover the true scale of trajectory and scene models solely through monocular cameras without prior knowledge. Therefore, in order to better apply them to practical processes, monocular cameras are often tightly coupled with inertial measurements to recover the absolute scale of monocular cameras. The earliest tightly coupled method appeared in MSCKF [25], which regards camera poses as system states and feature points as state constraints, and uses sliding window filtering to update and solve the problem of dimension explosion in EKF-SLAM. Based on [19,29,30], MSCKF was further improved and extended to stereo vision by the team of OKVIS [16,17]. Experiments have shown that batch nonlinear optimization methods often have higher accuracy than recursive filtering methods. VINS-Mono provides a lightweight front-end complete SLAM system based on LK optical flow. Experiments have shown that front-end matching based on optical flow is more robust and faster than descriptor matching. VINS-Fusion [32, 33] extends it to Stereo and Stereo-inertial. ORB-SLAM3 [4] provides a fully descriptor-based SLAM system, which use a multi-map system further ensures robustness under extreme conditions such as texture distortion, and is suitable for various sensor schemes.

### 2.2 Map Representation

In SLAM systems, map representation is a crucial component. Researchers have developed various forms of map representation (point clouds, grids, topological, voxels, etc.) depending on various applications. However, the principle remains the same, which is to construct map directly or indirectly using visual landmarks, with the colors of objects in visual images depending solely on the colors of the point clouds. With the development of rendering theory, new view synthesis has gradually demonstrated its superior characteristics. [6, 24, 27] have shown that implicit neural representations excel in parametric geometry compared to point and grid-based representations and seamlessly allow for learning of prior shapes. The implicit representation of NeRF [1, 2, 5, 8, 23, 24, 26, 39] has achieved tremendous success in novel view synthesis, and rendering theory and neural radiance fields offer another solution for map construction. We obtain occupancy and color information of spatial points on the path from the radiance field to render views. Therefore, we do not need to represent all point clouds but instead find an implicit function that approximates the scene. The initial NeRF [24] proposed by Mildenhall,etc. stores features in MLP networks, which makes the network updates slow, and convergence of the radiance field model takes a long time. [37,42] have shown that storing feature vectors in a grid structure, even without the need for MLP networks, can rapidly converge the neural radiance field model. Instant-ngp [26] further stores features in a hash table, and by combining multi-resolution hash encoding with ray-marching strategies, the convergence speed of the neural radiance field is improved to seconds, achieving real-time effects. F2-NeRF [39] further optimizes the allocation of spatial resources based on Instant-ngp. Therefore, the combination of NeRF Mapping and SLAM is gradually becoming a research trend.

### 2.3 SLAM with NeRF Mapping

Traditional visual SLAM systems offer speed and robustness but often lack detailed scene features. Further advancements, such as iNeRF [41], have demonstrated that in well-trained NeRF models, it is possible to infer camera poses. Barf [20] has shown that in multi-view scenarios, we can formulate optimization problems as iterative image alignment problems while simultaneously inferring scene function expressions and camera poses. However, their use of large MLP as map representations has made both network inference and training slow. iMap [36] and Nice-SLAM [43] have shown the potential of combining SLAM with neural implicit mapping. However, they heavily rely on depth images. Orbeez-SLAM [7] and NeRF-SLAM [34] successfully construct NeRF-based SLAM pipelines with monocular images as input. However, Orbeez-SLAM utilizes ORB feature tracking, which proved prone to tracking failures in our tests and is incapable of handling loop closure adjustment problems. NeRF-SLAM, on the other hand, relies on the unscaled covariate depth map estimated by Droid-SLAM. Due to the substantial size of the Droid-SLAM network itself, achieving real-time performance is challenging. In situations where only grayscale image are available, these approaches result in catastrophic reconstruction outcomes.

## 3 Preliminaries

### 3.1 Tracking

#### *3.1.1 Visual*

As Fig. 2 a show, in the process of only-visual tracking, each keyframe can be represented by a set of state vectors,$S = \{T_0 \dots T_i\}, T_i = \{R_i, t_i\}$ and the task of tracking is to estimate the optimal state vectors for each keyframe.Similarly, we consider the position of map points as states$X = \{x_0 \dots x_j\}$. In order to optimize the tracking system, we need to estimate the positions of map points simultaneously. According [4], we can construct cost functions based on the reprojection relationship.

$$r_{vo} = \min_{T_i, x_j} \left( \sum_{i=1}^{k} \sum_{j=1}^{l} \left\| u_i - \pi(R_{iw} x_w^j + t_{iw}) \right\| \right) \tag{1}$$

where $\pi$ represents a mapping that projects points onto the imaging plane.

#### *3.1.2 Visual-Inertial*

As Fig. 2 b show,compared to only-visual SLAM, visual-inertial SLAM requires consideration of a greater number of state variables.Therefore, we expand the state vector of each keyframe accordingly. $S = \{s_0 \dots s_i\}, s_i = \{T_i, v_i, b_{ai}, b_{gi}\}$ ,where $v_i$ represents the velocity of the keyframe, $b_{ai}$represents the accelerometer bias of the keyframe, and $b_{gi}$ represents the gyroscope bias of the keyframe. Following the theory developed in [22]and formulated on manifolds in [12],the pre-integrated IMU increments can be denoted as $\Delta\mathbf{R}_{i,j}, \Delta\mathbf{v}_{i,j}, \Delta\mathbf{p}_{i,j}$.Moreover, based on [9], we can linearize the system to obtain the propagated IMU state after time $\delta t$,and it is propagated from the state at time $t$ (equation 2).

$$\Sigma_{t+\delta t}^{i} = F_{t+\delta t} \Sigma_{t}^{i} {F_{t+\delta t}}^{T} + G_{t+\delta t} \Sigma_{t+\delta t}^{n} {G_{t+\delta t}}^{T} \tag{2}$$

where $\Sigma_{t+\delta t}^{i}$ represents the covariance of the IMU at time $t+\delta t$,$\Sigma_{t}^{i}$ represents the covariance of the IMU at time $t$, and $\Sigma_{t+\delta t}^{n}$ represents the measurement covariance of the IMU at time $t+\delta t$.$F$ and $G$ are the first-order linearized Jacobian matrices of the IMU propagation equation with respect to the previous state variables and noise terms. According to [12], given states $S_i$ and $S_{i+1}$, we can construct the IMU error equation as shown in equation 3.

