Towards autonomous robotic structure inspection with dense-direct visual-SLAM

Diego Navarro, Raphael Antoine, Ezio Malis, Philipe Martinet

Towards autonomous robotic structure inspection with dense-direct visual-SLAM

1st Diego Navarro
ENDSUM & ACENTAURI teams
Cerema & Inria
Sophia-Antipolis, France
diego-navarro.tellez@inria.fr

2nd Raphael Antoine
ENDSUM team
Cerema Normandie-Centre
Rouen, France
raphael.antoine@cerema.fr

3rd Ezio Malis
ACENTAURI team
Centre Inria d'Université Côte d'Azur
Sophia-Antipolis, France
ezio.malis@inria.fr

4th Philipe Martinet
ACENTAURI team
Centre Inria d'Université Côte d'Azur
Sophia-Antipolis, France
philippe.martinet@inria.fr

***Abstract*—We present a comprehensive framework based on direct Visual Simultaneous Localization and Mapping (V-SLAM) to observe a vertical coastal cliff. The precise positioning of data measurements (such as ground-penetrating radar) is crucial for environmental observations. However, in GPS-denied environments near large structures, the GPS signal can be severely disrupted or even unavailable. To address this challenge, we focus on the accurate localization of drones using vision sensors and SLAM systems. Traditional SLAM approaches may lack robustness and precision, particularly when cameras lose perspective near structures.**

We propose a new framework that combines feature-based and direct methods to enhance localization precision and robustness. The proposed system operates in two phases: first, a SLAM phase utilizing a stereo camera to reconstruct the environment from a distance sufficient to benefit from a wide field of view; second, a localization phase employing a monocular camera. Experiments conducted in realistic simulated environments demonstrate the system's ability to achieve drone localization within 15-centimeter precision, surpassing existing state-of-the-art approaches.

***Index Terms*—Dense mapping, Precise localization, UAV positioning, Structure inspection**

I. Introduction

The diagnosis of vertical cliffs is crucial to prevent infrastructures damage and life losses. The example of the coastal cliffs of Normandy is particularly striking, with an increase in landslides due to the extreme events experienced in the region in recent years (drought and heavy rains).

Drones have emerged as an optimal solution for environmental inspection, due to their ability to access harsh environments and to carry out diverse instruments. In recent times, ground-penetrating radars (GPR) have been developed to image the interiors of structures and detect internal defects. A critical requirement for radar usage is to maintain close proximity to the surface under examination, typically within a few tens of centimeters, to optimize the signal-to-noise ratio.

Often, vertical cliffs and weather conditions can block or distort GPS signals, leading to localization errors or even signal loss. These interferences can be problematic when a drone is used close to the structure. An alternative approach involves using a digital twin to aid localization and navigation tasks. In scenarios requiring localization, Simultaneous Localization and Mapping (SLAM) methods involving LiDAR or cameras (visual SLAM, e.g. V-SLAM) are commonly employed. In this paper, we use V-SLAM because it is compatible with UAVs and meets their flight constraints. This choice is motivated by the need for robustness and adaptability tailored to our particular GPR use case.

One the one hand, direct methods, based on SfM (Structure from motion) are capable of locating the agent's location and exhibit robustness in low-texture regions. However, the loss of perspective due to the drone's proximity to the wall results in a decrease in the point density, potentially undermining the system's ability to localize the agent. On the second hand, feature-based methods, such as ORB SLAM 2 and 3 are known for their low computational cost and have become the leading solutions in V-SLAM [1]. Prior work has explored the creation of dense maps from sparse Key Frames, as presented in [2]. Zhang and Shu integrated a dense mapping component into the ORB-SLAM2 [3] framework. Stereo data were used to generate a point cloud, which was successfully merged with overlapping regions of the point cloud. However, this work did not tackle the challenge of localization in close inspection scenarios. In fact, the main limitation of ORB SLAM is the loss of perspective due to the proximity of the structure. The change of scale in the incoming image flow results in new descriptors that are not present in the map, leading to a decrease in the localization precision or even a complete loss of localization.

