Pacific Graphics 2024
R. Chen, T. Ritschel, and E. Whiting
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 43 (2024), Number 7

GauLoc: 3D Gaussian Splatting-based Camera Relocalization

Zhe Xin[1], Chengkai Dai[2], Ying Li[3] and Chenming Wu[†4]

[1]Institute of Automation, Chinese Academy of Sciences, China
[2]College of Mechanical Engineering and Automation, Huaqiao University, China
[3]School of Mechanical Engineering, Beijing Institute of Technology, China [4]Baidu Inc.

Abstract

3D Gaussian Splatting (3DGS) has emerged as a promising representation for scene reconstruction and novel view synthesis for its explicit representation and real-time capabilities. This technique thus holds immense potential for use in mapping applications. Consequently, there is a growing need for an efficient and effective camera relocalization method to complement the advantages of 3DGS. This paper presents a camera relocalization method, namely GauLoc, in a scene represented by 3DGS. Unlike previous methods that rely on pose regression or photometric alignment, our proposed method leverages the differential rendering capability provided by 3DGS. The key insight of our work is the proposed implicit featuremetric alignment, which effectively optimizes the alignment between rendered keyframes and the query frames, and leverages the epipolar geometry to facilitate the convergence of camera poses conditioned explicit 3DGS representation. The proposed method significantly improves the relocalization accuracy even in complex scenarios with large initial camera rotation and translation deviations. Extensive experiments validate the effectiveness of our proposed method, showcasing its potential to be applied in many real-world applications. Source code will be released at https://github.com/xinzhe11/GauLoc.

CCS Concepts

• *Computing methodologies* → *Image-based rendering; Computer vision;*

1. Introduction

Camera relocalization is a crucial task in 3D vision and embodied intelligence, involving estimating camera poses in known environments (*i.e.,* maps). This capability is fundamental for developing camera-based positioning systems used in various areas such as autonomous driving, robotics, and augmented reality [NB17, SHR*15, SLK17, WWD*20, XDW*22, GCL*24]. However, conventional map-based navigation approaches often adopt simple 3D scene representations, such as point clouds [SLK17, GB21], voxels [SPGS18, YK22], or meshes [PKS22, LYC*23], which tend to over-discretize the original 3D scenes, leading to a mismatch between the digital map and the real-world environment.

Neural Radiance Field (NeRF) [MST*21] has revolutionized the representation of scenes by offering a continuous and implicit field. By employing volumetric rendering techniques, NeRF effectively bridges the gap between digital maps and real-world scenes, which has garnered significant interest, both in terms of advancing its efficiency and exploring its potential for replacing traditional map representations. Despite the impressive performance demonstrated by NeRF and its variants on real-world data, achieving real-time efficiency often requires specialized hardware or customized programs. To overcome this, 3D Gaussian Splatting (3DGS) [KKLD23] provides an explicit representation with efficient training and real-time rendering capabilities, making it accessible for deployment on a variety of devices ranging from mobile phones to high-performance machines. This accessibility positions 3DGS as a promising map representation for robotics and 3D graphics applications, particularly in the realm of 3D visual navigation systems. With its potential to provide efficient and accessible 3D rendering, 3DGS is poised to gain popularity and significantly advance computer vision, graphics, and embodied intelligence. Specifically, by employing 3DGS as the map representation, the RGB, depth, and semantic images can be rendered on the fly using a given camera pose. This eliminates the need to store and cache a large number of precomputed images in the map, while point feature map representation requires retaining keyframes and depth maps within the map. Leveraging 3DGS for map representation significantly reduces storage requirements, particularly for localization tasks in large-scale environments.

Previous studies have explored relocalization within neural scenes, such as NeRF-Loc [LNLW23] and iNeRF [YCFB*21], *etc*. These works leverage the advantages of implicit field representation, often utilizing photometric loss that can be seamlessly propagated to the input camera poses. In contrast, the explicit, discrete representation in 3D space employed by 3DGS poses challenges

[†] Corresponding author.

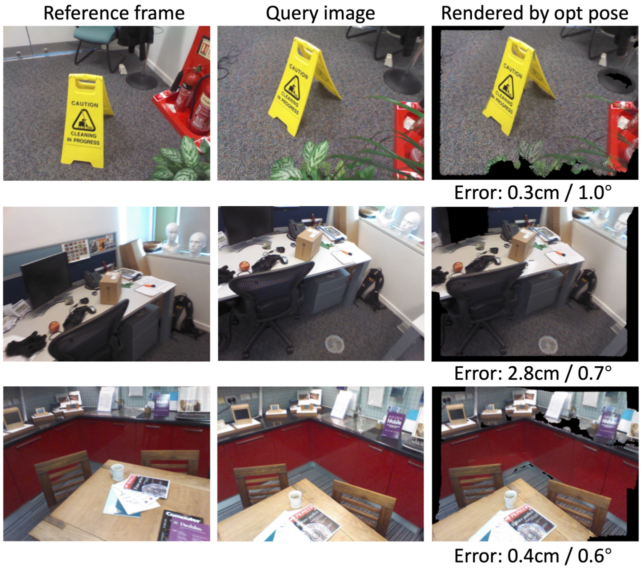


Figure 1: *Our proposed GauLoc leverages a coarse localization and employs a combination of neural and featuremetric optimization techniques to estimate the pose of query images. The errors in translation and rotation are showcased below the rendered images, corresponding to the optimized poses.*

in terms of optimization and stability. Additionally, the rendering process for each guess frame in implicit fields is time-consuming, whereas 3DGS excels in real-time rendering, enabling more efficient camera relocalization. To address these considerations, we propose GauLoc, drawing inspiration from previous NeRF-based approaches. Our main goal is to design an efficient camera relocalization algorithm to estimate the SE(3) pose within a map represented by 3DGS. Our proposed GauLoc adopts a two-step approach, starting with place recognition to efficiently focus on poses close to the target optimal pose. We then introduce an implicit alignment scheme that incorporates point and feature matching from image features to alleviate the difficulties in pose optimization. Furthermore, we leverage ideas from epipolar geometry, *i.e.,* Perspective-n-Point (PnP) [FB81a], to facilitate the convergence of the explicit 3D Gaussian map. Our emphasis is on designing a method well-suited for real-world applications, which require fast computation, and robustness to significant viewpoint changes. As a result, our proposed GauLoc achieves robust camera relocalization even in the case of poor initial poses and varied environmental changes, such as motion blur, reflective surfaces, repeating structures, and textureless areas. The contributions of this paper can be summarized as follows.

- We propose a novel camera relocalization method specifically designed 3DGS scenes, outperforming existing methods, such as scene coordinate regression and NeRF-based methods, in terms of efficiency and precision.
- We introduce an implicit featuremetric alignment meachism, which applies pixel- and region-wise warping to align the features for camera relocalization, providing a continuous space for easier optimization.
- We conduct extensive experiments on two widely used datasets for benchmarking camera relocalization. Experimental results validate the effectiveness and efficiency of our proposed method.

2. Related Work

2.1. Pose Optimization within Scenes

Our research work focuses on pose optimization within scenes. The pioneering works involve Structure-from-Motion (SfM) [SF16, SSS06] and dense visual SLAM [MAMT15, QLS18]. With the rapid development of deep learning and neural representation, the body of literature in this area is getting expanded, with notable contributions such as [WWX*21] proposes a joint optimization approach for camera poses and scene representations. [LMTL21] integrates bundle adjustment techniques for more accurate 3D reconstructions. [JAC*21] introduces a self-calibration mechanism for pose optimization. [BWL*23] presents an innovative approach for pose estimation using neural implicit fields. These studies have enriched the field, advancing accuracy and flexibility in pose estimation and scene reconstruction. In the context of large-scale scene reconstruction, LocalRF [MLG*23] introduces a progressive optimization strategy to improve view synthesis robustness, which entails a significant time investment in the optimization process. Neural SLAM approaches such as [SLOD21, WWA23, ZPL*22, ZPL*24, LGY*23, XYZW24, JCF23] demonstrate the effectiveness of optimizing poses in neural implicit scenes. More recently, 3DGS has revolutionized the field of neural 3D reconstruction and mapping with its real-time rendering performance and hardware compatibility. A great portion of works appear such as Photo-SLAM [HLCY24], SplaTAM [KKJ*24], GS-SLAM [MMKD24], Colmap-free 3DGS [FLK*24], and GGRt [LGZ*24], which consider a time-continuous image sequence and uses local 3DGS to estimate the relatively small pose transformation between adjacent frames, our method addresses the relocalization problem. In our approach, the reference and query images are not sequentially adjacent, and a prior relative pose cannot be obtained. Building upon the success of these existing approaches, our work presents a novel solution for RGB-D camera relocalization specifically designed for indoor scenes by introducing a feature warping loss, which helps mitigate errors caused by non-repetitive feature point extraction that cannot be resolved by reprojection and keypoint warping loss.