$$\begin{aligned} r_{vi} &= \left( \mathbf{r}_{\Delta \mathrm{R}_{i,j}}, \mathbf{r}_{\Delta \mathrm{v}_{i,j}}, \mathbf{r}_{\Delta \mathrm{p}_{i,j}} \right) \\ \mathbf{r}_{\Delta \mathrm{R}_{i,j}} &= \mathbf{Log}\left( \Delta\mathbf{R}_{i,j}^{\mathrm{T}} \mathbf{R}_i^{\mathrm{T}} \mathbf{R}_j \right) \\ \mathbf{r}_{\Delta \mathrm{v}_{i,j}} &= \mathbf{R}_i^{\mathrm{T}} \left( \mathbf{v}_j - \mathbf{v}_i - \mathbf{g}\Delta t_{i,j} \right) - \Delta\mathbf{v}_{i,j} \\ \mathbf{r}_{\Delta \mathrm{p}_{i,j}} &= \mathbf{R}_i^{\mathrm{T}} \left( \mathbf{p}_j - \mathbf{p}_i - \mathbf{v}_i \Delta t_{i,j} - \frac{1}{2}\mathbf{g}\Delta t^2 \right) - \Delta\mathbf{p}_{i,j} \end{aligned} \tag{3}$$

Combining visual constraints, the cost function for visual-inertial can be expressed as:

$$\min_{S,X} \left\{ \sum_{i=0}^{k} \rho_{Hub}\left( \|r_{vi}\|_{\Sigma_{vi}^{-1}}^{2} \right) + \sum_{i=0}^{n-1} \rho_{Hub}\left( \|r_{vo}\|_{\Sigma_{vo}^{-1}} \right) \right\} \tag{4}$$

### 3.2 NeRF Mapping

NeRF Mapping is a new representation method for map. Different from the traditional approach of directly projecting point clouds,its core idea is to indirectly reconstruct 3D scenes by fitting the scene density and color functions. The density and color of any point at any position in the scene can be directly calculated through the radiance field $F$ and the viewing direction$[\theta, \phi]$, represented by the formula:

$$F : (x, y, z, \theta, \phi) -> (R, G, B, sigma) \tag{5}$$

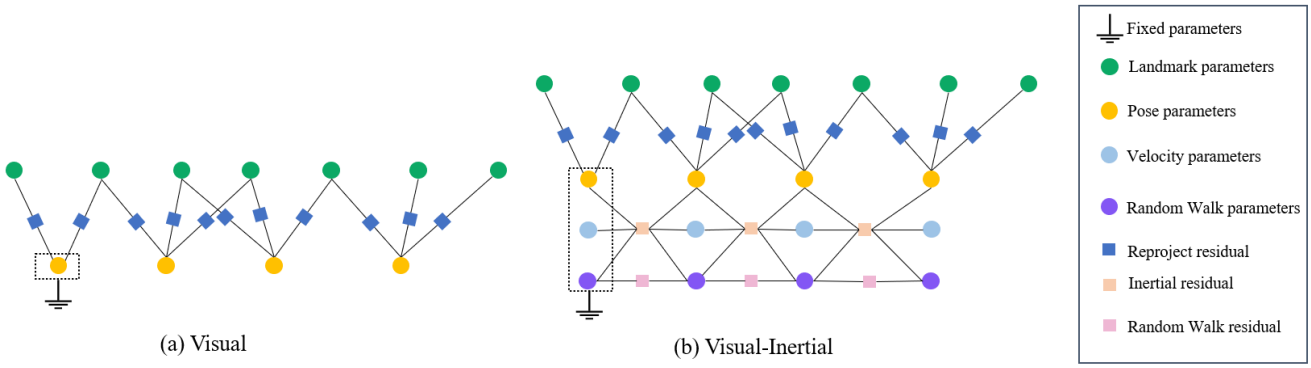


**Fig. 2 Factor Graph** In the Rule-Based SLAM System, factor graph optimization representations are shown along different scenarios. **Figure a**) represents the factor graph optimization representation under the sole visual constraints. **Figure b)** represents the factor graph optimization representation when there is a tight coupling between visual and inertial states.

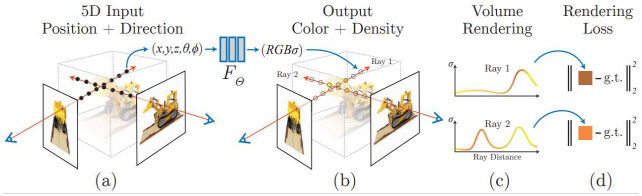


**Fig. 3** The original representation and rendering process of the neural radiance field scene can be summarized as follows. The image synthesis (a) is achieved by sampling 5D coordinates along the camera rays. These positions are then input into a MLP to generate color and volume density (b). These values are synthesized into an image using volume rendering techniques (c). The parameters of the radiance field are optimized (d) by minimizing the residual between the synthesized image and the actually observed image.(Reprinted from [24])

As shown in Fig. 3, sampling is performed along the ray propagation direction. By employing ray tracing and volume rendering(equation 6), one can obtain the views of the NeRF observed from different poses. Finally, the convergence of the neural radiance field is supervised by comparing it with the ground truth views.

$$
\begin{aligned}
I_{pre} &= \sum_{i} \mathcal{T}_i \left(1 - \exp\left(-\sigma_i \delta_i\right)\right) c_i \\
\mathcal{T}_i &= \exp\left(-\sum_{j=1}^{i-1} \sigma_j \delta_j\right)
\end{aligned}
\tag{6}
$$

where $I_{pre}$ is the predicted color of a pixel,$\mathcal{T}_i$ is the transparency of the ray at position,$c_i$ is the color prediction of point,and $\sigma_i$is the occupancy prediction of point.

## 4 Method

The main idea of our paper is to integrate the representation of NeRF map into the rule-based SLAM optimization process to achieve real-time dense reconstruction. We employ deep neural networks for feature extraction and matching to enhance short-term data association. Through the reprojection constraints of feature points, we obtain the camera pose of camera keyframes and the depth values of semi-dense feature points in the map. This provides semi-dense depth supervision for NeRF, allowing anticipation of empty regions and concentrating sampled points more densely on the surfaces of objects. In loop detection, we compute the spatial transformation before and after the loop closure to expedite adjustments to the NeRF map. The NeRF map section adopts a multi-resolution hash encoding architecture to accelerate the convergence of the neural radiance field.To restore the true scale of the entire NeRF reconstructed scene and to achieve faster and more robust estimation of camera states, we integrate inertial data for pose estimation and scale measurement. The detailed description of the entire system is provided in Section 4.1, while Section 4.2 introduces the dynamic compensation process in IMU noise handling. Finally, our optimization details are presented in Section 4.3.

