

Collision-free local planner for unknown subterranean navigation

Sunggoo Jung¹  | Hanseob Lee² | David Hyunchul Shim² | Ali-akbar Agha-mohammadi³

¹Autonomous Flight Research Section, Electronics and Telecommunications Research Institute, Daejeon, Rep. of Korea

²Unmanned Systems Research Group, KAIST, Daejeon, Rep. of Korea

³JPL Robotics, NASA-JPL, California Institute of Technology, Pasadena, California, USA

Correspondence

Hanseob Lee, Unmanned Systems Research Group, KAIST, Daejeon, Rep. of Korea.

Email: hslee89@kaist.ac.kr

Funding information

Ministry of Science and ICT, South Korea, Grant/Award Number: 2017-0-00067

When operating in confined spaces or near obstacles, collision-free path planning is an essential requirement for autonomous exploration in unknown environments. This study presents an autonomous exploration technique using a carefully designed collision-free local planner. Using LiDAR range measurements, a local end-point selection method is designed, and the path is generated from the current position to the selected end-point. The generated path showed the consistent collision-free path in real-time by adopting the Euclidean signed distance field-based grid-search method. The results consistently demonstrated the safety and reliability of the proposed path-planning method. Real-world experiments are conducted in three different mines, demonstrating successful autonomous exploration flights in environment with various structural conditions. The results showed the high capability of the proposed flight autonomy framework for lightweight aerial robot systems. In addition, our drone performed an autonomous mission in the tunnel circuit competition (Phase 1) of the DARPA Subterranean Challenge.

KEY WORDS

Aerial autonomy, drone, GPS-denied navigation, path-planning, subterranean

1 | INTRODUCTION

Autonomous exploration in an unknown cluttered environment poses a set of challenging problems in many levels [1,2]. Unavailability of GPS signals and limited lighting conditions render the perception problem extremely difficult [3]. Confined spaces between walls and low height of priory unknown environment require compact drones which cannot rely on high-performance sensor-suites [4]. These constraints require careful attention in path-planning due to the existing state estimation errors. The path with hard constraints generates energy

optimal shortest path to the goal. However, it may bring about too close path to walls or obstacles, making quadrotor flying unreliable when there are control or perception errors. Besides, it requires a large amount of computation as the environmental information grows. Thus, real-time behavior decisions of the drones are not easy to realize in large-area exploration applications.

To solve these limitations, this study¹ proposes a three-step path generation procedure to make the

¹This paper describes one of the large systems of the COSTAR teamMore details are provided in [5].

generated path strongly remain in the middle of the obstacles or walls to safely explore the unknown confined areas. A grid or graph search method requires the destination. For unknown environment exploration, there are no goals or global plans. Thus, we use a frontier-based approach [6] and design a novel method that picks a receding-horizon local goal (we call it “end-point”) using occupancy grid information. A high-level collision-free path is developed based on grid-search algorithm [7]. To make the path more reliable to fly, we generate the nearest occupied voxel information represented by the euclidean signed distance field (ESDF) grid map and make the novel cost function of the grid-search method which penalizes the adjacent obstacle cost from ESDF information. Finally, we introduce a method to reduce the computational burden caused by the expansion of online maps as the exploration progresses. We employ the local-map approach which has $5\text{ m} \times 5\text{ m} \times 5\text{ m}$ fixed map size to overcome this problem. Figure 1 is an example snapshot of our method during the real-world experiment. The ESDF and local replanning paths are generated within this local-map area to reduce the computational cost and achieve real-time performance. These aspects confer fully autonomous and robust exploration capability to real-world robots. This study makes the following contributions:

- A collision-free local planning framework is designed for small-sized (330 mm in size) aerial robots with limited computation power and low payloads.
- An end-point selection method that rapidly generates collision-free local-goals in local ESDF maps is developed.
- The algorithm performance is compared to those of the state-of-the-art local planning methods.

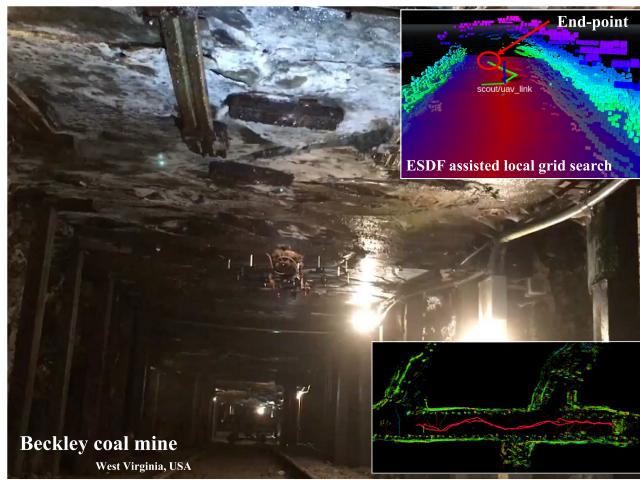


FIGURE 1 Snapshot of the autonomous exploration experiment of the lightweight drone in the coal mine environment. Location: Beckley mine, West Virginia, United States

- Extensive simulation and real-world tests validate the robustness of the framework.
- The code is publicly released to the community. It can run on any type of (for example, ground, aerial or underwater) robot for autonomous exploration applications².

