RESEARCH ARTICLE



A hierarchical multi-UAV cooperative framework for infrastructure inspection and reconstruction

Chuanxiang Gao¹ · Xinyi Wang¹ · Xi Chen¹ · Ben M. Chen¹

Received: 12 August 2023 / Revised: 22 September 2023 / Accepted: 5 October 2023 / Published online: 7 March 2024 © The Author(s) 2024

Abstract

Unmanned aerial vehicles (UAVs) are emerging as a powerful tool for inspections and repair works in large-scale and unstructured 3D infrastructures, but current approaches take a long time to cover the entire area. Planning using UAVs for inspections and repair works puts forward a requirement of improving time efficiency in large-scale and cluster environments. This paper presents a hierarchical multi-UAV cooperative framework for infrastructure inspection and reconstruction to balance the workload and reduce the overall task completion time. The proposed framework consists of two stages, the exploration stage and the exploitation stage, resolving the task in a sequential manner. At the exploration stage, the density map is developed to update global and local information for dynamic load-balanced area partition based on reconstructability and relative positions of UAVs, and the Voronoi-based planner is used to enable the UAVs to reach their best region. After obtaining the global map, viewpoints are generated and divided while taking into account the battery capacity of each UAV. Finally, a shortest path planning method is used to minimize the total traveling cost of these viewpoints for obtaining a high-quality reconstruction. Several experiments are conducted in both a simulated and real environment to show the time efficiency, robustness, and effectiveness of the proposed method. Furthermore, the whole system is implemented in real applications.

Keywords Multi-UAV · Coverage path planning · Infrastructure inspection and reconstruction

1 Introduction

As the city grows, there are an increasing number of outdated buildings, making incidents like building collapses more common. Regular inspections and repair works are conducive and indispensable for building health and urban

This work was supported in part by the Research Grants Council of Hong Kong SAR (Nos. 14209020, 14206821) and in part by the Hong Kong Centre for Logistics Robotics (HKCLR).

 Xinyi Wang xywangmae@link.cuhk.edu.hk
 Chuanxiang Gao cxgao@mae.cuhk.edu.hk
 Xi Chen xichen002@cuhk.edu.hk

> Ben M. Chen bmchen@cuhk.edu.hk

¹ Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong, Shatin, NT, China safety. These inspections typically require workers to enter dangerous areas that are risky and difficult to access. With the maturity of unmanned aerial technologies, in many emergent scenarios, unmanned aerial vehicles (UAVs) play an essential role in collecting data efficiently and safely throughout the inspection process [1–6] (such as infrared images, optical images, etc.).

For inspection with a single UAV, Phung et al. [7] formulated the building inspection path planning problem into a traveling salesman problem (TSP) and verified their method on open datasets. Tan et al. [8] proposed a building information model (BIM) and UAV-integrated method to realize automatic inspection data collection. The inspection path planning problem is solved by a genetic algorithm based on the BIM of the target building. For real-time path planning, Kuang et al. [9] proposed a real-time flight path planning method based on the reconstructed point clouds and potential coverage areas with the top view of the building as the initial model. The approach can achieve coverage of the target building.



Fig. 1 CUHK Cheng Ming Building. A 52 m long, 14 m wide, and 16 m high building. **a** Physical building. **b** Reconstruction result

However, they concentrate on using a single UAV to complete the inspection task, which takes a long time to cover the entire building. Take the building shown in Fig. 1 as an example. It is 52 m long, 14 m wide, and 16 m high building. To achieve full coverage of this building, 870 viewpoints need to be reached, and it takes about 3 h for a single UAV to complete the whole task.

To provide fast and high-quality data for 3D building models, multiple UAVs are used for inspection and reconstruction problems to resolve task complexity and improve time efficiency. Jing et al. [10] proposed an offline approach by converting a multi-UAV inspection problem into a setcovering vehicle routing problem and used a modified genetic algorithm to solve it. In [11], Zheng et al. presented a UAV path planning methodology based on the pre-obtained rough 3D model of the building to cover the target building. The above works [12-14], either generated the viewpoints and path offline, or needed an initial model from pre-flight, Google earth, or open datasets for planning. However, in many real applications, the 3D model of the building and surrounding environment cannot be acquired in advance. Therefore, it becomes especially important to design an exploration method that can quickly acquire information about the target building and environment.

Recently, more and more researchers focus on using multiple UAVs to autonomously explore the unknown environment and reach their best regions. More specifically, numerous solutions tend to use cellular decomposition methods to reduce the workload for each UAV and perform a complete coverage task in real time. The most popular method is to utilize the Voronoi partition. It is tractable for a large number of robots or in large-scale environments, and beneficial to obtaining a quick and feasible solution only based on the position of robots [15, 16]. Collins et al. [17] proposed a Voronoi-based scalable approach by splitting the non-convex area for each UAV to minimize the task completion time. However, this work mainly focused on 2D environments without considering the 3D geometry of the target. The researchers further expand upon the convergence findings of the aforementioned work, transitioning from planar environments to encompass 3D scenarios [18]. However, it is important to note that these studies primarily focus on addressing the challenges posed by a static Voronoi partition, assuming prior knowledge of the environment. Regrettably, the consideration of unforeseen out-of-range obstacles occurring abruptly is seldom taken into account within these works.