### 4.1 VI-NeRF-SLAM System Pipeline

Our front-end tracking pipeline is improved based on the VINS-Fusion framework. The overall framework is shown in Figure 4. Building upon the foundation of VINS-Fusion [33], we re-adopt deep features as short-term data associations. SuperPoint [11] extracts robust feature points based on global image information, and simultaneously, LightGlue [21] performs adaptive and fast association of feature points according to matched images. This simplifies the entire system's process and accelerates the tracking process. After performing sliding window optimization, we obtain keyframes with more accurate poses and sparse feature points.

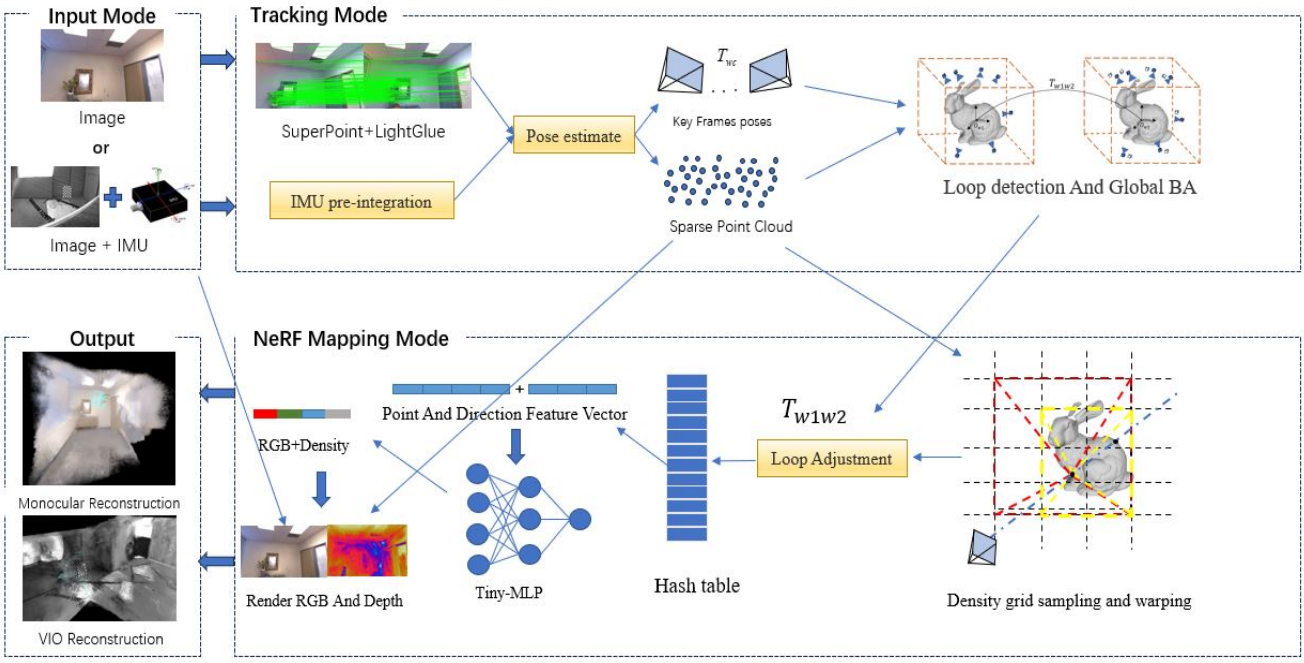


**Fig. 4 System Pipeline.** The overall system pipeline is illustrated in the figure 4. To achieve fast tracking, we have redesigned the conventional SLAM framework by employing deep neural networks for feature extraction and matching, resulting in improved short-term data association. Through the reprojection constraints of feature points, we obtain the camera pose of camera keyframes and the depth values of semi-dense feature points in the map. This provides semi-dense depth supervision for NeRF, allowing anticipation of empty regions and concentrating sampled points more densely on the surfaces of objects. In loop detection, we compute the spatial transformation before and after the loop closure to expedite adjustments to the NeRF map. The NeRF map will adopt a multi-resolution hash encoding architecture [26] to accelerate the convergence of the neural radiance field. 4.1

We have incorporated loop closure detection, which contributes to building a globally consistent NeRF map. In the absence of loop closure and map fusion, we use RGB information and jointly supervise the convergence of some triangulated depth with neural radiance field, assigning more loss weight to pixels with depth information. When loop closure and map fusion occur, as shown in Figure 5, we seek a bijective transformation between the reconstruction centers before and after loop closure optimization. Based on this, we map the sampled points to preserve more pre-optimization information.

As described in Basalt [20], the neural radiance field can simultaneously optimize camera poses and map representation, converging to a small range. Therefore, we ultimately synchronize the optimization of camera states and map features in the NeRF map.

## 4.2 IMU Dynamic Compensation

To improve the robustness of the system, we optimized the covariance terms in the noise propagation process for the Visual-Inertial Odometry. In the noise propagation process of VIO (Equation 7), after a time interval of $\delta t$, the noise covariance $\Sigma^Z_{t+\delta t}$ at time $t+\delta t$ has two sources: the noise covariance propagated from the previous time $\Sigma^Z_t$, and the noise covariance of the current observation $\Sigma^n_t$. These two covariances are fused using a linear Gaussian model.

$$\Sigma^Z_{t+\delta t} = (I + F_t\delta t)\Sigma^Z_t(I + F_t\delta t)^T + (G_t\delta t)\Sigma^n_t(G_t\delta t)^T \tag{7}$$

where $F_t$ is first-order derivative of the state variables with respect to the IMU state propagation equation, and $G_t$ is derivative of the noise with respect to the IMU state propagation equation. [32]

