

RESEARCH ARTICLE

FLIE: First-Look Enabled Inspect-Explore Autonomy Toward Visual Inspection of Unknown Distributed and Discontinuous Structures

VIGNESH KOTTAYAM VISWANATHAN^{1b}, (Student Member, IEEE),

SUMEET GAJANAN SATPUTE^{1b}, (Member, IEEE),

AND GEORGE NIKOLAKOPOULOS^{1b}, (Member, IEEE)

Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, 97187 Luleå, Sweden

Corresponding author: Vignesh Kottayam Viswanathan (vigkot@ltu.se)

This work was supported in part by the illuMINEation Horizon 2020 Project through the European Union's Horizon 2020 Research and Innovation Program under Grant 869379.

ABSTRACT In this article, the problem of an online autonomous aerial inspection, specifically for discontinuous and distributed objects is presented. The proposed approach imposes view culling and photogrammetric constraints based on a geometrically modeled three-dimensional view pyramid, a view cone to filter surfaces by desired observation angle, a framework-integrated passive collision-avoidance scheme with the object under inspection, and a dynamically enveloping bounding-box region to map the visited surfaces. Furthermore, the proposed inspect-explore framework is validated for the case of an unknown environment with no prior knowledge of the object model under inspection. The overall inspection scheme is based on the novel *First-Look* approach, enabling the UAV to progressively adapt its inspection path to match the profile of the structure autonomously. The implemented exploration strategy imposes a tiered policy enabling the UAV to search, identify and navigate towards the structure for inspection. The presented work utilizes a unified architecture of the aforementioned inspect-explore framework to improve situational awareness in a previously unknown environment by enabling the UAV to explore its surrounding space and identify structures to execute closer inspection tasks. Extended simulations to evaluate the efficacy of the proposed inspect-explore framework are presented with multiple structure scenarios.

INDEX TERMS Visual inspection, aerial robotics, distributed objects, first-look, online planning, unknown environment.

I. INTRODUCTION

The rapid rise of the scope of on-board autonomy to enable deployment of Unmanned Aerial Vehicles (UAVs) towards fulfilling labour and time-intensive tasks, such as inspection of ageing infrastructures [1], [2], [3], surveying underground mines [4], [5], [6], in search and rescue operations [7], [8] has received a lot of attention since the past few years.

On the topic of visual inspection using UAV(s), planning the inspection path is driven by the factors, such as ensuring reaching the view-points satisfactory, obtaining the required resolution of imagery data, identifying the presence of surface

defects, maintaining photogrammetric constraints, such as overlap percentage between successive images to enable a rich 3D accurate reconstructed mesh of the structure. In addition, generating an overall collision-free and less resource-demanding coverage path is one of the important factors in inspection planning. While the task of visual inspection of a known solitary object is inherently challenging, based on the fulfilment of the aforementioned conditions, the level of complexity increases when considering the inspection of unknown multiple distributed and discontinuous structures. The field of rescue robotics, such as in search and rescue operations of high-rise structures is primarily model-based due to the availability of Building Information Modelling (BIM) of the structure being inspected. However, a research niche

The associate editor coordinating the review of this manuscript and approving it for publication was Wai-Keung Fung^{1b}.

arises when such information cannot be used to plan for an inspection of a partially damaged or fractured structure as its current state does not correspond to the structural information available at hand. The need to integrate and implement a hybridized approach to inspection-exploration is required at this stage. Hence, the proposed work in this article aims to present a proof-of-concept for the application of hybridized autonomy towards improving inspection outcomes with the desired situational awareness from the point of deployment, as a vital component during emergency situations.

Figure 1 presents the overall proposed concept pertaining towards the deployment scenario of the proposed FLIE autonomy framework.

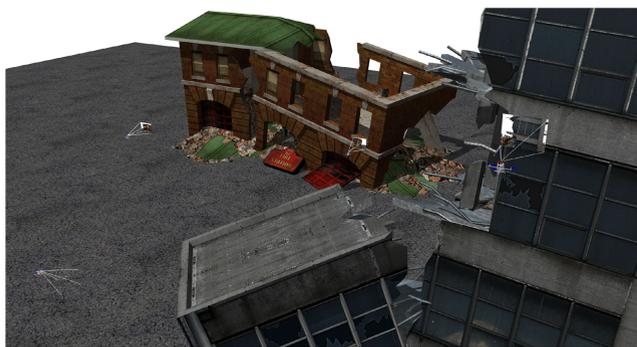


FIGURE 1. Conceptual depiction of the proposed FLIE autonomy towards implementation in a scenario consisting of distributed and discontinuous structures.

In view of the related works towards inspecting unknown structures, [9] presents a Frontier based Next-Best-View (NBV) online inspection scheme to observe a local surface, based on the desired threshold of information gained in an unknown environment. The presented contribution builds a volumetric map and implements sampling-based optimization to iteratively plan the optimal path to visit the desired sequence of view configurations, which provides maximum coverage. Towards a unified exploration and inspection in an unknown environment, authors in [10] built on the initial Next-Best View (NBV) determination approach by [11], which evaluates the performance of an online sampling-based receding-horizon NBV method that iteratively provides an optimal sensor configuration to reduce unmapped surfaces or spaces. Similarly, in [12], the authors present a unified explore-inspection architecture in constrained environments, such as the water ballast tanks of ships. This work combines *RRT*-nodes, which provide a collision-safe path to visit unexplored regions, with a region-based convolutional neural network, which identifies regions of decay in the structure. In [13], the authors present an online coverage path planner for a bridge, which provides sensor configurations based on the surface vector obtained from identified nearest set of points from the *k*-nearest neighbours (*K*-NN) search. A real-time heuristic-based dynamic optimization of the coverage path to perform a visual inspection, for both online and offline fashion, has been presented in [14]. A Weighted Gain

Next-Best-View Planner (WG-NBVP) is presented in [15], to inspect a hazardous and contaminated environment with the help of an Unmanned Ground Vehicle (UGV). This approach tackles the dual task of obtaining a volumetric map of the target Region-Of-Interest (ROI) in an unknown environment and inspecting the relevant areas of interest by means of optimization for a maximum weighted information gain, which considers free space covered, measurement readings and visited space at the candidate camera pose. This pose is derived from Rapidly Exploring Random Tree (*RRT*) to provide the nearest optimal pose contingent on a selection threshold parameter.

Compared to previous works on unified approaches, where voxel maps of the unknown environment are generated from sensor data subsequent to which the inspection planner plans a path to observe the surfaces, the proposed methodology in this article operates in a frame-by-frame manner, where the sensor input, in the form of pointcloud is directly used to plan the next immediate view-pose and provide visual coverage of the observed surface. Moreover, the inspection planner simultaneously generates and updates the bounded shape of the observed surface, thereby providing a 3D shape of inspected structures at the end of the mission.

The novelty of the proposed approach stems from the inspection framework, the First-Look approach, presented in [16]. In our previous work, the First-Look approach enables the inspecting UAV to quickly adapt its inspection path to the profile of the surface being inspected subject to photogrammetric and collision-avoidance constraints. In the proposed work, we present a synergetic framework incorporating the recursive inspection methodology along with an exploration policy for close visual inspection of fractured and distributed structures. The exploration module operates in a tiered fashion to explore the vicinity around the UAV to identify and provide locations of unvisited surfaces lying within the camera field-of-view. This is extremely useful in scenarios where visual inspection of structurally damaged objects is necessary to maximize situational awareness, especially in areas where prior knowledge is not applicable or available after disastrous events. Moreover, the proposed work enforces a view-planning policy on instantaneous sensor input, thereby decoupling the need to build a global map of the operating environment and offsetting the high memory overhead needed to store volumetric maps. The recursive nature, of the view-planning process, ensures a low-complexity approach to enable information-guided inspection and exploration in an unknown environment in addition to being robust against variations in positioning, as a direct result of the view-planning policy on instantaneous measurements rather than on a progressively built map.

