Cooperative Aerial Robot Inspection Challenge: A Benchmark for Heterogeneous Multi-UAV Planning and Lessons Learned

Muqing Cao[1,2*], Thien-Minh Nguyen[2*†], Shenghai Yuan[2], Andreas Anastasiou[3], Angelos Zacharia[3], Savvas Papaioannou[3], Panayiotis Kolios[3], Christos G. Panayiotou[3], Marios M. Polycarpou[3], Xinhang Xu[2], Mingjie Zhang[4], Fei Gao[5], Boyu Zhou[4], Ben M. Chen[6], Lihua Xie[2]

***Abstract*—We propose the Cooperative Aerial Robot Inspection Challenge (CARIC), a simulation-based benchmark for motion planning algorithms in heterogeneous multi-UAV systems. CARIC features UAV teams with complementary sensors, realistic constraints, and evaluation metrics prioritizing inspection quality and efficiency. It offers a ready-to-use perception-control software stack and diverse scenarios to support the development and evaluation of task allocation and motion planning algorithms. Competitions using CARIC were held at IEEE CDC 2023 and the IROS 2024 Workshop on Multi-Robot Perception and Navigation, attracting innovative solutions from research teams worldwide. This paper examines the top three teams from CDC 2023, analyzing their exploration, inspection, and task allocation strategies while drawing insights into their performance across scenarios. The results highlight the task's complexity and suggest promising research directions in cooperative multi-UAV systems. The simulation framework, including source code and detailed instructions, is publicly available at https://ntu-aris.github.io/caric.**

***Index Terms*—Multi-Robot Systems, Inspection, Task and Motion Planning.**

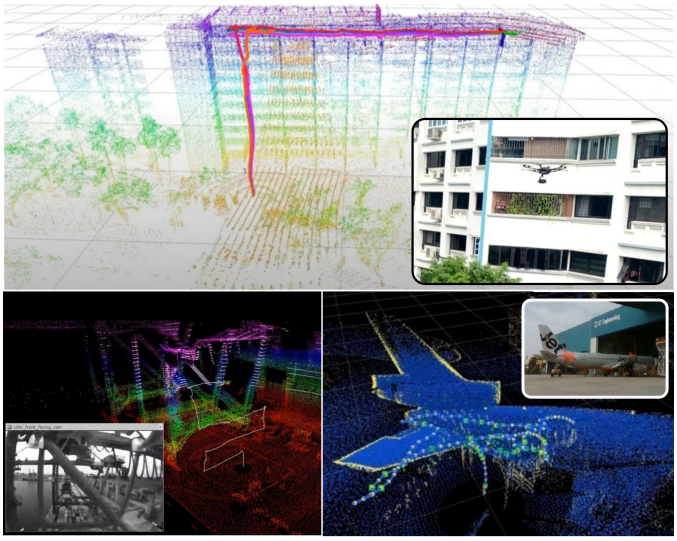


Figure 1: Examples of aerial inspection mission for building, crane, aircraft in our previous projects.

I. Introduction

Aerial robots have become widely adopted for the inspection of complex structures, such as buildings, cranes, bridges, and airplanes [1]–[3]; see Figure 1. Existing commercial systems mainly rely on human operators who manually pilot UAVs. While effective, this approach is labor-intensive, time-consuming, and prone to human error. As a result, the robotics community has increasingly focused on autonomous structural inspection using UAVs to improve efficiency and reliability.

However, most autonomous inspection solutions rely on path planning algorithms that require a detailed prior model of the structure to be inspected [1], [4], [5]. Obtaining such models often requires a separate data collection phase, which adds to operational complexity and cost. Efforts to perform online modeling and inspection using onboard cameras have been explored [6], [7], but cameras are generally inefficient for mapping due to their limited field of view and sensitivity to lighting conditions. LiDAR sensors, on the other hand, are highly effective in capturing dense 3D information over large areas quickly, making them ideal for creating a structural prior for inspection.

Given the cost and payload limitations of equipping all UAVs with LiDAR, heterogeneous teams of drones, some equipped with both LiDAR and camera for mapping and inspection and others with only cameras, offer a promising solution. This setup allows for fast and efficient online modeling and inspection, leveraging the strengths of both sensors to balance cost-effectiveness and operational efficiency. However, designing effective strategies for task allocation, path planning, and coordination in heterogeneous teams is a complex problem, especially with practical limitations such as battery and communication constraints. While single-robot or homogeneous multi-robot inspection systems have been studied extensively [8], [9], heterogeneous multi-UAV inspection remains underexplored, leaving critical gaps in research and development.

Despite the abundance of research in structural inspection planning, most of the works are tested in some in-house simulation environments with user-specified structures. There is a lack of a readily available simulation framework that includes a range of realistic structures and considers

[1] Robotics Institute, Carnegie Mellon University. [2] Centre for Advanced Robotics Technology Innovation (CARTIN), Nanyang Technological University. [3] KIOS Research and Innovation Center of Excellence, Department of Electrical and Computer Engineering, University of Cyprus. [4] School of Artificial Intelligence, Sun Yat-sen University. [5] Zhejiang University. [6] Department of Mechanical and Automation Engineering, Chinese University of Hong Kong.

[*] Equal contribution. [†] Corresponding author: Thien-Minh Nguyen (thienminh.npn@ieee.org)

This research is supported by the National Research Foundation, Singapore, under its Medium-Sized Center for Advanced Robotics Technology Innovation (CARTIN).

the challenges of a real-world inspection operation, such as communication loss. Furthermore, the evaluation of different inspection planning algorithms presents a challenge. Most of the existing works apply evaluation criteria focusing on the completeness and efficiency including the inspection coverage rate, overall inspection duration, and total length traveled by the UAVs, however, these metrics do not reflect the quality of inspection, i.e. whether the video or images taken during the inspection are in sufficient quality for the detection of structural weakness. It is possible to have the UAVs scan the entire structure in a short time but get only blurry images with low levels of detail that are not useful for inspection. There is a need for a benchmark toolset that considers the challenges of a multi-UAV inspection operation in a realistic setting while providing a comprehensive evaluation of the algorithms applied.

To address these gaps, we introduce the Cooperative Aerial Robot Inspection Challenge (CARIC) to stimulate innovative solutions for heterogeneous UAV inspection and provide a platform for researchers and practitioners to test their multi-robot inspection task and motion planning algorithms. The challenge offers a lightweight and ready-to-use simulation framework compatible with ROS1 and ROS2, featuring a team of high-cost mapping and exploration drones and low-cost inspection drones tasked with inspecting simulated industrial sites under conditions mimicking real-world operations. The simulation framework provides sample environments with diverse structural properties and integrates practical operational constraints, including the lack of prior structural model and the communication loss beyond line of sight. The inspection performance is evaluated using realistic metrics considering image quality, including blurriness and resolution, to ensure the captured data meets the standards for structural analysis.