Regarding the covariance of observation noise, most of the existing approach measuring the noise when the inertial unit is stationary and fitting a Gaussian model to obtain the covariance. This covariance is then applied to the observation noise at all discrete time steps. However, in actual motion scenarios, the observation noise model under IMU motion is not always suitable for a Gaussian model, especially when experiencing high-acceleration motion where the observation noise becomes non-Gaussian white noise. Its specific behavior is influenced by the coupling of linear acceleration and angular acceleration. We extend the propagation model for observation noise in the existing approach by appro-

priately enlarging the covariance fitted with Gaussian white noise model at the stationary moment, considering the motion state. We aim to find an appropriate Gaussian model to fit the non-Gaussian white noise model for each discrete observation noise. (Equation 8 and 9)

$$\hat{\Sigma}^{Z}_{t+\delta t} = (I + F_t\delta t)\hat{\Sigma}^{Z}_{t}(I + F_t\delta t)^T + P_t(G_t\delta t)\Sigma^{n}_{t}(G_t\delta t)^T \tag{8}$$

where

$$\begin{aligned} P_t &= \theta^{\|\bar{a}_v - g\|_2 + \|\bar{a}_w t\|_2} \\ \bar{a}_v &= \frac{1}{n_{\delta t}} \sum_{t}^{t+\delta t} a_{vt} \\ \bar{a}_w &= \frac{1}{n_{\delta t}} \sum_{t}^{t+\delta t} a_{wt} \end{aligned} \tag{9}$$

We use the $P_t$ to appropriately inflate the measured covariance, where $a_v$ represents linear acceleration and $a_w$ represents angular acceleration. For different IMUs, we set different base compensation values $\theta$. In a static state, the compensation function value for measuring covariance is 0. When there is a sudden change in motion along the trajectory, the measured covariance is compensated based on the linear and angular accelerations. This compensates for the covariance to increase the measurement uncertainty of the IMU and reduces additional observation noise caused by motion discontinuity. It promotes the continuity and robustness of pose estimation in backend optimization.

### 4.3 Map Loss

As shown in equation 10, our neural implicit mapping loss function consists of three parts: image color loss $\mathcal{L}_{\text{LI}}$, depth loss $\mathcal{L}_{\text{depth}}$, edge-aware disparity map smoothness loss $\mathcal{L}_{\text{smooth}}$. We also incorporate hyperparameters weights to balance their contributions.

$$\mathcal{L}_{\text{loss}} = \lambda_{\text{LI}}\mathcal{L}_{\text{LI}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}} + \lambda_{\text{smooth}}\,\mathcal{L}_{\text{smooth}} \tag{10}$$

We aim to make our neural implicit mapping applicable not only for RGB image scene reconstruction but also for grayscale image scene reconstruction. For RGB images, we combine the color losses from all three channels. For single-channel grayscale images, we expand them to three channels. Additionally, to achieve higher-quality rendering, we map the images to linear space.

$$\mathcal{L}_{\text{I}} = \frac{1}{3HW}\sum_{i}\left|\widehat{I}_i - I_{igt}\right|^2 \tag{11}$$

Due to the semi-dense nature of the obtained point cloud in the pre-processing stage, we only apply depth supervision to pixels with triangulated depth. Additionally, since triangulated depth comes with uncertainty, we relax the depth requirements (formula n) and only compute the loss for depth values that deviate significantly from the triangulated depth multiplied by $\lambda_d$, where $0.1 < \lambda_d < 0.5$. This allows us to constrain the depth estimation of the neural implicit mapping within a small range around the triangulated depth, while accelerating the network's inference process.

$$\begin{aligned} \mathcal{L}_{depth} &= \frac{1}{N}\sum_{i}\left|\widehat{d}_i - {d_i}^{est}\right|^2 \\ \hat{d}_i &\in \left\{\hat{d}_i \mid \left|\hat{d}_i - d_i^{est}\right| > \lambda_d d_i^{est}\right\} \end{aligned} \tag{12}$$

We aim to maintain smoothness in the inverse depth of the image within regions of pixel smoothness to achieve consistent and continuous depth estimation. Therefore, we introduce the edge-aware disparity map smoothness loss [13, 14, 18], which is defined as:

$$\begin{aligned} \mathcal{L}_{\text{smooth}} &= \left|\partial_x \hat{\mathcal{D}}^*\right| \exp^{-|\partial_x \text{I}|} + \left|\partial_y \hat{\mathcal{D}}^*\right| \exp^{-|\partial_y \text{I}|} \\ \hat{\mathcal{D}}^* &= \hat{D}/\overline{D} \end{aligned} \tag{13}$$

where $\hat{D} = 1/\hat{d}_i$, $\partial_x$ and $\partial_y$ are the image gradients, and $\hat{\mathcal{D}}^*$ is the mean-normalized disparity.

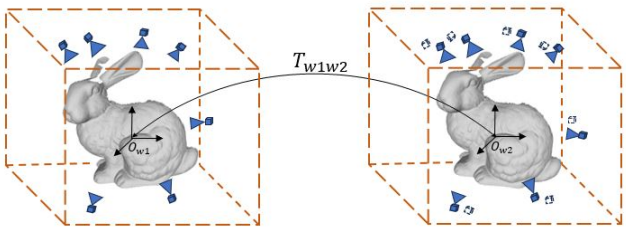


**Fig. 5 Loop Adjustment in NeRF Mapping.** Our NeRF Map consists of a canonical 3D volume and a bijection, where the canonical 3D volume is typically represented by a 3D volume at initialization. When a loop fusion occurs, we search for a bijection matrix that allows the 3D volume after the loop BA to be correctly mapped back to the canonical field, retrieving the previously converged volume density and color values for queries.

## 5 Experimental Results

We evaluated our proposed method and architecture on several datasets and compared our approach with some existing benchmark methods to validate the geometric and photometric accuracy of our method.

