

Perception-aware Exploration for Consumer-grade UAVs

Svetlana Seliunina, Daniel Schleich, and Sven Behnke

Autonomous Intelligent Systems group, Computer Science Institute VI – Intelligent Systems and Robotics, University of Bonn

Abstract. In our work, we extend the current state-of-the-art approach for autonomous multi-UAV exploration to consumer-level UAVs, such as the DJI Mini 3 Pro. We propose a pipeline that selects viewpoint pairs from which the depth can be estimated and plans the trajectory that satisfies motion constraints necessary for odometry estimation. For the multi-UAV exploration, we propose a semi-distributed communication scheme that distributes the workload in a balanced manner. We evaluate our model performance in simulation for different numbers of UAVs and prove its ability to safely explore the environment and reconstruct the map even with the hardware limitations of consumer-grade UAVs.

Keywords: autonomous exploration, unmanned aerial vehicles

1 Introduction

Autonomous exploration is a rapidly developing field with numerous applications, and multi-UAV exploration holds significant potential. The use of several coordinating robots allows for a faster exploration of big environments, especially considering the limited battery life of a single quadrotor. Recently, many works were published on the topic of exploration planning for UAVs, but most of them assume that the UAV possesses a reliable state estimator, a depth sensor, and sufficient computational capabilities [16]. However, in our work, we aim to extend the current state-of-the-art approach for autonomous multi-UAV exploration to consumer-level UAVs, such as the DJI Mini 3 Pro. These only provide us with monocular images, and velocity and orientation measurements of limited precision (± 0.1 m/s, $\pm 0.1^\circ$).

Our pipeline consists of state and depth estimators, as well as an exploration planner, which extends the baseline to meet the limitations of these odometry and depth estimators. Our main modifications are connected with the viewpoint pairs selection process and with the path planning through them. After we ensure that our single-UAV exploration with consumer-grade DJI drones is successful, we extend our approach to a multi-UAV case. The pipeline for the multi-UAV exploration consists of several UAVs that perform the exploration independently from each other, with the planner from the single-UAV approach. However, we add the base node, which is responsible for coordinating UAVs, accumulating data about the explored space, and assigning workload.

The contributions of the paper are as follows:

- We propose a method of sampling and evaluating viewpoint pairs to produce images suitable for depth estimation. Additionally, we adapt the hierarchical exploration planner to viewpoint pairs instead of single viewpoints.
- We modify the trajectory planner to satisfy motion constraints by introducing a yaw-aware A* planner and a joint optimization step with rotation constraints.
- We introduce a semi-distributed multi-UAV exploration method, where the base node handles global planning and map updates due to the limited computational capabilities of consumer-grade UAVs.

2 Related Works

Exploration of unknown environments is a fundamental task, and one of the first steps is determining how to choose the best viewpoints. Most classical approaches can be divided into three groups: sampling-based, frontier-based, and a combination of the two. Sampling-based approaches randomly generate viewpoints and then evaluate them based on the information gain they provide, as in [11], [3], and [13]. This approach offers a way to directly determine the next best view, but it is computationally demanding. Frontier-based approach, on the other hand, is less computationally demanding, but does not take into account the information gain, and was used in [8], [1], and [4]. Many methods, such as [7], [6], and [2], combine frontier-based and sampling-based approaches to trade off computation time and information gain.

FUEL [18] is the pipeline for autonomous UAV exploration of the unknown environment. Their primary focus was on maintaining the high exploration rate while ensuring safe and energy-efficient movement of the UAV. The pipeline consists of two main parts: an incremental frontier information structure and a hierarchical planner, as well as a mapping module and a motion controller.

The authors introduce a concept of frontier information structure (FIS), which contains all the necessary information for exploration and can quickly be updated incrementally when new information arrives. After a new sensor measurement is received, they update the information about affected frontiers. The viewpoint poses are sampled in the cylindrical coordinate system around the average, and evaluated based on the coverage. The connection cost between each pair of clusters denotes the time lower bound when moving between the best viewpoints of each frontier. The motion is planned in a coarse-to-fine manner. Firstly, they create a global path that passes through all unknown regions. Then they create and refine the local path that passes through the desired viewpoints to ensure high information gain and a smooth path. And finally, they generate a safe and dynamically optimal motion of the quadrotor for it to reach the first viewpoint.

