HGSLoc: 3DGS-based Heuristic Camera Pose Refinement

Zhongyan Niu[1], Zhen Tan[1], Jinpu Zhang[1], Xueliang Yang[1], Dewen Hu*[1]

***Abstract*— Visual localization refers to the process of determining camera poses and orientation within a known scene representation. This task is often complicated by factors such as illumination changes and variations in viewing angles. In this paper, we propose HGSLoc, a novel lightweight, plug-and-play pose optimization framework, which integrates 3D reconstruction with a heuristic refinement strategy to achieve higher pose estimation accuracy. Specifically, we introduce an explicit geometric map for 3D representation and high-fidelity rendering, allowing the generation of high-quality synthesized views to support accurate visual localization. Our method demonstrates a faster rendering speed and higher localization accuracy compared to NeRF-based neural rendering localization approaches. We introduce a heuristic refinement strategy, its efficient optimization capability can quickly locate the target node, while we set the step-level optimization step to enhance the pose accuracy in the scenarios with small errors. With carefully designed heuristic functions, it offers efficient optimization capabilities, enabling rapid error reduction in rough localization estimations. Our method mitigates the dependence on complex neural network models while demonstrating improved robustness against noise and higher localization accuracy in challenging environments, as compared to neural network joint optimization strategies. The optimization framework proposed in this paper introduces novel approaches to visual localization by integrating the advantages of 3D reconstruction and heuristic refinement strategy, which demonstrates strong performance across multiple benchmark datasets, including 7Scenes and DB dataset. The implementation of our method will be made open-source.**

I. INTRODUCTION

Visual localization is a research direction aimed to determine the pose and orientation of a camera within a known scene by analyzing and processing image data. This technique has significant applications in various fields, such as augmented reality (AR), robot navigation, and autonomous driving. By enabling devices to accurately identify their spatial location in complex 3D environments, visual localization facilitates autonomous navigation, environmental awareness, and real-time interaction. The core objective of visual localization is to estimate the camera's absolute pose. However, this task is challenging due to factors like illumination changes, dynamic occlusions, and variations in viewing angles, necessitating the development of robust and efficient algorithms to address these complexities.

Two major categories of methods in visual localization are Absolute Pose Regression (APR) [1]–[8] and Scene Coordinate Regression (SCR) [9]–[11]. APR is an end-to-end deep learning approach that directly regresses the camera's pos from the input image. The key advantages of APR lie in its simplicity and computational efficiency. However, APR exhibits notable limitations, particularly in complex or previously unseen environments, where its generalization capability is weak [12]. In contrast, SCR adopts an indirect strategy for pose estimation. It first predicts the 3D scene coordinates of each image pixel using a deep learning model, followed by the computation of the camera's pose through spatial transformation of these coordinates. While SCR demonstrates high accuracy and robustness in familiar scenes, it incurs substantial computational costs due to the need to predict a large number of pixel-wise coordinates.

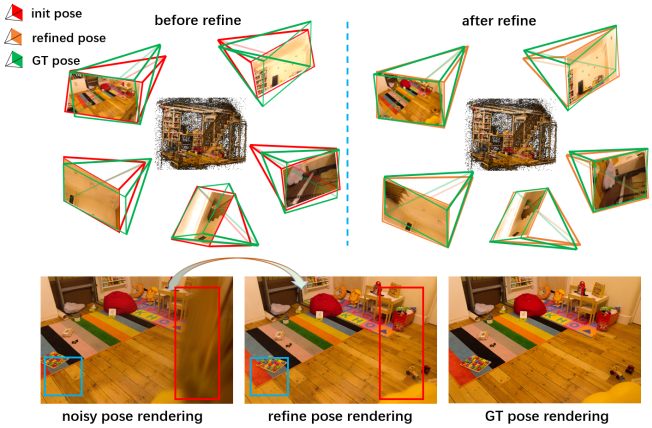

Fig. 1. HGSLoc significantly reduces the error between the coarse pose and the GT, and exhibits strong noise resistance.

In this paper, we propose a novel paradigm based on classical visual localization methods, aimed at improving the precision and accuracy of pos estimation in visual localization by integrating 3D reconstruction. Neural Radiance Field (NeRF) [13], a neural network-based 3D scene modeling approach, is capable of synthesizing and rendering high-quality 3D scene images through neural network training. However, NeRF's pixel-wise training and inference mechanism results in significant computational overhead, limiting its practical applications. In contrast, 3D Gaussian Splatting (3DGS) [14] mitigates this issue by representing scene points as Gaussian distributions, thereby significantly reducing the data processing load during rendering. Furthermore, 3DGS leverages CUDA kernel functions to accelerate training and inference, making it a prominent method in the field of 3D reconstruction. In known or partially known static environments, several approaches, such as 3DGS-ReLoc [15] and GSLoc [16], have been developed. The 3DGS-ReLoc method requires grid search for efficiency in coarse localization using the normalized cross-correlation (NCC) [17] metric, which affects the localization accuracy.The GSLoc

