Evaluating Gaussian Splatting Maps for Absolute Visual Localization of UAVs

Felix Rocco Matzke[1], Hans-Gerd Maas[1], Anette Eltner[1]

[1] Institute of Photogrammetry and Remote Sensing, TUD Dresden University of Technology, Germany
- (felix_rocco.matzke, hans-gerd.maas, anette.eltner)@tu-dresden.de

Keywords: UAV Navigation, Absolute Visual Localization, Gaussian Splatting, Structure from Motion

Abstract

Localization within a global reference frame is critical for the safe operation of UAVs. It is typically realized through GNSS measurements, however when signals are jammed, spoofed, occluded or reflected, this approach can lead to errors or fail. As most UAVs are equipped with cameras, absolute visual localization using georeferenced map representations offers a promising alternative. The recent invention of Gaussian Splatting introduces new opportunities for this task, leveraging real-time rendering from novel views to establish 2D-3D correspondences for pose estimation. In this work, we investigate the use of Gaussian Splatting maps for absolute visual localization of UAVs with a particular focus on geometric accuracy and its impact on the accuracy of position estimation. Through experiments with real-world data, we show that an initialization with dense Structure from Motion point clouds does not improve geometric accuracy compared to sparse initialization under the current training scheme. Additionally, constraining the position optimization of Gaussian Splats shows potential for improved pose estimation but introduces challenges during training. Despite these limitations, our results demonstrate the feasibility of Gaussian Splatting-based absolute visual localization for UAVs.

1. Introduction

The navigation of most uncrewed aerial vehicles (UAV) relies on the well-established fusion of Global Navigation Satellite System (GNSS) and Inertial Measurement Unit (IMU) data. Inertial sensors provide estimates of orientation, velocity, and position by integrating angular rate and specific force measurements over time at high frequency. However, measurement noise and sensor biases in the low-cost inertial sensors commonly used on UAVs cause integration errors that lead to rapid drift within a few seconds. These drift effects are mitigated by GNSS measurements, which are less frequent but provide absolute position and velocity information in a global reference frame. When combined with real-time kinematic (RTK) corrections, GNSS can achieve an accuracy of a few centimeters, effectively constraining the long-term drift of the inertial solution. However, GNSS measurements can be prone to errors due to multipath effects, jamming, and spoofing, or may become temporarily unavailable as a result of signal occlusions. Consequently, alternative approaches for absolute pose estimation are required. Given that most UAVs are equipped with cameras, absolute visual localization constitutes a viable solution to estimate the camera (and thereby the UAV) pose within a global reference frame.

Visual localization aims to estimate the camera pose by matching an acquired (query) image to a pre-existing map representation of the environment. When this map is georeferenced, the resulting pose can be expressed in a global coordinate frame, a process commonly referred to as absolute visual localization. Existing approaches differ mainly in the type of geodata employed for map representation, which may be either two-dimensional (2D) or three-dimensional (3D). Common 2D geodata sources include satellite imagery, orthophotos, street-view images, and vector data (Wang et al., 2025). These representations require comparatively less memory but typically allow only 3-DoF pose estimation (2D position and heading). In contrast, 3D map representations such as city models, point clouds, meshes, voxel grids or orthophotos in combination with a digital surface model enable full 6-DoF pose estimation. Recently, Neural Radiance Fields (NeRFs) and Gaussian Splatting have been explored as alternative 3D map formats due to their intrinsic coupling of 2D image appearance with 3D scene geometry and their ability to render novel views, thereby mitigating view-dependence issues (Jiang et al., 2025).

In this work, we investigate the use of Gaussian Splatting maps for absolute visual localization of UAVs. Specifically, we focus on creating 2D Gaussian Splatting maps (Huang et al., 2024) and leveraging their real-time rendering capabilities to generate RGB and depth images from novel views. These rendered images enable the establishment of 2D-3D correspondences, which are then used for pose estimation. Our primary objective is to understand the limitations of the current 2D Gaussian Splatting implementation, particularly in terms of geometric fidelity, and how these limitations affect pose estimation accuracy. Recent approaches have attempted to improve geometric fidelity by initializing Gaussian Splats with dense LiDAR data (Jiang et al., 2025). However, these methods lack a direct comparison between sparse and dense initialization, as well as a detailed geometric analysis of the resulting Gaussian Splatting maps. To address this gap, we utilize sparse and dense Structure from Motion (SfM) point clouds for initialization and analyze the geometric accuracy of the resulting models. In summary, this study aims to answer the following research questions:

- How does the position of 2D Gaussian Splats change during optimization?
- How do these positional changes impact pose estimation accuracy?
- Can a dense point cloud initialization improve geometric fidelity and pose estimation performance?

https://doi.org/10.5194/isprs-archives-XLIX-B2-2026-1183-2026 | © Author(s) 2026. CC BY 4.0 License.