2.2. Camera Relocalization from Images

There exist various localization approaches relying on local features such as [SDMR20] and commonly use Structure-from-Motion (SfM) point clouds to represent the scene. The query pose is estimated by matching features between the query images and the 3D points in the scene model, utilizing a minimal solver [PN18] within a RANSAC scheme [FB81b, BNIM20]. Hierarchical localization approaches [IZSB09, SCSD19, HCG*20] enhance scalability by employing intermediate image retrieval [GARL17, RARdS19] to focus on smaller parts of the scene for 2D-3D matching. Neural networks can be trained to regress the camera pose of a query image, either through direct pose regressor [CLWP22, MPT*22], scene coordinate regression [WWD*20, LWZ*20, XDW*22, GCL*24, BR21, BR18, BR21], and scene-agnostic estimation [YBT*19, TTH*21]. The recent progress in

NeRF and 3DGS is transforming map representation and opening up opportunities for improved camera relocalization [GDP*22, LNLW23]. iNeRF [YCFB*21] estimates camera translation and rotation with respect to a 3D object or scene using NeRF while lengthy optimization iteration is required. To improve efficiency, Loc-NeRF [MAS*23] introduces parallel Monte-Carlo sampling using particle filtering. CROSSFIRE [MPB*23] aligns dense local features obtained from volume rendering of implicit fields for camera relocalization. In contrast, our proposed method uses an explicit 3DGS representation and aligns features by warping between frames. A recent work [SWZ*23] also employs a 3DGS representation with keypoint matching for camera relocalization but focuses on object-level scenes by explicitly imposing losses on the matched keypoints.

3. Proposed Method

3.1. Overview

As shown in Fig. 2, our proposed method uses a 3D Gaussian representation of the spatial map, capturing both the scene's geometry and visual appearance. The map consists of two parts, a GS map and a database, where the database stores the global features and poses of all keyframes. Following the relocalization paragdiam [XCL*19], our method consists of a place recognition module and a pose estimation module. The place recognition module identifies the most similar keyframes in the map, providing an initial pose through feature matching and PnP, while the pose estimation module refines the pose iteratively. We optimize the SE(3) pose by employing an end-to-end optimizer that minimizes the loss incurred by rendering each pixel's color based on inferred camera parameters. In addition, we introduce an implicit featuremetric alignment scheme to enhance geometric and semantic consistency between the rendered frame and the query frame, enabling accurate localization even in complex scenarios. The geometric consistency leverages pixel disparities among matched key points to facilitate rapid convergence toward the correct position. The semantic consistency leverages keypoint features to mitigate interference caused by mismatches.

3.2. 3DGS Map Representation

As map-based navigation systems need maps that are efficient not only in storage but also in processing internal data. This requirement makes NeRF [MST*21] less suitable; however, 3DGS [KKLD23] is an ideal technique. 3DGS proposes to use explicit 3D Gaussian as its primary components to represent a scene. Each 3D Gaussian is represented by a 3D point that possesses Gaussian attributes, and can be denoted by the mathematical function $e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\intercal}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}$, where $\boldsymbol{x}$ represents the 3D coordinates of the point, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ represent the spatial mean and covariance matrix. Each Gaussian point is also associated with an opacity η, scale s, and a view-dependent color $\boldsymbol{c}$ represented by spherical harmonic coefficients $\boldsymbol{f}$. During the forward process from a specific viewpoint, the 3D Gaussians undergo projection onto the view plane through splitting. The pixel color is produced by alpha-blending a sequential stacking of N Gaussians:

$$C = \sum_{i\in N} T_i \alpha_i \boldsymbol{c}_i \quad \text{with} \quad T_i = \prod_{j=1}^{i}(1-\alpha_j). \tag{1}$$

We refer to the detailed splatting computations to [KKLD23]. In short, the opacity factor α is computed by multiplying η with the contribution of the 2D covariance, calculated from $\boldsymbol{\Sigma}'$ and the pixel coordinate in image space. The covariance matrix $\boldsymbol{\Sigma}$ is parameterized using a unit quaternion $\boldsymbol{q}$ and a 3D scaling vector $\boldsymbol{s}$. Simply put, within the 3DGS framework, a 3D map $\mathbf{G}$ is represented by a collection of 3D Gaussians, i.e., $\mathbf{G} = \{P_i\}$, where each Gaussian P_i is parameterized as $(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, \boldsymbol{q}_i, \boldsymbol{s}_i, \eta_i, \boldsymbol{f}_i)$.

3.3. SE(3) Pose Optimization

Consider a 3DGS map $\mathbf{G}$ constructed by a set of keyframe features $\mathbf{K}$, and a given query image $\mathbf{I}_q$, the problem we tackle is to efficiently and accurately obtain the SE(3) pose P of $\mathbf{I}_q$ within the map $\mathbf{G}$. Directly optimizing SE(3) pose within $\mathbf{G}$ from scratch is challenging. Thus, we follow a coarse-to-fine strategy to decompose the relocalization problem using a place recognition module and a pose estimation module. In particular, we use cosine distance to search a reference keyframe $\mathbf{I}_\mathrm{r}$ in the keyframe set $\mathbf{K}$ that exhibits the largest similarity with the query image. This can be achieved by minimizing the following function:

$$\mathbf{I}_\mathrm{r} = \arg\min_{k\in\mathbf{K}} ||\mathbf{GF}_q - \mathbf{GF}_k||^2 \tag{2}$$

where $\mathbf{GF}$ is the global feature of this image. The pose of $\mathbf{I}_\mathrm{r}$ is employed to render RGB and depth images from the 3DGS map to calculate the prior pose of $\mathbf{I}_q$ through feature matching for the later fine-level pose optimization. Our proposed iterative optimization method incorporates the geometric coordinates and pixel disparities between the rendered image and query image to measure the error in the image space. This can be flowed to the pose objective by gradient propagation. However, in our practice, we observe that directly propagating the gradient to the pose can lead to unstable optimization. This may be attributed to the explicit representation employed by 3DGS. To tackle this problem, we use pixel-wise rendering loss. Concretely, we render the RGB frame $\widehat{\mathbf{I}}_\mathrm{q}$ given the iterative updated pose T' for computing the pixel-wise rendering loss functions, which are formulated as following L_1 and L_{SSIM} errors. We incorporate both SSIM and L1 loss, as SSIM loss excels in preserving structural information and perceptual quality, leading to improved performance in image processing and generation tasks. It is worth noting that Eq. 3 is identical to the rendering loss used in map reconstruction.

$$\mathcal{L}^{\mathrm{r}}_{\mathrm{RGB}} = \sum_i (||\widehat{\mathbf{I}}_\mathrm{q} - \mathbf{I}_q||_1 + ||\widehat{\mathbf{I}}_\mathrm{q} - \mathbf{I}_q||_{SSIM}), \tag{3}$$

3.4. GS-based Featuremetric Alignment

To further enhance the accuracy and efficiency of pose optimization in the context of 3DGS, we propose an implicit feature alignment scheme by explicitly supervising the spatial relationship between the reference and query frames. Only employing rendering