Thus, the main contributions of this article can be summarized as follows: (a) A novel unified map-independent inspect-explore autonomy composed of an improved First-Look based inspection strategy with 3D view-space culling to consider only points detected within the sensor field-of-view coupled with a view-cone to ensure surfaces being viewed

maintain sufficient viewing angle as per user specification and a hierarchical exploration policy to enable the UAV to visually locate, identify and navigate towards to execute the required inspection task, (b) An integrated passive collision-avoidance scheme that adjusts and maintains dynamically the desired viewing distance based off the local viewing surface thus adapting the inspection path to the profile of the structure while ensuring safety. (c) A multi-tiered search policy to address discontinuities in surface ensuring effective resumption of inspection task for fractured objects in addition to locating the presence of structures within the vicinity of the current structure dependent entirely on visual feedback. (d) Evaluation of the performance of the framework against multiple forms of simulated distributed and discontinuous objects.

The remainder of the article is structured as follows: Section II provides a mathematical formulation of the problem addressed in this work, and Section III presents a detailed description of the proposed solution. The controller architecture implemented in this work is presented in Section IV. The implemented simulation setup is presented in Section V along with the obtained results paired with the corresponding detailed analysis in Section VI. Section VII presents information on potential limitations of the proposed inspect-explore autonomy. Finally, Section VIII concludes the article with a comprehensive discussion of the performance of the proposed solution.

II. PROBLEM STATEMENT

Given a three-dimensional object, which is registered by the framework as a transformed set of pointcloud $\gamma_{x,y,z} \in \mathbb{R}^3$. Let the pose of the UAV be represented by $\xi = (p_{uav}, \psi) \in \mathbb{R}^4$ where $p_{uav} \in \mathbb{R}^3$ be the position and $\psi \in R$ be the yaw orientation of the UAV. As the optical sensor is mounted on the front of the UAV, having a view direction along the X -axis of the UAV, the roll and pitch orientation can be excluded from consideration during view-planning. The UAV is assumed to be equipped with a visual sensor capable of providing depth pointcloud information $\rho \in \mathbb{R}^3$. The aim of the planning problem is to determine a series of viewpoints denoted by $(\xi_1, \xi_2, \dots, \xi_n)$, where $n \in \mathbb{N}$ that provides complete visual coverage of the target structure. The addressed problem can be defined as finding the set of required view poses $\{\xi_n \mid n \in \mathbb{N}, \gamma \in \mathcal{A}\}$, where \mathcal{A} represents the bounding region of the registered pointclouds, that provides complete coverage of structures located in and around the deployment zone.

III. FIRST-LOOK INSPECT-EXPLORE FRAMEWORK

To establish situational awareness in an unknown environment, the proposed approach plans the next inspection view pose ξ_{n+1} based on the current sensor input ρ and the view pose ξ_n of the UAV subjected to photogrammetric constraints in addition to executing a hierarchical exploration strategy in the event loss of viewable surface or to identify and navigate

towards the next candidate structure located in the vicinity of the UAV. Figure 2 presents a graphical depiction of the proposed autonomous aerial inspection framework for distributed and discontinuous objects. A detailed explanation of the presented modules is given in the following sections.

A. INSPECTION MODULE

The inspection module comprises of a collection of three sub-modules mainly: (1) View-frame culling where a 3D view-space is designed to allow culling of points outside the sensor's field-of-view and to determine the viewable regions of the detected surface based on the current view orientation of the UAV; (2) Determination of view orientation where a kd-tree based nearest neighbour search to determine the view-orientation to be maintained by the UAV to view the closest point of interest; and finally (3) Photogrammetric constraints where each successive view-points are projected subjected to the viewing constraints such as image overlap on both horizontal and vertical direction of travel during the inspection.

1) VIEW-FRAME CULLING

Figure 3 provides the graphical representation of the 3D view space considered in this work. Given the horizontal field of view ($fov_h \in \mathbb{R}$), the vertical field of view ($fov_v \in \mathbb{R}$) and the distance of the view plane ($d_{vp} \in \mathbb{R}^+$) of the modelled camera sensor, the view pyramid can be modelled as a collection of planes λ_i with $i \in [1 \dots 5]$ with corresponding unit normal vector $\vec{n}_i \in \mathbb{R}^3 \mid i \in [1 \dots 5]$, such that for any point $(\gamma_{(x,y,z)})$ to be viewed as within the 3D viewing volume, it satisfies the condition,

$$\gamma^L = \vec{n}_i \cdot (p_\gamma - p_{uav}^k) \geq 0 \quad \forall \gamma_{(x,y,z)} \quad (1)$$

where $\gamma^L \in \mathbb{R}^3 \mid \gamma^L \subseteq \gamma_{x,y,z}$ represents the locally viewable culled-set of point clouds, $p_\gamma \in \mathbb{R}^3$ is the position of the candidate pointcloud point and $p_{uav}^k \in \mathbb{R}^3$ denotes the position of the UAV at the k^{th} step. After obtaining γ^L for the current view orientation, the viewability of the generated surface patch Λ , represented in the form of a triangular mesh, from the current pose of the UAV is defined. This allows the framework to prioritize local surface regions that satisfy desired photogrammetric parameters, such as the incident viewing angle σ on the determined centroid $p_{centroid}^j \in \mathbb{R}^3$ with $j \in \mathbb{Z}^+$, of the j^{th} triangular mesh Λ_j . Let the unit normal vector from the centroid of the triangular mesh be defined as \vec{n}_{Λ_j} , then the valid surface patch can be identified by,

$$\Lambda_{valid} = \left| \arccos \left(\frac{\vec{n}_{\Lambda_j} \cdot (p_{centroid}^j - p_{uav}^k)}{|\vec{n}_{\Lambda_j}| |p_{centroid}^j - p_{uav}^k|} \right) \right| \leq \sigma \quad \forall \Lambda_j \quad (2)$$

where $\Lambda_{valid} \subseteq \Lambda$. By evaluating (2), from the bounds of σ we trace the exterior profile as that of a cone where the viewed faces lie within the circumscribing radius of the cone.

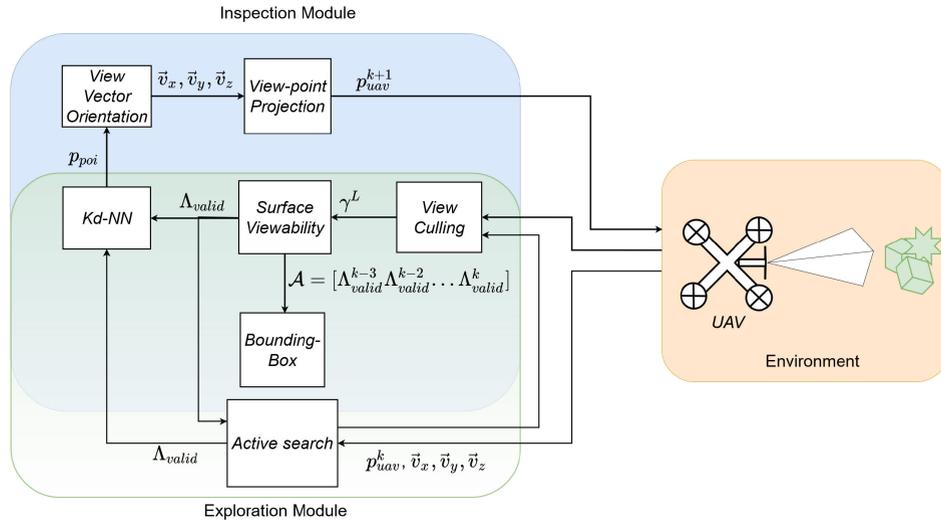


FIGURE 2. Graphical representation of the proposed inspect-explore framework.