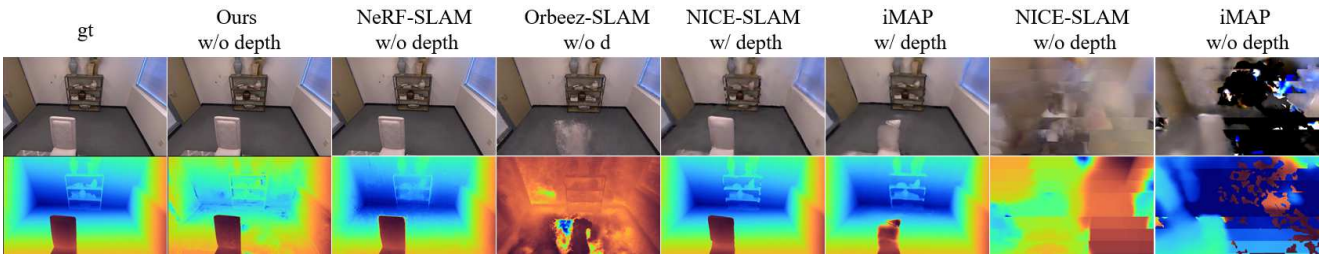


**Fig. 6 Replica Result.**In this work, we evaluate our approach, NeRF-SLAM, Orbeez-SLAM, NICE-SLAM, and iMAP (denoted as "w/o depth"), in the absence of using deep supervision. From the figures, it is evident that NICE-SLAM and iMAP fail to reconstruct the scene when lacking depth supervision. NeRF-SLAM, on the other hand, achieves the best reconstruction results in the absence of depth information by estimating noisy depth maps through supervised NeRF networks. However, as shown in Table 5, NeRF-SLAM requires substantial training resources and time.In contrast, our proposed method combines traditional reprojection techniques with NeRF training, leading to significantly improved training speed while achieving approximate reconstruction accuracy. Additionally, we also estimate NICE-SLAM and iMAP with depth supervision (denoted as "w/ depth") simultaneously.

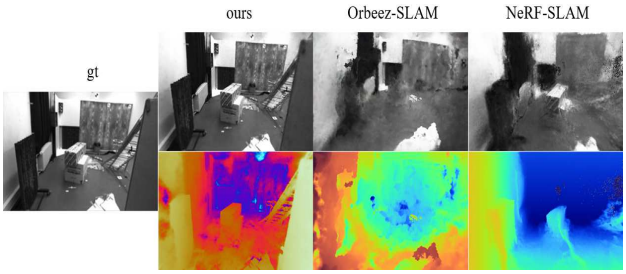


**Fig. 7 Euroc Result.**We evaluated our method on the euroc dataset, comparing it with existing single-view NeRF-based SLAM approaches. The results demonstrate that our method achieves the highest reconstruction accuracy among them. Furthermore, our approach benefits from the incorporation of an inertial measurement unit, allowing us to estimate the scene's scale relative to the real-world scene. During rendering, we apply a fixed-scale scaling based on this estimation. This advancement is instrumental in facilitating the practical application of NeRF-based SLAM in real-world scenarios.

## 5.1 Experimental Setup

### *5.1.1 Datasets*

The Euroc dataset [3] includes indoor and outdoor datasets and consists of rich sensor data, including camera and IMU data. This dataset is widely used in research areas such as robot vision, camera pose estimation, camera calibration, localization, and navigation. Its main features include providing highly accurate sensor data and realistic indoor scenes, which can be used to evaluate the reconstruction accuracy and localization precision of our methods using grayscale image sequences and tightly-coupled IMU measurements. The Replica dataset [35] provides 3D scene reconstruction and virtual visual data for multiple real-world indoor environments. These scenes include residences, offices, schools, commercial places, etc., with diverse structures and layouts, allowing for comprehensive evaluation of our methods' reconstruction accuracy and localization precision on RGB image sequences.

### *5.1.2 Metrics*

We use Absolute Trajectory Error (ATE) **[cm]** to evaluate our tracking accuracy. ATE calculates the Root Mean Square Error (RMSE) between the aligned estimated trajectory and the ground truth trajectory. A smaller ATE value indicates that the trajectory of the SLAM system is closer to the true trajectory, indicating better localization performance. For the accuracy of implicit neural mapping, we generate observed implicit neural scene RGB and Depth along the ground truth trajectory path. We calculate the mean of the L1**[cm]** loss between the generated depth and the ground truth depth to evaluate the geometric accuracy of the scene. Additionally, we calculate the Peak Signal-to-Noise Ratio (PSNR)**[dB]** to evaluate the photometric accuracy of the scene.

### *5.1.3 Baselines*

Since our final optimization results for pose estimation and scene mapping rely more on the unified optimization of the neural radiation field, we consider our SLAM algorithm as a NeRF-Based SLAM. Therefore, in the following text, we will compare our method, VI-NeRF-SLAM, with some existing NeRF-Based SLAM methods: NeRF-SLAM [34], NICE-SLAM [43], iMAP [36], and Orbeez-SLAM [7].

## 5.2 Evaluation

**Implementation Details.** We conducted all experiments on a server equipped with an Intel(R) Xeon(R) Platinum 8369B CPU @ 2.90GHz and an NVIDIA RTX

3090 GPU. For the optimization of reprojection error and IMU error, we used the Ceres optimizer, while the neural implicit mapping was optimized using the tiny-cuda-nn framework.To accommodate different scene scales in the euroc dataset, we set different values for the parameter $\theta$ in the range of $[10, 20]$.To estimate the real-world size of the euroc scenes, we aligned gravity using IMU data. Additionally, for faster rendering, we proportionally scaled and translated the scene before optimizing its representation.Regarding the loop closure adjustment module, we only activated it in large-scale scenes to mitigate pose drift errors after prolonged tracking. Furthermore, we applied small rigid transformations to the voxels in the NeRF map to maximize the reuse of optimization information and avoid retraining.

**Evaluation on Euroc** As shown in Table [3,4,5], we evaluated methods in baselines that do not rely on depth images using the Euroc dataset. The experiments demonstrate that our approach exhibits superior reconstruction capabilities in scenarios with only grayscale images, achieving higher processing speeds while attaining the highest tracking accuracy. Moreover, by incorporating IMU data to align with gravity, we are able to assess the real-world scale of the scene and apply it in practical applications. Table 3 shows our tracking accuracy, while Table 5 demonstrates that our system achieves the fastest training speed, indicating it requires fewer computational resources. We discovered that training NeRF Map relies more on multi-angle observations, so for small-scale motion, we only need to collect images with significant viewpoint changes to reduce noise and improve training speed. For images with minor motion, we focus solely on tracking accuracy to maintain pose output consistency.For the Euroc dataset, we can simultaneously incorporate IMU tightly coupled integration to estimate the real-scale of the scene.

**Evaluation on Replica** As shown in Table 2 1 5, for the Replica dataset, we conducted a comprehensive evaluation of both SOTA and our method, considering tracking and reconstruction results with and without depth supervision simultaneously.The table demonstrates that in the absence of depth images, our approach can improve tracking quality and speed without sacrificing reconstruction accuracy, achieving the optimal overall performance. Scene reconstruction, as depicted in Figure 6, showcases the effectiveness of our approach. In monocular mode, NeRF-SLAM estimates dense optical flow for noisy supervision, yielding the lowest L1 loss for reconstructed depth maps. However, due to its substantial network size, NeRF-SLAM suffers from lower execution efficiency. By integrating deep learning features with traditional rule-based tracking, we achieve the highest precision in both tracking accuracy and PSNR reconstruction accuracy.Combining the information from Table 1,we observe a significant enhancement in overall reconstruction speed and PSNR accuracy with minimal loss in depth precision. Figure 6 demonstrating that our reconstruction results closely align with true depth values. Notably, NICE-SLAM and iMAP experience catastrophic failures in the absence of depth supervision.

