Automation in Construction 177 (2025) 106360

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Flight path planning of UAV-driven refinement inspection for construction sites based on 3D reconstruction

Xin Liu [a,b], Wen Yi [c], Penglu Chen [a,b], Yi Tan [a,b,*]

[a] *Key Laboratory for Resilient Infrastructures of Coastal Cities, Shenzhen University, Ministry of Education, Shenzhen, China*
[b] *Sino-Australia Joint Research Center in BIM and Smart Construction, Shenzhen University, Shenzhen, China*
[c] *Department of Building and Real Estate, The Hong Kong Polytechnic University, Hong Kong, China*

ARTICLE INFO

Keywords:
3D reconstruction
BIM
Construction inspection
Inspection path planning
UAV

ABSTRACT

Construction site inspection usually requires high labor demands and remains inefficiency. Current UAV-based inspection mainly relies on manual operation with challenging low-altitude path in complex site environment. This paper proposes an automatic inspection flight path planning method based on BIM, UAVs, and 3D reconstruction for autonomous site inspection. First, the UAV performs rough flights to reconstruct a 3D model of the construction site, which, combined with BIM, enables environmental perception. Then, waypoint spaces are generated by mesh expansion, and an improved artificial potential field method combined with a greedy algorithm determines inspection waypoints and headings. Next, A* and simulated annealing algorithms are used to compute a globally optimal path. Finally, local coordinates are converted to real-world coordinates to generate UAV flight files and complete the autonomous inspection task. Experiments demonstrate enhanced safety and efficiency, with a 20.3 % improvement in inspection speed and reduced battery usage from 1.51 to 1.32 units.

1. Introduction

Safety management during construction is a crucial part of the construction industry, playing an important role in ensuring the safety of construction workers and the quality of projects. According to the data from the Ministry of Housing and Urban-Rural Development of China, there were 689 safety accidents in the housing and municipal engineering sector in 2020, resulting in 794 deaths [1]. According to statistics from the U.S. Federal OSHA, 1056 people died from safety accidents in the construction industry in 2022, an 11 % increase from the previous year [2]. Therefore, with the gradual reduction of the labor force in the construction industry and the high incidence of safety accidents, reducing the occurrence of safety incidents and guaranteeing labors' safety are necessary. The most efficient way to reduce safety accidents is through construction inspection. However, with the increasing complexity of construction projects and rising safety as well as quality requirements, traditional inspection can no longer adequately meet the demands of construction site management. Traditional inspections often rely on manual checks and on-site monitoring, which results in low inspection efficiency.

Currently, various information technologies have been applied to construction site inspection. Based on the types of non-destructive testing (NDT) devices, these technologies can be broadly categorized into vision-based sensors (such as RGB cameras [3,4] and thermal infrared sensors [5]) and LiDAR-based systems [6]. Among them, vision sensors are the most widely used due to their low cost and ease of deployment. In practical inspection systems, these sensing devices are often integrated with mobile platforms such as ground robots [7] and unmanned aerial vehicles (UAVs) [8], enabling efficient perception and data acquisition on construction sites. However, UAVs are primarily manually operated for macro-level inspection of overall progress or large structures on construction sites, being limited in aspects like safety hazards, material management, and small equipment management. Additionally, due to the complex environment and dense obstacles on construction sites, current UAVs typically operate at relatively high altitudes, making it difficult to capture detailed images of structural components. As a result, fine-grained inspection methods based on the deep integration of visual imaging and UAV platforms have become a focal point of recent research and represent a key direction for enhancing the intelligent management of construction sites.

However, existing UAV-based refined inspection methods for construction sites still face several significant challenges. First, the lack of effective acquisition and modeling of site environmental information leads to difficulties in obstacle avoidance and ensuring sufficient

[*] Corresponding author at: Key Laboratory for Resilient Infrastructures of Coastal Cities, Shenzhen University, Ministry of Education, Shenzhen, China.
E-mail address: tanyi@szu.edu.cn (Y. Tan).

https://doi.org/10.1016/j.autcon.2025.106360
Received 9 December 2024; Received in revised form 17 June 2025; Accepted 17 June 2025
Available online 23 June 2025

coverage during path planning. Second, current waypoint setting strategies are often suboptimal, resulting in redundant or unevenly distributed inspection points, which negatively affect flight efficiency and safety. Lastly, although UAV-based fine-grained inspection has been successfully applied in domains such as power line inspection [9], building facade inspection [10], mining [11], and agriculture [12], it remains difficult to adapt these methods to the highly dynamic, occluded, and spatially constrained environments typical of construction sites. Therefore, it is essential to develop an environment information modeling approach tailored to construction scenarios, combined with optimized waypoint selection and path planning strategies, in order to reduce redundant inspections, enhance waypoint safety, and improve the autonomy and effectiveness of UAV-based inspection in construction settings.

Based on the current issues and research challenges, this paper proposes a UAV-based refined inspection path planning method for construction sites, combining 3D reconstruction, BIM models, and UAV. The method includes several parts: (1) UAVs perform rough flights to gather environmental information and segment on-site inspection areas, (2) calculation of UAV inspection waypoints, and (3) planning and optimization of inspection paths. The proposed method aims to effectively address the complexity of construction site environments.

The rest of this paper is as follows: Section 2 is a review of related literature, Section 3 outlines the framework for the UAV-based refined inspection path planning method using 3D reconstruction, Section 4 selects a complex building to verify the feasibility of the proposed method, Section 5 discusses improvements and limitations of the proposed method, and Section 6 presents the conclusions of the paper. The main contributions of this paper are as follows: (1) a UAV inspection path planning method is proposed that integrates 3D reconstruction and BIM to improve path generation efficiency in complex construction scenarios; (2) an enhanced potential field function and greedy strategy are developed to enable efficient waypoint selection and optimized image coverage; (3) an energy-oriented path optimization mechanism is designed to reduce flight costs; (4) a construction site segmentation method is introduced that fuses BIM and 3D reconstruction data to enhance the automatic identification and modeling of inspection areas.

2. Related work

This section reviews related work from three main aspects: applications of UAV-based inspections, space models for inspection planning, and UAV path planning. Based on the review findings, the research gaps are further identified and summarized.

2.1. Applications of UAV-based inspections

Inspection tasks usually involve the entire lifecycle of various engineering projects, including preliminary surveys, construction, and operation & maintenance (O&M) stages. In recent years, with the development of UAV technology, its stability and flexibility have been continuously improved, and it has gradually been applied in various stages of different engineering projects. Many studies focus on the inspection process during the O&M stage. For example, in power facility inspections, Huang et al. [13] proposed a UAV-based method for detecting structural defects in dampers using UAV images. Nicolai et al. [14] proposed and validated an integrated UAV system for autonomous power line inspections, enabling UAVs to conduct inspections autonomously. In the field of infrastructure inspection, Zhao et al. [3] implemented health monitoring and damage detection of dam structures using UAV photogrammetry and image-based 3D reconstruction technology. Seo et al. [15] proposed a UAV-based bridge damage inspection method for identifying bridge defects. Furthermore, they also developed a five-phase UAV bridge inspection method [16]. Xiao et al. [17] proposed a bridge crack detection method based on UAV point cloud semantic segmentation and 3D-to-2D projection, which reduces the impact of background interference on the monitoring results. Miquel et al. [18] proposed a UAV-based method for inspecting communication route bridges. Considering the potential quality issues in bridge inspection photos, Wang et al. [19] proposed a UAV image quality evaluation method during flight to assess whether photos meet required standards. Regarding photo quality issues, Jeong et al. [20] proposed a machine learning-based image acquisition protocol using convolutional neural networks and UAVs. Zhang et al. [21] and Adibfar et al. [22] summarized existing literature and found that UAVs still require significant manual intervention for bridge inspections. Tan et al. [8] proposed an automatic building surface inspection method combining UAVs with BIM to achieve accurate and high-quality building inspections. Through optimization of inspection strategies, they also introduced a “global-local” adaptive detection method [23], which greatly improves inspection efficiency. Tan et al. developed a method for quantifying cracks [24] and then mapped UAV images onto the Building Information Model (BIM) [25], enhancing the application potential of UAV-based building inspection. Zhang et al. [26] further optimized the aforementioned image-to-BIM mapping process and integrated it into the inspection workflow, enhancing the overall automation of UAV-based inspection. The application scenarios for UAVs in O&M inspections of various engineering projects are extensive. In the construction phase, Gheisari et al. [27] proposed that UAV systems could have a positive impact on managing construction sites. Liang et al. [4] developed a multi-class object detection system for construction sites based on low-altitude UAV remote sensing, proving its robustness. Yu et al. [28] developed a UAV flight path planning system for known site environments. Jhonattan et al. [29] discussed the new steps required to apply UAVs to safety planning and inspections on construction sites and identified that UAV inspections improved hazard identification and assessment on high-rise projects. Wu et al. [30] developed a rapid safety monitoring and analysis method for excavation construction based on UAVs, proposing the safety slope angle of excavation sidewalls as a safety monitoring indicator. Rafaela et al. [31] proposed a construction site safety inspection system using UAS as a data collection tool, providing timely on-site information for decision-making. David et al. [32] summarized existing literature and highlighted the various challenges encountered when using UAVs for construction site inspections.

A summary of existing literature shows that UAVs hold significant potential and advantages for construction site inspections. However, there are still shortcomings in their application for construction inspections. For instance, the coverage of UAV inspection scenarios is limited, UAV maneuverability during inspections is poor, and most UAV flight operations are manually controlled by on-site personnel. Therefore, it is necessary to diversify UAV inspection scenarios and enhance the safety of UAV flights during inspections.

2.2. Space models for inspection planning

The environmental model used by UAVs affects the calculation of inspection waypoints and flight path planning. Common environmental models in inspection planning include BIM, point cloud, and 3D reconstruction model (mesh model). Many researchers have conducted a series of studies based on these environmental models. In the case of BIM models, Tan et al. [8] extracted the required inspection areas from the BIM model of the target building and used BIM as the environmental model for path planning, enabling the collection of image data from building facades. Huang et al. [33] converted the BIM model into a point cloud model and achieved inspection waypoint calculation. Hesam et al. [34], considering the complexity of indoor environments, used 4D BIM to plan the optimal indoor data collection schedule for UAVs. Amir et al. [35] also used the prior availability of 4D BIM to calculate inspection points for monitoring construction progress. BIM is primarily used for inspecting existing buildings, while for dynamic environments such as construction sites, a 4D BIM model with time-dependent information is used.

Regarding point cloud model-based inspection planning, Sun et al. [36] used a rough scan plane to obtain a point cloud model and computed the required inspection waypoints and paths based on the point cloud model. Antonio et al. [37] used LiDAR scanning to obtain structural point cloud data and provided a solution for UAV inspection positions and trajectory calculations for bridge engineering. Iago et al. [38] developed a metaheuristic algorithm based on point cloud data to inspect slopes and dam structures, validating its effectiveness in both virtual and real environments. Li et al. [39] proposed a UAV autonomous inspection system based on laser point cloud precise positioning, effectively ensuring flight safety and reducing potential risks in the power grid. An et al. [40] constructed a GDT-VOI based on 3D point clouds and achieved accurate obstacle modeling through voxel-based spatial partitioning. Liu et al. [41] achieved path planning for wheeled or tracked robots in a 2.5D environment by using point cloud data input. This approach has significant advantages in terms of smooth robot movement, especially when considering the minimum path length. Yan et al. [42] used 3D point clouds to build sparse semantic maps for autonomous vehicles, enabling fast fuzzy positioning with minimal environmental data, achieving very small storage requirements even in large-scale environments.

In terms of 3D reconstruction models, Koch et al. [43] proposed a UAV flight path planning framework based on image-based 3D reconstruction, used for refined 3D reconstruction path planning while considering environmental semantic attributes to meet high-resolution 3D reconstruction path planning requirements. Zhang et al. [44] used a coarse 3D reconstruction model as a reference model for multi-rotor UAV path planning, achieving accurate 3D reconstruction of complex buildings. Zhang et al. [45] proposed a path planning algorithm combining 3D reconstruction monolithic models with Geographic Information System (GIS) systems for power grid inspections, enabling effective path planning for power grid inspections.

When selecting an environmental model, BIM or 4D BIM is still a common choice in the construction process, as it reflects the planned progress of the construction site. However, it is not perfect in expressing the complex environmental information of the construction site. For point cloud models, they require normal direction estimation, and there is a potential issue with distinguishing between the interior and exterior of a structure. Currently, 3D reconstruction as an environmental model can better express environmental information, but its application remains limited.

2.3. UAV path planning

UAV path planning has been well studied with the goal of finding the optimal flight path for UAVs. All path planning algorithms can be categorized into three types: exact algorithms, heuristic algorithms, and metaheuristic algorithms [46]. Exact algorithms includes Voronoi diagrams [47] and linear programming algorithms [48]. Priyadarshi et al. [49] proposed a method for finding collision-free paths from the starting point to the target point based on Voronoi diagrams, which generally meets practical needs. Chen et al. [50] combined Voronoi diagrams with the Dijkstra algorithm and optimized it to achieve path planning in dynamic environments. Exact algorithms have high computational requirements and often need a significant amount of computation time. Additionally, as the number of targets increases, their applicability decreases significantly.

Heuristic algorithms are also commonly used in path planning, including greedy algorithms [51], A* algorithms [52], and divide-and-conquer algorithms [53]. Christian et al. [54] found that the A* algorithm has a significant advantage over the Rapidly-exploring Random Tree (RRT) algorithm in generating shorter paths, but may experience oscillation in different resolution spaces. They improved the A* algorithm to reduce this oscillation. Song et al. [55] developed a 3D A* path generation method that reduces jaggedness, results in smoother paths, and eliminates redundant waypoints, with better efficiency than traditional algorithms. Afshin et al. [56] implemented a shortest path from the start to the target in 2D space based on the A* algorithm, considering energy consumption and path turning angles.