[1]the College of Intelligence Science and Technology, National University of Defense Technology, China.
* indicates corresponding authors: D. Hu (dwhu@nudt.edu.cn)

method has more steps and also uses MASt3R [18] for assisted localization. Whereas, our method is a lightweight framework that enables efficient positional optimization for any image. As shown in Fig. 1, by incorporating 3DGS, richer geometric information is available for pose estimation, and through heuristic optimization of coarse pos estimates, the accuracy of localization can be significantly enhanced in complex scenes.

Absolute Pose Regression (APR) and Scene Coordinate Regression (SCR) provide coarse pose estimates that serve as a foundation for further refinement. To achieve high-quality scene rendering, we introduce the 3D Gaussian Splatting (3DGS), which enriches the database imagery by constructing a dense point cloud, facilitating more detailed scene reconstruction. Building on this, we employ a heuristic refinement algorithm [19] to optimize the estimated poses. With its efficient pathfinding capabilities, combined with a custom-designed heuristic function, the algorithm efficiently adjusts the rendered view of the current pose to match the query image, resulting in more precise pose alignment. Our modular approach significantly reduces dependence on expensive neural network training, offering a more cost-effective solution compared to deep learning methods typically used for pose optimization. Additionally, our method exhibits strong generalization capabilities, maintaining rapid convergence and substantial improvements in pose accuracy, even in the presence of noisy pose data. This adaptability is particularly valuable in practical applications, as it ensures that the proposed method can be deployed across diverse platforms and data quality levels, providing a robust solution for a wide range of scenarios. The effectiveness of our approach is demonstrated through experiments conducted on several benchmark datasets, including 7Scenes and DB. These results underscore the method's performance on classical visual localization datasets as well as those related to 3D Gaussian splatting. The contributions of our approach are summarized as follows:

- We propose a lightweight, plug-and-play pose optimization framework that facilitates efficient pose refinement for any query image.
- We design a heuristic refinement strategy and set the step-level optimization step to adapt various complex scenes.
- Our proposed framework achieves higher localization accuracy than NeRF-based neural rendering localization approaches [20] and outperforms neural network joint pose optimization strategy in noisy conditions.

II. RELATED WORK

In this section, we introduce visual localization methods and 3D Gaussian Splatiing.

A. Visual localization

PoseNet represents a foundational work in the domain of Absolute Pose Regression (APR) [1]–[8], pioneering the direct regression of pose from image data using convolutional neural networks (CNNs). Unlike traditional localization techniques, which typically involve intricate feature extraction, matching, and geometric computation, PoseNet [1] introduces an end-to-end framework that seamlessly integrates these steps into a unified neural network learning process. This approach simplifies the mapping of image data to pose estimation, making it highly suitable for visual localization tasks across diverse environments. Building on PoseNet, MS-Transformer [7] enhances performance by incorporating global context modeling, enabling more effective handling of objects and structures at various scales within an image. The introduction of a multi-head self-attention mechanism allows for a better understanding of complex scenes, leading to significant improvements in pose regression accuracy. Likewise, DFNet [6] extends the capabilities of APR by integrating information from multimodal sensors, offering more comprehensive and detailed modeling of visual scenes. This fusion of multimodal data leverages the complementary strengths of different data sources, enhancing robustness and adaptability to various environmental factors. However, despite the advantages offered by APR methods, they remain vulnerable to noise and environmental variability. Under adverse conditions, such as poor lighting, unfavorable weather, or occlusions, the regression models' accuracy in pose estimation can degrade significantly.

Scene Coordinate Regression (SCR) methods [9]–[11] estimate camera pose by learning the mapping between image pixels and corresponding 3D scene coordinates. These approaches bypass the complex feature matching procedures characteristic of traditional localization methods, thereby enhancing the efficiency and robustness of pose estimation. DSAC* [9] further advances SCR by introducing a differentiable hypothesis selection mechanism, allowing the model to learn how to choose the optimal pose hypothesis during network training. Additionally, it accommodates both RGB and RGB-D image inputs, incorporating depth map information into the pose estimation process, which enhances the model's ability to interpret and manage complex scenes. On the other hand, ACE [10] accelerates feature matching by optimizing the encoding and decoding of image coordinates, which enables faster processing. Furthermore, it demonstrates resilience to noise and lighting variations, improving its robustness in dynamic or less controlled environments. By addressing these common challenges, ACE contributes to more reliable pose estimation in scenes where traditional methods may struggle to maintain accuracy.

