Unlocking Single-View Constraints for Efficient Camera Relocalization with Keypoint-Level Multi-View Geometric Consistency in Training

Hu Lin[1] Chengjiang Long[2*†] Jiqing Zhang[3*] Chuanlu Jiang[1]
Huilin Ge[4*] Erwei Yin[5] Baocai Yin[6*] Xin Yang[1]

[1]Key Laboratory of Social Computing and Cognitive Intelligence, Dalian University of Technology
[2]ByteDance Inc. [3]Dalian Maritime University [4]Jiangsu University of Science and Technology
[5]Tianjin Artificial Intelligence Innovation Center
[6]Beijing Key Lab of Multimedia and Intelligent Software Technology, Beijing University of Technology

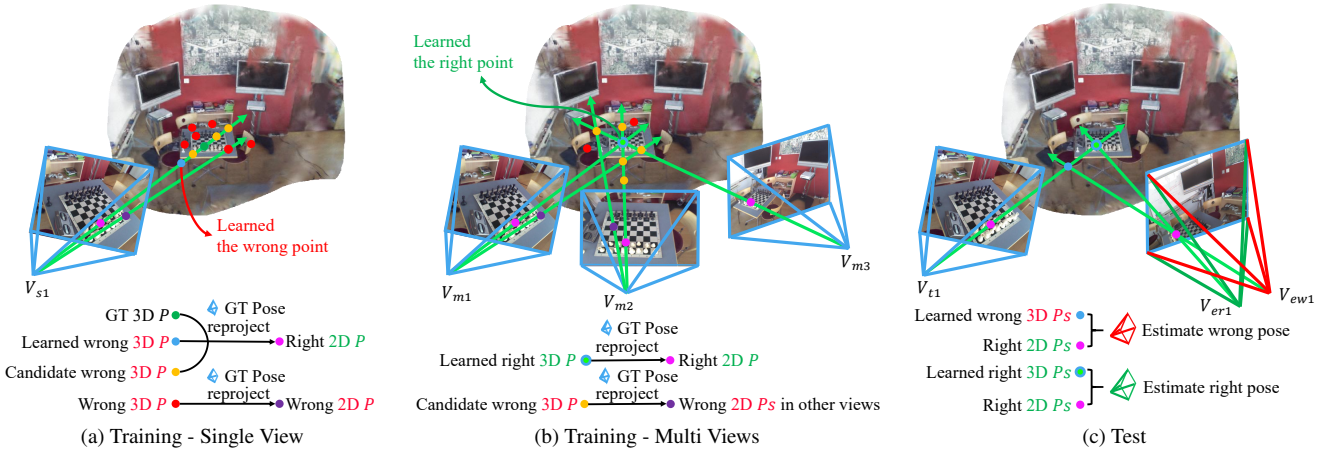


Figure 1. **Motivation of multi-view constraints.** In single-view training (a), depth ambiguities can lead the network to learn incorrect 3D correspondences, which result in erroneous pose estimates at inference. By introducing keypoint-level multi-view supervision (b), our method enforces geometric consistency across viewpoints, ensuring that the correct 3D keypoints are learned. During testing (c), these accurate correspondences enable precise 6-DoF camera relocalization, even under challenging viewpoint changes.

Abstract

Camera relocalization aims to accurately estimate a camera's six degrees of freedom (6-DoF) pose within a known scene, playing a key role in virtual reality, augmented reality, autonomous driving, and robotic navigation. However, camera relocalization from a single image is inherently constrained by limited geometric information, often resulting in lower accuracy compared to multi-view approaches. Yet, multi-view datasets used for training—collected with Structure-from-Motion—embed rich geometric consistency that is rarely exploited to its full potential. In this work, we unlock these single-view constraints by introducing keypoint-level multi-view geometric consistency into the training process of scene coordinate regression. Instead of jointly processing whole image pairs, we select geometrically discriminative keypoints and establish explicit correspondences across views, thereby transferring multi-view geometric knowledge into a single-view inference framework with minimal overhead. Our pipeline further integrates 3D Gaussian rendering to augment keypoint observations and an iterative 2D keypoint refinement during pose optimization, collectively boosting relocalization accuracy without sacrificing efficiency. Experiments on indoor and outdoor benchmarks demonstrate that our method achieves state-of-the-art performance using standard GPUs, and our code will be available at https://github.com/DUT-ICCD/USVC_Reloc.

*Corresponding authors.
†The work was completed when Dr. Chengjiang Long was at Meta.

1. Introduction

Camera relocalization seeks to recover the six degrees of freedom (6-DoF) pose of a camera within a known scene from a *single* input image. It is a core capability in vision-driven systems for virtual/augmented reality, autonomous driving, and robotic navigation [21, 30, 51], where reliability hinges on high precision under diverse viewpoints and environments.

Despite recent advances, single-view relocalization remains fundamentally constrained by limited geometric cues. Even with dense 2D-to-3D predictions in scene coordinate regression (SCR) [9–11, 39], the absence of explicit inter-view constraints leaves depth ambiguities unresolved, occlusions incompletely handled, and scene scale underdetermined [37]. This **single-view bottleneck** often erodes accuracy in complex or visually repetitive scenes, as illustrated in Fig. 1, while classical multi-view geometry, such as Structure-from-Motion (SfM) [48] or SLAM [35], overcomes these issues by optimizing reprojection error across many viewpoints via bundle adjustment, naturally recovering depth, scale, and topology.

However, integrating such multi-view constraints directly into deep neural relocalization is rarely practical for two reasons: (i) **Excessive computation.** jointly processing multiple high-resolution images inflates memory and runtime, making end-to-end training on commodity GPUs infeasible, and (ii) **Incomplete viewpoint coverage.** real training datasets are limited by physical scene access and sensor constraints, restricting the diversity of geometric observations available.

To address these above issues, we propose a novel framework that *unlocks single-view constraints* by injecting *keypoint-level multi-view geometric consistency* during training, as illustrated in Figure 2. We integrate keypoint-level multi-view constraints and Gaussian-based augmentation during training to strengthen 3D scene representation. This ensures the inference to employ a keypoint-guided iterative refinement to recover precise poses from sparse correspondences.

Note that instead of image-level bundle adjustment, we extract geometrically reliable scene keypoints via descriptor matching and confidence filtering, then establish explicit cross-view correspondences. By training on these compact, high-salience features, we retain the benefits of multi-view geometry while slashing the computational cost. To broaden viewpoint diversity without demanding extra captures, we employ *3D Gaussian splatting* [34] to synthesize novel, photorealistic views centered on these keypoints, providing strong geometric priors under varied perspectives.