In terms of metaheuristic algorithms, commonly used ones include genetic algorithms, simulated annealing, particle swarm optimization, and ant colony optimization (ACO). Roberge et al. [57] implemented UAV path calculation based on genetic algorithms on GPUs, significantly reducing computation time while finding better flight paths in a timely manner. Pehlivanoglu et al. [58] used genetic algorithms to generate collision-free UAV paths, enhancing the generation of the initial population to accelerate the convergence process. Shafiq et al. [59] used ACO to achieve stable and fast UAV path planning. Zhang et al. [60] used RRT for UAV path planning, but encountered issues with low efficiency and path smoothness. Diao et al. [61] used RRT to calculate the flight path planning capability of UAVs for disaster relief in hazardous environments, but identified problems with slow convergence, low efficiency, and uneven path points. Currently, using a single algorithm for UAV path planning has many shortcomings. As a result, many researchers combine various algorithms. Some scholars combine heuristic algorithms with metaheuristic algorithms [23,54], or metaheuristic algorithms with other metaheuristic algorithms [58,62], to improve performance. As proposed by An et al. [40], a hybrid algorithm combining GA and A* was developed, in which voxelized obstacle information was used to constrain the path search space, thereby ensuring a safe distance while reducing unnecessary turns. Some researchers have also adopted more advanced algorithms for path planning. For example, Ivic et al. [63] proposed a multi-UAV trajectory planning method based on the Heat Equation Driven Area Coverage (HEDAC) algorithm, which generates a 3D potential field to guide UAV inspection path planning.

Although current path planning algorithms are well-developed and generally capable of finding complete paths, most studies have not taken into account the differences in energy consumption during actual UAV flights. The paths generated are typically the shortest in terms of physical distance and remain at the theoretical level, not fully applicable to real-world scenarios. In practical tests, UAVs experience significant variations in energy consumption during horizontal flight, ascent, and descent [64]. As a result, the shortest path may not be the optimal path in terms of energy consumption. For UAV inspections, the path with the lowest energy consumption is more critical. Therefore, optimizing for energy consumption may be a more meaningful approach.

3. Methodology

In order to enhance the precision, efficiency, and automation of UAV inspections at construction sites, this paper proposes a refined UAV inspection flight path planning method based on 3D reconstruction. Fig. 1 illustrates the development framework of the proposed method, which is composed of four parts: (1) Construction site environment perception, (2) Inspection waypoint calculation, (3) Inspection path planning and optimization, and (4) Inspection parameter calculation.

3.1. Construction site environment perception

Construction site environment perception primarily consists of two aspects: 3D reconstruction of the site environment using UAV images and the segmentation of inspection areas. The following subsection first introduces the method for generating the 3D environmental model of the construction site.

3.1.1. UAV images-based 3D reconstruction of construction sites

The primary requirement for UAV inspection at construction sites is acquiring accurate environmental information. In this paper, DJI Terra, a commercial 3D reconstruction software developed by DJI, is used to process UAV-captured images. Although DJI Terra is closed-source, its internal workflow adheres to the widely accepted structure-from-motion (SfM) and multi-view stereo (MVS) principles. To improve transparency

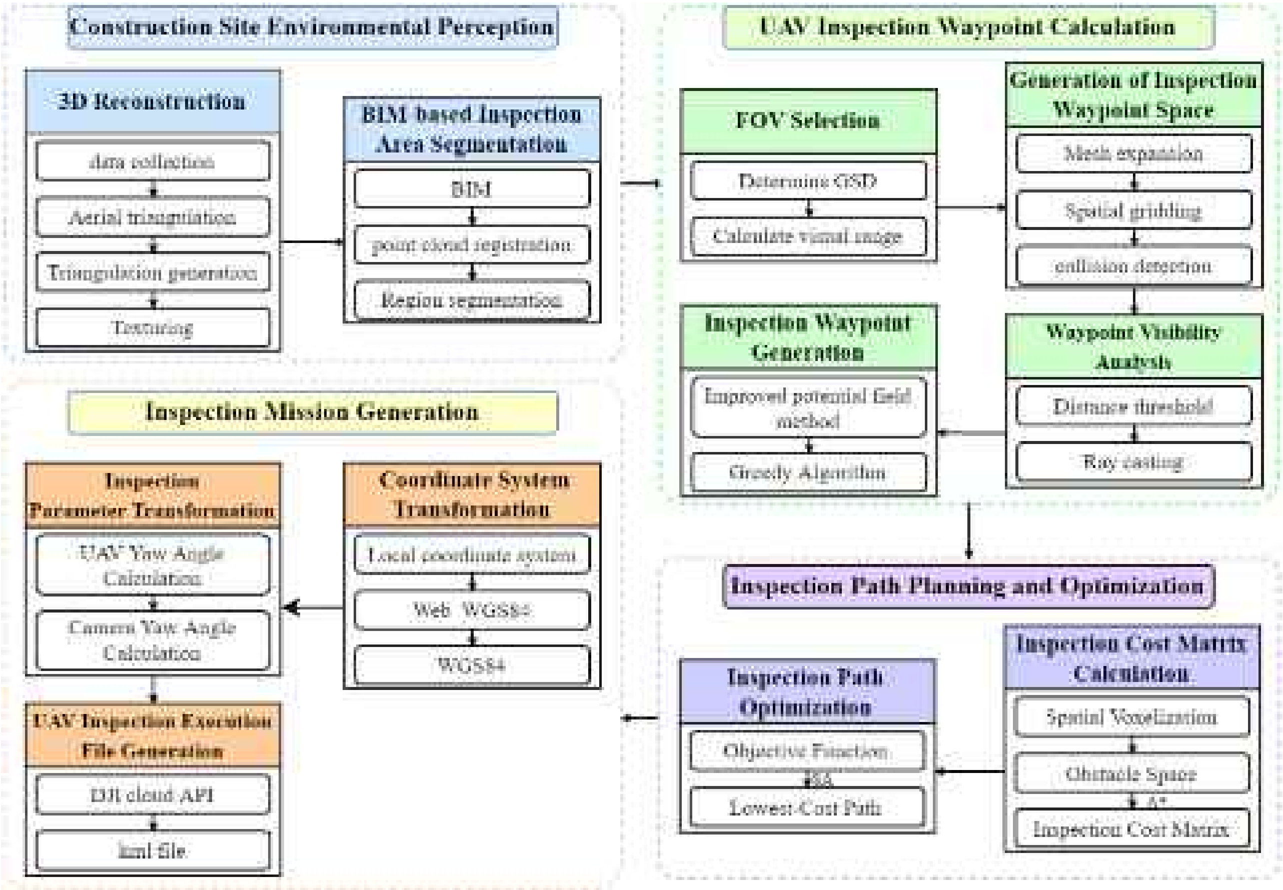

Construction Site Environmental Perception
3D Reconstruction
data collection
Aerial triangulation
Triangulation generation
Texturing
BIM based Inspection Area Segmentation
BIM
point cloud registration
Region segmentation
UAV Inspection Waypoint Calculation
FOV Selection
Determine GSD
Calculate visual image
Generation of Inspection Waypoint Space
Mesh expansion
Spatial gridding
collision detection
Inspection Waypoint Generation
Improved potential field method
Greedy Algorithm
Waypoint Visibility Analysis
Distance threshold
Ray casting
Inspection Mission Generation
Inspection Parameter Transformation
UAV Yaw Angle Calculation
Camera Yaw Angle Calculation
Coordinate System Transformation
Local coordinate system
Web WGS84
WGS84
UAV Inspection Execution File Generation
DJI cloud API
kml file
Inspection Path Planning and Optimization
Inspection Path Optimization
Objective Function
ISA
Lowest-Cost Path
Inspection Cost Matrix Calculation
Spatial Voxelization
Obstacle Space
A*
Inspection Cost Matrix

Fig. 1. Research framework.

and reproducibility, the typical theoretical pipeline is briefly introduced below to contextualize the process.

The first step in 3D reconstruction is image acquisition. The 3D model required for this paper serves primarily for inspection planning and geometric analysis, so high-resolution textures are not prioritized. Therefore, a relatively high UAV flight altitude is selected to capture broad site coverage. Feature point extraction and matching are performed to align images. The Scale-Invariant Feature Transform (SIFT) algorithm [65] is used to extract keypoints, which are matched across image pairs using a nearest-neighbor strategy. Lowe's ratio test filters out ambiguous matches. To enhance geometric consistency, the RANSAC algorithm is applied to estimate the fundamental matrix F, eliminating incorrect matches that violate epipolar constraints. The combined use of KNN-based matching and RANSAC-based filtering is a standard feature correspondence strategy, supporting high-quality input for subsequent pose estimation and point cloud reconstruction. After obtaining reliable matches, camera calibration is carried out to estimate the relative position and orientation of each image. This step includes the computation of the This process primarily involves the calculation of the fundamental matrix F and the essential matrix E, as shown in Eq. (1) and Eq. (2):

$$\mathrm{E}=\mathrm{K}_2^{\mathrm{T}}\mathrm{F}\mathrm{K}_1. \tag{1}$$

$$x_2^T E x_1 = 0 \tag{2}$$

where K_1 and K_2 represent the camera's intrinsic matrices, and x_1 and x_2 are the corresponding coordinates of the same feature in two images. The essential matrix can further be decomposed into the relative rotation matrix R and translation vector t, as shown in Eq. (3):

$$E = [t]\times R \tag{3}$$

Where [t] is the skew-symmetric matrix generated by the displacement vector t. By decomposing the essential matrix, the pose R_i and t_i of each image can be calculated, which in turn allows the determination of the projection matrix P_i for each image. To minimize the projection error, bundle adjustment [65] is used to calculate the distance between the projected points and the observed points, as shown in Eq. (4):

$$E\left(\{R_i,T_i\}_{i=1..m},\{X_j\}_{j=1..N}\right)=\sum_{i=1}^{m}\sum_{j=1}^{N}\theta_{ij}\left\|x'_{ij}-\pi\left(R_i,T_i,X_j\right)\right\|^2 \tag{4}$$

Where $\{R_i,T_i\}_{i=1...m}$ represents the camera extrinsic parameters of all images, with R_i being the rotation matrix of the *i-th* image and T_i the corresponding displacement vector. $\{X_j\}_{j=1..N}$is the set of all 3D points, with X_j being the coordinates of the *j-th* 3D point. x'_{ij} is the image coordinates of the *j-th* 3D point in the i image, and $\pi(R_i,T_i,X_j)$ is the 2D coordinates obtained by projecting the 3D point *Xj* onto the *i-th* image using the camera projection model. θ_{ij} indicates whether the *j-th* 3D point is observed in the *i-th* image. It is 1 if observed, and 0 if not observed.

By continuously calculating feature points, a sparse point cloud is generated. Fig. 2(a) shows the sparse point cloud obtained through the above process. Based on this sparse point cloud, the Semi-Global Matching (SGM) algorithm can be used to generate a dense 3D point cloud. Finally, using Poisson surface reconstruction, a 3D mesh is reconstructed from the dense point cloud. Texture is then mapped onto the surface of the 3D model to create the digital environment model of the construction site. Fig. 2(b) shows the final 3D reconstruction model of the construction site, as produced by DJI Terra.

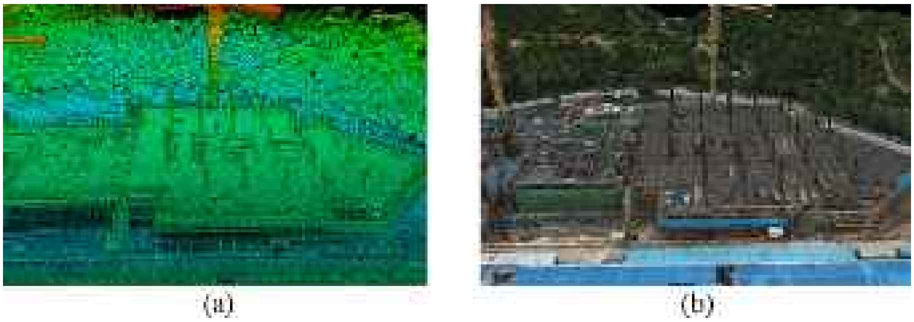


Fig. 2. 3D Reconstruction Results (a) Sparse Point Cloud (b) 3D Reconstruction Model.

3.1.2. BIM-based inspection area segmentation

To ensure that UAVs focus only on relevant parts of the site, segmentation of the 3D reconstruction model is required to isolate inspection areas. As shown in Fig. 3, this paper proposes a BIM-based inspection area segmentation workflow consisting of two main stages: (1) registering the BIM model with the 3D reconstruction model; and (2) segmenting the inspection areas using the bounding boxes derived from BIM components.

Since the data formats of the BIM model and the 3D reconstruction model are different, making direct registration operations difficult, both the BIM model and the 3D reconstruction model are converted into two sets of point cloud data. These point clouds are then down-sampled using voxel-based down-sampling to ensure consistent point cloud density. The registration process of the two sets of point clouds mainly includes two steps: coarse registration and fine registration. Since the initial two sets of data have large translations and rotations due to differences in coordinate systems, coarse registration is first used to roughly align the point clouds and ensure that they are within the approximate range of the same coordinate system, serving as the foundation for fine registration. Fine registration focuses on accurately aligning local details in the point clouds to ensure precise matching of all parts. To accelerate the registration efficiency, the sparse-dense fusion registration strategy proposed by Tan et al. [66] is used. Coarse registration uses the K4PCS algorithm, which is robust to noise and local variations. This algorithm primarily aligns two point cloud models by exploiting the collinearity relationship between four points (four-point set) in the point cloud. Afterward, the ICP (Iterative Closest Point) method is used for fine registration. To ensure the effectiveness of point cloud registration, Root Mean Square Error (RMSE) and overlap rate are used as key evaluation metrics. RMSE quantifies the average alignment error between the point clouds after registration, and its calculation is shown in Eq. (5):

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\|p_i - q_i\|^2} \tag{5}$$

where p_i and q_i represent the corresponding points in point clouds P and Q, respectively, and N is the total number of corresponding point pairs. The overlap rate measures the spatial overlap between the two point cloud model after registration, and its calculation is shown in Eq. (6):

$$OP = \frac{N_{overlap}}{N_{total}} \times 100\% \tag{6}$$

where $N_{overlap}$ represents the number of corresponding point pairs in the registered point clouds whose distance is less than a set threshold, and N_{total} is the total number of points in the point clouds. By combining these two metrics, this paper can systematically assess and verify the accuracy and effectiveness of point cloud registration.