2. Related Work

The introduction of 3D Gaussian Splatting (Kerbl et al., 2023) revolutionized the field of novel view synthesis by transitioning from implicit neural representations of radiance fields to an explicit representation using volumetric primitives called 3D Gaussians. Unlike NeRF (Mildenhall et al., 2020), which relies on computationally expensive ray-marching, 3D Gaussians can be rasterized efficiently, enabling much higher rendering speeds and faster training. Since its inception, numerous variants of 3D Gaussian Splatting have been developed to address specific challenges. For example, Scaffold-GS (Lu et al., 2023) introduces anchors based on a sparse voxel grid derived from an SfM point cloud and generates neural Gaussians depending on the viewing angle, thereby reducing redundancy and improving adaptation to varying scene complexities. Another notable extension, 2D Gaussian Splatting (Huang et al., 2024), flattens 3D Gaussians into 2D disks, which better align with object surfaces and address the multi-view inconsistency problem. Additionally, adaptations for large-scale scenes have been proposed (Kerbl et al., 2023).

Gaussian Splatting is increasingly incorporated into Simultaneous Localization and Mapping (SLAM) systems and visual localization pipelines, with various adaptations demonstrating its potential for navigational tasks. PINGS (Pan et al., 2025) combines Gaussian Splatting radiance fields and signed distance fields into a novel neural point map representation for LiDAR-Visual SLAM, thereby improving both geometric accuracy and photorealism. In VINGS-Mono (Wu et al., 2025), 2D Gaussian Splatting is used as a map representation in the backend of a monocular inertial SLAM framework. The novel view synthesis capabilities are leveraged for loop-closure detection, and an additional score manager evaluates, manages, and prunes 2D Gaussian Splats, enabling operation in outdoor, kilometer-scale environments.

In the domain of visual localization, Gaussian Splatting has been explored in systems such as Splat-Nav (Chen et al., 2024), which successfully utilizes precomputed Gaussian Splatting maps for real-time pose localization and planning in small indoor environments. However, Gaussian Splatting has been rarely applied for absolute visual localization in outdoor environments. The 3DGS-ReLoc (Jiang et al., 2025) framework integrates LiDAR and camera data into a precomputed 3DGS map, which serves as a foundation for visual localization using monocular camera images. While experiments conducted on autonomous driving scenes demonstrate the potential of using 3D Gaussian Splatting as scene representation, the geometric accuracy of the 3D Gaussian Splatting map was not evaluated. It therefore remains unclear whether initializing Gaussian Splats with LiDAR data improves geometric accuracy, especially given that the positions of the Gaussian Splats are optimized during training. Furthermore, with the emergence of newer Gaussian Splatting variants, 3D Gaussian Splatting may no longer be optimal for absolute visual localization, particularly in scenarios requiring high geometric accuracy.

To the best of our knowledge, no previous work has dealt with the absolute visual localization of UAVs using Gaussian Splatting maps in outdoor environments. In this paper, we address this gap by creating 2D Gaussian Splatting maps based on UAV imagery of a rural town and analyzing their geometric accuracy and pose estimation capabilities.