B. 3D Gaussian Splatting

3D Gaussian Splatting (3DGS) [14], an emerging method in 3D reconstruction, has rapidly gained prominence since its introduction. This method significantly accelerates the synthesis of new views by modeling the scene with Gaussian ellipsoids and utilizing advanced rendering methods. Within the realm of 3DGS research, various techniques have enhanced and optimized 3DGS in different aspects, such as quality improvement [21], compression and regularization [22], dynamic 3D reconstruction [23], and handling chal-

lenging inputs [24]. The advancement of 3DGS methods not only enhances the quality of scene reconstruction but also speeds up rendering, offering novel and improved approaches for visual localization tasks. For instance, GSLoc leverages rendered images from new viewpoints for matching and pose optimization, while the InstantSplat [25] method, utilizing DUSt3R [26], achieves rapid and high-quality scene reconstruction by jointly optimizing poses with 3D Gaussian parameters. Our proposed method builds upon 3DGS reconstructed scenes and employs heuristic pose optimization to enhance pose accuracy in specific scenarios while preserving the original pose accuracy.

III. METHOD

In this section, we outline the fundamental principles of the 3D Gaussian Splatting (3DGS) and heuristic refinement strategy, along with their integrated implementation. An overview of our framework is depicted in Fig. 2.

A. Explicit Geometric Map

3D Gaussian Splatting (3DGS) [14] is a method for representing and rendering three-dimensional scenes. It models the distribution of objects within a scene using 3D Gaussian functions and approximates object surface colors through spherical harmonic coefficients. This method not only delivers an accurate depiction of scene geometry but also effectively captures and renders the lighting and color variations. In 3DGS, each primitive is characterized by a three-dimensional covariance matrix $\Sigma_i \in \mathbb{R}^{3\times3}$ and mean value $\mu_i \in \mathbb{R}^3$:

$$g_i(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\mu_i)^\top \Sigma_i^{-1}(\mathbf{x}-\mu_i)} \tag{1}$$

where $\Sigma = \mathbf{RSS}^\top\mathbf{R}^\top$, $\mathbf{R} \in \mathbb{R}^{3\times3}$ represents the rotation, $\mathbf{S} \in \mathbb{R}^3$ represents the anisotropy scale.

When projecting onto the viewing plane, 3D Gaussian Splatting (3DGS) utilizes a 2D Gaussian directly, rather than performing the axial integral of a 3D Gaussian. This approach addresses the computational challenge of requiring a large number of samples by limiting the computation to the number of Gaussians, thereby enhancing efficiency. The projected 2D covariance matrix and means are $\Sigma' = \mathbf{JW}\Sigma\mathbf{W}^\mathbf{T}\mathbf{J}^\mathbf{T}$ and $\mu' = \mathbf{JW}\mu$, respectively, where $\mathbf{W}$ represents the transformation from the world coordinate system to the camera coordinate system and $\mathbf{J}$ denotes the radial approximation of the Jacobian matrix for the projection transformation.

During the rendering phase, spatial depth and tile ID are utilized as key values to sort the Gaussian primitives using GPU-based ordering. Subsequently, the color of each pixel is computed based on the volume rendering formula:

$$C = \sum_{i\in\mathcal{N}} \mathbf{c}_i p_i \alpha_i \prod_{j=1}^{i-1}(1-p_i\alpha_j) \tag{2}$$

Where:

$$p_i = e^{-\frac{1}{2}(x-\mu_i')^T\Sigma'^{-1}(x-\mu_i')} \tag{3}$$

$$\alpha_{2d} = 1 - \exp\left(-\frac{\alpha_{3d}}{\sqrt{\det(\Sigma_{3d})}}\right) \tag{4}$$

A major advantage of 3D Gaussian Splatting (3DGS) is its efficient rendering speed. By leveraging CUDA kernel functions for pixel-level parallel processing, 3DGS achieves rapid training and rendering. Additionally, 3DGS employs adaptive control strategies to accommodate objects of various shapes, enhancing both the accuracy and efficiency of rendering. This results in high-quality reconstructed scenes and more realistic new-view images, which provide opportunities for further advancements in pose accuracy.

B. Heuristic Algorithm Implementation

Heuristic approaches [19] are often implemented to path planning and graph search that combines the strengths of depth-first search (DFS) and breadth-first search (BFS). It has been widely applied to various real-world problems, including game development, robot navigation, and geographic information systems (GIS). The primary goal of the heuristic algorithm is to efficiently find the optimal path from an initial node to a goal node, where each node represents a state within the search space. The algorithm relies on an evaluation function, $f(n)$, to prioritize nodes for expansion. This function typically consists of two components:

$$f(n) = g(n) + h(n) \tag{5}$$

Where $g(n)$ function is the actual cost from the start node to the current node; $h(n)$ function is the estimated cost from the current node to the target node.