Later, the authors of FUEL extended their approach to the multi-quadrotor case with RACER [17]. In their work, they improved their autonomous exploration pipeline and introduced a communication protocol for multiple UAVs.

As a result, they ensure that all quadrotors explore distinct regions with only limited asynchronous communication between them.

The volumetric map is constantly subdivided into disjoint hgrid cells that are used as elementary task units for UAVs to explore. The main difference between the previous exploration planner and the new one is the addition of a global coverage path, which passes through all unexplored hgrid cells assigned to the robot. Cells are distributed between UAVs via pairwise interactions to minimize the length of coverage paths through them. Additionally, UAVs exchange not only their allocated grids, but also the trajectories and explored map chunks.

3 Exploration Using Consumer-grade UAVs

3.1 Single-UAV Exploration Method

Our autonomous exploration method is presented in Figure 1. The first part of the pipeline consists of state and depth estimators that process the UAV’s sensor data to produce the current odometry and depth image required for the exploration planner. In the second part, the exploration planner creates and updates the map, identifies frontiers and picks viewpoints, plans the trajectory, and sends velocity commands. All of these steps are performed on a ground control station, since the consumer-grade UAVs do not have sufficient onboard computing capabilities. In case of DJI UAVs, the generated velocity commands can be executed via the Mobile SDK as used in [12].

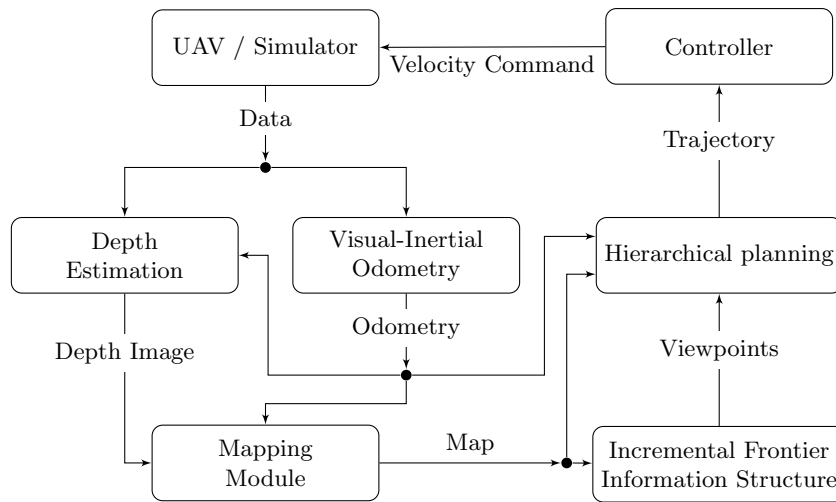


Fig. 1: Overview of the proposed pipeline for single-UAV exploration.

As the state estimator for our model, we use the visual-inertial odometry estimation model, which was proposed in [12]. The mapping module computes both the volumetric occupancy map and the Signed Euclidean Distance Field

(ESDF) as proposed in [5], as well as performs hgrid decomposition. The controller includes the feedback loop to account for the difference between sent and executed velocity commands, and its output is rounded to the resolution of the velocity readings.

Depth Estimation To avoid revisiting frontiers and to create a high-quality map, we decided not to use fast depth estimation pipelines that produce only sparse depth maps. Instead, we use the transformer-based deep learning depth estimator MAST3R [9], which produces dense depth maps. The model only accepts images as input and approximates camera intrinsics and extrinsics, which can result in scaling errors. However, since camera poses and parameters are known to us, we extract dense image correspondences before they are passed to the model aligner. From these, we estimate the corresponding 3D positions using the VIO poses and iterative DLT triangulation. Since depth estimation requires multiple different viewpoints, we can either send images to the depth estimator uniformly after some time has passed, or explicitly plan at least two views per frontier.

