



Article

Multi-UAV Cooperative and Continuous Path Planning for High-Resolution 3D Scene Reconstruction

Haigang Sui, Hao Zhang , Guohua Gou , Xuanhao Wang, Sheng Wang, Fei Li and Junyi Liu

State Key Laboratory Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430070, China; 00201543@whu.edu.cn (H.S.); zhanghao.1003@whu.edu.cn (H.Z.); xuanh.w@whu.edu.cn (X.W.); wsheng@whu.edu.cn (S.W.); fei.li@whu.edu.cn (F.L.); liujunyi@whu.edu.cn (J.L.)
* Correspondence: guohua.gou@whu.edu.cn

Abstract: Unmanned aerial vehicles (UAVs) are extensively employed for urban image captures and the reconstruction of large-scale 3D models due to their affordability and versatility. However, most commercial flight software lack support for the adaptive capture of multi-view images. Furthermore, the limited performance and battery capacity of a single UAV hinder efficient image capturing of large-scale scenes. To address these challenges, this paper presents a novel method for multi-UAV continuous trajectory planning aimed at the image captures and reconstructions of a scene. Our primary contribution lies in the development of a path planning framework rooted in task and search principles. Within this framework, we initially ascertain optimal task locations for capturing images by assessing scene reconstructability, thereby enhancing the overall quality of reconstructions. Furthermore, we curtail energy costs of trajectories by allocating task sequences, characterized by minimal corners and lengths, among multiple UAVs. Ultimately, we integrate considerations of energy costs, safety, and reconstructability into a unified optimization process, facilitating the search for optimal paths for multiple UAVs. Empirical evaluations demonstrate the efficacy of our approach in facilitating collaborative full-scene image captures by multiple UAVs, achieving low energy costs while attaining high-quality 3D reconstructions.

Keywords: multi-UAV; cooperation; continuous path planning; scene reconstruction; image capture



Citation: Sui, H.; Zhang, H.; Gou, G.; Wang, X.; Wang, S.; Li, F.; Liu, J. Multi-UAV Cooperative and Continuous Path Planning for High-Resolution 3D Scene Reconstruction. *Drones* **2023**, *7*, 544. <https://doi.org/10.3390/drones7090544>

Academic Editor: Geert Verhoeven

Received: 17 July 2023

Revised: 13 August 2023

Accepted: 17 August 2023

Published: 22 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Due to the rapid advancements in UAV technology, small commercial UAVs equipped with a single high-resolution camera are becoming more affordable. Typically, the great success achieved in image-based reconstruction [1–4] has led to the widespread adoption of UAVs for the image captures and 3D reconstructions of large scenes. Commercial software, such as DJI-Pilot [5] and Pix4Dcapture [6], offers the capability to automatically generate grid flight paths for the complete coverage of the designated area. Nevertheless, these paths typically capture scene images uniformly at a fixed height and viewing angle, resulting in challenges when capturing the detailed elevation information of buildings. Consequently, certain studies [7–13] have employed a two-stage “explore-and-exploit” approach. Initially, during the “explore” phase, images are rapidly captured using an overhead grid path, leading to the reconstruction of a coarse scene model known as a proxy. Subsequently, during the “exploit” phase, the optimal path for reconstruction is generated under the guidance of the proxy. These works have demonstrated that high quality 3D models can be effectively reconstructed using the “explore-and-exploit” strategy.

Most existing methods [7–13] follow a framework where the optimal set of viewpoints is initially determined during the “explore” phase. And paths are generated by solving the traveling salesman problem (TSP). However, this approach leads to the creation of non-continuous paths for a single UAV, necessitating frequent acceleration, deceleration, and hovering. Such behavior poses challenges in two main regards. Firstly, non-continuous

trajectories will result in the wasteful consumption of energy for UAVs designed for in-flight photography. Secondly, in the case of large-scale scenes, a single UAV is extremely inefficient. While multiple UAVs can concurrently capture images by dividing a single trajectory into segments, the potential for collisions among the UAVs must be acknowledged.

This paper introduces a methodology that tackles the previously mentioned challenges through the planning of multiple continuous trajectories. The objective is to ensure safety and minimize energy costs, thus enabling collaborative image captures by multiple UAVs. Our approach employs the “explore-and-exploit” strategy, with the difference that we introduce a framework for path planning centered around task and search principles. Specifically, we make key designs in three aspects. Firstly, regarding reconstructability estimation, we develop a submodular reconstructability heuristic and generate a reconstructability loss map (RLM) in the horizontal dimension to determine the priority task locations. Secondly, in terms of task allocation, we address the optimization of task sequences for multiple UAVs by minimizing a fitness function encompassing the corners and lengths of task sequences in a continuous real-numbered space. Thirdly, in path searching, we assess safety and energy costs based on trajectory dynamics. Concurrently, we optimize these factors along with scene reconstructability to generate optimal paths for multiple UAVs to the task locations.

We extensively evaluated our method in diverse synthetic [14] and real environments. Figure 1 displays the experimental results for some real-world scenarios. The experimental results demonstrate that our method enables collaborative image captures using multiple UAVs, leading to a significant improvement in image capture efficiency.

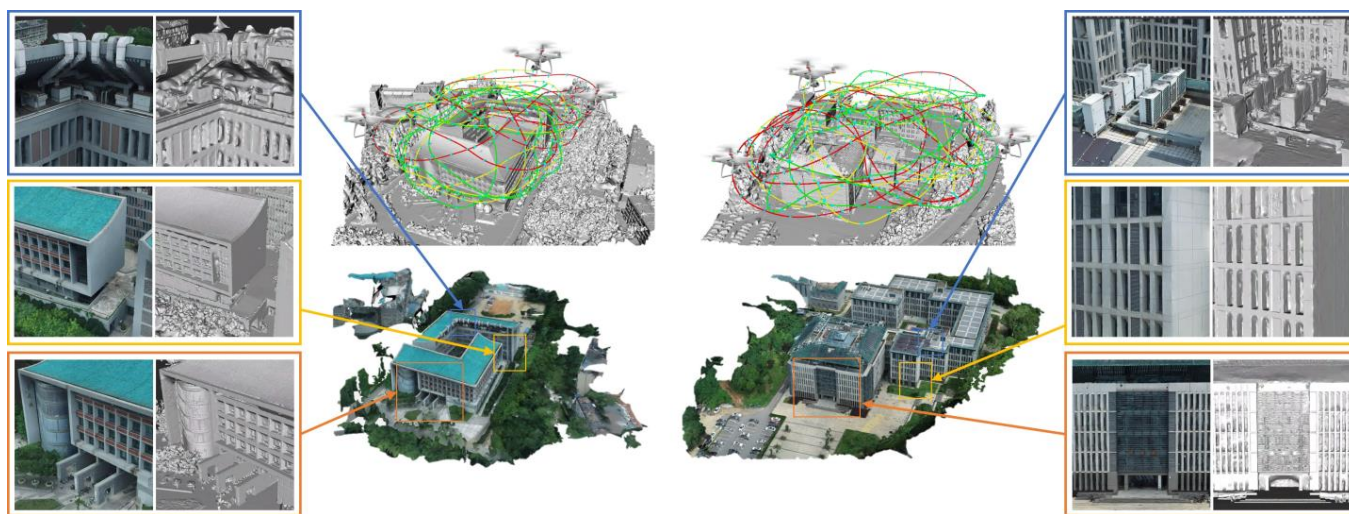


Figure 1. Our proposed method efficiently plans safe and continuous trajectories for three commercial UAVs equipped with a monocular camera and has excellent performance in terms of trajectory and reconstruction quality.

In summary, the main contributions of this paper are as follows:

- A path planning framework rooted in task and search principles, which distinguishes from prior research focused on path generation via TSP solutions is proposed. This framework enables the collaborative capture of scene images by multiple UAVs;
- A submodular heuristic-based reconstructability loss map is introduced for predicting global reconstructability. This map guides the identification of pivotal tasks’ locations, enhancing the overall optimality of reconstructions.
- A task allocation method driven by task sequence corners and lengths within the realm of real-numbered continuous space is proposed. This method promotes collaboration among multiple UAVs and curbs energy costs for continuous trajectories.

- A path searching method that concurrently optimizes trajectory safety, energy costs, and scene reconstructability is presented. This approach enhances trajectory and reconstruction quality while upholding safety standards.

2. Related Works

This section provides a summary and analysis of significant research in various domains, including path planning for scene reconstructions, task allocation for guiding multi-UAV collaborations, and path searching frameworks aimed at extending safe paths.

2.1. Path Planning for Scene Reconstruction

In the context of unknown scenarios lacking prior information, path planning for scene reconstructions often relies on widely adopted strategies such as the Next-Best-View (NBV) strategy [15–20] and the “explore-and-exploit” strategy [7–13].

2.1.1. NBV Strategy

The NBV strategy enables UAVs to predict, in real time during flight, the optimal viewpoint based on the currently explored area. This iterative process allows for the gradual extension of the trajectory until the complete scene is explored. Yamauchi et al. [18] introduced the concept of frontier and extended the trajectory by continuously searching for viewpoints that optimize the expansion of known area frontiers. Zhou et al. [19] presented a layered framework that employs a frontier information structure to systematically search for a path that covers the entire scene. Feng et al. [20] introduced a coarse structure prediction module, which enables them to plan a trajectory at a local level, thereby optimizing the reconstruction quality. However, all of these methods necessitate real-time onboard processing and rely on costly equipment capable of performing real-time depth computation. Therefore, these methods are impractical for low-cost commercial UAVs.

2.1.2. “Explore-and-Exploit” Strategy

The “explore-and-exploit” strategy has gained widespread adoption as a means to decrease dependence on high-cost hardware devices [7–13,21,22]. To guide the optimization of the trajectory, [9–11] developed mathematical models that approximated the actual reconstruction properties. Zhou et al. [12] introduced a novel Max–Min optimization method aimed at maximizing scene reconstructability using an equal number of viewpoints. Liu et al. [13] put forward the pioneering learning-based reconstructability predictor and employed it to guide UAV path planning. However, these methods only yield non-continuous paths that necessitate the UAV to hover at each viewpoint, resulting in substantial energy consumption. Zhang et al. [21] consequently modeled the correlation between trajectory turning angles and time consumption. They further incorporated time consumption and scene reconstruction quality to optimize the generation of a continuous trajectory. However, the elapsed time of the trajectory alone does not provide an accurate reflection of UAVs’ battery consumption. Furthermore, Zhang et al. [21] only considered a single UAV trajectory generation, which remained inefficient for accomplishing large-scale image capture tasks. While Zheng et al.’s path planning method [22] enables simultaneous image captures by multiple UAVs, the trajectories lack continuity and pose a risk of collisions among the UAVs. Consequently, this paper places emphasis on the generation of cooperative and energy-efficient trajectories for multiple UAVs.

2.2. Task Allocation for Multi-UAVs

Through judicious task allocation among multiple UAVs, a cooperative approach is adopted, resulting in a synergistic effect that surpasses the cumulative impact of individual contributions. Certain studies [23–25] utilized mixed integer linear programming models to determine the optimal task allocation solution. However, these models entail significant computational time when applied to large solution spaces. Consequently, Wang et al. [26] employed a heuristic multi-objective shuffled frog-leaping algorithm, utilizing matrix

binary encoding, to efficiently obtain an approximate optimal solution for the task allocation problem. Swarm intelligence algorithms are prevalent in multi-task allocations. Classical approaches like the genetic algorithm [27], ant colony optimization algorithm [28], and particle swarm algorithm [29] can achieve task allocation quickly, but they focus solely on optimizing theoretical task execution efficiency, disregarding the smoothness and energy cost of the UAV's continuous trajectory.

2.3. Path Searching Frameworks

To ensure the safe navigation of UAVs towards the target location, optimal paths should be sought within the available free space. Over the past few decades, a wide range of path-searching frameworks have been proposed, encompassing both sampling-based approaches [30–32] and grid-based methods [33–35]. LaValle et al. [30] introduced a notable sampling-based framework called the rapidly exploring random tree (RRT). This framework navigates the tree towards the target location through random sampling within the free space. Following the RRT, numerous enhanced variants were proposed, including RRT*, RRT-Connect [31], and RRG [32]. Nevertheless, due to their reliance on random sampling, these methods do not consistently yield the optimal path. Global optimization of the search process can be attained by discretizing the free space and converting the path search problem into a graph search problem. A* is the most representative of these frameworks, with widely used variants including JPS [33], ARA* [34], etc. Among these frameworks, Kurzer et al. [35] presented the Hybrid A*, which incorporates trajectory smoothing factors to enhance its suitability for generating continuous trajectories. However, these works are limited to pursuing a singular and 3D reconstruction-independent goal and fail to simultaneously optimize both scene reconstructability and trajectory energy costs, which constitutes another research focus of this paper.