Based on the registration results of the BIM model and the 3D reconstruction, the construction site is segmented. Since most buildings in construction are relatively regular in shape, this paper directly uses the oriented bounding boxes (OBB) of individual BIM model components as the bounding boxes for segmentation. However, due to potential registration errors, direct projection of OBBs may lead to segmentation failure, such as missing or partially overlapping target regions. To address this, we tested multiple real-world construction sites and found that expanding the bounding boxes with a scaling factor ϑ can effectively compensate for such deviations and ensure robust coverage. Depending on the project scale and point cloud accuracy, a suitable value of ϑ is selected to balance coverage and precision. These two steps allow for reliable segmentation of the inspection area, providing the UAV with accurate target information for subsequent inspection tasks.

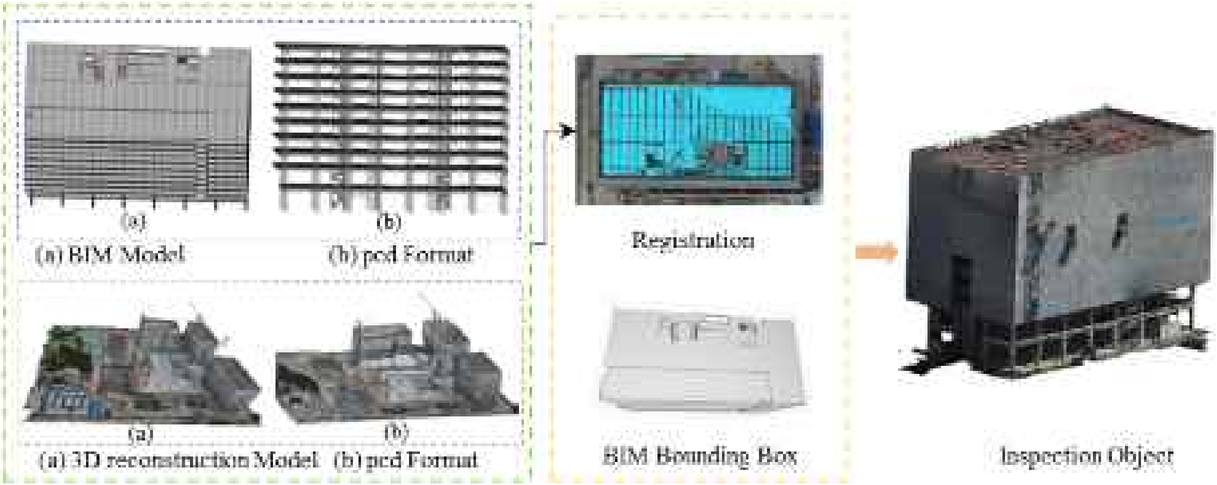


Fig. 3. Workflow of BIM-based inspection area segmentation.

3.2. UAV inspection waypoint calculation

This subsection primarily includes the determination of the UAV's field of view (FOV), the generation of inspection candidate spaces, visibility analysis of candidate waypoints, and the final selection of inspection points.

3.2.1. FOV selection

After obtaining segmented on-site inspection area, fine grained inspection can then be planned using UAVs. The Field of View (FOV) significantly impacts the number of inspection waypoints and the ground resolution. An appropriate FOV configuration enables UAVs to obtain clear and detailed visual information about the construction site while reducing the number of waypoints required for complete coverage. Considering that UAVs are typically equipped with gimbals that allow for flexible rotation during flight, this paper proposes a gimbal-based dynamic FOV model. Unlike traditional fixed-angle FOV models, the proposed model utilizes the gimbal's pitch angle θ and yaw angle ψ to enable multi-angle image acquisition at a single inspection waypoint. During flight, the UAV hovers at each selected waypoint while the gimbal rotates within a predefined angular range at regular intervals in either the pitch or yaw direction. An image is captured at each step, allowing a single waypoint to achieve extended viewing coverage. This strategy significantly reduces the need for redundant waypoints that would otherwise be required due to limited fixed-view coverage. The number of images required at each waypoint can be calculated based on the sweep range and angular step size, while the ground sampling resolution of each image can be estimated using Eq. (7):

$$\mathrm{GSD} = \alpha \bullet \frac{\mathrm{H}}{\mathrm{f}} \tag{7}$$

where Ground Sampling Distance (GSD) represents the resolution of the captured image, α is the ground resolution, *f* is the camera's focal length in pixels, and H is the distance from the UAV to the object being photographed. The GSD is dynamically considered based on the actual requirements, the UAV's safe flight distance, and the environmental conditions. Fig. 4 illustrates the relationship between the UAV's FOV and the focal length *f*.

3.2.2. Generation of inspection waypoint space

The goal of this step is to generate a potential UAV inspection point space that can cover the entire object to be inspected. The inspection object model is saved in the obj format, and thus, the generation method of the inspection space involves expanding outward along the vertex normal vectors. First, the normal vectors of all the faces at the vertices are calculated, then the normal vectors of all the faces are weighted and summed to compute the outward expansion direction. The specific calculation of the vertex is shown in Eq. (8):

$$\mathrm{n_i} = \sum_{\mathrm{f} \in \mathrm{F_i}} \mathrm{A_j} \bullet \mathrm{n_j} \tag{8}$$

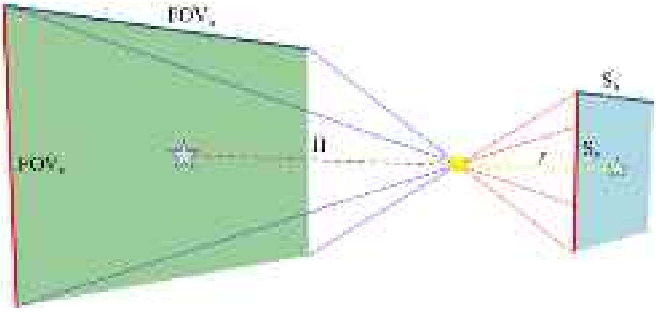


Fig. 4. Relationship between UAV FOV and focal length.

where F_i is the set of all faces containing vertex v_i, A_j is the area of face j, and n_j is the unit normal vector of face j. Finally, all the vertex normal vectors are normalized to become unit vectors.

After calculating the normal vector, the vertex is moved along the direction of the normal vector by a given distance d to achieve expansion, as shown in Eq. (9).

$$V_i^{'} = V_i + d \bullet n_i \tag{9}$$

where V_i is the original position of vertex p_i, d is the distance moved along the normal, and n_i is the unit normal vector of vertex p_i. The minimum expansion distance of the vertex is the UAV's minimum safety distance, and the maximum expansion distance is the distance corresponding to the maximum ground resolution required by the inspection.

A closed inspection space is obtained based on the two expanded faces. Specifically, the boundary edges on the original surface and the expanded surface are identified (i.e., the edges that belong to only one face). For each boundary edge $\mathrm{E_{min\text{-}i}}$ on the minimum expansion surface, there is a corresponding edge $\mathrm{E_{max\text{-}i}}$ on the maximum expansion surface. For each matched pair of edges E1 and E2, a quadrilateral face is created. This process is repeated until a closed geometric body is formed, which is the initial inspection waypoint space, as illustrated in Fig. 5.

Since the entire waypoint space is continuous, in order to prevent the infinite search for waypoints, a voxel down-sampling method is used to discretize the entire initial inspection waypoint space. The size of each voxel is determined based on the expanded distance of the UAV's volume. After discretizing the continuous space, all voxel centers are taken as the candidate waypoint set M. An example of discretized waypoints in space is illustrated in Fig. 6.

Due to the complexity of the construction site, some of the waypoints in the expanded space, generated by extending the vertices along their normal vectors, may intersect with the original model or fail to meet the safety distance requirements. Therefore, these waypoints need to be removed from the waypoint set M to obtain the final candidate waypoint space $\mathrm{M_c}$. First, for the waypoints that may be inside the original model, rays are emitted vertically upward from all candidate waypoints for intersection detection with the original model, as shown in Fig. 7. If an intersection is detected, the waypoint is marked as an internal waypoint and removed from the candidate waypoint set.

For points with insufficient safety distance, a sphere with a radius equal to the minimum safety distance $\mathrm{h_s}$ is created around each waypoint. A Boolean operation is performed between all spheres and the original model. If an intersection occurs, the waypoint is considered to violate the safety requirements and is removed from the candidate

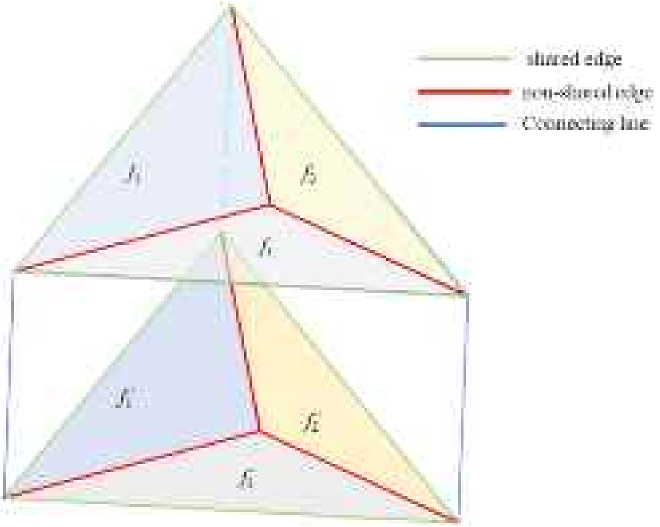


Fig. 5. Waypoint space generation diagram.

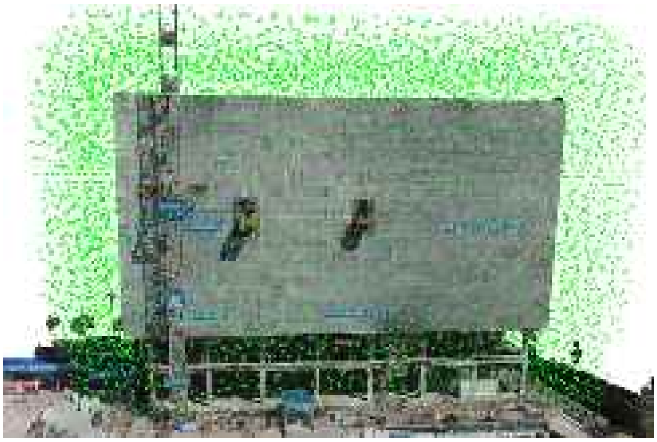


Fig. 6. Discretized waypoints in space.

waypoint set. After performing these two steps to detect the waypoints, the final waypoint space M_c is obtained.

3.2.3. Waypoint visibility analysis

When performing UAV inspections, it is essential to select the most suitable waypoints. The first step is to determine the visible faces of the waypoints. The inspected object is represented as a mesh format, so ray-casting is used to determine visibility. Before performing ray-casting, the distance between the faces and the waypoints must be evaluated to exclude faces that are outside the maximum visibility range, thus reducing the number of faces that need to be tested for visibility. Rays are then emitted from each waypoint towards the center of every face. If the first face hit by the ray is the initial endpoint of the ray, the face is considered visible. Any other scenario indicates that the face is not visible. The principle of ray-casting is shown in Fig. 8.

3.2.4. Inspection waypoint generation

Based on the waypoint space M_c generated in Section 3.2.2, a finite number of waypoints are selected to cover as much of the inspected object's surface as possible. This results in a set of waypoints that can cover the inspected object with the fewest waypoints and the shortest inspection path. However, achieving this while simultaneously optimizing both waypoint selection and inspection path is an NP-hard problem. To reduce complexity and improve efficiency, this paper separates the waypoint generation from the inspection path search.

(1) Calculation of Inspection Waypoint Orientation

For the calculation of inspection waypoints, the first step is to compute the yaw angle of the UAV when it reaches each waypoint. The UAV's yaw angle is illustrated in Fig. 9. Based on this yaw angle, the UAV's gimbal rotation is performed to complete the inspection task. This paper uses a potential field method to compute the yaw angle for each inspection waypoint. Considering the complexity of the construction site environment, where the size of the surface patches in the 3D reconstruction model varies significantly, an area-based potential field method is proposed. The potential field is composed of all the triangular surface patches. The strength of the potential field is governed by Eq. (10):

$$\Phi_i = \begin{cases} k \bullet \mathrm{Area}_i, & \text{if } d_i \leq d_{\max} \\ 0, & \text{if } d_i > d_{\max} \end{cases} \tag{10}$$

where Φ_i is the potential field strength generated by the *i-th* surface patch at the UAV's inspection location, k is a constant, $Area_i$ is the area of the *i-th* surface patch, d_i is the distance between the UAV and the *i-th* surface patch, and d_{max} is the maximum distance at which the patch can generate a potential field.

The potential field strength Φ within the camera's visible range is set to a fixed value, and when the surface patch is beyond the visible distance, the potential field strength becomes 0. The attraction force of patch i to the waypoint is F_i, and based on the defined potential field strength, the calculation of the potential field force F_i for patch i is given by Eq. (11):

$$F_i = \begin{cases} k \bullet \mathrm{Area}_i \bullet \dfrac{\mathbf{r}_{\mathrm{mav}} - \mathbf{r}_i}{\|\mathbf{r}_{\mathrm{uav}} - \mathbf{r}_i\|}, & \text{if } d_i \leq d_{\max} \\ 0, & \text{if } d_i > d_{\max} \end{cases} \tag{11}$$

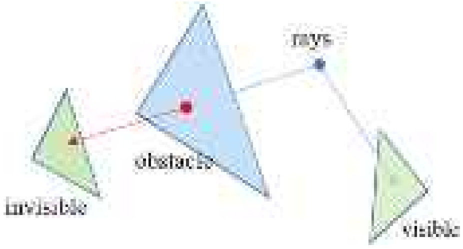


Fig. 8. Surface visibility judgment.

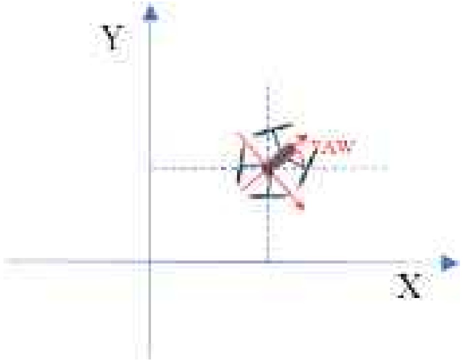


Fig. 9. UAV yaw angle.