To summarize, our contributions are three-fold as follows:

- A keypoint-based multi-view constraint formulation for SCR that enriches geometric understanding while remaining GPU-efficient.
- A multi-view observation pipeline for scene keypoints that leverages Gaussian splatting augmentation to improve robustness to viewpoint changes.
- A multi-view optimized training strategy and composite loss design deliver superior performance to state-of-the-art methods on both indoor and outdoor relocalization benchmarks.

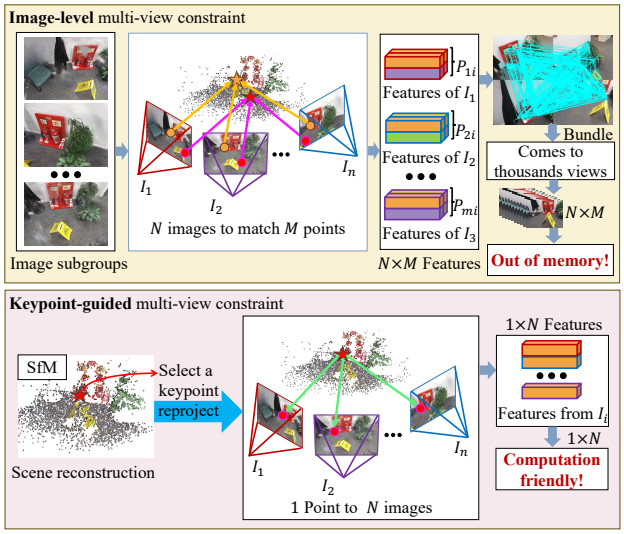


Figure 2. **From image-level to keypoint-level multi-view constraints: unlocking single-view efficiency.** *Top:* Conventional image-level multi-view supervision enforces geometric consistency by jointly matching M scene points across N high-resolution images, incurring $N \times M$ storage and heavy computation, which makes its integration into single-view training impractical. *Bottom:* Our keypoint-guided formulation selects stable 3D points from SfM reconstruction and reprojects each to only its visible images, reducing complexity to $1 \times N$ while preserving strong geometric consistency. This shift to keypoint-level constraints removes the computational bottleneck, enabling multi-view geometric benefits in single-view training without sacrificing efficiency.

2. Related Work

2.1. Camera Relocalization

The development of camera relocalization can be traced back to scene recognition, where techniques were explored for retrieving similar images [21, 30, 51]. This evolution has progressed from traditional bag-of-words models [3] to the learning-based NetVLAD [1, 2] methods. Some approaches have aimed to enhance keypoint extraction and matching techniques, such as the generic feature point extraction and description methods like SiLK [28], SuperPoint [20], and Dedode [25]. Additionally, there are implicit [40, 42] and explicit [4, 22, 27, 43] feature point methods specifically for

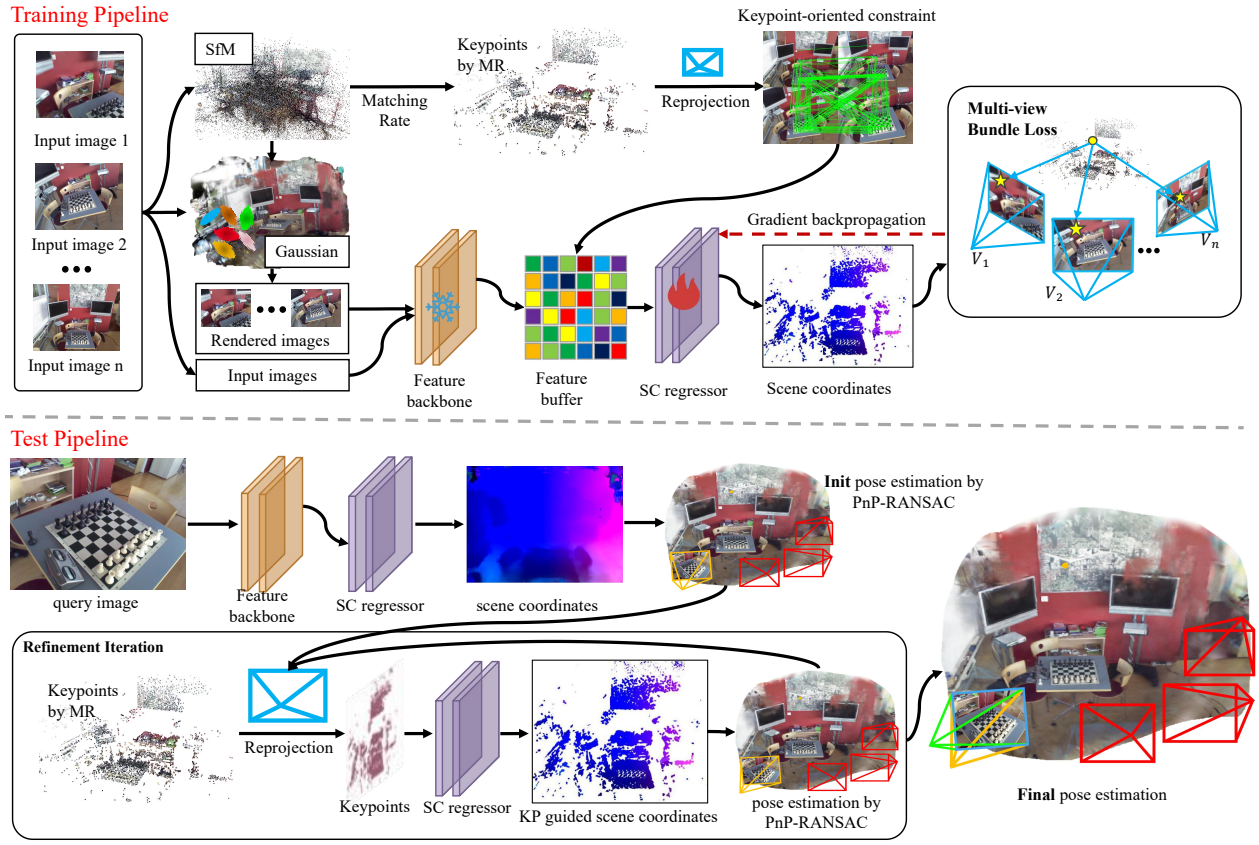


Figure 3. **Pipeline: injecting multi-view geometric consistency into single-view relocalization training.** Given multi-view training images and known camera poses, we identify geometrically reliable scene keypoints through descriptor matching in a sparse SfM reconstruction. A per-scene 3D Gaussian splatting model then renders novel, viewpoint-consistent observations of these keypoints, augmenting real images and expanding geometric coverage without additional data capture. Both real and synthetic keypoint observations are coupled with their camera poses to enable our *bundle loss*—a keypoint-level multi-view constraint that the network learns during training. At inference, these learned keypoint priors guide iterative refinement of sparse 2D–3D correspondences, allowing precise 6-DoF pose estimation from a single image input.

relocalization. The emergence of learning-based methods has propelled advancements in relocalization techniques. Some methods utilize convolutional networks to directly regress camera poses [19, 31–33, 53], while others employ random forest regression [16, 23, 50] to estimate scene coordinates. Recent innovations leverage convolutional networks to regress scene coordinates [6–12, 44, 52], with some scene regression methods attempting to improve localization precision by utilizing match rates [39]. Furthermore, some methods [38] have expanded to novel sensors, such as event cameras, enabling relocalization methods to be applied in more challenging scenes and environments. GLACE [54] is trying to explore implicit scene geometry constraints. FastForward [5] constructs a map representation and performs real-time relocalization of a query image within a single forward pass. ACE-G [13] decouples the coordinate regressor from the map representation, employing a generic Transformer together with a scene-specific map code, which facilitates generalization from mapped images to previously unseen query images.