3. Methodology

Our method takes the coarse proxy as input and comprises four primary steps: preliminary preparation, reconstructability estimation, task allocation, and path searching. The method ultimately generates multiple continuous trajectories and a high-quality 3D model, as illustrated in Figure 2.

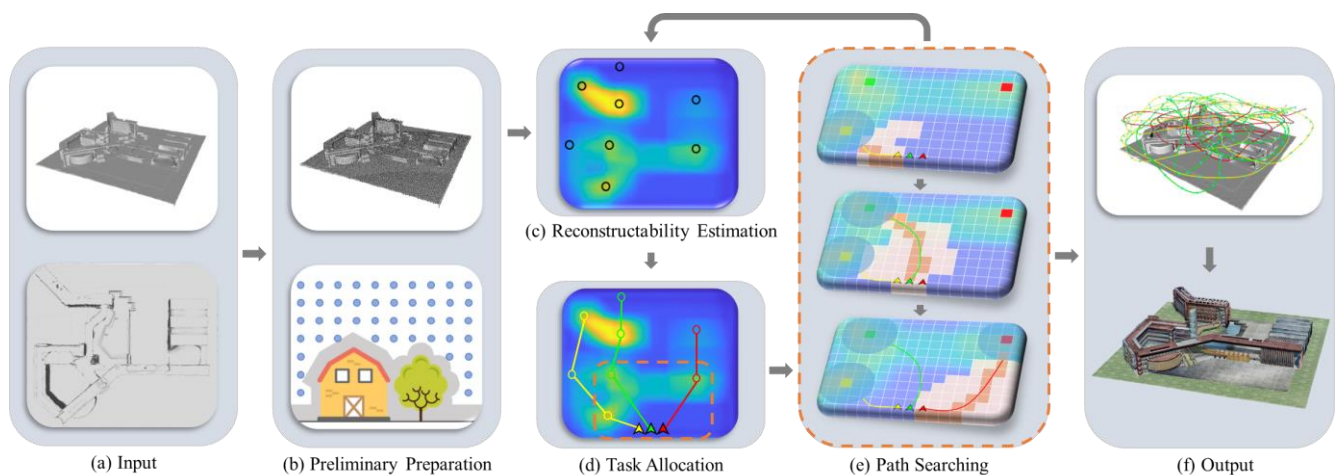


Figure 2. Overview of our method. (a) Coarse proxy. (b) The black dots on the upper proxy represent the surface points, and the solid blue circles below indicate the candidate viewpoints. (c) The RLM is shown, represented by a gradient heat map from blue (low) to yellow (high). The circles represent tasks. (d) The dashes indicate multi-UAV task sequences. (e) The light orange areas indicate the searched space, the dark orange areas depict the optimal path, and the curve represents the continuous trajectory. (f) The output multi-UAV continuous trajectories and reconstruction results.

In the preliminary preparation phase (b), we uniformly sample N surface points on the surface of the input proxy, where each surface point is considered as a representative of its surrounding region. Additionally, we generate candidate viewpoints V_{candi} uniformly in the safe space, facilitating the subsequent path search in phase (e). In the reconstructability estimation phase (c), we establish a submodular reconstructability heuristic to estimate the reconstruction effect of each surface point. The heuristic is employed to generate a global RLM. Furthermore, we identify critical regions within the RLM to serve as multi-UAV tasks. In the task allocation phase (d), we transfer the discrete task assignment problem to a continuous real number space and obtain the optimal task sequence for multiple UAVs by minimizing the fitness function. In the path searching phase (e), we employ a novel A* algorithm to search and extend the trajectories of each UAV in the candidate viewpoints (V_{candi}) to visit their respective task targets sequentially by task sequences. In this extension process, we optimize by considering trajectory energy costs alongside scene reconstruction contributions, among other factors, achieving high-quality multi-UAV path generations. Reconstructability estimation is repeated once all task targets have been visited, continuing until the scene reconstructability reaches the target or the multi-UAV trajectory's energy cost surpasses the threshold.

3.1. Reconstructability Estimation

This section provides a comprehensive description of the reconstructability estimation method for surface points, denoted as $S = \{s_i\}_{i=1,\dots,N}$, in the presence of a viewpoint set, $V = \{v_i\}_{i=1,\dots,M}$, where M represents the number of viewpoints. The approach involves the formulation of reconstructability heuristics (Section 3.1.1) and the creation of a reconstructability loss map to localize tasks for multi-UAVs (Section 3.1.2).

3.1.1. Reconstructability Heuristic

By establishing a heuristic relationship between the viewpoints and the reconstructability of the scene, the reconstruction effect of the scene can be quickly predicted [10]. As shown in Figure 3, the reconstructability contribution of the viewpoint pair (v_i, v_j) to the surface point s_k is defined as follows:

$$w(s_k, v_i, v_j) = w_1(\alpha)w_2(\alpha)w_3(d_m) \cos(\theta_m), \quad (1)$$

where the distance $d_m = \max(\|\overrightarrow{s_k v_i}\|, \|\overrightarrow{s_k v_j}\|)$ and the angle $\theta_m = \max(\theta_i, \theta_j)$. w_1 , w_2 , and w_3 are, respectively, defined as

$$w_1(\alpha) = (1 + \exp(-k_1 \cdot (\alpha - \alpha_1)))^{-1}, \quad (2)$$

$$w_2(\alpha) = 1 - (1 + \exp(-k_2 \cdot (\alpha - \alpha_2)))^{-1}, \quad (3)$$

$$w_3(d_m) = 1 - \min\left(\frac{d_m}{d_{\max}}, 1\right). \quad (4)$$

Here, we set the parameters $k_1 = 32$, $\alpha_1 = \frac{\pi}{16}$, $k_2 = 8$, and $\alpha_2 = \frac{\pi}{4}$ as suggested by Smith et al. [10]. The parameter d_{\max} represents the maximum observable distance of a viewpoint, and typically, a smaller value of d_{\max} leads to improved reconstruction accuracy. Smith et al. [10] proposed an additive heuristic to quantify the collective reconstructability contribution of the viewpoint set $V = \{v_i\}_{i=1,\dots,M}$ to the surface point s_k :

$$h(s_k, V) = \sum_{\substack{i=1, \dots, M \\ j=i+1, \dots, M}} \delta(s_k, v_i) \delta(s_k, v_j) w(s_k, v_i, v_j). \quad (5)$$

Here, the visibility function, denoted as $\delta(s, v)$, determines whether the surface point s is within the field of view of the viewpoint v . If s is not visible from v , $\delta(s, v)$ is set to 0; otherwise, it is set to 1.

However, the reconstructability of surface points tends to exhibit diminishing returns [36]. In other words, as the existing reconstructability of a surface point increases, the additional gain from an extra viewpoint decreases, demonstrating a submodular characteristic [37]. Consequently, when the scene reconstructability is high, the additive heuristic proposed by [10] may struggle to accurately estimate the true reconstruction impact on surface points. For this reason, we improve the total reconstructability contribution of the viewpoint set V to the surface point s_k as

$$h'(s_k, V) = 2 \cdot h_{\max} \cdot \left(0.5 - (1 + \exp(k_3 \cdot h))^{-1}\right). \tag{6}$$

Here, h_{\max} represents the maximum reconstructability value for surface point s_k . As h approaches infinity, the reconstructability heuristic $h'(s_k, V)$ converges to h_{\max} , indicating a decrease in the reconstruction depth error of the surface point. Thus, the contribution of a single viewpoint, v_i , to the overall reconstructability of the surface points S within the entire scene can be expressed as

$$\langle(S, V, v_i) = \sum_{k=1, \dots, N} \frac{h_0(s_k, V, v_i)}{h(s_k, V)} \cdot h'(s_k, V), \tag{7}$$

where

$$h_0(s_k, V, v_i) = \sum_{j=1, \dots, i-1, i+1, \dots, M} \delta(s_k, v_i) \delta(s_k, v_j) w(s_k, v_i, v_j). \tag{8}$$

To improve the efficiency of estimating scene reconstruction effects, we integrate a submodularization feature into the additive reconstructability heuristic. This augmentation benefits our path planning algorithm in two aspects. First, it enhances the accurate localization of key task regions within the scene (Section 3.1.2). Second, it enhances the quality of scene reconstructions by employing the submodular reconstructability heuristic to ascertain the viewing direction for each viewpoint (Section 3.3.3).

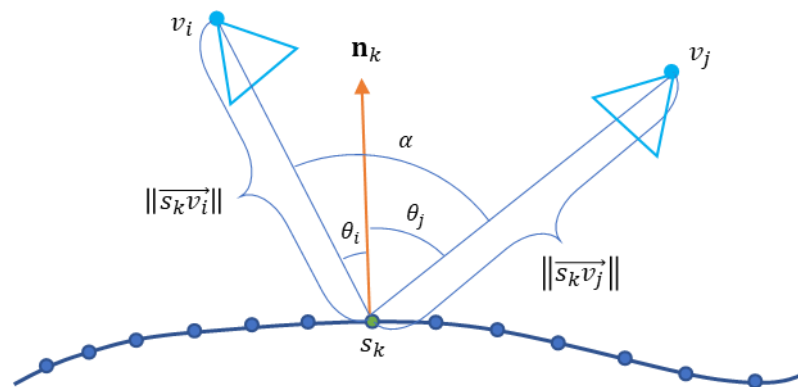


Figure 3. Reconstructability heuristics. This figure shows the profile view when viewpoints v_i, v_j are simultaneously observing the surface point s_k . \mathbf{n}_k denotes the surface normal of s_k , while θ_i and θ_j represent the angles between the vectors $\vec{s_k v_i}, \vec{s_k v_j}$, and \mathbf{n}_k . α denotes the angle between $\vec{s_k v_i}$ and $\vec{s_k v_j}$.

3.1.2. Reconstructability Loss Map and Task Localization

We create a reconstructability loss map for the efficient retrieval of reconstructability loss data across any scene region. This map aids our path planning algorithm in identifying crucial image capture points as task targets, thereby mitigating local optimization issues related to reconstruction.

To construct the RLM, we undertake two primary steps: Firstly, we partition the scene’s point cloud, sized $X \times Y$ in the horizontal dimension, using resolution r_0 . This generates a grid denoted as $G = \{g_{i,j}\}_{\substack{i=1,\dots,N_x \\ j=1,\dots,N_y}}$, as depicted in Figure 4a. Here, $N_x = \lceil \frac{X}{r_0} \rceil$ and $N_y = \lceil \frac{Y}{r_0} \rceil$ indicate the count of grid cells along the x and y directions, respectively. The scene’s point cloud comprises surface points denoted as S . Secondly, we evaluate the reconstructability loss value for every cell within grid G . With regard to the set of viewpoints V , the reconstructability loss within a region encompassing a radius r_1 , centered on grid cell $g_{i,j}$, is formulated as follows:

$$\mathcal{G}(S, V, g_{i,j}) = \sum_{k=1,\dots,M} (h_{\max} - h'(s_k, V)) \cdot \rho_1(g_{i,j}, s_k), \tag{9}$$

where $\rho_1(g_{i,j}, s_k)$ represents the distance evaluation function. If the distance between the surface point s_k and the position of $g_{i,j}$ is greater than r_1 , $\rho_1(g_{i,j}, s_k) = 0$; otherwise, $\rho_1(g_{i,j}, s_k) = 1$. Typically, the radius r_1 is determined based on the maximum visible distance d_{\max} . In our study, we set it as $\frac{3d_{\max}}{4}$. Upon completing the computation of all grid cells in G , an RLM is generated, as depicted in Figure 4b.

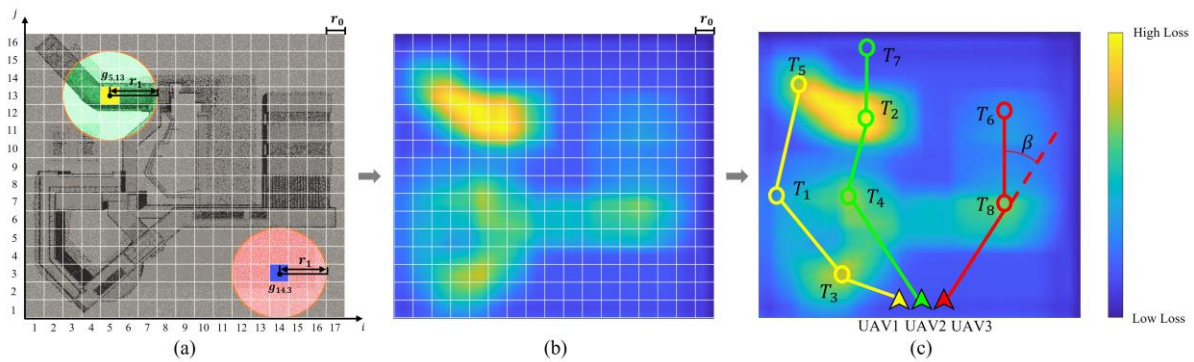


Figure 4. Reconstructability loss map generation principle and multi-UAV task allocation. (a) shows the point cloud composed of surface points S , where the green point clouds fall within the computation range of grid $g_{5,13}$ and the red point clouds fall within the computation range of grid $g_{14,3}$. In (b), the RLM is represented by a heat map from blue (low) to yellow (high). (c) displays circles that denote the centers of high loss areas in the RLM, which are utilized as task targets for multiple UAVs. The different-colored lines indicate the task sequences for different UAVs.

