2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

MACE: Mixture-of-Experts Accelerated Coordinate Encoding for Large-Scale Scene Localization and Rendering

Mingkai Liu[1], Dikai Fan[1], Haohua Que[2,5], Haojia Gao[3], Xiao Liu[1], Shuxue Peng[1], Meixia Lin[1], Shengyu Gu[1], Ruicong Ye[4], Wanli Qiu[4], Handong Yao[2], Ruopeng Zhang[6], Xianliang Huang[1,*]

***Abstract*— Efficient localization and high-quality rendering in large-scale scenes remain a significant challenge due to the computational cost involved. While Scene Coordinate Regression (SCR) methods perform well in small-scale localization, they are limited by the capacity of a single network when extended to large-scale scenes. This limitation directly impacts robotics applications, where accurate and efficient scene understanding is essential for navigation and interaction in complex environments. To address these challenges, we propose the Mixed Expert-based Accelerated Coordinate Encoding method (MACE), which enables efficient localization and high-quality rendering in large-scale scenes. Inspired by the remarkable capabilities of MOE in large model domains, we introduce a gating network to implicitly classify and select sub-networks, ensuring that only a single sub-network is activated during each inference. Furtheremore, we present Auxiliary-Loss-Free Load Balancing (ALF-LB) strategy to enhance the localization accuracy on large-scale scene. Our framework provides a significant reduction in costs while maintaining higher precision, offering an efficient solution for large-scale scene applications. Additional experiments on the Cambridge test set demonstrate that our method achieves high-quality rendering results with merely 10 minutes of training.**

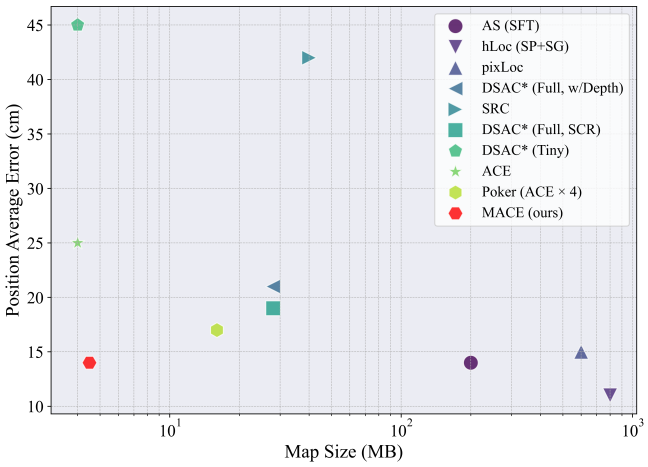

Fig. 1. **Map Size vs Position Average Error.** Comparison with several counterparts on the Cambridge dataset demonstrates that our method reduces activation weight by 72% vs. Poker (ACE × 4) while maintaining low localization errors.

I. INTRODUCTION

Large-scale scene localization and rendering holds significant value in computer vision, which involves recovering the camera pose from a sequence of images and reconstructing visually and structurally complete scenes. However, this task faces considerable challenges stemming from the fundamental trade-off between efficiency, computation burden, and accuracy. Providing robust solutions for large-scale localization and rendering has a direct impact on key applications such as augmented reality [1], embodied robotic perception [2], and robotic navigation [3], [4].

Mainstream localization methods can be divided into two categories: structure-based and Scene Coordinate Regression (SCR) approaches. Structure-based methods utilize SfM to reconstruct 3D point clouds and estimate camera poses via 2D–3D descriptor matching and PnP solvers. However, they incur high computational and storage costs, especially in large-scale scenes due to the need to store dense visual descriptors. In contrast, SCR methods encode scene maps implicitly via deep neural networks, directly regressing 3D coordinates from 2D features without explicit descriptor matching. They perform well in small scenes [5], [6] with fast training times. However, their scalability is limited, as a single network struggles to capture global information in large-scale environments.

Existing methods [7], [8] address large-scale scene localization by employing scene clustering and multi-subnetwork training, where each sub-network specializes in a specific sub-region. While these methods improve accuracy through parallel inference and optimal sub-network selection, it incurs high computational costs and suffers from performance degradation due to suboptimal clustering, limiting its practical applicability. Other approaches [9], [10] leverages global graph encoding and data augmentation, along with depth-aware losses to enhance scalability, but often introduce additional network complexity and preprocessing overhead.

On the other hand, a visually realistic and complete reconstruction of the environment is essential for many applications, going beyond the sparse point sets typically used for localization. In most VR/AR applications, users often need to quickly capture specific real-world views and reproduce them with high quality. Existing methods like SCR and SfM-based approaches typically produce sparse, textureless point clouds that lack details, especially in low-texture regions. They cannot be directly used in application development platforms like Unity for developers to build

[1]PICO, ByteDance Inc. [2]University of Georgia. [3]SIGS, Tsinghua University. [4]Peking University. [5]Infinite Robotics. [6]Chongqing Vocational Institute of Engineering.*Corresponding author: Xianliang Huang (email: huangxianliang@bytedance.com).

AR or robotics applications. Recently, 3DGS has become a mainstream solution for 3D scene representation due to its excellent real-time rendering capability and high-fidelity visual effects. However, traditional 3DGS heavily relies on accurate point cloud priors provided by methods such as SfM, limiting its practicality in large-scale scene rendering.

To address the aforementioned challenges of large-scale scene localization and rendering, we propose an innovative framework **M**ixture-of-experts **A**ccelerated **C**oordinate **E**ncoding termed **MACE**. Different with the Mixture-of-Experts (MoE) paradigm [11], MACE utilizes a gating network to classify global descriptors and dynamically activate a single subnetwork per inference, reducing computational cost to the level of small-scale scenes. To balance sub-network training across experts, we introduce an Auxiliary-Loss-Free Load Balancing (ALF-LB) strategy, which improves both angular and translational accuracy in large-scale localization. For rendering, MACE's high-precision point cloud output is used to regress 3DGS parameters via a Gaussian prediction head. Specifically, the point clouds inferred by MACE are serve as Gaussian centers, while features from fully convolutional upsampled maps are integrated to predict the remaining pixel-aligned 3DGS parameters, enabling high-quality rendering from the input image perspective.