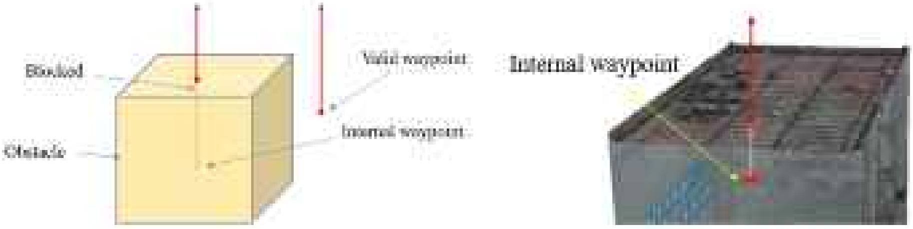


Fig. 7. Principle of waypoint inclusion and exclusion judgment.

where $\mathbf{r}_{\mathrm{uav}}$ is the position of the UAV, $\mathbf{r}_i$ is the position of the surface patch, and the vector $\frac{\mathbf{r}_{\mathrm{mav}}-\mathbf{r}_i}{\|\mathbf{r}_{\mathrm{uav}}-\mathbf{r}_i\|}$ represents the direction vector from the center of the surface patch to the UAV.

The yaw angle direction of the UAV is determined by the resultant force $\mathrm{F}_{\mathrm{total}}$ generated by all the surface patches, and the calculation of the resultant force is shown in Eq. (12):

$$\mathrm{F}_{\mathrm{total}} = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{F}_{\mathrm{i}} \tag{12}$$

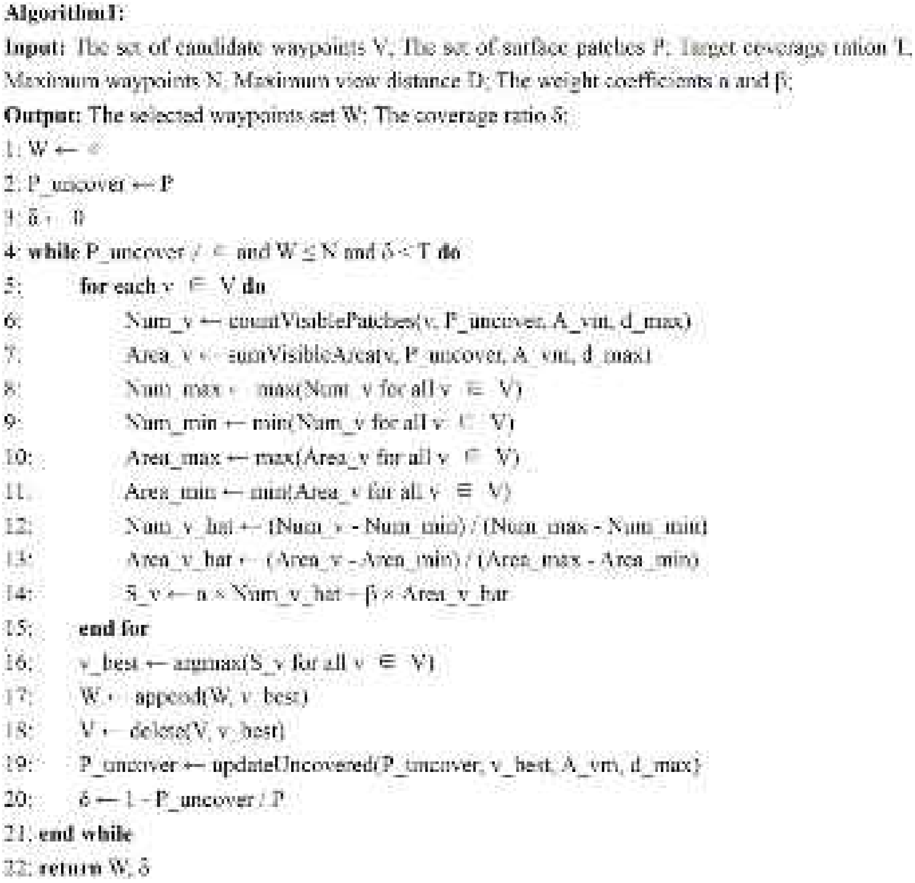
the direction of the resultant force $\mathrm{F}_{\mathrm{total}}$ is the target heading of the UAV at a given inspection waypoint. The yaw angle of the UAV can be calculated based on the resultant force vector.

(2) Waypoint Calculation Based on Greedy Strategy

After determining the UAV's orientation for the inspection, a greedy strategy is used to select the required inspection waypoints. The greedy strategy in this paper is based on a combined score S, which takes into account both the number of visible surface patches and the area of these patches at each potential inspection waypoint. The specific calculation formula is shown in Eq. (13):

$$S_i = \alpha \bullet \mathrm{Num}_i + \beta \bullet \left(\frac{\sum\limits_{j \in \mathscr{M}_i} \mathrm{Area}_j}{\mathrm{Num}_i} \right) \tag{13}$$

where S_i is the dimensionless combined score for the candidate inspection waypoint P_i, $\mathscr{M}_i$ is the set of surface patches visible from the candidate inspection waypoint P_i, and α and β are weight coefficients used to adjust the importance of the number of patches and the area, respectively. Where the score combines the number of surface patches visible from each candidate inspection waypoint and the area of those patches. To ensure scale consistency between the number of patches and the total area, both metrics are normalized before calculating the combined score. In each iteration, the inspection waypoint with the highest combined score, aligned with the direction of the resultant force, is selected as the target inspection waypoint. The specific algorithm flow is shown in Fig. 10.



```
Algorithm1:
Input: The set of candidate waypoints V; The set of surface patches P; Target coverage ration T;
Maximum waypoints N; Maximum view distance D; The weight coefficients α and β;
Output: The selected waypoints set W; The coverage ratio δ;
W ← ∅
P_uncover ← P
δ ← 0
while P_uncover ≠ ∅ and W ≤ N and δ < T do
      for each v ∈ V do
            Num_v ← countVisiblePatches(v, P_uncover, A_vm, d_max)
            Area_v ← sumVisibleArea(v, P_uncover, A_vm, d_max)
            Num_max ← max(Num_v for all v ∈ V)
            Num_min ← min(Num_v for all v ∈ V)
            Area_max ← max(Area_v for all v ∈ V)
            Area_min ← min(Area_v for all v ∈ V)
            Num_v_hat ← (Num_v - Num_min) / (Num_max - Num_min)
            Area_v_hat ← (Area_v - Area_min) / (Area_max - Area_min)
            S_v ← α × Num_v_hat + β × Area_v_hat
      end for
      v_best ← argmax(S_v for all v ∈ V)
      W ← append(W, v_best)
      V ← delete(V, v_best)
      P_uncover ← updateUncovered(P_uncover, v_best, A_vm, d_max)
      δ ← 1 - P_uncover / P
end while
return W, δ
```

Fig. 10. Iterative process of UAV inspection waypoints.

3.3. Inspection path planning and optimization

The inspection path refers to the flight path planned for the UAV to cover all inspection waypoints. If the inspection waypoints are visited in the order initially calculated, it may significantly increase the length of the UAV's inspection path, leading to higher energy consumption and time cost. Therefore, it is necessary to plan and optimize the UAV's inspection path. This section will be unfolded into three subsections, including UAV inspection waypoint path matrix calculation, UAV shortest path planning, and inspection path optimization.

3.3.1. Inspection cost matrix calculation

(1) Voxelization of the Environment Space

Based on the maximum and minimum 3D coordinates of the inspected object and its surrounding environment, a 3D bounding box is established. The space within this bounding box is then voxelized, as shown in Fig. 11. Different inspection requirements and environmental conditions necessitate different voxel sizes. The voxel size is determined by considering factors such as the UAV's dimensions, path calculation accuracy, and computational efficiency. To avoid unsafe proximity between the UAV and obstacles during path planning, the voxel size is generally set larger than the UAV's size. A voxel smaller than or equal to the UAV's size may result in feasible paths being planned too close to obstacles, neglecting the necessary safety buffer required for the UAV's body and propellers. Therefore, using a larger voxel size effectively introduces an implicit safety margin within the voxelized environment, ensuring that the UAV maintains sufficient clearance from obstacles during inspection tasks. The 3D grid is then divided into obstacles and non-obstacles. The division is based on the densest point cloud obtained from the region partitioning described in Section 3.1. All point clouds are checked, and if any point cloud exists within a voxel, that voxel is considered an obstacle.

In actual UAV flight, the UAV cannot fly too close to the inspected object or surrounding environment, so an expansion of the inspected target and the environment is required. This ensures that the UAV maintains a safe distance during flight, as shown in Fig. 12. Through multiple on-site construction tests, setting a minimum safety distance of 2 m has been determined to be a reasonable value. The expanded surface and the original inspection surface will have all the grid cells within their spatial range classified as obstacles. After these two steps, the remaining environment space is considered as the safe, flyable area.

(2) Inspection Path Generation

Due to the complexity of the actual construction site environment, there are many obstacles, making it impossible to directly connect inspection waypoints with a straight line. This paper adopts a two-stage method to generate the inspection path. First based on the voxelized space, each pair of inspection waypoints is connected using a straight line, and the feasibility of this direct path is evaluated. If the line does not pass through any obstacle voxels, it is directly used as the inspection path. If the line does pass through obstacle voxels, the A* algorithm is applied to find an alternative path. The A* algorithm is a heuristic search algorithm that finds the shortest path from the start point to the endpoint while minimizing the number of search nodes required. The path cost obtained by the A* algorithm can be calculated using Eq. (14):

$$\mathrm{f(x)} = \mathrm{g(x)} + \mathrm{h(x)} \tag{14}$$

$$\mathrm{g(x)} = \mathrm{g(p(x)\,)} + \mathrm{cost(p(x)\,,x)} \tag{15}$$

where *f(x)* denotes the estimated total cost from node *x* to the goal, *g(x)* represents the accumulated distance from the start point to node *x*, and *h(x)* is the heuristic estimate from node *x* to the target node. In Eq. (15), *p(x)* refers to the parent node of *x* in the current search tree. This recursive formulation allows the algorithm to iteratively accumulate cost along the path from the starting node to the current node. In this paper, the Euclidean distance is used for heuristic estimation, as shown in Eq. (16):

$$h(x) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{16}$$

where (x_1,y_1,z_1) are the spatial coordinates of the current node *x*, and (x_2,y_2,z_2) represent the fixed coordinates of the goal node. To fully account for the UAV's flexibility, the A* algorithm allows for 26 possible movement directions, as shown in Fig. 13. The distance for each direction is shown in Eq. (17):

$$\mathrm{Dt}(p(x)\,,x) = \begin{cases} 1 & \text{if moving along the principal axis direction (6 directions)} \\ \sqrt{2} & \text{If moving along the diagonal direction of the plane (12 directions)} \\ \sqrt{3} & \text{If moving along the diagonal direction of the cube (8 directions)} \end{cases} \tag{17}$$

where C_1, C_2, and C_3 are the energy consumption coefficients for ascent, horizontal flight, and descent directions of the UAV, L_{ascend}, $L_{horizontal}$, and $L_{descent}$ are the displacement distances of the UAV in ascent, horizontal flight, and descent directions, respectively. $W_{redundant}$ is the redundancy cost required for safe flight, and E_{total} is the total cost required for the UAV's inspection.

Based on the above inspection cost function, the inspection cost between inspection points is calculated, and the inspection cost matrix is obtained from the path costs between inspection points. If the A* algorithm is used for pathfinding, only one calculation is required, i.e., $f(x)_{i\rightarrow j} = f(x)_{j\rightarrow i}$. When the inspection distance is too long, replenishment points need to be set up for the UAV, and the number of replenishment points is calculated as shown in Eq. (19):

$$\mathrm{N_{replenish}} = \frac{\mathrm{E_{total}}}{\mathrm{W}} \tag{19}$$

where $N_{replenish}$ is the number of replenishment points, *W* is the total cost for the inspection. The inspection path calculated by the method in Eq. (18) may reduce the length of the ascent or descent routes and increase the horizontal flight distance. Consequently, the UAV's inspection energy consumption decreases, and compared with the shortest path method, the number of replenishment points is reduced. The reduction

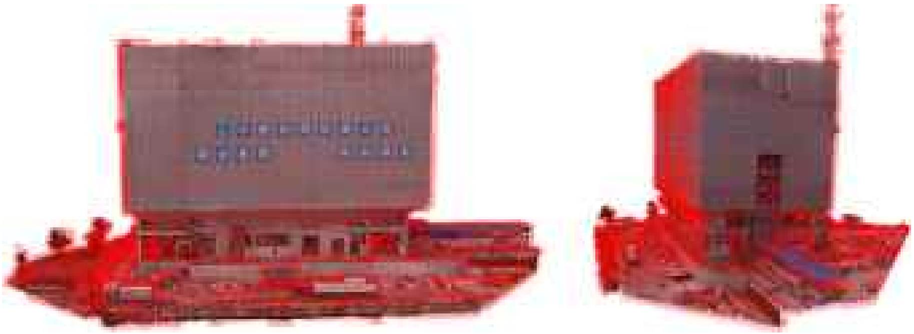


Fig. 11. UAV obstacle space.

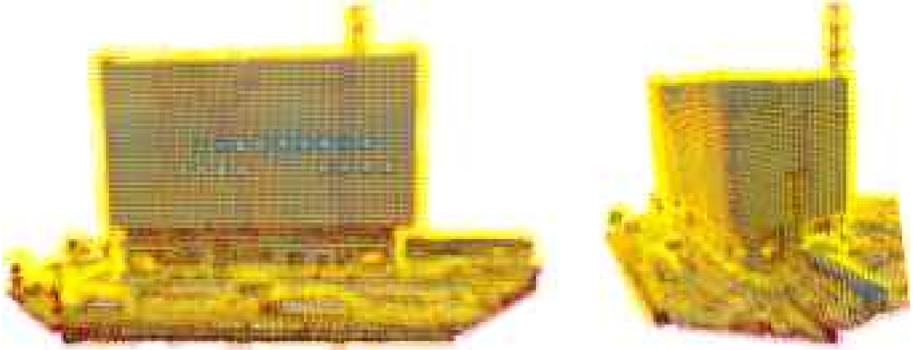


Fig. 12. Obstacle expansion space.