Viewpoint Planning The Frontier Information Structure (FIS) stores and incrementally updates information about frontiers, using principal component analysis to estimate their orientations. In the sampling step, we search for a pair of viewpoints from which the frontier can be reconstructed. We sample candidate poses uniformly around the edges of the frontier, with the yaw sampled around the direction towards the center. After filtering for sufficient FoV coverage, pairs of viewpoints V_1, V_2 are evaluated based on a heuristic reconstructability score h , which is inspired by [14]. It does not only consider overlap between the observed voxels C_1, C_2 but also maximal distance d , parallax α , and observation angles θ_1, θ_2 :

$$h(V_1, V_2, C_1, C_2) = w_1(\alpha)w_2(d)w_3(\alpha)w_4(\theta) \frac{|C_1 \cap C_2|}{|C_1 \cup C_2|}. \quad (1)$$

The parallax dependency of triangulation error and multi-view matchability are modeled as

$$w_1(\alpha) = \frac{1}{1 + e^{-k_1(\alpha - \alpha_{\min})}} \quad \text{and} \quad w_3(\alpha) = 1 - \frac{1}{1 + e^{-k_3(\alpha - \alpha_{\max})}}, \quad (2)$$

with k_1, k_3 determining the slope, α_{\min} the minimal parallax for successful triangulation and α_{\max} the maximal parallax for successful stereo matching. Additionally, w_2 favors close-up views and w_4 disfavors shallow observation angles:

$$w_2(d) = 1 - \min\left(1, \frac{d}{d_{\max}}\right), \quad w_4(\theta_1, \theta_2) = \min(\|\cos(\theta_1)\|, \|\cos(\theta_2)\|), \quad (3)$$

where d_{\max} is the maximal desired distance that depends on ray length limitations in the mapping module. After numerous tests, we chose parameters to be $k_1 = 32$, $k_3 = 8$, $\alpha_{\min} = \frac{\pi}{8}$, $\alpha_{\max} = \frac{\pi}{3}$ and $d_{\max} = 3.5$ m.

Close view poses with significantly different yaw angles are difficult to connect with rotation-constrained trajectories. Thus, we first sample viewpoints on lines parallel to the frontier. The corresponding yaw angles are chosen such that the frontier and line are both covered by the field of view (FoV). Thus, they can be traversed without rapid rotations. Only if no pair with a sufficient reconstruction heuristic is found, do we fall back to uniform sampling.

Trajectory Planning Exploration planning is divided hierarchically into three steps in a coarse-to-fine manner: global path planning, local viewpoint refinement, and trajectory generation. In global planning, a coverage path through all frontiers is generated by solving the Traveling Salesman Problem. To keep the search space small, we do not consider each viewpoint individually but represent each viewpoint pair by its midpoint. As in FUEL, connection costs are estimated as time lower bounds when moving between two frontiers. When restricting flight directions to the FoV, this significantly underestimates rotation costs. Thus, we additionally insert an intermediate yaw point $\gamma_k = \arctan \frac{(p_j - p_i)_y}{(p_j - p_i)_x}$.

After the global path is found, we locally refine viewpoints. Here, we have to refine viewpoints in one pair jointly because changing a viewpoint in one pair independently of its partner may result in a significant decrease in reconstructability. Thus, we build a graph of viewpoint candidate nodes, where each viewpoint is only connected to its partner and to viewpoints corresponding to other frontiers. We then find the shortest path in the graph, starting from the current position, passing through the viewpoints, and ending in the midpoint of the last frontier in the local path window.

In contrast to the baseline trajectory generator, where position and yaw trajectories are created separately, we added a joint optimization step and considered more rotational constraints to avoid losing sight of visual features. Keeping the yaw aligned with the heading of the robot is vital for safe exploration. We introduce a limited FoV-constraint inspired by [10], which penalizes the difference between yaw and the angle of the velocity vector. The result of the optimization procedure heavily depends on the initialization, so we propose to use a yaw-aware A* planner. In addition to adding yaw to the state space, we consider two constraints in the node expansion step. Firstly, we want to avoid rotations on the spot. To do so, we check that there is a non-zero translational step between the current point and the neighbor. Secondly, we make sure that the velocity vector between the current pose and the neighbor is always within the FoV.

