SYNERGY PUBLISHING PTE. LTD.

Modern Electronic Technology

http://www.advmodoncolres.sg/index.php/amor/index

An Improved Multi-UAV Rapid Autonomous Exploration Method Based on Environmental Complexity Mode Switching

Menglong Ma Chunyu Chen[*†]

Information and Communication Engineering, Harbin Engineering University, Harbin, 150001, China

ARTICLE INFO

Article history
Received: 25 February 2024
Revised: 1 March 2024
Accepted: 23 March 2024
Published Online: 30 March 2024

Keywords:
Aerial System
Perception and Autonomy
Path Planning
Environmental Complexity

ABSTRACT

With the continuous development of drone technology, rapid exploration strategies are of significant importance for tasks such as search and rescue and surveying. Current autonomous exploration systems often face issues of partial small-area information omission in cluttered environments, leading to repeated visits by drones. This paper proposes an improved multi-drone autonomous exploration system, which introduces a novel mode-switching mechanism based on a rapid autonomous exploration framework. This mechanism dynamically adjusts the exploration mode of the drones using the density information of surrounding obstacles. By doing so, drones can avoid missing small pieces of information that result in repeated visits in complex environments, while maintaining high exploration efficiency in simpler environments. This flexible exploration planning approach effectively addresses varying levels of environmental complexity. Evaluations conducted in three different environments of varying complexity demonstrate that the proposed method achieves higher exploration efficiency and reconstruction quality.

1. INTRODUCTION

In modern society, indoor building environments are frequently accessed by people. During disasters, minimizing secondary casualties is critical, and robots become an essential solution. Due to the lack of detailed information about the task area and the complexity of indoor environments, GPS signals often become weak or unavailable, limiting robotic search and rescue operations. In such cases, autonomous exploration and visual positioning technologies are crucial tools. Autonomous exploration enables robots to independently navigate disaster-stricken buildings, search for victims, and assess potential hazards, reducing the risk of further injuries.

Utilizing visual or radar-based positioning technologies such as VINS or SLAM, robots can accurately determine their location without GPS and build real-time maps of the environment, improving obstacle avoidance and navigation capabilities. Thus, the application of autonomous exploration technology in robots significantly enhances the efficiency and safety of rescue operations.

In recent years, Unmanned Aerial Vehicles(UAV) have demonstrated exceptional adaptability in various exploration tasks due to their small size and flexibility, particularly in complex indoor environments. These include narrow corridors and intricate stair structures filled with obstacles.

**Corresponding Author:*
Chunyu Chen,
E-mail(s): springrain@hrbeu.edu.cn;
Contributing authors:
ImagineDragon.Ma@gmail.com;
[†]*These authors contributed equally to this work.*

 DOI: https://doi.org/10.26549/met.v8i1.19489

With advanced sensing technologies, drones can autonomously plan paths, explore environments, and quickly adapt to changes, completing pre-rescue exploration tasks efficiently. This reduces the risk to rescue personnel and significantly improves the efficiency of rescue operations. As drone technology advances, the limitations of single-drone missions in largescale scenarios have led to the development of multi-drone systems. Compared to a single drone, multiple drones can cover more areas simultaneously, greatly enhancing search efficiency and reducing rescue time, making operations faster and more efficient.

This paper focuses on decentralized multidrone autonomous exploration technology, aiming to complete exploration tasks quickly in cluttered environments with unevenly distributed obstacles. The goal is to achieve comprehensive map coverage while minimizing repeated visits and task switching. We propose a mode-switching algorithm that allows drones to adaptively switch exploration modes based on their environment without changing their assigned tasks. Drones can explore quickly in simple environments and conduct detailed exploration in obstacle-dense areas to avoid missing information and repeated visits. Multi-robot coordination is achieved through direct point-to-point communication in a decentralized manner, with a task allocation module reasonably dividing the areas to be explored. We tested the proposed method in three different maze simulations, evaluating its performance. The results revealed the method's superior efficiency, achieving higher overall drone speed and shorter exploration times.

The main contributions of this paper are as follows:

- Implementing mode-switching in exploration considering the varying complexity of different areas without changing the task.
- Conducting extensive simulation evaluations demonstrating superior performance compared to existing technologies.

2. RELATED WORK

For many years, researchers have focused on mobile robot exploration in unknown environments. The current mainstream approach for autonomous exploration is frontier-based, emphasizing the planning of efficient exploration paths to create more complete maps in less time, with shorter distances, and lower energy consumption. Frontiers are defined as the boundaries between known and unknown spaces, a concept first introduced by Yamauchi et al. in 1997 [1]. They used frontiers to identify information-rich areas and plan paths sequentially. Subsequent research introduced information gain to evaluate candidate actions and balance exploration time and information gain based on the environment, as seen in Stachniss et al.'s work [2-3].