The CARIC framework has been successfully adopted in two competitions, drawing wide participation and recognition from the research community. The first CARIC competition was held in conjunction with the IEEE Conference on Decision and Control (CDC) 2023, and the second was part of the IROS 2024 Workshop on Multi-Robot Perception and Navigation Challenges in Logistics and Inspection Tasks. These competitions have showcased diverse and innovative approaches to the multi-UAV inspection problem.

In this paper, we present and analyze the strategies adopted by the top three teams from the first CARIC competition, highlighting key insights and critical lessons learned to inform and inspire future research in multi-UAV inspection. The competition results show that achieving effective and efficient multi-UAV inspection is a nontrivial challenge, necessitating careful integration of exploration and inspection phases, accurate workload estimation for balanced task allocation, and robust mechanisms to manage communication and path-planning failures to ensure operational reliability. Notably, no single approach achieved consistently strong performance across all test scenarios, clearly indicating the need for continued research advancements in this domain.

This paper is organized as follows. In Section II, we review relevant robot planning benchmarks, competitions and evaluation metrics for inspection planning. Section III details our proposed benchmark framework and evaluation metrics. Section IV presents the rules and results of the CARIC competition at CDC 2023, and Section V details the approaches designed by the top-performing teams. Section VI analyzes the strategies of the teams and draws insights and lessons from their performance. Section VII concludes the paper.

II. Related Work

A. Robot Planning Benchmark Platforms and Competitions

The recent fast development of robot planning algorithms has stimulated tools and competitions to evaluate and benchmark the performance of algorithms. The DogeDrone and Barn Challenge [10] evaluates the performance of goal-driven planning algorithms for UAVs and UGVs in complex simulated environments. For exploration planning, simulation-based platforms [11], [12] have also been proposed to benchmark the efficiency of learning-based and classical approaches. Regarding inspection and coverage planning, although some research has open sourced their algorithms and simulation environments [1], they offer limited simulation environments and instructions for users to integrate their algorithms. The ICUAS 2023 UAV Competition provides instructions for user integration but offers a single-UAV simulation environment for inspecting a small factory building. In CARIC, we offer a benchmark toolset tailored for implementing and benchmarking multi-robot inspection planning algorithms, with detailed instructions for installation, control interface, and evaluation to facilitate usage. We offer diverse simulation environments featuring different complexities and scales of structures (shown in Figure 3) and simulate practical constraints such as communication dropout. CARIC is integrated into both ROS1 and ROS2 for easy usage.

B. Evaluation Metrics of Inspection Planning

Effective inspection planning requires evaluating three key aspects: inspection quality, completeness, and efficiency. Completeness is commonly assessed using the coverage rate, which measures how much of the structure's surface has been inspected [5], [8]. Efficiency is typically quantified by metrics such as total path length [8], [13], inspection duration [7], and the number of viewpoints required to complete the inspection [5], [13]. Many existing works aim to improve completeness and efficiency while maintaining a satisfactory inspection quality. This is often achieved by imposing constraints on the UAV's viewing angle and distance from the target surface [1], [6], [7], ensuring that the resulting images have adequate spatial resolution for identifying structural details. To quantify spatial resolution more precisely, some studies use ground sampling distance (GSD)—the physical distance between adjacent pixel centers projected onto the target surface. For example, [14] and [15] adopt GSD as a key metric, with [14] explicitly optimizing it along with coverage rate and total time in their planning algorithm. Other quality-focused metrics include detection accuracy of defects or features [3], and model reconstruction accuracy based on comparison with ground truth data [4], [9]. However, image quality is not determined by spatial resolution alone. Factors

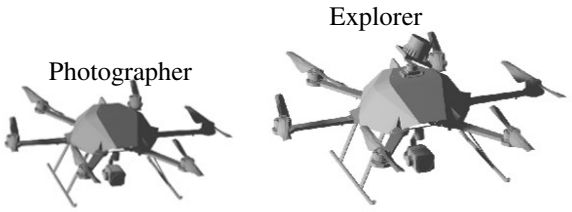


Figure 2: The heterogeneous UAV team.

such as blurriness and exposure also play critical roles in determining whether the collected data is useful for inspection purposes [16]. Ignoring these can result in high-resolution images that are unusable due to motion blur or poor lighting.

To address this, CARIC adopts a more holistic evaluation metric for inspection performance. Given the limited flight time of UAVs, we treat inspection time as a fixed constraint and focus our scoring on the quality of collected data. Rather than simulating full image effects, which is computationally intensive, we apply analytical equations to efficiently approximate image quality, accounting for both spatial resolution and blurriness. In our framework, inspection completeness is measured by the number of interest points successfully detected, while the overall inspection score incorporates each detection's quality, which encourages strategies that balance wide coverage, high-resolution views, and motion stability.

III. Benchmark Overview

A fleet of UAVs is tasked with inspecting infrastructures such as building facades, cranes, and airplanes by capturing images of the surfaces and looking for locations with a high risk of defects. Unlike conventional approaches that rely on detailed prior models, the UAVs are provided only with bounding boxes that encapsulate the structures. Therefore, the UAVs should explore the unknown volume to obtain the details of the surfaces to be inspected. During the inspection process, some interest points on the surface are identified as locations with high vulnerability and, therefore, need to be closely inspected with high-resolution images. The objective of the multi-UAV system is to achieve the highest possible inspection score by capturing as many interest points as possible with the highest possible score for each point.

The simulation is implemented using the Gazebo simulator and RotorS [17]. On our website, we provide instructions for users to implement their algorithms in both ROS1 and ROS2. The following sections define the heterogeneous UAV fleet, sensor data, control and communication requirements, sample scenes, and score calculation.

A. UAV Fleet

The benchmark features a heterogeneous UAV team comprising of N drones, categorized as explorers and photographers (Figure 2):

- Photographer: A small UAV carrying an inspection camera placed on a motorized gimbal. The main task of a photographer is to capture images of the points of interest. This setting is typical of a commercially available camera-equipped UAV, such as DJI Mavic 3.
- Explorer: with reference to an autonomous LiDAR-based inspection UAV [1], we designed the explorer to be larger than a photographer and equipped with an inspection camera and a Lidar placed on a motorized gimbal. Besides image capturing, an explorer is capable of generating a point cloud map of the surroundings.

The team composition is flexible, with N_p photographers ($N_p \in 0, \ldots, N - N_e$) and N_e explorers. Note that besides the UAV, a *control station* is also present to represent human supervision and tallies the final score for the mission.