To tackle these challenges, this paper presents an explorethen-exploit framework for multi-UAV cooperative building inspection and reconstruction. Building upon our previous work in [16], this framework guarantees the coverage of the target buildings, balanced workload assignment for each UAV, and realizes online data collection without a pre-obtained 3D model of the target building. The main contributions of this paper are as follows:

- We propose an explore-then-exploit framework for multi-UAV cooperative building inspection and reconstruction, which can significantly enhance data collection quality.
- We design a density map construction method that supplies both global and local information for realtime planning and execution. Concurrently, a dynamic Voronoi-based safe region is established to decrease the total inspection time by evenly distributing the workload among multi-UAV.
- Simulations are conducted in various environments to show the time efficiency, robustness, and effectiveness of the proposed framework. Furthermore, our developed system, which includes a team of three UAVs, is used to physically inspect different large-scale industrial buildings. This approach considerably reduces the overall operation time, effectively minimizing potential hazards.

The rest of the paper is organized as follows. In Sect. 2, the overall framework is introduced. The design of the density map, spatial deployment, exploration, and exploitation method has been covered in Sect. 3. The real applications are described in Sect. 4. Experimental results and evaluations are provided in Sect. 5. Conclusion is given in Sect. 6.

2 Explore-then-exploit framework

Existing methods for multi-UAV inspection problems can be divided into two steps. In the first step, a set of viewpoints are generated around the surface of the target facility. Then, in the second step, the shortest paths that travel through these viewpoints are founded by solving a multiple traveling salesman problem (mTSP). However, mTSP is an NP-hard (Non-deterministic polynomial hard) problem when the environment is large and complex. It takes a long time to solve mTSP with plenty of viewpoints to be optimized. Moreover, for large-scale infrastructures, even though multiple UAVs are used, each UAV needs to travel back and forth frequently, which is very time-consuming. As a result, we



Fig. 2 Overview of the proposed framework

propose an explore-then-exploit framework to decouple this complex and time-consuming problem and solve it in two stages. This framework reduces the requirement of knowing the 3D model of the target building and the whole process can be processed online. The overall framework is shown in Fig. 2.

During the online exploration of unknown environments, a density map is updated with consideration of the building's reconstructability. According to the information obtained from the density map and relative state information, a Voronoi-based planner is implemented to construct the safe corridor of the flight region, and UAVs cooperatively converge their corresponding working domains in a distributed way to cover the building as best as they can. This balanced workload deployment can reduce the total inspection time and improve flight efficiency. The Voronoi corridor is constructed to avoid obstacles in the movement. The advantages of performing multi-agent coverage exploration in a distributed way include robustness to agents being added to or leaving the system, less information must be shared with each other agent, and there is no central point of failure.

After completion of exploration, the UAVs have access to environmental and building information. The viewpoints for each UAV are generated within its working region. With regard to the number of viewpoints and the endurance of the batteries of each UAV, a capacity-constrained Voronoi is used to distribute the viewpoints for each flight. Therefore, a complex building inspection task is simplified into a series of TSPs with a small number of viewpoints that can be calculated online. Furthermore, in consideration of the safety constraints of the UAV, an A* planner is used to connect each viewpoint, and the distance of the trajectory is regarded as the travel cost. Eventually, the shortest trajectory with minimization of task completion time that travels through all these viewpoints can be generated by solving TSPs. During both the exploration and exploitation stage, the online sensing and positioning module plays a significant role in providing robust localization and mapping information. We utilize a Lidar-based simultaneous localization and mapping (SLAM) [19] to calculate the local position and update the surface point cloud of the target building in real time. Euclidean distance map can be built with the acceleration of GPU [20]. Furthermore, the density map can be updated based on the surface point cloud, the reconstructability, the Euclidean distance map, and the local position. It is used to predict the distribution of viewpoints to achieve balanced workload deployment.

With the combination of the above three modules, multiple UAVs can fly autonomously in an unknown environment without the 3D model of the target building. UAVs can cover the entire building with an optimal and collision-free trajectory and reach their best region. After task assignment, UAVs can follow the generated trajectory and collect the images at specified locations and orientations. The collected images can be used in reconstruction and inspection for further applications.

3 Methodology

3.1 Density map construction

In multi-UAV region division, especially for the centroid Voronoi-based method, just using an occupancy grid map or Euclidean distance map to represent the environment is not enough. A map that represents the information about the environment is needed to guide the UAV. To reduce the complexity of tasks and improve time efficiency, the whole area needs to be divided into non-intersecting regions uniformly for each UAV, and the best region division is highly related to the viewpoints for inspection in each region. However, before getting the 3D model of the target building, the distribution of the viewpoints is unknown. As a result, a density map needs to be built to predict the distribution of the viewpoints over the environment W.

A common method for creating a density map involves utilizing an occupancy grid map to divide the environment into discrete spaces, each represented by a voxel that holds a predicted density value. This density value is influenced by two factors: the reconstructability and the distance to the obstacles. The distance to the obstacles can be obtained from the Euclidean distance map built by the GPU acceleration method [20]. Furthermore, the calculation of reconstructability is based on two theorems: stereo camera-based theorem [21] and light field-based theorem [22].