Frontier-based methods and Next-Best-View approaches typically consider only the information gain of the next exploration viewpoint, neglecting the information gain along the path to the next viewpoint. Bircher et al. [4] proposed the Receding Horizon Next-Best-View planner (RH-NBVP), which finds the best branch in a computed random tree and repeats the process for better exploration of large environments. Some papers, like Meng et al. [5], combined frontier with sampling methods to improve computational efficiency by sampling viewpoints around the frontier. Recently, Zhou et al. [6] introduced a method using incremental frontier detection and surrounding viewpoint sampling to generate effective global paths, achieving safe and agile local exploration operations. Duberg et al. [7] argued that the size of the information gain is not crucial, instead selecting view-points with information gain above a threshold and then moving to the nearest one.

To further improve exploration efficiency and reduce time, early papers like [8] proposed multirobot collaboration, initially using centralized communication and task allocation via a central server. Burgard et al. [9] coordinated multi-robot tasks by greedily choosing robot-target pairs based on distance and information gain. Tian et al. [10] introduced the multi-Travelling Salesman Problem (mTSP) to allocate robots to candidate frontiers. However, real-world factors like environment complexity and server reliability affect communication quality, leading to decentralized coordination methods. The first decentralized multi-robot exploration, developed by Palazzolo et al. [11], had robots share map information and move to the nearest frontier. Despite its simplicity, coordination was ineffective without central control. Kabir et al. [12] and Yu et al. [13] proposed methods to mitigate these issues, but required stable communication. Klodt et al. [14] reduced communication dependency through pairwise interactions, and Zhou et al. [15] introduced a method to divide navigation areas and ensure different exploration regions, enhancing robustness for limited communication.

However, Bartolomei et al. [16] highlighted that Zhou's method assumes uniform obstacle distribution, leading to decreased coordination in cluttered environments and leaving unexplored islands. They proposed a flexible task-switching navigator to address this, changing some UAVs' tasks to clear unknown spaces. This improved efficiency but altered local tasks. To avoid this, we developed a new mode-switching method that adjusts path planning costs based on environmental complexity, ensuring slow and cautious exploration in complex environments to

DOI: https://doi.org/10.26549/met.v8i1.19489

avoid missing small areas and reducing repeated visits. This approach enhances task efficiency without changing UAV tasks.

3. METHODOLOGY

3.1 System Overview

The overall system design is depicted in Figure 1. This study focuses on exploring unknown environments with communication constraints using UAVs equipped with front-facing depth cameras. We employ the Visual-Inertial Navigation System (VINS) method as outlined in [17] for distributed state estimation, ensuring accurate localization of both self and other quadcopters.

Our approach utilizes a decentralized coordination strategy where agents exchange map information, viewpoints, execution modes, and planned trajectories to allocate exploration areas and collaborate on map construction. Each UAV operates through a pipeline consisting of three main components: mapping system, collaborative task allocator, mode selector, and path planner.

Inspired by reference [16], we have also developed a mode-switching algorithm. The mapping system generates a voxel grid map of the environment using odometry and depth information [30]. With each update, newly observed voxels are stored in blocks and communicated to nearby UAVs. Next, Frontiers are extracted and clustered. Subsequently, using paired interactions and the Capacitated Vehicle Routing Problem (CVRP) formula [15], UAV tasks are coordinated. Upon receiving their respective tasks, UAVs plan exploration routes, extracting frontier information to identify viewpoints at boundaries between known and unknown maps.

Time lower bounds related to path time cost and angular velocity cost are obtained using ATSP to stratify UAVs for global and local path planning. The mode switcher determines mode switching based on the density of surrounding obstacles. Specifically, point cloud information from the mapping system is used to extract obstacle information. Gaussian distribution sampling is performed on this information to determine obstacle density, dynamically triggering mode switches. Different modes assign varying speeds, task boundaries, and proportions of path time cost and angular velocity cost to UAVs.

3.2 Multi-Task Allocation

In this study, we adopt the Hgrid map division method mentioned in [2]. For multi-task allocation, we employ the vehicle routing problem (VRP). In a standard VRP, routes form closed loops from a central depot. However, since exploration tasks lack a central depot, [2] introduces a virtual depot and designs connection costs to reduce this variant to an asymmetric VRP. Following this approach, we design the CVRP formula. Suppose there are N_g grid cells and N_x UAV nodes. The relevant cost matrix $C_{avrp} \in \mathbb{R}^{(N_g+N_x+1)\times(N_g+N_x+1)}$ is as follows:

$$C_{avrp} = \begin{bmatrix} 0 & -\mathbf{Inf}_{d,x} & \mathbf{Inf}_{d,g} \\ \mathbf{Inf}_{d,x} & \mathbf{Inf}_{x} & \mathbf{C}_{x,g} \\ 0 & 0 & \mathbf{C}_{g} \end{bmatrix} \tag{1}$$