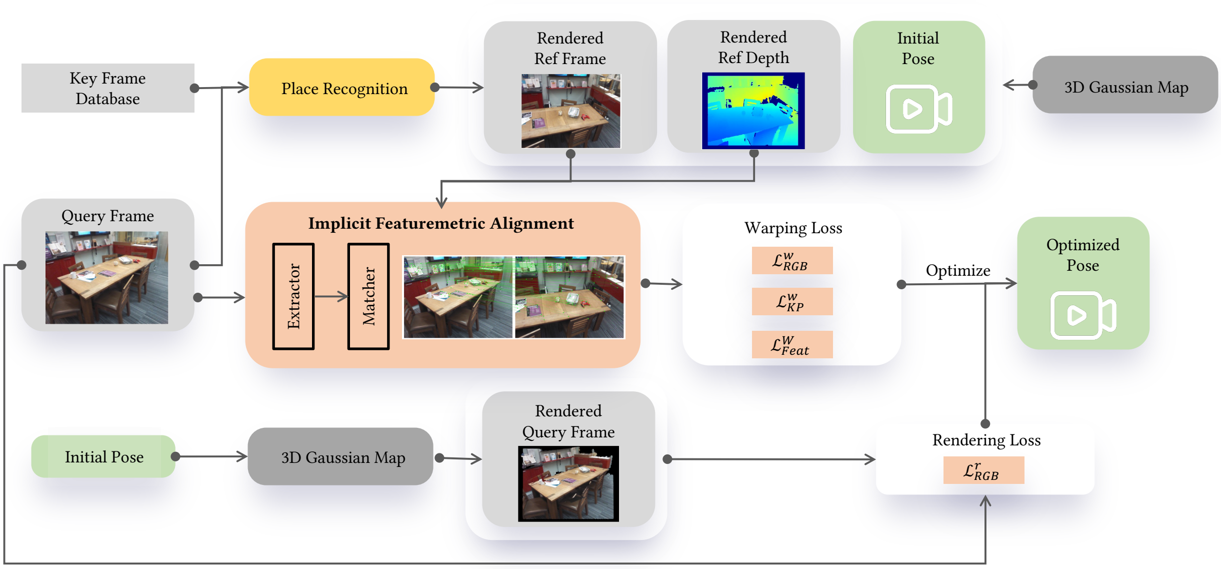


Figure 2: *Our proposed GauLoc begins with place recognition, retrieving the most similar keyframe from the map. At the position of the keyframe, RGB and depth images are rendered by 3DGS map representation. After extracting and matching local features between the query image and keyframe, 3D-2D correspondences can be established according to 2D-2D matches and the rendered depth map. Then, the initial camera pose is acquired through a basic PnP solver with Ransac. Then, GauLoc employs an iterative optimization approach to estimate the pose of query images. This estimation is achieved by combining explicit rendering and implicit featuremetric alignment optimization.*

loss to maintain the consistency of appearance similarity can lead to failures where there is a significant pose difference between the frames, resulting in suboptimal optimization and convergence towards local minima. Our implicit feature alignment, inspired by previous works [ZDJF14, DRMS07, WSSZ23], introduces explicit supervision through warping to enhance relocalization accuracy. Specifically, we employ SuperPoint [DMR18], a deep learning-based method designed for joint detection and description of interest points, to extract local descriptors **LF** from each frame, followed by using LightGlue [LSP23] for feature map matching, which integrates context through self- and cross-attention units with positional encoding. This approach allows for introspection of the feature maps and prediction of correspondences S_{ij} between frames based on pairwise similarity and unary matchability.

We utilize the Structural Similarity Index (SSIM) with 3 × 3 patches, enabling the computation of the RGB warping loss. Based on the 2D-2D correspondences established in the place recognition module and the rendered depth image of the reference keyframe, given an inlier keypoint i in the rendered reference keyframe as $\widehat{\mathbf{I}}_{ri}$, its matched keypoint in the query image is $\mathbf{I}_{qj}$. The warping point of $\widehat{\mathbf{I}}_{ri}$ in the query image, named $\mathbf{I}_{qi'}$, can be calculated as,

$$\mathbf{I}_{qi'} = \Pi(\mathbf{D}_{ri}\mathbf{R}^q_r\Pi^{-1}(\widehat{\mathbf{I}}_{ri}) + \mathbf{t}^q_r). \tag{4}$$

where $\widehat{\mathbf{D}}_{ri}$ is the rendered depth of $\widehat{\mathbf{I}}_{ri}$, Π is the projection matrix of intrinsic camera parameter and $\mathbf{R}^q_r$ and $\mathbf{t}^q_r$ are the relative rotation and translation from the reference keyframe to the query image.

By croping a patch $\mathbf{P_{qi'}}$ centered at $\mathbf{I}_{qi'}$ and a patch $\mathbf{P_{qj}}$ centered at $\mathbf{I}_{qj}$, we obtain the corresponding warped patch. To exclude the warping of invisible patches, we utilize the visibility mask M from a previous study [DBD*22]. The warping RGB loss is defined by Eq. 5.

$$\mathcal{L}^w_{\mathrm{RGB}} = \frac{\sum_{(i,j)\in S_{ij}} M_i \cdot \mathrm{SSIM}(\mathbf{P_{qi'}}, \mathbf{P_{qj}})}{\sum_i M_i}. \tag{5}$$

We further incorporate featuremetric descriptors in image space, providing additional supervision and guidance for enhanced robustness of the optimization. The feature points and feature maps are supervised via pixel-wise loss functions (Eq. 6), where $\mathcal{L}^w_{\mathrm{FP}}$ measures the pixel distance and $\mathcal{L}^w_{\mathrm{FM}}$ measures the feature distance.

$$\begin{aligned} \mathcal{L}^w_{\mathrm{FP}} &= \frac{\sum_{(i,j)\in S_{ij}} M_i ||\mathbf{I}_{qi'} - \mathbf{I}_{qj}||_2}{\sum_{ij} M_i}. \\ \mathcal{L}^w_{\mathrm{FM}} &= \frac{\sum_{(i,j)\in S_{ij}} M_i \cdot ||\mathbf{LF}_j(\mathbf{I}_{qi'}) - \mathbf{LF}_j(\mathbf{I}_{qj})||_2}{\sum_{ij} M_i}. \end{aligned} \tag{6}$$

Overall Loss. In our relocalization framework, we put all the loss terms together to form the overall loss for the optimization, α and β are utilized to balance the ratios of different losses. Because the rendering loss is for all pixels of the whole image, and the warping loss is only for the feature points, β is used to maintain the proportion of the two losses and set as a constant value. α is calculated as the square of pixel errors between pairs of feature points. When the prior pose is inaccurate, in the early stages of iteration, α is large, and the warping loss dominates the optimization, leveraging the rich scene texture to converge the pose near the global optimum rapidly. Then, the rendering loss plays a major role in optimizing the pose more finely and reducing the impact of mismatches.

$$\mathcal{L} = \frac{1}{\alpha\beta}\mathcal{L}^{r}_{\mathrm{RGB}} + \alpha(\mathcal{L}^{w}_{\mathrm{RGB}} + \mathcal{L}^{w}_{\mathrm{FP}} + \mathcal{L}^{w}_{\mathrm{FM}}). \tag{7}$$

4. Experiments

4.1. Experimental Setup

Datasets. We utilize the widely-used visual relocalization datasets 7-Scenes [SGZ*13] and 12-Scenes [VDN*16], both were captured using a hand-held camera with a structure light sensor scanning depth images, providing ground truth poses for each image. Although the scenes are static with only minor illumination changes, they still present challenging conditions including motion blur, reflective surfaces, repeating patterns, and textureless areas. In 12-Scenes, both color and depth images are well registered. For 7-Scenes, we adopt [BR21] to register the color and depth images through calibration. Each scene consists of multiple sequences, and training and testing sequences have already been separated by dataset authors.

Evaluation Metrics. We evaluate the precision of relocalization using median translation (cm) and rotation (°) errors. Additionally, we employ the accuracy of relocalization, where the correct relocalization is deemed if the rotation error is below 5° and the translation error is below 5cm.

Baselines. To validate the effectiveness of our method comprehensively, we benchmark it against various representative localization techniques. Absolute pose regression networks such as DFNet [CLWP22] and LENS [MPT*22] are included. Scene coordinate regression methods like HACNet [LWZ*20], DSAC++ [BR18], and DSAC* [BR21] are also considered. Additionally, scene-agnostic estimation methods including SANet [YBT*19] and DSM [TTH*21] are part of the comparison. For high-level feature-based pipelines, PixLoc [SUL*21] and InLoc [TOS*18] are included. Furthermore, we compare our method with learning-based approaches utilizing implicit map representations like FQN [GDP*22], NeRFLoc [LNLW23], and CROSSFIRE [MPB*23]. However, we do not evaluate our method against iNeRF [YCFB*21] and related methods such as iComMA [SWZ*23] as they primarily focus on object-level datasets.