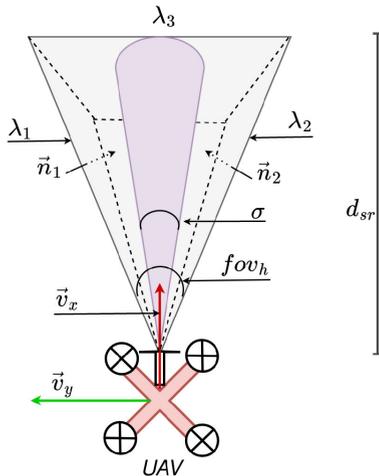


FIGURE 3. (Top-down-view) Visual representation of the 3D view space modelled in the proposed work.

2) DETERMINATION OF VIEW ORIENTATION

Initially presented in [17], the concept of representing multi-dimensional data in the form of a binary search tree has allowed much faster and more efficient organizing operations to be carried out. In this work, the k -dimensional tree (k -d tree) organizational structure is employed, in the multi-dimensional pointcloud γ^L , to primarily calculate the nearest available centroid, $p_{poi} \in \mathbb{R}^3$, given the current UAV position p_{uav} and observed patches from current view pose, Λ_{valid} . This enables the UAV to register a *look-at* vector $\vec{v}_{la} \in \mathbb{R}^3$ given as $\vec{v}_{la} = p_{poi} - p_{uav}$. Let \vec{v}_x be a normalized vector representing the view direction of the UAV obtained from \vec{v}_{la} ,

$$\vec{v}_x = \frac{\vec{v}_{la}}{\|\vec{v}_{la}\|} \quad (3)$$

Considering a UAV model following the *ROS* convention for the coordinate frame representation with X -axis pointing

forwards, Y -axis to the left and Z -axis completing the right-handed triad, let $\vec{n}_z \in \mathbb{R}^3$ be the upward pointing vector of the UAV along the positive direction of Z -axis given as $[0 \ 0 \ 1]^T$. The set of equations given in (3), (4) and (5) provide the required view vectors to be maintained by the UAV to observe the current local surface.

$$\vec{v}_y = \vec{n}_z \times \vec{v}_x \quad (4)$$

$$\vec{v}_z = \vec{v}_y \times \vec{v}_x \quad (5)$$

3) PHOTOGRAMMETRIC CONSTRAINTS

The requirement of photogrammetric constraints, while inspecting ensures enough matching features are available between successive images to perform accurate post-inspection 3D reconstruction from the acquired image database of the object under inspection. Utilizing the sensor specifications, mentioned in III-A1, the required overlap factor $\beta \in \mathbb{R}$, is given as a function of viewing distance between p_{uav} and p_{poi} and the horizontal field-of-view fov_h . Proceeding to model the required next UAV position at a distance (overlap distance), $d_{hov} \in \mathbb{R}$, from 2D sensor footprint,

$$d_{hov} = 2 \tan 0.5fov_h \|p_{poi} - p_{uav}^k\| - 2\beta_h \tan 0.5fov_h \|p_{poi} - p_{uav}^k\| \quad (6)$$

where β_h is the horizontal overlap factor. Fig. 4 graphically depicts the relation shared by the view-point (p_{uav}^k) at k^{th} step. Similarly from (6), with a minor change to the representation, the required distance $d_{vov} \in \mathbb{R}$ to maintain vertical overlap β_v is given as,

$$d_{vov} = 2 \tan 0.5fov_v \|p_{poi} - p_{uav}^k\| - 2\beta_v \tan 0.5fov_v \|p_{poi} - p_{uav}^k\| \quad (7)$$

With the applicable constraints defined, the core of the proposed inspection architecture is the sequential projection of

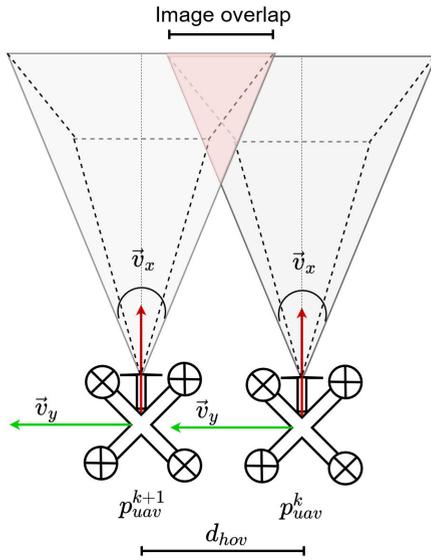


FIGURE 4. (Top-down-view) Pictorial depiction of the horizontal photogrammetric constraint considered in this work.

the view-point modelled to be dependent on the current view orientation of the UAV,

$$p_{uav}^{k+1} = p_{uav}^k + \vec{v}_y d_{hov} \quad (8)$$

where p_{uav}^k and p_{uav}^{k+1} defines the position of the uav at the k^{th} and $k + 1^{th}$ sequence. Equation (8) allows the UAV to iteratively adapt its view orientation to the availability of the p_{poi} within the 3D view space whilst being subject to the photogrammetric constraints. In order to prevent collision with the surface under observation and also to maintain a decent resolution of the image captured, an integrated passive collision-avoidance scheme is proposed. Thus (8) can be rewritten to include the safety distance $d_{safety} \in \mathbb{R}$ directed along the viewing direction as,

$$p_{uav}^{k+1} = p_{uav}^k + \vec{v}_y d_{hov} \pm f(d_{safety}, p_{poi}, p_{uav}^k, \vec{v}_x) \quad (9)$$

where,

$$f(d_{safety}, p_{poi}, p_{uav}^k) = \vec{v}_x (d_{safety} - \|p_{poi} - p_{uav}^k\|)$$

The set of visible centroids Λ_{valid} at p_{uav}^{k+1} is then found using (2) and (1). Using the properties of alphashapes [18], a bounding-box \mathcal{A} to contain the visited points Λ_{valid} at p_{uav}^k is created. This enables the progressive online update of, previously unknown regions, occupied by fractured objects.

To model the inspection logic to visit structures having a vertical height greater than the image width, (8) is restructured to reflect the desired vertical overlap distance as:

$$p_{uav}^{k+1} = p_{uav}^k + \vec{v}_z d_{vov} \quad (10)$$

During inspection, we perform culling of points lying outside the modelled view pyramid defined in Sect. III-A1 and denoted by **ViewSpaceCulling**(p_{uav}^k, \vec{v}_x). Once the observable patches Λ_{valid} are determined, the nearest centroid p_{poi} is

Algorithm 1 Inspection

```

1: k=1
2: while not inspected all do
3:    $\Lambda_{valid} \leftarrow \mathbf{ViewSpaceCulling}(p_{uav}^k, \vec{v}_x)$ 
4:    $\mathcal{A} \leftarrow \Lambda_{valid}$ 
5:    $p_{poi} \leftarrow \mathbf{SearchkdNN}(\Lambda_{valid}, p_{uav}^k)$ 
6:    $[\vec{v}_x \ \vec{v}_y \ \vec{v}_z] \leftarrow \mathbf{GetViewOrientation}(p_{poi}, p_{uav}^k)$ 
7:    $p_{uav}^{k+1} = p_{uav}^k + \vec{v}_y d_{hov}$ 
8:   if  $\emptyset \leftarrow \mathbf{ActiveSearch}$  then
9:     break
10:  else if  $\|p_{uav}^k - p_{uav}^0\| < d_{hov}$  and  $|\gamma^L \cap \mathcal{A}| > \beta_h |\gamma^L|$ 
11:    then
12:       $p_{uav}^{k+1} = p_{uav}^k + \vec{v}_z d_{vov}$ 
13:    else
14:      continue
15:  end if
16:  k = k + 1
17: end while