To summarize, our main contributions are as follows: (1) We propose MACE, a Mixture-of-Experts-based framework for large-scale scene localization and rendering, which activates only one sub-network per inference to reduce computation without sacrificing accuracy. (2) We introduce an *Auxiliary-Loss-Free Load Balancing* strategy to ensure effective sub-network training without additional loss terms, achieving lower angular and translational errors. (3) We leverage MACE-inferred point clouds with a Gaussian regression head to predict 3DGS parameters, bridging the gap between localization and rendering. (4) Extensive experiments demonstrate both efficient localization and high-quality rendering on large-scale scenes, outperforming state-of-the-art methods.

II. Related work

A. Pose Regression and Feature Matching

Camera pose estimation has traditionally relies on feature detection and matching [12], [13], but such methods suffer from degraded performance in low-overlap scenarios. Learning-based pose regression methods [14] directly predict 6-DoF poses from images, offering improved robustness but generally lower accuracy compared to geometry-based approaches. Relative pose regression techniques [15], [16], [17] improve generalization by predicting relative transforms between query and reference images, yet their localization precision remains limited. Feature matching becomes the dominant paradigm for visual localization [18], [19], [20], establishing 2D–3D correspondences between query images and pre-constructed 3D models. To support large-scale environments, many approaches [21], [22] employ image retrieval for coarse localization followed by fine-grained matching. However, these methods often incur substantial storage and computational costs due to descriptor-heavy 3D representations. Recent methods such as GoMatch [23] and MeshLoc [24] reduce memory overhead by matching against scene geometry, but still depend on computationally expensive structure-from-motion pipelines. Even with recent acceleration strategies [20], [25], feature-based systems still constrain by long mapping times and large memory usage.

B. Large-scale Scene Coordinate Regression

Scene Coordinate Regression (SCR) methods bypass explicit descriptor matching by directly regressing 3D scene coordinates from 2D image pixels using implicit representations encoded in neural networks [5], [26], [27]. Early SCR approaches relied on random forests [5], [28] or adopted CNN-based architectures [29], [30], offering compact map sizes. ACE [6] has demonstrated strong performance in small-scale scenarios, achieving rapid training speed by forgoing explicit 3D reconstruction. Recently, several approaches have been proposed to improve the scalability and performance of SCR in large-scale scenes. These methods often rely on ground truth 3D coordinates and aim to handle large scenes by dividing them into smaller segments, such as spatial regions [8], voxels [31], or hierarchical clusters [9], [30]. However, extending SCR to large-scale scenes remains challenge. The limited capacity of a single network often constrains performance, and existing solutions typically rely on ensembles of specialized subnetworks [8], [6], leading to increased computational overhead. Our work addresses this gap by introducing a MoE architecture that dynamically activates a single subnetwork per inference. This design preserves the efficiency of single-network methods in small scenes while enabling effective scalability to large environments.

C. Feed-Forward Rendering

Feed-forward methods enable fast inference from sparse views by leveraging large-scale priors and are broadly categorized into NeRF-based and Gaussian-based approaches. NeRF-based methods [32], [33], [34] pioneered the feed-forward rendering paradigm. While Neural Radiance Fields [35] produce photorealistic results, their rendering speed remains a major limitation. In contrast, 3D Gaussian Splatting [36] achieves real-time rendering by replacing expensive volume sampling with rasterized Gaussian primitives. Therefore, Gaussian-based feed-forward extensions, such as GPS-Gaussian [37], pixelSplat [38], and MVSplat [39] outperform previous NeRF-based methods. However, they require hours of per-scene optimization in large-scale scenes. To address this, variants like DepthSplat [40] predict Gaussian parameters from multi-view monocular depth cues, yet still depend on depth supervision. Alternatively, data-driven approaches such as LGM [41] utilize large-scale pretraining but incur significant computational overhead. In this work, we bridges this gap by leveraging accurate geometric priors for feed-forward model with a Gaussian regression head, enabling single-view reconstruction for AR developers.