Implementation Details. We use the Adam optimizer [KB14] for both camera pose and Gaussian parameter optimization. For each scene in both 7-Scenes and 12-Scenes datasets, the Gaussian models are pre-trained using all available training sequences with ground truth poses. Depth images are used for training to recover better geometry structures, but not used in relocalization, since depth sensors are not available in most application scenarios. To remove redundant information and ensure a balanced map representation, we selected one image as a keyframe for every 20 training images and extracted global features for each keyframe using [XCL*19] in all experiments. These global features are then utilized for place recognition during the relocalization phase. The map of each scene consists of two parts, a Gaussian model and a database, the database stores the global features and poses of all keyframes. There's no need to store any images in the map, we can effortlessly render RGB and depth images from the Gaussian model by providing a pose. Utilizing 3DGS for map representation effectively reduces storage requirements, especially for localization in large-scale environments. As for relocalization, the top three keyframes are retrieved from the database by cosine distance through place recognition. At most 128 SuperPoint [DMR18] keypoints are extracted per image. After matching the query image with all three keyframes using LightGlue [LSP23], the one with the most inliers is set as the reference image and used to calculate the prior pose of the query image by PnP(Perspective-n-Point). Then, the pose is fine-tuned by integrating rendering and warping losses iteratively, the iteration number is 50, and β is 10 for all experiments.

4.2. Comparison with Baseline Methods

Tab. 1 shows the comprehensive results for all seven scenes in the 7-Scenes dataset. Our method demonstrates superior localization accuracy, achieving the best overall performance. Unlike methods such as LENS [MPT*22] and DSM [TTH*21], which solely rely on implicit map representations, our 3DGS map enhances scene generalization. In comparison to local feature-based methods like InLoc [TOS*18] and PixLoc [SUL*21], which often struggle with repetitive and mismatched feature points, our method integrates rendering and warping losses, enabling rapid convergence of the camera pose. Furthermore, methods like NeRFLoc [LNLW23] and CROSSFIRE [MPB*23] establish 3D-2D correspondences using NeRF and employ a basic PnP solver with RANSAC, the interference from mismatches affects localization accuracy. Our approach optimizes the camera pose in an end-to-end manner and incorporates feature-metric losses and rendering loss to mitigate the impact of mismatching.

Additionally, Tab. 2 presents the accuracy of relocalization on the 12-Scenes dataset, our method achieves the highest success rate, surpassing both traditional and deep learning-based approaches. Detailed results of each scene are shown in Tab. 3. SuperPoint+PnP indicates the prior pose accuracy, after iteratively fine-tuning the pose employing rendering and warping losses, the final results have been significantly improved.

4.3. Ablation Study

We conduct ablation studies to justify the individual components of the proposed method by testing different combinations of losses and the effectiveness of the prior pose. Losses include RGB L1 loss, SSIM loss, patch-wise RGB warping loss, keypoint warping loss, and pixel-wise feature warping loss. The prior pose includes using PnP results or the pose of best matched keyframe. Tab. 5 presents the average median translation (cm) and rotation (°) errors across all scenes in 7-Scenes [SGZ*13]. When using the prior pose from PnP, the result of only using rendering loss is better than only using warping loss. This is due to the interference from the keypoint repeatability and mismatches, which cannot be eliminated during pose optimization. Incorporating feature warping supervi-

Table 1: *Evaluation results for the 7-Scenes indoor dataset. Median translation (cm) and rotation (°) errors are reported for each scene, and Acc. is the average accuracy of all scenes. The results of other methods are from [LNLW23] and [MPB*23], which have the precision at the centimeter level. We present the results in terms of millimeters.*

Method	Errors(cm/°) ↓								Acc.(%) ↑
	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs	**Avg.**	
SANet [YBT*19]	3.0/0.9	3.0/1.1	2.0/1.5	3.0/1.0	5.0/1.3	4.0/1.4	16.0/4.6	5.0/1.7	68.2
DFNet [CLWP22]	5.0/1.9	17.0/6.5	6.0/3.6	8.0/2.5	10.0/2.8	22.0/5.5	16.0/3.3	12.0/3.7	-
LENS [MPT*22]	3.0/1.3	10.0/3.7	7.0/5.8	7.0/1.9	8.0/2.2	9.0/2.2	14.0/3.6	8.0/3.0	-
FQN-MN [GDP*22]	4.0/1.3	10.0/3.0	4.0/2.4	10.0/3.0	9.0/2.4	16.0/4.4	140.0/34.7	28.0/7.3	-
InLoc [TOS*18]	3.0/1.1	3.0/1.1	2.0/1.2	3.0/1.1	5.0/1.6	4.0/1.3	9.0/2.5	4.0/1.4	66.3
DSM [TTH*21]	2.0/0.7	2.0/0.9	**1.0/0.8**	3.0/0.8	4.0/1.2	4.0/1.2	5.0/1.4	3.0/1.0	78.1
HACNet [LWZ*20]	2.0/0.7	2.0/0.9	1.0/0.9	3.0/0.8	4.0/1.0	4.0/1.2	3.0/0.8	3.0/0.9	84.8
PixLoc [SUL*21]	2.0/0.8	2.0/0.7	**1.0/0.8**	3.0/0.8	4.0/1.2	3.0/1.2	5.0/1.3	3.0/1.0	75.7
NeRF-Loc [LNLW23]	2.0/1.1	2.0/1.1	1.0/1.9	2.0/1.1	3.0/1.3	3.0/1.5	3.0/1.3	2.0/1.3	89.5
CROSSFIRE [MPB*23]	**1.0/0.4**	5.0/1.9	3.0/2.3	5.0/1.6	3.0/0.8	**2.0/0.8**	12.0/1.9	4.0/1.4	-
DSAC++ [BR18]	2.0/0.5	2.0/0.9	1.0/0.8	3.0/0.7	4.0/1.1	4.0/1.1	9.0/2.6	4.0/1.1	74.4
DSAC* [BR21]	2.0/1.1	2.0/1.2	1.0/1.8	3.0/1.2	4.0/1.3	4.0/1.7	3.0/1.2	3.0/1.4	85.2
SuperPoint+PnP	2.1/0.7	2.5/0.9	1.6/0.9	3.2/0.9	4.0/1.1	3.5/1.2	6.2/1.5	3.3/1.1	76.4
Ours	1.3/0.5	**1.3/0.6**	1.1/0.8	**1.9/0.6**	**2.0/0.7**	2.6/1.0	**1.4/0.4**	**1.7/0.7**	**93.1**

Table 2: *The accuracy for indoor localization on the 12Scenes dataset. The results of other methods are from [LNLW23]*

Method	Accuracy(%) ↑
ORB+PnP	53.7
DSAC++ [BR18]	96.8
SuperPoint + PnP	97.6
DSAC* [BR21]	99.1
HACNet [LWZ*20]	99.3
NeRF-Loc [LNLW23]	99.8
Ours	**99.9**

Table 3: *Evaluation results for the 12Scenes localization dataset. Median translation (cm) and rotation (°) errors and accuracy are reported for each scene.*

Scene	SuperPoint+PnP		Ours	
	Acc.	Errors	Acc.	Errors
apt1_kitchen	1.00	0.8/0.43	1.00	**0.6/0.36**
apt1_living	0.98	1.4/0.52	1.00	**1.1/0.44**
apt2_bed	0.95	2.1/0.83	1.00	**1.7/0.75**
apt2_kitchen	1.00	1.1/0.63	1.00	**1.1/0.54**
apt2_living	1.00	1.3/0.46	1.00	**0.9/0.40**
apt2_luke	0.97	1.5/0.62	0.99	**0.9/0.46**
office1_lounge	0.95	2.0/0.53	0.99	**1.2/0.47**
office1_gates362	0.98	1.2/0.50	1.00	1.2/0.50
office1_manolis	0.98	1.1/0.51	1.00	**1.0/0.46**
office1_gates381	0.99	1.3/0.59	1.00	**1.1/0.52**
office2_5	0.91	1.5/0.63	1.00	**0.9/0.41**
office2_5b	0.97	1.6/0.53	1.00	**1.2/0.43**

sion to maintain the semantic information's consistency around feature points is beneficial to mitigate outliers. For experiments using the keyframe pose as input, the relocalization performance is the opposite of the above description. The result of only warping loss is better than only rendering loss. When using only render losses, the abundance of weak texture areas, similar regions, and planes in indoor scenes hinders accurate camera pose estimation under poor initial poses. Incorporating keypoint and pixel-wise feature warping loss significantly enhances accuracy. The point-to-point constraint remains unaffected by scene appearance changes and ensures that optimization consistently converges around the global optimum when the matching is correct. For experiments using the reference keyframe poses as input, the optimization iteration is set to 100 to ensure the convergence of the pose regression.