```

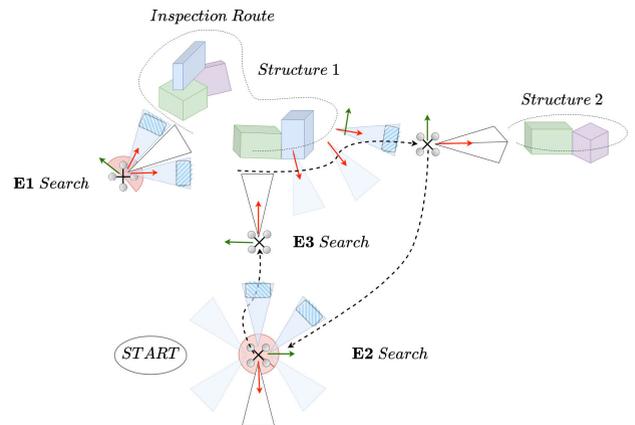


FIGURE 5. Graphical representation of the multi-tiered exploration strategy in the proposed FLIE framework.

found through **SearchkdNN**($\Lambda_{valid}, p_{uav}^k$) from the valid set of patches, which provides the required view direction from **GetViewOrientation**(p_{poi}, p_{uav}^k) defined in Sect. III-A2 to be maintained by the UAV, while inspecting.

The pseudo-code, Algorithm 1, for the inspection module explained in this subsection is given below.

B. EXPLORATION MODULE

Once the UAV finishes the inspection of the local structure, registers $max(\gamma_z^L) > p_{uav}^{k+1} z = \emptyset$ and determines at ξ_{curr} that $|\gamma^L \cap \mathcal{A}| > \beta_h |\gamma^L|$. The UAV’s mission is modelled to look for new unregistered surfaces from its current pose, where with the sensor specification on fov_h , knowledge on current view direction \vec{v}_x and the maximum desired sensor range $d_{sr} \in \mathbb{R}^+$, the UAV proceeds to execute a tiered exploration strategy to identify the presence of nearby structures. Figure. 5 presents the overview of the implemented tiered exploration strategy, while the process of searching for the next nearest p_{poi} is explained in this subsection.

The exploration behaviour of the UAV is modelled in three stages within the **Active Search: E1, E2** and **E3** stage respectively. While being engaged in the inspection of a structure, in order to address surface discontinuity in the form of gaps and absence of observable surface, the initial **E1** stage enables the UAV to search for available structures within $\pm 90^\circ$ from current viewing orientation. This is made possible by decomposing the view space around the UAV into sectors $n_{sectors} \in \mathbb{R}$ based on the onboard optical sensor's fov_h . If at the end of **E1**, $\Lambda_{valid} \rightarrow \phi$, i.e. no viewable surfaces exist at the current pose, the algorithm moves to execute the **E2** search policy. During the second stage, the behaviour is modelled to search 360° search space around the UAV following a similar method of decomposition as mentioned previously. This accounts for the presence of any structure within the immediate vicinity of the UAV that was not possible to observe during the initial **E1** search due to the observed view direction by the UAV. If any observable surfaces are found, the UAV continues with its inspection scheme otherwise if at the culmination of **E2** search, $\Lambda_{valid} \rightarrow \phi$, the algorithm proceeds to execute a travelling **E3** search based on backtracking through the past visited viewpoints with a 180° offset to view orientation that was maintained by the UAV at each corresponding backtracked viewpoints. This behaviour enables the UAV to comprehensively explore and inspect observable surfaces located around the current target structure under inspection.

Thus, the tiered exploration strategy account for the effective identification of available structures based on the limited sensing capabilities of the equipped sensors onboard the UAV. Algorithm 2 presents an overview of the proposed exploration architecture.

IV. CONTROLLER ARCHITECTURE

The controller architecture implemented follows the approach presented in [19] where the nonlinear model predictive controller (NMPC) is fed the reference view-points $\xi = [p_{uav} \ \psi]$ obtained from the FLIE module. The NMPC provides a corresponding control action containing commanded angle, thrust and yaw rate references $u = [\theta_{ref} \ \phi_{ref} \ T \ \dot{\psi}]$ based on the current estimated full state $\hat{x} = [p \ v \ \theta \ \phi \ \psi \ \omega]$ with the on-board odometry sensor providing the UAV position, velocity and current orientation. The reference control commands are then fed to the low-level controller onboard the UAV to be translated to motor speed commands $[n_1 \dots n_4]$. Figure 6 presents the overview of the controller architecture implemented in this work.

V. SIMULATION SETUP

The proposed work has been fully implemented in MATLAB and visualized in GAZEBO [20] over the Robot Operating System (ROS) network. The 3D CAD models of fractured structures were obtained from the GAZEBO models database. The boxes shown were modelled in Fusion 360. CloudCompare software was used to generate a down-sampled pointcloud of the candidate objects.

Algorithm 2 Exploration

```

1: Require  $\xi$ 
2: while explore flag do
3:    $i = 1$ 
4:    $\psi_i = \psi$ 
5:   Check if UAV is engaged in inspection
6:   if engaged then
7:      $n_{sectors} = \frac{\pi}{0.5fov_h}$ 
8:     for  $i \in n_{sectors}$  do
9:        $\psi_i = \psi_i \pm 0.5(fov_h)$ 
10:       $\Lambda_{valid} = \mathbf{E1}(p_{uav}^k, \psi_i)$ 
11:    end for
12:    return  $\xi_{E1}$  if not  $\phi \stackrel{\Lambda}{\leftarrow} \text{valid}$ 
13:  else
14:     $n_{sectors} = \frac{2\pi}{0.5fov_h}$ 
15:    for  $i \in n_{sectors}$  do
16:       $\psi_i = \psi_i \pm 0.5(fov_h)$ 
17:       $\Lambda_{valid} = \mathbf{E2}(p_{uav}^k, \psi_i)$ 
18:    end for
19:    return  $\xi_{E2}$  if not  $\phi \stackrel{\Lambda}{\leftarrow} \text{valid}$  else,
20:    for  $i \in \text{length}(\xi)$  do
21:       $\psi_{end-i} = \psi_{end-i} + \pi$ 
22:       $\Lambda_{valid} \leftarrow \mathbf{E3}(p_{uav}^{end-i}, \psi_{end-i})$ 
23:    end for
24:    return  $\xi_{E3}$  if not  $\phi \stackrel{\Lambda}{\leftarrow} \text{valid}$  else,
25:    return  $\phi$ 
26:  break
27: end if
28: end while

```

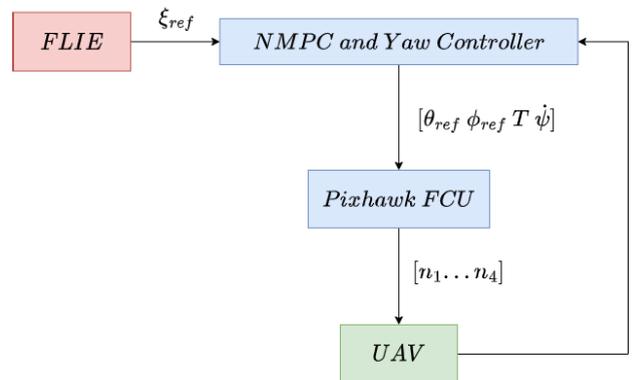


FIGURE 6. Graphical representation of the controller framework implemented in simulation.