The remainder of the paper is organized as follows. In Section 2, we present an overview of related works. Section 3 describes the hardware and software of the proposed framework. In Section 4, we present the details of the proposed approach. The simulations and practical implementation results are illustrated and evaluated in Section 5. Section 6 presents the conclusions and discussions.

2 | RELATED WORK

Considerable amount of methods have been investigated focusing on the problem of robotic exploration [6,8]. The path planning uses mixed integer linear programming (MILP) [9,10]. However, MILP is an NP-hard; thus, it can only be used in a finite-size planning problem. Model predictive control (MPC)-based receding-horizon navigation methods [11,12] and urban environment exploration conceptual works [13,14] were proposed; however, they are limited by the finite area range.

Early results in unknown area exploration planner were proposed by Bircher et al. [15]. This study showed the capability of priorly unknown area exploration while avoiding collisions. Kino-dynamic real-time planning using quadrotor has been presented in Allen and Pavone [16]. This study showed full-stack planning architecture; however, the state estimation is rely on the VICON motion capture system. Front-end path planning and back-end path smoothing works have been proposed in Gao et al. [17] and showed impressive results in a small range of indoor and outdoor areas. The MPC-based online path-planning work showed dynamic replanning by considering multi-object concept [18]; however, the study finished within simulation without real-world validation. One-line segment combination path smoothing approach is presented in Lai et al. [19]. This study generated the path using a two-point boundary value problem approach to reduce the computational cost. This study demonstrated reliable flight results; however, the piecewise segment of path joint optimization is not considered. Interesting works are presented by Popovic et al. [20]. They developed an online path-planning algorithm to reduce energy costs while detecting weeds. They showed

²Codes are available at our project github page: https://github.com/SunggooJung/local_planner_ROS

impressively reducing the energy cost by effective path-planning algorithm, however only validated in simulation.

Recently, the large-area exploration problem has attracted significant attention because of DARPA subterranean challenge [21]. Sampling-base method [15], graph-based approaches, [22,23] and learning-based approaches have been proposed [24]. The above systems require high size, weight and power (SWAP) systems making the robot relatively heavy and large. Thus, the development of a compact-sized exploration framework is an essential challenge to explore confined and cluttered spaces.

Some solutions to the above have been presented such as resilient micro flyer design [25] or range based sensor array approach [26]. These demonstrate simply penetrating a small-sized tunnel or structured area, which is impractical for large-area exploration.

Motivated by these studies, we discovered that the study for “lightweight autonomy” in confined and cluttered areas has not been properly investigated. This contribution aims to provide a meaningful low SWAP system alternative by combining both algorithms, the local end-point selection and the ESDF-assisted collision-free path, with fully on-board computing.

3 | SYSTEM OVERVIEW

In this study, which focuses on the collision-free navigation, a “350 mm in size” frame is designed. The designed concept of the system is lightweight but high performance with an acceptable flight time. The platform has four DJI Snail racing drone propulsion systems, which can lift all the sensor systems required for autonomous exploration in 12 min. We chose the PixRacer R15 autopilot with PX4 Stack for low-level flight control and NVIDIA Jetson TX2 GPU for high-level mission management. The TX2 module is mounted on a third-party carrier board (Auvidea, J120 [27]), which has RS-232 serial ports for communication with the PX4 flight controller.

With the constraint of the payload limit, the drone is equipped with a two-dimensional (2D) laser scanner (Hokuyo, UST-20LX) instead of three-dimensional (3D) LiDAR, such as the popular Velodyne VLP-16. It is worth noting that in the texture-less environment, the localization using only the laser scanner can be ineffective since the features on the walls are monotonous. Thus we integrate an upward-facing Qualcomm Snapdragon Flight (QSF), which delivers on-board visual-inertial-odometry and its fisheye camera data for localization robustness. For altitude measurement, time-of-flight based 1D LiDAR is equipped (Terabee, TeraRanger-Evo).

To aid and offer robust perception to vision source in the dark environment, we integrate an upward-facing high-power LED (Lume, Cube 2.0, housing case uncovered). The above sensor data are run on the NVIDIA Jetson TX2 (Ubuntu 18.04 variant and ROS Melodic). The total weight of the system, including all the sensing and processing components, and the battery is 1.43 kg.