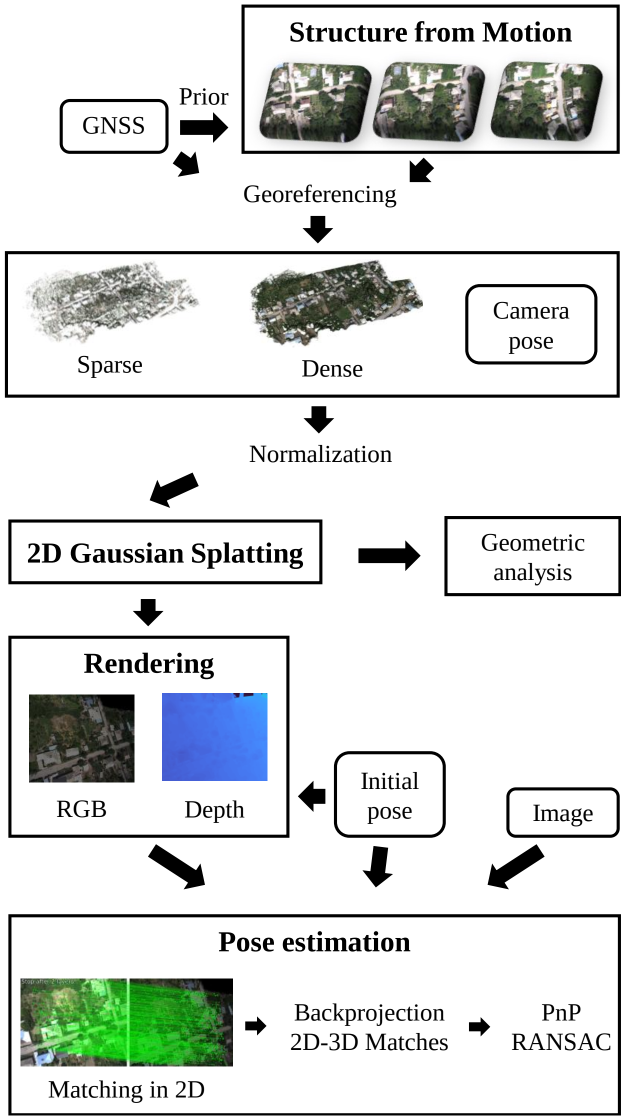


Figure 1. Overview of the methodology.

3. Methods

The methodology employed in this work is outlined in Figure 1. The process begins with the creation of the required inputs for training the Gaussian Splatting map using Structure from Motion. To explore the impact of initialization strategies, we perform both sparse and dense reconstructions. GNSS data is incorporated as a soft prior during the sparse reconstruction process and is used to georeference the inputs, ensuring alignment with real-world coordinates. To facilitate optimization, the inputs are normalized, simplifying the training process. For pose estimation, given an initial pose, RGB and depth images are rendered from the Gaussian Splatting map. These images are then used in a pose estimation pipeline, which includes 2D feature matching, backprojection to establish 2D-3D correspondences, and a final pose refinement step using PnP RANSAC.

3.1 2D Gaussian Splatting

In our approach, we employ 2D Gaussian Splatting (Huang et al., 2024), which models 2D disks in 3D space, as opposed to traditional 3D Gaussian Splatting that uses volumetric primitives. Each 2D Gaussian Splat (GSplat) is defined by a central point $\mathbf{p}_k$, which anchors the GSplat in 3D space, and two orthonormal tangential vectors $\mathbf{t}_u$ and $\mathbf{t}_v$. The cross product of these tangential vectors defines the normal vector $\mathbf{t}_w = \mathbf{t}_u \times \mathbf{t}_v$, and together, $\mathbf{t}_u$, $\mathbf{t}_v$, and $\mathbf{t}_w$ form a rotation matrix $\mathbf{R} = [\mathbf{t}_u, \mathbf{t}_v, \mathbf{t}_w]$, which represents the orientation of the GSplat in 3D space. The spread of the 2D GSplat is controlled by a scaling vector $(\mathbf{s}_u, \mathbf{s}_v)$ that can be arranged as a diagonal matrix $\mathbf{S} = \mathrm{diag}(\mathbf{s}_u, \mathbf{s}_v, 0)$. Combining these parameters yields the transformation matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{RS} & \mathbf{p}_k \\ \mathbf{0} & 1 \end{bmatrix},$$

which defines the 2D GSplat in world space. Similar to 3D GSplats, each 2D GSplat is also associated with an opacity α and a view-dependent appearance $\mathbf{c}$.

To ensure multi-view consistent rendering, 2D Gaussian Splatting uses an explicit ray-splat intersection algorithm based on homogeneous plane equations (Huang et al., 2024). Furthermore, depth distortion and normal consistency regularization are applied to improve the geometric accuracy of the reconstruction. A point cloud reconstructed using UAV images models the scene surface. Since 2D Gaussian splatting aims to improve surface reconstruction, this should contribute to improved pose estimation compared to 3D Gaussian splatting. Furthermore, consistency across multiple views is crucial for the accurate rendering of images from new viewpoints.

Building upon the mathematical formulation of 2D Gaussian Splatting presented in the preceding section, we describe the training methodology adopted for absolute visual localization. The training pipeline requires two primary inputs: a set of known camera poses and an initial point cloud used to seed the 2D Gaussian primitives. To facilitate the use of 2D Gaussian Splatting models for absolute camera pose estimation, both the camera poses and the initial point cloud were expressed in the Earth-centered, Earth-fixed (ECEF) coordinate frame. However, operating directly in the ECEF frame introduced numerical difficulties during optimization, owing to the large absolute coordinate magnitudes inherent to this reference system. Although such instabilities can in principle be mitigated through hyperparameter adjustments (for instance, by modifying the gradient threshold) this approach would require scene-specific tuning and therefore lacks generalizability. A more robust solution was to normalize both the point cloud coordinates and the camera translation vectors prior to training to a scene-centered reference frame, which is constrained between -1 and 1. This normalization substantially stabilized the optimization process, enabling the use of default training hyperparameters and allowing direct adoption of learning rates and regularization factors established in prior work.