In the task localization method, we begin by sorting the reconstructability loss values of all grid cells in the RLM. Then, we select N_t cells with the highest loss values among them, ensuring that the distance between each selected cell exceeds the maximum visible distance, d_{\max} . The locations of these cells are designated as the task targets $\mathcal{T} = \{T_1, T_2, \dots, T_{N_t}\}$ for the multi-UAV system, as depicted in Figure 4c.

To summarize, we construct the RLM by calculating the sum of reconstructability losses in each region in the horizontal dimension, which allows us to make a quick sense of the global reconstructability. In addition, we consider a number of locations with the worst reconstructability as task targets, which enables the UAV to capture the scene images more uniformly and thus reconstructs a higher-quality 3D model within a limited flight time.

3.2. Task Allocation

We steer the collaborative image captures by multiple UAVs through the allocation of optimal task sequences. Our approach involves two primary design aspects. Firstly, to convert the discrete task allocation problem for multiple UAVs into a continuous real number space, we formulate a set of rules that serve as a codec for the task sequence-solution space. Considering N_{uav} UAVs and N_t tasks, we treat any real vector with N_t dimensions and

values within the interval $(1, N_{\text{uav}} + 1)$ as a potential solution for task allocation. Each task is uniquely represented by a real number within the solution, serving as a code. Tasks associated with the same integer code are designated for execution by the same UAV, with the task execution order determined by the numerical magnitude of the code.

In our implementation, considering 3 UAVs and 8 task targets, the solution vector at a specific instance is given as $[1.56 \ 2.80 \ 1.23 \ 2.02 \ 1.79 \ 3.20 \ 2.91 \ 3.11]^T$. Table 1 illustrates the mapping between individual tasks and codes. Following the decoding of this solution, we acquire the task execution sequence for each UAV, as demonstrated in Table 2.

Table 1. Multi-UAV task sequences-solution space coding table.

Task	Code	Task	Code
T_1	1.56	T_5	1.79
T_2	2.80	T_6	3.20
T_3	1.23	T_7	2.91
T_4	2.02	T_8	3.11

Table 2. Solution space-multi-UAV task sequences decoding table.

UAV ID	Code Comparison	Task Sequence
1	$1.23 < 1.56 < 1.79$	$T_3 \rightarrow T_1 \rightarrow T_5$
2	$2.02 < 2.80 < 2.91$	$T_4 \rightarrow T_2 \rightarrow T_7$
3	$3.11 < 3.20$	$T_8 \rightarrow T_6$

Secondly, we introduce a novel fitness function. Our objective is to ensure that the task allocation outcomes facilitate the creation of seamless, energy-efficient continuous trajectories. By minimizing the fitness function, we anticipate achieving several effects as outlined below:

1. Minimize β_{max} , the maximum inflection angle. As shown in Figure 4c, a significant inflection angle β can result in high flight energy costs;
2. Minimize d_{ave} , the average of the distances between adjacent tasks. This objective ensures that each UAV gives higher priority to tasks closer to its current location. Notably, when calculating d_{ave} , it is important to consider the distance from the starting point to the first task;
3. Minimize d_{delta} , the difference between the longest and shortest distances in multiple task sequences. This ensures that the path lengths of multiple UAVs are as similar as possible.

Therefore, the fitness function is designed as follows:

$$f(\mathcal{T}, \mathcal{S}) = k_4\beta_{\text{max}} + k_5d_{\text{ave}} + k_6d_{\text{delta}}, \quad (10)$$

where \mathcal{S} is the starting state of the multi-UAV, including the starting position and velocity direction. k_4 , k_5 , and k_6 are the weight parameters.

We establish a mapping of task sequences into a continuous real solution space by formulating codec rules for the task sequence-solution space. This mapping enables us to employ swarm intelligence algorithms [27–29,38] for the swift acquisition of the vector solution that corresponds to the fitness function's minimum value. Post-decoding, we can retrieve the optimal task sequences for multiple UAVs. These innovations not only reduce the time required for task allocation but also empower the task allocation outcomes to inform the creation of energy-efficient paths.

3.3. Path Searching

Subsequent to the task allocation outcomes, the multiple UAVs embark on sequential path search procedures. Individual UAVs visit designated task targets according to their

respective task sequences. Throughout this process, we assess the safety and energy costs of continuous trajectories (Section 3.3.1) and employ a novel A* algorithm (Section 3.3.2) to prolong the trajectory towards the task target's location. This extension aims to minimize an objective function that integrates energy costs, safety considerations, and contributions to scene reconstructability (Section 3.3.3).

3.3.1. Safety and Energy Costs

To facilitate our path planning algorithm in generating safe and energy-efficient trajectories for multiple UAVs, we must assess both the safety and flight energy costs associated with continuous trajectories for these UAVs. A continuous trajectory traversing all viewpoints [39–41] can be expressed as $\mathcal{L}(t) = \{L_k(t)\}_{k=x,y,z}$, with $L_k(t)$ representing a smooth curve tracing the trajectory's coordinates in dimension k as a function of time t . Typically, this curve is represented using an n -degree polynomial, which also embodies a dynamic characteristic of the continuous trajectory [39]:

$$L_k(t) = \sum_{i=0,1,\dots,n} a_{k,i}t^i, \quad (11)$$

where $a_{k,i}$ represents the polynomial coefficient of the curve. In this work, we set $n = 5$.

To ensure the safety of continuous trajectories, we compute the real-time distance, denoted as d_{uav} , between any two UAVs using the following equation:

$$d_{\text{uav}}(\mathcal{L}, \mathcal{L}', t) = \sqrt{\sum_{k=x,y,z} (L_k(t) - L'_k(t))^2}. \quad (12)$$

Here, \mathcal{L} and \mathcal{L}' denote the flight trajectories of distinct UAVs, respectively. To prevent collisions between multiple UAVs, it is crucial to ensure that the minimum value of d_{uav} is consistently greater than the safe distance. Although the state-of-the-art methods [9,10,12,21] enable the simultaneous capture of image data by dividing a single trajectory into multiple segments, they still entail the risk of collision among multiple UAVs.

To evaluate the energy costs of continuous trajectories, we calculate the squared integral of the acceleration trajectory's derivative:

$$\mathcal{J}(\mathcal{L}) = \sum_{k=x,y,z} \int_0^\tau (L_k^{(3)}(t))^2 dt, \quad (13)$$

where τ represents the duration of the trajectory. Equation (13) can be employed to represent the dynamical continuity and smoothness of the trajectory throughout its duration. A lower value of $\mathcal{J}(\mathcal{L})$ indicates a shorter trajectory duration and smoother acceleration/deceleration, resulting in reduced trajectory energy costs. Therefore, in this study, we regard $\mathcal{J}(\mathcal{L})$ as an approximation and quantification of the trajectory's energy costs.

In summary, our approach involves calculating the real-time distance between any two UAVs and evaluating trajectory smoothness based on the dynamics of the continuous trajectory. These calculations serve to quantify both the safety and energy costs of the multi-UAV continuous trajectory. This integrated assessment enables us to optimize trajectory energy costs while simultaneously ensuring the safety of multiple UAVs during the path searching process.

3.3.2. Searching Process

We have developed a novel A* algorithm to extend the path of UAV u towards the region where the task target T is located. The schematic diagram and pseudo-code of the algorithm are presented in Figure 5 and Algorithm 1, respectively. The algorithm follows a similar framework to the traditional A*, and the process is outlined as follows: We begin by defining two sets of viewpoints, *Open_Set* and *Closed_Set*. Additionally, we assign a *Score* attribute to each candidate viewpoint in V_{candi} . Following initialization, the viewpoint v_{curr}

with the lowest *Score* in *Open_Set* is successively moved to *Closed_Set* until it approaches the task goal T . If v_{curr} is not yet close to T , we search for neighboring viewpoints V_{neib} in the candidate viewpoints V_{candi} that are not in *Closed_Set* and are in a safe space. The viewpoints in V_{neib} that are not part of *Open_Set* are added to *Open_Set*, and we calculate the score S for each neighboring viewpoint v_{neib} . If S is lower than the original *Score* of v_{neib} , we update the *Score* of v_{neib} to S and its source viewpoint to v_{curr} .

Algorithm 1: Path_Searching

Input : $\mathcal{P} = \{P_i = \{v_{i,k}\}_{k=1}^{M_i}\}_{i=1}^{N_{uav}}$, initial paths of the multi-UAV, where $v_{i,k}$ is the k -th viewpoint of UAV i and M_i is the number of viewpoints on the initial path P_i of UAV i ;
 u , the ID of the UAV that needs to perform the path searching;
 V_{candi} , candidate viewpoints in the scene;
 T , the current task;
 $d_{ext}, d_{end}, r_{neib}$, distance parameters related to path searching;

Output: P_{final} , the final trajectory of the UAV u .

```

1  Generate Closed_Set, Open_Set, and add the last viewpoint  $v_{u,M_i}$  of the initial path  $P_u$  of the UAV  $u$  as the
   starting point into Open_Set;
2  Update the Score of all viewpoints in  $V_{candi}$  to Inf;
3  while Open_Set is not empty do
4      Move the viewpoint  $v_{curr}$  with the lowest Score from the Open_Set to the Closed_Set;
5      if the distance between  $v_{curr}$  and  $T$  is less than  $d_{end}$  then
6          break;
7      end
8       $\mathcal{L}_{curr} \leftarrow$  Obtain the path with  $v_{curr}$  as the end viewpoint by backtracking  $v_{curr}$ , and generate the
       continuous trajectory [39–41];
9       $D_{curr} \leftarrow$  End tangent direction of  $\mathcal{L}_{curr}$ ;
10      $v_{ext} = v_{curr} + d_{ext} \times D_{curr}$ ;
11      $V_{neib} \leftarrow$  Find neighbors based on K-NN in  $V_{candi}$  with  $v_{ext}$  as the center and  $r_{neib}$  as the radius;
12     For each  $v_{neib}$  in  $V_{neib}$  do
13         if  $v_{neib}$  is not in Closed_Set and is in safe space then
14              $S \leftarrow$  Calculate the score of  $v_{neib}$ ; % see Section 3.3.3
15             if  $v_{neib}$  is not in Open_Set then
16                 Insert  $v_{neib}$  into Open_Set;
17             end
18             if  $S <$  the Score of  $v_{neib}$  then
19                 Update  $v_{neib}$ 's source viewpoint to  $v_{curr}$ ;
20                 Update the Score of  $v_{neib}$  to  $S$ ;
21             end
22         end
23     end
24 end
25  $P_{final} \leftarrow$  Tracing the  $v_{curr}$  to get the final path of UAV  $u$ ;

```

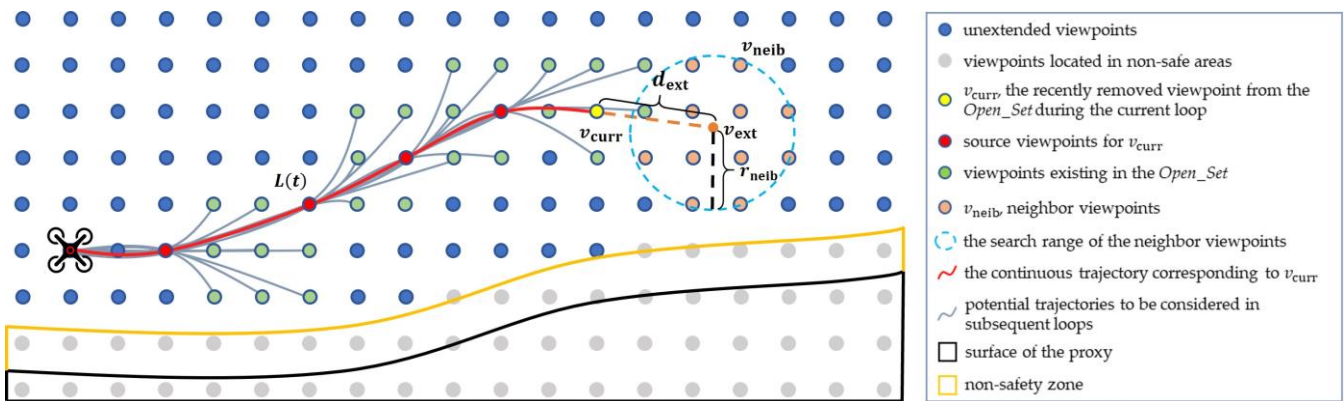


Figure 5. Schematic diagram of path search and extension. The solid circles, depicted in various colors, represent candidate viewpoints.