3.2 Multi-UAV Exploration Method

In the baseline, the authors established a pairwise request-response communication protocol, enabling UAVs to update the assigned workload with only limited communication. The assigned cells are redistributed between the pair of robots to minimize the length of the coverage path, subject to capacity constraints. All robots also continuously advertise their trajectories and explored map chunks. Since consumer-grade UAVs do not possess the computational power and receive

commands from the controller or a ground station, we propose to use a semi-distributed communication scheme. Our pipeline for semi-distributed multi-UAV exploration consists of a single base node and multiple UAVs that perform exploration of individual areas, similar to a single UAV pipeline.

The base node is responsible for three main things. It creates the global map from the positions of UAVs and depth images, and sends the map and the frontier list to UAVs. Secondly, when the UAV has visited the viewpoint, it sends its position and image to the base node, which accumulates them and communicates with the depth estimator. Finally, to avoid suboptimal pairwise hgrid reassignment, it performs the workload assignment and global planning for all UAVs in one iteration.

4 Experiments

We assess the performance of our model compared to the baseline when used with consumer-grade UAVs. The evaluation is done in a simulation. We use Flightmare [15], which is a modular simulator that features a rendering engine built on Unity, capable of outputting RGB and depth images and odometry for multiple quadrotors, and visualizing them in the chosen environment. We provide results of exploring the outdoor industrial environment with a size of 15 x 15 x 3 m. We tested out our pipeline in different scenes, but chose this one because it contains the following challenging parts: a homogeneous open area, a narrow gate, obstacles of different heights, such as crates, and an inaccessible region inside a big pipe. To evaluate the quality of depth estimation, we collect reconstruction accuracy (ACC), surface coverage (COV), and average Hausdorff distance (AHD) metrics. To evaluate the quality of depth estimation, we collect relative pose error (RPE) and absolute trajectory error (ATE).

method	AHD [m] ↓	ACC ↑	COV ↑
no	0.68 ± 0.09	0.53 ± 0.02	0.39 ± 0.06
overlap	0.41 ± 0.05	0.74 ± 0.06	0.59 ± 0.05
heuristic	0.30 ± 0.02	0.93 ± 0.02	0.69 ± 0.02

Table 1: Map quality metrics for viewpoint pairs evaluation methods.

First, we test the effect of our modifications that we proposed in the Viewpoint Planning section for viewpoint pairs sampling and evaluation. We compare the reconstructed map from images sent uniformly every 0.5 m with the one from images sent at viewpoint pairs, where viewpoints were sampled uniformly around edges of the frontier. Additionally, we test the influence of the heuristic-based pair evaluation method on the reconstructed map compared to the basic overlap-based method. Resulting metrics are presented in Table 1 and the map is shown in Figure 2. Viewpoint pair sampling increased the quality of depth estimation because it ensures that the overlap between images is sufficient and viewpoints are located far enough away from each other for the triangulation to work. Additionally, the heuristic-based method significantly outperforms the overlap-based

method, because it considers the difference in the observation angles and avoids motion perpendicular to the frontier.

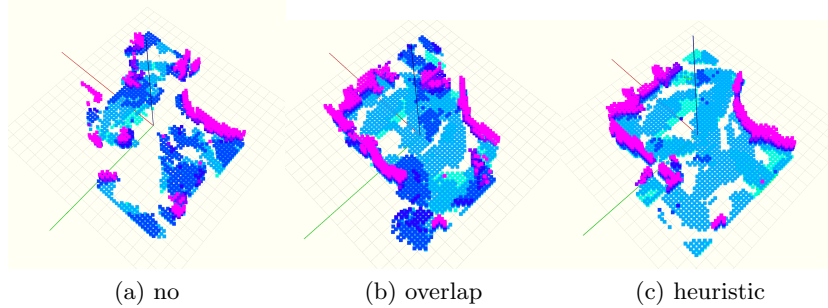


Fig. 2: Map for different viewpoint pairs evaluation methods.