As shown in Fig. 3, the reconstruction of two images can be simplified into a stereo camera image matching and depth estimation process. The depth estimation error of a stereo camera is determined by the baseline between two cameras and the distance between the observed object and the cameras [23]. Based on this definition, the reconstructability can be calculated by the view correlation relationship between each viewpoint. As shown in Fig. 4. α is the parallax angle between two viewpoints V_1 and V_2 , and s is the surface point of the building. Based on the stereo camera matching theorem, the error in reconstruction grows as the distance increases and the parallax decreases. Concurrently, the capability to reconstruct points using stereo-image matching reduces when dealing with a larger parallax. As a result, we define the view correlation relationship between two viewpoints as follows:

$$c(\alpha) = \frac{1}{1 + e^{-(\alpha - \alpha_1)}} \left(1 - \frac{1}{1 + e^{-(\alpha - \alpha_2)}} \right),$$
 (1)

where α_1 and α_2 are the lower and upper bounds of the parallax angle.

Furthermore, according to the light field theorem, as the camera gets farther from the measured object, the effective field of view of the camera becomes smaller. To determine which surface points are effect, the effective field of view needs to be calculated. We define θ_{max} being the maximum field of view of each viewpoint, d_i being the distance between V_i and the surface f. d_{\min} is the minimum attenuation distance and γ_{decay} is the decay parameter. When the distance between the viewpoint and the surface is less than the minimum attenuation distance, the effective field of view θ_{eff}^i will not attenuate. Based on the above definition, the effective field of view of each viewpoint V_i can be calculated by Eq. (2)

$$\theta_{\rm eff}^{i} = \theta_{\rm max} e^{\frac{-\max(d_i - d_{\rm min}, 0)}{\gamma_{\rm decay}}}.$$
(2)



Fig. 3 The relationship between stereo camera depth estimation and the reconstruction process



Fig. 4 View correlation relationship

After getting the effective field of view of each viewpoint, we can use the following equation to determine if the point *s* is in the field of view of V_i :

$$\Gamma_{V_i}^s = \begin{cases} 1, & \text{if } \left| \arccos\left(\frac{\boldsymbol{n}_s \cdot \boldsymbol{v}_i}{\|\boldsymbol{n}_s\| \|\boldsymbol{v}_i\|}\right) \right| < \theta_{\text{eff}}^i, \\ 0, & \text{otherwise,} \end{cases}$$
(3)

where n_s represents the normal of point *s* and v_i represents the vector of V_i and *s*. As shown in Fig. 5, when the distance d_i is larger than d_{\min} , the effective field of view becomes smaller as the distance becomes farther.

The density value is also related to the distance to the obstacles, because the UAV needs to maintain a safe distance from the environment. Define the distance value of each voxel as D_i . Based on the above equations, the total reconstructability stored as a value on each voxel can be cal-



Fig. 5 Light field theorem. s_3, s_4, \ldots, s_7 can be seen in V_1 . s_2, s_3, \ldots, s_8 can be seen in V_2

culated by Eq. (4)

$$R_{i} = \sum_{j=1, j \neq i}^{N_{v}} \sum_{k=1}^{N_{s}} \Gamma_{V_{j}}^{s_{k}} c(\alpha) + \lambda D_{i}, \qquad (4)$$

where N_v is the total number of voxels, N_s is the total number of surface points, and λ is coefficient. Each voxel coordinate corresponds to a position on a map. As a result, the density map of the environment can be represented as a function of position R(p), $p \in W$ to guide the UAV inspecting the building. A sample of generated density map is shown in Fig. 6.

3.2 Spatial deployment

After constructing the density map, the area coverage method needs to be designed to drive each UAV moving to its working region to achieve the best coverage of the target building. In particular, the Voronoi partition is conducive to obtaining convergence results for complete coverage problems while strictly avoiding duplicated task execution and collisions.

The Voronoi cell is used for generating non-intersecting convex regions for each UAV. For each time t, the Voronoi cell for each UAV is only determined by neighboring UAVs and the building, thus can be formed as the intersection of the following half-spaces:

- 1. n-1 half-spaces separating UAV *i* from UAV *j* with the parameters of separating hyper-planes a_{ij} and b_{ij} , where $i, j \in \{1, ..., n\}$;
- 2. One half-spaces separating UAV *i* from target building with parameters of separating hyper-planes a_{io} and b_{io} .