4 | PROPOSED APPROACH

The proposed lightweight flight autonomy method consists of three collaborative modules: ESDF-assisted path planning, local-map generation, and end-point selection. The structure of this collaborative module setup is shown in Figure 2. First, the proposed method generates occupancy representation [28] and calculates ESDF, which allows collision-free paths. The planning module exploits the ESDF cost using the grid-search method, offering obstacle-aware safety paths. Then, to reduce the computational cost and ensure real-time performance for computing cost-constrained lightweight drones, every iteration of the path-planning generates the trajectory within $5 \text{ m} \times 5 \text{ m} \times 5 \text{ m}$ local-map around the current robot location. Finally, the proposed method introduces the end-point selection to make the planner generates trajectory within the local-map. As the exploration progresses, the occupancy information around the robot location changes. If the end-point, the destination of the path planner, is at the obstacle, the correct path cannot be created. Thus, the occupancy data and pointcloud sensor information is used to select the obstacle-free end-point. This section discusses these methods with respect to the lightweight flight autonomy concept.

4.1 | Algorithm description

Algorithm 1 outlines path-planning steps of the proposed framework. Every iteration starts by acquiring an occupancy probability map ($M_{\text{prob}}(\cdot)$) from the simultaneous localization and mapping (SLAM) module.

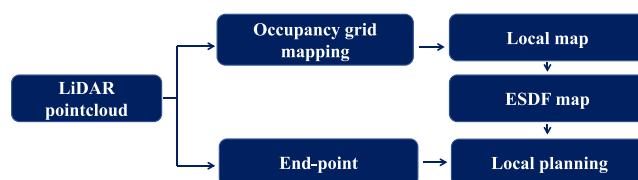


FIGURE 2 Diagram of lightweight autonomy framework. An ESDF map is developed using the mapped knowledge. ESDF-assisted planner determines good exploration paths starting at the current location to the end-point

From the certain threshold condition (T_ζ), $M_{\text{prob}}(\cdot)$ is converted to Boolean-valued voxel grid, marking flyable area by value of one (true) and fill the rest area with zero (false) ($M_{\text{bool}}(\cdot)$). M_{bool} is used to generate an ESDF map representing distance information from the adjacent obstacle. The local end-point selection module receives current state ($\xi = [x, y, z, \psi]$), LiDAR sensor data (σ_k). This module find best end-point b_k^* from the end-point candidate S_k . Then, the b_k^* passes a low-pass filter to prevent oscillation of the selected point. Finally, the planner receives the current state, selected end-point and ESDF info to generate a collision-free trajectory. It uses cost modified ESDF-assisted grid-search. The key functionalities are explained in detail in the following subsections.

Algorithm 1 ESDF-assisted Path

```

1: function BooleanVoxelGrid( $M_{\text{prob}}(\mathbf{p})$ )
   for each grid point  $\mathbf{p}$ , in  $M_{\text{prob}}(\mathbf{p})$ 
      if  $M_{\text{prob}}(\mathbf{p}) > T_\zeta$ 
          $M_{\text{bool}}(\mathbf{p}) \leftarrow \text{true}$ 
2: function ESDT( $M_{\text{bool}}(\cdot)$ ) return ESDF
3: function FindEndPoint( $\xi, \sigma_k, \psi_k$ )
   if  $\sigma_k \geq \text{inf}$ 
       $S_k \leftarrow 5 \text{ m}$ 
   else if  $4 \text{ m} \leq \sigma_k \leq 15 \text{ m}$ 
       $S_k \leftarrow 4 \text{ m}$ 
   else  $S_k \leftarrow \sigma_k$ 
   findBkStar( $S_k, \psi_k$ )
   lowPassFilter( $b_k^*$ , freq)
4: function Planner( $\xi, b_{k_{lpf}}^*, \text{ESDF}$ )
   function InitializePathFinder( $\xi, b_{k_{lpf}}^*$ )
   function FindPath( $\xi, b_{k_{lpf}}^*, \text{ESDF}$ )
      convert pose  $\xi$  to grid node  $n$ 
      find a path using grid-search
      where,  $f_{\text{total}}(n) = f_{\text{score}}(n) + f_{\text{dist}}(n)$ 
   return path

```

4.2 | Local map approach

The computational complexity of the grid or graph search-based planner depends on the map resolution and size. In the fixed-horizon path-planning problem, the user usually sets the admissible map size prior to ensuring the planner performance. The unknown area exploration problem, the area to search, is not known. Thus, as the exploration goes, the computational complexity grows in a cube of grid size ($O(n^3)$, n = grid size), which is impractical for a low-cost embedded on-board

processor. The proposed local-map approach is designed to reduce the computational cost by performing local planning in local-map area [sL, sW, sH] and resolution r . Thus, the time complexity remains fixed $O(sLsWsH/r^3)$ since the local-map maintains a constant size regardless of the exploration progress. The complexity only depends on the volume of the local-map when the map resolution is fixed. The local-map is updated at every 0.1 s (10 Hz); thus, this method can assure the vehicle velocity of about 50 m/s.