However, these methods primarily treat the camera relocalization task as a scene coordinate regression task. Inspired by recent research [39, 54], we approach camera relocalization as a task of geometric understanding of the scene. By incorporating scene keypoints from [39], we implement an explicit multi-view constrained scene coordinates regression method, which significantly enhances localization accuracy.

2.2. Multi-View Constraint Optimization and Camera Relocalization

Camera relocalization originates from online SLAM [35] and offline SfM [48], both of which rely on bundle adjustment for accurate 3D reconstruction. Although bundle adjustment remains largely confined to SLAM/SfM, several relocalization methods introduce multi-view cues:

CRGNN [58] aggregates features across neighboring images via graph networks. VGGT [55] employs a Transformer for implicit scene reconstruction. DUST3R [56] and MUST3R [14] integrate explicit multi-view geometric constraints. The angle-based method [37] uses multi-view consistency to improve pose estimates. Despite these advances, they still frame relocalization as pose regression or scene-coordinate estimation. In contrast, our approach treats relocalization as a network-driven exploration of 3D scene geometry, yielding further gains in pose accuracy.

2.3. Model Rendering based Camera Relocalization

Recent view-synthesis advances (e.g., NeRF [41], Gaussian splatting [34]) allow arbitrary-view rendering for data augmentation. Many relocalization methods—such as LTVL [17, 18, 57, 60]—refine camera poses by iteratively optimizing against rendered images, and SFN-eRF [57] further incorporates shadow removal and multi-resolution hash encoding to handle lighting variations. GS-ReLoc [26] leverages 3DGS models to augment image retrieval databases. STDLoc [29] introduces a sparse-to-dense pose estimation framework based on Gaussian splatting. However, these approaches remain constrained by unstable 2D homography. In contrast, we stress 3D structural constraints—largely neglected in prior work—by employing Gaussian rendering to densely sample key scene points, thereby improving keypoint observation and 3D geometry understanding.

Overall, on one hand, our method addresses the deficiency in previous camera relocalization approaches regarding the inadequate utilization of multi-view constraints. On the other hand, our approach leverages the latest Gaussian rendering techniques to actively enhance scene observation under multi-view constraints. These improvements ultimately lead to a significant enhancement in the effectiveness of camera relocalization.

3. Method

Our approach enhances Scene Coordinate Regression (SCR) by explicitly incorporating *multi-view geometric constraints* at the *keypoint level*, thereby avoiding the high computational overhead of image-level bundle adjustment. In addition, we leverage recent advances in *3D Gaussian splatting* to synthesize viewpoint-consistent observations, enriching geometric diversity and improving scene understanding.

As shown in Fig. 3, we operates in two stages:
(1) **Training stage**: sparse scene reconstruction, keypoint selection, Gaussian-based view augmentation, and multi-view–optimized network training;
(2) **Testing stage**: keypoint-guided iterative pose refinement for accurate 6-DoF camera pose estimation.

3.1. Training Stage

The training stage aims to learn a robust scene-coordinate regression network C_ϕ that integrates strong geometric priors via multi-view constraints. We begin by reconstructing a sparse scene from multi-view images, identify geometrically informative scene keypoints with stable visibility and descriptors, enrich their observations via Gaussian splatting, and finally optimize the network with losses combining per-view accuracy and cross-view consistency.

3.1.1. Sparse Scene Reconstruction

We reconstruct a sparse 3D point cloud $\mathcal{X} = \{X_j\}_{j=1}^{N}$ from the input training images using Structure-from-Motion (SfM) [48]. SfM serves purely as a *geometric backbone*, providing: (i) camera extrinsics (R_i, t_i), where $R_i \in SO(3)$ and $t_i \in \mathbb{R}^3$; (ii) camera intrinsics $K \in \mathbb{R}^{3\times3}$; (iii) sparse 3D scene points X_j in world coordinates; and (iv) associated multi-view descriptors for each point. This stage is crucial for building the geometric foundation from which multi-view constraints can be derived.

3.1.2. Acquisition of Scene Keypoints

We focus training on geometrically salient 3D keypoints that maintain consistent multi-view visibility and descriptor similarity. [39] For each point $X \in \mathcal{X}$, we define its set of multi-view observations as:

$$\mathcal{O}(X) = \{(\mathbf{x}_i,\, \mathbf{d}_i)\}_{i=1}^{n},$$

where $\mathbf{x}_i \in \mathbb{R}^2$ is the image coordinate in view i, and $\mathbf{d}_i \in \mathbb{R}^D$ is the associated descriptor vector.

To quantify descriptor consistency, we compute a *matching score*:

$$M(X) = \sum_{i=1}^{n} w_i \, q(\boldsymbol{\phi}(X),\, \mathbf{d}_i)\,, \tag{1}$$

where the mean descriptor $\boldsymbol{\phi}(X)$ and similarity function $q(\cdot)$ are defined as:

$$\boldsymbol{\phi}(X) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{d}_i, \quad q(u,v) = \exp\!\left(-\frac{\|u-v\|_2^2}{2\sigma^2}\right). \tag{2}$$

Here, $w_i \in \mathbb{R}^+$ are per-view importance weights and $\sigma > 0$ is the kernel bandwidth parameter.

We retain only high-confidence keypoints via score thresholding:

$$\mathcal{K} = \{\, X \in \mathcal{X} \mid M(X) \geq \tau \,\}, \quad \tau = 1.5\ [39]. \tag{3}$$

This filtering discards unreliable observations (e.g., due to occlusion or low texture), concentrating training on features with stable multi-view support.

3.1.3. Training Buffer Generation

To efficiently utilize selected keypoints, we construct a training buffer that integrates augmented and real multi-view observations.

Gaussian–Splatting–Enhanced Observations. We train a per-scene 3D Gaussian splatting model [34] using all training images and high-confidence keypoints $\mathcal{K}$. This model generates photorealistic, viewpoint-consistent renderings that preserve geometric structure more faithfully than planar homographies [11], improving the network's generalization across views.