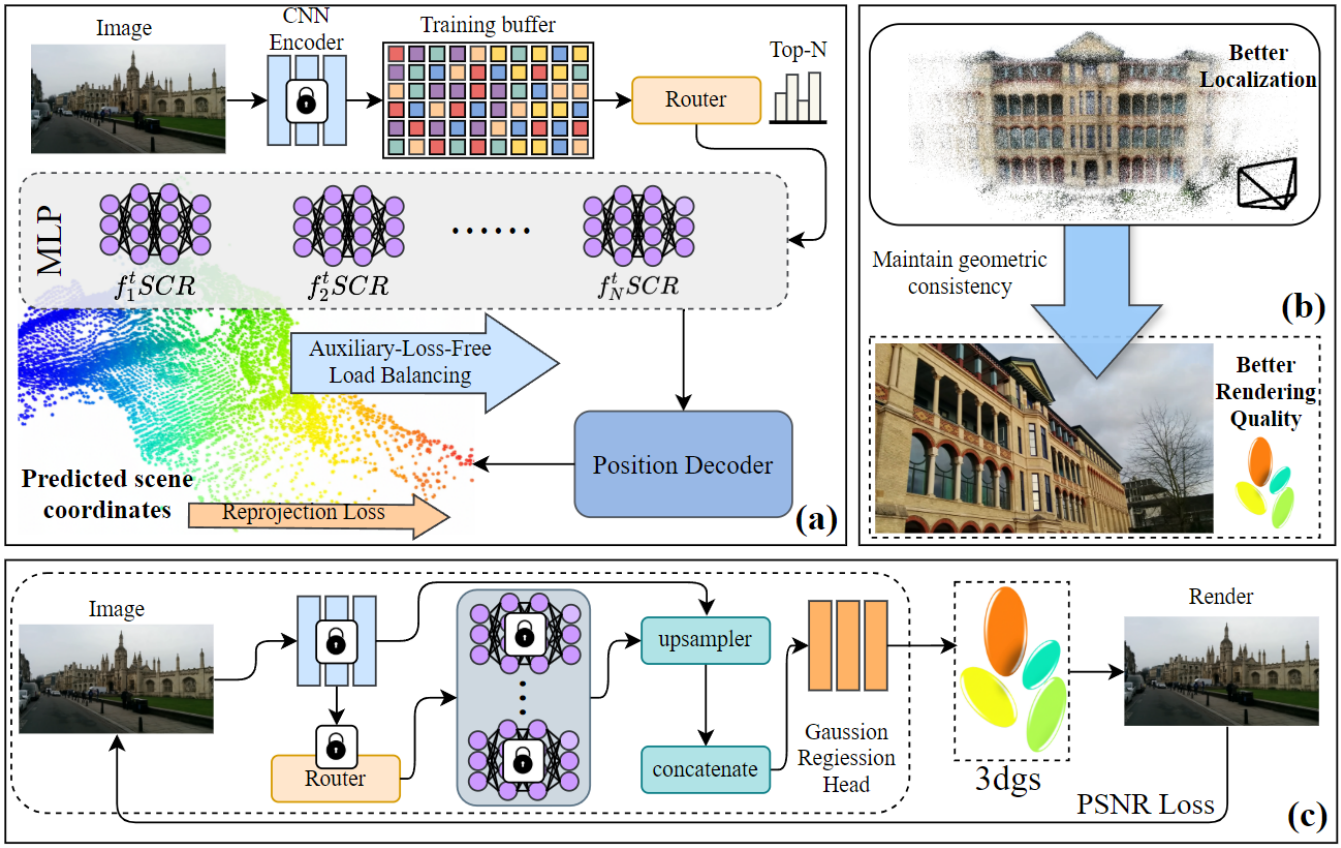

Fig. 2. **Overview of MACE. (a)** Localization Pipeline. Input images are encoded via a pretrained CNN, with local features stored in a training buffer. A router, guided by *Auxiliary-Loss-Free Load Balancing*, selects expert MLPs (f_i^tSCR) jointly optimized by reprojection loss to enforce geometric consistency. The MLP output is refined by a *Position Decoder* to generate final coordinate predictions. **(b)** Localization-rendering cascade. Better localization via the MACE localization pipeline directly improves rendering quality. **(c)** The router selects pretrained expert MLPs for coordinate prediction, which are fused with features and point clouds, then processed by a Gaussian head to generate static views optimized with PSNR loss.

III. Preliminaries

A. Accelerated Coordinate Encoding

SCR methods aim to establish implicit 2D-3D correspondences by predicting a dense scene coordinate map from input images using a convolutional neural network. Given an image patch p_i centered at pixel (x_i, y_i), the network predicts its corresponding 3D coordinate z_i as:

$$z_i = f(p_i), \tag{1}$$

where f denotes the regression function implemented by the neural network.

Traditionally, SCR models were trained using ground-truth 3D scene coordinates as supervision. However, recent advances have enabled training without ground-truth through the use of a differentiable reprojection loss. Accelerated Coordinate Encoding (ACE) [6] adopts this unsupervised training paradigm and achieves strong performance on small-scale scenes. The training is driven by minimizing the reprojection loss over all training views $\{I_i\}_{i=1}^N$:

$$\arg\min_w \sum_{i=1}^{N} \sum_{x_j \in I_i} \ell_\pi \left(x_j, z_j, T_i\right), \tag{2}$$

where w represents the learnable parameters of ACE, and T_i is the camera pose matrix for view I_i. The function ℓ_π is the DSAC*-based reprojection loss [27] that measures the discrepancy between the projected 3D point and its observed 2D location. By optimizing this objective, ACE effectively learns scene geometry in an end-to-end manner without requiring explicit 3D supervision.

B. Challenges of ACE in Large-Scale Scenes

While ACE achieves state-of-the-art performance in small-scale scenes, its extension to large-scale environments introduces critical limits stemming from both its core mechanism and practical deployment constraints.

From a theoretical standpoint, ACE relies on an implicit triangulation process by independently regressing each 2D observation to a 3D point through a neural network. Owing to the smooth nature of neural function approximation, similar visual features are encouraged to regress to similar spatial coordinates by minimizing the aggregate reprojection error:

$$\mathscr{L}_{\text{reproj}} = \sum_i \|\pi(\mathbf{X}_i) - \mathbf{X}_i\|^2, \tag{3}$$

where $\mathbf{x}_i$ is the 2D observation, $\mathbf{X}_i$ is the regressed 3D coordinate, and $\pi(\cdot)$ denotes the camera projection function.

This approach works well in small scenes with consistent visual descriptors but struggles in large-scale settings. Specifically, ACE handles large-scale scenes by dividing them into

spatial clusters, and a dedicated subnet is trained for each. During inference, all subnets are executed in parallel, and the most confident prediction is selected. However, these approaches have two key limitations:

- **Suboptimal clustering boundaries:** Scene partitioning by pose proximity may separate visually or geometrically coherent regions, disrupting feature continuity and impairing learning.
- **High inference cost:** Concurrent execution of all subnets incurs substantial computational overhead and scales linearly with cluster count, limiting scalability to larger scenes and real-time applicability.

These drawbacks motivate a more robust architecture that ensures feature consistency and computational efficiency without hard scene partitioning. Additionally, repeated textures and viewpoint variations cause same point exhibit distinct features under varying perspectives. The mismatch between visual similarity and spatial correspondence, leading to significant localization errors.