In contrast to the traditional A*, firstly, our approach draws inspiration from Hybrid A* [35]. We prioritize trajectory smoothness by refining the process of locating neighboring viewpoints. The process begins by backtracking the viewpoint v_{curr} , generating a continuous trajectory denoted as \mathcal{L}_{curr} . Furthermore, we extend the distance d_{ext} from v_{curr} along the end tangent direction of \mathcal{L}_{curr} to reach the position v_{ext} . Finally, within V_{candi} , we perform a K-NN search to find multiple nearest neighbors V_{neib} using v_{ext} as the center and r_{neib} as the radius. Secondly, to calculate the score of viewpoints, we design an objective function that incorporates factors such as trajectory energy costs, scene reconstructability, and flight safety. The detailed description of the specific objective function can be found in Section 3.3.3.

In essence, building upon the foundation of the classical A* algorithm, we curtail the direction and extent of the continuous trajectory expansion. This is achieved by strategically situating neighboring viewpoints along the tangent line at the trajectory's terminus, thereby guaranteeing heightened trajectory smoothness. Furthermore, our objective function enables the concurrent optimization of the trajectory and reconstruction quality.

3.3.3. Objective Function

As described in Section 3.3.2, the objective function is utilized in each path searching loop to calculate the score of v_{neib} , which represents a neighboring viewpoint of v_{curr} . This score guides the selection of viewpoints and the extension of paths. The objective function is designed to achieve the following effects on path guidance:

Close to the task target. To guide the current path towards the region where the task target T is located, we aim to predict the minimum energy cost of reaching T after adding v_{neib} to the path. A lower cost indicates that the UAV is closer to the task target. Therefore, we backtrack from v_{curr} to acquire the path denoted as P_{curr} . Furthermore, we extend P_{curr} by adding v_{neib} , along with a viewpoint sharing both the horizontal position of T and the height of v_{neib} . This augmentation results in the creation of P_{pre} , serving as the foundation for generating a continuous trajectory labeled as \mathcal{L}_{pre} which originates at the initial path point, traverses v_{curr} , encompasses v_{neib} , and culminates in reaching T . The energy cost $\mathcal{J}(\mathcal{L}_{pre})$ represents the minimum value required to reach T . However, we observed in practice that when T is distant from the search starting point, the trajectory may take a considerable amount of time to extend to T . Consequently, we design an exponential function as a component of the objective function:

$$W_1(v_{curr}, v_{neib}, T) = \exp(b_1(\mathcal{J}(\mathcal{L}_{pre}) - \mathcal{J}(\mathcal{L}_{start}))), \quad (14)$$

where \mathcal{L}_{start} denotes the initial continuous trajectory of the path before initiating the path search process. The parameter b_1 corresponds to W_1 .

Maximize reconstruction contribution. In order to calculate the score of v_{neib} , it is necessary to determine the direction with the highest reconstruction contribution for v_{neib} , which requires the design of a contribution evaluation function. Our reconstructability heuristic indicates that the reconstructability of a surface point can only be increased if it is effectively observed by at least two viewpoints simultaneously. If we defined the contribution of a viewpoint as solely the improvement in reconstructability of surface points, the viewpoints would tend to observe regions that have already been explored, resulting in inadequate scene coverage. Therefore, we introduce a coverage attribute U_i for each viewpoint v_i to keep track of the number of new surface points observed by v_i . The contribution evaluation function of viewpoint v_i is

$$c(S, V, v_i) = \mathcal{H}(S, V, v_i) + k_7 \cdot U_i, \quad (15)$$

where V represents all the viewpoints selected currently, encompassing v_{neib} , the viewpoints in P_{curr} , and viewpoints from other paths. k_7 denotes the coverage weight. The incorporation of the coverage attribute aims to stimulate viewpoints to actively observe previously unexplored regions.

Maximize the average contribution of each viewpoint to the scene. Excessive density of viewpoints not only increases the reconstruction elapsed time but can also introduce errors that reduce the reconstruction quality [11]. Hence, it is crucial to minimize the number of viewpoints while striving to maximize the trajectory's contribution to the scene. To achieve this, we incorporate W_2 as a component of the objective function:

$$W_2(P_{\text{neib}}, S, V) = \frac{\sum_{i=1, \dots, M_{\text{neib}}} c(S, V, v_i)}{M_{\text{neib}}}, \quad v_i \in P_{\text{neib}}, \quad (16)$$

where P_{neib} represents the path obtained by appending v_{neib} to the end of P_{curr} and M_{neib} denotes the number of viewpoints present on this path.

Maximize the contribution to the scene per unit of energy cost. One of our objectives is to capture scene images that result in a higher quality 3D model while minimizing the energy cost. Therefore, we aim to maximize the reconstruction contribution of the trajectory per unit of energy cost. This is achieved through the design of W_3 as follows:

$$W_3(P_{\text{neib}}, S, V) = \frac{\sum_{i=1, \dots, M_{\text{neib}}} c(S, V, v_i)}{\mathcal{J}(\mathcal{L}_{\text{neib}})}, \quad v_i \in P_{\text{neib}}, \quad (17)$$

where $\mathcal{L}_{\text{neib}}$ represents the continuous trajectory that traverses all the viewpoints in P_{neib} .

Consequently, the objective function can be designed as follows:

$$W(S, T, v_{\text{curr}}, v_{\text{neib}}) = (W_1(v_{\text{curr}}, v_{\text{neib}}, T) - b_2 \cdot W_2(P_{\text{neib}}, S, V) - b_3 \cdot W_3(P_{\text{neib}}, S, V)) \cdot \rho_2(\mathcal{L}_{\text{neib}}), \quad (18)$$

Here, b_2 and b_3 represent weight parameters. ρ_2 signifies the safety evaluation function for the trajectory $\mathcal{L}_{\text{neib}}$. Real-time computation of the minimum separation between $\mathcal{L}_{\text{neib}}$ and ongoing continuous paths of multiple UAVs is determined using Equation (12). When the computed distance falls below the safety threshold, ρ_2 is assigned the value of Inf; otherwise, it takes on the value of 1.

In this work, we maintain an ongoing influence over viewpoint selection and trajectory extension, driven by the minimization of the designated objective function, denoted as W . The objective function encompasses energy costs, contributions to reconstructability, and safety considerations. This approach not only ensures that the trajectories of our multi-UAV system achieve efficiency in terms of energy costs but also ensures the secure acquisition of scene images with substantial reconstruction contributions.

4. Experiments

We conducted a comprehensive series of experiments in both synthetic and real environments to validate the effectiveness of our approach. Firstly, we introduce the

dataset utilized in the experiments (Section 4.1). Subsequently, we provide a description of the hardware devices employed in the experiments, along with the implementation details (Section 4.2). Following that, we conduct a self-evaluation of various components of our method (Section 4.3). Lastly, to demonstrate the superiority of our approach, we compare it with the state-of-the-art methods in both synthetic (Section 4.4) and real environments (Section 4.5).

4.1. Benchmark

To conduct a comprehensive evaluation and comparison with advanced methods, we utilized the UrbanCity dataset, published by Lin et al. [14]. This dataset encompasses diverse synthetic scenes and includes trajectory and image data generated through a variety of state-of-the-art technologies [10,12,21], along with an oblique photography method for each scene. This facilitates convenient comparison experiments for our study. Three representative scenes with distinct characteristics (School, Town, and Castle) were chosen from UrbanCity, as depicted in Figure 6. These three scenes encompass the majority of prevalent artificial building types.

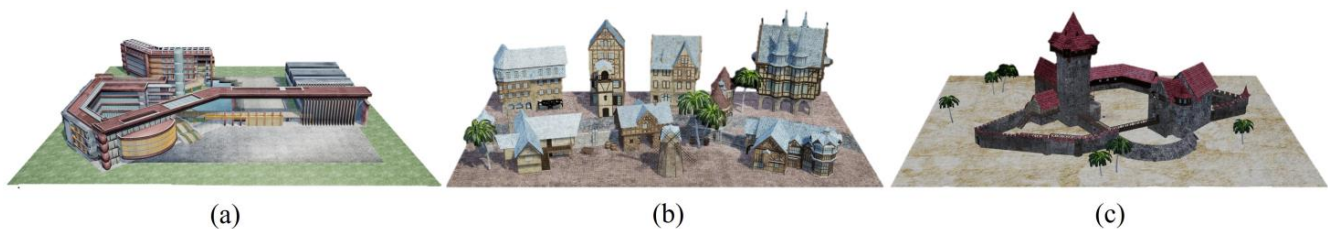


Figure 6. Three selected representative synthetic scenes: (a) School, characterized by its expansive size and modern buildings with unique architectural designs. (b) Town, a compact area densely populated with buildings. (c) Castle, an open space featuring a bamboo structure.

4.2. Experiment Details

This section outlines the experiment’s details. We commence by elucidating the hardware and software employed. Subsequently, we expound upon the methodology of obtaining the coarse proxy. Following this, a comprehensive depiction of parameter settings ensues. Conclusively, we explicate the evaluation metrics adopted within the experimental framework.

4.2.1. Hardware and Software

The algorithm is executed on a computer equipped with an 11th Gen Intel® Core™ i7-11700 @ 2.50 GHZ CPU, 32 GB RAM, and NVIDIA GeForce RTX 3080 Ti GPU. For real scenario experiments, we employed three DJI Phantom 4 RTK UAV devices, each equipped with a single camera featuring focal lengths ranging from 8.8 mm to 24 mm. We utilized DasEarth [42] to reconstruct 3D models from captured images. In the interest of fairness, all reconstructed models in the experiments were generated using DasEarth’s default settings.

4.2.2. Coarse Proxy

In real scenes, we employ the DJI-Pilot [5] to automatically generate a vertical photography path that covers the target area. Coarse proxies are reconstructed from the captured images. The UrbanCity dataset [14] provides four levels of precision proxies for each synthetic scene, ranging from coarse to fine: *box*, *coarse*, *inter*, and *fine*. The *inter* proxy closely resembles the reconstructed effect achieved using vertical photography. Consequently, all path planning experiments conducted in the synthetic scenes were based on *inter*-level precision proxies.

4.2.3. Parameter Settings

For the reconstructability heuristic, we set $h_{\max} = 20$ and $k_3 = 0.24$. For the reconstructability loss map, we defined the map resolution as $r_0 = 5$. During the task allocation

process, we assigned 12 tasks ($N_t = 12$) for large-scale scenes and 7 tasks ($N_t = 7$) for small-scale scenes. For the fitness function in the task allocation, we determined $k_4 = 0.2$, $k_5 = 0.8$, and $k_6 = 0.1$ based on experience. In Algorithm 1, we set $d_{\text{ext}} = 7$ and $r_{\text{neib}} = 5$ for small-scale scenes and $d_{\text{ext}} = 10$ and $r_{\text{neib}} = 7$ for large-scale scenes. We consider the path to be in proximity to the target when the horizontal distance between the path and the task target is below $d_{\text{end}} = 30$. Based on experience, we assigned the weight parameters in the objective function as follows: $b_1 = 9$, $b_2 = 15$, $b_3 = 1$, and $k_7 = 0.4$. Finally, for the termination condition of our method, we set to output multi-UAV trajectories when the proportion of surface points with a reconstructability reaching 12 exceeds 92% or the total energy cost of multiple trajectories surpasses 65.

4.2.4. Evaluation Metrics

We employed two evaluation metrics, Error and Completeness, proposed in [10], to quantify the disparity between the reconstruction results and the ground truth. Error is determined by calculating the closest distance between each surface point of the reconstruction result and the ground truth surface. The distances are sorted from smallest to largest, and the values corresponding to the 90th and 95th percentiles are designated as Error 90% and Error 95%, respectively. Meanwhile, Completeness is evaluated based on the surface points of the ground truth. We calculate the closest distance between each surface point of the ground truth and the reconstructed surface. The percentage of surface points with distances less than 0.020 m, 0.050 m, and 0.075 m relative to the total are referred to as Completeness 0.020 m, Completeness 0.050 m, and Completeness 0.075 m, respectively. These metrics indicate that lower Error and higher Completeness values indicate a better reconstruction quality. Additionally, we assess the quality of the multi-UAV trajectories based on the total length, total energy cost, and maximum time cost.

4.3. Self Evaluation

This study introduces a path planning framework rooted in task-oriented and search-based principles. The framework encompasses three distinct modules: reconstructability estimation, task allocation, and path searching. Within this section, we subject each module to isolated testing for individual assessment. Additionally, we conduct a direct comparative analysis of each module's impact on the path planning framework through ablation experiments, as elaborated in Section 4.3.4. Finally, we present a comprehensive evaluation of the method's collective efficacy in Section 4.3.5.