Figure 7 presents the simulated 3D CAD model of a fractured and discontinuous object which is considered for an inspection-only scenario. The structure has a size of $20 \times 22 \times 24$ meters in length, width and height respectively. For the scenario of multiple distributed fractured structures, Fig. 8 shows the simulation setup in GAZEBO. The use of MAV aerial simulator RotorS [21] has been implemented in this work. A voxel grid filter has been utilized to present

the downsampled pointcloud for visualisation purposes in *GAZEBO*.

Table 1 provides the specifications of the optical sensor present onboard the UAV used in *MATLAB*. The UAV is assumed to be equipped with a stereo camera providing an input of depth cloud points, which is obtained through a view culling operation performed on the model pointcloud.

TABLE 1. Sensor specification on-board the UAV.

f_{ov_h}	69.4	degrees
f_{ov_v}	42.5	degrees
Image width	640	pixels
Image height	480	pixels
β_h	0.6	
β_v	0.5	
d_{sr}	10	m

For the simulation, we consider $\sigma = 0.8$ rad as the viewing angle constraint to inspect surfaces. The view poses obtained from *MATLAB* are transmitted over *ROS* to *GAZEBO* and fed to the NMPC controller. For odometry within *GAZEBO*, an in-built odometry plugin via the *RotorS* repository is utilized.

In order to ensure a smooth 3D reconstruction, from the captured RGB images during the mission, the UAV is commanded to rest for 0.5 s at each view-pose to obtain unblurred images of the locally viewed surface. This also ensures that the UAV exhibits a stable jerk-free motion as it passes through view-points throughout the mission.

VI. RESULTS AND DISCUSSION

Figure 9 provides the simulated result of the autonomous inspection framework targeting the presence of multiple sub-structures. In this simple scenario, the performance of the FLIE architecture to progressively inspect distributed objects, with spaces in between, can be seen generating view-points that expand and cover the multiple sub-structures. While along surfaces with a smooth profile, the framework can be seen adhering to the modelled photogrammetric constraints however, inspecting edges causes the framework to deviate as the presence of the nearest centroid, compared to the previous one, results in a change in viewing orientation of the UAV. Moreover, the framework can be seen generating view-points adhering to the modelled viewing distance d_{safety} of 1.5m.

In Figure 10, an *RVIZ* view of implemented FLIE framework towards the inspection of a single structure is presented. The UAV is initialized facing the structure in order to simulate an inspection of a singular object using FLIE autonomy. However, due to the implemented surface viewing constraints, the framework proceeds to **E2** search at the point of initialization of the mission to map observable surfaces. The *red* and *black* color paths denote the reference path generated by the planner and the actual path taken by the UAV. The *red* directional arrows represent the commanded view orientation to be maintained by the UAV during the

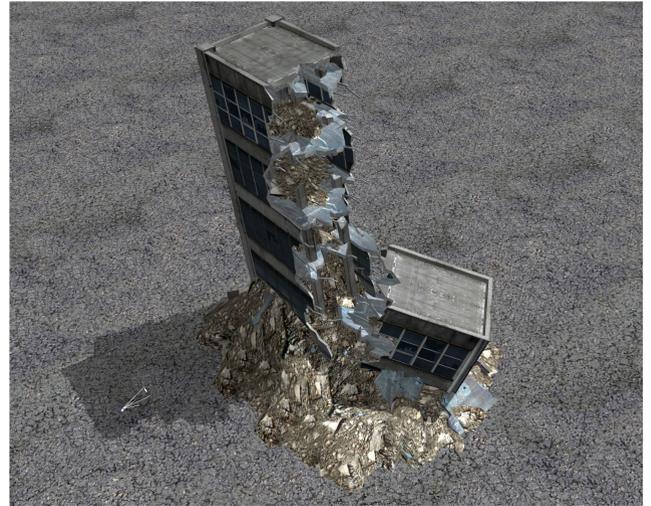


FIGURE 7. The model of a collapsed industrial building used to represent a fractured structure during the simulation of an inspection-only scenario.

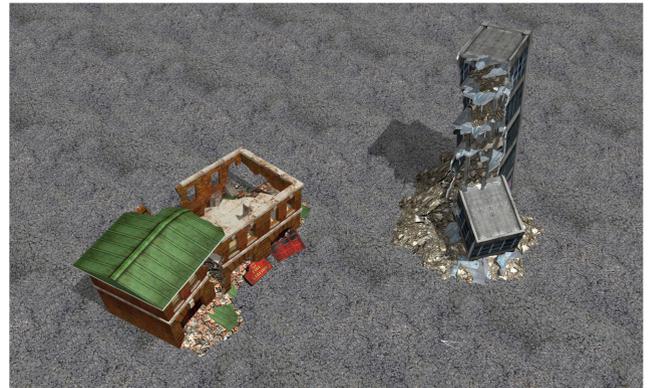


FIGURE 8. The simulation scenario of distributed and discontinuous structures in *Gazebo* for the proposed FLIE framework. On the left is a collapsed fire station and on the right is a collapsed industrial station.

inspection. An inspection distance of $d_{safety} = 5$ m has been utilized for this simulation scenario.

Figure 11 presents the 3-dimensional alphashape constructed from the surface points seen by the planner during simulation in *MATLAB*. The presence of holes in ill-mapped regions is one of the many challenges during 3D reconstruction. As it can be seen from the figure, the generated 3D mesh lacks holes in expected regions, such as surface discontinuity in the lower half of the building and in the upper floors. In Fig. 12, the performance of the controller towards maintaining the required yaw reference is shown. The yaw angles are bounded between $[-\pi \pi]$. As can be seen in Fig. 10, FLIE framework enforces a loop-by-loop inspection approach. Thus, the UAV primarily is commanded to move in a 2D plane while the inspection height is fixed for the current inspection loop. Figure. 13 provides the linear velocity of the UAV along *X* and *Y* axes. For the case of a single fractured high-rise structure, the UAV travels with an average velocity of 0.06 m/s along *X*-axis and 0.42 m/s

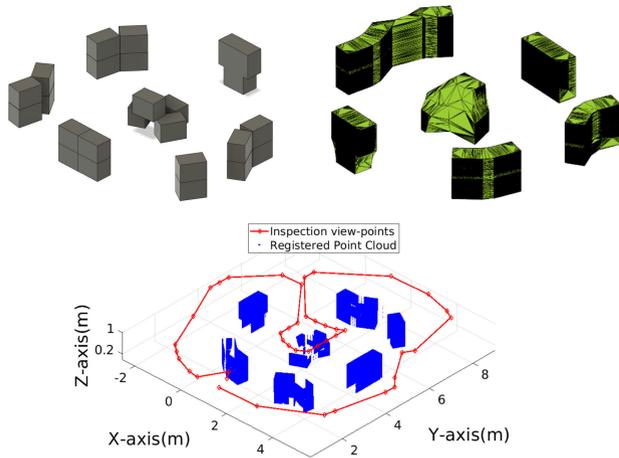


FIGURE 9. A demonstration of the proposed FLIE framework in a simple scenario where multiple boxes are stacked and are placed to represent a distributed and discontinuous structure. On the top-left is the 3D model of the boxes considered and on the bottom are the generated view points shown as red spheres along with the point cloud processed during the inspection and on the top-right is the constructed bounding volume of the observed pointcloud.