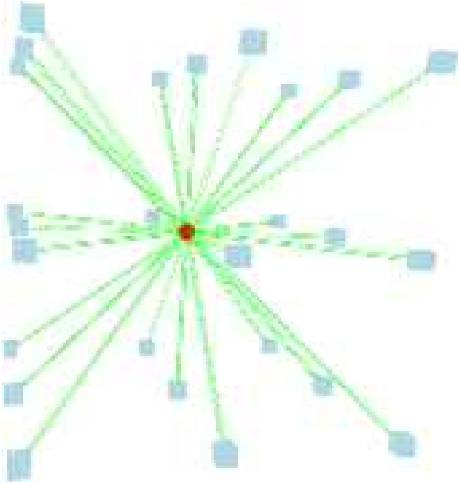


Fig. 13. 26 Motion directions of the UAV-Based on the computed shortest path, the UAV's flight cost is calculated. In current research in the inspection field, the inconsistency of flight costs in different directions of the UAV's actual flight has not been considered. According to the UAV's actual flight conditions, under the same environment, the energy consumption for ascent and descent per unit distance is significantly higher than for horizontal flight. Therefore, this paper proposes a method for calculating the UAV's flight cost, as shown in Eq. (18):

$$\mathrm{E_{total}} = \mathrm{C_1} \cdot \mathrm{L_{ascent}} + \mathrm{C_2} \cdot \mathrm{L_{horizontal}} + \mathrm{C_3} \cdot \mathrm{L_{descent}} + \mathrm{W_{redundant}} \tag{18}$$

in the number of supply points can be determined based on Eqs. (20)–(21):

$$\Delta \mathrm{E} = \mathrm{C_1} \cdot \Delta \mathrm{L_{ascend}} + \mathrm{C_3} \cdot \Delta \mathrm{L_{descent}} - \mathrm{C_2} \cdot \Delta \mathrm{L_{horizontal}} \tag{20}$$

$$\Delta \mathrm{N_{replenish}} = \frac{\Delta \mathrm{E}}{\mathrm{W}} \tag{21}$$

where ΔE is the reduced cost consumption of the path calculated by this paper's method, and ΔL_{ascend} and $\Delta L_{descent}$ are the reduced ascent and descent flight distances, respectively.

3.3.2. Inspection path optimization

In the construction inspection process, finding an optimal inspection path is crucial for the UAV's inspection efficiency and safety. In Section 3.3.1, the flight cost matrix and flight path between two inspection waypoints were calculated, but a global optimal flight path is lacking. Therefore, this section aims to find a complete inspection path. In a network with n nodes, each with distances between nodes, the objective is to find the shortest path that passes through all nodes and returns to the starting waypoint, which is equivalent to solving the Traveling Salesman Problem (TSP). Therefore, the problem can be transformed into finding the minimum value of Eq. (22):

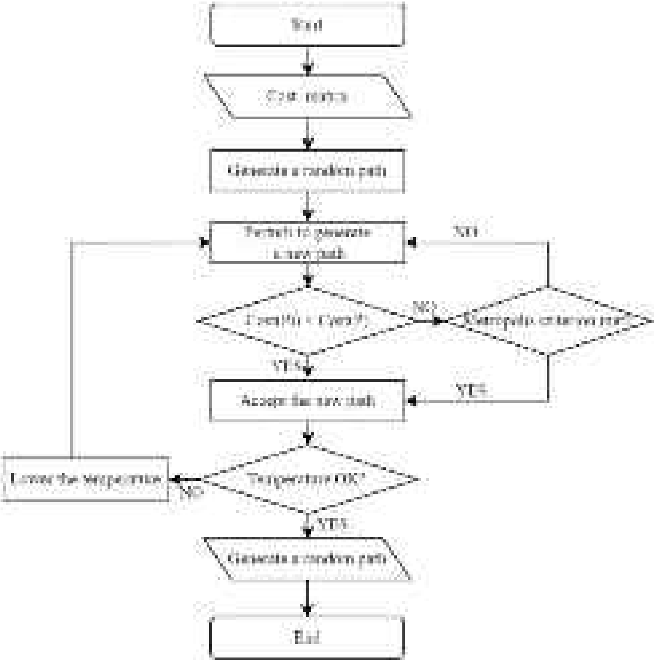


Fig. 14. SA solving process.

$$Cost(P) = \sum_{i=0}^{n-1} D_{P_i,P_{i+1}} + D_{P_n,P_0} \tag{22}$$

where n is the number of inspection waypoints, $cost(p)$ is the total cost of the UAV flight, and $D_{P_i,P_{i+1}}$ is the flight cost between waypoint i and waypoint $i+1$.

Finally, based on the cost matrix, the optimal solution is obtained using the Simulated Annealing (SA) algorithm. The flowchart is shown in Fig. 14. Initially, a random path, initial temperature, cooling factor, and minimum temperature are generated. Small perturbations are applied to the path to generate neighboring solutions. The decision to accept the new solution is based on the temperature and cost difference. As the temperature decreases, the probability of accepting worse solutions also decreases, until the conditions are met and the final path is output.

3.4. Inspection mission generation

After calculating the inspection waypoints and flight paths, the UAV needs to autonomously execute the task. This process requires converting the obtained parameters into formats that the UAV can

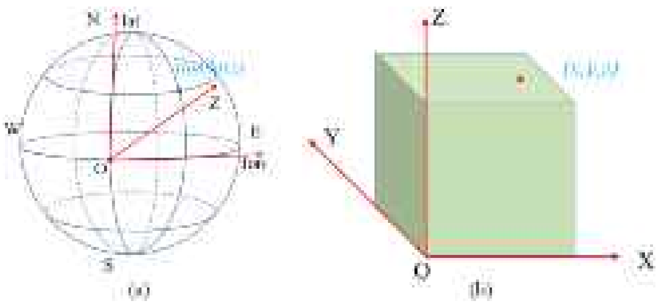


Fig. 15. Coordinate system diagram (a) WGS84 coordinate system (b) Local coordinate system.

understand, mainly involving coordinate system transformations and UAV flight parameter conversions.

3.4.1. *Coordinate system transformation*

In actual UAV flight, the WGS84 coordinate system, which is a latitude and longitude coordinate system, is generally used. However, for the calculation of inspection waypoints and flight paths, a local coordinate system is adopted, which is a Cartesian coordinate system with a local point as the origin. Therefore, it is necessary to convert between these two coordinate systems. To facilitate the conversion between these coordinate systems, a pseudo-Mercator coordinate system is introduced as a bridge for the transformation. The relationship between the two coordinate systems is shown in Fig. 15. The transformation between the local coordinate system and the pseudo-Mercator coordinate system is given by Eq. (23), and the transformation between the pseudo-Mercator coordinate system and the WGS84 coordinate system is described by Eqs. (24)–(25) [67]:

$$\begin{pmatrix} \mathrm{X_m} \\ \mathrm{Y_m} \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} X_A \\ Y_A \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{23}$$

$$X_m = R{\cdot}\lambda{\cdot}\frac{\pi}{180} \tag{24}$$

$$Y_m = R{\cdot}\ln\left(\tan\left(\frac{\pi}{4} + \frac{\phi \times \frac{\pi}{180}}{2}\right)\right) \tag{25}$$

Where λ and ϕ represent the longitude and latitude of the coordinates, $\mathrm{X_A}$ and $\mathrm{Y_A}$ represent the x and y values in the local coordinate system, $\mathrm{X_m}$ and $\mathrm{Y_m}$ represent the x and y values in the pseudo-Mercator coordinate system, and R is the radius of the Earth.

Using these formulas, the conversion from the local coordinate system to the WGS84 coordinate system can be completed. For the UAV's altitude, this paper uses the relative altitude with respect to the takeoff point to achieve more accurate height measurements.

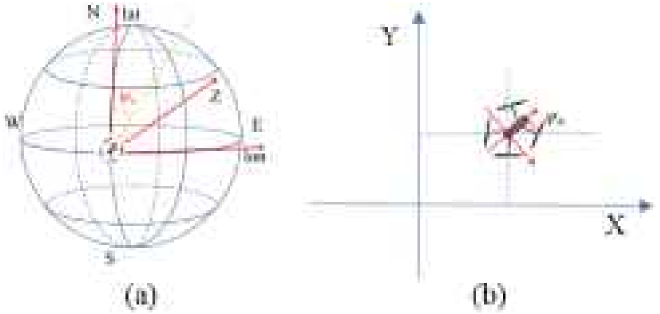


Fig. 16. Relationship between yaw angle and coordinate system.

3.4.2. *UAV inspection parameter transformation*

When the UAV collects data at inspection waypoints, it is necessary to ensure the UAV's attitude in order to complete the inspection task. For this purpose, this paper only considers the UAV's flight attitude yaw angle and gimbal attitude angle. The yaw angle of the flight attitude generally refers to the angle between the UAV and the true north direction. The relationship between the yaw angle and the coordinate system is shown in Fig. 16. The range of the yaw angle is [−180, 180], where 0 degrees corresponds to the UAV facing true north, with clockwise being positive and counterclockwise being negative. The direction vector of the inspection waypoints has been calculated in Section 3.2, so now we need to convert this direction into an angle that the UAV can understand. Using the coordinate transformation method from Section 3.4.1, we can convert the local coordinates to the pseudo-Mercator coordinate system. In this coordinate system, the Y-axis points towards true north. Therefore, we calculate the angle between the direction vector and the positive Y-axis to determine the yaw angle. The calculation formula is given by Eq. (26):

$$\theta = \arccos\left(\frac{\mathrm{v}{\cdot}\mathrm{n}}{\|\mathrm{v}\|\|\mathrm{n}\|}\right) \tag{26}$$

where v is the UAV's directional vector at the waypoint, and n is a unit vector pointing towards the geographic North. If the x-component of the direction vector is less than 0, the calculated angle θ is the clockwise angle from the north direction; if the x-component of the direction vector is greater than 0, the angle θ is the counterclockwise angle from the north direction.

Since the Field of View (FOV) in this paper is determined not only by the camera size but also by the dynamic FOV based on the gimbal rotation, it is necessary to calculate the gimbal's yaw angle. The yaw angle of the gimbal is also the angle between the UAV's orientation and the true north direction, with the same range as the UAV's yaw angle. Therefore, the actual yaw angle of the gimbal, $\psi_{absolute}$, can be calculated based on the UAV's yaw angle ψ_{UAV}, as shown in Eq. (27).

$$\psi_{\mathrm{absolute}} = \psi_{\mathrm{UAV}} + \psi_{\mathrm{gimbal}} \tag{27}$$

However, when the UAV's body yaw angle exceeds the range of [−180,180], a modulus operation is required to adjust it to the correct angle. The camera's pitch angle θ is based on the UAV's horizontal plane and does not require parameter conversion. The pitch angle can be used directly within its range.

3.4.3. *UAV inspection execution file generation*

After the required parameters for UAV inspection are calculated, manual operation of the UAV can be quite slow. Therefore, it is necessary to convert the corresponding parameters into a UAV inspection execution file. The execution file includes not only the UAV inspection parameters but also the UAV flight speed, waypoint hover time, photo capture interval, and more. This paper automatically generates a KML flight path file for UAV execution using Python, based on the flight route writing requirements of DJI's Cloud API.

4. Case study

Before selecting this complex project, the proposed method was preliminarily validated on several smaller-scale construction sites, where it demonstrated promising performance in terms of stability and applicability. To further assess its robustness under more challenging conditions, this paper ultimately selected a large-scale complex construction project of an irregular building in Shenzhen, China as the experimental subject, as shown in Fig. 17(a)–(b). The project features a highly intricate structural form, divided into five independent construction zones with overlapping workflows. The site environment exhibits typical characteristics of irregularity and obstruction, posing significant challenges for path planning in complex construction

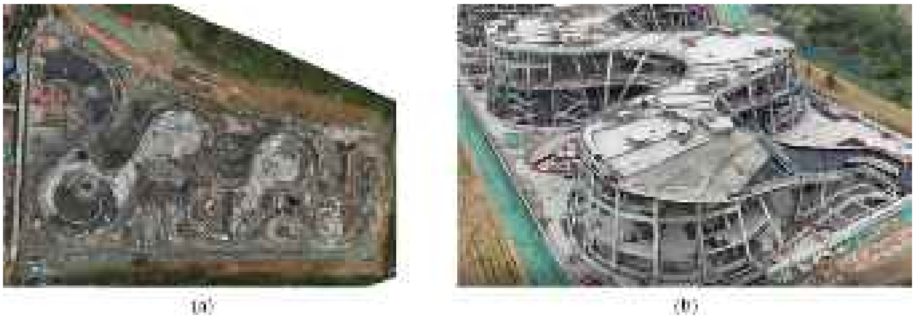


Fig. 17. Construction site situation (a) Orthophoto (b) Site image.

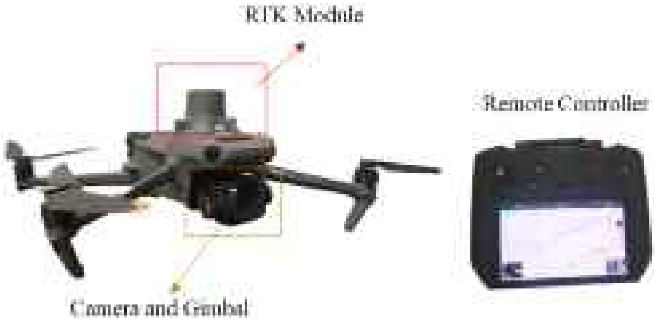


Fig. 18. DJI Mavic 3 Enterprise UAV and controller.

scenarios. Therefore, the experimental validation is conducted across the entire construction site to comprehensively evaluate the adaptability and effectiveness of the proposed method in real-world complex environments. This paper uses the DJI Mavic 3 Enterprise as the experimental UAV, as shown in Fig. 18. The UAV is small in size and has a relatively long flight time, with a maximum flight time of 45 min according to the official website. It offers high flexibility and practicality for construction site inspections. Additionally, the UAV can directly import flight path files for execution, without the need for manual setup of flight parameters. The UAV is equipped with RTK functionality, which provides precise UAV positioning. The computer used for 3D reconstruction and flight path planning is equipped with an Intel Core i9-12950HX CPU, Nvidia RTX 3080Ti Laptop 16G GPU, and 64GB of RAM.