The core idea of the heuristic algorithm is to minimize the number of expanded nodes by guiding the search direction using a heuristic function, $h(n)$, while ensuring the least costly path. The heuristic function must satisfy two important properties: Admissibility and Consistency. Admissibility ensures that $h(n)$ never overestimates the cost of traveling from node n to the target node. Consistency requires that for any node n and its neighboring node n', the heuristic function satisfies the following condition:

$$h(n) \leq g(n,n') + h(n') \tag{6}$$

Where $g(n,n')$ denotes the actual cost from n to n', which ensures that the algorithm does not repeatedly return to an already expanded node. The algorithm has Optimality and Completeness, i.e., it is guaranteed to find the most optimal path from the start node to the goal node, and for a finite search space, the algorithm always finds a solution.

We use 3DGS as a new-viewpoint image renderer with the goal of finding a more suitable pose within a certain range around the initial pose. A pose is characterized by $(q_w, q_x, q_y, q_z, t_x, t_y, t_z)$, where q_i represent quaternion of a rotation and t_i represent translation. We set the rotation and translation variations δ_{q_i} and δ_{t_i}, and the current node is transformed to other neighboring nodes by different variations. The pose can be viewed as nodes in the search space, while the transitions between different pose correspond to edges in the graph, and this process can be viewed as expanding nodes in the search graph. In this application, the key to the heuristic algorithm is to design a reasonable cost function. We design the actual cost of a child node as the

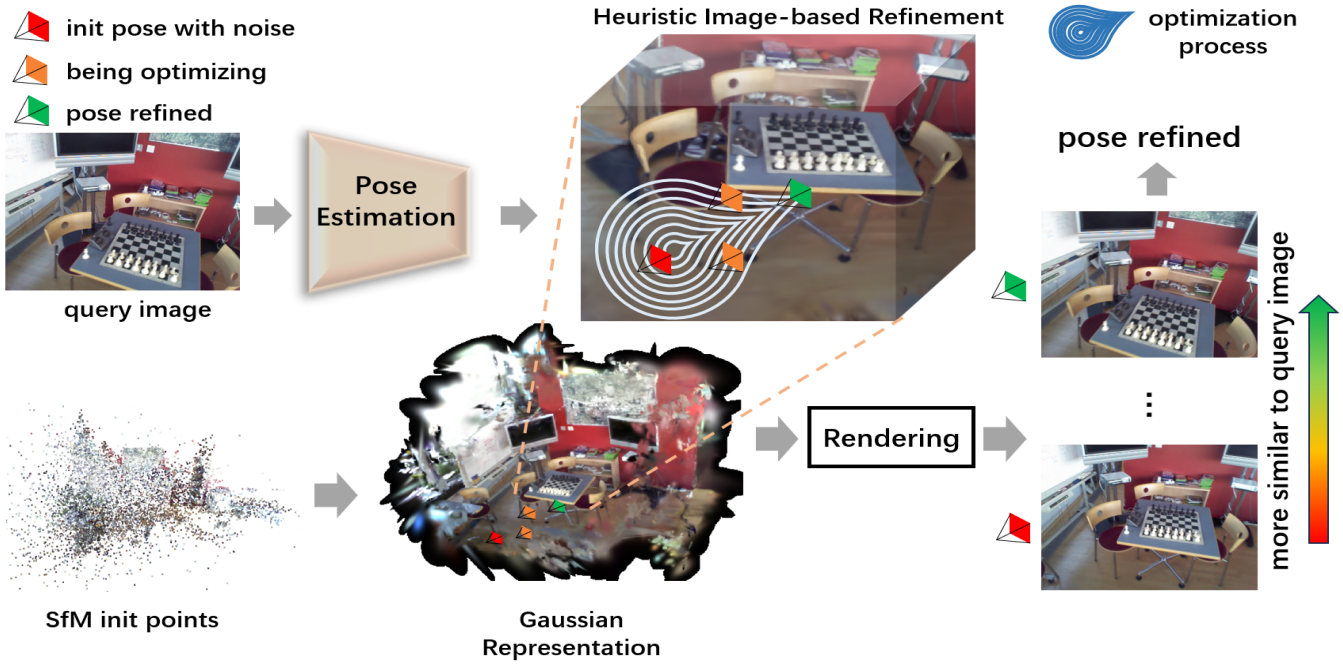

Fig. 2. Overview of HGSLoc. Coarse pose estimates are generated by a pre-trained pose estimator, while high-quality reconstructed scenes are obtained through Gaussian densification. The rendered image of the coarse pose in the scene differs significantly from the query image. After applying the heuristic optimization algorithm, the rendered image aligns much more closely with the query image, resulting in a more accurate pose estimate.

sum of the actual cost of the current node and the length of the path to the child node, and the estimated cost as the difference value between the rendered image and the query image corresponding to the pose of the current node:

$$g(n_{child}) = g(n_{current}) + 1 \tag{7}$$

$$h(n_{child}) = \Sigma |I_q - I_{nchild}| \tag{8}$$

Where the I_q represents the current query image and I_{nchild} represents the rendering image of current child node.