A central challenge in applying 2D Gaussian Splatting to camera pose estimation concerns the drift of GSplat centers $\mathbf{p}_k$ during training. Under the standard 2D Gaussian Splatting training protocol, the positions of the Gaussian primitives are continuously updated, however, such positional drift is undesirable in the pose estimation context, where the GSplats should correspond as closely as possible to the original scene structure. To alleviate this drift, we investigated two complementary strategies. First, we initialized the 2D Gaussian Splatting model from a dense point cloud, thereby providing comprehensive spatial coverage of the scene and reducing the incentive for large positional updates. Second, we attenuated the position learning rate

Figure 2. Locations of the training images used in our experiments.

to impose a soft constraint on GSplat displacement throughout training. To assess the geometric fidelity of the trained models, which is a critical prerequisite for accurate pose estimation, we evaluated the reconstructed GSplat geometry against the reference dense point cloud. In line with our primary objective of characterizing the suitability of the 2D Gaussian Splatting framework for pose estimation, no further modifications to the model architecture or training procedure were introduced.

3.2 Absolute visual localization

To estimate the camera pose, we leverage a pipeline inspired by (Elias et al., 2019) and similar to (Chen et al., 2024, Jiang et al., 2025), which renders images to establish 2D-3D correspondences between an image and the captured scene. Based on the trained 2D Gaussian Splatting model, camera parameters and an initial camera position (derived, for example, from measurements by other sensors or a previously known state and an estimated motion), we render both RGB and depth images. Using these rendered RGB images, we extract features and perform feature matching with the corresponding real RGB images. To detect key points, we use SuperPoint (DeTone et al., 2018), and the robust matching of key points between the real and rendered images is performed using LightGlue (Lindenberger et al., 2023). Once feature correspondences are established, the matched 2D points from the rendered image are backprojected into the 3D camera coordinate system using the corresponding depth values from the rendered depth image and the camera intrinsics. Leveraging initial camera extrinsics, the 3D points are then transformed into the world coordinate system. This process establishes a set of 3D-2D correspondences between the backprojected 3D points and the matched 2D points from the real RGB image. To determine the camera pose, we solve the Perspective-n-Point (PnP) problem using RANSAC to robustly handle outliers. Specifically, we use a solver based on Levenberg-Marquardt optimization to refine the pose estimation.

4. Experiments

This section presents the experiments conducted to evaluate the use of 2D Gaussian Splatting maps for absolute visual localization of UAVs. We begin by describing the data used and the preprocessing steps, including the creation of sparse and dense reconstructions with SfM and the incorporation of GNSS data for georeferencing. Next, we detail the training of different 2D Gaussian Splatting models and analyze their geometric

Model	Number of Gaussian Splats		Iterations	PSNR	Time
	Initial	Final			
2DGS_Sparse	154,563	3,676,541	(7000) 30,000	(27.742) 31.543	59:53
2DGS_Dense	9,629,581	5,140,024	(7000) 30,000	(28.486) 32.283	1:10:02
2DGS_Dense_Pos	9,629,581	5,989,063	7,000	27.676	18:27

Table 1. Reconstruction results for different initialization strategies and training settings.

accuracy. Finally, these models are employed in various pose estimation scenarios to assess the capabilities and limitations of the current approach, providing insights into the potential and challenges of Gaussian Splatting for UAV localization.

4.1 Dataset

For real-world experiments, we used a sequence from the MARS-LVIG dataset (Li et al., 2024) that was captured over a rural town in Armenia using a DJI M300 RTK platform equipped with a customized hardware-synchronized sensor suite. Specifically, we use RTK-GNSS and IMU orientation data collected by the DJI UAV, as well as images from the mounted global-shutter RGB camera, taken from a subset of the sequence covering a flight duration of 76 seconds (between the GPS timestamps 1658133189.385730076 and 1658133265.382839481). This subset consists of straight flight segments in opposite directions with a turn in between (see Figure 2), recorded at a height of approximately 80 meters above the ground and at a flight speed of 8 m/s. The camera captured images at a frame rate of 10 Hz. For the scene and camera pose reconstruction with COLMAP (Schönberger and Frahm, 2016), we used only one image per second, resulting in a training dataset for the Gaussian Splatting models that consists of 77 images. To test the pose estimation from new views, we selected images captured between the training samples, also at a rate of one image per second, which yielding a testing dataset of 76 images.