To establish a hyper-plane for inter-robot, we need to separate a safe region for UAV *i*, so that each point in this region is



Fig. 6 An illustration of density map in top view, where the voxel with high value means high reconstructability

closer to the position of the UAV than to other UAVs. We can calculate a_{ij} and b_{ij} by finding a perpendicular line of any two positions of UAV p_i , p_j using the following equations:

$$\begin{cases} \boldsymbol{a}_{ij} = \boldsymbol{p}_{ij} = \boldsymbol{p}_i - \boldsymbol{p}_j, \\ \boldsymbol{b}_{ij} = \boldsymbol{p}_{ij}^{\mathrm{T}} \frac{\boldsymbol{p}_i + \boldsymbol{p}_j}{2}. \end{cases}$$
(5)

In the context of constructing hyper-planes between the robot and the building, we regard the building as a bounded convex hull decided by a vertex vector $\boldsymbol{q}_o = [\boldsymbol{q}_1, \dots, \boldsymbol{q}_{n_o}] \in \mathbb{R}^{d \times n_o}$. The object is to minimize the length of the collision vector from the UAV to the building, and the \boldsymbol{a}_{io} can be quickly calculated by solving the following low-dimensional optimization problem:

min
$$\boldsymbol{a}_{io}^{\mathrm{T}}\boldsymbol{a}_{io}$$
,
s.t. $(\boldsymbol{q}_{l} - \boldsymbol{p}_{i})^{\mathrm{T}}\boldsymbol{a}_{io} \geq 1, \ \forall l \in \{1, \dots, n_{o}\}.$ (6)

We then shift the hyper-plane to be tight with the building. Thus, $b_{io} = \min \mathbf{a}_{io}^{\mathrm{T}} \mathbf{q}_{o}$.

Based on the above, for a team of *n* UAVs localized at $P = [p_1, ..., p_n]$ in an environment W, the Voronoi tessellation,

As a result, each UAV can compute its own working domain region \mathcal{V}_r^i according to the relative positions of others in a distributed fashion.

To evaluate how well the structure is inspected by the UAVs, we can use the following coverage function to optimize the location of deployment based on the density map:

$$\mathcal{H}(\boldsymbol{P}) = \sum_{i=1}^{n} \mathcal{H}_{i}(\boldsymbol{P}) = \sum_{i=1}^{n} \int_{\mathcal{V}_{i}^{r}} \|\boldsymbol{p} - \boldsymbol{p}_{i}\|^{2} R(\boldsymbol{p}) \mathrm{d}\boldsymbol{p}.$$
(8)

Solving the above optimization problem, a gradient decent control policy u_i is designed to steer UAVs to converge to a centroidal Voronoi tessellation (CVT) [24], that is

$$\boldsymbol{u}_{i} = -\boldsymbol{u}_{\max} \frac{\boldsymbol{p}_{i} - \boldsymbol{C}_{\boldsymbol{\mathcal{V}}_{r}^{i}}}{\|\boldsymbol{p}_{i} - \boldsymbol{C}_{\boldsymbol{\mathcal{V}}_{r}^{i}}\|},\tag{9}$$

where u_{max} is the maximum velocity of UAV. Using the control policy in Eq. (9), Each UAV can cooperatively update its disjoint Voronoi cell according to its and neighbors' positions at each time step. Eventually, a load-balanced separation of the workspace can be constructed. This division, which is crucial for optimal operations, ensures that the workload is distributed evenly among the available spaces. In this specific context, each UAV is assigned a distinct portion of the building to cover.

3.3 Online exploration

We now give a detailed process of our online exploration method described in Algorithm 1. In real-world applications, each agent is equipped with a sensor with limited sensing capability. The density map that reflects the information on the environment needs to be updated online. At the beginning of the inspection task, we assume that the building is a rectangle with estimated surface points \bar{S} , and the exact surface points *S* of the building will be constantly updated as the group of UAVs moves. First, the global density map *R* is constructed based on the rough estimate of the building. It is formulated as a grid map and the initial value of R(p) is defined as

$$R(\mathbf{p}) = \begin{cases} 1, \text{ if } \mathbf{p} \in \mathcal{W} \setminus \mathcal{B}, \\ 0, \text{ if } \mathbf{p} \in \mathcal{B}, \end{cases}$$
(10)

where \mathcal{B} is the target building.

During exploring the surface of the building at each time step, the Voronoi region \mathcal{V}_i^r for each UAV *i* can be constructed

Algorithm 1 Distributed online exploration

Input: \bar{S}, W, P, B Output: $\mathcal{V}^{\mathrm{r}}(\boldsymbol{P}), S$ 1: $R \leftarrow \text{MapConstruction}(\bar{S}, W)$ 2: for each UAV $i \in n$ do while $u_i \neq 0$ do 3: 4: $\mathcal{V}_r^i \leftarrow \text{GetVoronoiRegion}(\boldsymbol{P}, \mathcal{B})$ 5: $u_i \leftarrow \text{ChooseNextMove}(\mathcal{V}_i^{\mathrm{r}}, R)$ 6: $\boldsymbol{p}_i \leftarrow \boldsymbol{p}_i + \boldsymbol{u}_i$ $S \leftarrow \text{UpdateSurfacePoints}(\mathcal{B})$ 7: for each $i_n \in N_v$ do 8: 9: $R_{i_n} \leftarrow \text{UpdateLocalMap}(S, W)$ 10: end for $R \leftarrow \text{UpdateMap}(R_1, R_2, \ldots, R_{N_n})$ 11: 12. end while 13: end for

based on the relative positions with other UAVs and the building. Then, the control policy in Eq. (9) drives each UAV to cover the building guided by the density map. During the movement, the shape of the surface of the building is gradually updated according to the limited sensor range and the point clouds generated by Lidar SLAM. The point clouds that are in the building area \mathcal{B} are extracted to replace the initial surface points. Meanwhile, the density map can be updated based on the surface points and the relative position using Eq. (4). After several iterations of these processes, a group of agents will complete the exploration of the unknown environment. Subsequently, a spatial deployment can be formulated, aiming to offer locally optimal sensor coverage across the structure.