4.1. Construction site environmental modeling

To generate a high-resolution 3D model of the construction site, the UAV was programmed to follow a predefined flight plan with overlapping image capture. A total of 312 aerial images were collected using the DJI Mavic 3 Enterprise and subsequently processed using DJI Terra to produce the environmental model. The UAV's data acquisition configuration—covering flight altitude, speed, camera resolution, gimbal angles, image overlap, and other key parameters—is summarized in Table 1. These settings were selected to ensure complete site coverage, sufficient overlap, and optimal ground sampling accuracy for reliable reconstruction. The resulting 3D triangulation and the final environmental model are shown in Fig. 19(a) and Fig. 19(b), respectively.

Table 1
UAV data acquisition parameters for 3D reconstruction.

Parameter	Value	Parameter	Value
Flight Altitude	120 m	FOV	84°
Flight Speed	10 m/s	GSD	2.1 cm/ pixel
Camera Resolution	5280 × 3956 pixels	Gimbal Pitch Angle	−90°/−45°
Camera Sensor Size	17.4 mm × 13 mm	Image Overlap (Side)	70 %
Focal Length	12 mm	Image Overlap (Forward)	80 %

The structural BIM of a specific building is converted into PCD point cloud format, and the construction site environmental model is also converted into point cloud format. Using the inspection area segmentation method proposed in Section 3.1.2, point cloud registration is performed to obtain the rotation matrix. Subsequently, the BIM models of other construction areas on the site are registered and aligned using the obtained rotation matrix, enabling unified segmentation of the construction scene. As shown in Fig. 20(a), the original BIM model of one construction area is illustrated, while Fig. 20(b) presents the segmentation result for that area.

4.2. Inspection waypoint calculation

The DJI Mavic 3Enterprise UAV is equipped with a wide-angle camera and a zoom camera. The dimensions of the wide-angle camera are 17.4 mm × 13 mm, with a focal length of 12 mm, and the zoom camera has a sensor size of 6.4 mm × 4.8 mm with a focal length of 30 mm. Based on the construction site's fine inspection quality GSD (Ground Sample Distance) requirements and the safety flight distance determined by the construction conditions, the UAV's distance to the inspected object is set between 6 and 15 m. This paper will calculate the inspection waypoints based on this inspection distance.

Considering the UAV's gimbal pitch angle range of [−90, 35] and yaw angle range of [−35, 35], and taking into account the actual gimbal situation, this paper sets the pitch angle range as [−90, 10] and the yaw angle range as [−20,20]. The inspection waypoint space is generated based on the inspection distance, and the space is voxelized to achieve spatial discretization, with a voxel size of 50 cm, which corresponds to half the size of the UAV in flight. Using the proposed waypoint selection strategy, a total of 108 waypoints were generated, resulting in a coverage rate of 92.30 %. The spatial distribution of these waypoints, as well as the corresponding inspection coverage performance, is illustrated in Fig. 21.

4.3. Inspection path optimization

Based on the environmental model of the construction site, feasible spaces are divided into safe and unsafe areas. The A* algorithm is used to compute paths that cannot be directly flown to, resulting in a waypoint distance matrix. Using this distance matrix, the simulated annealing algorithm is applied to compute the minimum-cost inspection path. Initial temperature, cooling rate, and termination conditions are set, and the iteration starts by randomly generating an initial solution. The starting and ending waypoints of the path are fixed. After 20,053 iterations, the algorithm stops when the termination condition is met, providing the relatively optimal solution. Fig. 22(a) shows the iteration process, where the UAV's flight cost is 3502.62. Fig. 22(b) shows the flight path after the iteration.

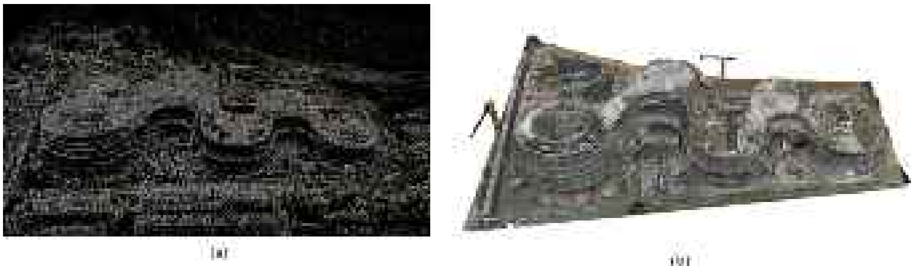


Fig. 19. 3D reconstruction results (a) Triangulation results (b) 3D reconstruction model.

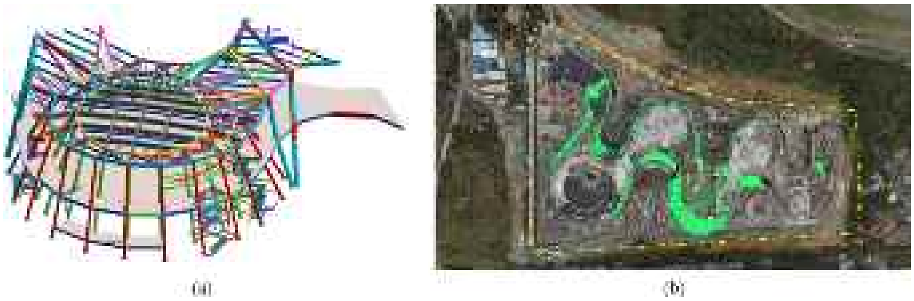


Fig. 20. Inspection area segmentation (a) Structural BIM model (b) Construction area segmentation result.

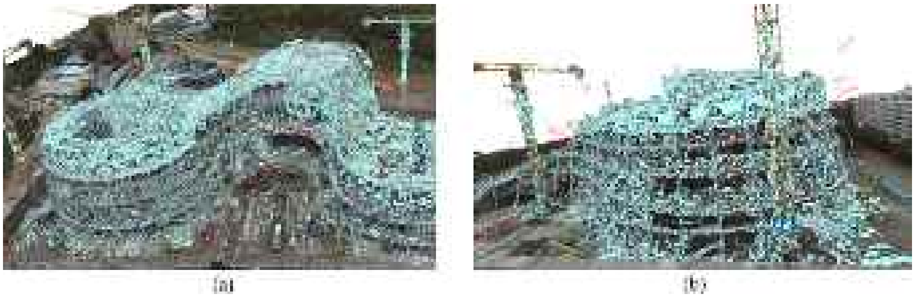


Fig. 21. Inspection waypoints of the inspection area.

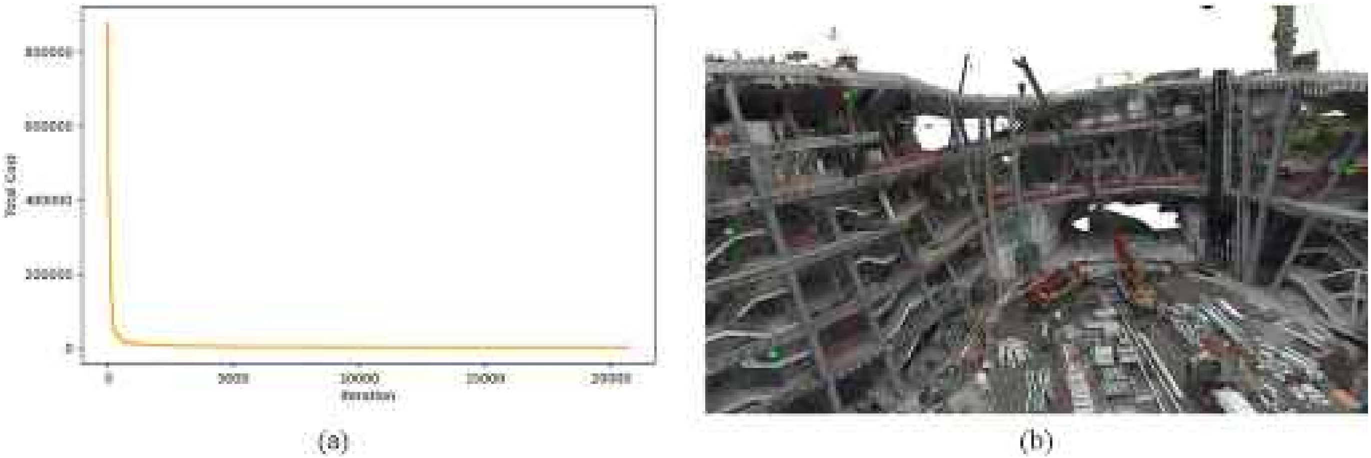


Fig. 22. UAV inspection path (a) Simulated annealing algorithm iteration process (b) Inspection path result.

4.4. UAV inspection execution file generation

To automate the UAV inspection task, this paper uses Python to implement the conversion of waypoint coordinate systems. Two points are selected at the construction site, and their geographic coordinates are measured. Then, their local coordinate system values are measured. The conversion parameters are calculated using Eqs. (20)–(22), and based on these parameters, all inspection coordinates are converted into executable UAV coordinates. The UAV's yaw angles for all inspection points are converted from the local coordinate system to the UAV 's

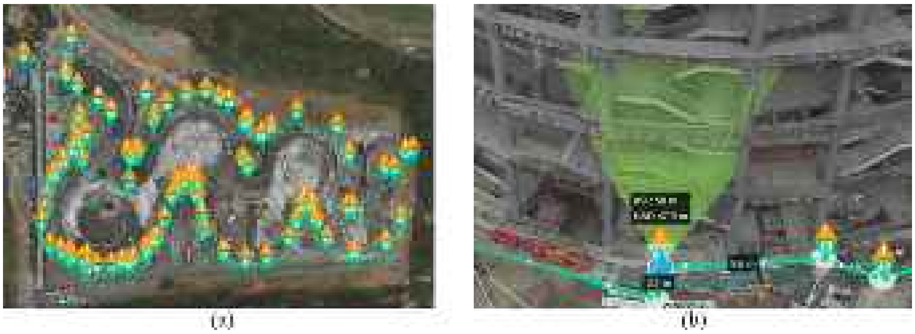


Fig. 23. Coordinate conversion results (a) Converted UAV inspection path (b) Example of photo capture angle.

Table 2
Conversion results of inspection waypoint parameters.

	Local Coordinate System			WGS84 Coordinate System			
	X	Y	Z	Lat	Lon	Z_{WSG84}	YAW
1	37.37643003	105.2255001	−94.5720	114.35740	22.69528	62.3319	109.5642
2	62.87643003	46.72550011	−94.5720	114.35766	22.69475	62.3320	24.9701
3	34.37643003	85.72550011	−94.5720	114.35738	22.69510	62.3319	85.6351
	…	…	…	…	…	…	…
106	196.3764	154.7255	−111.0720	114.35894	22.69575	45.8319	−114.9671
107	64.3764	126.2255	−76.5720	114.35766	22.69547	80.3312	130.9413
108	88.3764	163.7255	−88.5720	114.35789	22.69582	68.3319	158.3592

coordinate system. Additionally, the camera's yaw angles are transformed into the UAV 's coordinate system using Eq. (23). Fig. 23 shows the converted UAV flight path and an example of an angle for capturing photos. The conversion results for the UAV inspection point coordinates, yaw angles, and camera tilt angles are presented in Table 2. Using all the calculated parameters, a Python script is used to automatically generate the UAV execution file and create a KML file.

4.5. Results analysis

To validate the practical applicability of the proposed fine-grained inspection path planning method, we conducted detailed analyses from three aspects: waypoint effectiveness, inspection efficiency, and energy cost.

4.5.1. Waypoint effectiveness analysis

In order to systematically assess the usability of the generated waypoints in real-world construction scenarios, this paper defines waypoint effectiveness as the capability of the waypoint set to achieve sufficient spatial coverage of the target inspection area while ensuring the safety of UAV flight. Comparative experiments were conducted using two representative methods. The first extracts waypoints directly from BIM models, and the second generates waypoints automatically based on point cloud data. The results, as summarized in Table 3, show that the proposed method achieved a coverage rate of 92.30 %, which is higher than that of the point cloud-based approach and slightly lower than that of the BIM-based approach. However, in terms of flight safety, the proposed method performed best, generating a total of 108 waypoints without any identified as unsafe. In comparison, the BIM-based approach had a high proportion of unsafe waypoints, accounting for 25.4 %, while the point cloud-based approach had a proportion of 8.9 %. These findings demonstrate that although the BIM-based method achieved the highest coverage, it tended to generate risk-prone waypoints due to its inability to account for occlusions or obstacles in actual scenes. In contrast, the proposed method effectively balances inspection coverage with flight safety, indicating its superior waypoint validity and strong potential for deployment in complex construction environments.

Table 3
Comparison of inspection waypoints.

	Number of Waypoints	Unsafe Waypoints	Unsafe Ratio	Coverage Rate
Proposed Method	108	0	0%	92.30 %
BIM [33]	268	68	25.4 %	94.26 %
Point Cloud [40]	168	15	8.9 %	90.71 %

To further verify the effectiveness and safety of the proposed waypoint planning strategy, a real-world flight test was conducted. The actual UAV trajectory was recorded and compared with the planned waypoint data. Fig. 24(a) shows the distance from each waypoint to the 3D mesh surface for both the planned path and the actual UAV flight. All values remained above the defined 5-m safety threshold, indicating that the planned path is feasible and provides sufficient clearance from surrounding structures under real flight conditions. Fig. 24(b) presents the position errors between the planned and actual UAV locations in both the horizontal and vertical directions. The horizontal error was mostly within ±0.15 m, and the vertical error was within ±0.2 m. These results confirm that the UAV can closely follow the planned path with high accuracy, and that the proposed method is robust and reliable in maintaining safe inspection distances during autonomous execution.