The camera positions for our experiments were calculated using a combination of RTK GNSS data and the provided calibration parameters between the RTK GNSS, LiDAR, and camera. To account for the UAV’s motion, the IMU orientation data from the DJI Matrice 300 RTK was used, which showed no noticeable drift; however, the exact method used for internal drift correction is unknown. The required UAV orientation was therefore estimated at the time of image capture by interpolating the IMU data using quaternion spherical linear interpolation (slerp). However, a key challenge arose due to the unknown boresight alignment between the IMU of the DJI platform and the LiDAR, which served as a reference point for the calibrations in the custom sensor suite. Based on the dataset documentation, we assumed no misalignment beyond an axis swap. Additionally, we treated the system as rigid, despite the use of rubber-damping balls in the sensor suite’s mounting, which could cause relative displacements of the sensor unit of slightly less than 1.5 cm (Li et al., 2024). These assumptions inevitably lead to inaccuracies in the calculated camera position.

The required camera pose and scene reconstruction for the training of Gaussian Splatting models were performed using COLMAP (Schönberger and Frahm, 2016). We incorporated the provided camera intrinsics, including focal length, principal point, and distortion parameters (Li et al., 2024) into the sparse reconstruction and used the calculated camera positions as soft constraints (priors). By extending the existing sparse reconstruction, we determined the camera positions for the test images. The camera positions were also used to georeference the sparse training reconstruction, resulting in a final median alignment error of 0.25 m between the camera positions reconstructed by SfM and the calculated camera positions. The SfM reconstruction produced a sparse point cloud consisting of 154,563 points and a dense point cloud with 9,629,581 points. Since the Gaussian splatting models are based on the SfM reconstructions, the camera poses reconstructed by SfM were used as ground truth in our experiments.

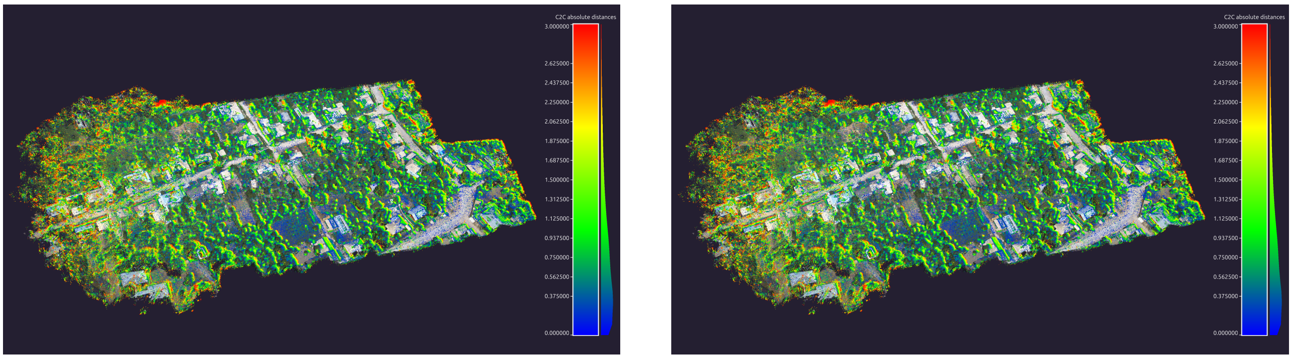
In addition, we used the provided LiDAR-to-camera calibration, the IMU orientation data, and the assumed LiDAR-to-IMU alignment to calculate the camera orientations. We observed an offset between the calculated orientations and the reconstructed orientations provided by the SfM process. This was to be expected, since the unknown misalignment between the IMU and the LiDAR mentioned in the previous paragraph leads to inaccuracies, and the poses reconstructed using SfM cannot be determined without error. We tried to improve the alignment by estimating an offset orientation using the Kabsch algorithm (Kabsch, 1976) and applied it to the calculated orientations. Despite this correction, the difference between the calculated and SfM-derived camera poses remained significant, with a median angular difference of 6.75° and a maximum difference of 11.47°. Due to this discrepancy, the original idea of using these calculated camera orientations as a reference for the pose estimation experiments was discarded. Nevertheless, they provide an interesting test case when used as an initial pose estimate in our experiments.