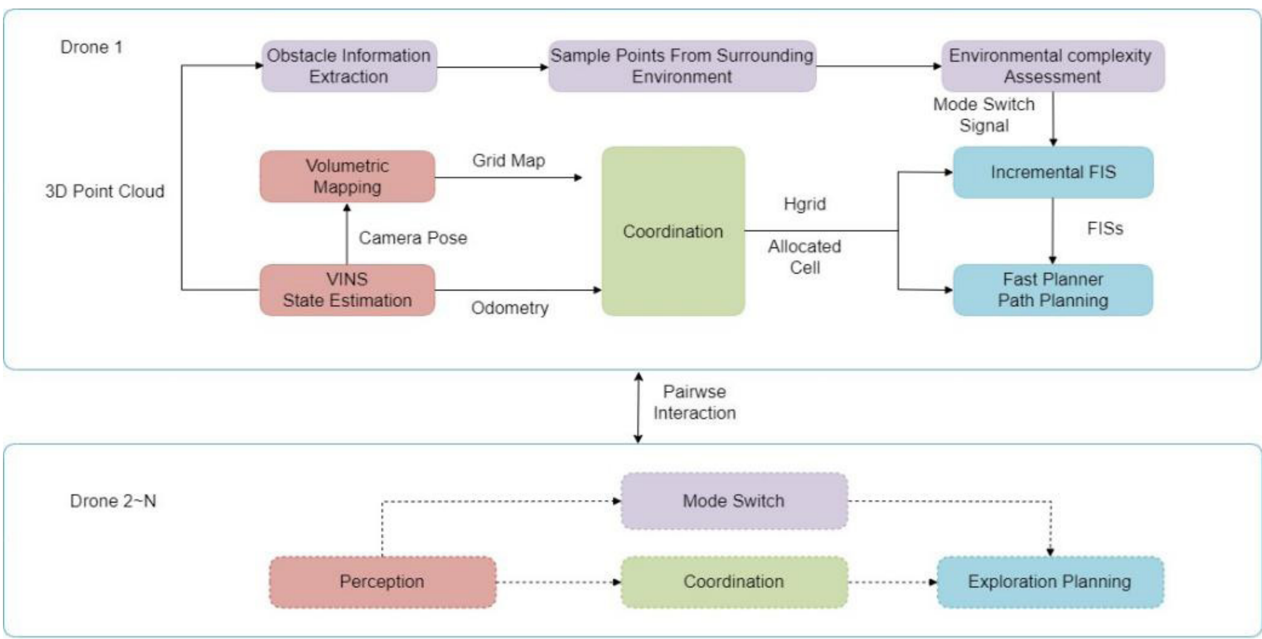


Figure 1. Overview of the Multi-UAV Autonomous Exploration System.

 DOI: https://doi.org/10.26549/met.v8i1.19489

Formula 1 represents a basic CVRP cost matrix. Since drones do not need to return to the same point, all diagonal elements of the cost matrix are set to 0. In this matrix, $-\mathbf{Inf}_{d,x}$ denotes the connection cost from the virtual depot to all drones, set to a large negative value to allow direct connections between the virtual depot node and the drones. $\mathbf{Inf}_{d,g}$ and $\mathbf{Inf}_{d,x}$ represent the connection costs from the virtual depot to grid cells and from drones to the virtual depot, respectively, both set to a large positive value to prohibit infeasible paths. $\mathbf{Inf}_x$ represents the connection costs between different drones, with all elements except the diagonal set to a large positive value to indicate infeasible paths. $\mathbf{C}_{x,g}$ denotes the connection costs between drones and grid cells, while $\mathbf{C}_g$ represents the connection costs between grid cells, typically calculated using the path length $Len(\cdot)$.

3.3 Exploration Planning

In this paper, the boundary is obtained using the method in [6]. Then, sampling is performed uniformly in a cylindrical coordinate system centered at the cluster center to obtain viewpoints with coverage angles greater than the threshold. The angle with the largest coverage is selected as the angle bound to the viewpoint. Each viewpoint $q_{i,j}$ is represented by its position and yaw angle $\{x_{i,f}, \varphi_{ij}\}$.

1) Fast Exploration Mode: Driven by frontiers, the goal of the fast exploration mode is to cover large previously unknown areas quickly. Considering that different factors need to be taken into account for path planning in various environments, we processed the most important time cost t_{lb} from [6], changing the original method of directly selecting the maximum of the path time cost $T_{L,cost}$ and the turning time cost $T_{\varphi,cost}$ to a weighted sum method:

$$t_{lb}(\mathbf{x}_{k_1,j_1}, \mathbf{x}_{k_2,j_2}) = w_L T_{L,cost} + w_T T_{\varphi,cost} \tag{2}$$

$$T_{L,cost} = \frac{Len[P(\mathbf{x}_{k_1,j_1}, \mathbf{x}_{k_2,j_2})]}{v_{max}} \tag{3}$$

$$T_{\varphi,cost} = \frac{min(|\varphi_{k_1,j_1} - \varphi_{k_2,j_2}|, 2\pi - |\varphi_{k_1,j_1} - \varphi_{k_2,j_2}|)}{\dot{\varphi}_{max}} \tag{4}$$

$T_{L,cost}$ and $T_{\varphi,cost}$ represent the path time cost and turning time cost respectively. w_L and w_T are their respective weights. In fast exploration mode, we set a higher weight for the path time cost w_L and a smaller weight for the turning time cost w_T. This encourages the drones to explore unknown areas more greedily, similar to the approach described in the paper [7], thereby improving exploration efficiency.