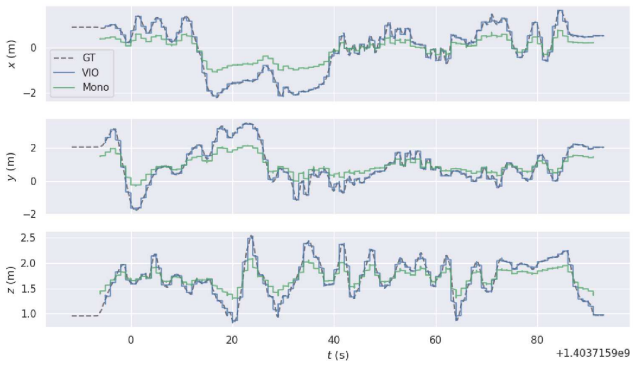


**Fig. 8 Mono and VIO Tracking** We have validated the disparities between monocular and monocular-inertial modes in pose estimation. The monocular-inertial mode demonstrates a closer approximation of the actual camera trajectory, yielding smaller errors in camera pose estimation and exhibiting superior robustness.

## 5.3 Ablation

**Visual-Inertial IMU Fusion.** As shown in Figure 8, if the camera can integrate IMU data for tracking, it will lead to higher accuracy and better robustness. Table [4 3] illustrates the map reconstruction accuracy and tracking accuracy in monocular and VIO mode, indicating the necessity of fusing IMU data for achieving higher NeRF map reconstruction accuracy. In our experiments, we observed that combining NeRF map features with camera poses introduces more errors. While existing NeRF methods can optimize camera poses and map representation simultaneously, their accuracy is still far from traditional rule-based optimization. Therefore, we recommend using rule-based tracking for deriving camera poses, while disabling pose optimization in the NeRF reconstruction phase to achieve higher map reconstruction accuracy.

**IMU Dynamic Error Compensation.** To validate the importance of IMU dynamic error compensation for pose estimation, we conducted experiments on the Euroc dataset. By randomly deleting certain frames between consecutive frames, extending the integration time of the IMU, and adding random noise to simulate

**Table 1** Mapping evaluate for the Replica dataset.We evaluated the geometric accuracy of the scenes using the depth L1 loss and assessed the photometric accuracy using PSNR. The optimal results are indicated in bold.In the absence of depth supervision, NeRF-SLAM achieved the best reconstruction accuracy by estimating depth through dense optical flow. Our method demonstrated reconstruction accuracy close to NeRF-SLAM.When equipped with depth supervision, NICE-SLAM achieved the best reconstruction accuracy among all methods.

| | | room-0 | room-1 | room-2 | office-0 | office-1 | office-2 | office-3 | office-4 | Avg |
|---|---|---|---|---|---|---|---|---|---|---|
| iMAP | Depth(L1) | 6.21 | 5.96 | 7.26 | 10.65 | 11.41 | 8.28 | **7.94** | 6.70 | 8.05 |
| (w/ depth) | PSNR | 5.96 | 5.45 | 6.31 | 8.35 | 9.41 | 10.25 | 9.65 | 6.32 | 7.67 |
| NICE-SLAM | Depth(L1) | **3.61** | **2.23** | **2.31** | **3.22** | **1.85** | **6.31** | 9.45 | **3.56** | **4.06** |
| (w/ depth) | PSNR | **30.14** | **28.56** | **23.56** | **21.35** | **26.43** | **24.20** | **19.65** | **21.99** | **24.47** |
| NICE-SLAM | Depth(L1) | 10.95 | 11.68 | 19.05 | 10.96 | 9.56 | 15.37 | 20.69 | 18.85 | 14.62 |
| (w/o depth) | PSNR | 19.56 | 20.21 | 18.72 | 19.45 | 16.54 | 18.94 | 17.45 | 16.59 | 18.43 |
| Orbeez-SLAM | Depth(L1) | 11.56 | 12.37 | 10.59 | 15.44 | 12.45 | 16.79 | 12.54 | 10.78 | 12.81 |
| (w/o depth) | PSNR | 28.31 | 31.20 | 28.89 | 32.56 | 29.25 | 31.56 | 32.96 | 28.25 | 30.37 |
| NeRF-SLAM | Depth(L1) | **4.45** | **5.03** | **4.47** | **4.52** | **2.29** | **9.87** | **10.53** | **6.90** | **6.13** |
| (w/o depth) | PSNR | 33.40 | 31.26 | 38.45 | 40.30 | **43.41** | 37.47 | 36.96 | 31.76 | 36.52 |
| **ours** | Depth(L1) | 6.37 | 8.98 | 7.21 | 6.32 | 5.38 | 10.64 | **11.77** | 8.30 | 8.12 |
| **(w/o depth)** | PSNR ↑ | **34.01** | **32.57** | **40.12** | **41.20** | 38.57 | **38.63** | **37.92** | **32.30** | **36.92** |

**Table 2** Tracking result for the Replica dataset.Absolute trajectory error evaluation. iMAP and NICE-SLAM initially employed ground truth depth supervision for tracking evaluation. In the absence of depth supervision, iMAP and NICE-SLAM perform worse, and therefore, their tracking accuracy is no longer assessed in the absence of depth supervision.

| | room-0 | room-1 | room-2 | office-0 | office-1 | office-2 | office-3 | office-4 | Avg ↓ |
|---|---|---|---|---|---|---|---|---|---|
| iMAP (w/ depth) | 5.32 | 6.53 | 4.21 | 4.76 | 4.09 | 5.98 | 5.64 | 6.03 | 5.32 |
| NICE-SLAM (w/ depth) | **1.98** | **2.33** | **1.56** | **0.89** | **0.65** | **1.65** | **4.22** | **2.59** | **1.98** |
| Orbeez-SLAM (w/o depth) | 2.16 | 2.65 | 1.54 | 1.32 | 2.96 | 2.03 | 3.96 | 2.98 | 2.45 |
| NeRF-SLAM (w/o depth) | 0.86 | 0.49 | 0.57 | 1.65 | 0.96 | 1.30 | 2.33 | 0.35 | 1.06 |
| **ours(w/o depth)** | **0.066** | **0.015** | **0.046** | **0.098** | **0.095** | **0.050** | **0.103** | **0.025** | **0.062** |