IV. Methodology

A. Overview

In this section, we introduce MACE framework to overcome the limitation of ACE in large-scale settings. To enhance localization, we introduce an *Auxiliary-Loss-Free Load Balancing* (ALF-LB) strategy for efficient expert selection. Additionally, a *Position Decoding* module is utilized to mitigate unimodal prior bias. Finally, we integrate the localization component into static-view rendering pipeline, demonstrating that improved localization not only accelerates rendering but also enhances visual quality.

B. MoE for Implicit Global Description

To address the challenge of modeling global information in large-scale scenes, we propose an implicit global representation framework based on a Mixture-of-Experts architecture. As illustrated in Fig. 2, the system dynamically selects a pretrained coordinate regression expert, enabling efficient localization without introducing explicit global descriptors.

a) Expert Pretraining: We first partition the scene into K spatial clusters based on camera pose distribution. For each cluster, we pretrain an expert network $\mathscr{E}_k$ following the ACE [6] pipeline:

$$\mathbf{z}_j = \mathscr{E}_k(\mathbf{f}_j), \tag{4}$$

where $\mathbf{f}_j$ is the local feature at pixel (x_j, y_j) and $\mathbf{z}_j$ is the predicted 3D scene coordinate. Each expert learns a local feature-to-coordinate mapping specific to its subregion, ensuring spatial specialization.

b) Gating Network: With experts fixed, we train a gating network $\mathscr{G}$ to predict the most suitable expert for a given image I_i. The input to $\mathscr{G}$ is an image-level feature embedding, and the predicted expert index $\hat{k}$ minimizes the reprojection loss of selected coordinates:

$$\hat{k} = \arg\min_k \sum_{x_j \in I_i} \ell_\pi\left(\mathscr{E}_k(\mathbf{f}_j), \mathbf{T}_i\right), \tag{5}$$

where $\mathbf{T}_i$ is the ground-truth pose and ℓ_π is the DSAC*-based reprojection loss [27]. This training encourages the gating network to learn global spatial distributions from local features.

c) Joint Optimization: After pretraining, we jointly optimize the gating and expert networks. The final coordinate prediction for pixel (x_j, y_j) in image I_i is given by:

$$\mathbf{z}_j = \mathscr{E}_{\mathscr{G}(I_i)}(\mathbf{f}_j). \tag{6}$$

The entire framework is trained end-to-end with reprojection loss.

The advantage of this architecture lies in its implicit encoding of global context. Since local features encode structural and textural cues, their spatial patterns naturally reflect global subregion affiliation. The gating network learns to leverage this implicit global signal, enabling accurate subregion classification without explicit global descriptors or additional computational overhead.

C. Auxiliary-Loss-Free Load Balancing

The ALF-LB strategy is introduced to mitigate the expert imbalance issue in traditional MoE architectures, where certain sub-networks are excessively activated while others are rarely utilized. In contrast to entropy-based auxiliary loss methods, our approach employs a closed-loop bias modulation mechanism that ensures balanced expert activation while maintaining end-to-end differentiability and avoiding conflicting optimization objectives.

a) Gating with Bias Embedding: The gating network first computes the raw expert logits via an MLP:

$$z = \mathrm{MLP}(x), \tag{7}$$

where x is the input feature and $z \in \mathbb{R}^K$ are the unnormalized selection logits over K experts. A learnable bias term $b \in \mathbb{R}^K$ is then added to guide expert selection:

$$\tilde{z} = z + b. \tag{8}$$

b) Differentiable Expert Selection: We adopt the Gumbel-Softmax trick to enable differentiable sampling from the expert distribution:

$$\alpha_k = \frac{\exp((\tilde{z}_k + g_k)/\tau)}{\sum_{j=1}^K \exp((\tilde{z}_j + g_j)/\tau)}, \quad g_k \sim \mathrm{Gumbel}(0,1), \tag{9}$$

where τ is the temperature, and α_k denotes the soft assignment weight for expert k.

c) EMA-Based Bias Adjustment: To ensure long-term load balancing, we maintain a running estimate of expert usage via exponential moving average (EMA):

$$u_k^{(t)} = \gamma u_k^{(t-1)} + (1-\gamma)\alpha_k^{(t)}, \tag{10}$$

where $u_k^{(t)}$ is the usage of expert k at step t, and γ is the EMA decay rate. The bias b_k is updated as:

$$b_k \leftarrow b_k - \eta \cdot (u_k^{(t)} - \bar{u}), \tag{11}$$

where $\bar{u} = \frac{1}{K}\sum_{j=1}^K u_j^{(t)}$ is the average usage, and η is the adjustment rate.

The final output is computed as a weighted sum of expert predictions:

$$y = \sum_{k=1}^{K} \alpha_k \cdot f_k(x), \tag{12}$$

where f_k is the k-th expert sub-network.

This auxiliary-free balancing strategy enhances the spatial specialization of sub-networks, which is crucial in large-scale scene localization. Balanced expert usage ensures that each expert focuses on distinct spatial regions, improving the discriminative quality of global descriptors and mitigating the issue of spatial coverage bias caused by overfitting in dominant experts.

D. Position Decoding

Previous research [15] has shown that the design of the final layer in a convolutional neural network significantly affects the model's prior when regressing spatial positions. Specifically, when the final linear layer outputs a linear combination of weight bases, it restricts the flexibility of position parameterization. This limitation becomes particularly pronounced in scenarios lacking ground-truth scene coordinate supervision, where the model relies on its prior for implicit triangulation. In the original ACE method, the network predicts an offset relative to a fixed training camera center c in homogeneous coordinates. This design imposes a unimodal prior, causing the predicted positions to cluster around the camera center.