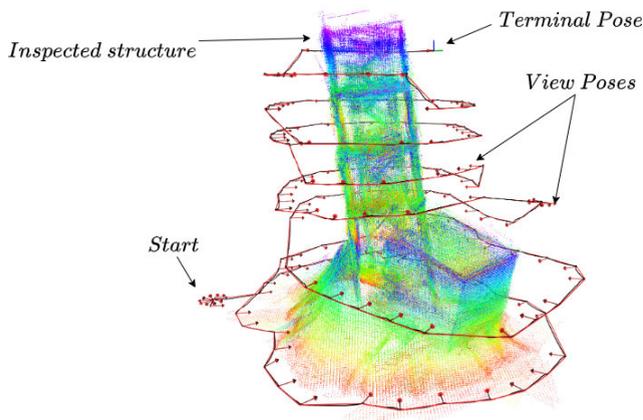


FIGURE 10. A RVIZ view of the simulated inspection carried out on the collapsed industrial building by the First-Look approach. The solid black line indicates the path tracked by the UAV during the inspection. The captured depth pointcloud from onboard *RealSense D435* sensor is visualized.

along *Y*-axis to reach the commanded view poses with a maximum recorded velocity of 1.4 m/s along both *X* and *Y* axes respectively. The relatively low average speed of the UAV is primarily due to the desired wait time of 0.5 s at each viewpoint.

Figure 14 presents the RVIZ view of the simulated scenario considering multiple distributed and fractured structures. Similar to Fig. 10, Fig. 14 shows that the framework remains robust to the varying profile of inspected structure and maintains the desired d_{safety} . In Fig. 15, the corresponding 3-dimensional alphashape of the observed surface points by the framework is shown. In Fig. 14, it is seen that the planner executes a repeated inspection at the same level of the structure in the top portion of the fractured industrial building. This is primarily due to the continued detection of unique

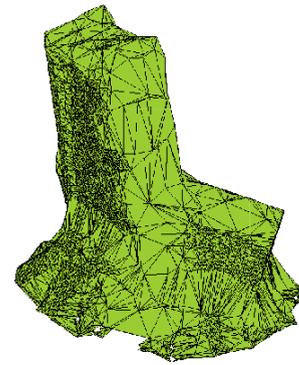


FIGURE 11. Graphical representation of the alpha shape reconstruction from utilized pointcloud set in the case on the inspection-only scenario of the high-rise structure.

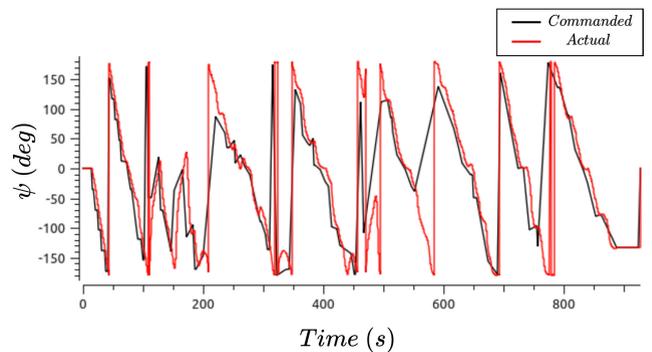


FIGURE 12. Graphical representation of the performance of reference tracking controller implemented in this work. In red, the actual yaw angle (in degrees) tracked by the UAV is shown along with the commanded yaw reference, shown in black, by the FLIE framework during the mission.

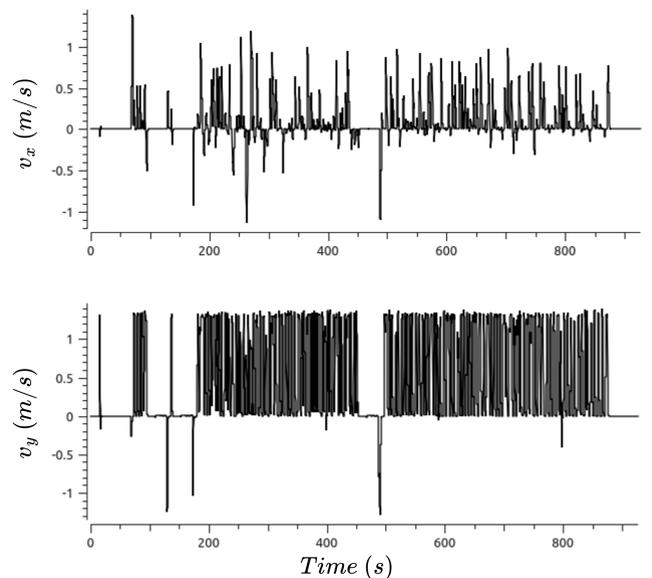


FIGURE 13. A combined plot of the translational velocities of the UAV along *X* and *Y* axes recorded during the mission is provided.

points representing the structure that is caused due to the UAV maintaining a different view orientation compared to the previous loop. While inspecting the building to the left

in Fig. 14, the planner can be seen to provide an overlapping path during close inspection towards the roof of the building. This behaviour also adheres to the explanation provided above since the vertical overlap condition is set to respond only when the similarity threshold of observed points crosses a threshold factor of 0.9. Therefore, the UAV continues the inspection of the structure until the condition is met. From the above simulated scenarios, a challenging situation arises especially in regions with large gaps due to reduced detection of points for the planner to continue inspecting the local surface. As a result, the implementation of a tiered search policy enables the planner to identify and navigate towards the next nearest surface to resume the current inspection task.

The proposed inspect-explore autonomy enforces a recursive view-planning policy. Given in (8), the policy projects the next view-point subject to the viewing direction and horizontal overlap to be maintained by the UAV. As such, it exhibits a behaviour similar to waypoint navigation and since the framework ensures successive view poses to adapt to the surface being observed, refer Fig. 10 and Fig. 14. Thus, the UAV remains safe from colliding with any projections/extensions from the structure. In addition to that, during E3-search, the UAV is tasked to backtrack through the previously visited view poses with a 180 deg offset in the commanded yaw orientation. Thus, an overall use of a path planner to determine the flight path of the UAV between current and commanded view-pose is avoided as the modelled nature of the algorithm ensures the UAV operates in a safe and stable manner. Quantification of inspection accuracy of apriori unknown structures is a challenging task. In this work, the authors have taken the approach to measure and present the performance of the proposed framework through a volumetric comparison between the 3D alphashape models generated from the pointcloud utilized by the planner and the actual pointcloud fed to the planner. For the case of a single high-rise structure, the framework exhibits the performance of 70.49% inspected volume compared to 82.04% obtained for the second scenario. The authors would like to stress that this result would vary depending on the actual density of the point cloud fed to the planner as a two-stage filtration of observed surface that leads to a reduced inspection of the inclined surfaces, such as the roof of the lower part of the fractured building not being considered seen by the planner.

In Fig. 16, the performance of the controller towards maintaining the required yaw reference, for the case of multiple distributed structures is provided. The inspection behaviour of the UAV can be similar to Fig. 12 until 900 s into the mission. This region corresponds to the UAV reaching the top of the structure under inspection. The execution of the E3-search results in the UAV exhibiting a flipped behaviour from 900 s till approximately around 1700 s when compared to the initial inspection run. The yaw angles are bounded between $[-\pi, \pi]$. Figure. 17 provides the linear velocity of the UAV along X and Y axes. The UAV travels with an average velocity of 0.03 m/s along X -axis and 0.37 m/s along Y -axis to reach the commanded view poses during the simulation.

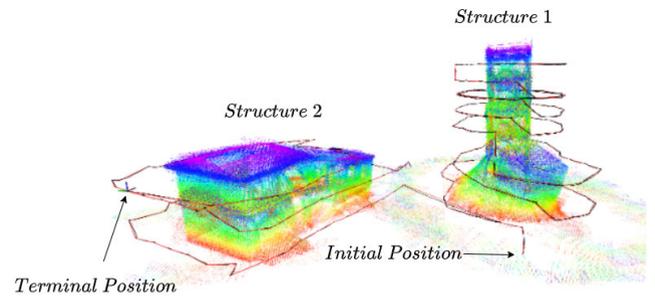


FIGURE 14. RVIZ view of the simulation scenario consisting of multiple distributed and discontinuous structures.