4.5.2. Inspection efficiency analysis

A quantitative evaluation of inspection efficiency was conducted by comparing the proposed method with manual inspection procedures and the conventional UAV-based shortest-path approach. In typical on-site operations, inspection personnel carry out four rounds of inspections per day, with each round taking approximately 80 min. In contrast, the UAV-based method involves three main stages, namely environment modeling, waypoint computation, and automated flight. As shown in Table 4, the proposed method requires a total time of 255 min per inspection cycle. Compared with the 320 min needed for manual inspection, this represents a time saving of approximately 65 min, with an improvement in overall efficiency of 20.3 %. Although the proposed method takes slightly longer than the shortest-path approach, which

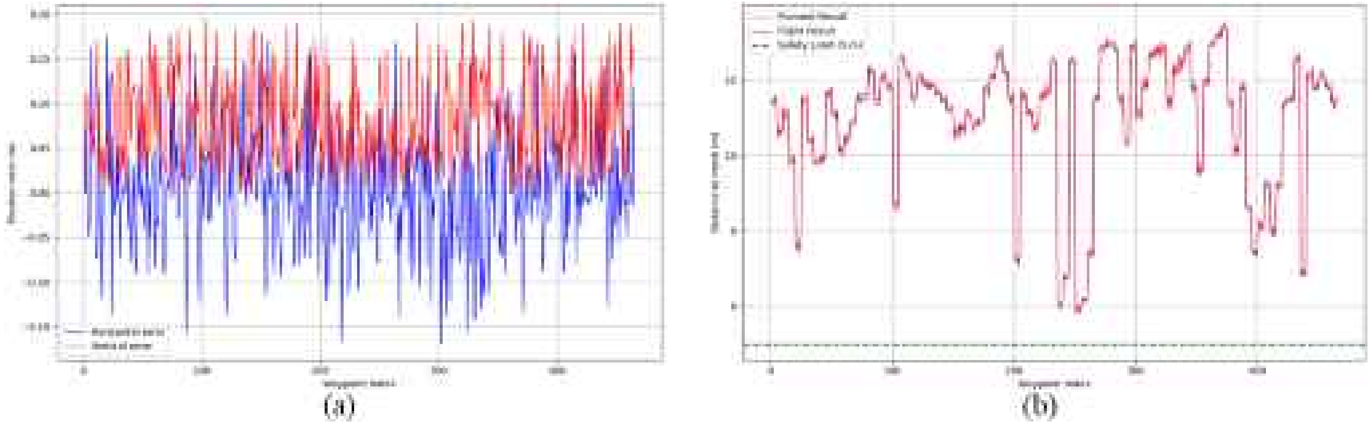


Fig. 24. Flight execution results of UAV inspection tasks and waypoint deviation outcomes (a) Distance between actual UAV trajectory and 3D mesh surface (b) Horizontal and vertical position errors at each waypoint.

Table 4
Comparison of inspection efficiency.

	Environment Modeling (min)	Waypoint Computation (min)	Flight Duration (min)	Total Duration (min)
Manual	0	0	320	320
Proposed Method	30	45	180	255
Shortest-Path	30	45	168	233

Table 5
Inspection cost calculation results.

	Path Length	Total Energy Cost	Estimated Battery Consumption
Traditional	1492 m	4030.92 units	1.51 units
Proposed Method	1517 m	3502.62 units	1.32 units

completes the cycle in 233 min, it provides better performance in terms of waypoint rationality, flight safety, and inspection completeness. This demonstrates that the proposed method can significantly enhance efficiency while maintaining inspection quality, confirming its applicability in dynamic and complex construction environments.

4.5.3. Energy cost analysis

Most existing UAV inspection methods rely on shortest-path planning, without considering the significant differences in energy consumption associated with upward, horizontal, and downward flight directions. This paper adopts a directional energy consumption model, as introduced in Section 3.3.1, to compare the proposed method with the conventional shortest-path approach. As shown in Table 5, although the proposed method results in a slightly longer path length, the total energy cost is reduced by 15.08 %, and battery consumption is reduced by 12.58 %. This improvement is attributed to the optimization process which incorporates directional energy coefficients, including 2.16 for ascending flight, 1.98 for descending, and 1.26 for horizontal movement. These results indicate that the proposed method offers considerable advantages in terms of energy efficiency. Furthermore, in large-scale inspection tasks where the route is long and waypoint density is high, the proposed approach can significantly reduce the number of required battery replacements or supply stations, thereby enhancing mission continuity and overall system robustness.

5. Discussion

In previous studies, the focus was primarily on the inspection coverage of UAVs and waypoints planning. This paper uses a 3D reconstruction model as the environmental model for UAV inspections, improving the safety, efficiency, and automation of the inspection process. With the increasing demand for smarter construction sites and the rising cost of labor, reducing the number of inspection personnel becomes a feasible solution. The primary role of construction workers during inspections is visual inspection and issuing reminders to other personnel. Therefore, an automatic UAV-based inspection solution can effectively serve as a replacement.

5.1. Advantages

In traditional construction inspection workflows, on-site personnel often need to perform repetitive and hazardous tasks such as climbing, navigating confined spaces, or operating in high-risk zones. In contrast, UAV-based inspection methods significantly reduce such safety risks by enabling remote data acquisition. Moreover, UAVs allow for flexible scheduling of inspection tasks, which enhances both temporal resolution and responsiveness to construction progress. Compared with existing UAV waypoint generation studies, the approach proposed in this research presents the following innovations: (1) It introduces a UAV inspection path planning scheme grounded in 3D reconstruction, which avoids redundant waypoints and minimizes exposure to dangerous areas; (2) It incorporates a BIM-based scene segmentation method that enables automatic decomposition of complex construction environments, facilitating more precise and structured inspection region definition; (3) It integrates directional flight cost modeling into the path planning process, allowing for realistic evaluation and minimization of energy expenditure during UAV operations. These methodological improvements enhance the adaptability, safety, and cost-effectiveness of UAV-based inspection workflows in real-world construction scenarios. In addition, the system is designed to operate at a flight altitude of 120 m, which complies with local regulatory limits and has been experimentally validated to provide sufficient resolution and model completeness for most typical construction scenarios.

5.2. Limitations

While the proposed framework successfully integrates UAV technology, BIM, and 3D reconstruction to enhance inspection automation, several limitations remain to be addressed: First, the effectiveness of BIM-based scene segmentation is influenced by the quality and granularity of the BIM data. In complex or unstructured environments, inaccurate semantic segmentation may affect the precision of inspection region modeling. Future work could explore the use of standardized component classification or hybrid learning models to improve segmentation robustness. Second, the computational efficiency of distance matrix generation and path optimization becomes a bottleneck when applied to extremely large or densely structured scenes. This may impact real-time performance or scalability in large-scale projects. Algorithmic optimizations or hierarchical path planning schemes may be required to alleviate this limitation. Third, although the use of 3D reconstruction provides an accurate representation of the physical environment, it requires regular updates to remain aligned with the dynamically changing construction site. In this paper, model updates were performed every 3–4 days; however, in rapidly evolving projects, this temporal resolution may be insufficient. A potential solution is the integration of real-time reconstruction methods or adaptive update triggers based on construction progress. Fourth, the current framework assumes that a UAV flight altitude of 120 m provides sufficient reconstruction accuracy for most construction inspection tasks. While this holds true for low- to mid-rise structures, tall buildings or sites with complex vertical geometry may suffer from reduced detail in upper façade regions, particularly when only nadir or near-nadir views are available. In such cases, incomplete reconstruction can lead to missing or inaccurate obstacle representations, which may compromise waypoint generation and path-planning accuracy. To address this, future research may explore adaptive flight altitude strategies, integration of oblique photogrammetry, or deployment of UAVs with zoom capabilities to enhance reconstruction quality in high-rise environments.

6. Conclusion

This paper presented a refined construction site inspection path planning method based on 3D reconstruction. First, an environmental model for the UAV is constructed through preliminary flight and 3D reconstruction, providing a solid foundation for safe flight and waypoint calculation. Second, based on the BIM model, the construction scene is segmented by building, and the inspection units are defined. The segmentation of the scene is determined based on two indicators: overlap rate and RMSE. Next, the UAV inspection waypoints and headings are calculated using an improved artificial potential field method and a greedy algorithm. Finally, optimal cost path planning for the UAV is achieved using the A* algorithm and simulated annealing algorithm.

To validate the proposed methodology, comprehensive experiments were conducted on a real construction site. Comparative analysis with BIM-based and point cloud-based methods demonstrated that the proposed approach achieves a better balance between inspection coverage and waypoint safety. Specifically, the method ensures full coverage with zero dangerous waypoints, outperforming traditional schemes that showed up to 25.4 % unsafe waypoints. In terms of inspection efficiency, the proposed UAV-based method reduced total operational time by 20.3 % compared to manual inspection, and achieved a 15.08 % reduction in flight energy cost relative to the traditional shortest-path algorithm. These results confirm the method's applicability in real-world construction scenarios and its potential to significantly enhance safety, coverage effectiveness, and energy efficiency. Moreover, the modular segmentation of BIM and reconstruction data facilitates scalable deployment across multi-zone construction projects.

However, limitations still exist. The accuracy of scene segmentation depends on the quality and granularity of the BIM model and its alignment with real-time progress. In addition, while the A* and simulated annealing algorithms enable reliable path computation, their computational complexity may limit responsiveness in highly complex environments. Future work should focus on real-time adaptive segmentation strategies and lightweight path planning algorithms to further enhance the robustness and scalability of the system. In conclusion, this paper provides a practical and scalable UAV inspection planning framework suitable for complex and evolving construction scenarios.

In future work, the proposed inspection path planning framework will be extended to accommodate multi-UAV cooperative inspection scenarios, aiming to further improve task parallelism and inspection efficiency in large-scale construction environments. To support this, advanced decomposition strategies—such as divide-and-conquer algorithms—will be introduced to optimize the distribution of inspection tasks among multiple UAVs. These strategies are expected to significantly reduce the computational complexity involved in waypoint generation and flight cost matrix construction. Furthermore, efforts will be made to integrate real-time scene updates and adaptive segmentation techniques, allowing the system to dynamically adjust to evolving construction progress. To enhance the safety and reliability of UAV operations, future implementations will also consider the deployment of onboard sensing modules capable of real-time environmental perception. By analyzing proximity, obstacles, and environmental changes during flight, these sensors will enable dynamic risk assessment and avoidance, ensuring safer flight paths in uncertain or cluttered construction environments. Such developments will enhance the robustness, scalability, and real-time responsiveness of the UAV inspection system, facilitating its deployment in increasingly complex and dynamic site conditions.

CRediT authorship contribution statement

Xin Liu: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Conceptualization. **Wen Yi:** Writing – original draft, Supervision, Methodology, Formal analysis. **Penglu Chen:** Writing – review & editing, Software, Methodology, Conceptualization. **Yi Tan:** Writing – review & editing, Supervision, Methodology, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research is supported by the National Natural Science Foundation of China (NSFC) (Grant No. 52308319 and Grant No. 72201229).

Data availability

Data will be made available on request.

References

[1] O.O.T.M.O.H.A.U.R. Development, Notice from the General Office of the Ministry of Housing and Urban Rural Development on Production Safety Accidents in Housing and Municipal Engineering in 2020. https://www.mohurd.gov.cn/gongkai/zhengce/zhengcefilelib/202210/20221026_768565.html, 2022.

[2] OSHA, Commonly Used Statistics. https://www.osha.gov/data/commonstats, 2025.

[3] S. Zhao, F. Kang, J. Li, C. Ma, Structural health monitoring and inspection of dams based on UAV photogrammetry with image 3D reconstruction, Autom. Constr. 130 (2021) 103832, https://doi.org/10.1016/j.autcon.2021.103832.

[4] H. Liang, J. Cho, S. Seo, Construction site multi-category target detection system based on UAV low-altitude remote sensing, Remote Sens. 15 (6) (2023) 1560, https://doi.org/10.3390/rs15061560.

[5] D. Lin, N. Yang, Q. Miao, X. Cui, D. Xu, True 3D thermal inspection of buildings using multimodal UAV images, J. Build. Eng. 100 (2025) 111806, https://doi.org/10.1016/j.jobe.2025.111806.

[6] J. Wang, W. Sun, W. Shou, X. Wang, C. Wu, H.-Y. Chong, et al., Integrating BIM and LiDAR for real-time construction quality control, J. Intell. Robot. Syst. 79 (2015) 417–432, https://doi.org/10.1007/s10846-014-0116-8.

[7] S.-N. Yu, J.-H. Jang, C.-S. Han, Auto inspection system using a mobile robot for detecting concrete cracks in a tunnel, Autom. Constr. 16 (3) (2007) 255–261, https://doi.org/10.1016/j.autcon.2006.05.003.

[8] Y. Tan, S. Li, H. Liu, P. Chen, Z. Zhou, Automatic inspection data collection of building surface based on BIM and UAV, Autom. Constr. 131 (2021) 103881, https://doi.org/10.1016/j.autcon.2021.103881.

[9] C. Liu, Y. Liu, H. Wu, R. Dong, A safe flight approach of the UAV in the electrical line inspection, Int. J. Emerg. Electr. Power Syst. 16 (5) (2015) 503–515, https://doi.org/10.1515/ijeeps-2015-0021.

[10] T. Rakha, A. Gorodetsky, Review of unmanned aerial system (UAS) applications in the built environment: towards automated building inspection procedures using drones, Autom. Constr. 93 (2018) 252–264, https://doi.org/10.1201/9780429172410-14.

[11] H. Ren, Y. Zhao, W. Xiao, Z. Hu, A review of UAV monitoring in mining areas: current status and future perspectives, Int. J. Coal Sci. Technol. 6 (2019) 320–333, https://doi.org/10.1007/s40789-019-00264-5.

[12] J. Su, X. Zhu, S. Li, W.-H. Chen, AI meets UAVs: a survey on AI empowered UAV perception systems for precision agriculture, Neurocomputing 518 (2023) 242–270, https://doi.org/10.1016/j.neucom.2022.11.020.

[13] X. Huang, Y. Wu, Y. Zhang, B. Li, Structural defect detection technology of transmission line damper based on UAV image, IEEE Trans. Instrum. Meas. 72 (2022) 1–14, https://doi.org/10.1109/tim.2022.3228008.

[14] N. Iversen, O.B. Schofield, L. Cousin, N. Ayoub, G. Vom Bögel, E. Ebeid, Design, integration and implementation of an intelligent and self-recharging drone system for autonomous power line inspection, in: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2021, pp. 4168–4175, https://doi.org/10.1109/iros51168.2021.9635924.

[15] J. Seo, L. Duque, J.P. Wacker, Field application of UAS-based bridge inspection, Transp. Res. Rec. 2672 (12) (2018) 72–81, https://doi.org/10.1177/0361198118780825.