To overcome this limitation, we adopt a more flexible position decoding strategy previously proposed in GLACE [42]. Camera positions in the training set are first grouped into k clusters via K-Means, producing cluster centers $\{\mathbf{c}_i\}_{i=1}^{k}$. The final MLP layer then outputs k logits $\{s_i\}_{i=1}^{k}$ and an offset vector. A convex combination of the cluster centers is computed utilizing the softmax-normalized logits, replacing the original fixed center $\mathbf{c}$. This strategy allows for more expressive and adaptive position estimation, as illustrated below:

$$\hat{\mathbf{c}} = \sum_{i=1}^{k} \frac{e^{s_i}}{\sum_j e^{s_j}} \mathbf{c}_i. \tag{13}$$

This scheme introduces multimodal characteristics through the dynamic weighting of cluster centers, and the effect of this scheme can be observed in the ablation experiments.

E. Combined with Forward Rendering Pipeline

Conventional rendering pipelines rely heavily on Structure-from-Motion (SfM) [43] for point cloud priors, but SfM is inefficient in large-scale scenes. Existing 3D reconstruction methods require depth supervision, point clouds, or data-intensive training. In contrast, we propose the first unsupervised feed-forward rendering pipeline based on Gaussian Splatting [36], enabling scale-consistent rendering without explicit geometric supervision via a localization-optimized framework.

As illustrated in the Fig. 2 (c), we first freeze the parameters of the trained gating network and expert sub-networks from our localization framework. Given a large-scale input view, the network infers a per-view feature map $\mathbf{F}$ and sparse point cloud $\mathscr{P} = \{\mathbf{p}_i \in \mathbb{R}^3\}_{i=1}^{N}$. Then, the feature map $\mathbf{F}$ is up-sampled to a target resolution via fully convolutions. The corresponding point cloud $\mathscr{P}$ is bilinearly interpolated into the same grid, resulting in a dense feature-volume pair $\{\mathbf{F}', \mathbf{P}'\} \in \mathbb{R}^{H \times W \times (C+3)}$. Additionally, we fix $\mathbf{P}'$ as spatial anchors and use a fully convolutional network $\mathscr{F}$ to regress the remaining Gaussian parameters:

$$[\alpha_i, \mathbf{c}_i, \Sigma_i] = \mathscr{F}([\mathbf{F}'_i, \mathbf{P}'_i]), \tag{14}$$

where α, $\mathbf{c}$ and Σ_i are the opacity, color, and covariance, respectively. This factorization decouples geometry from appearance, facilitating efficient learning with consistent structure. The final rendered image $\mathbf{I}_{\text{render}}$ is supervised by a photometric loss against the input image $\mathbf{I}_{\text{gt}}$:

$$\mathscr{L}_{\text{gs}} = (1-\lambda)\mathscr{L}_{\text{MSE}}(\mathbf{I}_{\text{render}}, \mathbf{I}_{\text{gt}}) + \lambda \mathscr{L}_{\text{D-SSIM}}. \tag{15}$$

Our method removes reliance on SfM priors, depth labels, or large datasets. Leveraging localization-enhanced features, the unsupervised feed-forward 3DGS model enables high-fidelity rendering for large-scale rendering.

V. Experiments

A. Experimental Setup

Datasets. We evaluate our MACE on the Cambridge Landmarks dataset [14], which contains extensive outdoor scenes of historic buildings in Cambridge city center. The dataset includes rich sets of mapping and query images, with ground-truth camera poses jointly reconstructed via SfM, providing a reliable benchmark for localization and rendering accuracy.
Baselines. To assess MACE's effectiveness in managing activation map scale for large-scale localization, we compare it with baselines from three paradigms—FM, APR, and SCR—focusing on localization accuracy and computational cost. Key settings and results are summarized in Tab. I.
Metrics. We adopt multi-dimensional metrics to evaluate MACE in large-scale localization and forward rendering tasks. Localization accuracy is measured by the median translation error and rotation error between the predicted and ground-truth poses. Computational efficiency is reflected by the memory footprint of activated map weights. For rendering quality, we assess visual fidelity using Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [44], and Learned Perceptual Image Patch Similarity (LPIPS) [45], comparing rendered views against reference frames.
Implementation Details. MACE is implemented in PyTorch, building upon the public ACE codebase [6]. For standard Cambridge scenes, we train with a batch size of 40K, buffer size of 16M, and 16 epochs on an NVIDIA RTX 3090. For the more complex *GreatCourt* scene, we adopt an enhanced gating network, increase batch size to 160K and buffer size to 64M, extend training to 30 epochs, and utilize an NVIDIA A800. The number of activated sub-networks is dynamically adjusted based on scene complexity. The number of decoder clusters is set to 50, determined via hyperparameter tuning.

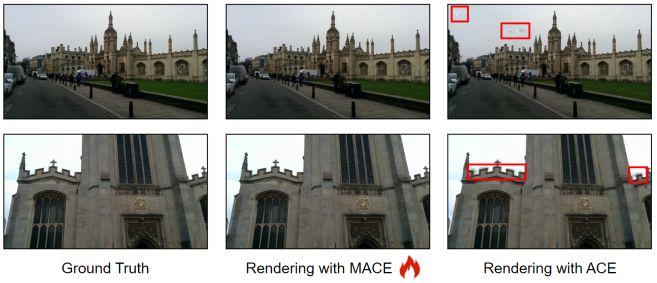

Fig. 3. **Qualitative comparison across different localization frameworks.** Red boxes in ACE highlight artifacts such as misaligned architectural details, emphasizing the superior visual fidelity achieved by MACE through more accurate localization.

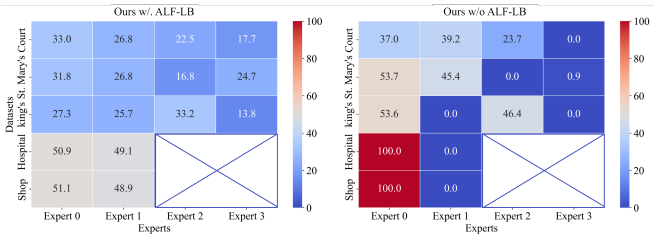

Fig. 4. **Ablation of MoE architecture.** We compare the heatmaps of gating network activations with and without ALF-LB strategy. The left exhibits balanced expert utilization, whereas the right reveals significant imbalance, highlighting ALF-LB's effectiveness in mitigating utilization bias.