4.3.1. Reconstructability Estimation

During the reconstructability estimation phase, we assess the scene reconstruction effectiveness through the formulation of a reconstructability heuristic. In contrast to the simple additive heuristic [10], we incorporate submodular features into this heuristic. To establish that the reconstructability heuristic can more effectively predict reconstruction depth errors following the integration of submodular features, we formulate the subsequent experiment:

Firstly, we created 30 viewpoints facing a wall within a virtual environment (see Figure 7, top left). These viewpoints were randomized in terms of both position and viewing direction. We reconstructed a 3D model of the wall using images captured from these viewpoints. Additionally, we conducted a dense sampling of the ground truth surface of the wall. For each surface point, we calculated the closest distance to the reconstructed surface. This calculation allowed us to determine the reconstructed depth error for all surface points, as depicted in the upper right panel of Figure 7. Finally, we estimated the reconstructability of the ground truth surface points using two distinct heuristics: the additive heuristic [10] and our developed submodular heuristic. The resulting estimates are displayed in the bottom two panels of Figure 7.

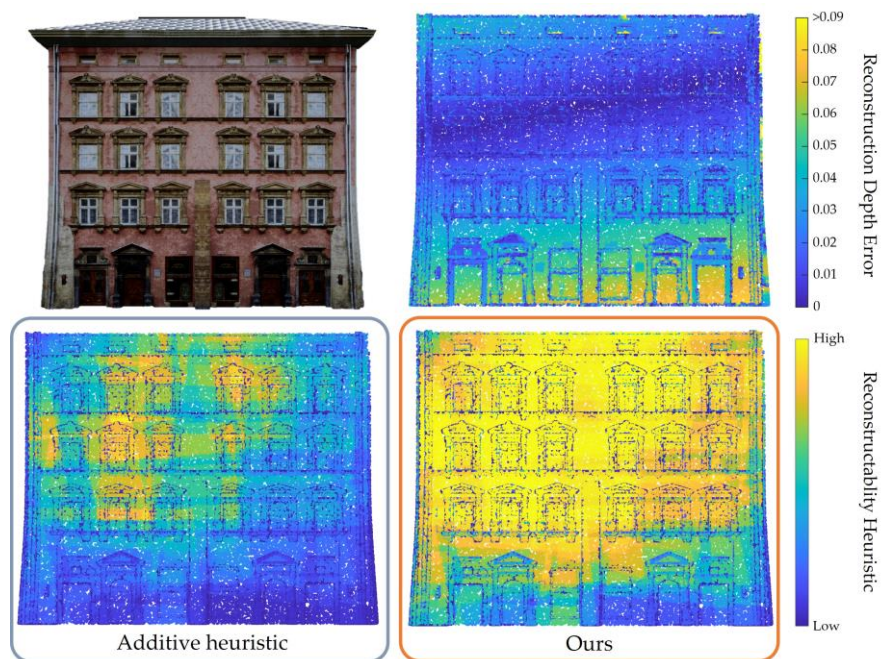


Figure 7. Comparison of the reconstructability heuristic. The **upper left panel** displays the ground truth of the wall, while the **upper right panel** presents the reconstruction depth error. The **bottom two figures** show the reconstructability estimation results of the additive heuristic [10] and our submodelized heuristic, respectively.

Furthermore, we computed the mean reconstruction depth error for each 0.01 interval of the reconstructability heuristic. We plotted the resulting trend of the mean reconstruction depth error in relation to the reconstructability heuristic. This visualization, presented in Figure 8, serves to illustrate the connection between the actual reconstruction error and the estimated reconstructability.

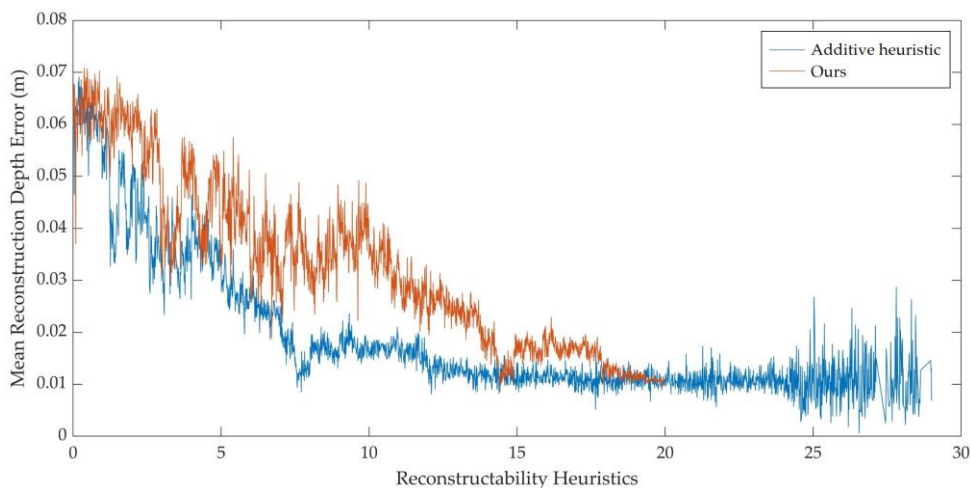


Figure 8. The relationship between the mean reconstruction depth error and different reconstructability heuristics.

In Figure 7, the reconstruction depth error plot uses a blue color to represent a low error, while the reconstructability heuristic plot uses a yellow color to indicate high reconstructability. Notably, the yellow region in our reconstructability heuristic plot shows a substantial overlap with the blue region in the reconstruction depth error plot. This visual

overlap serves to demonstrate that our reconstructability heuristic excels in estimating and predicting reconstruction depth errors.

Figure 8 illustrates that with an increment in the reconstructability heuristic, the reduction in depth error stemming from an equivalent increase in the additive heuristic [10] diminishes progressively. This trend aligns with the principle of diminishing returns inherent in image-based 3D reconstructions. Furthermore, the Pearson correlation coefficient between the mean reconstruction depth error and the additive reconstructability heuristic [10] is computed as -0.7866 , whereas ours achieves a correlation of -0.9362 . This result provides additional empirical evidence that, at the data level, our submodularized reconstructability heuristic exhibits a more robust predictive capability for reconstructability.

4.3.2. Task Allocation

During the task allocation phase, we encode task sequences for multiple UAVs and optimize the allocation of tasks by minimizing the fitness function. To assess the impact of individual terms in the fitness function, we systematically remove these terms from the function f . Subsequently, we devise flight paths for multiple UAVs within the synthetic School scenario. Here, we consider a configuration with three UAVs ($N_{\text{uav}} = 3$) and eight tasks ($N_t = 8$). Figure 9 displays the outcomes from the initial task allocation round, while Table 3 presents the ultimate quality assessment of the trajectories for multiple UAVs.

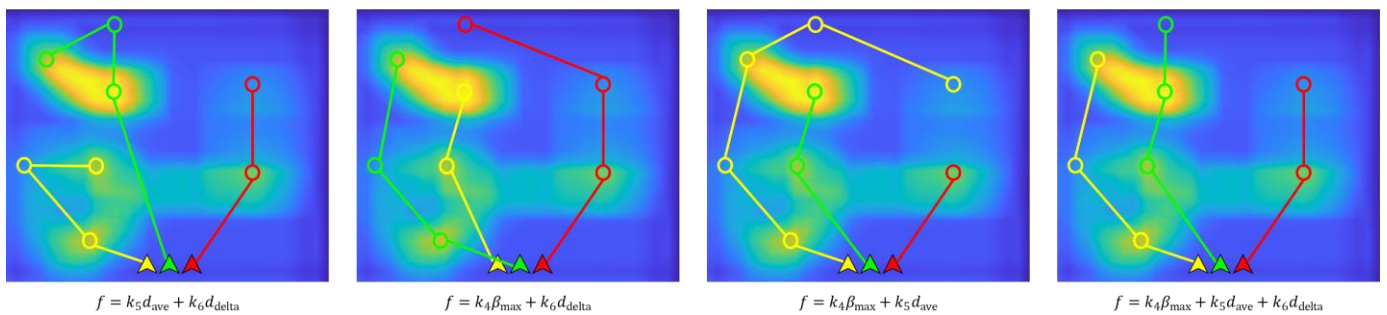


Figure 9. Task allocation results corresponding to different fitness functions (from left to right: without β_{max} , without d_{ave} , without d_{delta} , and the complete version). Circles denote task targets, while distinct colored lines depict task sequences associated with individual UAVs.

Table 3. Effect of different fitness functions on trajectory quality. Total Length represents the combined length of the continuous trajectories of multiple UAVs. Total Energy Cost denotes the total energy expenditure across all continuous trajectories of multi-UAVs. Max Time Cost indicates the longest time taken by multiple continuous trajectories, reflecting the overall mission completion time of the multi-UAV system. The data marked in bold represents the optimal values.

Fitness Function f	Total Length (m)	Total Energy Cost	Max Time Cost (min)
without β_{max}	8381.76	71.361	27.36
without d_{ave}	7412.43	50.496	17.94
without d_{delta}	6744.96	44.911	22.13
complete f	6949.42	46.449	15.81

Observing Figure 9 and Table 3 reveals three main findings. First, in the absence of β_{max} , the task allocation favors directing UAVs towards the nearest task, disregarding corners between adjacent tasks. Consequently, this inclination leads to significantly elevated trajectory lengths and energy costs. Second, excluding d_{ave} results in instances where neighboring tasks are distantly positioned, consequently impairing the trajectory quality. Third, the omission of d_{delta} overlooks length gaps between task sequences. Notably, although the collective trajectories attain their lowest overall length and energy costs at this

juncture, specific UAV paths sometimes become excessively prolonged, thereby extending the overall task completion time.

Ultimately, when the fitness function remains complete, the trajectory length and energy cost approach optimality, resulting in minimized overall task time consumption. This outcome suggests that the fitness function's terms effectively regulate task corners, distances between neighboring tasks, and length gaps within task sequences. By achieving equilibrium among these factors, the production of superior continuous trajectories is accomplished.

4.3.3. Path Searching

During the path searching phase, we formulate a novel A* algorithm to optimize multi-UAV trajectories by minimizing the objective function. The cumulative optimization impact of the search algorithm on trajectories will be presented in Section 4.3.4. Within this section, we will conduct targeted tests of the influence of each component in the objective function on the planning outcomes. These experiments were carried out in the synthetic School scenario, and the results are presented in Table 4.

Table 4. Impact of objective function compositions on reconstruction and trajectory quality.

Objective Function W	Num. Images	Error 95% (m) ↓	Comp. 0.075 m (%) ↑	Total Length (m)	Total Energy Cost
only W_1	700	0.4531	30.23	10,531.73	65.000
W_1 and W_3	413	0.0831	42.43	7169.86	46.371
W_1 and W_2	324	0.0894	44.83	7628.49	64.648
without coverage	345	0.0798	34.59	6511.37	50.158
full W	332	0.0824	46.76	6949.42	46.449

Firstly, we examined the impact of trajectory planning and 3D reconstructions when only W_1 is present in the objective function. The purpose of extending the path at this time is solely to reach the target area, which results in poor quality reconstructed models based on the captured images, even if high-energy cost, long-distance multi-UAV trajectories with numerous viewpoints are extended.

Secondly, we added W_3 to the objective function and showed that the multi-UAV trajectories were able to reconstruct a higher quality 3D model with lower energy costs than if only W_1 was present. Nevertheless, the excessively high number of viewpoints would result in an extended 3D reconstruction time. And the opposite outcome occurs when only W_1 and W_2 are incorporated.

Thirdly, we examined the impact of incorporating coverage on the trajectory and reconstruction quality. When coverage is excluded from the composition of the reconstruction contribution, the reconstruction results exhibit a marginal error improvement. Nonetheless, this change leads to a significant reduction in completeness.

Finally, by employing the complete objective function, our approach successfully devises multi-UAV trajectories characterized by a minimized energy cost, a reduced length, and an exceptional reconstruction quality. The incorporation of W_2 and W_3 effectively empowers our path planning approach to achieve a well-calibrated equilibrium among the reconstruction quality, viewpoint count, and trajectory excellence. Additionally, the introduction of coverage markedly enhances the comprehensive reconstruction quality of the 3D model.

4.3.4. Contribution of Each Module

Within the proposed path planning framework, which is rooted in task-oriented and search-based principles, the three distinct modules are allocated specific roles while being interdependent. To comprehensively compare the influence of each module on the overarching path planning framework, we conducted three ablation experiments in the synthetic School scenario: Ablation 1 entails the omission of the reconstructability estimation module's task location determination based on RLM, opting instead to uniformly distribute task objectives across the scene's horizontal plane. Ablation 2 involves generating

randomized task sequences for multiple UAVs within the task allocation module. Ablation 3 entails substituting the A*-based searching process within the path searching module with the NBV strategy to extend multi-UAV trajectories. The extension objective is restricted solely to proximity to the task target location. The outcomes of the experiments are depicted in Figure 10.