4.2 Gaussian Splatting maps

To evaluate the geometric accuracy of 2D Gaussian Splatting maps and the resulting impact on camera pose estimation, we trained three different models following different initialization and training strategies. For efficiency, we adhered to the default image rescaling during training, resizing images to a width of 1.6K pixels. The first model was initialized using a sparse point cloud (2DGS_Sparse), while the second model was initialized using a dense point cloud (2DGS_Dense). Both models were trained with default training parameters to investigate whether the dense initialization improves the geometric accuracy of the scene representation and how this change impacts the final pose estimation. The third model (2DGS_Dense_Pos) was also initialized using the dense point cloud, but the position optimization was constrained by setting the initial and final position learning rates to values 100 times smaller than the default. Additionally, the depth distortion regularization parameter was set to 1 to further restrict the positioning of 2D GSplats.

The reconstruction results after training are summarized in Table 1. The model initialized with the dense point cloud achieved an improved Peak Signal-to-Noise Ratio (PSNR) compared to the model initialized with the sparse point cloud, but training took 10 minutes longer. As expected, the Adaptive Density Control mechanism led to an increase in the number of

https://doi.org/10.5194/isprs-archives-XLIX-B2-2026-1183-2026 | © Author(s) 2026. CC BY 4.0 License.



(a) 2DGS model initialized by a sparse SfM point cloud (2DGS_Sparse)

(b) 2DGS model initialized by a dense SfM point cloud (2DGS_Dense)

Figure 3. Absolute cloud-to-cloud distances between the 2DGS models and a dense SfM point cloud of the scene

2D GSplats when the sparse point cloud was used for initialization. When the dense point cloud was used, the final number of 2D GSplats was higher than in the sparse initialization case, but compared to the initial number, it was significantly reduced. For the third model, which was initialized with the dense point cloud and trained with constrained position learning rates, a GPU memory error occurred during the training at 14,450 iterations due to a massive increase in the number of GSplats. To address this, we decided to evaluate the model at the 7,000-iteration checkpoint, which is a standard checkpoint in Gaussian Splatting literature (Kerbl et al., 2023). For a fair comparison, the PSNR values for the other models were additionally reported at 7,000 iterations. The PSNR for the constrained model (2DGS_Dense_Pos) was lower than that of the unconstrained dense initialization model. Interestingly, the number of Gaussians, while still higher than for the 2DGS_Dense model after 30,000 iterations, was already greatly reduced after 7,000 iterations. Another important factor to consider in the further analysis is that, according to the default training settings, the normal regularization parameter was not changed from 0 to 0.05 until after 7,000 iterations.

The geometric accuracy of the trained models was analyzed by visualizing the absolute cloud-to-cloud distances with respect to the dense point cloud (see Figure 3). In both the sparsely and densely initialized models, deviations from the dense point cloud are visible throughout the scene. The positional shift of the GSplats is particularly evident for trees and rooftops. Interestingly, the magnitude and distribution of these differences are very similar between the two models, suggesting that the dense initialization does not provide a significant advantage in terms of geometric fidelity. This observation is supported by the median values listed in Table 2, which show a shift of 0.88 m for the model initialized with the sparse point cloud (2DGS_Sparse) and 0.85 m for the model initialized with the dense point cloud (2DGS_Dense). Despite the dense initialization, the positional gradients appear to be high, causing the initial 2D Gaussian splats to either be removed or undergo significant shifts during training. Interestingly, the shift of the 2D Gaussian splats is predominantly horizontal. If the position learning rate is constrained (2DGS_Dense_Pos), the cloud-to-cloud distance decreases significantly, even if normal regularization has not been applied.

4.3 Pose estimation

To evaluate the performance of our camera pose estimation approach, we conducted four distinct experiments with each of the three trained 2D Gaussian Splatting maps. The first experiment

Model	**Median C2C distance** [m]
2DGS_Sparse	0.88
2DGS_Dense	0.85
2DGS_Dense_Pos	0.05

Table 2. Median absolute cloud-to-cloud distances for the trained 2D Gaussian Splatting maps

aimed to determine the maximum achievable pose estimation accuracy by estimating the camera pose for each training image using a perfect initial pose derived from the SfM reconstruction (SfM_Train). This provided a baseline for the best-case performance of the models. The second experiment assessed the generalization capability of the approach to slightly modified views. Specifically, we estimated the camera pose for test images captured at similar distances and view angles to the ground but with horizontal shifts. For this, we used the exact initial pose for the test images (SfM_Test).To simulate a more realistic scenario, the third experiment estimated the camera pose for the test images using calculated camera poses as the initial pose (Calc_Test). While the translation of these initial poses was close to the exact pose, angular differences of up to 11.5 degrees were introduced. This experiment provided insights into the performance of the pose estimation under varying view angles and with roughly known initial camera orientations. Finally, the fourth experiment evaluated the robustness of the approach when the initial pose was only accurate to GPS-level precision. To simulate this, we added random noise values between -10 and 10 (drawn from a uniform distribution) to each coordinate of the initial pose (SfM_Test pos-10). This experiment tested the ability of the models to handle initial position errors, which can be encountered in real-world UAV applications. The results of our experiments are presented in Table 3.