This paper goes further in the use of dense maps, in multi-session localization and navigation. It introduces a V-SLAM framework specifically designed for close structure inspection, capable of localization even in challenging scenarios where other systems might falter, while maintaining precision

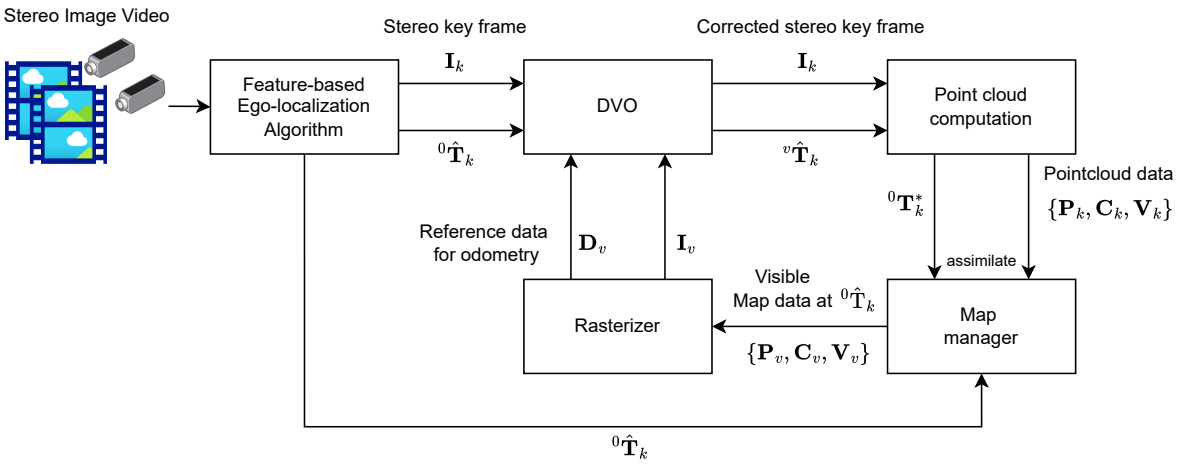

Fig. 1. Workflow for the mapping step. Most computations are currently computed offline, but the system can be integrated into a real-time system with proper optimization.

throughout the trajectory. Our V-SLAM system combines the speed of feature-based methods with the accuracy of direct methods and the richness of dense maps. During the mapping phase, we propose creating a dense map with stereo images and an initial ego-pose estimation using a faster feature-based method. It must be noticed that the system is designed to be modular and any feature-based ego-localization method for pose estimation can be used. Subsequently, during the second phase, we aim to achieve precise localization using a monocular camera. The primary objective of this approach is to enhance localization accuracy when the agent is far from the mapping trajectory, leading to a multi-session optimized map that not only improves precision but also aids in navigation tasks.

II. Proposed Method

Our method involves a two-step approach. In the initial step, a stereo camera captures images during a mapping flight, utilizing a feature-based method for ego-pose estimation. The system's modular design allows for flexibility in choosing various ego-localization methods.

The resulting dense map serves dual purposes: aiding mission planning and enhancing localization in the second step. In this subsequent stage, precise localization is achieved using a monocular camera, addressing weight and energy constraints posed by attached measurement equipment such as radar and thermal cameras.

A. Global Map Generation & Update

During the mapping flight the ego-localization system is expected to select a set of keyframes composed of a pair of stereo images $\mathbf{I}_k$ and an estimated pose ${}^0\hat{\mathbf{T}}_k$. The selection criteria may vary, but the frames shall respect some conditions for photogrammetry reconstruction such as the overlap and the coverage of the target area. Once the keyframes have been collected, the system carries the pointcloud registration and fusion process.

The registration process (see Figure 1) is carried out by aligning the incoming image $\mathbf{I}_k$ with a render of the global map $\mathbf{I}_v$ and its depth map $\mathbf{D}_v$ at the current estimated pose ${}^0\hat{\mathbf{T}}_k$. The alignment procedure leverages the map points observable at the current pose $\mathbf{M}_v$.

Inspired by the work of [4], the framework uses the EWA(Elliptical Weighted Average) volume splatting method [5] to render (the process of converting the pointcloud into an image) $\mathbf{I}_v$ and $\mathbf{D}_v$ without need of a mesh.

Specifically, we use an implementation of the EWA method tailored to render low density point-clouds.