4.4. Robustness to Viewpoint Changes

Place Recognition aims to retrieve the most similar keyframes, which means that in most cases, the viewpoint changes between the query image and keyframe are not very drastic. To evaluate the viewpoint change robustness of our method, we randomly select some keyframes within several certain thresholds, The results are shown in Tab. 4. Notably, in indoor scenes like 7-Scenes, even slight angular deviations can lead to dramatic reductions in the common viewing area. In the early stages of iteration, the warping loss dominates the optimization, leveraging the rich scene texture to converge the pose near the global optimum rapidly. Then, the rendering loss plays a major role in optimizing the pose more finely and reducing the impact of mismatches.

4.5. Efficiency Analysis

In Tab. 6, we analyze the elapsed time of each stage. It is worth noting that SuperPoint and LightGlue do not require a backward pass during relocalization. The experiments are conducted on a desktop PC with a 3.60GHz Intel Core i9-9900K CPU and an NVIDIA

Table 4: *Analysis of the impact of prior pose accuracy and the robustness of our method to viewpoint changes. As the difference between the prior pose and the ground truth pose increases, the viewpoint change also increases. δt is the translation difference, varying within $\pm 20cm$. δr is the rotation difference, varying within $\pm 1^\circ$. The median translation (cm) and rotation (°) errors are reported for all scenes of the 7Scenes dataset. The effectiveness of the rendering and warping losses are also evaluated.*

Pose		PnP	Losses		PnP	Losses		PnP	Losses		PnP	Losses	
			Render	Warp		Render	Warp		Render	Warp		Render	Warp
		✓	×	×	✓	✓	×	✓	×	✓	✓	✓	✓
$\delta t = 1m$	$\delta r = 10^\circ$	4.7/1.5			3.3/1.0			4.5/1.2			**3.0/0.9**		
	$\delta r = 30^\circ$	7.2/2.1			4.1/1.4			6.8/1.9			**3.5/1.3**		
	$\delta r = 40^\circ$	11.5/3.6			6.9/2.6			11.5/3.6			**4.6/2.0**		
	$\delta r = 50^\circ$	14.3/4.6			7.6/3.1			12.1/4.5			**6.2/2.6**		
$\delta t = 2m$	$\delta r = 10^\circ$	8.6/2.4			4.8/1.5			8.2/1.8			**4.2/1.4**		
	$\delta r = 30^\circ$	9.8/2.4			5.7/1.6			10.4/2.2			**5.1/1.4**		
	$\delta r = 40^\circ$	13.5/3.4			7.0/2.1			13.7/3.3			**6.3/2.1**		
	$\delta r = 50^\circ$	19.4/4.6			8.8/3.0			16.5/4.3			**8.8/2.9**		
$\delta t = 3m$	$\delta r = 10^\circ$	36.2/6.0			60.3/13.8			36.8/5.7			**21.4/3.8**		
	$\delta r = 30^\circ$	49.8/9.3			30.0/7.5			36.1/7.1			**19.3/3.9**		
	$\delta r = 40^\circ$	35.6/9.1			22.4/5.9			34.1/8.0			**14.8/4.3**		
	$\delta r = 50^\circ$	35.5/9.6			66.4/21.5			37.8/9.5			**26.3/4.6**		

Table 5: *Average median translation (cm) and rotation (°) errors across all 7Scenes, assessed using different prior poses and combinations of losses. 'RF' indicates the pose of the reference keyframe*

Prior	Render Losses		Implicit Warp Losses			Errors
Pose	SSIM	L1	RGB	Keypoint	Feature	(cm/°) ↓
PnP	×	×	×	×	×	3.3/1.10
PnP	×	×	✓	✓	✓	2.8/0.99
PnP	✓	✓	×	×	×	1.8/0.75
PnP	✓	✓	✓	✓	×	1.7/0.70
PnP	×	✓	✓	✓	✓	2.2/0.84
PnP	✓	✓	✓	✓	✓	**1.7/0.68**
RF	✓	✓	×	×	×	5.1/1.74
RF	×	×	✓	✓	✓	3.4/1.20
RF	✓	✓	×	✓	×	2.5/0.94
RF	✓	✓	✓	✓	×	2.1/0.77
RF	×	✓	✓	✓	✓	2.4/0.91
RF	✓	✓	✓	✓	✓	**1.9/0.76**

Table 6: *Elapsed time by each stage of the proposed method.*

Stage	Elapsed time
Global feature extraction	16ms / image
Place recognition	0.009ms / image pair
SuperPoint extraction	12ms / image
LightGlue matching	30ms / image pair
PnP	2ms / image pair
Opt by gaussian	15ms / iter

Table 7: *Comparison with NeRF-Loc [LNLW23] using various initializations, averaged on all scenes of 7-Scene [SGZ* 13]. Following the same experimental setting.*

Method	Good Init.	Total Accuracy ↑
NeRF-Loc [LNLW23]	✓	89.5%
	×	81.2%
Ours	✓	93.1%
	×	85.6%

RTX 3090 GPU. The whole relocalization process includes retrieving the top three keyframes, extracting keypoints and matching between the query image and all three keyframes, calculating the prior pose of the query image by PnP, and optimizing the pose by integrating rendering and warping losses iteratively. As shown in Tab. 6, the whole process can be done in 1 second, which is enough for the camera relocalization problem and indicates the ability of the proposed method to operate in practical applications. Moreover, we have observed that in the majority of cases, the pose tends to converge after roughly 30 rounds of iterations. Speedup can be achieved by monitoring the reduction of losses and stopping early when converging.

4.6. Qualitative Results and Failure Discussion

Fig. 3 provides visualizations of failure cases. In the first case, specifically in the **pumpkin** scene of 7-Scenes, the matching results reveal that most of the inliers are mismatches, primarily caused by mirror reflections. This leads to an incorrect camera pose due to the presence of a large, texture-consistent plane in the image, namely the cabinet. This issue could be addressed by incorporating semantic estimation and implementing stricter consistency checks. The second failure case occurs in the **stairs** scene of 7-Scenes. Despite having correct matching results, each stair step appears very similar in terms of texture and depth, causing the render loss to become

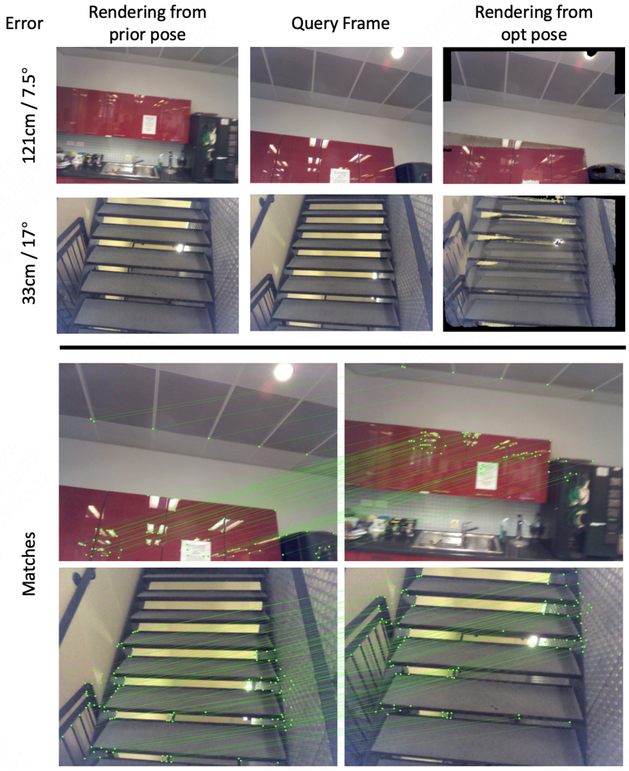


Figure 3: *Visualizations of pose estimation failures resulting from mirror reflection and repetitive textures.*

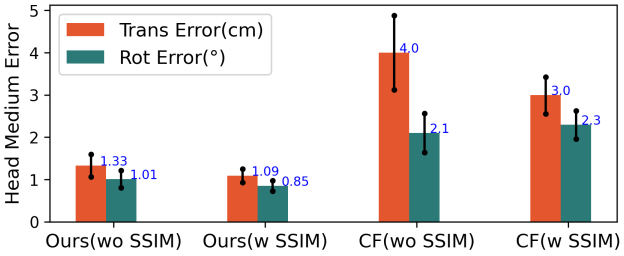


Figure 4: *Error bars representing translation and rotation, highlighting the effectiveness of SSIM loss and comparing our results with CROSSFIRE [MPB*23] on the Head scene of 7Scenes.*

trapped in a local minimum. To potentially overcome this issue, heuristic training strategies and progressive training could be explored as future research directions. Fig. 1, Fig. 5 and Fig. 6 visualize successful relocalization cases, showing rendered views, inlier matches, and relocalization errors. To better showcase the viewpoint changes, all prior poses for visualization are the poses of reference keyframes.