Gradient Decoupling for Multi–View Constraints. Enforcing multi-view constraints directly over all training images is computationally costly. Following ACE [11], we decouple gradient computation at the keypoint level by storing tuples:

$$(k,\, \mathbf{d}_k,\, R_i,\, t_i,\, \mathbf{u}_i^{\text{obs}}(X_k)),$$

where k is the keypoint index, $\mathbf{d}_k$ its descriptor, in there the descriptor $\mathbf{d}_k$ is from ACE feature backbone, (R_i, t_i) the camera pose of view i, and $\mathbf{u}_i^{\text{obs}} \in \mathbb{R}^2$ its ground-truth projection.

Keypoint–Driven Multi–View Correspondences. Given (R_i, t_i) and $X \in \mathcal{K}$, the projection is:

$$\mathbf{u}_i(X) = \pi(X; R_i, t_i) = K(R_i X + t_i) \in \mathbb{R}^2, \quad (4)$$

yielding consistent correspondences that form the basis of our bundle loss.

3.1.4. Loss Functions

We supervise the network with two complementary losses:

Per-View Reprojection Loss. Following ACE [11], for all valid coordinate predictions $\mathcal{V}$, predicted $\mathbf{y}_i \in \mathbb{R}^3$ and ground-truth $\mathbf{x}_i \in \mathbb{R}^2$ under pose $\mathbf{h}_i^* = (R_i^*, t_i^*)$:

$$L_\pi(\mathbf{x}_i, \mathbf{y}_i, \mathbf{h}_i^*) = \begin{cases} e_\pi(\mathbf{x}_i, \mathbf{y}_i, \mathbf{h}_i^*) & \text{if } \mathbf{y}_i \in \mathcal{V}, \\ \|\mathbf{y}_i - \bar{\mathbf{y}}_i\|_0 & \text{otherwise}, \end{cases} \quad (5)$$

with

$$e_\pi = \|\pi(\mathbf{y}_i; \mathbf{h}_i^*) - \mathbf{x}_i\|_2^2.$$

Multi-View Bundle Loss. Grouping by keypoint ID s_k:

$$\mathcal{G}_j = \{X_k \mid s_k = j\}, \quad (6)$$

the bundle loss for group j is:

$$L_j = \sum_{X_k \in \mathcal{G}_j} \sum_{i \in \mathcal{V}_k} \|\mathbf{u}_{ik} - \mathbf{u}_{ik}^{\text{obs}}\|_2^2. \quad (7)$$

Total Loss. Our final objective combines the two losses:

$$L_{\text{total}} = \lambda_\pi L_\pi + \lambda_{\text{B}} \sum_j L_j, \quad (8)$$

where $\lambda_\pi = 1.0$ and $\lambda_{\text{B}} = 1.0$ are weighting coefficients balancing single-view accuracy and multi-view consistency.

3.2. Testing Stage

During inference, the pipeline adapts our keypoint-oriented training to estimate the camera pose from a single input image. Unlike training, where dense and augmented observations are available, testing relies on sparse yet discriminative 2D-3D correspondences derived from the selected scene keypoints.

3.2.1. Dense Scene Coordinate Initialization

Given a query image I, we feed it into C_ϕ to predict:

$$\hat{\mathbf{Y}} = \{\hat{\mathbf{y}}_p \in \mathbb{R}^3 \mid p \in \Omega\},$$

where p denotes a pixel and Ω is the image domain. We then apply RANSAC-PnP [36] to $(p, \hat{\mathbf{y}}_p)$ to obtain an initial pose $(R^{(0)}, t^{(0)})$.

3.2.2. Keypoint-Guided Iterative Refinement

We refine the initial pose by iteratively leveraging $\mathcal{K}$:

Keypoint Mask Extraction. Project each $X_j \in \mathcal{K}$ into the query image:

$$\mathbf{u}_j^{(t)} = \pi(X_j; R^{(t)}, t^{(t)}), \quad (9)$$

construct a binary mask $\mathcal{M}^{(t)}$ indicating projected keypoint positions.

Sparse 2D-3D Correspondence Refinement. From $\mathcal{M}^{(t)}$, extract:

$$\mathcal{C}^{(t)} = \{(p, \hat{\mathbf{y}}_p^{(t)}) \mid \mathcal{M}^{(t)}(p) = 1\},$$

and update $(R^{(t+1)}, t^{(t+1)})$ with RANSAC-PnP.

Iterative Update. Repeat until:

$$\|R^{(t+1)} - R^{(t)}\|_F + \|t^{(t+1)} - t^{(t)}\|_2 < \delta,$$

or $t = T_{\max}$.

3.2.3. Final Output

The converged pose $(R^{(T)}, t^{(T)})$ is returned as the final estimate, benefiting from multi-view-informed training and keypoint-guided refinement.

3.3. Implementation

Network Architecture. Our framework comprises two main modules: (i) a scene coordinate regression network C_ϕ based on a ResNet backbone; and (ii) a feature extraction network F_ψ instantiated with the ACE descriptor extractor [11].

Data Processing and Scene Modeling. The implementation builds upon the open-source ACE codebase in PyTorch and employs COLMAP for Structure-from-Motion (SfM) reconstruction. To accelerate modeling of indoor scenes, we limit the number of input images to at most 500, selected via farthest-point sampling in camera-pose space to preserve spatial coverage. For larger scenes, additional images can be used at the cost of increased computation.

Table 1. Pose relocalization results compared with other methods on 7Scenes dataset within (5°, 5cm).

Scene	DSAC [10]	DSAC++ [6]	Cas. [16]	DSAC*[9]	ACE [11]	EMR [39]	STD(S) [29]	STD(S+D) [29]	**Ours**
Chess	94.6%	93.8%	**100%**	96.7%	**100%**	**100%**	**100.0%**	99.3%	**100%**
Fire	74.3%	75.6%	99.7%	92.9%	99.5%	99.9%	**100%**	**100%**	**100%**
Heads	71.7%	18.4%	**100%**	98.2%	99.7%	**100%**	99.8%	**100%**	**100%**
Office	71.2%	75.4%	99.5%	87.1%	**100%**	99.5%	99.2%	99.8%	99.6%
Pumpkin	53.6%	55.9%	90.9%	60.7%	99.9%	99.9%	**100%**	99.9%	**100%**
Redkitchen	51.2%	50.7%	90.7%	65.3%	98.2%	99.7%	98.4%	**100%**	99.9%
Stairs	4.5%	2.0%	94.2%	64.1%	81.9%	88.6%	80.8%	94.7%	**96.7%**
Average	60.2%	60.4%	96.4%	80.7%	97.0%	98.2%	96.88%	99.09%	**99.45%**

Table 2. Results under smaller thresholds. Accuracy for errors within (2°, 2cm) and (1°, 1cm).