4.3 | End-point selection

As discussed above, when exploring areas fully unknown *a priori*, to use search-based planning method, the destination must be defined in advance. Therefore, the “end-point,” the destination in local-map area must be determined based on the obstacle information from local-map window. To resolve this problem, we implement the end-point selection method using the range measurements for search-based planning in the unknown area. The range sensor gives the index and distance of the point denoted as $[k, \sigma_k]$. To select the end-point, we sample evenly 48 points out of 1440 point clouds from the laser scanner range sensor. The samples are at 30 intervals starting from the 1st index of the points among a total of 1440 raw points of laser scanner inputs (see Figure 3). This constitutes a set of 48 sample points surrounding the vehicle at 240 degrees (the sample field of view Ω). Next, we collect a set of points exceeding a certain distance l . Here, “ l ” is the design factor, meaning the planning range of the local planner. For instance, if l is set to 5 m, the local planner plans the path 5 m from current vehicle pose ξ . Then, the method determines the maximum index of this set and converts the pose of its points into a body frame coordinate system. Finally, the point is selected as the end-point (2).

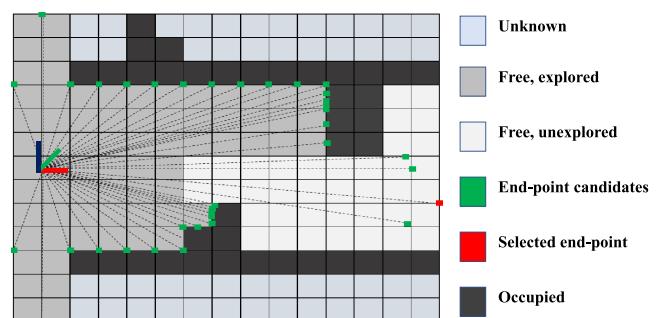


FIGURE 3 Example illustration of the end-point selection using Equations (1) and (2)

$$S_k = \begin{cases} l & \text{if } \sigma_k = \infty, \\ l-1 & \text{if } l-1 \leq \sigma_k \leq l_{\max}, \\ \sigma_k & \text{otherwise,} \end{cases} \quad (1)$$

$$S^* = \max_{k \in N} \xi_k + S_k \cdot \left(\frac{\cos(\psi_k + k \cdot \frac{\Omega}{N} - \frac{\Omega}{2})}{\sin(\psi_k + k \cdot \frac{\Omega}{N} - \frac{\Omega}{2})} \right), \quad (2)$$

where the sampled points $S = \{S_k\}_{k=1,\dots,K}, S_k \in \mathbb{R}^2$ and total number of samples N . Figure 3 shows the end-point selection example in three different path shapes, such as straight, right open space, and four-way junction. The selected S^* is defined as the end-point of the grid-search path. Finally, a low-pass filter is implemented to prevent end-point oscillation due to the noise of raw laser scanner input.

4.4 | ESDF weighted cost function

The online map from the Cartographer [28] gives the map of the environment in a regular probability grid form. With this probability, we generate a Boolean-valued voxel representation which characterizes the flyable area, defined as

$$M_{\text{bool}}(\xi) = \begin{cases} 1 & \text{if } M_{\text{prob}}(x) > T_\zeta, \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

where $M_{\text{prob}}(\cdot) : \mathbb{R}^2 \rightarrow \mathbb{R}$ is the occupancy probability function of the local-map at grid point x , $M_{\text{bool}}(\cdot)$ is the Boolean-value representation, and T_ζ is the constant probability threshold value (which is set for 70 (range: 0 to 100) in this work). $M_{\text{bool}}(\cdot)$ is then converted to distance field using Euclidean signed distance transform [29].

For planning, we implement a grid-search method to determine the shortest path from the vehicle's current position to the selected end-point. We model the cost function using the following two terms: an ESDF term f_{dist} , which measures the cost of being near obstacles; and a prior term f_{score} , which measures the distance of the path between the current position and end-point. Therefore, we present the objective as follows:

$$f_{\text{total}}(n) = f_{\text{score}}(n) + f_{\text{dist}}(n), \quad (4)$$

$$f_{\text{score}}(n) = g(n) + h(n), \quad (5)$$

where $g(n)$ is the cost from start to node n , $h(n)$ is the cost from n to goal, $\eta(n)$ is the depth of the search and N is the estimated length of the path to reach the goal [30].

4.5 | Localization and mapping

Underlying mapping representation of this study takes Cartographer by Google [28], which showed an outstanding performance using continuous-time SLAM by Ceres-based scan matcher and optimizer [31] and grouped probability grids called submaps. The Cartographer receives an odometry source as a secondary input to correct position drift in a featureless environment. Therefore, we employed a vision-based state estimation source from QSF as an additional measurement source. Figure 4 shows the mapping result of LiDAR-only SLAM and SLAM with a visual correction at the Beckley mine in West Virginia. Since we fused vision-based state estimation from QSF boards, which are highly integrated single-board computers, time synchronization was performed using Chrony [32] to cooperate this board with the mission computer.