The heuristic function effectively guides the algorithm toward the optimal pose, ultimately identifying the pose that produces a rendered image most similar to the query image. We provide the pseudo-code for the algorithm's implementation in Tab. I. In this pseudo-code, OpenList is used to store nodes awaiting expansion, while ClosedList contains nodes that have already been expanded.

TABLE I

HEURISTIC POSE OPTIMIZATION STRATEGY

Heuristic Algorithm
while openList is not empty:
1. pop top node with $min(f(n))$ from openList.
2. if top is destination node:
break
3. closeList.push(top)
4. for each child node of top:
if child in closeList:
continue
computes the $cost_{tentative}$ from the start node to child.
if child not in openList:
$g(n_{child}) = g(n_{current}) + 1$
$h(n_{child}) = \Sigma \lvert I_q - I_{nchild} \rvert$
openList.push(child)
elif $cost_{tentative} < g(n_{child})$:
$g(n_{child})$ = tentative cost
heap adjustments

IV. EXPERIMENT

In this section, we compare and analyze the coarse pose with the optimized pose, including pose accuracy and precision.

A. Implementation

The deep learning framework employed in this work is PyTorch [29]. Each scene is reconstructed using 3D Gaussian Splatting (3DGS) with 30,000 training iterations, running on RTX 4090 GPUs. For the 7Scenes datasets, we adopt the SfM ground truth (GT) provided by [30].

B. Datasets, Metrics and Baselines

a) Datasets: We evaluated our method on two public datasets: 7scenes and Deep Blending. In the case of the 7scenes datasets [31], [32], the official test lists were used as query images, while the remaining images were utilized for training. For the Deep Blending dataset, we specifically selected the drjohnson and playroom scenes, and we constructed a test image set following the 1-out-of-8 approach suggested by Mip-NeRF [33].

b) Evaluation Metrics: We show the median rotation and translation error, and also provide the ratio of pose error within 1cm/1°.

TABLE II

WE PRESENT THE RESULTS OF COMPARISON EXPERIMENTS ON THE 7SCENES DATASET, HIGHLIGHTING THE MEDIAN TRANSLATION AND ROTATION ERRORS (CM/°) OF THE POSE RELATIVE TO THE GROUND TRUTH (GT) POSE FOR VARIOUS METHODS ACROSS SEVEN SCENES. THE BEST RESULTS ARE INDICATED IN BOLD. "NRP" REFERS TO NEURAL RENDER POSE ESTIMATION.

	Method	chess	fire	heads	office	pumpkin	redkitchen	stairs	Avg.↓[cm/°]
APR	Marepo [8]	1.9/0.83	2.3/0.91	2.2/1.27	2.8/0.93	2.5/0.88	3.0/0.99	5.8/1.50	2.9/1.04
SCR	ACE [10]	0.6/0.18	0.8/0.31	0.6/0.33	1.1/0.28	1.2/0.22	0.8/**0.20**	2.9/0.81	1.1/0.33
NRP	HR-APR [27]	2.0/0.55	2.0/0.75	2.0/1.45	2.0/0.64	2.0/0.62	2.0/0.67	5.0/1.30	2.4/0.85
	NeRFMatch [28]	0.9/0.3	1.3/0.4	1.6/1.0	3.3/0.7	3.2/0.6	1.3/0.3	7.2/1.3	2.7/0.70
	Marepo+HGSLoc	1.5/0.68	1.4/0.62	1.5/0.92	2.7/0.80	1.8/0.46	2.2/0.63	4.8/1.34	2.3/0.78
	ACE+HGSLoc	**0.5/0.17**	**0.6/0.25**	**0.5/0.29**	**1.0/0.25**	**1.1/0.21**	**0.7/0.20**	**2.8/0.69**	**1.0/0.29**

c) Benchmark: Our approach builds on an initial coarse pose estimation. For the APR [1]–[8] framework, we have selected the widely recognized Marepo [8] method as the benchmark for comparison. Similarly, for the SCR [9]–[11] framework, we have chosen the classical ACE [10] method as the benchmark for comparison.