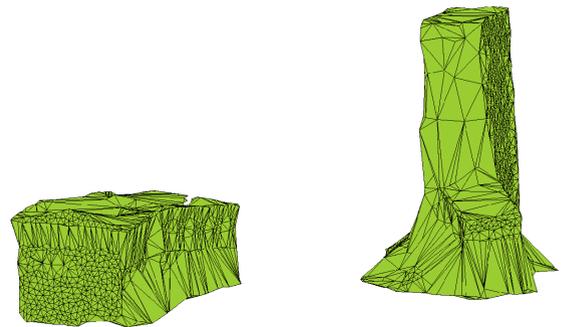


FIGURE 15. 3D alphashape generated from the set of observed surface points by the FLIE framework in MATLAB.

Moreover, a maximum velocity of 1.39 m/s and 1.43 m/s along X and Y axes were recorded during the simulation run.

Inferring from both Fig. 16 and Fig. 17 the proposed FLIE autonomy scheme can be seen adapting the required inspection behaviour based on the structure's surface profile. The inspect-explore behaviour for high-rise structure is gathered to last till approximately 1700 s subsequent to which the framework detects the secondary structure. Due to the increasing occurrences of surface discontinuity, present across the second structure, the UAV can be seen to present higher instances of switching behaviour between inspection and exploration, thus preventing premature termination of the mission.

The simulation video captured for both representative scenarios can be accessed on <https://youtu.be/6yEWISr4jyE>.

VII. LIMITATIONS

To the extent of the capability of the proposed work, the influence of localization errors on the performance of the framework remains relevant. Since the recursive view-planning policy during inspection is based on instantaneous sensor input rather than operating on a global map, the policy ensures the UAV remains robust to any accumulated drift due to positioning errors. However, such variations might affect the performance of the framework while backtracking through previously visited view-poses. In addition to that, aerial vehicles are noted to be affected by wind disturbances at higher altitudes and self-induced vibrations. In this work, the authors assume the absence of such perturbations and suggest the use

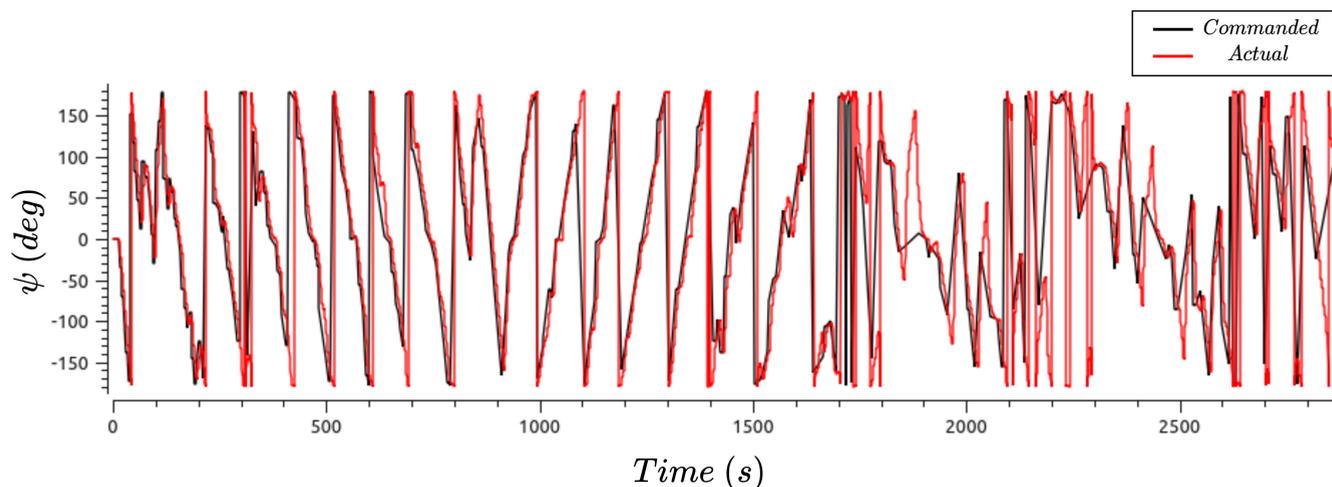


FIGURE 16. Graphical representation of the performance of the tracking controller implemented to follow the yaw reference generated by FLIE framework for the case of multiple distributed and fractured structures.

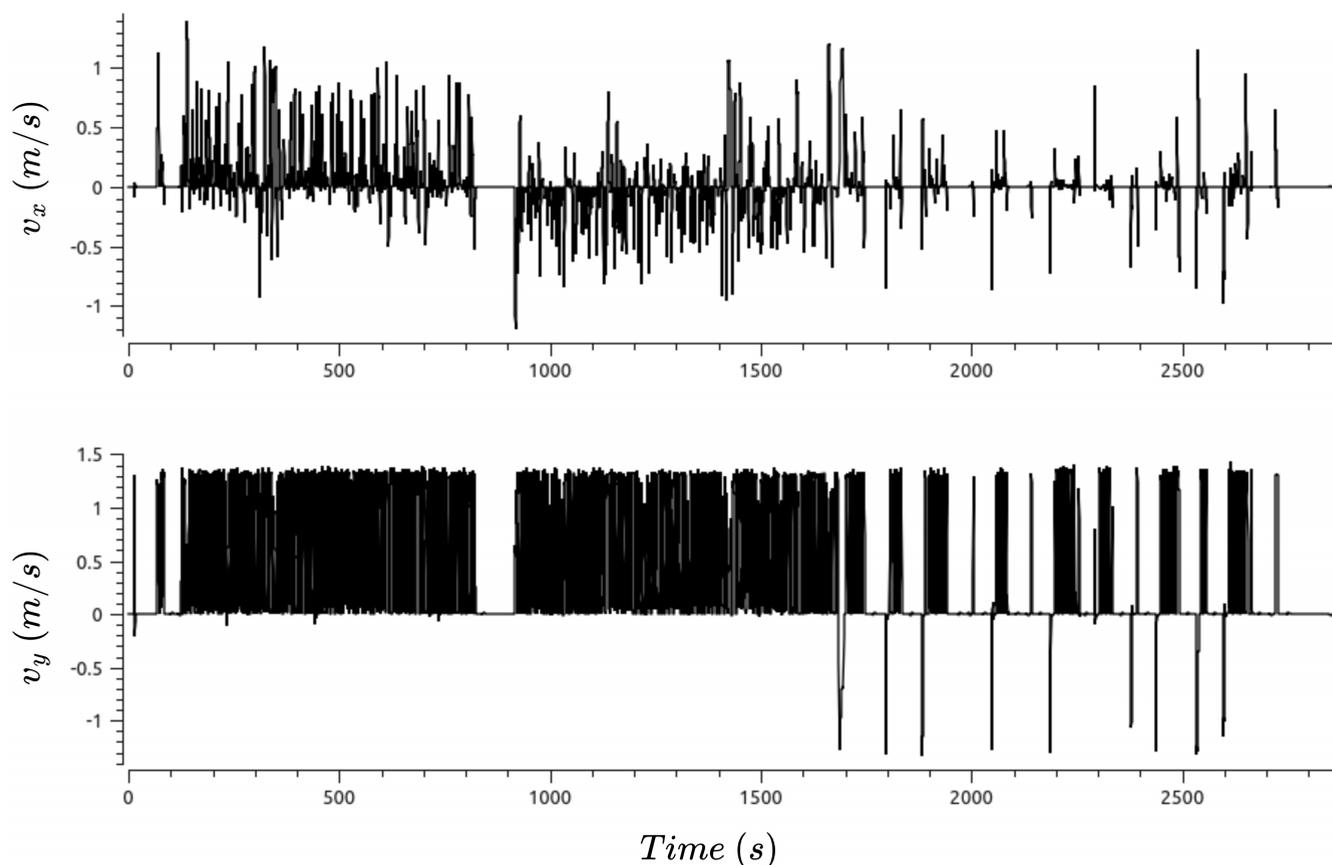


FIGURE 17. The recorded translational velocities of the UAV along X and Y axes obtained for the second scenario.

of a gimbal configuration to neutralize the disturbances to an acceptable level as part of future work.