5 | EVALUATION STUDIES

To evaluate the lightweight autonomy framework, a series of simulations and experimental studies were conducted. Within the experimental field results, experiments in three different underground mine environments were performed to validate the robustness and scalability of the system. The experimental environments were a coal mine in South Korea and gold and inert coal mines in the United States.

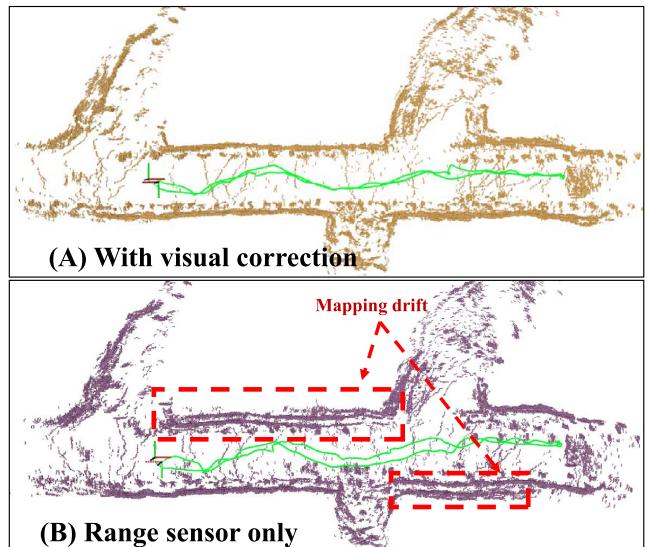


FIGURE 4 Experimental comparison of localization with a visual correction (top) and LiDAR-only localization (bottom). LiDAR-only SLAM shows a mapping drift of 0.14/m due to lack of features in this underground mine environment. Location: Beckley Mine in West Virginia, United States

5.1 | Simulation results

To evaluate and compare the proposed lightweight flight autonomy with other state-of-the-art methods, a simulation study was conducted before its real-world verification. Simulations can perform large-scale iterative tests without any cost loss. This allows us to test the stability and capability of the newly applied algorithms before conducting hardware experiments. Three simulation configurations consisting of an office, forest, and subterranean environments were utilized (see Figure 5). The office environment consisted of narrow hallway and multiple rooms. The forest environment consisted of cluttered trees and unstructured models. Finally, the subterranean environment consisted of multi-way intersections and narrow corridors with lengths of several kilometers. Examples were performed using a high-fidelity Gazebo simulator with a P x 4 flight stack [33] for flight performance evaluation.

The custom robot is developed for simulation studies assuming a quadrotor 350 mm MAV model with a laser scanner, a 2D rotating scanner up to 20 m. The local planning window is set to $5 \text{ m} \times 5 \text{ m} \times 5 \text{ m}$ around the current vehicle pose, and the flight speed was set to 2 m/s. In all simulated tests the performance comparison is conducted with receding horizon rapidly-exploring random tree* (RHRRT*) [34] and likelihood-based planning [35]. Simulation results presenting the performance of the (a) proposed planner, (b) Falco, and (c) RHRRT* planner are shown in Figures 6, 7, and 8. In these simulation tests, all methods use the same vehicle configuration, except for the range measurement sensor. Our planner and RHRRT* used a 2D laser scanner-based 2.5D mapping while Falco used the LOAM [36] based 3D mapping method due to the basic framework setting difference. Our method and Falco showed the best performance.

The proposed lightweight framework successfully explored the three simulation environments. Falco also delivered reasonable performance. However, it showed staying or back-and-forth motions because it cannot define the next step that maximizes the likelihood of the drone flying to the goal. Falco further requires destination points. Because it cannot define the destination in advance (at least in the current Falco open-source package), Falco is inadequate for unknown area exploration problems. The RHRRT* method was excluded from the forest and subterranean simulation test because it consumes much time to generate the path. Thus, it mostly fails to obtain a meaningful result.

Comparative analysis from simulation studies showed that the proposed lightweight autonomy framework has appropriate capability in various environments (see Table 1). In three scenarios, the exploration time of each method was tested five times in the same initial configuration. In all tests, the maximum flight speed was set to 2 m/s. Falco and the proposed planner successfully reached the destination in office and forest tests. On average, the proposed planner performs more than 10 s faster, as Falco showed a time lag while scrambling in a narrow passage. For underground tests, Falco was unable to escape from the forked road where one road was blocked. This is considered an inevitable problem because Falco proceeds by changing only the heading while looking at only the small local area. An extensive set of field experiments is presented in the next subsection to further verify the proposed framework in a real-world environment.