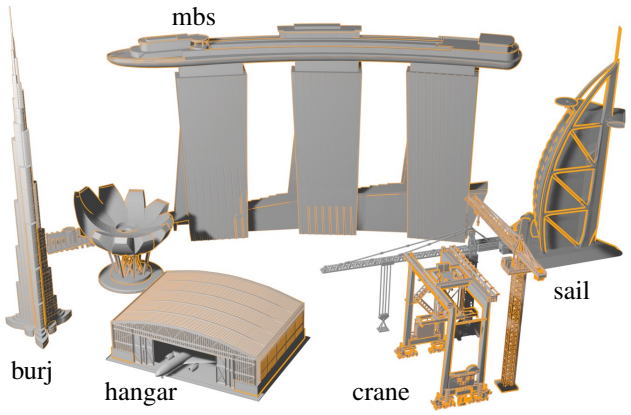


Figure 3: Sample scenarios in CARIC.

B. Sensor Data

Each UAV has access to the following basic information during simulation:

- Odometry: Ground truth pose of the UAV in the global frame of reference.
- Gimbal pose: Relative orientation of the inspection camera with respect to the UAV body frame.
- Inspection score: Each captured image includes a computed score that reflecting the quality of inspection at the interest points, detailed in Section III-E.
- LiDAR (Explorer only): A rotating LiDAR scanner generates 3D point cloud data of the surroundings at 10Hz.

Besides, we assume an underlying multi-robot localization and mapping system is available. Hence the UAVs can access certain information from *the neighbours that are in its LoS*, namely *odometry* and *key frame point cloud*. These data ensure that the *photographers* can be aware of the neighbors as well as the surroundings.

C. Control and Communication

We implement a so-called `unicon` package in the CARIC stack to allow user to control the UAVs in two ways:

- Partial control: users may send target position or velocity or acceleration and target yaw angle to the controller.

[1] https://enterprise-insights.dji.com/blog/m300-rtk-emesent-hovermap-autonomous-underground-flight

- Full control: users may send a command including the target position, velocity, acceleration, and yaw.

The controller node will compute lower-level control inputs to drive the UAV to the target states. The controller will ignore a command exceeding the kinematic constraints. For camera control, users can adjust the camera's direction by controlling gimbal pitch and yaw, and trigger image capture via a command.

Communication between UAVs is restricted to LoS. A UAV can only send or receive messages from another UAV if they have direct visibility. We implemented this communication scheme via a so-called `ppcomrouter` node, which observes the LoS condition between objects, and relays or drops the messages accordingly.

D. Sample Scenarios

We provide several sample structures for testing, as shown in Figure 3, which include structures with diverse properties (dimensions, geometric complexity, surface flatness, etc) to enable evaluation of the generalizability of the algorithms:

- Marina Bay Sands (MBS): the iconic structure of Singapore consisting of three large buildings connected by an overhanging bridge and a smaller structure (ArtScience Museum) similar to the shaper of a lotus flower.
- Hangar: A commercial passenger aircraft parked in a hangar for inspection.
- Crane: A typical crane structure at a port harbor consisting of several connected long and thin beams.
- Burj Khalifa: The tallest building in the world with a gradually reducing horizontal section.
- Burj Al Arab: The iconic building of Dubai consisting of a large curved facade.

E. Evaluation Criteria

The performance of the multi-UAV planning strategy is judged based on the **total score of the captured interest points received by the control station**, which is denoted as Q and defined as follows:

$$Q = \sum_{i=1}^{I} \max_{j \in \{1,\dots N\}} q_{i,j}, \tag{1}$$

where I is the set of interest points in the scene, N is the total number of UAVs, $q_{i,j}$ is the visual inspection score of interest point i by the UAV j.

The visual inspection score $q_{i,j}$ is computed as follows:

$$q_{i,j} = \max_{k \in [0,K]} q_{i,j,k}, \quad q_{i,j,k} = q_{\text{seen}} \cdot q_{\text{blur}} \cdot q_{\text{res}}, \tag{2}$$

where K is the total mission time, $q_{i,j,k}$ is the visual inspection score of interest point i obtained by UAV j at a time k, $q_{\text{seen}} \in \{0,1\}$, $q_{\text{blur}} \in [0,1]$, $q_{\text{res}} \in [0,1]$ are the line of sight (LOS), motion blur, and resolution metrics, which are elaborated in Sections III-E1, III-E2 and III-E3, respectively. Fig. 4 shows an example of the interest points (red squares) with the scores.

The value Q will be calculated at the end of each mission, where only the scores of UAVs that successfully completed the mission without collision be considered.

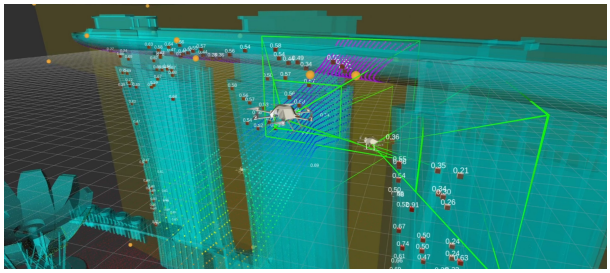


Figure 4: Example of the scores for captured interest points.

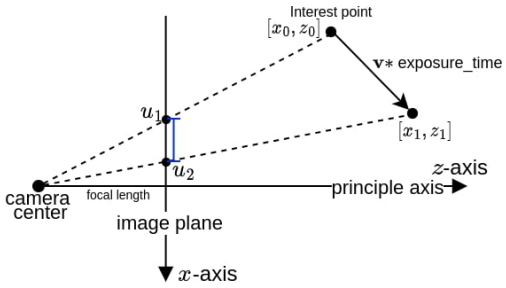


Figure 5: Illustration of Motion Blur.

1) Line of sight and field of view: The term q_{seen} is a binary-valued metric value that is 1 when the interest point falls in the field of view (FOV) of the camera, and the camera has a direct line of sight (LoS) to the interest point (not obstructed by any other objects), and 0 otherwise.

2) Motion blur: The motion blur metric q_{blur} is based on the motion of the interest point during the camera exposure duration τ (a provided value) [16]. It can be interpreted as the number of pixels that an interest point moves across during the exposure time, i.e.:

$$q_{\text{blur}} = \min\left(\frac{c}{\max\left(|u_1 - u_0|, |v_1 - v_0|\right)}, 1.0\right), \tag{3}$$

where c is the pixel width and $\|u_1 - u_0\|$, $\|v_1 - v_0\|$ are the horizontal and vertical movements on the image plane that are computed by:

$$u_0 = f \cdot \frac{x_0}{z_0}, u_1 = f \cdot \frac{x_1}{z_1}, v_0 = f \cdot \frac{y_0}{z_0}, v_1 = f \cdot \frac{z_1}{z_1}, \tag{4}$$

$$[x_1, y_1, z_1]^\top = [x_0, y_0, z_0]^\top + \mathbf{v} \cdot \tau, \tag{5}$$

with f being the focal length, $[x_0, y_0, z_0]^\top$ the position of the interest point at the time of capture, and $[x_1, y_1, z_1]^\top$ the updated position considering the velocity of the interest point in the camera frame at the time of capture, denoted as $\mathbf{v}$. The calculation of this velocity requires advance kinematic analysis, and is detailed on our website [2]. Figure 5 illustrates the horizontal motion blur by showing the horizontal (X-Z) plane of the camera frame, where the vertical motion blur can be interpreted similarly.