3.4 Exploitation

Upon completion of coverage exploration, the multi-agent system has access to environmental and structure surface information *S*. The online exploitation module is performed to let the UAVs collect the reconstruction and inspection data. First, N_p viewpoints are generated according to the shape of the surface of the target building. The viewpoints are generated in the normal direction of the surface with the consideration of the shooting distance and obstacles [25].

To consider the endurance of battery life during flight, we need to uniformly divide the number of viewpoints to be traveled N_p for each UAV on each flight. Partitioning N_p into *m* clusters will be equivalent to a problem that divides the working area of each UAV \mathcal{V}_r^i into *m* smaller Voronoi cells. Therefore, capacity-constrained Voronoi diagrams are introduced to solve this clustering problem.

Consider a set \boldsymbol{P}_m of m generators determines a new Voronoi partition $\mathcal{V}_r^i(\boldsymbol{P}_m) = \{\mathcal{V}_1^{\text{vpt},i}, \dots, \mathcal{V}_m^{\text{vpt},i}\}$ in the region $\mathcal{V}_r^i, \boldsymbol{p} \in \mathcal{V}_r^i$. Denote the capacity $|\mathcal{V}_k^{\text{vpt},i}|$ of a generator $\boldsymbol{p}_k \in \boldsymbol{P}_m$ with respect to its Voronoi region $\mathcal{V}_k^{\text{vpt},i} \in \mathcal{V}_r^i(\boldsymbol{P}_m)$ as follows:

$$|\mathcal{V}_{k}^{\text{vpt},i}| = \int_{\boldsymbol{p}\in\mathcal{V}_{k}^{\text{vpt},i}} R(\boldsymbol{p}) \mathrm{d}\boldsymbol{p}.$$
(11)

To improve the flight efficiency and reduce the total execution time, the distribution of *m* generators needs to satisfy the capacity constraint $C = \{c_k | k \in \{1, ..., m\}\}$, where c_k is defined as

$$c_k = \frac{1}{m} \int_{\mathcal{V}_r^i} R(\boldsymbol{p}) \mathrm{d}\boldsymbol{p}.$$
 (12)

Then, a capacity-constrained Voronoi tessellation with a capacity constraint fulfills the following equation:

$$\sum_{k}^{m} \left(|\mathcal{V}_{k}^{\text{vpt},i}| - c_{k} \right) = 0.$$
(13)

Our method first generates an initial partition that fulfills the capacity constraint for all sub-regions. Similar to the above centroid Voronoi tessellation, this partition is then optimized so that *m* generators are relocated to the centroids of their sub-regions, while simultaneously maintaining the capacity for each site. As a result, a capacity-constrained Voronoi partition separates the working region of each UAV into a set of sub-regions that balance the workload for the batteries and minimize the sum of the shortest distances from the viewpoints to assigned generators.

After viewpoints distribution, the N_p number of viewpoints is divided into several small subsets N_p^m . For each subset, to get the shortest path that travels through all these viewpoints, we formulate it into a collision-free trajectorybased TSP. Based on the occupancy grid map, every two viewpoints are connected by A* planner [26]. A collision-free path can be extracted, and the distance of the path is the travel cost c_{ij} between every two viewpoints. As a result, it becomes an asymmetric traveling salesman problem. The formulation is shown in Eq. (14). v_{ij} represents the connection relationship between viewpoint v_i and v_j . u_i and u_j are any real numbers. The objective function of this optimization problem is to minimize the total distance traveling through all these viewpoints

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} v_{ij},$$

s.t.
$$\sum_{j \in \bar{N}} v_{ij} = 1, \ \forall i \in N_p^m,$$

$$\sum_{i \in \bar{N}} v_{ij} = 1, \ \forall j \in N_p^m,$$

$$u_i - u_j + nv_{ij} \le n - 1, \ \forall i, j \in N_p^m,$$

$$v_{ij} \in \{0, 1\}, \ \forall i, j \in N_p^m,$$

$$u_i, u_j \in \mathbb{R}, \ \forall i, j \in N_p^m.$$

(14)

Because the number of each set is small, the TSP of each set can be solved online. Finally, the UAVs will follow the generated paths and collect the images of the target building.

4 Applications

To verify the performance of our coverage control within a real-time operating platform, particularly when faced with limited computational resources, we have implemented the system into a real-world inspection application. The inspection task is conducted by three fully autonomous UAVs. These UAVs are tasked with the role to inspect and reconstruct an academic building, as shown in Fig. 1.