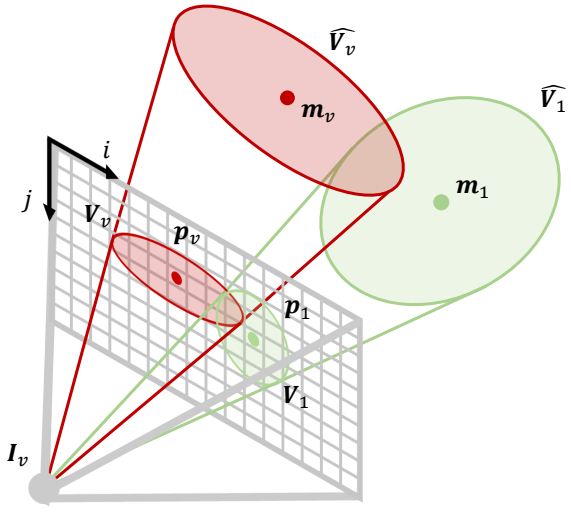

Fig. 2. The position incertitudes in the camera frame $\hat{\mathbf{V}}_1, \ldots, \hat{\mathbf{V}}_v$(characterized by the covariance) are mapped as ellipsoid regions in the image space $\mathbf{V}_1, \ldots, \mathbf{V}_n$ through perspective projection linearised around the center of the ellipsoids $\mathbf{m}_1, \ldots, \mathbf{m}_v$ whose coordinates in the image plane are $\mathbf{p}_1, \ldots, \mathbf{p}_v$. Then the colors are merged using the EWA method.

As the first Keyframe is used as reference and therefore directly saved into the map, no rendering is needed. However, for the next frames the map manager will collect the visible map points $\mathbf{M}_v$, their colors $\mathbf{C}_v$ and its respective covariance matrices $\mathbf{V}_v$ into a *virtual* surface that will serve for rendering. The surface is then projected into the image space to produce $\mathbf{I}_v$ and $\mathbf{D}_v$ effectuating the sum of each color value $\mathbf{c}_v$ pondered by its correspondent weight in the ellipsoid w_{tab}.

Then, the DVO module [6] computes the pose ${}^v\tilde{\mathbf{T}}_k$ that represents the estimated error between the estimated pose and the real pose. The corrected pose of the Keyframe can now be composed from the estimated pose and the error estimation. The corrected pose is then used to align the incoming pointcloud data with the global map data. This map is build as a graph of *surfaces* $\mathbf{S}_s$, composed of a set of 3D points $\mathbf{M}_s$, a set of colors $\mathbf{C}_s$, a set of covariance matrices $\mathbf{V}_s$ and a pose ${}^0\hat{\mathbf{T}}_s^*$. This serves to model a fictional camera to organize the data in a grid corresponding to an image plane $\mathbf{I}_s$. This eases the process of updating the map and the selection of points to render.

Following the correction by odometry, the new pointcloud is computed and corrected based on its quality determined by the scale coherence with $\mathbf{D}_v$. If the correction fails to improve the quality of the incoming depth map, the system rejects the new pointcloud. Accepted point clouds are then processed by the map manager which decides either to generate a new surface or update an existing one. This decision is done based in multiple criteria. The first criterion is the distance from the last reference surface, a straightforward measure ensuring proximity for alignment in image space. The second one is the overlap of the reference surface and the incoming point-cloud. If visible area of the keyframes changes significantly, a new surface is generated to maximize information gathering.

The third criterion is a custom measure based on the coverage of interesting areas. Even with substantial overlap, is still possible to neglect some interesting areas observed just once or twice during the mapping session. In this case, the system will compute a salient map of the incoming point-cloud and compare it with the reference surface saliency map projected in the same image space. If particularly interesting areas are not covered by the reference surface, the system will generate a new one.

New key surfaces $\mathbf{S}_s$ are initialized using information from surrounding surfaces and the incoming point clouds. Point matches are obtained through alignment in image space by projecting 3D points in the same pixel grid(see Figure 3). In the empty cells the grid assimilates the new values. In populated cells, the new value is averaged with existing ones if the distance is within an acceptable range. Otherwise, the old value is overwritten. During the update of the points, the system updates the covariance matrix of the position $\mathbf{V}_s$ with the new values. The fusion of the color information $\mathbf{C}_s$ is effectuated in the LAB color space with the intention to better capture the perceptual changes in the color. After processing all keyframes, the system will generate a surface graph that will be used in the localization step.