4.7. Comparision with NeRF-based Methods

In this section, we compare our approach to NeRF-based methods, including NeRF-Loc [LNLW23] and CROSSFIRE [MPB*23], to showcase the effectiveness of the proposed feature alignment losses and the use of 3DGS as the map representation. First, we validate the effectiveness in terms of robustness to viewpoint changes. As demonstrated in Tab. 7 and Tab. 8, regardless of whether the initial pose is good or not, our method consistently achieves the best results in both the single scene of Chess and the overall accuracy across all scenes in 7-Scenes [SGZ*13]. Unlike NeRF-Loc [LNLW23] and CROSSFIRE [MPB*23], which rely on rendered depth information to provide 2D-3D matches for pose computation using PnP + RANSAC and optimize over a sparse set of pixels, our method uses per-pixel dense photometric errors. Leveraging 3DGS, which offers significantly faster rendering, our approach constructs a loss with dense pixels, helping to mitigate errors from sparse point matching while maintaining rapid viewpoint changes. The same performance is evident in Fig. 4. When using the same SSIM rendering loss for pose optimization, our method exhibits lower localization errors, while the mismatches negatively impact the localization accuracy of CROSSFIRE. Our approach optimizes the camera pose in an end-to-end manner, incorporating feature-metric losses and rendering loss to reduce the effects of mismatching. Besides, we compare the storage between our method and NeRF-Loc [LNLW23] and CROSSFIRE [MPB*23] on the 7Scenes dataset. The storage requirement for NeRF-Loc is 25.1 MB, while CROSSFIRE requires 50 MB (48 MB for the hash tables and 2 MB for the neural networks). In contrast, our method only needs approximately 20 MB for the Gaussian points and their parameters, considering the real-time rendering capability, our method is especially suitable for deployment on a variety of devices. Moreover, another key reason for choosing 3DGS as the map representation over NeRF is our ultimate goal of developing a comprehensive SLAM system, with GauLoc as a crucial component. After loop closure, which addresses the cumulative error of poses, the explicit 3DGS helps refine the map to maintain global consistency.

Table 8: *Comparison with CROSSFIRE [MPB*23] using various initializations, on the Chess scene of 7-Scene [SGZ*13]. Following the same experimental setting, good initialization means getting prior from image retrieval, and bad means using the same prior for all test images.*

Method	Good Init.	Error (cm/°) ↓
CROSSFIRE [MPB*23]	✓	2.0/0.7
	×	12.0/2.8
Ours	✓	1.3/0.5
	×	6.4/2.0

5. Conclusion and Future Work

This paper introduces a camera relocalization method for scenes represented by 3DGS. Unlike previous approaches relying on pose regression or photometric alignment, our method utilizes the differential rendering capabilities of 3DGS. The proposed implicit featuremetric alignment optimizes the alignment between rendered frames and associated close frames, facilitating the convergence of camera pose estimation. Extensive experiments demonstrate the effectiveness of our approach, highlighting its potential for various real-world applications in robotics. Our current optimization routine requires pre-calibrating the camera's intrinsic parameters. However, we believe that optimizing camera intrinsics and implementing online calibration is worth further investigation, and we plan to explore this idea in future work.

References

[BNIM20] Barath D., Noskova J., Ivashechkin M., Matas J.: Magsac++, a fast, reliable and accurate robust estimator. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2020), pp. 1304–1312. 2

[BR18] Brachmann E., Rother C.: Learning less is more-6d camera localization via 3d surface regression. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2018). 2, 5, 6

[BR21] Brachmann E., Rother C.: Visual camera re-localization from rgb and rgb-d images using dsac. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI) 44*, 9 (2021), 5847–5865. 2, 5, 6

[BWL*23] Bian W., Wang Z., Li K., Bian J.-W., Prisacariu V. A.: Nope-nerf: Optimising neural radiance field with no pose prior. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2023), pp. 4160–4169. 2

[CLWP22] Chen S., Li X., Wang Z., Prisacariu V. A.: Dfnet: Enhance absolute pose regression with direct feature matching. In *European Conference on Computer Vision* (2022), Springer, pp. 1–17. 2, 5, 6

[DBD*22] Darmon F., Bascle B., Devaux J.-C., Monasse P., Aubry M.: Improving neural implicit surfaces geometry with patch warping. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2022). 4

[DMR18] DeTone D., Malisiewicz T., Rabinovich A.: Superpoint: Self-supervised interest point detection and description. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops* (2018), pp. 224–236. 4, 5

[DRMS07] Davison A. J., Reid I. D., Molton N. D., Stasse O.: Monoslam: Real-time single camera slam. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI) 29*, 6 (2007), 1052–1067. 4

[FB81a] Fischler M. A., Bolles R. C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM 24*, 6 (1981), 381–395. 2

[FB81b] Fischler M. A., Bolles R. C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* (1981). 2

[FLK*24] Fu Y., Liu S., Kulkarni A., Kautz J., Efros A. A., Wang X.: Colmap-free 3d gaussian splatting. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2024). 2

[GARL17] Gordo A., Almazan J., Revaud J., Larlus D.: End-to-end learning of deep visual representations for image retrieval. *Intl. Journal of Computer Vision (IJCV) 124*, 2 (2017), 237–254. 2

[GB21] Gridseth M., Barfoot T. D.: Keeping an eye on things: Deep learned features for long-term visual localization. *IEEE Robotics and Automation Letters 7*, 2 (2021), 1016–1023. 1

[GCL*24] Guo X., Chen T., Li B., Liu Q., Jia H., Dai Y.: Learn to triangulate scene coordinates for visual localization. *IEEE Robot. Automat. Lett. (RA-L)* (2024). 1, 2

[GDP*22] Germain H., DeTone D., Pascoe G., Schmidt T., Novotny D., Newcombe R., Sweeney C., Szeliski R., Balntas V.: Feature query networks: neural surface description for camera pose refinement. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2022). 3, 5, 6

[HCG*20] Humenberger M., Cabon Y., Guerin N., Morat J., Revaud J., Rerole P., Pion N., de Souza C., Leroy V., Csurka G.: Robust image retrieval-based visual localization using kapture. *arXiv preprint* (2020). 2

[HLCY24] Huang H., Li L., Cheng H., Yeung S.-K.: Photo-slam: Real-time simultaneous localization and photorealistic mapping for monocular, stereo, and rgb-d cameras. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2024). 2

[IZSB09] Irschara A., Zach C., Sch"onberger J.-M., Bischof H.: From structure-from-motion point clouds to fast location recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2009). 2

[JAC*21] Jeong Y., Ahn S., Choy C., Anandkumar A., Cho M., Park J.: Self-calibrating neural radiance fields. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)* (2021), pp. 5846–5854. 2

[JCF23] Johari M. M., Carta C., Fleuret F.: Eslam: Efficient dense slam system based on hybrid representation of signed distance fields. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2023). 2

[KB14] Kingma D. P., Ba J.: Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014). 5

[KKJ*24] Keetha N., Karhade J., Jatavallabhula K. M., Yang G., Scherer S., Ramanan D., Luiten J.: Splatam: Splat, track & map 3d gaussians for dense rgb-d slam. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2024). 2

[KKLD23] Kerbl B., Kopanas G., Leimkühler T., Drettakis G.: 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics 42*, 4 (2023), 1–14. 1, 3

[LGY*23] Li H., Gu X., Yuan W., Yang L., Dong Z., Tan P.: Dense rgb slam with neural implicit maps. In *Proc. of the Int. Conf. on Learning Representations (ICLR)* (2023). 2

[LGZ*24] Li H., Gao Y., Zhang D., Wu C., Dai Y., Zhao C., Feng H., Ding E., Wang J., Han J.: Ggrt: Towards generalizable 3d gaussians without pose priors in real-time. *arXiv preprint* (2024). 2

[LMTL21] Lin C.-H., Ma W.-C., Torralba A., Lucey S.: Barf: Bundle-adjusting neural radiance fields. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)* (2021), pp. 5741–5751. 2

[LNLW23] Liu J., Nie Q., Liu Y., Wang C.: Nerf-loc: Visual localization with conditional neural radiance field. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)* (2023). 1, 3, 5, 6, 7, 8

[LSP23] Lindenberger P., Sarlin P.-E., Pollefeys M.: Lightglue: Local feature matching at light speed. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)* (2023). 4, 5