VIII. CONCLUSION

This article proposes a novel online model-independent autonomous aerial inspection framework targeting its

application towards the inspection of distributed and discontinuous structures. The proposed framework, with an integrated passive collision-avoidance scheme, imposes view space culling to find surfaces that satisfy a predefined viewability criterion. In addition to that, photogrammetric constraints such as image overlap are taken into consideration

to enable an accurate 3D reconstruction. The presented inspection framework is based on the *First-Look* approach, defining view orientations on the basis of the nearest located centroid, which is based on *kd*-tree Nearest Neighbours. This enables the framework to adaptively orient the viewing vector to follow closely the profile of the surface being inspected. Furthermore, the inspection framework composes of an exploration module allowing the framework to search for new and unobserved surfaces present within the sensor range and near the vicinity of the UAV. The efficacy of the scheme has been evaluated in simulation for various arrangements of fractured objects.

REFERENCES

- [1] S. S. Mansouri, C. Kanellakis, E. Fresk, D. Kominiak, and G. Nikolakopoulos, "Cooperative coverage path planning for visual inspection," *Control Eng. Pract.*, vol. 74, pp. 118–131, May 2018.
- [2] Y. Tan, S. Li, H. Liu, P. Chen, and Z. Zhou, "Automatic inspection data collection of building surface based on BIM and UAV," *Autom. Construct.*, vol. 131, Nov. 2021, Art. no. 103881.
- [3] N. Ayoub and P. Schneider-Kamp, "Real-time on-board deep learning fault detection for autonomous UAV inspections," *Electronics*, vol. 10, no. 9, p. 1091, May 2021.
- [4] C. Kanellakis, S. S. Mansouri, G. Georgoulas, and G. Nikolakopoulos, "Towards autonomous surveying of underground mine using MAVs," in *Proc. Int. Conf. Robot. Alpe-Adria Danube Region*. Cham, Switzerland: Springer, 2018, pp. 173–180.
- [5] A. Agha et al., "NeBula: Quest for robotic autonomy in challenging environments; TEAM CoSTAR at the DARPA subterranean challenge," 2021, *arXiv:2103.11470*.
- [6] S. S. Mansouri, C. Kanellakis, D. Kominiak, and G. Nikolakopoulos, "Deploying MAVs for autonomous navigation in dark underground mine environments," *Robot. Auto. Syst.*, vol. 126, Apr. 2020, Art. no. 103472.
- [7] B. Lindqvist, C. Kanellakis, S. S. Mansouri, A.-A. Agha-mohammadi, and G. Nikolakopoulos, "COMPRO: A COMPact reactive autonomy framework for subterranean MAV based search-and-rescue operations," 2021, *arXiv:2108.13105*.
- [8] M. Petrlík, T. Báča, D. Heřt, M. Vrba, T. Krajník, and M. Saska, "A robust UAV system for operations in a constrained environment," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2169–2176, Apr. 2020.
- [9] S. Song and S. Jo, "Online inspection path planning for autonomous 3D modeling using a micro-aerial vehicle," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 6217–6224.
- [10] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon path planning for 3D exploration and surface inspection," *Auto. Robots*, vol. 42, no. 2, pp. 291–306, Feb. 2018.
- [11] C. Connolly, "The determination of next best views," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 2, Mar. 1985, pp. 432–435.
- [12] R. Y. Brogaard, R. E. Andersen, L. Kovac, M. Zajaczkowski, and E. Boukas, "Towards an autonomous, visual inspection-aware 3D exploration and mapping system for water ballast tanks of marine vessels," in *Proc. IEEE Int. Conf. Imag. Syst. Techn. (IST)*, Aug. 2021, pp. 1–6.
- [13] D. Choi, E. M. Lee, and H. Myung, "Online 3D coverage path planning using surface vector," in *Proc. 18th Int. Conf. Ubiquitous Robots (UR)*, Jul. 2021, pp. 392–396.
- [14] A. G. Melo, M. F. Pinto, A. L. M. Marcato, L. M. Honório, and F. O. Coelho, "Dynamic optimization and heuristics based online coverage path planning in 3D environment for UAVs," *Sensors*, vol. 21, no. 4, p. 1108, Feb. 2021.
- [15] M. Naazare, F. G. Rosas, and D. Schulz, "Online next-best-view planner for 3D-exploration and inspection with a mobile manipulator robot," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 3779–3786, Apr. 2022.
- [16] V. K. Viswanathan, S. G. Satpute, B. Lindqvist, and G. Nikolakopoulos, "First-look enabled autonomous aerial visual inspection of geometrically fractured objects in constrained environments," in *Proc. IEEE 31st Int. Symp. Ind. Electron. (ISIE)*, Jun. 2022, pp. 295–300.
- [17] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. ACM*, vol. 18, no. 9, pp. 509–517, 1975.
- [18] H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel, "On the shape of a set of points in the plane," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 4, pp. 551–559, Jul. 1983.
- [19] B. Lindqvist, J. Haluska, C. Kanellakis, and G. Nikolakopoulos, "An adaptive 3D artificial potential field for fail-safe UAV navigation," in *Proc. 30th Medit. Conf. Control Autom. (MED)*, Jun. 2022, pp. 362–367.
- [20] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 3, Oct. 2004, pp. 2149–2154.
- [21] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, *Robot Operating System (ROS): The Complete Reference*, vol. 1. Cham: Springer, 2016, pp. 595–625, doi: 10.1007/978-3-319-26054-9_23.



VIGNESH KOTTAYAM VISWANATHAN (Student Member, IEEE) received the master's degree in space science and engineering from the Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology (LTU), Kiruna, Sweden, where he is currently pursuing the Ph.D. degree with the Robotics and Artificial Intelligence Group, under the supervision of Prof. George Nikolakopoulos. His current research interest includes the planning and execution of visual inspection in both aerial and space applications agnostic of utilized platforms in addition to the autonomous deployment of multi-agent formations flying toward deep space exploration.



SUMEET GAJANAN SATPUTE (Member, IEEE) received master's degree in electrical engineering with a specialization in control systems from the Veermata Jijabai Technological Institute (VJTI), India, and the Ph.D. degree from the Onboard Space Systems Group, Luleå University of Technology (LTU), Luleå, Sweden. He is currently a Postdoctoral Researcher with the Robotics and Artificial Intelligence Group, LTU. His current research interests include multiple spacecraft formation, control and path planning problems, coverage and inspection of infrastructures, and autonomous planetary explorations with multiple agents.



GEORGE NIKOLAKOPOULOS (Member, IEEE) was with the NASA Jet Propulsion Laboratory, for conducting collaborative research on aerial planetary exploration, and participated in the DARPA Grand Challenge on Sub-T Exploration with the CoSTAR Team of JPL-NASA. He is currently a Chair Professor of robotics and artificial intelligence with the Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology (LTU), Sweden. He is also the Director of euRobotics and a member of the Scientific Council of ARTEMIS and IFAC TC on Robotics. He established the Digital Innovation Hub on Applied AI at LTU and also represents Sweden in the Technical Expert Group on Robotics and AI in the Project MIRAI 2.0 promoting collaboration between Sweden and Japan. His main research interests include field robotics, space autonomy, UAVs, automatic control applications, networked embedded controlled systems, wireless sensor and actuator networks, cyber-physical systems, and adaptive control.

...