	Within (2°, 2cm)						Within (1°, 1cm)			
	DSAC*	ACE	EMR	STD(S)	STD(S+D)	Ours	DSAC*	ACE	EMR	Ours
Chess	32.8%	99.0%	99.6%	97.20%	98.70%	**99.8%**	0.5%	81.9%	91.8%	**96.4%**
Fire	55.2%	87.1%	95.2%	94.90%	**98.75%**	98.1%	14.8%	57.0%	63.9%	**74.4%**
Heads	87.3%	98.2%	**98.9%**	94.50%	98.40%	98.6%	40.0%	85.3%	87.1%	**92.5%**
Office	32.1%	81.0%	91.2%	80.68%	94.88%	**95.2%**	5.9%	28.0%	60.4%	**69.9%**
Pumpkin	19.8%	84.5%	88.9%	85.65%	85.10%	**95.2%**	4.7%	27.0%	60.7%	**67.6%**
Redkitchen	14.9%	87.0%	94.9%	78.66%	**97.70%**	96.8%	2.6%	45.5%	73.9%	**74.2%**
Stairs	11.4%	24.1%	38.0%	32.50%	**78.60%**	65.2%	1.1%	4.0%	8.1%	**25.5%**
Average	36.2%	80.1%	86.7%	80.58%	**93.16%**	92.7%	9.9%	47.0%	63.7%	**71.5%**

Training Configuration. We train C_ϕ using the AdaBelief optimizer with an initial learning rate of 1×10^{-4}, a feature pool size of 8×10^6, and a total of 32 epochs. For Gaussian-based augmentation, we adopt 3D Gaussian splatting [34], supervising on all training images but restricting augmentation to high-confidence SfM keypoints filtered by matching-rate thresholds. The Gaussian model is optimized for 30,000 iterations with random view-angle sampling to cover unseen perspectives and enrich geometric diversity.

Hardware and Runtime. All experiments are conducted on a dual NVIDIA GeForce RTX 2080Ti setup. Under the indoor scene image-limit protocol described above, per-scene training requires approximately 30 minutes.

3.4. Differences from Recent SCR Methods

The primary distinction between our approach and recent SCR methods lies in the integration of multi-view constraints during training. To achieve this, we shift the utilization of multi-view constraints from an image-based paradigm to a point-based one, and leverage 3D Gaussian splatting to augment multi-view observation capabilities. From the perspective of incorporating explicit geometric constraints into existing SCR pipelines, to the best of our knowledge, our approach is the first of its kind.

4. Experiments

4.1. Overview

We conducted careful experimental validation on publicly available datasets, including the indoor dataset 7Scenes and the outdoor datasets Cambridge Landmarks. In the 7Scenes dataset [50], ground truth (GT) poses were obtained using the KinectFusion system. However, scene coordinate regression methods are better suited for poses from SfM frameworks, so we used SfM-derived poses as pseudo ground truth (pGT). The Cambridge Landmarks dataset [33] contains six outdoor scenes captured by pedestrians with a Google LG Nexus 5 smartphone. Ground truth was obtained using SfM, with a 2Hz sampling rate and approximately 1-meter intervals between camera positions.

4.2. Indoor

Table 1 summarizes results on 7Scenes. By exploiting multi-view geometric constraints, we outperform prior methods, achieving $> 99.5\%$ accuracy within (5°, 5 cm) on every scene—including 96.7% on the challenging *Stairs*. Extending the comparison in Table 2 to tighter thresholds, we obtain an average of 92.7% at (2°, 2 cm) (most scenes $< 90\%$) and exceed 96.4% in *Chess* and 92.5% in *Heads* at (1°, 1 cm). Figure 5 visualizes estimated trajectories against ACE and EMR, clearly showing smaller pose errors, even in *Stairs*.

Table 3. Pose relocalization results compare with other methods on Cambridge Landmarks dataset.

Method	Publication	Year	Type	Depth	Cambridge Landmarks					Average
					Court	King's	Hospital	Shop	St.Mary's	(cm/°)
AS (SIFT) [47]	TPAMI	2016	FM	✗	24/0.1	13/0.2	20/0.4	4/0.2	8/0.3	14/0.2
hLoc(SP+SG) [45]	CVPR	2019	FM	✗	16/0.1	12/0.2	15/0.3	4/0.2	7/0.2	11/0.2
pixLoc [46]	CVPR	2021	FM	✗	30/0.1	14/0.2	16/0.3	5/0.2	10/0.3	15/0.2
GoMatch [61]	ECCV	2022	FM	✗	N/A	25/0.6	283/8.1	48/4.8	335/9.9	N/A
HybridSC [15]	CVPR	2019	FM	✗	N/A	81/0.6	75/1.0	19/0.5	50/0.5	N/A
PoseNet17 [32]	CVPR	2017	APR	✗	683/3.5	88/1.0	320/3.3	88/3.8	157/3.3	267/3.0
MS-Transformer [49]	ICCV	2021	APR	✗	N/A	83/1.5	181/2.4	86/3.1	162/4.0	N/A
SANet [59]	ICCV	2019	SCR	✓	328/2.0	32/0.5	32/0.5	10/0.5	16/0.6	84/0.8
SRC [24]	3DV	2022	SCR	✓	81/0.5	39/0.7	38/0.5	19/1.0	31/1.0	42/0.7
DSAC* [9]	TPAMI	2021	SCR	✗	98/0.5	27/0.4	33/0.6	11/0.5	56/1.8	45/0.8
ACE [11]	CVPR	2023	SCR	✗	43/0.2	28/0.4	31/0.6	5/0.3	18/0.6	25/0.5
Poker(ACE×4) [11]	CVPR	2023	SCR	✗	28/0.1	18/0.3	25/0.5	5/0.3	**9/0.3**	17/0.3
EMR [39]	ACM MM	2024	SCR	✗	46/0.5	21/0.4	23/0.6	5/0.4	12/0.6	22/0.5
GLACE [54]	CVPR	2024	SCR	✗	**19/0.1**	19/0.3	**17/0.4**	**4/0.2**	**9/0.3**	**14/0.3**
ACE-G [13]	ICCV	2025	SCR	✗	51/0.3	21/0.3	23/0.5	7/0.3	25/0.8	25/0.4
Ours	CVPRF	2026	SCR	✗	40/0.1	**17/0.3**	20/0.4	5/0.3	14/0.5	19/0.3

Table 4. Ablation study on 7Scenes datasets (In percentage).