[LWZ*20] Li X., Wang S., Zhao Y., Verbeek J., Kannala J.: Hierarchical scene coordinate classification and regression for visual localization. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2020), pp. 11983–11992. 2, 5, 6

[LYC*23] Lin J., Yuan C., Cai Y., Li H., Ren Y., Zou Y., Hong X., Zhang F.: Immesh: An immediate lidar localization and meshing framework. *IEEE Transactions on Robotics* (2023). 1

[MAMT15] Mur-Artal R., Montiel J. M. M., Tardos J. D.: ORB-SLAM: a versatile and accurate monocular slam system. *IEEE Trans. on Robotics (TRO) 31*, 5 (2015), 1147–1163. 2

[MAS*23] Maggio D., Abate M., Shi J., Mario C., Carlone L.: Loc-nerf: Monte carlo localization using neural radiance fields. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)* (2023), IEEE, pp. 4018–4025. 3

[MLG*23] Meuleman A., Liu Y.-L., Gao C., Huang J.-B., Kim C., Kim M. H., Kopf J.: Progressively optimized local radiance fields for robust view synthesis. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2023), pp. 16539–16548. 2

[MMKD24] Matsuki H., Murai R., Kelly P. H., Davison A. J.: Gaussian splatting slam. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2024). 2

[MPB*23] Moreau A., Piasco N., Bennehar M., Tsishkou D., Stanciulescu B., de La Fortelle A.: Crossfire: Camera relocalization on self-supervised features from an implicit representation. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)* (2023). 3, 5, 6, 8

[MPT*22] Moreau A., Piasco N., Tsishkou D., Stanciulescu B., de La Fortelle A.: Lens: Localization enhanced by nerf synthesis. In *Conference on Robot Learning* (2022), PMLR, pp. 1347–1356. 2, 5, 6

[MST*21] 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 (2021), 99–106. 1, 3

[NB17] Naseer T., Burgard W.: Deep regression for monocular camera-based 6-dof global localization in outdoor environments. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2017). 1

[PKS22] Panek V., Kukelova Z., Sattler T.: Meshloc: Mesh-based visual localization. In *European Conference on Computer Vision* (2022), Springer, pp. 589–609. 1

[PN18] Persson M., Nordberg K.: Lambda twist: An accurate fast robust perspective three point (p3p) solver. In *ECCV* (2018). 2

[QLS18] Qin T., Li P., Shen S.: Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. on Robotics (TRO) 34*, 4 (2018), 1004–1020. 2

[RARdS19] Revaud J., Almazán J., Rezende R. S., de Souza C. R. d.: Learning with average precision: Training image retrieval with a listwise loss. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)* (2019), pp. 5107–5116. 2

[SCSD19] Sarlin P.-E., Cadena C., Siegwart R., Dymczyk M.: From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2019), pp. 12716–12725. 2

[SDMR20] Sarlin P.-E., DeTone D., Malisiewicz T., Rabinovich A.: Superglue: Learning feature matching with graph neural networks. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2020). 2

[SF16] Schonberger J. L., Frahm J.-M.: Structure-from-motion revisited. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2016), pp. 4104–4113. 2

[SGZ*13] Shotton J., Glocker B., Zach C., Izadi S., Criminisi A., Fitzgibbon A.: Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2013), pp. 2930–2937. 5, 7, 8

[SHR*15] Sattler T., Havlena M., Radenovic F., Schindler K., Pollefeys M.: Hyperpoints and fine vocabularies for large-scale location recognition. In *ICCV* (2015). 1

[SLK17] Sattler T., Leibe B., Kobbelt L.: Efficient & effective prioritized matching for large-scale image-based localization. *PAMI* (2017). 1

[SLOD21] Sucar E., Liu S., Ortiz J., Davison A. J.: imap: Implicit mapping and positioning in real-time. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2021), pp. 6229–6238. 2

[SPGS18] Schönberger J. L., Pollefeys M., Geiger A., Sattler T.: Semantic visual localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), pp. 6896–6906. 1

[SSS06] Snavely N., Seitz S. M., Szeliski R.: Photo tourism: exploring photo collections in 3d. In *Proc. of the Intl. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)* (2006), pp. 835–846. 2

[SUL*21] Sarlin P.-E., Unagar A., Larsson M., Germain H., Toft C., Larsson V., Pollefeys M., Lepetit V., Hammarstrand L., Kahl F., et al.: Back to the feature: Learning robust camera localization from pixels to pose. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2021). 5, 6

[SWZ*23] Sun Y., Wang X., Zhang Y., Zhang J., Jiang C., Guo Y., Wang F.: icomma: Inverting 3d gaussians splatting for camera pose estimation via comparing and matching. *arXiv preprint* (2023). 3, 5

[TOS*18] Taira H., Okutomi M., Sattler T., Cimpoi M., Pollefeys M., Sivic J., Pajdla T., Torii A.: Inloc: Indoor visual localization with dense matching and view synthesis. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2018), pp. 7199–7209. 5, 6

[TTH*21] Tang S., Tang C., Huang R., Zhu S., Tan P.: Learning camera localization via dense scene matching. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2021). 2, 5, 6

[VDN*16] Valentin J., Dai A., Niessner M., Kohli P., Torr P., Izadi S., Keskin C.: Learning to navigate the energy landscape. In *Proc. of the Intl. Conf. on 3D Vision (3DV)* (2016), IEEE, pp. 323–332. 5

[WSSZ23] Wu C., Sun J., Shen Z., Zhang L.: MapNeRF: Incorporating map priors into neural radiance fields for driving view simulation. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)* (2023). 4

[WWA23] Wang H., Wang J., Agapito L.: Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2023). 2

[WWD*20] Wang J., Wang P., Dai D., Xu M., Chen Z.: Regression forest based rgb-d visual relocalization using coarse-to-fine strategy. *IEEE Robot. Automat. Lett. (RA-L) 5*, 3 (2020). 1, 2

[WWX*21] Wang Z., Wu S., Xie W., Chen M., Prisacariu V. A.: Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint* (2021). 2

[XCL*19] Xin Z., Cai Y., Lu T., Xing X., Cai S., Zhang J., Yang Y., Wang Y.: Localizing discriminative visual landmarks for place recognition. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)* (2019), pp. 5979–5985. 3, 5

[XDW*22] Xie T., Dai K., Wang K., Li R., Wang J., Tang X., Zhao L.: A deep feature aggregation network for accurate indoor camera localization. *IEEE Robot. Automat. Lett. (RA-L) 7*, 2 (2022), 3687–3694. 1, 2

[XYZW24] Xin Z., Yue Y., Zhang L., Wu C.: Hero-slam: Hybrid enhanced robust optimization of neural slam. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)* (2024). 2

[YBT*19] Yang L., Bai Z., Tang C., Li H., Furukawa Y., Tan P.: Sanet: Scene agnostic network for camera localization. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)* (2019). 2, 5, 6