The movement of the interest point should be smaller than 1 pixel for a sufficiently sharp image. Interest points captured with a pixel movement greater than 1 receive a lower score.

[2] https://ntu-aris.github.io/caric/docs/CARIC_motion_blur.pdf

3) Image Spatial Resolution: The spatial resolution of the image is expressed in millimeter-per-pixel (MMPP), representing the size of the real-world object captured in one image pixel. To achieve a satisfactory resolution for defect inspection, the computed horizontal and vertical resolutions should be smaller than the desired MMPP value. Therefore, the resolution metric is computed as:

$$q_{\text{res}} = \min\left(\frac{r_{\text{des}}}{\max\left(r_{\text{horz}}, r_{\text{vert}}\right)}, 1.0\right), \tag{6}$$

where r_{des} is the desirable resolution, r_{horz}, and r_{vert} denote the resolution in the horizontal and vertical image axis, respectively. We compute spatial resolution in a way similar to the ground sampling distance (GSD) [15] with consideration of the surface normal direction.

IV. Competition at CDC 2023

The first CARIC competition was held in conjunction with the IEEE Conference on Decision and Control (CDC) in December 2023 in Singapore. The participating teams were required to submit software packages to the organizers for evaluation. The competition featured three distinct scenarios with the following team compositions:

- MBS: Two explorers and three photographers.
- Hangar: One explorer and two photographers.
- Crane: Two explorers and three photographers.

The number, locations, and sizes of the bounding boxes were unknown to participants before the competition and were provided to the software as parameters at runtime. Interest points, representing areas of potential structural vulnerability, were sampled on the surfaces within the bounding boxes. Each scenario had a strict time budget: 600 seconds for MBS and Crane, and 240 seconds for Hangar. The time starts to count when any drone takes off, allowing for offline computation before the flight.

The solution of each team was evaluated over five runs, with each test including all three scenarios. Teams were ranked according to the *max* score across the five runs. The score is reported to each team after each evaluation, and the teams are also allowed to update the solution before the deadline. To ensure fairness, all evaluations were conducted on two desktop computers with different hardware configurations: Intel Core i9-13900 CPU with an NVIDIA RTX 4080 GPU, and Intel Xeon W2295 CPU with dual NVIDIA Titan RTX GPUs.

The rankings were consistent across both hardware setups, demonstrating robustness in the evaluation. A total of eight teams submitted software packages, of which seven achieved valid scores while adhering to communication constraints. The top three teams were:

- KIOS CoE (University of Cyprus),
- XXH (Nanyang Technological University), and
- STAR (Sun Yat-sen University).

V. Winning Team Approaches

The first CARIC competition showcased diverse approaches to the multi-UAV inspection problem, with teams employing unique strategies for exploration, inspection, and task allocation. The following section presents the approaches of the top three teams, focusing on their methodologies.

A. Team KIOS CoE

1) Overview: Team KIOS CoE emerged as the top-performing team in the CARIC competition. Their approach is structured into two key stages: *Environmental Mapping* and *Cooperative Inspection*, as illustrated in Fig. 7. In the first stage, an operational volume $\mathcal{O}$ is defined based on a set of bounding boxes $\mathcal{B}$, which enclose points of interest. The area is discretized into a voxel grid, represented as a graph $\mathcal{G}$, with edges indicating feasible paths (Fig. 6a-b). Explorers traverse the grid along precomputed paths, using LiDAR to construct occupancy maps. These maps are opportunistically shared with photographers when line-of-sight (LoS) is available, enabling the creation of a global map (Fig. 6c-d).

In the second stage, UAVs collaboratively inspect operational volume. Inspection waypoints are generated around the occupied voxels (Fig. 6e). The UAV fleet employs a distributed approach to solve the Multiple Traveling Salesman Problem (mTSP), ensuring efficient path planning. Paths are executed using receding-horizon local planning, dynamically adjusting for real-time occupancy map updates and collision-free operation (Fig. 6f).

2) Environmental Mapping: The operational volume $\mathcal{O}$ is determined by calculating the smallest cuboid that contains both the infrastructure and the initial positions of the UAVs. This volume is divided into cubic voxels of size V, creating a connected graph $\mathcal{G}$, where vertices represent voxels and edges represent possible transitions between them. The adjacency matrix of $\mathcal{G}$ enables efficient navigation and path planning for UAVs.

Explorer UAVs traverse precomputed paths along the longest axis of $\mathcal{O}$, gathering LiDAR data to construct local occupancy maps. Each map identifies voxels that contain obstacles and marks them as occupied. The explorers merge these maps into a global occupancy map and communicate with photographers when there is LoS. This global map provides an initial layout of the operational area, guiding subsequent inspection by excluding paths through occupied voxels.

3) Cooperative Inspection: During the inspection phase, UAVs update their occupancy maps in real time by exchanging data whenever LoS is available. Each robot computes the waypoints around the occupied voxels to ensure thorough inspection, where a waypoint is associated with a direction vector to optimize the gimbal angles. Each robot computes a subset of the waypoints to travel by solving an mTSP distributively using a method inspired by [18]. Specifically, each robot generates a set of N paths given the current positions of the robots within LoS and then executes the path tailored to itself.

Paths are executed using a receding-horizon manner, where UAVs dynamically replan routes by solving the mTSP to handle changes in the environment. The adjacency matrix of $\mathcal{G}$ ensures that no two UAVs occupy the same voxel, maintaining collision-free operation. While following inspection