Scene	Within(5°,5cm)				Within(2°,2cm)				Within(1°,1cm)			
	EMR	Normal	Small	3D GS	EMR	Normal	Small	3D GS	EMR	Normal	Small	3D GS
Chess	**100.0%**	**100.0%**	**100.0%**	**100.0%**	99.6%	99.3%	**99.8%**	**99.8%**	91.8%	92.7%	95.9%	**96.4%**
Fire	99.9%	**100.0%**	**100.0%**	**100.0%**	95.2%	97.8%	97.8%	**98.1%**	63.9%	73.5%	**74.4%**	**74.4%**
Heads	**100.0%**	**100.0%**	**100.0%**	**100.0%**	98.9%	**99.8%**	98.0%	98.6%	87.1%	**92.5%**	92.3%	**92.5%**
Office	99.5%	99.5%	99.5%	**99.6%**	91.2%	93.9%	94.6%	**95.2%**	60.4%	61.6%	68.0%	**69.9%**
Pumpkin	99.9%	**100.0%**	**100.0%**	**100.0%**	88.9%	90.3%	94.2%	**95.2%**	60.7%	60.6%	66.0%	**67.6%**
Redkitchen	99.7%	98.9%	99.8%	**99.9%**	94.9%	91.9%	96.5%	**96.8%**	73.9%	72.5%	73.4%	**74.2%**
Stairs	88.6%	92.9%	96.4%	**96.7%**	38.0%	60.6%	**65.5%**	65.2%	8.1%	22.9%	23.9%	**25.5%**
Average	98.2%	98.7%	99.4%	**99.5%**	86.7%	90.5%	92.4%	**92.7%**	63.7%	68.0%	70.6%	**71.5%**

4.3. Outdoor

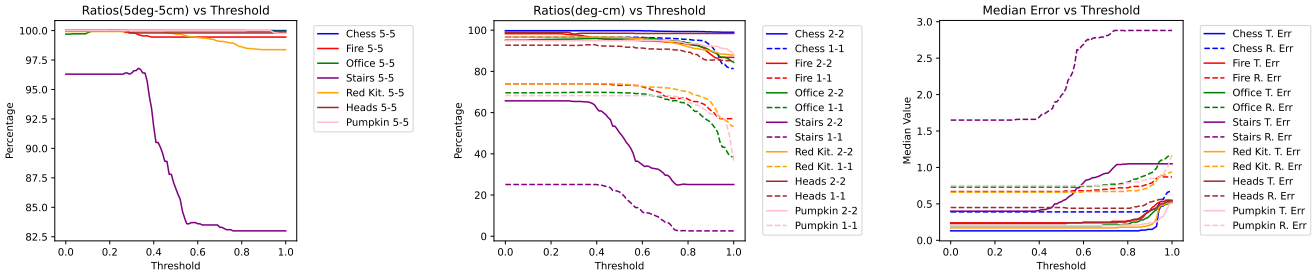
In Table 3, we report median translation and rotation errors on the Cambridge Landmarks dataset. Consistent with our indoor results, our method outperforms other SCR-based approaches in most scenes, yielding the lowest average median errors. The sole exception is St Marys, where dense tree cover occludes key structures and degrades geometric estimation. EMR [39] addresses this via a gating mechanism with a more generalized branch.

4.4. Ablation Study

Our approach combines multi-view geometric constraints with Gaussian-based observation enhancement. Table 4 reports ablations: the "Normal" setting (multi-view only) already outperforms the baselines in Tables 1 and 2, and adding 3D Gaussian splatting ("3D GS") further reduces pose errors. To improve inference speed, we introduce a scaled-feature variant ("Small"), which incurs only a slight accuracy drop while reducing runtime from 1800 ms to 100 ms. As shown in Table 6, this yields higher accuracy without substantially enlarging map size or mapping time. We also evaluated a gating mechanism across various pose-error thresholds (Fig. 4). Although it yields a minor gain within (5°, 5 cm), it offers no benefit at tighter thresholds

Table 5. Compared with the use of the gating mechanism, within (5°, 5 cm).

Method	**ACE** [11]	**Ours**	**Gating** [39]	**Threshold**
Chess	100%	100%	100%	0-100
Fire	99.5%	100%	100%	0-24
Heads	99.7%	100%	100%	0-28
Office	100%	99.6%	100%	19-81
Pumpkin	99.9%	100%	100%	0-93
Red Kit.	98.2%	99.9%	99.9%	0-42
Stairs	81.9%	95.7%	96.8%	33-33
Average	97.03%	99.46%	99.53%	-



(a) Ratio curves within (5°, 5cm). (b) Ratio curves within (2°, 2cm)& (1°, 1cm). (c) Median error curves.

Figure 4. Ratio Curves and Median Error Curves. We plotted ratio curves and median error curves under different threshold values using the gating mechanism in the 7Scenes dataset. Since the accuracies of (5°, 5 cm) and (2°, 2 cm) are relatively similar, we individually present the accuracy curve for (5°, 5 cm) in Figure 4a to facilitate clear comparison. In Figure 4b, we display the accuracy curves for (2°, 2 cm) and (1°, 1 cm). Additionally, Figure 4c illustrates the median error curves for both translation and rotation errors.

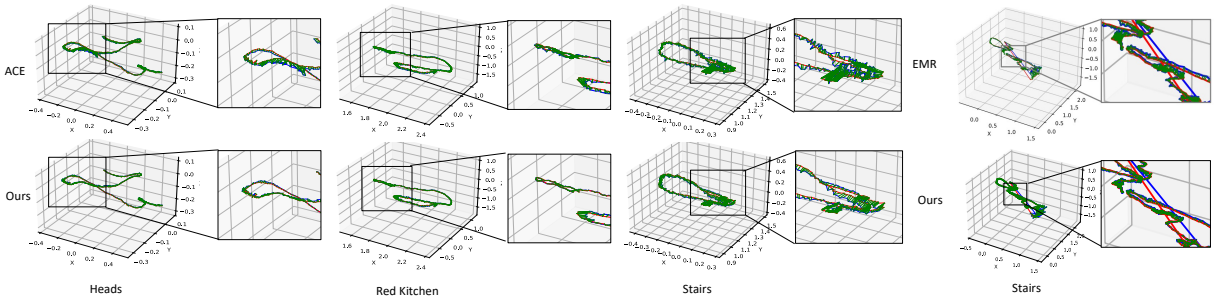


Figure 5. Trace Comaprison. We visualized the differences between the estimated pose trajectories and the ground truth (GT) pose trajectories. In the visualization, the red curve represents the GT trajectory, the blue curve represents the estimated trajectory, and the green lines connect corresponding points between the two trajectories. Consequently, longer green lines indicate larger errors, and more prominent green regions signify greater instability in the estimated trajectory.

Table 6. Computation comparison.