Finally, we show the influence of line sampling on the map and VIO quality in Table 2, and on the path in Table 3. As we can see, the inclusion of line sampling leads to longer paths with fewer rotations, which increases the quality of VIO. In all experiments below, we used the combined sampling approach, which trades off the path length and viewpoint pairs quality with fewer rotations and VIO quality.

method	AHD [m] ↓	ACC ↑	COV ↑	RPE [m] ↓	ATE [m] ↓
uniform	0.32 ± 0.02	0.92 ± 0.02	0.67 ± 0.03	0.02 ± 0.01	0.17 ± 0.04
line	0.39 ± 0.05	0.87 ± 0.05	0.58 ± 0.02	0.01 ± 0.00	0.11 ± 0.06
combined	0.33 ± 0.02	0.87 ± 0.02	0.66 ± 0.01	0.01 ± 0.00	0.12 ± 0.04

Table 2: Map and VIO quality metrics for viewpoint pairs sampling methods.

method	time [s] ↓	length [m] ↓	avg. velocity [$\frac{m}{s}$] ↑	energy [$\frac{m^2}{s^2}$] ↓
uniform	567 ± 5	161.9 ± 5.2	0.29 ± 0.01	50.37 ± 4.15
line	639 ± 48	205.6 ± 16.5	0.32 ± 0.01	79.02 ± 5.21
combined	663 ± 19	199.1 ± 6.2	0.30 ± 0.01	67.69 ± 1.47

Table 3: Path metrics for viewpoint pairs sampling methods.

Next, we compare our yaw-aware A* trajectory generator with the updated and baseline soft-constrained approaches. You can see the estimated and ground truth trajectories in Figure 3, the path metrics in Table 4, and map and odometry metrics in Table 5. Both baseline and updated soft-constrained methods fail to find the path between viewpoints within one pair that satisfies all desired constraints. A UAV travels on a straight line path with yaw perpendicular to the velocity vector, which leads to a shorter route and exploration time, but makes the trajectory unsafe. Our yaw-aware A* solves this problem at the cost of longer path length and exploration time.

Finally, we evaluate the baseline multi-UAV exploration method with our exploration planner compared with our semi-distributed multi-UAV exploration

method	time [s] ↓	length [m] ↓	avg. velocity [$\frac{m}{s}$] ↑	energy [$\frac{m^2}{s^2}$] ↓
baseline	420 ± 110	107.9 ± 18.9	0.27 ± 0.03	50.49 ± 12.06
constrained	372 ± 37	101.6 ± 6.3	0.28 ± 0.03	51.54 ± 9.73
A*	576 ± 24	171.7 ± 10.2	0.30 ± 0.01	63.98 ± 3.53

Table 4: Path metrics for trajectory generation methods.

method	AHD [m] ↓	ACC ↑	COV ↑	RPE [m] ↓	ATE [m] ↓
baseline	0.35 ± 0.01	0.86 ± 0.02	0.61 ± 0.02	0.01 ± 0.00	0.18 ± 0.06
constrained	0.31 ± 0.04	0.86 ± 0.06	0.66 ± 0.05	0.01 ± 0.00	0.26 ± 0.09
A*	0.35 ± 0.02	0.92 ± 0.01	0.64 ± 0.02	0.01 ± 0.00	0.14 ± 0.02

Table 5: Map and VIO quality metrics for trajectory generation methods.