Subsequently, following the method outlined in [6], we constructed an Asymmetric Traveling Salesperson Problem (ASTP) to find a complete path passing through each cluster. The cost from the current position $\boldsymbol{x}_0 = (p_0, \varphi_0)$ to the N_{ftr} clusters is calculated as shown in Formula 5(In the formula $k \in [1, N_{cls}]$)

$$M_{tsp}(0,k) = t_{lb}(\mathbf{x}_0, \mathbf{x}_{k,1}) + w_{con} \cdot t_{con}(\mathbf{x}_{k,1}) \tag{5}$$

To achieve more consistent movement and limit excessive turns, avoiding UAVs from frequently switching between different target points with similar t_{lb} values, we calculate the angle between the current direction and the target direction, where v_0 is the current velocity:

$$t_{con}(\mathbf{x}_{k,j}) = cos^{-1}\frac{(p_{k,j} - p_0)\cdot v_0}{\| p_{k,j} - p_0 \| \| v_0 \|} \tag{6}$$

Here, $p_{k,j}$ represents the current position of the drone, p_0 is the position of the drone before it moved, and v_0 is the speed of the drone.

The cost between each cluster and the current position x_0 is then uniformly set to 0. This avoids the exploration task forming a closed loop, allowing it to be converted into a traditional ATSP problem for processing. Disconnecting the closedloop path does not affect the optimality of the result.

$\mathbf{C}_f$ is the main symmetric block recording the connection costs between clusters, primarily the time cost from each cluster to other clusters:

$$\mathbf{C}_f(k_1,k_2) = t_{lb}(\mathbf{q}_{k_1,1}, \mathbf{q}_{k_2,1}), k_1, k_2 \in [1, N_{ftr}] \tag{7}$$

2) Cautious Exploration Mode: The cautious exploration mode aims to avoid small unexplored areas in complex environments as much as possible, preventing the need to revisit less explored areas on the map after the mission ends. In simple or open environments, drones can fly and explore in straight or near-straight paths, with small turns, making path costs generally more important than turning costs. However, in complex environments, such as narrow urban streets, forests, or building interiors, frequent turns and obstacle avoidance make turning costs more significant. In cautious exploration mode, drones reduce their maximum speed $v_{\max}$ and adjust the weights in the time lower bound function, increasing the weight of the turning time cost w_T and decreasing the weight of the path time cost w_L. This gives a new time lower bound t_{lb}, which is then used in the ATSP formula, making the drones explore more cautiously, reducing missed information. This also avoids large turns, maintaining safety while exploring unknown areas as thoroughly as possible.

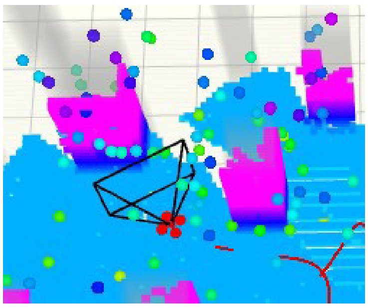
3.4 Environmental Complexity Mode Switching

In both exploration and planning tasks, assessing environmental complexity is crucial as it directly impacts robot performance, task success rates, and resource consumption. For instance, as discussed in [18], environmental

DOI: https://doi.org/10.26549/met.v8i1.19489

complexity is determined by the number of obstacles surrounding the drone. In this study, we simplify obstacle detection using point cloud data segmentation, where obstacles are reduced to points to determine levels of environmental complexity. We employ a region growing algorithm for cluster segmentation,

(a) Obstacle Informa-



(b) Sampling Points tion Extraction. Visualization

Figure 2. Extraction and Augmentation of Environmental Information: In figure (a), the green points represent the clustering results, while in figure (b), the small points of various colors represent the Gaussian distribution sampling results.

dividing the point cloud data P into n subregions $P_1, P_2, \ldots, P_n$, each representing a distinct obstacle.

Reducing obstacle information to points via clustering significantly reduces the data to be processed. To obtain more usable information for subsequent analysis, we randomly sample around these simplified obstacle points to obtain N threedimensional sampling points $\mathrm{X} = \{(\mathrm{x_i}, \mathrm{y_i}, \mathrm{z_i})\}_{\mathrm{i}=1}^{\mathrm{N}}$, which follow a multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$..