The building exhibits a range of defects of various magnitudes, with some necessitating more in-depth inspections to ensure the safety of the surrounding infrastructure. To completely cover this building, the UAVs must access roughly 900 different viewpoints for image capture. Carrying out this task is a lengthy process for a single UAV, especially considering the vast scale of the environment. As a result, to accomplish these tasks and decrease the total time needed, it is important to utilize a spatially load-balanced deployment of multiple UAVs. This approach allows for more efficient use of resources and a significant reduction in the time required to complete the inspection.

We deploy area coverage, sensing, viewpoints generation, and path generation module on these UAVs. Starting from one corner of the building, three UAVs autonomously fly to their own load-balanced areas based on the reconstructability obtained from the density map. The flight path and the generated density map are shown in Fig. 7. They coordinate with other UAVs to expand the range of perceived buildings gradually. They collaborate with other drones to progressively broaden the scope of buildings they can perceive until reaching the best regions. This figure suggests that our coverage control strategy encourages each agent to investigate the unexplored surroundings in a decentralized manner, ensuring safety throughout the dynamic coverage operation.

After reaching the best regions, as shown in Fig.8a, the 3D point cloud of this building can be obtained and the view-points can be generated along the vertical direction of the surface. Furthermore, the viewpoints are divided into several sub-regions by considering the battery endurance of the UAV.

Finally, the viewpoints are connected by solving the TSP and UAVs travel all of the viewpoints in the shortest path to collect images, which can be used by AI detection. The region division result and travel paths are shown in Fig. 8b, c. The paths with different colors represent the task of each flight. 3D reconstruction [27] is conducted using these collected images to verify the effectiveness. The result shown in Fig. 9 can achieve millimeter-level accuracy. To further evaluate the generated model, we conduct a scan to BIM [28] process. The



Fig. 7 Coverage process of our algorithm. Top row: paths of 3 UAVs taking off from a corner in an obstacles-dense environment. Bottom row: density map information

generated building information model is shown in the middle row of Fig. 9. The RGB and infrared information that reflect the found defects are shown in the right row. Despite some instability of the GPS signal and uncertainty of localization, these three UAVs successfully finished the inspection and reconstruction task using our proposed framework.

5 Experiment

To elaborate on the proposed explore-then-exploit cooperative framework, we conduct several experiments in both simulators and real scenarios. We use Gazebo as the simulation platform and Pixhawk as the flight controller. The simulation and algorithms are running on a laptop with an Intel i7-9700 CPU and 16 GB RAM. The parameters of our algorithm are set as follows. The resolution of the grid map is 1 m for all axes. To meet the real-world requirements, the maximum velocity of the agent is 2 m/s, and the time interval is chosen as $\Delta t = 0.1 \text{ s}$. We assume that each UAV is equipped with a lidar with the same range radius.

5.1 Validate time efficiency

To show the real-time performance of our proposed work, we select mTSP as the benchmark and conduct the experiments on a large-scale environment with different numbers of UAVs, as shown in Fig. 10. The scene is a school with 10 m high, 20 m wide, and 10 m long. We use the calculation time (CT) per re-planning iteration and total inspection time (IT) as the evaluation metrics. For a fair comparison, because the mTSP can only work in known environments, we assume that the 3D model of the building is known, and mTSP has converged. As shown in Table 1, CT per iteration for each UAV in our method remains the same basically, which shows that our method is intractable for a large-scale multi-UAV sys-



Fig. 8 Illustration of point clouds and viewpoints



Fig. 9 Sample of reconstruction result, building information model, and collected images

tem and satisfied with real-time requirements. On the other hand, as the number of coordinated UAVs becomes larger, IT has decreased dramatically shorter than the mTSP method. This is because our method distributes the multi-UAV into more workload-balanced working regions. It is observed that compared with others, our proposed work gains higher time efficiency satisfying the real-time requirements.

5.2 Validate robustness

We continue to assess how well our proposed system handles a situation where a single UAV fails during the mission. To check how broadly our algorithm can be applied, we run it in a different, messy environment. We keep the dimensions of this environment and the target building identical to those used earlier. The obstacles are scattered randomly and come from different heights. As depicted in Fig. 11, five agents, beginning from random starting points, collab-



Fig. 10 Simulation environments

oratively aim to cover the building extensively, guided by the reconstructability data from the density map. When the light blue agent becomes non-functional at the 7.6 s mark, its nearby agents (the dark blue and red ones) adjust their oper-

Table 1 Scalability analysis

| Metric | 3 UAVs | | 5 UAVs | | 7 UAVs | | 9 UAVs | |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| | CT (s) | IT (s) |
| mTSP | _ | 374 | _ | 225 | _ | 164 | _ | 134 |
| Ours | 0.205 | 353 | 0.194 | 207 | 0.197 | 138 | 0.201 | 98 |

Comparison of total inspection time and calculation time of our proposed method and mTSP over different numbers of UAVs in a simulation environment



Fig. 11 Coverage paths of 5 UAVs taking off from a random initial configuration. One of the UAVs (light blue) suddenly failed and others dynamically adjust the responsible area

ational areas to make up for the coverage shortfall due to the faulty drone. Ultimately, all agents manage to fully cover the building by employing our distributed spatial deployment algorithm, achieving this at the 14.6 s mark. It is noticeable that even with a UAV failure mid-flight, the remaining drones can still successfully complete the mission by adapting to the changes in the information density map.