C. Analysis of results

a) 7scenes dataset: For the 7Scenes dataset, we evaluate the performance of Marepo [8] and ACE [10] after incorporating HGSLoc. Tab. II demonstrates that our method effectively reduces the error in the coarse pose estimates obtained from both Marepo and ACE. Compared to other NRP methods, our approach achieves results with smaller relative pose errors. Furthermore, Tab. III presents the ratio of query images with relative pose errors of up to 1 cm and 1°, showing significant improvements after applying the HGSLoc framework. This indicates that our method efficiently optimizes cases involving small relative pose errors, further enhancing accuracy.

TABLE III

WE PRESENT THE AVERAGE PERCENTAGE OF POSE ERRORS WITHIN 1 CM AND 1° ON THE 7SCENES DATASET. "NRP" DENOTES NEURAL RENDER POSE ESTIMATION.

	Methods	Avg.↑[1cm,1°]
APR	Marepo [8]	6.2
SCR	ACE [10]	53.7
NRP	Marepo+HGSLoc	19.1
NRP	ACE+HGSLoc	59.1

b) DB dataset: We selected two scenes, "playroom" and "drjohnson," for testing. For both the Marepo [8] and ACE [10] methods, we observed that the coarse pose errors were significantly large. This may be attributed to the higher complexity of the DB dataset compared to the 7Scenes datasets, as well as the limited training data, which may have prevented model convergence. Consequently, we utilized an alternative method (HLoc) that leverages point clouds to obtain an initial pose estimate and compared the results. As shown in Tab. IV, the improvement from boosting is not pronounced, likely due to the high image quality of the DB dataset, which already provided relatively accurate preliminary poses with the HLoc framework. To better demonstrate the effectiveness of our pose optimization method, Tab. V introduces various levels of step noise, making the visualization results more intuitive.

TABLE IV

WE PRESENT THE MEDIAN TRANSLATION AND ROTATION ERRORS (CM/°) FOR BOTH THE INITIAL ESTIMATED POSE AND THE OPTIMIZED POSE RELATIVE TO THE GT POSE.

	init error	refine error
playrroom	0.7/0.060	0.6/0.059
drjohnson	0.3/0.055	0.3/0.054

TABLE V

WE SHOW THE MEDIAN TRANSLATION AND ROTATION ERROR (M/°) FOR THE POSES WITH NOISE AND FOR THE POSES AFTER OPTIMIZATION. (Q2, T1) DENOTES THE INTRODUCTION OF NOISE AT THE PERCENTILE OF QVEC, DECILE OF TVEC, AND THE REST IS THE SAME.

(a) playroom

	noise error	refine error	tvec↑	qvec↑
q2, t1	0.81/7.79	0.33/2.83	59.3%	63.7%
q2, t2	0.31/8.42	0.16/1.81	48.4%	78.5%
q3, t3	0.03/0.81	0.02/0.26	33.3%	67.9%

(b) drjohnson

	noise error	refine error	tvec↑	qvec↑
q2, t1	0.68/7.81	0.15/1.87	77.9%	76.1%
q2, t2	0.33/7.86	0.13/2.21	60.6%	71.9%
q3, t3	0.03/0.72	0.01/0.21	66.7%	70.8%

As shown in Tab. VI, to further demonstrate the effectiveness of our method, we compare it with an alternative joint optimization strategy [25]. For this comparison, a noise level of 1×10^{-3} granularity is introduced to the initial pose. Our method employs heuristic optimization based on high-quality scene reconstruction obtained through the 3DGS [14] method, whereas the alternative strategy jointly optimizes both the scene reconstruction and the initial pose [25].

c) Qualitative Analysis: By inputting the pose into the 3D reconstructed scene, we generate a rendered image that visualizes the pose. Each query image corresponds to the GT pose, and the discrepancy between the estimated pose and the GT pose is reflected in the rendered images from various viewpoints. To better observe this error and the improvement achieved through our optimization method, we

TABLE VI

WE SHOW THE MEDIAN TRANSLATION AND ROTATION ERROR (M/°) FOR HEURISTIC OPTIMIZATION AND JOINT OPTIMIZATION STRATEGIES.

	init error	joint error	heuristic error
playrroom	0.03/0.81	0.02/0.42	0.02/0.26
drjohnson	0.03/0.72	0.02/0.47	0.01/0.21