**Table 3 Tracking result** for the Euroc dataset. ATE(cm). We evaluated the tracking results of the indoor grayscale images in the Euroc dataset, with the best result highlighted in bold.

| | v101 | v102 | v103 | v201 | v202 | v203 | **Avg ↓** |
|---|---|---|---|---|---|---|---|
| Orbeez-slam | 0.25 | - | - | 0.32 | - | - | 0.29 |
| NeRF-SLAM | 0.103 | 0.165 | 0.158 | 0.102 | 0.115 | 0.204 | 0.141 |
| **ours (Mono)** | 0.018 | 0.016 | 0.076 | 0.032 | **0.035** | 0.086 | 0.043 |
| **ours (VI)** | **0.013** | **0.014** | **0.066** | **0.02** | 0.042 | **0.067** | **0.037** |

**Table 4 Mapping result** for the Euroc dataset(PSNR).We evaluated the reconstruction results of the indoor grayscale images in the Euroc dataset, with the best result highlighted in bold.

| | v101 | v102 | v103 | v201 | v202 | v203 | **Avg ↑** |
|---|---|---|---|---|---|---|---|
| Orbeez-slam | 20.54 | - | - | 23.01 | - | - | 21.76 |
| NeRF-SLAM | 21.51 | 20.85 | 20.67 | 24.38 | 23.30 | 20.64 | 20.39 |
| **ours (Mono)** | 28.31 | 28.25 | 28.30 | 28.63 | 27.52 | 26.80 | 27.96 |
| **ours (VI)** | **30.31** | **29.33** | **31.30** | **29.63** | **29.52** | **30.80** | **30.14** |

challenging motion conditions, as shown in Table 6, our proposed method in VIO mode effectively mitigates the impact of IMU noise in situations with drastic motion changes, correcting camera poses.

**Table 5** Runtime result.the average **fps** of all baselines and ours

| | EUROC | Replica |
|---|---|---|
| iMAP | | 0.033 |
| NICE-SLAM | | 0.054 |
| Orbeez-SLAM | 20.21 | 20.51 |
| NeRF-SLAM | 10.69 | 10.64 |
| **ours ↑** | **25.95** | **25.72** |

**Table 6** Ablation study on IMU dynamic error compensation(DEC).In the Euroc dataset, we reduced the image frame rate by 1/4 to prolong the integration time of the IMU, simultaneously introducing random noise to simulate challenging motion environments. We conducted an evaluation across the entire Euroc dataset, and in most instances, motion tracking without the DEC module resulted in losses, leading to mapping failures. In motion tracking with the DEC module, some errors introduced by high-noise IMU data were mitigated, making motion estimation more reliant on image information.

| DEC | ATE(cm) ↓ | PSNR(db) ↑ |
|---|---|---|
| ✓ | **6.45** | **15.09** |
| | - | 4.51 |

**Table 7** Ablation study on Loop Closure Correction. We tested the average tracking accuracy under loop closure optimization and non-loop closure optimization conditions. The results indicated that camera poses were more accurate after loop closure adjustment, highlighting the necessity of addressing loop closure issues in NeRF maps. As it is an online optimization process, to prevent the reinitialization of map parameters, we propose fine-tuning the 3D voxel features before and after loop closure to avoid a complete NeRF map reinitialization.

| Loop Closure | ATE ↓ |
|---|---|
| ✓ | **0.037** |
| | 0.164 |

**Loop Closure Correction.** As shown in Table 7, loop closure correction results in more accurate poses, inevitably leading to higher precision in NeRF map reconstruction. Therefore, to avoid a complete reinitialization of the NeRF map after loop closure, we fine-tune voxel features in NeRF, enabling the reuse of previous optimization information for faster convergence.

### 5.4 Limitation

In large-scale grayscale scene reconstruction, our method exhibits more severe artifacts compared to RGB reconstruction. This also indicates that a better approach is required when using only grayscale images for fast and dense reconstruction of large-scale scenes. In the fields of robotics and autonomous driving, the focus is often not on how the entire scene is rendered in RGB, but rather on the occupancy grid of the scene, where grayscale images are sufficient. Therefore, it is necessary to rapidly reconstruct a dense grayscale scene with an accurate occupancy grid. On the other hand, our method is not capable of real-time rendering and reconstruction of dynamic scenes, as well as rendering issues in unbounded scenes. These limitations currently exist in our approach.

## 6 Conclusion and Future Work

We have developed a SLAM system that can perform dense 3D scene reconstruction in real-time online without the need for pre-training, by combining the advantages of non-learning SLAM methods and new view synthesis. We attempted to integrate two different optimization approaches: utilizing feature point reprojection constraints to estimate pixel depth and pose (traditional SLAM methods) and employing ray integration and depth networks for optimization of 3D scene representation (NeRF mapping). The former optimizes quickly and provides more accurate pose estimation, while the latter enables rapid optimization of dense maps.In addition, we proposed a dynamic error compensation method for IMU and a technique to handle loop closures in NeRF map. Experimental results demonstrate that our method achieves the fastest RGB image scene reconstruction speed while maintaining reconstruction accuracy comparable to SOTA methods. Furthermore, in grayscale image reconstruction alone, both our speed and accuracy achieved SOTA-level performance. Additionally, our method can approximate the scale of the scene by incorporating inertial measurements.

Our approach has potential applications in the fields of autonomous driving, home service robotics, and environmental simulation. Moreover, future work could explore faster dynamic scene reconstruction, grayscale image occupancy grid reconstruction based on our method, as well as real-time tracking and dense reconstruction of fast, unbounded and free scenes.