When training the Gaussian regression head, we use the AdamW optimizer with a learning rate ranging from 2e-4 to 2e-3, adopting a one cycle learning rate scheduling strategy. To speed up training, the regression head is trained with half-precision floating point weights. All experiments are run on a single NVIDIA A800 GPU. During the 10-minute training, for the training set of a single scene, we conduct 8 epochs with a batch size of 14.

B. Localization and Rendering Results

Localization. As shown in Tab. I, our method significantly outperforms the state-of-the-art SCR methods and approaches the accuracy of FM methods. More importantly, compared with the current leading ACE method, our approach achieves better accuracy while requiring only an activation weight comparable to that of a single sub-network in ACE. This indicates that our method not only improves precision but also enhances computational efficiency by leveraging activation weights more effectively.

Beyond localization accuracy and parameter efficiency, we further evaluate the practical deployment feasibility of our method by analyzing its mapping efficiency. Experiments use A800x1 and RTX 3090x1 GPUs. As shown in Tab. II, similar to ACE, MACE can be trained on a single GPU, ensuring accessibility and practicality in deployment. While maintaining a comparable training time to ACE, MACE achieves significantly higher precision—striking a favorable balance between efficiency and accuracy that underscores its superiority in scene localization.

Rendering Results. As shown in Tab. III, MACE achieves an average PSNR of 34.15 dB, a truly excellent performance

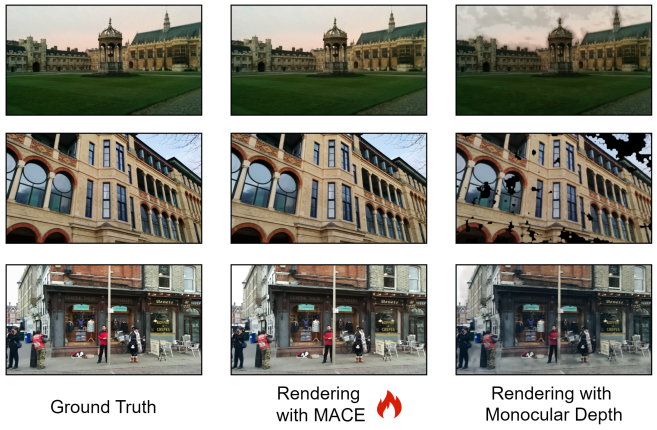

Fig. 5. **Ablation Comparison of Geometric Priors in AR Rendering.** Renderings with monocular depth exhibit severe artifacts like fragmented structures and blackened regions, while MACE - generated renderings closely match the ground truth, demonstrating the superiority of our SCR - based geometric prior in ensuring high - fidelity 3DGS rendering for AR static view tasks.

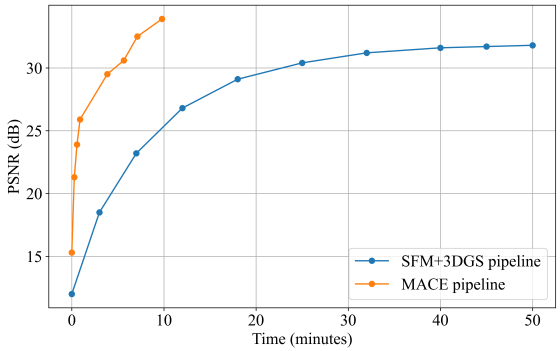

Fig. 6. **Training Time vs. PSNR.** Comparison with SFM+3DGS on the Cambridge dataset shows MACE achieves higher PSNR in 10 minutes than SFM+3DGS does in 50 minutes, highlighting its efficiency from geometric constraints.

that stands out in the field. Visual comparisons in Fig. 3 further confirm that MACE reduces distortions present in ACE-based reconstructions. This demonstrates that improved localization leads to better rendering quality.

To highlight our approach's advantage, we compare training time vs. PSNR with the traditional SFM+3DGS pipeline (Fig. 6). Our pipeline converges to higher rendering quality in under 10 minutes than SFM+3DGS achieves in 50 minutes, underscoring its efficiency and effectiveness in leveraging geometric constraints for better rendering.

C. Ablation Study

To evaluate the efficacy of MACE in large-scale localization tasks, we conduct ablation experiments on the Cambridge Landmarks dataset to assess two key components of MACE. As shown in Tab. I, removing the ALF-LB strategy leads to uneven expert utilization (Fig. 4) and significantly degrades localization accuracy. Additionally, replacing our decoder with ACE's single-mode prior results in an average increase of 3 cm in translation error. These findings confirm the importance of both components in achieving accurate and

TABLE I

CAMBRIDGE LANDMARKS [14] RESULTS. MEDIAN TRANSLATION AND ROTATION ERRORS (CM / °). **BOLD** INDICATES BEST PERFORMANCE IN SCR.