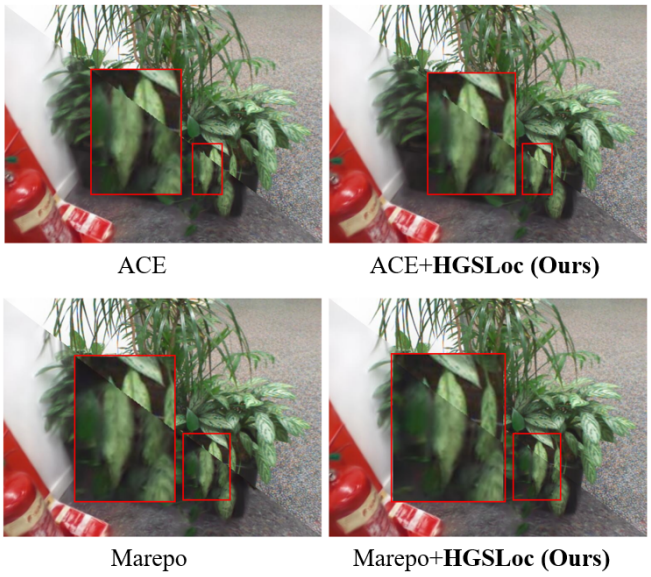

Fig. 3. HGSLoc demonstrates a significant optimization effect on the coarse poses obtained using the ACE and Marepo methods. Each subimage is divided by a diagonal line: the rendered image from the pose is shown in the bottom left part, while the GT image is shown in the top right part. The rendered images corresponding to the ACE and Marepo methods exhibit substantial misalignment with the GT images. To facilitate a clearer comparison, we provide a zoomed-in view of the image, highlighted within the red box.

select viewpoints with significant accuracy improvements for qualitative analysis. Fig. 3 demonstrate that, when using our framework on the 7Scenes datasets, the rendered images more closely match the GT images. Fig. 4illustrates the results of applying our framework to noisy poses in the DB dataset, showing that our method effectively refines the original pose, resulting in rendered images that closely resemble the GT images.

d) Ablation study: In our method, we use the sum of pixel-by-pixel differences as the heuristic function. To demonstrate the effectiveness of this heuristic function, Tab. VII compares the results obtained using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) as alternative heuristic functions. Higher values of PSNR and SSIM indicate better image quality and structural similarity, whereas we would like to see them take the opposite number as the value of the heuristic function is as small as possible. To illustrate the impact of different heuristic functions more clearly, we applied these comparisons to the DB dataset, which introduces significant noise.

$$h_1(n_{child}) = 100 - PSNR(I_q, I_{nchild}) \tag{9}$$

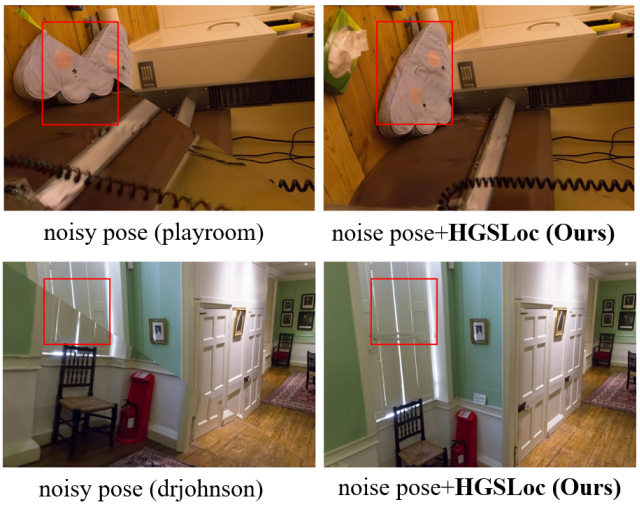

Fig. 4. Each subimage is divided by a diagonal line, with the image rendered by the estimated pose on the lower left and the GT image on the upper right. The diagonal lines in the optimized comparison image appear less distinct, reflecting improved alignment with the GT image. HGSLoc demonstrates its effectiveness in refining pose estimation, achieving precise values while mitigating the impact of band noise.

$$h_2(n_{child}) = 1.0 - SSIM(I_q, I_{nchild}) \tag{10}$$

TABLE VII

WE SHOW THE MEDIAN TRANSLATION AND ROTATION ERROR (M/°) FOR POSES WITH NOISE AND FOR POSES AFTER OPTIMIZATION USING DIFFERENT HEURISTIC FUNCTIONS.

	noise error	H(Sum of Diff)	H(PSNR)	H(SSIM)
playrroom	0.81/7.79	0.33/2.83	0.76/6.29	0.87/6.83
drjohnson	0.68/7.81	0.15/1.87	0.60/6.61	0.65/7.59

V. CONCLUSIONS

In this study, we propose a lightweight, plug-and-play visual localization optimization framework that combines heuristic refinement strategy with 3D reconstruction to significantly enhance pose estimation accuracy, achieving SOTA performance on two datasets. Compared to NeRF-based neural rendering localization methods [20], the proposed approach demonstrates superior rendering speed and enhanced localization accuracy. Through the integration of well-designed heuristic functions, the method efficiently optimizes and rapidly reduces errors in coarse localization estimations. Our modular approach not only reduces reliance on complex neural network training, enhancing the algorithm's flexibility and practicality, but also demonstrates robust performance in noisy environments, facilitating rapid convergence and higher accuracy. This robustness ensures that the method performs consistently across various platforms and data qualities. In summary, the integration of heuristic refinement strategy with 3D Gaussian distribution offers a novel and effective solution for visual localization, providing a valuable reference for the

development and optimization of future visual localization systems.