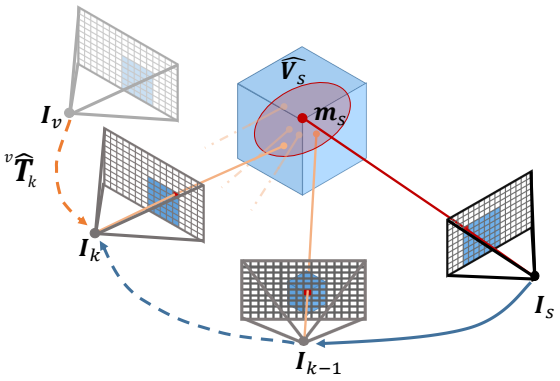

Fig. 3. Point cloud alignment is achieved through the minimisation of distance between the renderings at the estimated positions $\mathbf{I}_v$ and the keyframe images $\mathbf{I}_k$. The set of points that fall in the same pixels of $\mathbf{I}_s$ are used to create the covariance matrix of the map points $\mathbf{m}_s$.

B. Localization

In this step a scanning path is defined either by an algorithm or by the user. In the current state of the framework, the user shall choose the shape of the scan path. After defining the waypoints of the scanning path, the mission control module will generate a set of virtual keyframes to assist during the scanning operation.

First, the map produced by the offline registration is loaded. Then, the system generates a set of frames with a co-visibility criteria (see Figure 4).

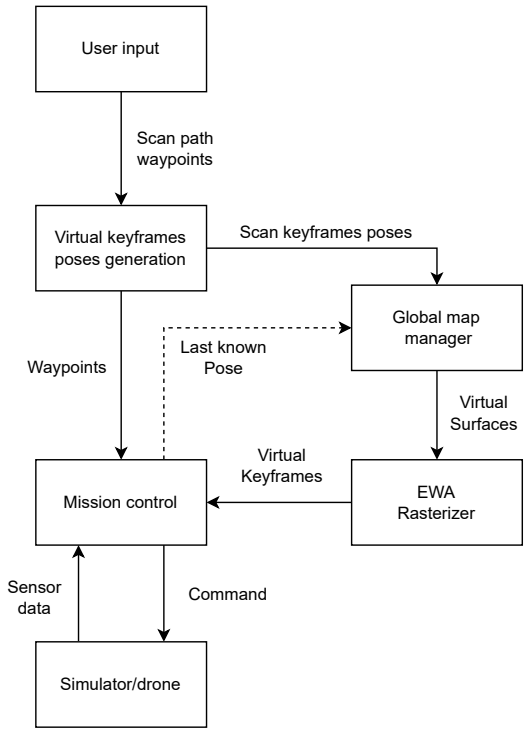

Fig. 4. The global map manager will take any pose and generate a virtual surface that will be used to generate a depth and color images necessary for the odometry.

First, the map produced by the offline registration is loaded. Then the system generates a set of frames with a co-visibility criteria (see Figure 4). By projecting the scanning path's lines onto the virtual image plane, the visible distance covered by the virtual camera's field of view (FOV) is computed. With this information the next pose of the virtual camera is chosen by overlap criterion. In the current implementation, the path generator computes the portion of the parametric line between the two waypoints that is covered by the virtual camera FOV. The selection of the point along this line is based on achieving the specified percentage of overlap. This chosen point becomes the image center for the next virtual image frame The path generator maps this pixel to the 3D space and adds an offset to get the next pose of the virtual camera. The offset place the camera at a distance behind the scan trajectory (in the sense of the expected orientation), so the image covers a larger area than the expected area to be covered by the real camera. This approach ensures that incoming images along the scan path can be reliably tracked by at least one virtual key frame.

During the scan path, the mission control module picks the closest virtual keyframe for localization. The selection is based on an estimation of the current pose. If the DVO module fails to converge, a new virtual KeyFrame is generated at the last known pose to restart the tracker. Since the DVO module estimates the pose from the incoming images to the reference virtual keyframes, the system can recover the global pose without accumulating errors. The localization is carried out using the same direct odometry method used during the global map generation. However, is worth to mention that the DVO module can localize the agent with different camera parameters during the mapping step. This can be done by configuring the rasterizer to emulate the camera parameters of the incoming images.

III. Simulations with realistic data

The data used for the experiments was collected in a simulated environment. Since our color profile for the color fusion is too simple we decided to carry the simulation in a lighting-controlled environment. The scene is composed by a segment of a cliff model done using the Agisoft metashape software [7].

The observed site is the Sainte-Marguerite-sur-Mer cliff (Normandy), monitored in the framework of the Defhy3geo project. A drone acquired geo-referenced images, allowing the generation of a 3D model of the cliff. The effective visible area is equivalent to a 60x20 meters wall with non-structured texture. This map proves the performance of the system in low texture environments.