5.2 | Experiments

In this section, we evaluate the performance of the proposed collision-free navigation strategy in various real-

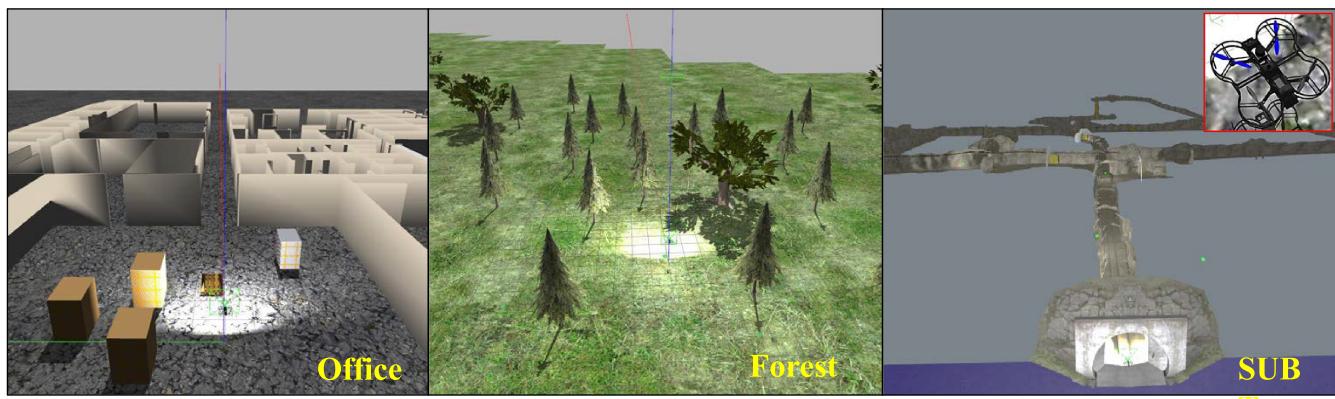


FIGURE 5 Various simulation environments setup with custom build 350 mm lightweight drone to evaluate and fine-tune the proposed method



FIGURE 6 Compared confined office area simulation results of flight autonomy frameworks in the local planning viewpoint. The proposed lightweight flight autonomy and Falco show good performance, but RHRRT* cannot determine the proper path. The proposed lightweight framework, as shown in (A), successfully finds the safe path. The snapshots of the local-map and ESDF map at each position are presented. The solutions from Falco exhibit overall good performance; however, it presents wavering or staying motions for a few seconds due to trapping in local minima, as shown in (B). The RHRRT* could not determine the path within a reasonable amount of time because it spans the trees over the entire map

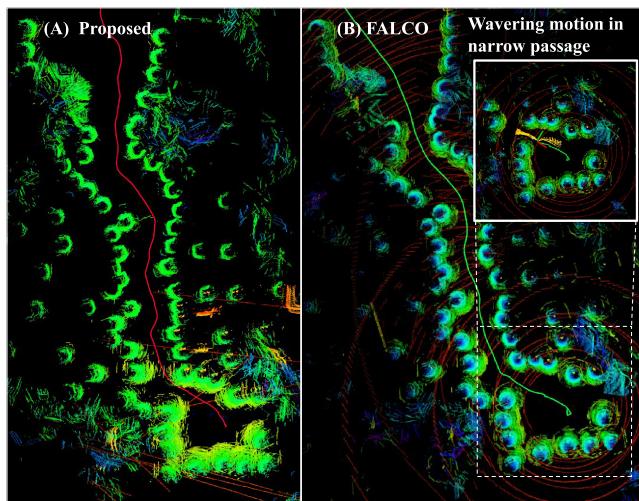


FIGURE 7 Compared forest area simulation results of flight autonomy frameworks in the local planning viewpoints. The proposed lightweight flight autonomy and Falco show good performance; however, the Falco shows wavering or staying motion in a narrow passage

world subterranean environments. The eagle mine in the United States, California, has a long narrow straight passage. The Hwasoon mine in South Korea has a completely dark environment and round-shape structure. The Beckley mine in the United States, West Virginia, has a square-shaped structure with several pillars inside the mine. These three mines appear similar but have different characteristics. The eagle mine is a gold mine with a large amount of sand dust. Being a gold mine, eagle mine has many wall features that are advantageous for localization but require sophisticated control through the narrow passage. Hwasoon mine is an active coal mine containing more coal dust than sand dust, but wider passageways than eagle mine. Finally, Beckley mine is an inert coal mine. The effects of dust are small because water is falling from the ceiling. This mine has a wider passageway but lower height than the other two mines, so exerts a greater ground effect during flight. All flight experiments were performed in fully autonomous mode in several test sites, which were