Taking inspiration from [19], in this study, we construct a square sliding window centered at the current drone position x_{new} to delineate local environmental regions for computation purposes. We measure environmental complexity using information entropy. Specifically, we assume each dimension is divided into n_x n_y n_z grids. We initialize a three-dimensional matrix $\mathbf{G}$ of size $n_x \times n_y \times n_z$ and traverse all sampling points. Based on their grid indices, we increment corresponding counts in the matrix.

$$\mathrm{index}_{\mathbf{x}} = \left\lfloor \frac{\mathbf{x}_i - \mathbf{x}_{\min}}{\Delta \mathbf{x}} \right\rfloor \tag{8}$$

Here, $\mathbf{x}$ represents the coordinates of the sampling points. After traversal, we obtain the matrix $\mathrm{G}\left(\mathrm{index}_{\mathrm{x_x}}, \mathrm{index}_{\mathrm{x_y}}, \mathrm{index}_{\mathrm{x_y}}\right)$ containing the number of sampling points in each grid. We then calculate the probability distribution of the sampling points in each grid:

$$P_{ijk} = \frac{\mathbf{G}(i,j,k)}{\sum_{i=0}^{n_x-1}\sum_{j=0}^{n_y-1}\sum_{k=0}^{n_z-1}\mathbf{G}(i,j,k)} \tag{9}$$

where $Pijk$ is the probability of sampling points in the (i,j,k) grid. Next, we calculate the information entropy within the sliding window:

$$H = -\sum_{i=0}^{n_x-1}\sum_{j=0}^{n_y-1}\sum_{k=0}^{n_z-1} P_{ijk} \log\left(P_{ijk} + \epsilon\right) \tag{10}$$

To avoid log(0), a small value ϵ is added.The resulting information entropy represents the environmental complexity

In specific exploration tasks, When the environmental complexity within the UAV's sliding window exceeds a threshold H_{max}, indicating a high-complexity environment, the UAV switches to cautious exploration mode. Conversely, if the number of sampling points around the UAV is less than H_{min} indicating a low-complexity environment, the UAV switches to fast exploration mode. For complexity levels between H_{min} and H_{max}, mode switching is not directly triggered. Instead, we simulate the complexity change rate based on the rate of change in the number of sampling points within the UAV's sliding window. The UAV's previous exploration mode and the trend in sampling point changes determine whether to switch modes. The rate of change a is calculated as follows:

$$a = \frac{\sum_{i=0}^{k-1}(i-\overline{t})(n_{t-i}-\overline{n})}{\sum_{i=0}^{k-1}(i-\overline{t})^2} \tag{11}$$

where $\overline{t}$ is the mean of the time series, and $\overline{n}$ is the mean of the sampling points. The slope a indicates the trend. When the number of sampling points is between n_{min} and n_{max}:

- If $a > 0$, indicating an increase in sampling points, and exceeds h_{thr} it suggests the UAV has entered a cluttered environment. If the UAV is in fast exploration mode, it should switch to cautious exploration mode.
- If $a < 0$, indicating a decrease in sampling points, and is below l_{thr}, it suggests the UAV has entered an open environment. If the UAV is in cautious exploration mode, it should switch to fast exploration mode.
- If a is between h_{thr}, and l_{thr}, the environmental change

DOI: https://doi.org/10.26549/met.v8i1.19489

is not significant, and the UAV continues in its current mode without switching.

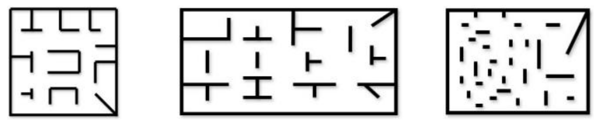
The thresholds h_{thr}, and l_{thr} are small positive and negative values of equal magnitude, respectively. This method ensures efficient mode switching based on real-time environmental complexity, optimizing the UAV's exploration efficiency.

4. EXPERIMENTS

4.1 Results of Different Multi-UAV Exploration Methods in Various Environments

To evaluate the performance and feasibility of the proposed method, we used the dynamics simulator from [15] for simulation. Each agent is equipped with a forward-facing depth camera with a resolution of 640 × 480 pixels and a field of view of 80° × 60°. Depth images are processed using the method in [17], with a maximum perception range of 5m. The maximum linear and angular velocities of the drones are set to 1.5*m/s* and 0.9*rad/s*, respectively. We compared the exploration process with RACER [15] and a Ufoexplorer-like algorithm [7] to verify the time required to complete scene exploration and the completeness of the established map. The core of the Ufoexplorer algorithm is selecting the nearest viewpoints; in our experiments, we implemented this by setting the path cost weight to 1 and the corner cost weight to 0.

We tested three scene sizes: a simple maze with dead ends 20*m*×20*m*×2*m*, a more complex maze with dead ends and loops 30*m* × 16*m* × 2*m*, and a highly cluttered maze with unevenly distributed obstacles 20*m* × 14*m* × 2*m*.