	Method	Mapping w/ Depth	Map Size	GreatCourt	Kings	Hospital	Shop	StMary	Average (cm / °)
FM	AS (SIFT)	No	∼200MB	24/0.1	13/0.2	20/0.4	4/0.2	8/0.3	14/0.2
	hLoc (SP+SG)	No	∼800MB	16/0.1	12/0.2	15/0.3	4/0.2	7/0.2	11/0.2
	pixLoc	No	∼600MB	30/0.1	14/0.2	16/0.3	5/0.2	10/0.3	15/0.2
	GoMatch	No	∼12MB	N/A	25/0.6	283/8.1	48/4.8	335/9.9	N/A
	HybridSC	No	∼1MB	N/A	81/0.6	75/1.0	19/0.5	50/0.5	N/A
APR	PoseNet17	No	50MB	683/3.5	88/1.0	320/3.3	88/3.8	157/3.3	267/3.0
	MS-Transformer	No	∼18MB	N/A	83/1.5	181/2.4	86/3.1	162/4.0	N/A
SCR w/ Depth	DSAC* (Full)	Yes	28MB	49/0.3	15/0.3	21/0.4	5/0.3	13/0.4	21/0.3
	SANet	Yes	∼260MB	328/2.0	32/0.5	32/0.5	10/0.5	16/0.6	84/0.8
	SRC	Yes	40MB	81/0.5	39/0.7	38/0.5	19/1.0	31/1.0	42/0.7
SCR	DSAC* (Full)	No	28MB	34/0.2	18/0.3	21/0.4	5/0.3	15/0.6	19/0.4
	DSAC* (Tiny)	No	4MB	98/0.5	27/0.4	33/0.6	11/0.5	56/1.8	45/0.8
	ACE	No	4MB	43/0.2	28/0.4	31/0.6	5/0.3	18/0.6	25/0.4
	Poker (ACE×4)	No	16MB	28/0.1	18/0.3	25/0.5	5/0.3	**9/0.3**	17/0.3
MACE	Ours w/o ALF-LB	No	4.25∼5.26MB	32/0.2	20/0.3	28/0.5	6/0.3	14/0.4	20/0.3
	Ours w/o Decoder	No	4.25∼5.26MB	27/0.2	18/0.3	21/0.5	5/0.3	11/0.4	16/0.3
	Full model	No	4.25∼5.26MB	**24/0.2**	**15/0.3**	**19/0.4**	**5/0.2**	**9/0.3**	**14/0.3**

TABLE II

MAPPING TIME ACROSS SCENES

Scene	GPU Configuration	Mapping Time
GreatCourt	A800x1	30min
KingsCollege	RTX 3090x1	30min
OldHospital	RTX 3090x1	20min
ShopFacade	RTX 3090x1	20min
StMarysChurch	RTX 3090x1	30min

TABLE III

QUANTITATIVE RENDERING RESULTS ON CAMBRIDGE LANDMARKS.

Scene	PSNR(dB) ↑	SSIM ↑	LPIPS ↓	Time ↓
Hospital	32.56	0.9729	0.0689	590s
King	32.70	0.9641	0.0940	609s
GreadCourt	34.12	0.9722	0.1101	614s
Shop	35.14	0.9809	0.0509	587s
StMary	36.24	0.9799	0.0722	607s
Average	34.15	0.9740	0.0792	601s

stable scene localization.

For downstream large-scale AR static view rendering, we extend ablation analysis to compare our SCR - based geometric prior with monocular depth - derived priors. We introduce a pioneering, unsupervised, and data-agnostic SCR-based fast training method that offers 3DGS a scale-consistent geometric prior. In experiments, we contrast it with using ZoeDepth (monocular depth model) [46] to generate priors via depth map prediction and unprojection. Comparative visuals in Fig. 5 show monocular depth - prior renderings have severe defects like fragmented structures and misalignments, unlike MACE. Our SCR prior encodes geometric consistency, avoiding monocular depth estimation errors (e.g., occlusions, texture - less areas), validating its superiority.

VI. CONCLUSION

We propose MACE, a novel framework for efficient large-scale scene localization and rendering. By introducing auxiliary-loss-free load balancing and an enhanced position decoding module, MACE achieves both accurate localization and efficient computation. Extensive evaluations on Cambridge Landmarks dataset demonstrate that MACE significantly reduces pose errors while maintaining compact activation maps. Furthermore, the integration with 3D Gaussian Splatting enables high-fidelity rendering, highlighting its potential for real-time AR applications on resource-constrained devices. MACE establishes a scalable and accurate paradigm for large-scale scene localization and rendering. Additionally, the proposed framework can be seamlessly integrated into robotics systems, providing robust and efficient scene understanding for autonomous navigation and interaction.

ACKNOWLEDGMENTS

This work was supported by the projects KJQN202503423 and CSTB2025NSCQ-GPX0799. Handong Yao's and Haohua Que's work has been done in the US and is not funded by any projects.

REFERENCES

[1] P.-E. Sarlin, M. Dusmanu, J. L. Schönberger, P. Speciale, L. Gruber, V. Larsson, O. Miksik, and M. Pollefeys, "Lamar: Benchmarking localization and mapping for augmented reality," in *European Conference on Computer Vision*. Springer, 2022, pp. 686–704.

[2] R. Zhang, M. Zhang, J. Zhou, Z. Guo, X. Liu, Z. Xu, Z. Zhong, P. Yan, H. Luo, and X. Li, "Mind-v: Hierarchical video generation for long-horizon robotic manipulation with rl-based physical alignment," *arXiv preprint arXiv:2512.06628*, 2025.

[3] L. Yang, Z. Bai, C. Tang, H. Li, Y. Furukawa, and P. Tan, "Sanet: Scene agnostic network for camera localization," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 42–51.

[4] Y. Liu, L. Liu, Y. Zheng, Y. Liu, F. Dang, N. Li, and K. Ma, "Embodied navigation," *Science China Information Sciences*, vol. 68, no. 4, pp. 1–39, 2025.

[5] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2930–2937.

[6] E. Brachmann, T. Cavallari, and V. A. Prisacariu, "Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 5044–5053.

[7] I. Budvytis, M. Teichmann, T. Vojir, and R. Cipolla, "Large scale joint semantic re-localisation and scene understanding via globally unique instance coordinate regression," *arXiv preprint arXiv:1909.10239*, 2019.

[8] E. Brachmann and C. Rother, "Expert sample consensus applied to camera re-localization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7525–7534.