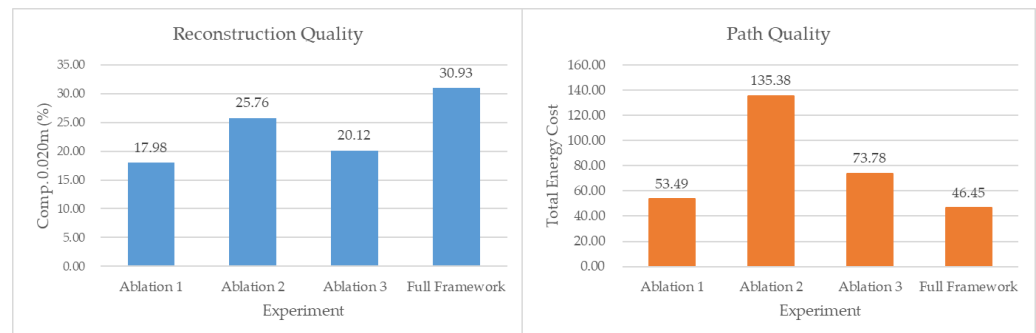


Figure 10. Comparison between various experiments regarding reconstruction quality (left) and path quality (right). We adopt Comp. 0.020 m as the metric for evaluating the reconstruction quality and Total Energy Cost for evaluating the path quality.

When compared to Full Framework, several conclusions can be drawn: Firstly, the reconstruction quality in Ablation 1 demonstrates that our approach, which determines task locations by generating RLM within the reconstructability estimation module, achieves superior global optimization in terms of reconstruction quality. Secondly, the total energy cost of the trajectories in Ablation 2 exceeds that of Full Framework, indicating that our task allocation module's multi-UAV task sequences effectively facilitate the generation of energy-efficient continuous trajectories. Thirdly, the unsatisfactory reconstruction and trajectory quality in Ablation 3 underscores the capability of our path searching module to jointly incorporate trajectory energy costs and scene reconstructability into optimization, thereby significantly enhancing reconstruction and trajectory quality.

4.3.5. Overall Performance Evaluation

In this section, we conduct a comprehensive assessment of our path planning algorithm, considering both its collaborative performance and path planning time.

Collaborative capability. Our method involves planning safe continuous trajectories for multiple UAVs to capture images of a scene. To demonstrate the collaborative capability of our method, we plan trajectories for clusters with different numbers of UAVs in the synthetic scene School, showcasing that multiple UAVs can complete tasks in a coordinated manner instead of capturing data relatively independently. The impact of the UAV number on the reconstruction results and trajectory quality is presented in Table 5. The results indicate that the total length and the total energy cost of multiple trajectories decrease as the number of UAVs increases, while maintaining a similar reconstruction quality. This demonstrates that our method effectively enables the collaborative capture of scene images by multiple UAVs. However, when the number of UAVs reaches nine, both the total length and the total energy cost of the multi-trajectories tend to increase. This indicates that an excessive number of UAVs in the same scene can lead to redundant trajectories, thereby diminishing the trajectory quality. To simplify the experiment, we set the number of UAVs to three in all subsequent synthetic and real scenes, which serves the experimental purpose and closely aligns with real-world application scenarios.

Table 5. Impact of UAV number on reconstruction results and trajectory quality.

Num. UAVs	Num. Images	Error 95% (m) ↓	Comp. 0.075 m (%) ↑	Total Length (m)	Total Energy Cost	Max Time Cost (min)
1	369	0.0835	44.15	7835.64	50.157	39.90
3	332	0.0804	46.76	6949.42	44.449	15.81
5	328	0.0812	45.56	6589.38	38.183	10.11
7	340	0.0791	47.01	6377.49	37.251	7.42
9	351	0.0779	46.13	6531.17	40.067	5.44

Path Planning Time. As part of the self-evaluation, we assessed the overall runtime of our algorithm. Our algorithm required 31.3, 18.1, and 16.5 min to plan the trajectories of three UAVs in the School, Town, and Castle environments, respectively. Despite its longer runtime compared to other state-of-the-art methods [10,12,21], our method facilitates collaborative image captures by multiple UAVs, resulting in a significant reduction of tens of minutes in the image capture session. Additionally, since the 3D reconstruction session often lasts several hours, the algorithm’s runtime is generally considered acceptable in comparison.

4.4. Comparisons in Synthetic Scenes

We compare our method with various state-of-the-art methods [10,12,21], as well as the oblique photography (OP) method, in the synthetic scenes of School, Castle, and Town. We obtained the trajectories of these methods used for comparison from the UrbanCity dataset [14]. Specifically, we selected trajectories with *low_overlap* generated by each method in different scenes. It is important to note that since all other methods can only generate a single trajectory, we divided these trajectories into three equal segments to ensure fairness and assess our multi-UAV collaboration performance. Furthermore, since the oblique photography trajectories in the UrbanCity dataset are generated assuming the UAV has five lenses, we compare the length, energy cost, and time cost values of the oblique photography trajectories with our method after applying a four-fold improvement, considering that all our experiments are based on the assumption of a UAV with a monocular camera.

Quantitative analysis. Table 6 presents a comparison of the *Error* and *Completeness* of the reconstruction results achieved by various methods across different scenes. Compared to other methods, our method demonstrates the capability to reconstruct a 3D model of similar or higher quality using fewer images, except in the Castle scene. In the Castle scene, although our method captures slightly more images, a substantial improvement in reconstruction quality is observed. The above observations suggest that our method exhibits greater robustness in capturing images for 3D reconstructions.

Table 6. Comparison of the reconstruction quality achieved by various methods across different scenes.

Scene	Methods	Num. Images	Error 90% (m) ↓	Error 95% (m) ↓	Comp. 0.020 m (%) ↑	Comp. 0.050 m (%) ↑	Comp. 0.075 m (%) ↑
School	OP	1079	0.1511	0.2671	16.25	25.36	34.79
	Smith et al. [10]	559	0.0418	0.0937	30.39	40.92	46.18
	Zhou et al. [12]	323	0.0456	0.0950	29.82	43.13	46.02
	Zhang et al. [21]	342	0.0805	0.1623	20.95	40.06	46.8
	Ours	321	0.0474	0.0824	30.93	42.20	46.76
Castle	OP	800	0.1271	0.2354	37.21	41.21	53.97
	Smith et al. [10]	251	0.0670	0.1230	50.96	61.87	66.20
	Zhou et al. [12]	213	0.0414	0.1295	54.71	59.46	62.76
	Zhang et al. [21]	213	0.0583	0.1729	54.83	59.74	63.12
	Ours	228	0.0205	0.0260	55.92	62.71	66.03
Town	OP	720	0.0652	0.1337	37.61	47.07	64.65
	Smith et al. [10]	155	0.0571	0.1208	26.51	42.51	55.63
	Zhou et al. [12]	259	0.0142	0.0207	41.54	51.69	64.94
	Zhang et al. [21]	258	0.0391	0.0819	35.03	45.72	59.68
	Ours	248	0.0126	0.0213	43.25	50.16	67.53

Figure 11 presents a comparison of different methods in terms of the trajectory quality. This figure reveals that our method is capable of planning trajectories with the lowest or nearly the lowest total length across different scenes. Moreover, leveraging the advantage of collaborative image captures by multiple UAVs, our method achieves full scene image

captures in the shortest time. Furthermore, by quantifying energy costs and incorporating them into optimization, our multi-trajectory exhibits a significantly lower total energy cost compared to other methods. This advantage enables us to effectively apply our method in real scenes, facilitating the completion of the entire image capture task without the need for battery replacement midway.

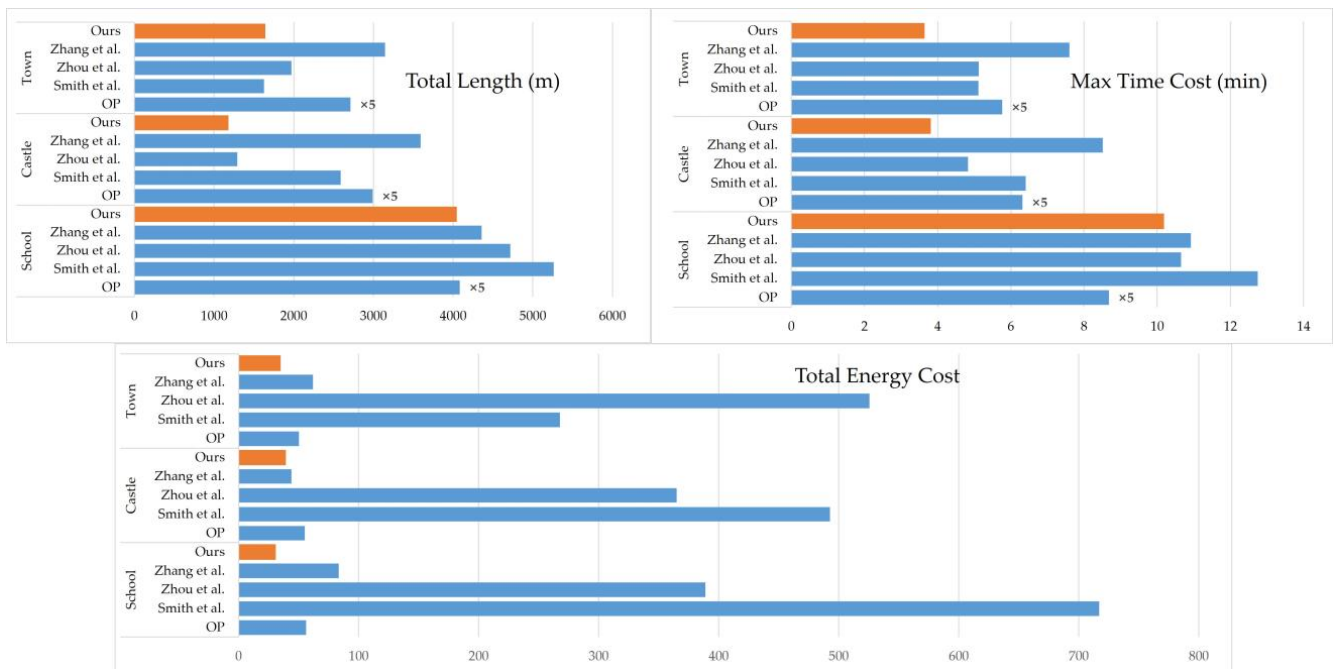


Figure 11. Comparison of trajectory quality among different methods in terms of Total Length, Max Time Cost, and Total Energy Cost. Due to the excessive total length and maximum time cost of oblique photography, only 1/5 of their values are presented in the figure to allow for clear comparison with other methods [10,12,21].

Qualitative analysis. Figure 12 depicts the ground truth surfaces densely sampled in each synthetic scene. We compute the closest distance between each surface point and the reconstructed surface of each method, representing it with a gradient color ranging from blue (low) to red (high). Figure 12 reveals that our method exhibits a predominant blue color with minimal occurrences of green and red areas. This observation signifies the outstanding reconstruction quality achieved by our method.

Figure 13 presents a visual comparison of trajectory and surface details across different scenes for various methods. In comparison to other state-of-the-art methods [10,12,21], our approach excels at capturing images that make a significant contribution to reconstructions by employing continuous trajectories in close proximity to the building surface. This capability enables the reconstruction of finer surface details.

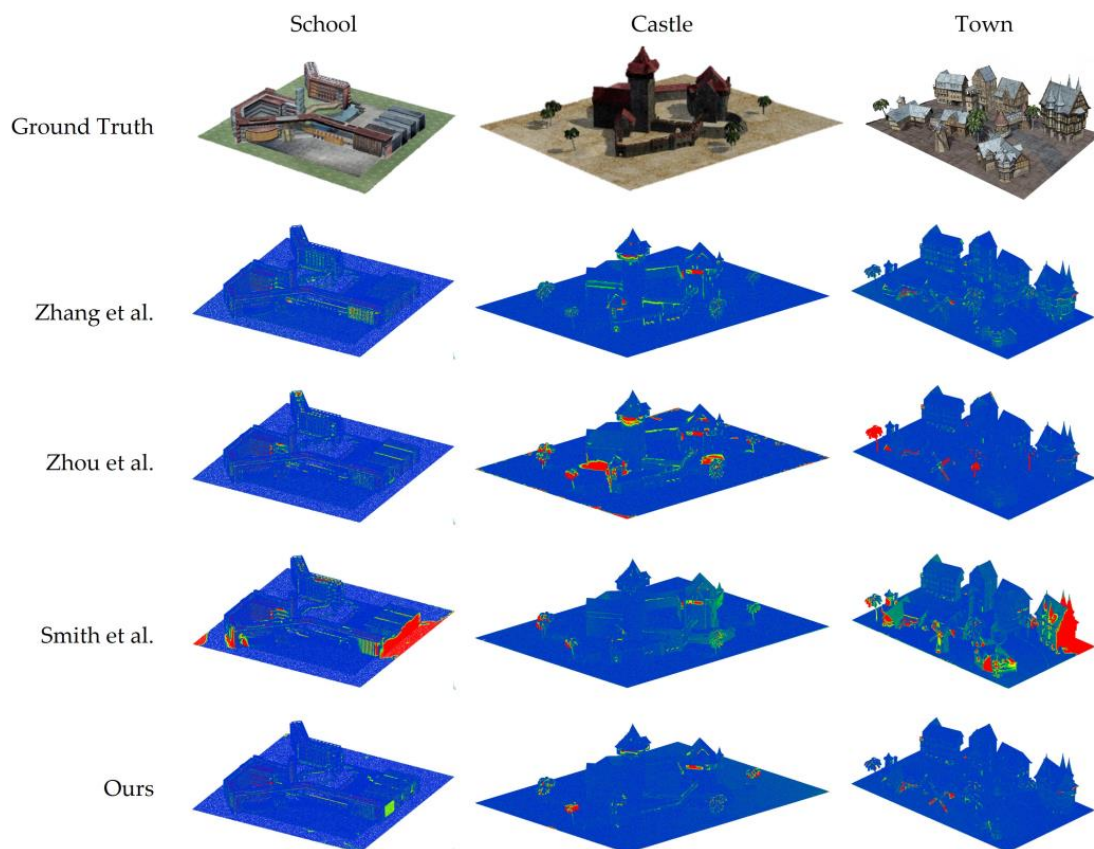


Figure 12. A comprehensive visual comparison of the reconstruction quality achieved by various methods. The reconstruction depth error is visualized using a gradient ranging from blue (low) to red (high) [10,12,21].