Method	DSAC*	hLoc	ACE	EMR	Ours
Map Size	28 MB	∼3.5 GB	4 MB	∼10MB	11 MB
Time	900 min	200 min	5 min	20min	30 min

and introduces threshold-selection instability; the average accuracy improvement is merely 0.07 %. We attribute our robust performance to Gaussian splatting's enhancement of sparsely observed regions and to multi-view geometric optimization's reduction of residual errors. Given the negligible benefit and added complexity of gating, it is omitted in our final design.

4.5. Limitations

Our method achieves state-of-the-art indoor accuracy of (5 °, 5 cm) and competitive outdoor results. However, its multi-stage pipeline—SfM reconstruction, keypoint extraction, and successive training of keypoint-selection, Gaussian-splatting, and scene-coordinate-regression networks—is more elaborate than end-to-end regressors such as PoseNet. Furthermore, the per-scene training time of approximately 30 minutes leaves considerable room for optimization.

5. Conclusion

In this work, we tackled the core limitation of single-view scene coordinate regression, the absence of explicit multi-view geometric constraints, by shifting the optimization focus to geometrically reliable keypoints and augmenting their observations through 3D Gaussian splatting. This design preserves the depth–scale–topology benefits of multi-view geometry while staying GPU-efficient, effectively removing the single-view bottleneck without the computational burden of image-level bundle adjustment. On challenging indoor benchmarks, our approach not only surpasses prior counterparts under stringent pose-error thresholds but also shows consistent robustness in visually repetitive scenes. Looking ahead, we aim to extend the keypoint-centric training paradigm to large-scale environments and integrate adaptive view synthesis, further strengthening geometric generalization across domains.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62332019, 62576157, U2 5A20442 and 62476113), the China Postdoctoral Science Foundation, China (Grant No.2024M760315 and 2025T18 0437), the Open Research Fund of The Key Laboratory of Social Computing and Cognitive Intelligence (Dalian University of Technology), Ministry of Education (Grant No. S CCI2025YB03), the Ningbo Major Research and Development Plan Project of China (Grant No. 2023Z225), and the Natural Science Foundation of Jiangsu Province of China (Grant No. BK20253020)

References

[1] Relja Arandjelovic and Andrew Zisserman. All about vlad. In *CVPR*, pages 1578–1585, 2013. 2

[2] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *CVPR*, pages 5297–5307, 2016. 2

[3] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. *Modern information retrieval*. ACM press New York, 1999. 2

[4] Axel Barroso-Laguna, Sowmya Munukutla, Victor Adrian Prisacariu, and Eric Brachmann. Matching 2d images in 3d: Metric relative pose from metric correspondences. In *CVPR*, pages 4852–4863, 2024. 2

[5] Axel Barroso-Laguna, Tommaso Cavallari, Victor Adrian Prisacariu, and Eric Brachmann. A scene is worth a thousand features: Feed-forward camera localization from a collection of image features. *arXiv preprint arXiv:2510.00978*, 2025. 3

[6] Eric Brachmann and Carsten Rother. Learning less is more-6d camera localization via 3d surface regression. In *CVPR*, pages 4654–4662, 2018. 3, 6

[7] Eric Brachmann and Carsten Rother. Expert sample consensus applied to camera re-localization. In *ICCV*, pages 7525–7534, 2019.

[8] Eric Brachmann and Carsten Rother. Neural-guided ransac: Learning where to sample model hypotheses. In *ICCV*, pages 4322–4331, 2019.

[9] Eric Brachmann and Carsten Rother. Visual camera relocalization from rgb and rgb-d images using dsac. *TPAMI*, 44(9):5847–5865, 2021. 2, 6, 7

[10] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *CVPR*, pages 6684–6692, 2017. 6

[11] Eric Brachmann, Tommaso Cavallari, and Victor Adrian Prisacariu. Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses. In *CVPR*, pages 5044–5053, 2023. 2, 5, 6, 7

[12] Eric Brachmann, Jamie Wynn, Shuai Chen, Tommaso Cavallari, Áron Monszpart, Daniyar Turmukhambetov, and Victor Adrian Prisacariu. Scene coordinate reconstruction: Posing of image collections via incremental learning of a relocalizer. *arXiv preprint arXiv:2404.14351*, 2024. 3

[13] Leonard Bruns, Axel Barroso-Laguna, Tommaso Cavallari, Aron Monszpart, Sowmya Munukutla, Victor Adrian Prisacariu, and Eric Brachmann. Ace-g: Improving generalization of scene coordinate regression through query pretraining. In *ICCV*, pages 26751–26761, 2025. 3, 7

[14] Yohann Cabon, Lucas Stoffl, Leonid Antsfeld, Gabriela Csurka, Boris Chidlovskii, Jerome Revaud, and Vincent Leroy. Must3r: Multi-view network for stereo 3d reconstruction. In *CVPR*, pages 1050–1060, 2025. 4

[15] Federico Camposeco, Andrea Cohen, Marc Pollefeys, and Torsten Sattler. Hybrid scene compression for visual localization. In *CVPR*, pages 7653–7662, 2019. 7

[16] Tommaso Cavallari, Stuart Golodetz, Nicholas A Lord, Julien Valentin, Victor A Prisacariu, Luigi Di Stefano, and Philip HS Torr. Real-time rgb-d camera pose estimation in novel scenes using a relocalisation cascade. *TPAMI*, 42(10): 2465–2477, 2019. 3, 6

[17] Shuai Chen, Zirui Wang, and Victor Prisacariu. Direct-posenet: Absolute pose regression with photometric consistency. In *3DV*, pages 1175–1185. IEEE, 2021. 4

[18] Shuai Chen, Xinghui Li, Zirui Wang, and Victor A Prisacariu. Dfnet: Enhance absolute pose regression with direct feature matching. In *ECCV*, pages 1–17. Springer, 2022. 4

[19] Shuai Chen, Tommaso Cavallari, Victor Adrian Prisacariu, and Eric Brachmann. Map-relative pose regression for visual re-localization. *arXiv preprint arXiv:2404.09884*, 2024. 3

[20] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPR workshops*, pages 224–236, 2018. 2

[21] Mingyu Ding, Zhe Wang, Jiankai Sun, Jianping Shi, and Ping Luo. Camnet: Coarse-to-fine retrieval for camera relocalization. In *ICCV*, pages 2871–2880, 2019. 2

[22] Tien Do and Sudipta N. Sinha. Improved scene landmark detection for camera localization. In *3DV*, 2024. 2

[23] Siyan Dong, Qingnan Fan, He Wang, Ji Shi, Li Yi, Thomas Funkhouser, Baoquan Chen, and Leonidas J Guibas. Robust neural routing through space partitions for camera relocalization in dynamic indoor environments. In *CVPR*, pages 8544–8554, 2021. 3