[9] X. Li, S. Wang, Y. Zhao, J. Verbeek, and J. Kannala, "Hierarchical scene coordinate classification and regression for visual localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 983–11 992.

[10] X. Jiang, F. Wang, S. Galliani, C. Vogel, and M. Pollefeys, "R-score: Revisiting scene coordinate regression for robust large-scale visual localization," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 11 536–11 546.

[11] Z. Yang, Y. Chai, X. Jia, Q. Li, Y. Shao, X. Zhu, H. Su, and J. Yan, "Drivemoe: Mixture-of-experts for vision-language-action model in end-to-end autonomous driving," *arXiv preprint arXiv:2505.16278*, 2025.

[12] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[13] L. Bo, X. Ren, and D. Fox, "Kernel descriptors for visual recognition," *Advances in neural information processing systems*, vol. 23, 2010.

[14] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2938–2946.

[15] T. Sattler, Q. Zhou, M. Pollefeys, and L. Leal-Taixe, "Understanding the limitations of cnn-based absolute camera pose regression," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3302–3312.

[16] M. O. Turkoglu, E. Brachmann, K. Schindler, G. J. Brostow, and A. Monszpart, "Visual camera re-localization using graph neural networks and relative pose supervision," in *2021 International Conference on 3D Vision (3DV)*. IEEE, 2021, pp. 145–155.

[17] S. Chen, T. Cavallari, V. A. Prisacariu, and E. Brachmann, "Map-relative pose regression for visual re-localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 665–20 674.

[18] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 224–236.

[19] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "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. 12 716–12 725.

[20] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 4938–4947.

[21] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5297–5307.

[22] A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, and T. Pajdla, "24/7 place recognition by view synthesis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1808–1817.

[23] Q. Zhou, S. Agostinho, A. Ošep, and L. Leal-Taixé, "Is geometry enough for matching in visual localization?" in *European Conference on Computer Vision*. Springer, 2022, pp. 407–425.

[24] V. Panek, Z. Kukelova, and T. Sattler, "Meshloc: Mesh-based visual localization," in *European Conference on Computer Vision*. Springer, 2022, pp. 589–609.

[25] G. Potje, F. Cadar, A. Araujo, R. Martins, and E. R. Nascimento, "Xfeat: Accelerated features for lightweight image matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 2682–2691.

[26] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother, "Dsac-differentiable ransac for camera localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6684–6692.

[27] E. Brachmann and C. Rother, "Visual camera re-localization from rgb and rgb-d images using dsac," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 9, pp. 5847–5865, 2021.

[28] J. Valentin, V. Vineet, M.-M. Cheng, D. Kim, J. Shotton, P. Kohli, M. Nießner, A. Criminisi, S. Izadi, and P. Torr, "Semanticpaint: Interactive 3d labeling and learning at your fingertips," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 5, pp. 1–17, 2015.

[29] Z. Huang, H. Zhou, Y. Li, B. Yang, Y. Xu, X. Zhou, H. Bao, G. Zhang, and H. Li, "Vs-net: Voting with segmentation for visual localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6101–6111.

[30] S. Wang, Z. Laskar, I. Melekhov, X. Li, Y. Zhao, G. Tolias, and J. Kannala, "Hscnet++: Hierarchical scene coordinate classification and regression for visual localization with transformer," *International Journal of Computer Vision*, vol. 132, no. 7, pp. 2530–2550, 2024.

[31] S. Tang, S. Tang, A. Tagliasacchi, P. Tan, and Y. Furukawa, "Neumap: Neural coordinate mapping by auto-transdecoder for camera localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 929–939.

[32] X. Huang, J. Gou, S. Chen, Z. Zhong, J. Guan, and S. Zhou, "Iddr-ngp: Incorporating detectors for distractors removal with instant neural radiance field," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 1343–1351.

[33] X. Huang, S. Chen, Z. Zhong, J. Gou, J. Guan, and S. Zhou, "Hi-nerf: Hybridizing 2d inpainting with neural radiance fields for 3d scene inpainting," in *Proceedings of the Asian Conference on Computer Vision*, 2024, pp. 2855–2871.

[34] X. Huang, Z. Zhong, S. Chen, Y. Xu, J. Guan, and S. Zhou, "Nerf-mir: Toward high-quality restoration of masked images with neural radiance fields," *IEEE Transactions on Neural Networks and Learning Systems*, 2026.

[35] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[36] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[37] S. Zheng, B. Zhou, R. Shao, B. Liu, S. Zhang, L. Nie, and Y. Liu, "Gps-gaussian: Generalizable pixel-wise 3d gaussian splatting for real-time human novel view synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 19 680–19 690.

[38] D. Charatan, S. L. Li, A. Tagliasacchi, and V. Sitzmann, "pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 19 457–19 467.

[39] Y. Chen, H. Xu, C. Zheng, B. Zhuang, M. Pollefeys, A. Geiger, T.-J. Cham, and J. Cai, "Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images," in *European Conference on Computer Vision*. Springer, 2024, pp. 370–386.

[40] H. Xu, S. Peng, F. Wang, H. Blum, D. Barath, A. Geiger, and M. Pollefeys, "Depthsplat: Connecting gaussian splatting and depth," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 16 453–16 463.

[41] J. Tang, Z. Chen, X. Chen, T. Wang, G. Zeng, and Z. Liu, "Lgm: Large multi-view gaussian model for high-resolution 3d content creation," in *European Conference on Computer Vision*. Springer, 2024, pp. 1–18.

[42] F. Wang, X. Jiang, S. Galliani, C. Vogel, and M. Pollefeys, "Glace: Global local accelerated coordinate encoding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 562–21 571.

[43] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.

[44] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[45] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[46] S. F. Bhat, R. Birkl, D. Wofk, P. Wonka, and M. Müller, "Zoedepth: Zero-shot transfer by combining relative and metric depth," *arXiv preprint arXiv:2302.12288*, 2023.