To benchmark our method's performance, we use ORB-SLAM3 [8] in localization mode as a baseline. In this mode, the system focuses solely on agent localization, sacrificing mapping capabilities for computational efficiency.

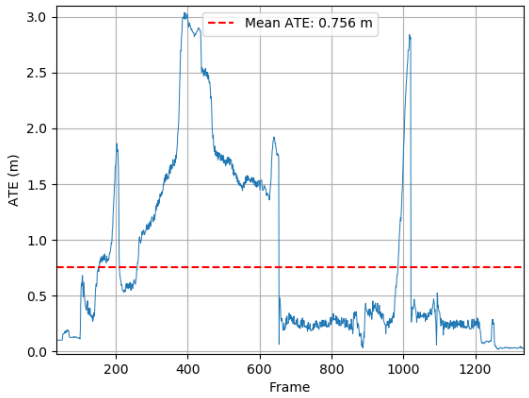

Fig. 5. ORB-SLAM3 absolute translation error (ATE) in meters during the scan path with stereo camera. Being stereo the best configuration, the closest ORB-SLAM3 could get to the structure is 5m from the structure, i.e. 8m from the mapping path.

We first tried to perform the scan mission with the ORB-SLAM3 in monocular mode, but the system gets lost when the agent is too close to the mapping trajectory. Then we tried to perform the mission in stereo mode, which is more robust to perspective loss (see Figure 5). The closest the camera could get to the structure in this mode was 5m.

A. Results

The first result of our method is the dense map. The Figure 6 presents the dense map generated by our method. It is capable of capturing most of the structure of the environment and the color information. Comparing the generated dense map with the ground truth, 50% of the points have less than 10cm of error, 36% are under 20cm and 10% under 30cm. In other terms, the mean error of the points in the map is 11cm.

The second result of our method is the capability to generate virtual frames. In the Figure 7 we can see a virtual frame generated by our method. As we can see, our method is capable of generating useful images even with low density pointclouds.

Our method's third contribution is global pose estimation, demonstrating the system's resilience in adverse conditions. In the Figure 8 we can see the dense map with the set of virtual reference poses. During this experiment the agent was capable of maintaining the global pose estimation with mean ATE of 0.16m at 2m from the structure.

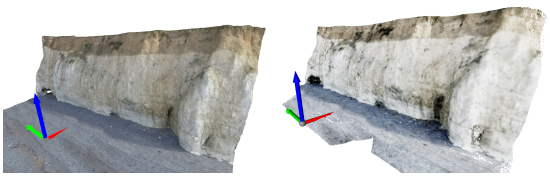

Fig. 6. The map generated through offline registration can assist subsequent navigation tasks. On the left is the ground-truth model in the simulation, and on the right is the reconstructed model.

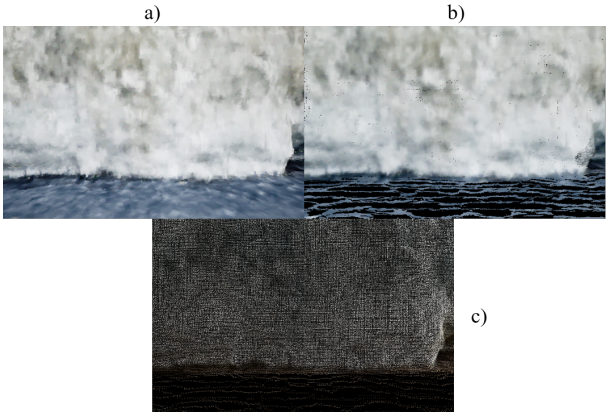

Fig. 7. Virtual frames. a) Image seen by the camera, b) image rendered from the dense model and c) centers of the kernels used for rendering.

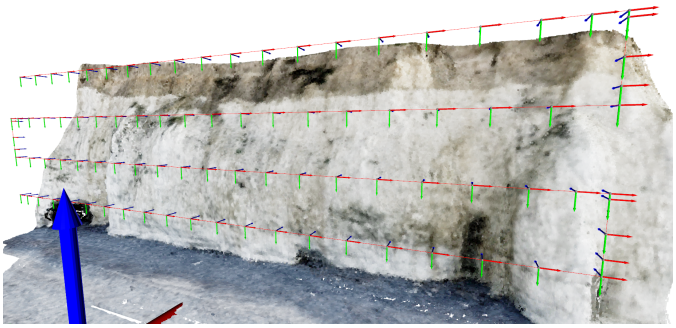

Fig. 8. Path followed by the agent during the scan mission, a set of virtual poses is generated (represented by 3D gizmos). The rasterizer renders an image for each one and generates the virtual reference Keyframes.