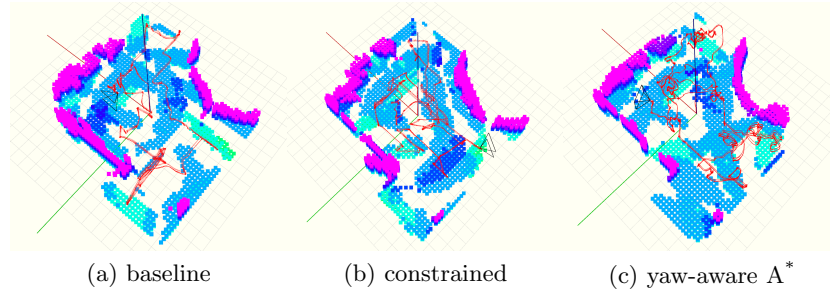


Fig. 3: Map for trajectory generation methods.

method. Path metrics are presented in Table 6, and reconstructed map and trajectories for different numbers of UAVs for our method are shown in Figure 4. Our method outperforms the baseline for both two and three UAVs due to a more optimal workload assignment, which is calculated for all UAVs in one iteration. The number of hgrid cells is limited, so calculating the coverage paths for all UAVs does not introduce a significant computational overhead.

method	UAVs	time [s] ↓	length [m] ↓	avg. velocity [$\frac{m}{s}$] ↑	energy [$\frac{m^2}{s^2}$] ↓
single	1	576 ± 24	171.7 ± 10.2	0.30 ± 0.01	63.98 ± 3.53
baseline	2	503 ± 81	147.0 ± 24.2	0.30 ± 0.01	53.52 ± 4.82
our	2	493 ± 34	134.0 ± 12.4	0.27 ± 0.03	47.40 ± 4.34
baseline	3	426 ± 47	130.5 ± 15.7	0.31 ± 0.01	49.06 ± 5.79
our	3	418 ± 24	97.6 ± 20.2	0.24 ± 0.04	37.24 ± 8.14

Table 6: Average path metrics for multi-UAV exploration methods. Length, avg. velocity, and energy are averaged per UAV.

5 Conclusion

In our work, we extended the baseline method for autonomous multi-UAV exploration to consumer-grade UAVs. To ensure the good quality of depth estimation,

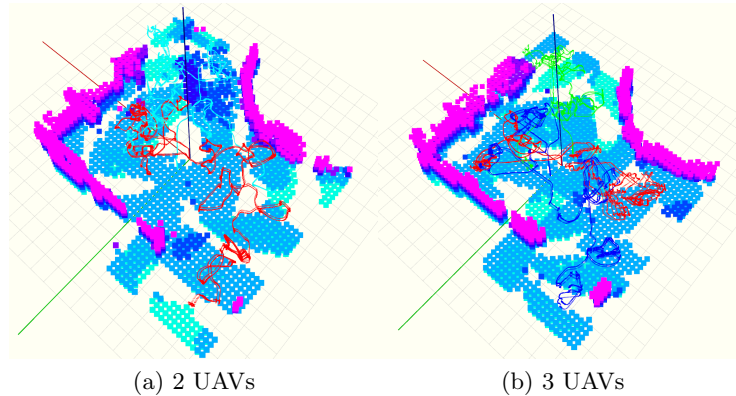


Fig. 4: Map and path for semi-distributed exploration method and different numbers of UAVs.

we proposed to use not viewpoints, but viewpoint pairs. We introduced the way to sample viewpoint pairs and evaluate their quality, and showed its influence on the reconstructed map. Odometry estimation and safety constraints require us to keep the direction of movement within the field-of-view of the UAV. To satisfy these constraints, we introduced the yaw-aware A^* planner and evaluated its performance compared to the baseline trajectory planner. Finally, due to the limited computational capabilities of consumer-grade UAVs, we proposed the semi-distributed communication scheme and showed that it outperforms the baseline fully-distributed method for different numbers of UAVs.

Our method can be extended for larger exploration areas without any changes in the pipeline due to its distributed work assignment and the coarse-to-fine planning methods. However, in real-life experiments, the connection with the ground station may become unstable when exploring large regions. In future works, we would like to implement our pipeline for exploration of the real-world environment. Our main goal will be to test the scalability and generalizability of our method outside of the simulated scenario.

Acknowledgement. This work has been supported by the German Federal Ministry of Research, Technology and Space (BMFT) in the project “Kompetenzzentrum: Etablierung des Deutschen Rettungsrobotik-Zentrums (E-DRZ)”, grant 13N16477.