In the best-case scenario (SfM_Train), where the initial pose is perfect, the maximum achievable translational accuracy is approximately 2–3 cm, with a rotational accuracy of around 0.02° for the 2DGS_Sparse and 2GS_Dense models. However, when estimating poses for new views with horizontal shifts (SfM_Test), the accuracy drops significantly, with errors increasing by roughly a factor of 10. Interestingly, while the position and orientation accuracy of the 2DGS_Dense_Pos model is lower in the SfM_Train scenario, this model appears to be more robust to slight view changes compared to the other models. For the 2DGS_Sparse and 2DGS_Dense models, the errors in this scenario are very similar, suggesting that dense initialization does not provide a clear advantage in generalization

Model	Init	MTE [m]	RMSE-T [m]	MRE [°]	RMSE-R [°]	init MTE [m]
2DGS_Sparse	SfM_Train	0.03	0.04	0.02	0.03	
	SfM_Test	0.28	0.29	0.11	0.13	
	Calc_Test	0.50	0.58	0.19	0.24	
	SfM_Test_pos-10	3.24	3.91	0.67	1.20	9.55
2GS_Dense	SfM_Train	0.03	0.03	0.02	0.03	
	SfM_Test	0.29	0.30	0.11	0.14	
	Calc_Test	0.49	0.60	0.22	0.31	
	SfM_Test_pos-10	3.28	3.66	0.58	0.81	9.73
2DGS_Dense_Pos	SfM_Train	0.06	0.08	0.05	0.07	
	SfM_Test	0.26	0.27	0.12	0.13	
	Calc_Test	0.46	0.54	0.21	0.26	
	SfM_Test_pos-10	3.77	3.98	0.64	0.79	9.60

Table 3. Quantitative comparison of the pose estimation accuracy for different 2D Gaussian Splatting maps and initial scenarios based on the mean translational error (MTE), RMSE for the translation (RMSE-T), the mean rotational error (MRE) and the RMSE for the rotation (RMSE-R).

to new views. In more realistic scenarios, where the initial pose provides only a rough orientation estimate (Calc_Test), the translational errors increase to approximately 0.5 m, with rotational errors ranging between 0.2° and 0.3° across all models. When the initial camera position is only known with GPS-level accuracy (SfM_Test pos-10), the translational error increases drastically to 3–4 m. However, given the initial pose error of more than 9.5 m, the error is reduced by a factor of 3, demonstrating the system's ability to refine the pose significantly. The rotational error also increases but remains mostly under 1°, indicating reasonable robustness to large initial pose errors. Overall, the results provide no clear evidence that dense initialization, using the same optimization process as in the original implementation (Huang et al., 2024), improves pose estimation accuracy. The fact that the 2DGS_Dense_Pos model was trained for only 7,000 iterations, exhibits lower reconstruction quality, and yet achieves comparable performance in pose estimation underscores the potential of position constraints.

5. Discussion

Our experiments show that initializing 2D Gaussian Splatting with a dense point cloud does not fundamentally improve geometric accuracy under the current training scheme. Although constraining the position learning rate showed promise for enhancing pose estimation accuracy, it simultaneously introduced training instabilities, exposing a trade-off between photorealism and geometric accuracy. Further research is needed to develop training and densification strategies specifically tailored to pose estimation. In addition to pruning steps during the training phase, a pruning of the initial point cloud should also be considered to achieve a more accurate representation of the scene.

Lever-arm and boresight alignment errors introduce inaccuracies in our dataset. However, since we initialize our Gaussian Splatting models using camera poses and point clouds derived from SfM and further use SfM camera poses as a reference for ground truth, these alignment inaccuracies can be neglected for the evaluation of localization results. Nevertheless, for real-world UAV operations, such errors must be carefully accounted for to ensure accurate and reliable performance. Our pose estimation results confirm the feasibility of absolute visual localization for UAVs using Gaussian Splatting maps, provided that the initial pose estimate is sufficiently accurate. This underscores the importance of tight integration with other sensors and the incorporation of past measurement data into future system designs. Notably, an initial positional error was found to be more detrimental to the pose estimation accuracy than initial angular errors.