6 Conclusion

This paper proposes an explore-then-exploit multi-UAV cooperative framework for building inspection and reconstruction. The proposed framework guarantees the coverage of the target buildings, balanced workload assignment for each UAV, and realizes online data collection without a preobtained 3D model of the target building. Despite an increase in obstacle density, the computational time of our suggested algorithm remains consistent, and the collision avoidance technique guarantees a safe region. We have effectively carried out a comprehensive inspection of a large industrial building using the system we developed. The high-quality images gathered by the multi-agent system can help property managers identify any structural issues that need to be addressed. The system we have proposed holds substantial value for the maintenance and management of industrial facilities. Furthermore, the comparison results show that our method has higher time efficiency compared with other method and the experiments verify the effectiveness of our method in different large-scale environments.

Acknowledgements The authors would like to thank the team members and visitors of the Unmanned Systems Research Group at the Chinese University of Hong Kong, particularly, Ruoyu Wang, Yizhou Chen, Yu Zhai, Zuoquan Zhao, Zixuan Guo, Guidong Yang, and Chenggang Wang, for their help and assistance in conducting physical inspection of the industrial building.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecomm ons.org/licenses/by/4.0/.

References

- Menendez, E., Victores, J. G., Montero, R., Martnez, S., & Balaguer, C. (2018). Tunnel structural inspection and assessment using an autonomous robotic system. *Automation in Construction*, 87, 117–126.
- Martinez-Carranza, J., & Rojas-Perez, L. O. (2022). Warehouse inspection with an autonomous micro air vehicle. *Unmanned Systems*, 10, 329–342.
- Khan, S., Tufail, M., Khan, M. T., Khan, Z. A., Iqbal, J., & Wasim, A. (2022). A novel framework for multiple ground target detection, recognition and inspection in precision agriculture applications using a UAV. Unmanned Systems, 10, 45–56.
- Huang, R., Xu, Y., Hoegner, L., & Stilla, U. (2022). Semanticsaided 3d change detection on construction sites using UAV-based photogrammetric point clouds. *Automation in Construction*, 134, 104057.
- Ortega, A., Muñoz, J. Reyes, McGee, M., Choudhuri, A. R., & Flores-Abad, A. (2020). Drone inspection flight path generation from 3d cad models: Power plant boiler case study. In *AIAA Scitech* 2020 Forum (pp. 1091).
- Ali, H., Batai, S., & Akynov, A. (2018). Vision-based robot for boiler tube inspection. In *Proceedings of International Conference* on Computational Vision and Bio Inspired Computing (pp. 475– 482).
- Phung, M. D., Quach, C. H., Dinh, T. H., & Ha, Q. (2017). Enhanced discrete particle swarm optimization path planning for UAV visionbased surface inspection. *Automation in Construction*, 81, 25–33.
- Tan, Y., Li, S., Liu, H., Chen, P., & Zhou, Z. (2021). Automatic inspection data collection of building surface based on BIM and UAV. *Automation in Construction*, 131, 103881.
- Kuang, Q., Wu, J., Pan, J., & Zhou, B. (2020). Real-time UAV path planning for autonomous urban scene reconstruction. In *Proceedings of 2020 IEEE International Conference on Robotics and Automation* (pp. 1156–1162).
- Jing, W., Deng, D., Wu, Y., & Shimada, K. (2020). Multi-UAV coverage path planning for the inspection of large and complex structures. In *Proceedings of 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 1480–1486).
- Zheng, X., Wang, F., & Li, Z. (2018). A multi-UAV cooperative route planning methodology for 3D fine-resolution building model reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 146, 483–494.
- Chen, F., Lu, Y., Cai, B., & Xie, X. (2021). Multi-drone collaborative trajectory optimization for large-Scale aerial 3D scanning. In Proceedings of 2021 IEEE International Symposium on Mixed and Augmented Reality Adjunct (pp. 121–126).
- Cao, C., Zhang, J., Travers, M., & Choset, H. (2020). Hierarchical coverage path planning in complex 3D environments. In *Proceedings of 2020 IEEE International Conference on Robotics* and Automation (pp. 3206–3212).
- Peng, C., & Isler, V. (2019). Adaptive view planning for aerial 3D reconstruction. In *Proceedings of 2020 IEEE International Conference on Robotics and Automation* (pp. 2981–2987).
- Cortes, J., Martinez, S., Karatas, T., & Bullo, F. (2004). Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2), 243–255.
- Wang, X., Gao, C., Chen, X., & Chen, B.M. (July 2023). Fast and secure distributed multi-agent coverage control with an application to infrastructure inspection and reconstruction. In *Proceedings of the 42nd Chinese Control Conference, Tianjin, China* (pp. 5998– 6005).
- Collins, L., Ghassemi, P., Esfahani, E.T., Doermann, D., Dantu, K., & Chowdhury, S. (2021). Scalable coverage path planning of multirobot teams for monitoring non-convex areas. In *Proceedings of*