References

1. Cieslewski, T., Kaufmann, E., Scaramuzza, D.: Rapid exploration with multi-rotors: A frontier selection method for high speed flight. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2135–2142. IEEE (2017)
2. Dai, A., Papatheodorou, S., Funk, N., Tzoumanikas, D., Leutenegger, S.: Fast frontier-based information-driven autonomous exploration with an mav. In: 2020

- IEEE international conference on robotics and automation (ICRA), pp. 9570–9576. IEEE (2020)
3. Dang, T., Tranzatto, M., Khattak, S., Mascariçh, F., Alexis, K., Hutter, M.: Graph-based subterranean exploration path planning using aerial and legged robots. *Journal of Field Robotics* **37**(8), 1363–1388 (2020)
 4. Deng, D., Xu, Z., Zhao, W., Shimada, K.: Frontier-based automatic-differentiable information gain measure for robotic exploration of unknown 3d environments. *arXiv preprint arXiv:2011.05288* (2020)
 5. Han, L., Gao, F., Zhou, B., Shen, S.: Fiesta: Fast incremental euclidean distance fields for online motion planning of aerial robots. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4423–4430. IEEE (2019)
 6. Heng, L., Gotovos, A., Krause, A., Pollefeys, M.: Efficient visual exploration and coverage with a micro aerial vehicle in unknown environments. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 1071–1078. IEEE (2015)
 7. Heng, L., Honegger, D., Lee, G.H., Meier, L., Tanskanen, P., Fraundorfer, F., Pollefeys, M.: Autonomous visual mapping and exploration with a micro aerial vehicle. *Journal of Field Robotics* **31**(4), 654–675 (2014)
 8. Juliá, M., Gil, A., Reinoso, O.: A comparison of path planning strategies for autonomous exploration and mapping of unknown environments. *Autonomous Robots* **33**, 427–444 (2012)
 9. Leroy, V., Cabon, Y., Revaud, J.: Grounding image matching in 3d with mast3r. In: *European Conference on Computer Vision*, pp. 71–91. Springer (2024)
 10. Liu, S., Moha, K., Atanasov, N., Kumar, V.: Towards search-based motion planning for micro aerial vehicles. *arXiv preprint arXiv:1810.03071* (2018)
 11. Papachristos, C., Khattak, S., Alexis, K.: Uncertainty-aware receding horizon exploration and mapping using aerial robots. In: 2017 IEEE international conference on robotics and automation (ICRA), pp. 4568–4575. IEEE (2017)
 12. Schleich, D., Quenzel, J., Behnke, S.: Autonomous consumer-grade uavs for real-time situational awareness in gnss-denied search & rescue operations. *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)* (2025)
 13. Schmid, L., Pantic, M., Khanna, R., Ott, L., Siegwart, R., Nieto, J.: An efficient sampling-based method for online informative path planning in unknown environments. *IEEE Robotics and Automation Letters* **5**(2), 1500–1507 (2020)
 14. Smith, N., Moehrle, N., Gesele, M., Heidrich, W.: Aerial path planning for urban scene reconstruction: A continuous optimization method and benchmark (2018)
 15. Song, Y., Naji, S., Kaufmann, E., Loquercio, A., Scaramuzza, D.: Flightmare: A flexible quadrotor simulator. In: *Proceedings of the 2020 Conference on Robot Learning*, pp. 1147–1157 (2021)
 16. Sonugür, G.: A review of quadrotor uav: Control and slam methodologies ranging from conventional to innovative approaches. *Robotics and Autonomous Systems* **161**, 104,342 (2023)
 17. Zhou, B., Xu, H., Shen, S.: Racer: Rapid collaborative exploration with a decentralized multi-uav system. *IEEE Transactions on Robotics* **39**(3), 1816–1835 (2023)
 18. Zhou, B., Zhang, Y., Chen, X., Shen, S.: Fuel: Fast uav exploration using incremental frontier structure and hierarchical planning. *IEEE Robotics and Automation Letters* **6**(2), 779–786 (2021)