(a) Simple **(b)** Complex **(c)** Highly

Figure 3. Three schematic diagrams depicting mazes of varying complexity.

Through testing each algorithm in different environments, we obtained the results shown in the Table 1.

Table 1. Map coverage and exploration efficiency data achieved by different numbers of drones using different methods in different environments.

num		Simple		Complex		Highly Complex	
1	RACER	96.16% ± 1.52%	111.16*s* ± 10.15	96.61% ± 1.61%	114.15*s* ± 10.21	96.34% ± 1.76%	116.23*s* ± 10.12
	Ufoexplorer	96.14% ± 1.24%	110.31*s* ± 9.75	96.10% ± 1.65%	117.36*s* ± 11.02	96.01% ± 1.81%	121.80*s* ± 11.41
	Ours	97.33% ± 0.51%	95.24*s* ± 5.17	97.42% ± 0.53%	103.98*s* ± 5.23	97.46% ± 0.63%	91.31*s* ± 5.14
2	RACER	96.19% ± 1.31%	97.69*s* ± 10.05	96.69% ± 1.60%	99.19*s* ± 10.07	96.40% ± 1.71%	101.33*s* ± 10.13
	Ufoexplorer	96.21% ± 1.21%	95.50*s* ± 9.51	96.17% ± 1.62%	102.21*s* ± 10.53	96.21% ± 1.79%	107.65*s* ± 10.21
	Ours	98.04% ± 0.45%	76.82*s* ± 5.04	98.03% ± 0.43%	79.68*s* ± 4.91	97.46% ± 0.57%	74.62*s* ± 4.87
4	RACER	96.46% ± 1.28%	76.15*s* ± 9.34	96.72% ± 1.31%	79.21*s* ± 9.65	96.79% ± 1.48%	78.70*s* ± 9.62
	Ufoexplorer	96.63% ± 1.17%	73.13*s* ± 9.12	96.52% ± 1.38%	83.12*s* ± 9.78	96.49% ± 1.63%	84.75*s* ± 9.77
	Ours	98.24% ± 0.43%	65.42*s* ± 4.56	98.31% ± 0.42%	68.42*s* ± 4.71	98.23% ± 0.51%	62.37*s* ± 4.09
6	RACER	96.82% ± 1.19%	67.31*s* ± 9.12	96.94% ± 1.28%	69.74*s* ± 9.15	96.61% ± 1.49%	68.72*s* ± 9.08
	Ufoexplorer	96.99% ± 1.16%	64.14 ± 8.98*s*	96.56% ± 1.31%	73.14*s* ± 9.41	96.64% ± 1.52%	75.71*s* ± 9.16
	Ours	98.23% ± 0.47%	61.48*s* ± 4.12	98.21% ± 0.43%	63.07*s* ± 4.67	98.12% ± 0.52%	56.83*s* ± 4.11
8	RACER	96.36% ± 1.49%	65.11*s* ± 9.21	96.72% ± 1.42%	69.12*s* ± 9.72	96.52% ± 1.53%	67.79*s* ± 9.12
	Ufoexplorer	96.32% ± 1.38%	62.68*s* ± 8.03	96.64% ± 1.45%	72.71*s* ± 10.13	96.43% ± 1.57%	74.12*s* ± 9.29
	Ours	98.13% ± 0.49%	58.28*s* ± 3.98	98.19% ± 0.47%	62.15*s* ± 4.93	98.21% ± 0.61%	56.75*s* ± 4.41

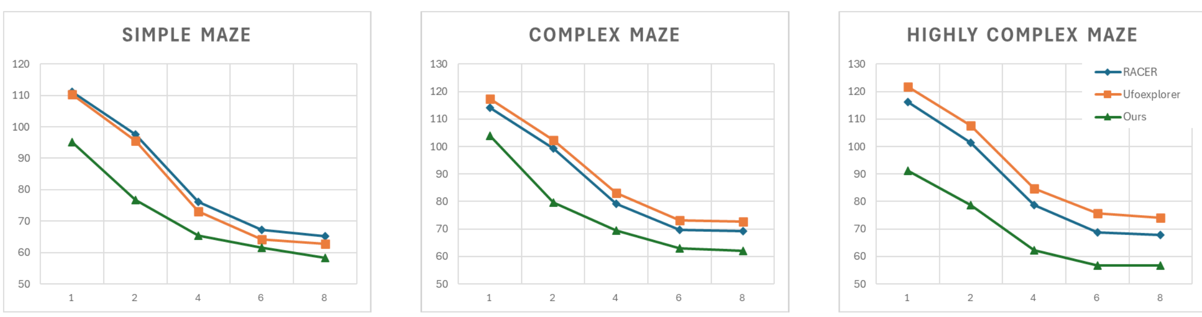


Figure 4: Illustration of exploration efficiency achieved by different numbers of drones using different methods in different environments.