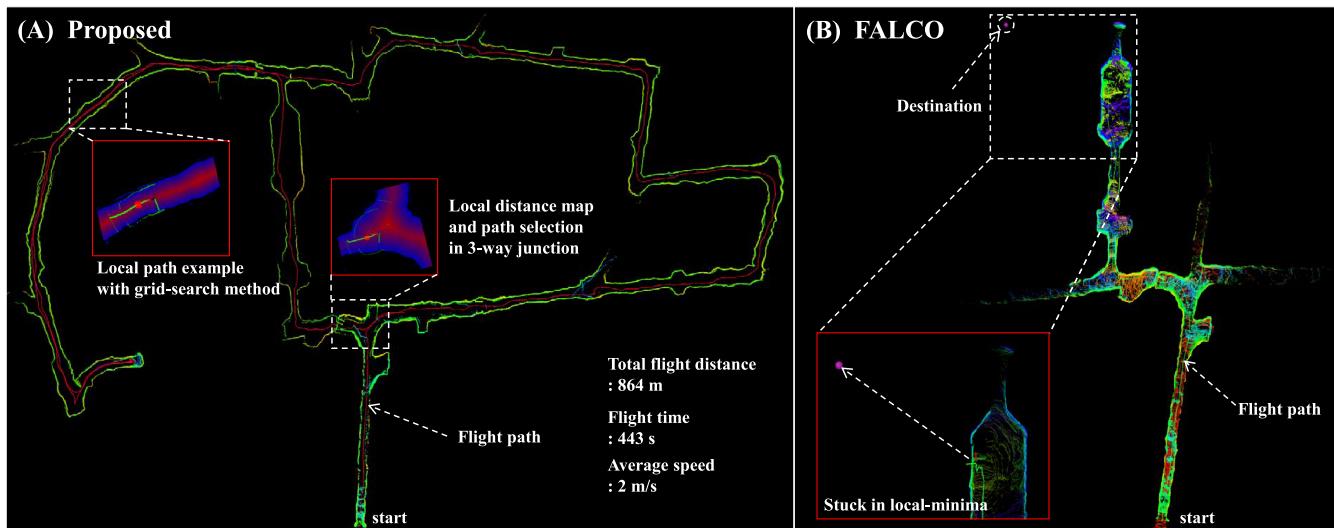


FIGURE 8 Compared underground mine (subterranean) area simulation results of flight autonomy frameworks in the local planning point of view. Our planner exhibits best performance while flying over 864 m in 443 s as shown in (A). Falco shows reasonable performance; however, it gets trapped in local minima and cannot proceed further as shown in (B)

TABLE 1 Performance comparison in each scenarios

Scene	Method	Exploration time (s)				Explored area (s)
		Avg	Std	Max	Min	
Office	Falco [35]	67.17	6.00	72.05	58.36	2.08
	RHRRT*	-	-	-	-	-
	Proposed	40.89	1.37	42.48	39.17	3.42
Forest	Falco [35]	59.83	7.74	69.29	49.89	2.50
	Proposed	42.19	1.49	43.78	40.11	3.55
Subt	Falco [35]	-	-	-	-	-
	Proposed	226.00	1.13	227.43	224.56	3.48

unknown in advance. The overall system architecture is illustrated in Figure 9.

5.2.1 | Test in the eagle mine

In this example, the drone explores a long narrow wall shaped gold mine, located in Southern California. Here, we evaluate the proposed algorithm integration performance. The end-point selection, local-map generation, ESDF in local-map area, and the grid-search algorithm. As presented in Figure 10, the proposed

lightweight autonomy framework is successfully integrated. The drone starts at the entrance of the mine and explores about 105 m in 62 s while passing a four-way junction. All environmental information is unknown before its flight test. The characteristics of this test site are presented in Table 2.

5.2.2 | Test in the Hwasoon mine

In this example, we present a coal mine exploration with a two-way junction and smooth curves terrain. Unlike the square-shaped narrow long-wall gold mine example presented in the previous section, the coal mine in South Korea is round-shaped and relatively wide. The coal mine in South Korea has an uneven and bumpy wall texture because the miners widen the mine with a shovel. Besides, wooden supports are supported every 1 m to prevent the mine from collapsing. Therefore, it is useful for scan-matching with sufficient features on the wall, but it is not easy to use for vision sources because of the dust on the floor. In this experiment, the quadrotor successfully explores 182 m for 142 s along a curved tunnel using the lightweight autonomy framework. Figure 11 shows all the integrated modular framework. More specifically, at the two-way junction, the end-point selection method found that the right forked path was blocked and chose the left path. This test allowed us to evaluate long-range flight capabilities with dead-end intersections. The specification of this test mine is presented in Table 3.