## References


1. Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. ICCV (2021)
2. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. CVPR (2022)
3. Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., Achtelik, M.W., Siegwart, R.: The euroc micro aerial vehicle datasets. The International Journal of Robotics Research (2016). DOI 10.1177/0278364915620033. URL `http://ijr.sagepub.com/content/early/2016/01/21/0278364915620033.abstract`
4. Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M., Tardós, J.D.: Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. IEEE Transactions on Robotics **37**(6), 1874–1890 (2021)
5. Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H.: Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo pp. 14124–14133 (2021)
6. Chen, Z.: Im-net: Learning implicit fields for generative shape modeling (2019)
7. Chung, C.M., Tseng, Y.C., Hsu, Y.C., Shi, X.Q., Hua, Y.H., Yeh, J.F., Chen, W.C., Chen, Y.T., Hsu, W.H.: Orbeez-slam: A real-time monocular visual slam with orb features and nerf-realized mapping. arXiv preprint arXiv:2209.13274 (2022)
8. Clark, R.: Volumetric bundle adjustment for online photorealistic scene capture pp. 6124–6132 (2022)
9. Crassidis, J.L.: Sigma-point kalman filtering for integrated gps and inertial navigation. IEEE Transactions on Aerospace and Electronic Systems **42**(2), 750–756 (2006)
10. Dai, A., Nießner, M., Zollhöfer, M., Izadi, S., Theobalt, C.: Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. ACM Transactions on Graphics (ToG) **36**(4), 1 (2017)

11. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 224–236 (2018)
12. Forster, C., Carlone, L., Dellaert, F., Scaramuzza, D.: On-manifold preintegration for real-time visual–inertial odometry. IEEE Transactions on Robotics **33**(1), 1–21 (2016)
13. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency pp. 270–279 (2017)
14. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J.: Digging into self-supervised monocular depth estimation pp. 3828–3838 (2019)
15. Koestler, L., Yang, N., Zeller, N., Cremers, D.: Tandem: Tracking and dense mapping in real-time using deep multi-view stereo pp. 34–45 (2022)
16. Leutenegger, S., Furgale, P., Rabaud, V., Chli, M., Konolige, K., Siegwart, R.: Keyframe-based visual-inertial slam using nonlinear optimization. Proceedings of Robotis Science and Systems (RSS) 2013 (2013)
17. Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., Furgale, P.: Keyframe-based visual–inertial odometry using nonlinear optimization. The International Journal of Robotics Research **34**(3), 314–334 (2015)
18. Li, J., Feng, Z., She, Q., Ding, H., Wang, C., Lee, G.H.: Mine: Towards continuous depth mpi with nerf for novel view synthesis pp. 12578–12588 (2021)
19. Li, M., Mourikis, A.I.: High-precision, consistent ekf-based visual-inertial odometry. The International Journal of Robotics Research **32**(6), 690–711 (2013)
20. Lin, C.H., Ma, W.C., Torralba, A., Lucey, S.: Barf: Bundle-adjusting neural radiance fields (2021)
21. Lindenberger, P., Sarlin, P.E., Pollefeys, M.: Lightglue: Local feature matching at light speed. arXiv preprint arXiv:2306.13643 (2023)
22. Lupton, T., Sukkarieh, S.: Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions. IEEE Transactions on Robotics p. 61–76 (2011). DOI 10.1109/tro.2011.2170332. URL `http://dx.doi.org/10.1109/tro.2011.2170332`
23. Martin-Brualla, R., Radwan, N., Sajjadi, M.S.M., Barron, J.T., Dosovitskiy, A., Duckworth, D.: NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections (2021)
24. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
25. Mourikis, A.I., Roumeliotis, S.I.: A multi-state constraint kalman filter for vision-aided inertial navigation pp. 3565–3572 (2007)
26. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Transactions on Graphics (ToG) **41**(4), 1–15 (2022)
27. Ortiz, J., Clegg, A., Dong, J., Sucar, E., Novotny, D., Zollhoefer, M., Mukadam, M.: isdf: Real-time neural signed distance fields for robot perception. arXiv preprint arXiv:2204.02296 (2022)
28. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation pp. 165–174 (2019)
29. Paul, M.K., Roumeliotis, S.I.: Alternating-stereo vins: Observability analysis and performance evaluation pp. 4729–4737 (2018)
30. Paul, M.K., Wu, K., Hesch, J.A., Nerurkar, E.D., Roumeliotis, S.I.: A comparative analysis of tightly-coupled monocular, binocular, and stereo vins pp. 165–172 (2017)
31. Prisacariu, V.A., Kähler, O., Golodetz, S., Sapienza, M., Cavallari, T., Torr, P.H., Murray, D.W.: Infinitam v3: A framework for large-scale 3d reconstruction with loop closure. arXiv preprint arXiv:1708.00783 (2017)
32. Qin, T., Li, P., Shen, S.: Vins-mono: A robust and versatile monocular visual-inertial state estimator. IEEE Transactions on Robotics **34**(4), 1004–1020 (2018)
33. Qin, T., Pan, J., Cao, S., Shen, S.: A general optimization-based framework for local odometry estimation with multiple sensors. arXiv preprint arXiv:1901.03638 (2019)
34. Rosinol, A., Leonard, J.J., Carlone, L.: Nerf-slam: Real-time dense monocular slam with neural radiance fields. arXiv preprint arXiv:2210.13641 (2022)
35. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., et al.: The replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
36. Sucar, E., Liu, S., Ortiz, J., Davison, A.J.: imap: Implicit mapping and positioning in real-time pp. 6229–6238 (2021)
37. Sun, C., Sun, M., Chen, H.T.: Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. CVPR (2022)
38. Teed, Z., Deng, J.: Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. Advances in neural information processing systems **34**, 16558–16569 (2021)
39. Wang, P., Liu, Y., Chen, Z., Liu, L., Liu, Z., Komura, T., Theobalt, C., Wang, W.: F2-nerf: Fast neural radiance field training with free camera trajectories. arXiv preprint arXiv:2303.15951 (2023)
40. Whelan, T., Leutenegger, S., Salas-Moreno, R., Glocker, B., Davison, A.: Elasticfusion: Dense slam without a pose graph (2015)
41. Yen-Chen, L., Florence, P., Barron, J.T., Rodriguez, A., Isola, P., Lin, T.Y.: inerf: Inverting neural radiance fields for pose estimation. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1323–1330. IEEE (2021)
42. Yu, A., Fridovich-Keil, S., Tancik, M., Chen, Q., Recht, B., Kanazawa, A.: Plenoxels: Radiance fields without neural networks. arXiv preprint arXiv:2112.05131 (2021)
43. Zhu, Z., Peng, S., Larsson, V., Xu, W., Bao, H., Cui, Z., Oswald, M.R., Pollefeys, M.: Nice-slam: Neural implicit scalable encoding for slam pp. 12786–12796 (2022)