Pose estimation precision depends on stereo pair-derived point cloud quality and dense odometry module accuracy. While odometry is fairly precise, depth estimation can be improved. The significant error between frames 1300 to 1500 in Figure 9 results from poor map quality in that area. This region is infrequently observed during mapping, causing the stereo disparity algorithm to under-perform. Despite this, our method exhibits a similar precision to ORB-SLAM3 but operates 4 meters closer to the structure (2m from the structure and 11m from the mapping path).

IV. Conclusion

In this paper, we introduce an innovative framework designed to tackle the challenge of localization close to structures, where traditional methods may falter. This versatile map proves invaluable not only for localization but also for mission control and cost-effective creation of digital twins for multi-session autonomous navigation.

A promising improvement would be changing the color-fusion profile to better capture the perceptual changes in the color. Another line of research is the exploitation of the dense map to generate a scanning path based on the regions of interest in the map. Since the surface information is stored in image-like data structures, image processing techniques

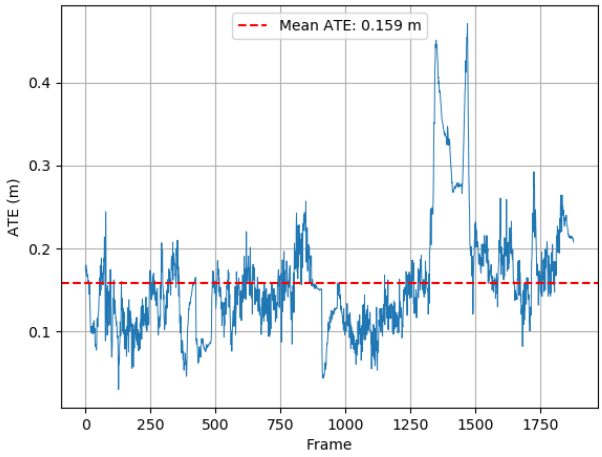

Fig. 9. Our method exhibits consistent Absolute Translation Error (ATE) behavior over 1900 frames, showing stability at various distances across multiple experiments. However, a notable shift occurs when the model is observed in regions with extremely low-texture conditions.

like saliency-maps can be used to define the most interesting camera positions to improve the precision of the localization.

While we have presented our findings in a simulation with realistic data, we anticipate that our method will prove equally effective in real-world scenarios, particularly those characterized by richer information and texture diversity.

Acknowledgment

This work is part of the ROAD-AI projet funded by Cerema and Inria. It is also part of the ”DEFHY3GEO” 2022–2024 regional project, co-funded by the Normandy County Council and European Union in the framework of the ERDF-ESF operationnal program 2014-2020.

References

[1] W. Chen, G. Shang, A. Ji, C. Zhou, X. Wang, C. Xu, Z. Li, and K. Hu, “An overview on visual slam: From tradition to semantic,” *Remote Sensing*, vol. 14, no. 13, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/13/3010

[2] B. Zhang and D. Zhu, “A Stereo SLAM System With Dense Mapping,” *IEEE Access*, vol. 9, pp. 151 888–151 896, 2021.

[3] R. Mur-Artal and J. D. Tardós, “ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras,” *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, Oct. 2016.

[4] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, “3d gaussian splatting for real-time radiance field rendering,” *ACM Transactions on Graphics (SIGGRAPH Conference Proceedings)*, vol. 42, no. 4, July 2023. [Online]. Available: http://www-sop.inria.fr/reves/Basilic/2023/KKLD23

[5] M. Zwicker, H. Pfister, J. van Baar, and M. Gross, “Ewa volume splatting,” in *Proceedings Visualization, 2001. VIS ’01.*, 2001, pp. 29–538.

[6] A. I. Comport, E. Malis, and P. Rives, “Real-time Quadrifocal Visual Odometry,” *The International Journal of Robotics Research*, vol. 29, no. 2, pp. 245–266, 2010. [Online]. Available: https://inria.hal.science/hal-00766839

[7] J. Crume, C. Crume, and B. Crume, “Metashape pro,” 2019.

[8] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM,” *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, Dec. 2021.