4.5. Comparisons in Real Scenes

In real scenes, we compare our method solely with the open-source approach [10] and the oblique photography method available in the commercial software DJI-Pilot [5]. As the DJI Phantom 4 RTK used in our experiment is equipped with a monocular camera, DJI-Pilot [5] generated five trajectories to simulate the use of five lenses, as shown in Figure 14. To ensure fairness, similar to the experiments conducted in the synthetic scenes, the trajectories generated by both the oblique photography method and [10] were divided into three equal segments. Subsequently, three UAVs were assigned to capture the images simultaneously.

Figure 14 presents the visual effects of various methods in the GradSch, LiterColl, and Gym scenes, focusing on trajectories and the reconstruction of surface details. Due to space limitations, the performance of the oblique photography method is only displayed in the Gym scene at the bottom of Figure 14. In the Gym scene, the oblique photography method struggles to accurately recover the façade details of the building, resulting in stretched textures. Furthermore, the presence of duplicated textures on the roof of Gym presents a reconstruction challenge, resulting in surface holes in the reconstructed output of the method proposed by Smith et al. [10]. Our method effectively resolves this issue. Additionally, in the other two scenes, our method produces 3D models with enhanced surface details and a closer resemblance to the original geometry. This demonstrates that our method maintains an excellent 3D reconstruction performance in real scenes.

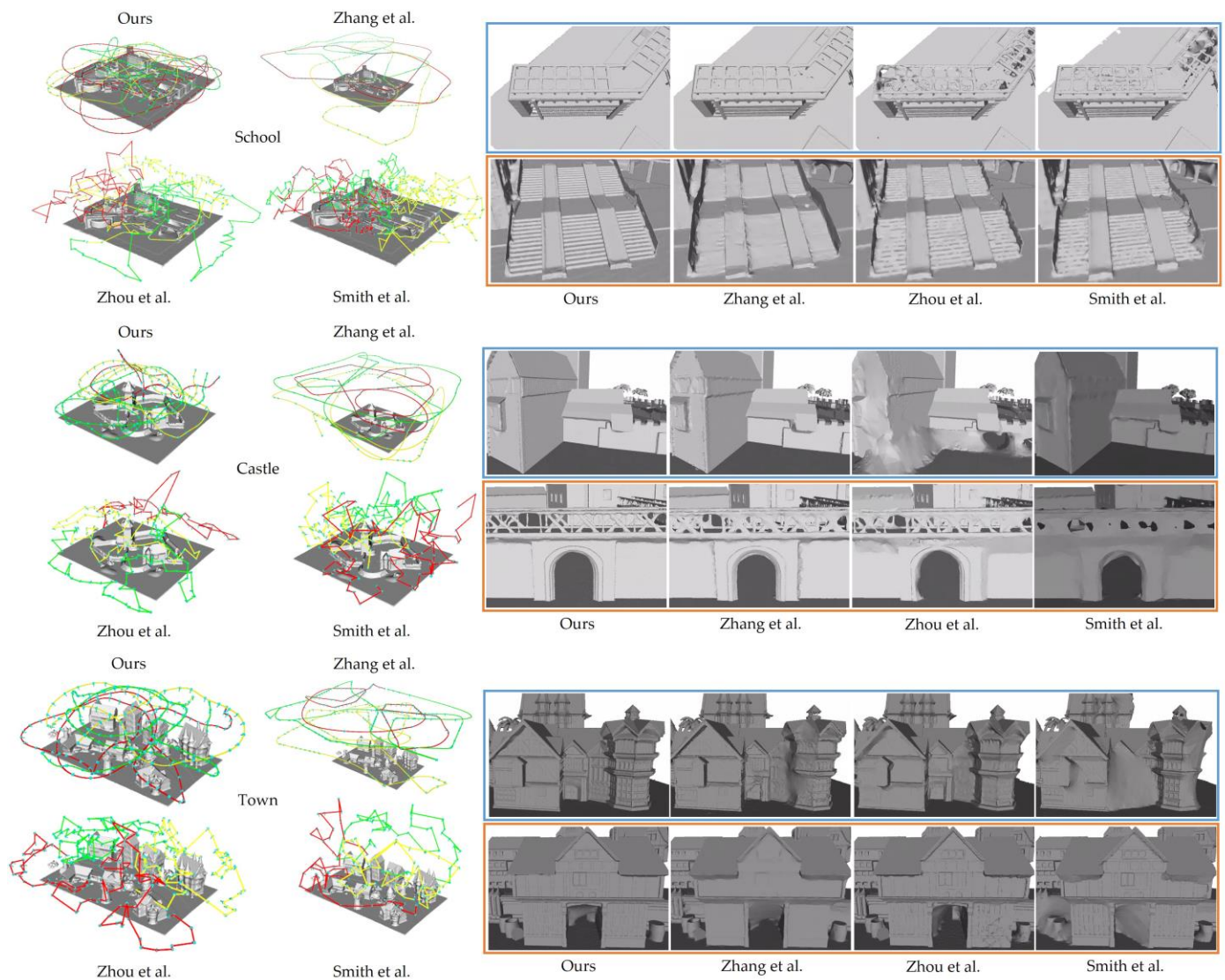


Figure 13. Visual comparison of trajectory and surface details across different scenes (School, Castle, and Town from top to bottom) for various methods [10,12,21].

Table 7 provides quantitative data regarding the size of each scene and the quality of trajectories generated by various methods. Due to the automated generation of oblique photography trajectories in each scene using commercial flight software, it was not possible to calculate the precise energy cost of the trajectories as described in Section 3.3.1. As a result, the corresponding positions in Table 7 are left blank. However, during the experiments in real scenes, we recorded the battery consumption of all UAVs and determined the average battery consumption of the three UAVs. This average was expressed as a percentage of the total battery power. Based on the results, we can draw the following two conclusions:

1. Merely having longer flight paths and longer flight times does not necessarily imply increased battery power loss. Instead, the energy cost metric we introduced exhibits a significant positive trend with battery consumption, demonstrating a superior ability to predict battery consumption;
2. In comparison to alternative methods, our approach empowers multiple UAVs to capture image data, making a greater contribution to scene reconstructions while consuming less battery and shorter times.

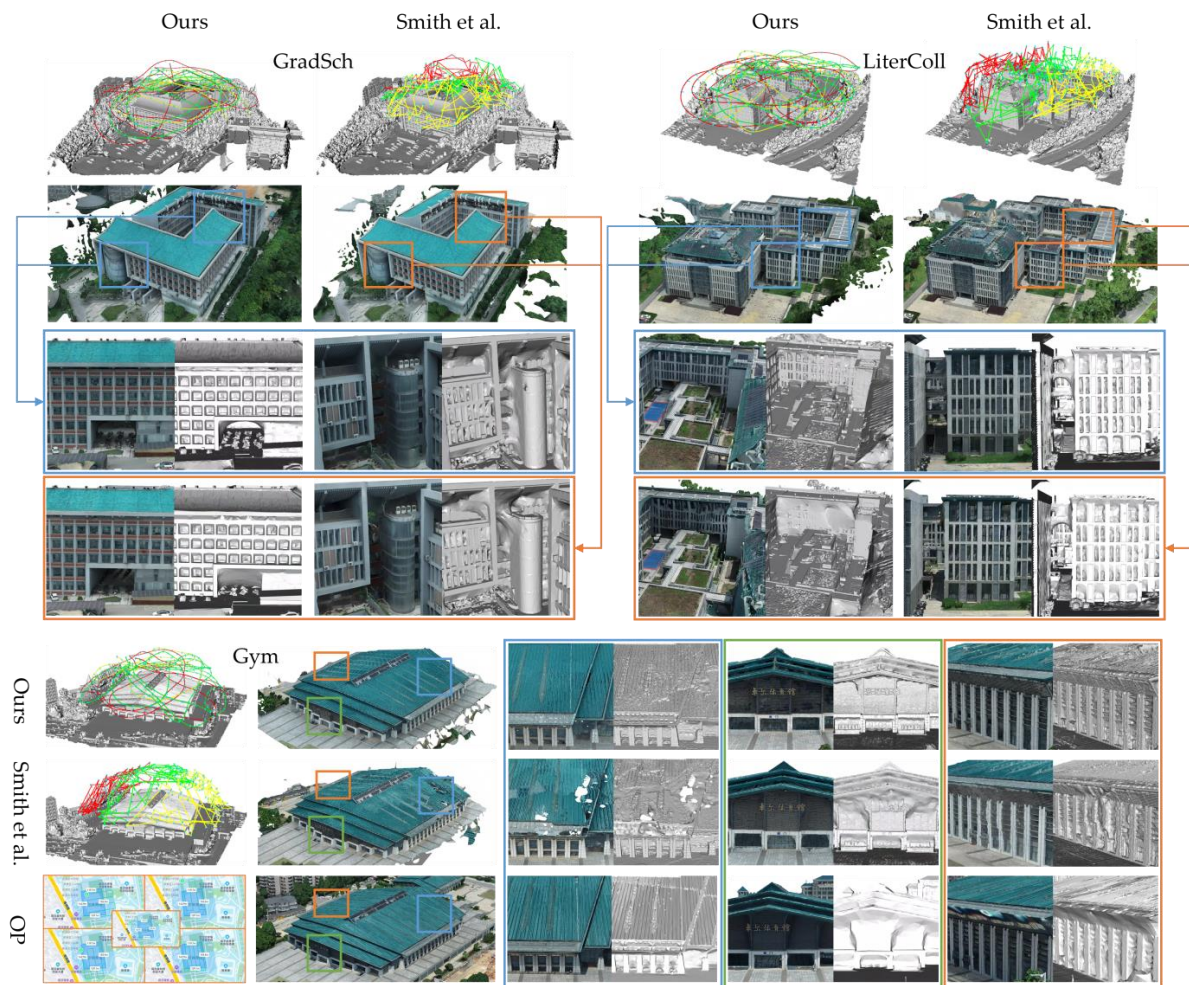


Figure 14. Visual effects of trajectories and reconstructed surface details of our method compared to Smith et al. [10] in real scenes: GradSch (top left), LiterColl (top right), and Gym (bottom). The paths (green dashes) and reconstruction effects of oblique photography are included only for the scene Gym due to space constraints.

Table 7. Comparison of the trajectory quality between our method, the method proposed by Smith et al. [10], and the oblique photography method in real scenes (GradSch, LiterColl, and Gym). The trajectories of the oblique photography method are automatically generated using commercial flight software, thus making the theoretical trajectory energy cost unavailable.

Scene	Area (m ²)	Hight (m)	Methods	Num. Images	Total Length (m)	Total Energy Cost	Max Time Cost (min)	Average Battery Consump. (%)
LiterColl	28,490	28.87	OP	618	9911.86	-	11.44	24.78
			Smith et al. [10]	388	6606.84	460.92	19.14	67.67
			Ours	379	4495.14	42.44	12.87	26.15
GradSch	11,772	31.00	OP	401	5901.14	-	7.83	17.31
			Smith et al. [10]	424	6532.91	435.64	15.51	74.67
			Ours	401	4495.58	38.07	10.89	21.56
Gym	18,848	28.24	OP	442	6457.25	-	8.67	20.12
			Smith et al. [10]	493	6937.18	844.85	20.17	92.62
			Ours	419	4702.58	42.11	11.61	22.01

Additionally, it is noteworthy that during our experiments in the GradSch scenario using the method proposed by Smith et al. [10], an incident arose where the mission was automatically halted due to a couple of UAVs being in close proximity. Only through our manual intervention was the mission able to proceed safely. In contrast, our method completely circumvents this situation.