DOI: https://doi.org/10.26549/met.v8i1.19489

From Figure 3, it is evident that in relatively simple environments, the efficiency of Ufoexplorer, which considers only path costs, is superior to RACER. This is because in simple environments, which are mostly open spaces, Ufoexplorer greedily explores based on path costs. However, during experiments, it was found that Ufoexplorer struggles more than RACER when encountering dead ends, resulting in similar performance between the two algorithms overall. RACER considers both path costs and the maximum angle cost, prioritizing lowering the path cost when facing dead ends to swiftly find an escape route. Continuously prioritizing low path costs might lead the drone to repeatedly attempt to navigate out of dead ends without making necessary turns.

Therefore, compared to Ufoexplorer, our mode-switching approach takes such situations into account. It not only enables fast exploration in open spaces similar to Ufoexplorer but also efficiently navigates out of dead ends akin to RACER, achieving higher exploration efficiency. In more complex mazes with numerous obstacles, the importance of angle costs becomes more apparent to enhance exploration efficiency and flight stability. Hence, in such scenarios, RACER generally outperforms Ufoexplorer. However, our mode-switching improvement method also performs reliably in complex environments with multiple obstacles, maintaining stable completion of exploration tasks.

In the last type of maze, characterized by numerous unevenly distributed obstacles, we designed the environment with partial open spaces and partially complex and uneven areas. Despite this map being the smallest, exploration time is longer compared to the previous two maps for both RACER and Ufoexplorer. Our modeswitching improvement method balances path costs and angle costs during operation, thereby achieving better operational efficiency. Additionally, by observing the error values of different algorithms, it is clear that our designed algorithm not only outperforms others in terms of performance but also exhibits superior stability during operation. Lower and less fluctuating error values indicate consistent efficiency and reliability over prolonged operation.

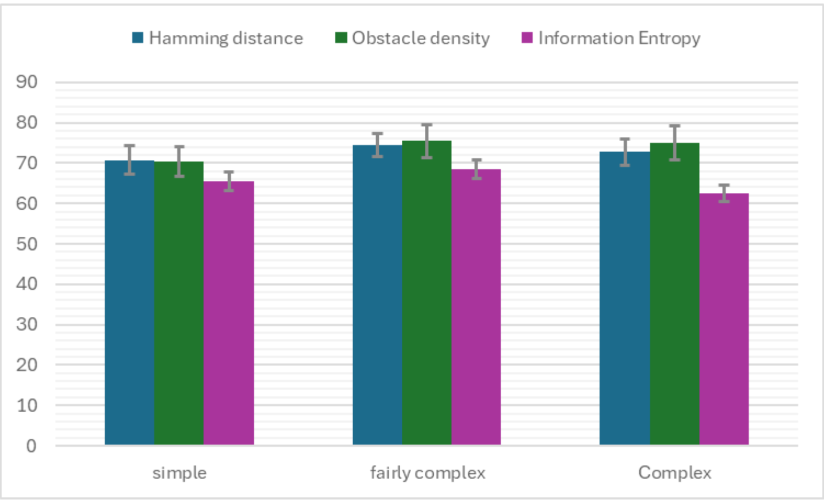


Figure 5. Illustration of the exploration efficiency of four drones under different environmental measurement methods.

Overall, these analytical results thoroughly demonstrate the superiority and practicality of our algorithm. By offering higher exploration efficiency and more stable operational performance, our algorithm provides a more efficient and reliable solution for multi-UAV cooperative exploration tasks.

4.2 The impact of environmental complexity measurement methods

To demonstrate the effectiveness of the entropybased environmental complexity measurement method used in this experiment, we conducted tests using four UAVs in the three environments from the previous experiments. We compared the efficiency of the Hamming distance method, the obstacle density method, and the entropy method as outlined in [18]. The results are presented in Table 2.

Table 2. Exploration efficiency data of four drones under different environmental measurement methods.

method	Simple	Complex	Highly Complex
Hamming distance	$70.75s \pm 6.86$	$74.42s \pm 5.51$	$72.71s \pm 6.61$
Obstacle density	$70.36s \pm 7.56$	$75.42s \pm 7.98$	$74.93s \pm 8.39$
Information Entropy	$65.42s \pm 4.56$	$68.42s \pm 4.71$	$62.37s \pm 4.09$

As shown in Figure 5, We used information entropy as a measure of environmental complexity, which outperformed Hamming distance and obstacle density methods in all three types of environments. The error bars in the

DOI: https://doi.org/10.26549/met.v8i1.19489

figures also indicate better stability. Unlike the Hamming distance, which only compares differences between rows and columns within a sliding window, we represent obstacles as ones and free spaces as zeros in a binary pixel matrix, treating uncertain areas in unknown environments as zeros like free spaces. While the Hamming distance method performs better than simple obstacle density in complex and moderately complex environments, it doesn't consider the probability distribution and uncertainty of the data, making it less comprehensive in reflecting environmental complexity. Thus, the information entropy method is more effective in measuring environmental complexity.