[16] J. Seo, L. Duque, J. Wacker, Drone-enabled bridge inspection methodology and application, Autom. Constr. 94 (2018) 112–126, https://doi.org/10.1016/j.autcon.2018.06.006.

[17] J.-L. Xiao, J.-S. Fan, Y.-F. Liu, B.-L. Li, J.-G. Nie, Region of interest (ROI) extraction and crack detection for UAV-based bridge inspection using point cloud segmentation and 3D-to-2D projection, Autom. Constr. 158 (2024) 105226, https://doi.org/10.1016/j.autcon.2023.105226.

[18] M. Cano, J.L. Pastor, R. Tomás, A. Riquelme, J.L. Asensio, A new methodology for bridge inspections in linear infrastructures from optical images and HD videos obtained by UAV, Remote Sens. 14 (5) (2022) 1244, https://doi.org/10.3390/rs14051244.

[19] F. Wang, Y. Zou, X. Chen, C. Zhang, L. Hou, E. del Rey Castillo, et al., Rapid in-flight image quality check for UAV-enabled bridge inspection, ISPRS J. Photogramm. Remote Sens. 212 (2024) 230–250, https://doi.org/10.1016/j.isprsjprs.2024.05.008.

[20] E. Jeong, J. Seo, J.P. Wacker, UAV-aided bridge inspection protocol through machine learning with improved visibility images, Expert Syst. Appl. 197 (2022) 116791, https://doi.org/10.1016/j.eswa.2022.116791.

[21] S. Zhang, M. Loosemore, R.Y. Sunindijo, S. Galvin, J. Wu, S. Zhang, Assessing safety risk management performance in Chinese subway construction projects: a multistakeholder perspective, J. Manag. Eng. 38 (4) (2022) 05022009, https://doi.org/10.1061/(ASCE)ME.1943-5479.0001062.

[22] A. Adibfar, M. Razkenari, A. Costin, Review and assessment of technical and legal challenges in application of unmanned aerial vehicles in monitoring and inspection of bridges, Intell. Transp. Infrastruct. 2 (2023) liad023, https://doi.org/10.1093/iti/liad023.

[23] Y. Tan, W. Yi, P. Chen, Y. Zou, An adaptive crack inspection method for building surface based on BIM, UAV and edge computing, Autom. Constr. 157 (2024) 105161, https://doi.org/10.1016/j.autcon.2023.105161.

[24] R. Cai, J. Li, Y. Tan, W. Shou, A. Butera, Automated geometric quantification of building Exterior Wall cracks based on computer vision, J. Perform. Constr. Facil. 38 (4) (2024) 04024015, https://doi.org/10.1061/jpcfev.cfeng-4618.

[25] Y. Tan, G. Li, R. Cai, J. Ma, M. Wang, Mapping and modelling defect data from UAV captured images to BIM for building external wall inspection, Autom. Constr. 139 (2022) 104284, https://doi.org/10.1016/j.autcon.2022.104284.

[26] C. Zhang, F. Wang, Y. Zou, J. Dimyadi, B.H. Guo, L. Hou, Automated UAV image-to-BIM registration for building façade inspection using improved generalised Hough transform, Autom. Constr. 153 (2023) 104957, https://doi.org/10.1016/j.autcon.2023.104957.

[27] M. Gheisari, B. Esmaeili, Unmanned aerial systems (UAS) for construction safety applications, in: Construction Research Congress 2016, 2016, pp. 2642–2650, https://doi.org/10.1061/9780784479827.263.

[28] L. Yu, M.M. Huang, S. Jiang, C. Wang, M. Wu, Unmanned aircraft path planning for construction safety inspections, Autom. Constr. 154 (2023) 105005, https://doi.org/10.1016/j.autcon.2023.105005.

[29] J.G. Martinez, M. Gheisari, L.F. Alarcon, UAV integration in current construction safety planning and monitoring processes: case study of a high-rise building construction project in Chile, J. Manag. Eng. 36 (3) (2020) 05020005, https://doi.org/10.1061/(ASCE)ME.1943-5479.0000761.

[30] J. Wu, L. Peng, J. Li, X. Zhou, J. Zhong, C. Wang, et al., Rapid safety monitoring and analysis of foundation pit construction using unmanned aerial vehicle images, Autom. Constr. 128 (2021) 103706, https://doi.org/10.1016/j.autcon.2021.103706.

[31] R.O. Rey, R.R.S. de Melo, D.B. Costa, Design and implementation of a computerized safety inspection system for construction sites using UAS and digital checklists–smart Inspecs, Saf. Sci. 143 (2021) 105430, https://doi.org/10.1016/j.ssci.2021.105430.

[32] D.D. York, A.J. Al-Bayati, Z.Y. Al-Shabbani, Potential applications of uav within the construction industry and the challenges limiting implementation, in: Construction Research Congress 2020, American Society of Civil Engineers, Reston, VA, 2020, pp. 31–39, https://doi.org/10.1061/9780784482889.004.

[33] X. Huang, Y. Liu, L. Huang, S. Stikbakke, E. Onstein, BIM-supported drone path planning for building exterior surface inspection, Comput. Ind. 153 (2023) 104019, https://doi.org/10.1016/j.compind.2023.104019.

[34] H. Hamledari, S. Sajedi, B. McCabe, M. Fischer, Automation of inspection mission planning using 4D BIMs and in support of unmanned aerial vehicle–based data collection, J. Constr. Eng. Manag. 147 (3) (2021) 04020179, https://doi.org/10.1061/(asce)co.1943-7862.0001995.

[35] A. Ibrahim, M. Golparvar-Fard, 4D BIM based optimal flight planning for construction monitoring applications using camera-equipped UAVs, in: ASCE International Conference on Computing in Civil Engineering 2019, American Society of Civil Engineers, Reston, VA, 2019, pp. 217–224, https://doi.org/10.1061/9780784482438.028.

[36] Y. Sun, O. Ma, Automating aircraft scanning for inspection or 3D model creation with a UAV and optimal path planning, Drones 6 (4) (2022) 87, https://doi.org/10.3390/drones6040087.

[37] A. Bono, L. D`Alfonso, G. Fedele, A. Filice, E. Natalizio, Path planning and control of a UAV fleet in bridge management systems, Remote Sens. 14 (8) (2022) 1858, https://doi.org/10.3390/rs14081858.

[38] I.Z. Biundini, M.F. Pinto, A.G. Melo, A.L. Marcato, L.M. Honório, M.J. Aguiar, A framework for coverage path planning optimization based on point cloud for structural inspection, Sensors 21 (2) (2021) 570, https://doi.org/10.3390/s21020570.

[39] D. Li, Y. Ding, G. Song, D. Xue, C. Jin, Laser point cloud location-based research on patrol inspection of transmission line UAV, in: Second International Conference on Green Communication, Network, and Internet of Things (CNIoT 2022) vol. 12586, SPIE, 2023, pp. 172–177, https://doi.org/10.1117/12.2667872.

[40] Y. An, J. Ning, C. Hou, J. Ou, Efficient low-collision UAV-based automated structural surface inspection using geometric digital twin and voxelized obstacle information, Autom. Constr. 171 (2025) 105972, https://doi.org/10.1016/j.autcon.2025.105972.

[41] M. Liu, Robotic online path planning on point cloud, IEEE Trans. Cybern. 46 (5) (2015) 1217–1228, https://doi.org/10.1109/tcyb.2015.2430526.

[42] F. Yan, J. Wang, G. He, H. Chang, Y. Zhuang, Sparse semantic map building and relocalization for UGV using 3D point clouds in outdoor environments, Neurocomputing 400 (2020) 333–342, https://doi.org/10.1016/j.neucom.2020.02.103.

[43] T. Koch, M. Körner, F. Fraundorfer, Automatic and semantically-aware 3D UAV flight planning for image-based 3D reconstruction, Remote Sens. 11 (13) (2019) 1550, https://doi.org/10.3390/rs11131550.

[44] S. Zhang, C. Liu, N. Haala, Guided by model quality: UAV path planning for complete and precise 3D reconstruction of complex buildings, Int. J. Appl. Earth Obs. Geoinf. 127 (2024) 103667, https://doi.org/10.1016/j.jag.2024.103667.

[45] G. Zhang, J. Lin, M. Jiang, Y. Wang, L. Gao, R. Wang, Route planning based on 3D single model and power grid GISPlatform, in: Sixth International Conference on Computer Information Science and Application Technology (CISAT 2023) vol. 12800, SPIE, 2023, pp. 520–527, https://doi.org/10.1117/12.3003864.

[46] X. Yu, W. Luo, Reinforcement learning-based multi-strategy cuckoo search algorithm for 3D UAV path planning, Expert Syst. Appl. 223 (2023) 119910, https://doi.org/10.1016/j.eswa.2023.119910.

[47] S. Garrido, L. Moreno, M. Abderrahim, F. Martin, Path planning for mobile robot navigation using voronoi diagram and fast marching, in: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2006, pp. 2376–2381, https://doi.org/10.1109/iros.2006.282649.

[48] M. Radmanesh, M. Kumar, Flight formation of UAVs in presence of moving obstacles using fast-dynamic mixed integer linear programming, Aerosp. Sci. Technol. 50 (2016) 149–160, https://doi.org/10.1016/j.ast.2015.12.021.

[49] P. Bhattacharya, M.L. Gavrilova, Roadmap-based path planning-using the voronoi diagram for a clearance-based shortest path, IEEE Robot. Autom. Mag. 15 (2) (2008) 58–66, https://doi.org/10.1109/mra.2008.921540.

[50] X. Chen, X. Chen, U.A.V. The, dynamic path planning algorithm research based on Voronoi diagram, in: The 26th Chinese Control and Decision Conference (2014 CCDC), IEEE, 2014, pp. 1069–1071, https://doi.org/10.1109/ccdc.2014.6852323.

[51] J. Chang, N. Dong, D. Li, W.H. Ip, K.L. Yung, Skeleton extraction and greedy-algorithm-based path planning and its application in UAV trajectory tracking, IEEE Trans. Aerosp. Electron. Syst. 58 (6) (2022) 4953–4964, https://doi.org/10.1109/taes.2022.3198925.

[52] M. Thoresen, N.H. Nielsen, K. Mathiassen, K.Y. Pettersen, Path planning for UGVs based on traversability hybrid a, IEEE Robot. Autom. Lett. 6 (2) (2021) 1216–1223, https://doi.org/10.1109/lra.2021.3056028.

[53] J. Chen, J. Kneis, S. Lu, D. Mölle, S. Richter, P. Rossmanith, et al., Randomized divide-and-conquer: improved path, matching, and packing algorithms, SIAM J. Comput. 38 (6) (2009) 2526–2547, https://doi.org/10.1137/080716475.

[54] C. Zammit, E.-J. Van Kampen, Comparison between a* and RRT algorithms for 3D UAV path planning, Unmanned Syst. 10 (02) (2022) 129–146, https://doi.org/10.1142/s2301385022500078.

[55] R. Song, Y. Liu, R. Bucknall, Smoothed a* algorithm for practical unmanned surface vehicle path planning, Appl. Ocean Res. 83 (2019) 9–20, https://doi.org/10.1016/j.apor.2018.12.001.

[56] A. Mardani, M. Chiaberge, P. Giaccone, Communication-aware UAV path planning, IEEE Access 7 (2019) 52609–52621, https://doi.org/10.1109/access.2019.2911018.

[57] V. Roberge, M. Tarbouchi, G. Labonté, Fast genetic algorithm path planner for fixed-wing military UAV using GPU, IEEE Trans. Aerosp. Electron. Syst. 54 (5) (2018) 2105–2117, https://doi.org/10.1109/taes.2018.2807558.

[58] Y.V. Pehlivanoglu, P. Pehlivanoglu, An enhanced genetic algorithm for path planning of autonomous UAV in target coverage problems, Appl. Soft Comput. 112 (2021) 107796, https://doi.org/10.1016/j.asoc.2021.107796.

[59] M. Shafiq, Z.A. Ali, A. Israr, E.H. Alkhammash, M. Hadjouni, J.J. Jussila, Convergence analysis of path planning of multi-UAVs using max-min ant colony optimization approach, Sensors 22 (14) (2022) 5395, https://doi.org/10.3390/s22145395.

[60] J. Zhang, Y. An, J. Cao, S. Ouyang, L. Wang, UAV trajectory planning for complex open storage environments based on an improved RRT algorithm, IEEE Access 11 (2023) 23189–23204, https://doi.org/10.1109/access.2023.3252018.

[61] Q. Diao, J. Zhang, M. Liu, J. Yang, A disaster relief UAV path planning based on APF-IRRT* fusion algorithm, Drones 7 (5) (2023) 323, https://doi.org/10.3390/drones7050323.

[62] H. Duan, Y. Yu, X. Zhang, S. Shao, Three-dimension path planning for UCAV using hybrid meta-heuristic ACO-DE algorithm, Simul. Model. Pract. Theory 18 (8) (2010) 1104–1115, https://doi.org/10.1016/j.simpat.2009.10.006.

[63] S. Ivić, B. Crnković, L. Grbčić, L. Matleković, Multi-UAV trajectory planning for 3D visual inspection of complex structures, Autom. Constr. 147 (2023) 104709, https://doi.org/10.1016/j.autcon.2022.104709.

[64] R. Alyassi, M. Khonji, A. Karapetyan, S.C.-K. Chau, K. Elbassioni, C.-M. Tseng, Autonomous recharging and flight mission planning for battery-operated autonomous drones, IEEE Trans. Autom. Sci. Eng. 20 (2) (2022) 1034–1046, https://doi.org/10.1109/tase.2022.3175565.

[65] B. Triggs, P.F. McLauchlan, R.I. Hartley, A.W. Fitzgibbon, Bundle adjustment—a modern synthesis, in: Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings, Springer, 2000, pp. 298–372, https://doi.org/10.1007/3-540-44480-7_21.

[66] Y. Tan, L. Chen, M. Huang, J. Li, G. Zhang, Automated geometric quality inspection for modular boxes using BIM and LiDAR, Autom. Constr. 164 (2024) 105474, https://doi.org/10.1016/j.autcon.2024.105474.

[67] A.A. Diakite, S. Zlatanova, Automatic geo-referencing of BIM in GIS environments using building footprints, Comput. Environ. Urban. Syst. 80 (2020) 101453, https://doi.org/10.1016/j.compenvurbsys.2019.101453.