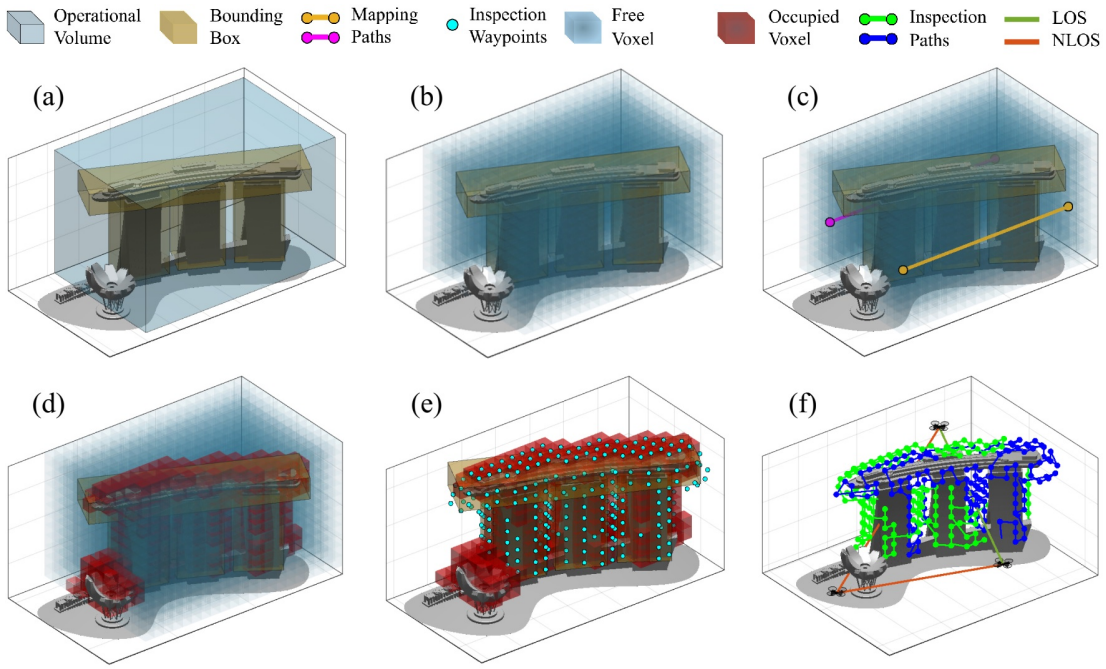


Figure 6: (a) Derivation of the operational volume, (b) Discretization of the operational volume, (c) Mapping path generation and execution, (d) Initial occupancy map generation, (e) Inspection waypoint generation, (f) Inspection path generation.

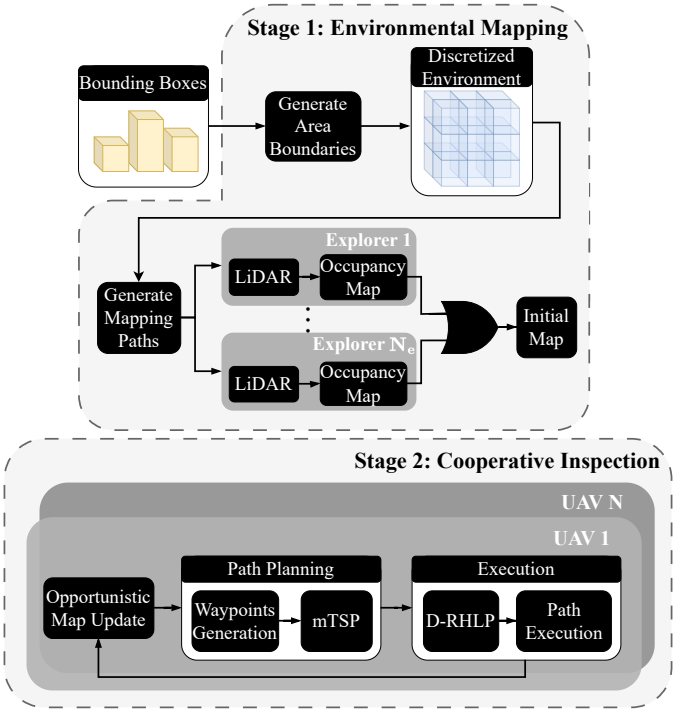


Figure 7: Overview of the proposed approach for 3D infrastructure inspection using multi-UAV system.

paths, UAVs adjust their gimbals to focus on specific areas. This iterative process of mapping, waypoint generation, and inspection continues until the mission concludes.

B. Team XXH

1) Overview: Team XXH adopts a team-based approach for multi-UAV inspection, grouping UAVs into teams based on their initial positions and roles. Tasks are allocated among teams proportionally to team size. Each explorer follows a spiral pattern to explore and map assigned regions systematically, while photographers inspect designated layers using similar trajectories.

2) Team Formation: The ground control station (GCS) initializes the process by determining the number of teams. Given N_e explorers and N_p photographers, $N_e < N_p$, the fleet is divided into N_e teams, each including one explorer responsible for environmental sensing. Photographers are assigned to teams by their proximity to the corresponding explorer.

3) Task Assignment: To allocate tasks proportionally among teams, the GCS computes an approximate minimum-length path traversing the longest dimension of all bounding boxes using a best-first search. Each team is then assigned a set of bounding boxes along this path, with the total volume of the assigned regions proportional to the team size. For balanced distribution, bounding boxes may be split along their longest side, dividing regions among teams.

4) Exploration Strategy: Each bounding box is represented as an axis-aligned voxel map, segmented into layers along the longest dimension. The explorer begins exploration at the base layer, following a spiral trajectory around the boundary of the bounding box to identify occupied and unknown voxels. **A*** search on the voxel map is used to find collision-free paths. Once the region is explored, the explorer transmits the updated map to the photographers. During exploration, photographers remain at the bounding box entry point to maintain line-of-sight communication with the explorer.

5) Inspection Strategy: With the updated map, the explorer and photographers collaboratively inspect the occupied regions within the bounding box. The layers are divided among team members, and each UAV scans its designated section using a spiral pattern. The angle of the gimbal is dynamically adjusted to ensure thorough coverage of all exposed voxel faces. To avoid collisions, UAVs share real-time position and trajectory data, with priority rules assigned to prevent deadlocks. Upon completing the inspection of one bounding box, the team transitions to the next assigned region, repeating the exploration and inspection process.

All UAVs exchange position and trajectory data. Obstacle avoidance priorities are assigned based on roles and naming conventions to resolve potential deadlocks.

C. Team STAR

1) Overview: Team STAR employs a systematic and team-based approach. Each team consists of one explorer and at least one photographer, with tasks allocated by solving an mTSP to optimize workload distribution. The explorer utilizes the efficient exploration framework FUEL [19] to explore the environment and transmit the map information to all photographers within the team. Photographers then employ a method similar to Star-Searcher [20] to inspect the received surfaces. To prevent redundant scans, photographers share their inspected surface information with one another. Concurrently, all inspected interest points are transmitted to the GCS. More details are illustrated in Fig. 8.

2) Team Formation: The team formation is similar to the approach of Team XXH, assigning one explorer to each team, and photographers are assigned to teams based on their distance from the explorers. The difference is that, to ensure balanced allocation, an upper limit of N_t photographers per team is set, where $N_t = \lceil N_p/N_e \rceil$.

3) Task Assignment: An mTSP is solved to optimize task assignment among the N_e teams for M bounding box task regions. The objective is to minimize the overall travel distance while efficiently assigning inspection tasks to each team. Specifically, a cost matrix for the mTSP solution is constructed such that the cost of inspecting each bounding box is a weighted combination of the distance from the explorer to the box's center and the box's volume. This cost computation provides an approximation of the workload to facilitate efficient assignment.

4) Explorer Strategy: The exploration framework FUEL [19] is adopted to acquire environmental information by the explorers. Frontiers are detected incrementally and then segmented into appropriately sized clusters using PCA-based clustering. Cylindrical sampling regions centered on each frontier are generated to produce viewpoints. For each frontier, the viewpoint that maximizes the coverage of frontier cells within the LiDAR field of view (FoV) is selected as the representative viewpoint. An Asymmetric Traveling Salesman Problem (ATSP) is solved based on all representative viewpoints to determine the next best viewpoint to visit. Additionally, when communication with photographers in the same team is available, the explorer transmits map chunks incrementally.