5. CONCLUSIONS

The primary objective of our work is to develop a multi-UAV autonomous exploration system capable of completing environment exploration in complex settings with numerous and unevenly distributed obstacles while ensuring map integrity. To achieve this, we have incorporated a modeswitching feature into the rapid autonomous exploration framework. This feature leverages the reconstructed spatial information from the mission, analyzes surrounding obstacle information to determine environmental complexity, and adaptively switches modes based on the complexity and previous mode.

Through this mode-switching system, UAVs can autonomously balance cautious exploration of unknown spaces with more aggressive exploration. This approach enhances the efficiency and reliability of multi-UAV cooperative exploration in complex environments while maintaining rapid exploration in simpler environments. In the future, we plan to apply more advanced reinforcement learning methods to improve the accuracy of UAV environmental assessments. Additionally, we intend to implement our current method in search and rescue systems to validate the feasibility and superiority of the rapid autonomous exploration approach in real-world applications.

References

[1] Yamauchi, B.: A frontier-based approach for autonomous exploration. In: Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation', pp. 146–151 (1997). IEEE

[2] Stachniss, C., Hahnel, D., Burgard, W.: Exploration with active loop-closing for fastslam. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566), vol. 2, pp. 1505–1510 (2004). IEEE

[3] Stachniss, C., Grisetti, G., Burgard, W.: Information gain-based exploration using rao-blackwellized particle filters. In: Robotics: Science and Systems, vol. 2, pp. 65–72 (2005)

[4] Bircher, A., Kamel, M., Alexis, K., Oleynikova, H., Siegwart, R.: Receding horizon" next-best-view" planner for 3d exploration. In: 2016 IEEE International Conference on Robotics and Automation (ICRA), pp. 1462–1468 (2016). IEEE

[5] Meng, Z., Qin, H., Chen, Z., Chen, X., Sun, H., Lin, F., Ang, M.H.: A two-stage optimized next-view planning framework for 3-d unknown environment exploration, and structural reconstruction. IEEE Robotics and Automation Letters **2**(3), 1680–1687 (2017)

[6] 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)

[7] Duberg, D., Jensfelt, P.: Ufoexplorer: Fast and scalable sampling-based exploration with a graph-based planning structure. IEEE Robotics and Automation Letters 7(2), 2487– 2494 (2022)

[8] Burgard, W., Moors, M., Fox, D., Simmons, R., Thrun, S.: Collaborative multi-robot exploration. In: Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065), vol. 1, pp. 476–481 (2000). IEEE

[9] Witting, C., Fehr, M., B¨ahnemann, R., Oleynikova, H., Siegwart, R.: History-aware autonomous exploration in confined environments using mavs. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1–9 (2018). IEEE

[10] Wang, C., Zhu, D., Li, T., Meng, M.Q.-H., De Silva, C.W.: Efficient autonomous robotic exploration with semantic road map in indoor environments. IEEE Robotics and Automation Letters 4(3), 2989–2996 (2019)

[11] Palazzolo, E., Stachniss, C.: Effective exploration for mavs based on the expected information gain. Drones 2(1), 9 (2018)

[12] Kabir, R.H., Lee, K.: Efficient, decentralized, and collaborative multi-robot exploration using optimal transport theory. In: 2021 American Control Conference (ACC), pp. 4203–4208 (2021). IEEE

[13] Yu, J., Tong, J., Xu, Y., Xu, Z., Dong, H., Yang, T., Wang, Y.: Smmr-explore: Submapbased multi-robot exploration system with multi-robot multi-target potential field exploration method. In: 2021 IEEE International Conference on Robotics and Automation

  DOI: https://doi.org/10.26549/met.v8i1.19489

(ICRA), pp. 8779–8785 (2021). IEEE

[14] Klodt, L., Willert, V.: Equitable workload partitioning for multi-robot exploration through pairwise optimization. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2809–2816 (2015). IEEE

[15] Zhou, B., Xu, H., Shen, S.: Racer: Rapid collaborative exploration with a decentralized multi-uav system. IEEE Transactions on Robotics (2023)

[16] Bartolomei, L., Teixeira, L., Chli, M.: Fast multi-uav decentralized exploration of forests. IEEE Robotics and Automation Letters (2023)

[17] Qin, T., Li, P., Shen, S.: Vins-mono: A robust and versatile monocular visual-inertial state estimator. IEEE Transactions on Robotics 34(4), 1004–1020 (2018)

[18] BoHan KANG, J.H.: Variable step size rapidly-exploring random tree (rrt) path planning algorithms and simulation of a mobile robot based on environment complexity. Journal of Beijing University of Chemical Technology 50(4), 87–93 (2023)

[19] Xia, X., Roppel, T., Hung, J.Y., Zhang, J., Periaswamy, S.C., Patton, J.: Environmental complexity measurement using shannon entropy. In: 2020 SouthEastCon, pp. 1–6(2020). IEEE