2021 IEEE International Conference on Robotics and Automation (pp. 7393–7399).

- Adaldo, A., Mansouri, S. S., Kanellakis, C., Dimarogonas, D. V., Johansson, K. H., & Nikolakopoulos, G. (2017). Cooperative coverage for surveillance of 3D structures. In *Proceedings of 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 1838–1845).
- Xu, W., Cai, Y., He, D., Lin, J., & Zhang, F. (2022). Fast-lio2: Fast direct lidar-inertial odometry. *IEEE Transactions on Robotics*, 38(4), 2053–2073.
- Chen, Y., Lai, S., Cui, J., Wang, B., & Chen, B. M. (2022). GPU-Accelerated incremental Euclidean distance transform for online motion planning of mobile robots. *IEEE Robotics and Automation Letters*, 7(3), 6894–6901.
- Smith, N., Moehrle, N., Goesele, M., & Heidrich, W. (2018). Aerial path planning for urban scene reconstruction: A continuous optimization method and benchmark. *ACM Transactions on Graphics*, 37, 1–15.
- Roberts, M., Dey, D., Truong, A., Sinha, S., Shah, S., Kapoor, A., Hanrahan, P., & Joshi, N. (2017). Submodular trajectory optimization for aerial 3D scanning. In *Proceedings of 2017 IEEE International Conference on Computer Vision* (pp. 5324–5333).
- Zhao, W., & Nandhakumar, N. (1996). Effects of camera alignment errors on stereoscopic depth estimates. *Pattern Recognition*, 29, 2115–2126.
- Pierson, A., & Rus, D. (2017). Distributed target tracking in cluttered environments with guaranteed collision avoidance. In 2017 International Symposium on Multi-Robot and Multi-Agent Systems (pp. 83–89).
- Gao, C., Wang, X., Wang, R., Zhao, Z., Zhai, Y., Chen, X., & Chen, B. M. (2023). A UAV-based explore-then-exploit system for autonomous indoor facility inspection and scene reconstruction. *Automation in Construction*, 148, 104753.
- Ducho, F., Babinec, A., Kajan, M., Beo, P., Florek, M., Fico, T., & Jurišica, L. (2014). Path planning with modified a star algorithm for a mobile robot. *Procedia Engineering*, *96*, 59–69.
- Yao, Y., Luo, Z., Li, S., Fang, T., & Quan, L. (2018). MVSNet: Depth inference for unstructured multi-view stereo. In *Proceedings* of the European Conference on Computer Vision (pp. 767–783).
- Esfahani, M. E., Rausch, C., Sharif, M. M., Chen, Q., Haas, C., & Adey, B. T. (2021). Quantitative investigation on the accuracy and precision of Scan-to-BIM under different modeling scenarios. *Automation in Construction*, *126*, 103686.



Chuanxiang Gao received the B.Eng. degree from Northwestern Polytechnical University (NWPU), Xi'an, China, in 2020. He is currently pursing his Ph.D. degree in mechanical and automation engineering with the Chinese University of Hong Kong, Hong Kong, China. His current research interests include task planning, motion planning, and mapping.



Xinyi Wang received the Bachelor's degree from Xiamen University, Xiamen, China, in 2019, and Ph.D. degree from the Department of Mechanical and Automation Engineering, the Chinese University of Hong Kong, in 2023. Dr. Wang is currently a Postdoctoral Fellow in Hong Kong Centre for Logistics Robotics Limited and an Honorary Postdoctoral Fellow in the Chinese University of Hong Kong. Her current research interests include multiagent coordination system, cover-

age control, and trajectory optimization.



Xi Chen received the B.Sc. degree in energy and environment systems engineering from Zhejiang University, in 2009, and the MPhil and Ph.D. degrees in building services engineering from the Hong Kong Polytechnic University, in 2011 and 2017, respectively. Dr. Chen is currently a Research Assistant Professor in the Chinese University of Hong Kong. He has over 10-year experience in sustainable building technology related to the smart cities, data mining, and built environ-

ment modeling, and has led or managed multiple research projects including ARC, MOST, RGC, and consultancy projects with the local government and industry. He has published over 50 papers in peer-reviewed international journals and coauthored a book in green building and renewable application areas. Dr. Chen has been awarded the DECRA Fellow in the Australian Research Council and Fulbright Scholar in the Lawrence Berkeley National Laboratory. Moreover, he services as the editorial board member of Buildings, Energies and Advances in Applied Energy.



Ben M. Chen (S'89-M'92-SM'00-F'07) is currently a Professor of Mechanical and Automation Engineering at the Chinese University of Hong Kong (CUHK). He was a Provost's Chair Professor in the Department of Electrical and Computer Engineering at the National University of Singapore, before joining CUHK in 2018. He was an Assistant Professor in the Department of Electrical Engineering at the State University of New York at Stony Brook, USA, in 1992–1993. His current research

interests are in unmanned systems and their applications. Dr. Chen is a Fellow of IEEE and Fellow of Academy of Engineering, Singapore. He had served on the editorial boards of a dozen international journals including Automatica and IEEE Transactions on Automatic Control. He currently serves as an Editor-in-Chief of Unmanned Systems and Editor of International Journal of Robust and Nonlinear Control.