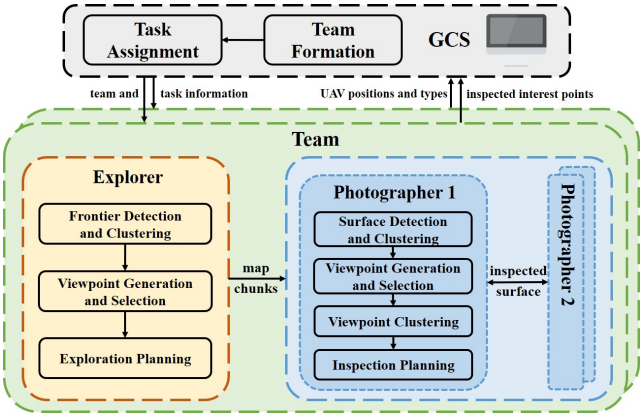


Figure 8: System overview of Team STAR's method.

5) Photographer Strategy: Upon receiving the map information from the explorer, the photographer inspects the surfaces on the map and incrementally shares the newly inspected surfaces with other photographers to avoid redundant scanning. Initially, surfaces are detected and clustered using a method similar to the explorer's, and viewpoints are generated and selected accordingly. However, due to the larger FoV of the LiDAR compared to the camera and the higher speed of the explorer relative to the photographer, the photographer often encounters a substantial number of surfaces requiring inspection. Consequently, this results in many viewpoints that need to be planned, leading to significant computational overhead. To address this issue, the visibility-based viewpoint clustering method and the hierarchical planning approach from Star-Searcher [20] are adopted. Viewpoints within a distance threshold and without occlusions between them are grouped into clusters. An ATSP is then solved over these viewpoint clusters to provide global guidance. Subsequently, for local planning, another ATSP is solved using the UAV's current position and all viewpoints within the first cluster.

6) Trajectory Planning: Fast-planner[3] is utilized for all UAVs to generate a smooth and collision-free trajectory from the current position to the next best viewpoint.

VI. Results and Discussion

Figure 9 shows the box plots of the overall scores obtained by the top three teams across tests, alongside the distribution of scores for individual scenarios. Figure 10 illustrates the paths of the UAVs and the final detection results of each scenario in their best-performing runs. Based on the approaches proposed by the teams and the observation of the teams' performance in the competition, we discuss important lessons from the weaknesses of the methods and highlight potential research directions.

A. Exploration vs Inspection

KIOS CoE and XXH employ a sequential strategy where exploration precedes inspection, and the photographers stay idle when the explorers initially explore the environment.

[3]https://github.com/HKUST-Aerial-Robotics/Fast-Planner

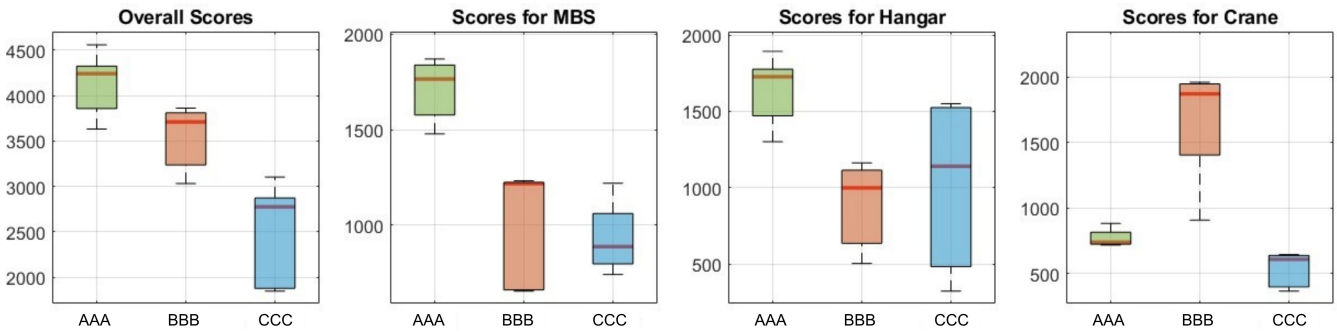
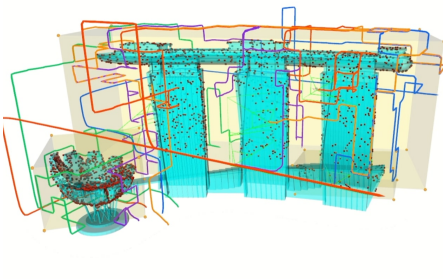
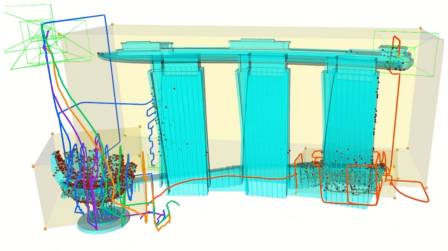


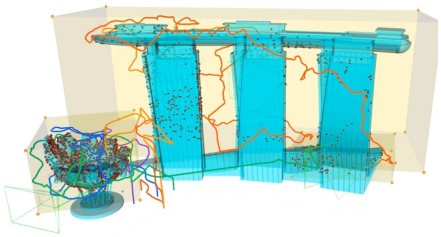
Figure 9: Box plots of the overall scores obtained by the top three teams across tests, alongside the distribution of scores for individual scenarios: MBS, Crane, and Hangar.



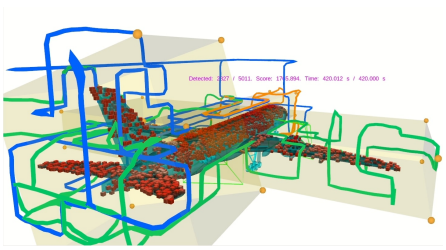
(a) KIOS CoE: 2543 points detected, 1828 score



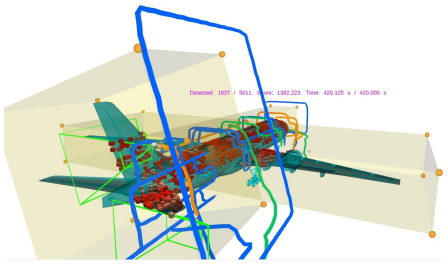
(b) XXH: 1915 point detected, 1425 score



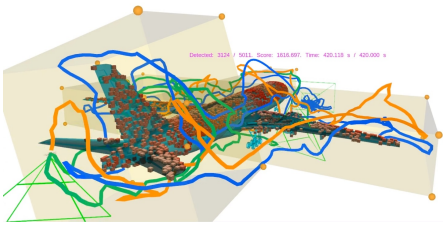
(c) STAR: 1655 points detected, 1122 score