[24] Siyan Dong, Shuzhe Wang, Yixin Zhuang, Juho Kannala, Marc Pollefeys, and Baoquan Chen. Visual localization via few-shot scene region classification. In *3DV*, pages 393–402. IEEE, 2022. 7

[25] Johan Edstedt, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. DeDoDe: Detect, Don't Describe — Describe, Don't Detect for Local Feature Matching. In *3DV*. IEEE, 2024. 2

[26] Károly Fodor and András Rövid. Gs-reloc: A gaussian-splatting relocalization method for robust and accurate mono camera pose estimation. *IEEE Access*, 2025. 4

[27] Hugo Germain, Daniel DeTone, Geoffrey Pascoe, Tanner Schmidt, David Novotny, Richard Newcombe, Chris Sweeney, Richard Szeliski, and Vasileios Balntas. Feature query networks: Neural surface description for camera pose refinement. In *CVPRW*, pages 5071–5081, 2022. 2

[28] Pierre Gleize, Weiyao Wang, and Matt Feiszli. Silk–simple learned keypoints. *arXiv preprint arXiv:2304.06194*, 2023. 2

[29] Zhiwei Huang, Hailin Yu, Yichun Shentu, Jin Yuan, and Guofeng Zhang. From sparse to dense: Camera relocalization with scene-specific detector from feature gaussian splatting. In *CVPR*, pages 27059–27069, 2025. 4, 6

[30] Hyo Jin Kim, Enrique Dunn, and Jan-Michael Frahm. Learned contextual feature reweighting for image geo-localization. In *CVPR*, pages 2136–2145, 2017. 2

[31] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *ICRA*, pages 4762–4769, 2016. 3

[32] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. In *CVPR*, pages 5974–5983, 2017. 7

[33] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *ICCV*, pages 2938–2946, 2015. 3, 6

[34] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM ToG*, 42(4), 2023. 2, 4, 5, 6

[35] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *ISMAR*, pages 225–234. IEEE, 2007. 2, 3

[36] Vincent Lepetit, Francesc Moreno-Noguer, and P Fua. Epnp: Efficient perspective-n-point camera pose estimation. *Int. J. Comput. Vis*, 81(2):155–166, 2009. 5

[37] Xiaotian Li, Juha Ylioinas, Jakob Verbeek, and Juho Kannala. Scene coordinate regression with angle-based reprojection loss for camera relocalization. In *ECCV*, pages 0–0, 2018. 2, 4

[38] Hu Lin, Meng Li, Qianchen Xia, Yifeng Fei, Baocai Yin, and Xin Yang. 6-dof pose relocalization for event cameras with entropy frame and attention networks. In *ACM SIGGRAPH VRCAI*, pages 0–8, 2023. 3

[39] Hu Lin, Chengjiang Long, Yifeng Fei, Qianchen Xia, Erwei Yin, Baocai Yin, and Xin Yang. Exploring matching rates: From keypoint selection to camera relocalization. In *ACM MM*, pages 506–514, 2024. 2, 3, 4, 6, 7

[40] Jianlin Liu, Qiang Nie, Yong Liu, and Chengjie Wang. Nerf-loc: Visual localization with conditional neural radiance field. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9385–9392. IEEE, 2023. 2

[41] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 4

[42] Arthur Moreau, Nathan Piasco, Moussab Bennehar, Dzmitry Tsishkou, Bogdan Stanciulescu, and Arnaud de La Fortelle. Crossfire: Camera relocalization on self-supervised features from an implicit representation. In *ICCV*, pages 252–262, 2023. 2

[43] Son Tung Nguyen, Alejandro Fontan, Michael Milford, and Tobias Fischer. Focustune: Tuning visual localization through focus-guided sampling. In *WACV*, pages 3606–3615, 2024. 2

[44] Jerome Revaud, Yohann Cabon, Romain Brégier, JongMin Lee, and Philippe Weinzaepfel. Sacreg: Scene-agnostic coordinate regression for visual localization. In *CVPRW*, pages 688–698, 2024. 3

[45] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *CVPR*, pages 12716–12725, 2019. 7

[46] Paul-Edouard Sarlin, Ajaykumar Unagar, Mans Larsson, Hugo Germain, Carl Toft, Viktor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, et al. Back to the feature: Learning robust camera localization from pixels to pose. In *CVPR*, pages 3247–3257, 2021. 7

[47] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *TPAMI*, 39(9):1744–1756, 2016. 7

[48] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, pages 4104–4113, 2016. 2, 3, 4

[49] Yoli Shavit, Ron Ferens, and Yosi Keller. Learning multi-scene absolute pose regression with transformers. In *ICCV*, pages 2733–2742, 2021.

[50] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, pages 2930–2937, 2013. 3, 6

[51] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *CVPR*, pages 7199–7209, 2018. 2

[52] Shitao Tang, Sicong Tang, Andrea Tagliasacchi, Ping Tan, and Yasutaka Furukawa. Neumap: Neural coordinate mapping by auto-transdecoder for camera localization. In *CVPR*, pages 929–939, 2023. 3

[53] Bing Wang, Changhao Chen, Chris Xiaoxuan Lu, Peijun Zhao, Niki Trigoni, and Andrew Markham. Atloc: Attention guided camera localization. In *AAAI*, pages 10393–10401, 2020. 3

[54] Fangjinhua Wang, Xudong Jiang, Silvano Galliani, Christoph Vogel, and Marc Pollefeys. Glace: Global local accelerated coordinate encoding. In *CVPR*, pages 21562–21571, 2024. 3, 7

[55] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, pages 5294–5306, 2025. 4

[56] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, pages 20697–20709, 2024. 4

[57] Shiyao Xu, Caiyun Liu, Yuantao Chen, Zhenxin Zhu, Zike Yan, Yongliang Shi, Hao Zhao, and Guyue Zhou. Camera relocalization in shadow-free neural radiance fields. *arXiv preprint arXiv:2405.14824*, 2024. 4

[58] Fei Xue, Xin Wu, Shaojun Cai, and Junqiu Wang. Learning multi-view camera relocalization with graph neural networks. In *CVPR*, pages 11372–11381. IEEE, 2020. 4

[59] Luwei Yang, Ziqian Bai, Chengzhou Tang, Honghua Li, Yasutaka Furukawa, and Ping Tan. Sanet: Scene agnostic network for camera localization. In *ICCV*, pages 42–51, 2019. 7

[60] Zichao Zhang, Torsten Sattler, and Davide Scaramuzza. Reference pose generation for long-term visual localization via learned features and view synthesis. *IJCV*, 129(4):821–844, 2021. 4

[61] Qunjie Zhou, Sérgio Agostinho, Aljoša Ošep, and Laura Leal-Taixé. Is geometry enough for matching in visual localization? In *ECCV*, pages 407–425. Springer, 2022. 7