REFERENCES

[1] 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.

[2] A. Kendall and R. Cipolla, "Modelling uncertainty in deep learning for camera relocalization," in *2016 IEEE international conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 4762–4769.

[3] ——, "Geometric loss functions for camera pose regression with deep learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5974–5983.

[4] B. Wang, C. Chen, C. X. Lu, P. Zhao, N. Trigoni, and A. Markham, "Atloc: Attention guided camera localization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, 2020, pp. 10 393–10 401.

[5] S. Chen, Z. Wang, and V. Prisacariu, "Direct-posenet: Absolute pose regression with photometric consistency," in *2021 International Conference on 3D Vision (3DV)*. IEEE, 2021, pp. 1175–1185.

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

[7] Y. Shavit, R. Ferens, and Y. Keller, "Learning multi-scene absolute pose regression with transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2733–2742.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] P. Jiang, G. Pandey, and S. Saripalli, "3dgs-reloc: 3d gaussian splatting for map representation and visual relocalization," *arXiv preprint arXiv:2403.11367*, 2024.

[16] C. Liu, S. Chen, Y. Bhalgat, S. Hu, Z. Wang, M. Cheng, V. A. Prisacariu, and T. Braud, "Gsloc: Efficient camera pose refinement via 3d gaussian splatting," *arXiv preprint arXiv:2408.11085*, 2024.

[17] Y. Hiasa, Y. Otake, M. Takao, T. Matsuoka, K. Takashima, A. Carass, J. L. Prince, N. Sugano, and Y. Sato, "Cross-modality image synthesis from unpaired data using cyclegan: Effects of gradient consistency loss and training data size," in *Simulation and Synthesis in Medical Imaging: Third International Workshop, SASHIMI 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Proceedings 3*. Springer, 2018, pp. 31–41.

[18] V. Leroy, Y. Cabon, and J. Revaud, "Grounding image matching in 3d with mast3r," *arXiv preprint arXiv:2406.09756*, 2024.

[19] V. Bulitko, N. Sturtevant, J. Lu, and T. Yau, "Graph abstraction in real-time heuristic search," *Journal of Artificial Intelligence Research*, vol. 30, pp. 51–100, 2007.

[20] S. Chen, Y. Bhalgat, X. Li, J.-W. Bian, K. Li, Z. Wang, and V. A. Prisacariu, "Neural refinement for absolute pose regression with feature synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 987–20 996.

[21] X. Song, J. Zheng, S. Yuan, H.-a. Gao, J. Zhao, X. He, W. Gu, and H. Zhao, "Sa-gs: Scale-adaptive gaussian splatting for training-free anti-aliasing," *arXiv preprint arXiv:2403.19615*, 2024.

[22] J. C. Lee, D. Rho, X. Sun, J. H. Ko, and E. Park, "Compact 3d gaussian representation for radiance field," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 719–21 728.

[23] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang, "4d gaussian splatting for real-time dynamic scene rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 310–20 320.

[24] Z. Zhu, Z. Fan, Y. Jiang, and Z. Wang, "Fsgs: Real-time few-shot view synthesis using gaussian splatting," *arXiv preprint arXiv:2312.00451*, 2023.

[25] Z. Fan, W. Cong, K. Wen, K. Wang, J. Zhang, X. Ding, D. Xu, B. Ivanovic, M. Pavone, G. Pavlakos, Z. Wang, and Y. Wang, "Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds," 2024.

[26] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, "Dust3r: Geometric 3d vision made easy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 697–20 709.

[27] C. Liu, S. Chen, Y. Zhao, H. Huang, V. Prisacariu, and T. Braud, "Hr-apr: Apr-agnostic framework with uncertainty estimation and hierarchical refinement for camera relocalisation," *arXiv preprint arXiv:2402.14371*, 2024.

[28] Q. Zhou, M. Maximov, O. Litany, and L. Leal-Taixé, "The nerfect match: Exploring nerf features for visual localization," *arXiv preprint arXiv:2403.09577*, 2024.

[29] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[30] E. Brachmann, M. Humenberger, C. Rother, and T. Sattler, "On the limits of pseudo ground truth in visual camera re-localisation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6218–6228.

[31] B. Glocker, S. Izadi, J. Shotton, and A. Criminisi, "Real-time rgb-d camera relocalization," in *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2013, pp. 173–179.

[32] 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.

[33] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 5855–5864.