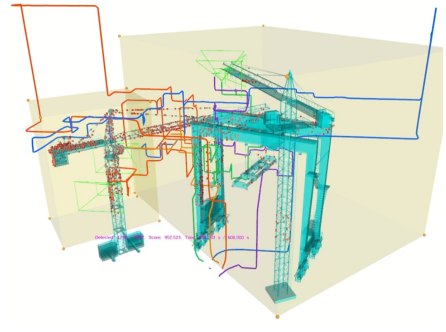
(d) KIOS CoE: 2827 points, 1765 score



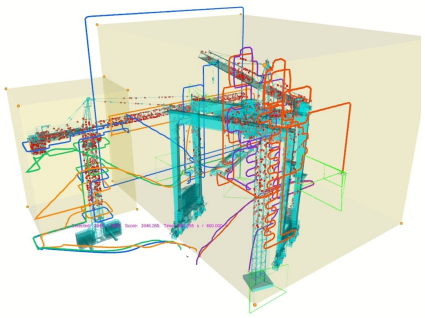
(e) XXH: 1937 points detected, 1382 score



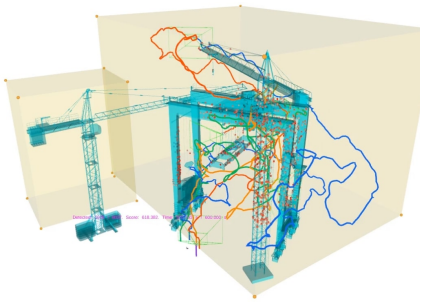
(f) STAR: 3124 points, 1616 score



(g) KIOS CoE: 1790 points detected, 950 score



(h) XXH: 2949 points detected, 1946 score



(i) STAR: 1024 points detected, 619 score

Figure 10: Illustration of the paths traveled and the points detected (red squares) for each approach for each scenario. The bounding boxes are shown in transparent yellow.

However, their approaches differ: KIOS CoE performs a rapid exploration along a straight path traversing the longest axis of the operation space containing all bounding boxes; XXH follows a more detailed layer-by-layer spiral path for each bounding box.

Figure 9 shows that KIOS CoE achieves consistently high scores in the MBS and Hangar scenarios, producing the best median and maximum scores. As seen in Figure 10a and 10d, KIOS CoE achieves extensive coverage of interest points, including a significant percentage of bounding box regions. In contrast, XXH achieves lower scores, leaving parts of the MBS facade (Figure 10b) and the airplane wings and tail (Figure 10e) being omitted from the scan. As revealed in the video recording of the tests of XXH, the explorers take a long time to conduct the layer-by-layer exploration of a bounding box. As a result, the photographers only start inspection after a long time, resulting in insufficient time to complete coverage. In contrast, KIOS's fast exploration strategy allows the team ample time to compute and execute inspection tasks, achieving higher mission scores due to more comprehensive coverage in scenarios with relatively straightforward structures.

However, rapid exploration by KIOS CoE can compromise the structural details of the map. This issue is revealed in the Crane scene, where the thin and closely spaced structures are underrepresented in the map of KIOS CoE, leading to photographers omitting many surfaces during the inspection (Figure 10g). On the other hand, XXH captures 64% more interest points than KIOS CoE (Figure 10h), demonstrating the importance of detailed exploration in complex environments.

The STAR team adopts a concurrent exploration-inspection strategy, where incremental map updates are immediately shared with photographers. While this approach works well in a constrained environment, such as Hangar (Figure 10f), it does not achieve good performance in environments with large empty volumes. This is likely because assigning bounding boxes as group tasks is inherently imbalanced, as box sizes can vary significantly, leading to uneven task allocation without a redistribution mechanism. Additionally, frontier-based exploration within large free areas of a box can result in excessive exploration of irrelevant regions, reducing efficiency.

These observations underscore a fundamental trade-off: rapid exploration maximizes inspection time but risks omitting critical details; detailed exploration provides better structural maps but delays inspection. An incremental exploration-inspection strategy may balance these trade-offs, capturing sufficient structural details without leaving photographers idle. However, such strategies demand sophisticated algorithms to ensure adaptability to diverse environments and compliance with communication constraints.

B. Inspection Quality vs Detection Rate

KIOS CoE and XXH use waypoint-based navigation, generating one waypoint per grid face to guarantee comprehensive coverage. Photographers briefly pause at each waypoint, adjusting the camera angles toward surface normals to achieve high-resolution and low-blur inspection images. In contrast, STAR optimizes smooth and continuous drone trajectories and camera movement, prioritizing dynamic feasibility and wide coverage.

While STAR's continuous yaw motion allows faster coverage and thus detects more interest points, it compromises inspection quality. For example, in the Crane scenario, STAR detects 10.5% more interest points compared to KIOS CoE, yet achieves 8.4% lower overall inspection scores. The lower inspection quality arises from the reduced resolution q_{res} due to oblique viewing angles (Equation 6), and increased blurriness q_{blur} from rapid camera movements (Equation 3). This shows that the proposed evaluation metrics effectively measure the inspection outcome by directly penalizing trajectories and camera movements that reduce image clarity, despite detecting more points.

This observation also highlights a trade-off between inspection quality and detection rate: increasing the drone speed or camera movement enables faster detection of interest points but with lower quality. A promising approach is task specialization, where one photographer focuses on rapid detection while another performs slower, high-quality inspection. Additionally, trajectory planning could incorporate inspection quality metrics to optimize drone speed and camera movement.

C. Task Allocation

KIOS CoE employs a distributed task allocation strategy where all drones collaboratively solve an mTSP to traverse viewpoints. In contrast, XXH and STAR divide robots into teams and assign regions based on bounding box volume and travel distance. However, the region-based allocation of XXH and STAR proves problematic, as reflected by their lower mission scores in the MBS scenario (Figures 10b, 10c). In XXH, four drones are assigned to the same team and commanded to inspect the ArtScience Museum (the smaller structure on the left) before moving to the main buildings. Another explorer is commanded to inspect the smaller bounding box at the bottom right of the main buildings. Ultimately, the slow pace of the inspection leaves a substantial portion of the facade unchecked. Similarly, STAR assigns more drones to inspect the ArtScience Museum than the main buildings.

These outcomes highlight how the effectiveness of task allocation directly impacts the number of interest points inspected and thus the mission scores. A good estimation of the inspection workload is critical for a fair allocation of tasks. A simple assignment strategy based on the volume of the bounding box may cause the robots to spend excessive time on small but complex structures. In contrast, KIOS CoE's distributed task allocation is based on the viewpoints to inspect the explored map of the structures. This provides a more accurate description of the workload, resulting in more efficient task allocation.

Additionally, separating a small number of robots into multiple teams may not yield efficiency gains compared to a single team due to the difficulty of workload allocation among teams. However, a single team may become infeasible when the fleet size increases due to the large communication and computation resources.