5. Discussion

Although our method effectively devises energy-efficient paths for multiple UAVs and demonstrates an exceptional reconstruction quality, it still has some limitations. Firstly, our approach necessitates a coarse proxy input. This implies that, for a completely novel scene, the vertical image must be pre-captured and the coarse proxy reconstructed, thereby leading to a further increase in overall flight elapsed time. Furthermore, our method prioritizes the global optimality of trajectories at the expense of algorithmic complexity. Consequently, the runtime is prolonged compared to alternative methods, presenting a challenge in the context of outdoor scene planning. Finally, despite real scenario experiments indicating a distinct positive trend in quantifying energy costs and real power losses, their linear correlation remains absent. Consequently, constraining trajectories based on UAV battery capacity continues to pose challenges. This results in planned trajectories that may not be executable by a UAV in a single pass. We believe that these limitations merit further exploration in future research endeavors.

6. Conclusions

This work introduces a task-oriented and search-based path planning framework that achieves the creation of energy-efficient continuous trajectories for multiple UAVs, enabling collaborative image captures for the reconstruction of high-quality 3D models. Our framework comprises three key modules. Firstly, the reconstructability estimation module enhances reconstructions' global optimality by constructing a submodular reconstructability heuristic and reconstructability loss map. These tools aid in pinpointing essential task locations within the scene. Secondly, the task allocation module involves the design of a task sequence-solution space codec, facilitating the allocation of optimal task sequences to multiple UAVs. This is achieved through the minimization of the fitness function within the real number space, promoting collaboration among UAVs and enhancing trajectory smoothness. Thirdly, the path searching module involves the quantification of safety and energy costs across multiple trajectories, guided by the dynamics of continuous trajectories. These quantifications are subsequently integrated into the path optimization process, together with scene reconstructability considerations. This integration significantly enhances both the path and reconstruction quality. Subsequent research could tackle the constraints of our approach, including the mitigation of time consumption by fully utilizing vertical photographic images. Furthermore, there is the potential to explore learning-based strategies to advance path planning performances.

Author Contributions: Conceptualization, H.Z.; methodology, H.Z.; software, H.Z. and X.W.; validation, H.Z., G.G. and F.L.; formal analysis, H.Z. and S.W.; investigation, G.G., F.L. and J.L.; resources, X.W.; data curation, H.Z. and X.W.; writing—original draft preparation, H.Z. and G.G.; writing—review and editing, H.Z., G.G. and J.L.; visualization, H.Z. and G.G.; supervision, H.S.; project administration, H.S. and H.Z.; funding acquisition, H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Hong Kong Research Grant Council (RGC) General Research Fund (HKBU 12301820) and the Guangxi Science and Technology Major Project (No. AA22068072).

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of the data; and in the writing of the manuscript, or in the decision to publish the results.

References

1. Fuhrmann, S.; Langguth, F.; Goesele, M. MVE—A Multi-View Reconstruction Environment. In Proceedings of the Eurographics Workshop on Graphics and Cultural Heritage, Darmstadt, Germany, 6–8 October 2014.

2. Schonberger, J.L.; Frahm, J.-M. Structure-from-Motion Revisited. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016.
3. Barron, J.T.; Mildenhall, B.; Verbin, D.; Srinivasan, P.P.; Hedman, P. Zip-NeRF: Anti-Aliased Grid-Based Neural Radiance Fields. *arXiv* **2023**, arXiv:2304.06706.
4. Jiang, C.; Zhang, H.; Liu, P.; Yu, Z.; Cheng, H.; Zhou, B.; Shen, S. H2-Mapping: Real-Time Dense Mapping Using Hierarchical Hybrid Representation. *arXiv* **2023**, arXiv:2306.03207.
5. DJI-Pilot. Available online: <https://www.dji.com/downloads/djiapp/dji-pilot> (accessed on 3 July 2023).
6. Pix4Dcapture Pro. Available online: <https://www.pix4d.com/product/pix4dcapture> (accessed on 3 July 2023).
7. Sharma, O.; Arora, N.; Sagar, H. Image Acquisition for High Quality Architectural Reconstruction. In Proceedings of the 45th Graphics Interface Conference on Proceedings of Graphics Interface 2019 (GI'19), Kingston, ON, Canada, 28–31 June 2019.
8. Hoppe, C.; Wendel, A.; Zollmann, S.; Pirker, K.; Irschara, A.; Bischof, H.; Kluckner, S. Photogrammetric Camera Network Design for Micro Aerial Vehicles. In Proceedings of the 17th Computer Vision Winter Workshop, Mala Nedelja, Slovenia, 1–3 February 2012.
9. Roberts, M.; Shah, S.; Dey, D.; Truong, A.; Sinha, S.; Kapoor, A.; Hanrahan, P.; Joshi, N. Submodular Trajectory Optimization for Aerial 3D Scanning. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
10. Smith, N.; Moehrl, N.; Goesele, M.; Heidrich, W. Aerial Path Planning for Urban Scene Reconstruction: A Continuous Optimization Method and Benchmark. *ACM Trans. Graph.* **2018**, *37*, 1–15. [[CrossRef](#)]
11. Hepp, B.; Nießner, M.; Hilliges, O. Plan3D: Viewpoint and Trajectory Optimization for Aerial Multi-View Stereo Reconstruction. *ACM Trans. Graph.* **2019**, *38*, 1–17. [[CrossRef](#)]
12. Zhou, X.; Xie, K.; Huang, K.; Liu, Y.; Zhou, Y.; Gong, M.; Huang, H. Offsite Aerial Path Planning for Efficient Urban Scene Reconstruction. *ACM Trans. Graph.* **2020**, *39*, 1–16. [[CrossRef](#)]
13. Liu, X.; Ji, Z.; Zhou, H.; Zhang, Z.; Tao, P.; Xi, K.; Chen, L.; Marcato Junior, J. An Object-Oriented UAV 3D Path Planning Method Applied in Cultural Heritage Documentation. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.* **2022**, *V-1-2022*, 33–40. [[CrossRef](#)]
14. Lin, L.; Liu, Y.; Hu, Y.; Yan, X.; Xie, K.; Huang, H. Capturing, Reconstructing, and Simulating: The UrbanScene3D Dataset. In Proceedings of the 17th European Conference on Computer Vision (ECCV), Tel Aviv, Israel, 23–27 October 2022.
15. Batinovic, A.; Ivanovic, A.; Petrovic, T.; Bogdan, S. A Shadowcasting-Based Next-Best-View Planner for Autonomous 3D Exploration. *IEEE Robot. Autom. Lett.* **2022**, *7*, 2969–2976. [[CrossRef](#)]
16. Bircher, A.; Kamel, M.; Alexis, K.; Oleynikova, H.; Siegwart, R. Receding Horizon “Next-Best-View” Planner for 3D Exploration. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016.
17. Dai, A.; Papatheodorou, S.; Funk, N.; Tzoumanikas, D.; Leutenegger, S. Fast Frontier-Based Information-Driven Autonomous Exploration with an MAV. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Virtual, 31 May–1 August 2020.
18. Yamauchi, B. A Frontier-Based Approach for Autonomous Exploration. In Proceedings of the 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. “Towards New Computational Principles for Robotics and Automation”, Monterey, CA, USA, 10–11 July 1997.
19. Zhou, G.; Yuan, J.; Yen, I.-L.; Bastani, F. Robust Real-Time UAV Based Power Line Detection and Tracking. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016.
20. Feng, C.; Li, H.; Gao, F.; Zhou, B.; Shen, S. PredRecon: A Prediction-Boosted Planning Framework for Fast and High-Quality Autonomous Aerial Reconstruction. *arXiv* **2023**, arXiv:2302.04488.
21. Zhang, H.; Yao, Y.; Xie, K.; Fu, C.-W.; Zhang, H.; Huang, H. Continuous Aerial Path Planning for 3D Urban Scene Reconstruction. *ACM Trans. Graph.* **2021**, *40*, 1–15. [[CrossRef](#)]
22. Zheng, X.; Wang, F.; Li, Z. A Multi-UAV Cooperative Route Planning Methodology for 3D Fine-Resolution Building Model Reconstruction. *ISPRS J. Photogramm. Remote Sens.* **2018**, *146*, 483–494. [[CrossRef](#)]
23. Dong, H.; Wu, N.; Feng, G.; Gao, X. Research on Computing Task Allocation Method Based on Multi-UAVs Collaboration. In Proceedings of the 2020 IEEE International Conference on Smart Internet of Things (SmartIoT), Xining, China, 9–11 August 2020.
24. Zhang, R.; Feng, Y.; Yang, Y.; Li, X. A Deadlock-Free Hybrid Estimation of Distribution Algorithm for Cooperative Multi-UAV Task Assignment With Temporally Coupled Constraints. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *59*, 3329–3344. [[CrossRef](#)]
25. Hong, Y.; Jung, S.; Kim, S.; Cha, J. Multi-UAV Routing with Priority Using Mixed Integer Linear Programming. In Proceedings of the 2020 20th International Conference on Control, Automation and Systems (ICCAS), Busan, Republic of Korea, 13–16 December 2020.
26. Wang, S.; He, J.; Huang, Y.; Wang, A. Cooperative Task Assignment of Uninhabited Combat Air Vehicles Based on Improved MOSFLA Algorithm. In Proceedings of the 2016 3rd International Conference on Systems and Informatics (ICSAI), Shanghai, China, 19–21 November 2016.
27. Wu, X.; Yin, Y.; Xu, L.; Wu, X.; Meng, F.; Zhen, R. MULTI-UAV Task Allocation Based on Improved Genetic Algorithm. *IEEE Access* **2021**, *9*, 100369–100379. [[CrossRef](#)]
28. Chen, L.; Liu, W.-L.; Zhong, J. An Efficient Multi-Objective Ant Colony Optimization for Task Allocation of Heterogeneous Unmanned Aerial Vehicles. *J. Comput. Sci.* **2022**, *58*, 101545. [[CrossRef](#)]

29. Kumar, G.; Anwar, A.; Dikshit, A.; Poddar, A.; Soni, U.; Song, W.K. Obstacle Avoidance for a Swarm of Unmanned Aerial Vehicles Operating on Particle Swarm Optimization: A Swarm Intelligence Approach for Search and Rescue Missions. *J. Braz. Soc. Mech. Sci. Eng.* **2022**, *44*, 56. [[CrossRef](#)]
30. LaValle, S.M. *Rapidly-Exploring Random Trees: A New Tool for Path Planning*; TR 98-11; Department of Computer Science, Iowa State University: Ames, IA, USA, 1998.
31. Kuffner, J.J.; LaValle, S.M. RRT-Connect: An Efficient Approach to Single-Query Path Planning. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), San Francisco, CA, USA, 24–28 April 2000.
32. Kala, R. Rapidly Exploring Random Graphs: Motion Planning of Multiple Mobile Robots. *Adv. Robot.* **2013**, *27*, 1113–1122. [[CrossRef](#)]
33. Harabor, D.; Grastien, A. Online Graph Pruning for Pathfinding On Grid Maps. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 7–11 August 2011.
34. Likhachev, M.; Gordon, G.J.; Thrun, S. ARA*: Anytime A* with Provable Bounds on Sub-Optimality. In Proceedings of the 16th International Conference on Neural Information Processing Systems, Whistler, BC, Canada, 9–11 December 2003.
35. Kurzer, K. Path Planning in Unstructured Environments: A Real-Time Hybrid A* Implementation for Fast and Deterministic Path Generation for the KTH Research Concept Vehicle. Master's Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2016.
36. Furukawa, Y.; Hernández, C. Multi-View Stereo: A Tutorial. *Found. Trends®Comput. Graph. Vis.* **2015**, *9*, 1–148. [[CrossRef](#)]
37. Krause, A.; Golovin, D. *Submodular Function Maximization*; Cambridge University Press: Cambridge, UK, 2014; pp. 71–104.
38. Yang, X.; Deb, S. Cuckoo Search via Lévy Flights. In Proceedings of the 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), Coimbatore, India, 9–11 December 2009.
39. Mellinger, D.; Kumar, V. Minimum Snap Trajectory Generation and Control for Quadrotors. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011.
40. Gao, F.; Wang, L.; Zhou, B.; Zhou, X.; Pan, J.; Shen, S. Teach-Repeat-Replan: A Complete and Robust System for Aggressive Flight in Complex Environments. *IEEE Trans. Robot.* **2020**, *36*, 1526–1545. [[CrossRef](#)]
41. Wang, Z.; Ye, H.; Xu, C.; Gao, F. Generating Large-Scale Trajectories Efficiently Using Double Descriptions of Polynomials. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021.
42. DasEarth. Available online: <https://earth.daspatial.com> (accessed on 3 July 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.