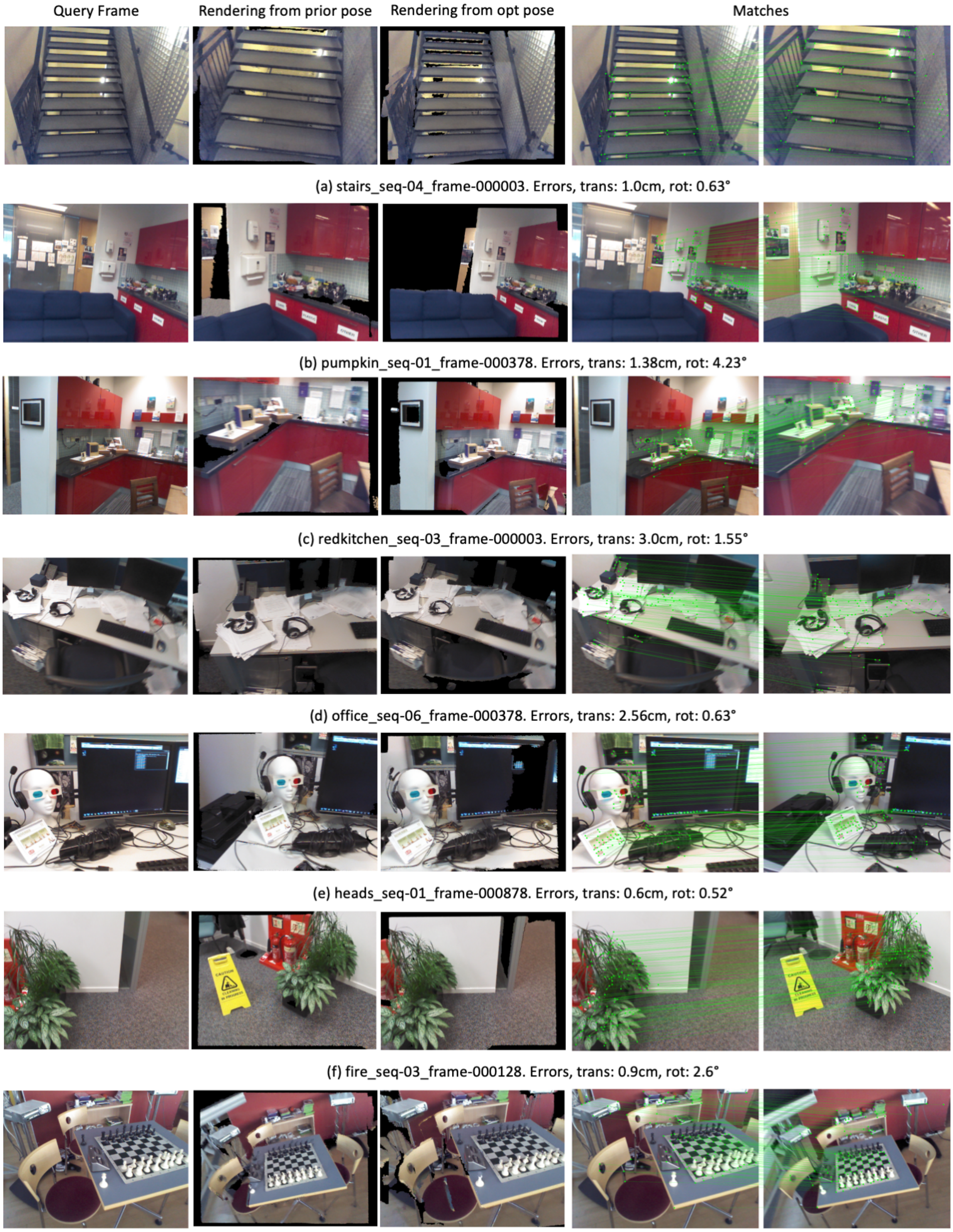
[YCFB*21] Yen-Chen L., Florence P., Barron J. T., Rodriguez A., Isola P., Lin T.-Y.: inerf: Inverting neural radiance fields for pose estimation. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)* (2021), IEEE, pp. 1323–1330. 1, 3, 5

[YK22] Yabuuchi K., Kato S.: Vmvg-loc: Visual localization for autonomous driving using vector map and voxel grid map. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2022), IEEE, pp. 6976–6983. 1

[ZDJF14] Zheng E., Dunn E., Jojic V., Frahm J.-M.: Patchmatch based joint view selection and depthmap estimation. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2014), pp. 1510–1517. 4

[ZPL*22] 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. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2022), pp. 12786–12796. 2

[ZPL*24] Zhu Z., Peng S., Larsson V., Cui Z., Oswald M. R., Geiger A., Pollefeys M.: Nicer-slam: Neural implicit scene encoding for rgb slam. In *Proc. of the Intl. Conf. on 3D Vision (3DV)* (2024). 2



(a) stairs_seq-04_frame-000003. Errors, trans: 1.0cm, rot: 0.63°

(b) pumpkin_seq-01_frame-000378. Errors, trans: 1.38cm, rot: 4.23°

(c) redkitchen_seq-03_frame-000003. Errors, trans: 3.0cm, rot: 1.55°

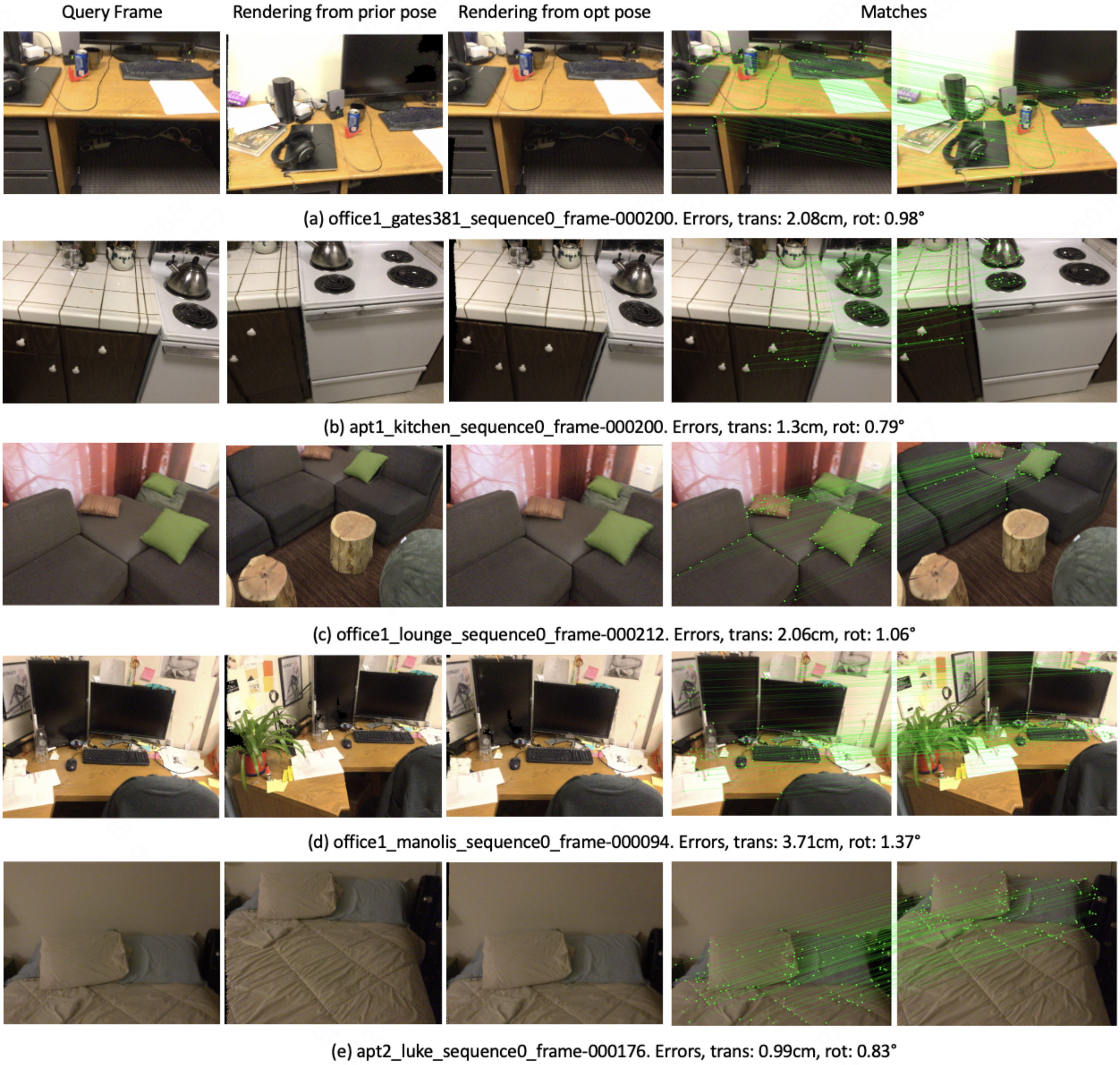
(d) office_seq-06_frame-000378. Errors, trans: 2.56cm, rot: 0.63°

(e) heads_seq-01_frame-000878. Errors, trans: 0.6cm, rot: 0.52°

(f) fire_seq-03_frame-000128. Errors, trans: 0.9cm, rot: 2.6°

(g) chess_seq-05_frame-000628. Errors, trans: 3.2cm, rot: 1.12°

Figure 5: *Visualization of rendered views from prior and opt poses, inlier matches, and relocalization errors for some sample images in 7-Scenes dataset.*



(a) office1_gates381_sequence0_frame-000200. Errors, trans: 2.08cm, rot: 0.98°

(b) apt1_kitchen_sequence0_frame-000200. Errors, trans: 1.3cm, rot: 0.79°

(c) office1_lounge_sequence0_frame-000212. Errors, trans: 2.06cm, rot: 1.06°

(d) office1_manolis_sequence0_frame-000094. Errors, trans: 3.71cm, rot: 1.37°

(e) apt2_luke_sequence0_frame-000176. Errors, trans: 0.99cm, rot: 0.83°

Figure 6: *Visualization of rendered views from prior and opt poses, inlier matches, and relocalization errors for some sample images in 12-Scenes dataset.*