D. Performance Variance

1) Variance within the Same Scenario: Both XXH and STAR experience significant variances in performance, with best-performing runs exceeding the worst by over 1000 scores in some scenarios (Figure 9). In some runs, XXH solution failed to find a feasible path for explorers, leaving the entire team idle for the whole mission duration. Investigations reveal that XXH solution has a bug in the waypoints selection logic, ultimately leading to the failure of pathfinding. Both methods incur collisions during initialization, particularly when drones start from close positions, disrupting subsequent operations. These issues highlight the need for robust pathfinding algorithms and improved initialization protocols to enhance consistency across runs.

2) Variance across Different Scenarios: No single approach consistently achieves the best performance across all scenarios. KIOS CoE demonstrates superior performance in the MBS and Hangar scenarios but struggles with the Crane scene. Conversely, XXH excels in the Crane scenario, and STAR achieves strong results in Hangar, but both methods underperform in other scenes. As highlighted in previous sections, these variations stem from differences in exploration strategies, inspection methods, and task allocation approaches, which interact uniquely with the challenges of each scenario. These findings underline the importance of developing algorithms that adapt dynamically to different environments and achieve robust performance across various scenarios.

VII. Conclusion

This paper introduces the Cooperative Aerial Robot Inspection Challenge (CARIC), a benchmark and simulation framework designed to tackle the challenges of heterogeneous multi-UAV inspection in environments without prior structural models. CARIC provides a ready-to-use simulation environment and realistic scenarios to support the development and evaluation of innovative task allocation and motion planning algorithms for heterogeneous UAV teams.

The CARIC CDC 2023 competition showcased diverse strategies for this complex problem. By analyzing the top three teams' approaches, we highlighted their unique methodologies, trade-offs, and the challenges inherent in the task. The results demonstrated that no single approach has achieved robust performance across all scenarios, emphasizing the difficulty of designing universally effective strategies for cooperative aerial inspection.

Looking ahead, CARIC serves as a foundation for advancing research on cooperative multi-UAV systems. Planned enhancements include expanding the framework with more diverse testing scenarios and bringing it to the upcoming robotics conferences. We hope the insights and lessons this work presents will inspire continued innovation in this rapidly evolving field.

References

[1] A. Bircher, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart, "Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 6423–6430.

[2] C. Zhang, Y. Zou, F. Wang, E. del Rey Castillo, J. Dimyadi, and L. Chen, "Towards fully automated unmanned aerial vehicle-enabled bridge inspection: Where are we at?" *Construction and Building Materials*, vol. 347, p. 128543, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950061822022036

[3] A. Saha, L. Kumar, S. Sortee, and B. C. Dhara, "An autonomous aircraft inspection system using collaborative unmanned aerial vehicles," in *2023 IEEE Aerospace Conference*, 2023, pp. 1–10.

[4] S. S. Mansouri, C. Kanellakis, E. Fresk, D. Kominiak, and G. Nikolakopoulos, "Cooperative coverage path planning for visual inspection," *Control Engineering Practice*, vol. 74, pp. 118–131, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967066118300315

[5] X. Huang, Y. Liu, L. Huang, S. Stikbakke, and E. Onstein, "Bim-supported drone path planning for building exterior surface inspection," *Computers in Industry*, vol. 153, 12 2023.

[6] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon path planning for 3d exploration and surface inspection," *Autonomous Robots*, vol. 42, pp. 291–306, 2018.

[7] A. Zacharia, S. Papaioannou, P. Kolios, and C. Panayiotou, "Distributed control for 3d inspection using multi-uav systems," in *2023 31st Mediterranean Conference on Control and Automation, MED 2023*. Institute of Electrical and Electronics Engineers Inc., 2023, pp. 164–169.

[8] W. Jing, D. Deng, Y. Wu, and K. Shimada, "Multi-uav coverage path planning for the inspection of large and complex structures," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 1480–1486.

[9] G. Hardouin, J. Moras, F. Morbidi, J. Marzat, and E. M. Mouaddib, "A multirobot system for 3-d surface reconstruction with centralized and distributed architectures," *IEEE Transactions on Robotics*, vol. 39, pp. 2623–2638, 8 2023.

[10] X. Xiao, Z. Xu, G. Warnell, P. Stone, F. G. Guinjoan, R. T. Rodrigues, H. Bruyninckx, H. Mandala, G. Christmann, J. L. Blanco-Claraco, and S. S. Rai, "Autonomous ground navigation in highly constrained spaces: Lessons learned from the second barn challenge at icra 2023 [competitions]," *IEEE Robotics & Automation Magazine*, vol. 30, no. 4, pp. 91–97, 2023.

[11] S. Zhu, J. Zhou, A. Chen, M. Bai, J. Chen, and J. Xu, "Maexp: A generic platform for rl-based multi-agent exploration," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 5155–5161.

[12] Y. Xu, J. Yu, J. Tang, J. Qiu, J. Wang, Y. Shen, Y. Wang, and H. Yang, "Explore-bench: Data sets, metrics and evaluations for frontier-based and deep-reinforcement-learning-based autonomous exploration," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 6225–6231.

[13] R. Almadhoun, T. Taha, D. Gan, J. Dias, Y. Zweiri, and L. Seneviratne, "Coverage path planning with adaptive viewpoint sampling to construct 3d models of complex structures for the purpose of inspection," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 7047–7054.

[14] H. Baik and J. Valenzuela, "Unmanned aircraft system path planning for visually inspecting electric transmission towers," *Journal of Intelligent & Robotic Systems*, vol. 95, pp. 1097–1111, 2019.

[15] W. Wu, Y. Funabora, S. Doki, K. Doki, S. Yoshikawa, T. Mitsuda, and J. Xiang, "Evaluation and enhancement of resolution-aware coverage path planning method for surface inspection using unmanned aerial vehicles," *IEEE Access*, vol. 12, pp. 16 753–16 766, 2024.

[16] F. Wang, Y. Zou, X. Chen, C. Zhang, L. Hou, E. del Rey Castillo, and J. B. Lim, "Rapid in-flight image quality check for uav-enabled bridge inspection," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 212, pp. 230–250, 6 2024.

[17] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, "Rotors—a modular gazebo mav simulator framework," *Robot Operating System (ROS) The Complete Reference (Volume 1)*, pp. 595–625, 2016.

[18] A. Anastasiou, P. Kolios, C. Panayiotou, and K. Papadaki, "Swarm path planning for the deployment of drones in emergency response missions," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2020, pp. 456–465.

[19] B. Zhou, Y. Zhang, X. Chen, and S. Shen, "Fuel: Fast uav exploration using incremental frontier structure and hierarchical planning," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 779–786, 2021.

[20] Y. Luo, Z. Zhuang, N. Pan, C. Feng, S. Shen, F. Gao, H. Cheng, and B. Zhou, "Star-searcher: A complete and efficient aerial system for autonomous target search in complex unknown environments," *IEEE Robotics and Automation Letters*, vol. 9, no. 5, pp. 4329–4336, 2024.