Several open research questions remain. Future work should investigate the robustness of the proposed approach to illumination changes. In addition, our experiments included images taken from a similar distance from the ground as the images used to train the Gaussian Splatting maps. Future research should examine performance under drastic changes in scene scale, such as those that occur during takeoff and landing maneuvers. For real-world operations, existing strategies must be reviewed and refined in order to extend the methodology to large-scale environments.

6. Conclusion

In summary, we demonstrate the potential of using 2D Gaussian Splatting maps for absolute visual localization of UAVs. Through real-world experiments, we highlighted the limitations of the current implementation of 2D Gaussian Splatting, particularly regarding geometric fidelity. Initial attempts to improve geometric accuracy, such as dense initialization of 2D Gaussian Splats and constraining their positional optimization, did not yield significant improvements under the current training scheme. These findings emphasize the need for further refinement of the optimization and densification processes to improve the final pose estimation. In the future, we will address the limitations of our current localization approach by focusing on closer integration with other sensors and exploring adaptations for large-scale environments.

Acknowledgment

This research was funded by the German Research Foundation (DFG) under grant number 508591287 as part of the RTG-AirMetro (2947/1).

For this project, the authors used You.com and GWDG's Academic Cloud AI Services for programming tasks and in the preparation of the manuscript to improve linguistic quality and readability. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

Chen, T., Shorinwa, O., Bruno, J., Yu, J., Zeng, W., Nagami, K., Dames, P., Schwager, M., 2024. Splat-Nav: Safe Real-Time Robot Navigation in Gaussian Splatting Maps. https://arxiv.org/abs/2403.02751.

DeTone, D., Malisiewicz, T., Rabinovich, A., 2018. SuperPoint: Self-Supervised Interest Point Detection and Description. https://doi.org/10.48550/arXiv.1712.07629.

Elias, M., Kehl, C., Schneider, D., 2019. Photogrammetric Water Level Determination Using Smartphone Technology. *The Photogrammetric Record*, 34(166), 10.1111/phor.12280.

Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S., 2024. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers '24*, 1–11. https://arxiv.org/abs/2403.17888.

Jiang, P., Bendapudi, N., Saripalli, S., Pandey, G., 2025. 3DGS-Loc: 3D Gaussian Splatting for Map Representation and Visual Localization. 10.1115/1.4069346.

Kabsch, W., 1976. A Solution for the Best Rotation to Relate Two Sets of Vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5), 10.1107/S0567739476001873.

Kerbl, B., Kopanas, G., Leimkuehler, T., Drettakis, G., 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4), 10.1145/3592433.

Li, H., Zou, Y., Chen, N., Lin, J., Liu, X., Xu, W., Zheng, C., Li, R., He, D., Kong, F., Cai, Y., Liu, Z., Zhou, S., Xue, K., Zhang, F., 2024. MARS-LVIG Dataset: A Multi-Sensor Aerial Robots SLAM Dataset for LiDAR-visual-inertial-GNSS Fusion. 43(8), 10.1177/02783649241227968.

Lindenberger, P., Sarlin, P.-E., Pollefeys, M., 2023. LightGlue: Local Feature Matching at Light Speed. https://doi.org/10.48550/arXiv.2306.13643.

Lu, T., Yu, M., Xu, L., Xiangli, Y., Wang, L., Lin, D., Dai, B., 2023. Scaffold-GS: Structured 3D Gaussians for View-Adaptive Rendering. https://doi.org/10.48550/arXiv.2312.00109.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., Ng, R., 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. https://arxiv.org/abs/2003.08934.

Pan, Y., Zhong, X., Jin, L., Wiesmann, L., Popović, M., Behley, J., Stachniss, C., 2025. PINGS: Gaussian Splatting Meets Distance Fields within a Point-Based Implicit Neural Map. https://doi.org/10.48550/10.48550/arXiv.2502.05752.

Schönberger, J. L., Frahm, J.-M., 2016. Structure-from-Motion Revisited. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 4104–4113. https://ieeexplore.ieee.org/document/7780814/.

Wang, Z., Shi, D., Qiu, C., Jin, S., Li, T., Qiao, Z., Chen, Y., 2025. VecMapLocNet: Vision-based UAV Localization Using Vector Maps in GNSS-denied Environments. 225, 10.1016/j.isprsjprs.2025.04.009.

Wu, K., Zhang, Z., Tie, M., Ai, Z., Gan, Z., Ding, W., 2025. VINGS-Mono: Visual-Inertial Gaussian Splatting Monocular SLAM in Large Scenes. *IEEE Transactions on Robotics*, 41, 10.1109/TRO.2025.3613536.