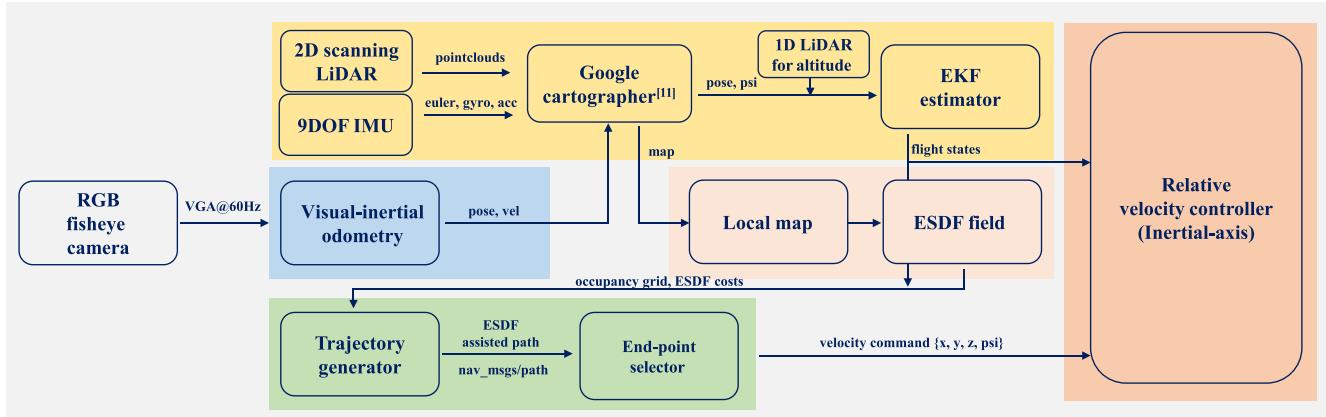


FIGURE 9 Software architecture overview of our lightweight autonomy system from sensor input to control command output

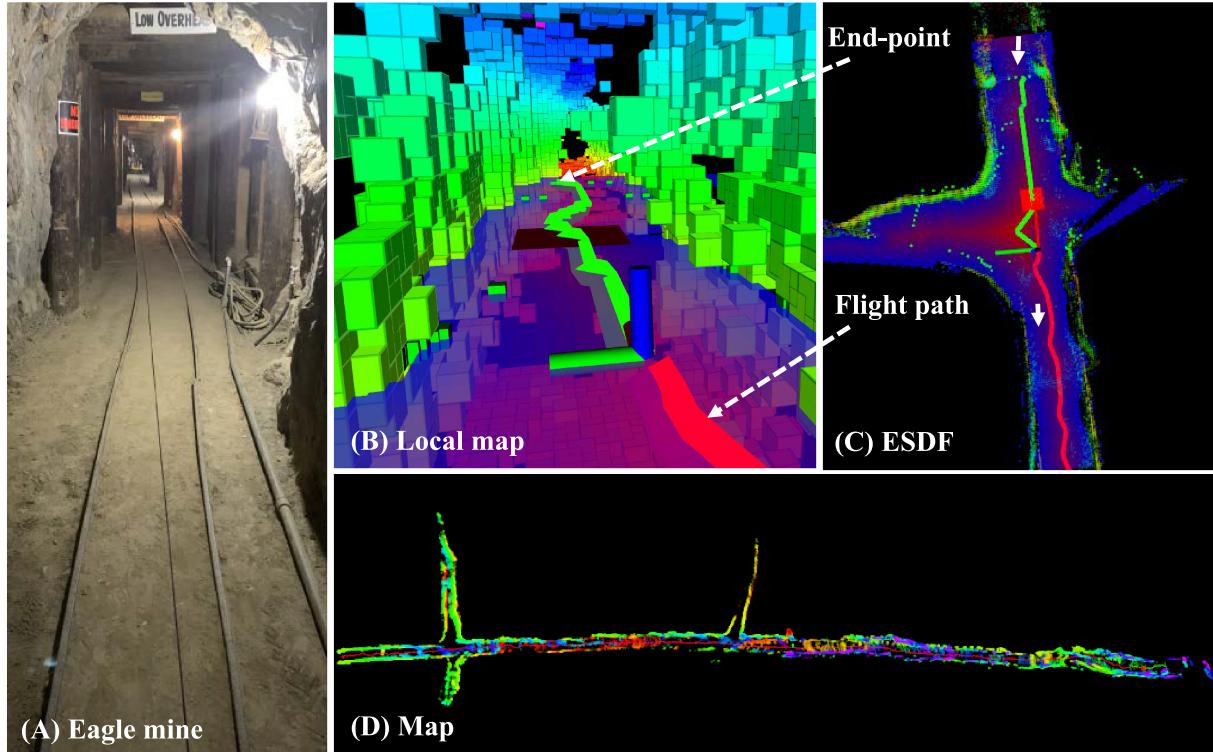


FIGURE 10 (A) Result of the first integrated experiment at the narrow gold mine environment. (B) Local-map of straight path, (C) ESDF map and consequently local-path planning results at four-way junction. This result shows the successful algorithm integration of (1) sensing; (2) local map generation; (3) ESDF generation; (4) planning. Location: eagle mine, United States, Southern California

TABLE 2 Characteristic of eagle mine

Parameter	Value	Parameter	Value
Width	≤ 1.2 m	Name	Eagle mine
Height	≤ 1.0 m	Location	Southern California
Tunnel shape	\cap shape	Characteristic	Dusty, Narrow

5.2.3 | Test in the Beckley mine

As an example, this section presents the results of the Beckley mine exploration. The Beckley mine is a coal mine with a square-shaped structure. Its widths are approximately 3 m. Here, we tested three types of

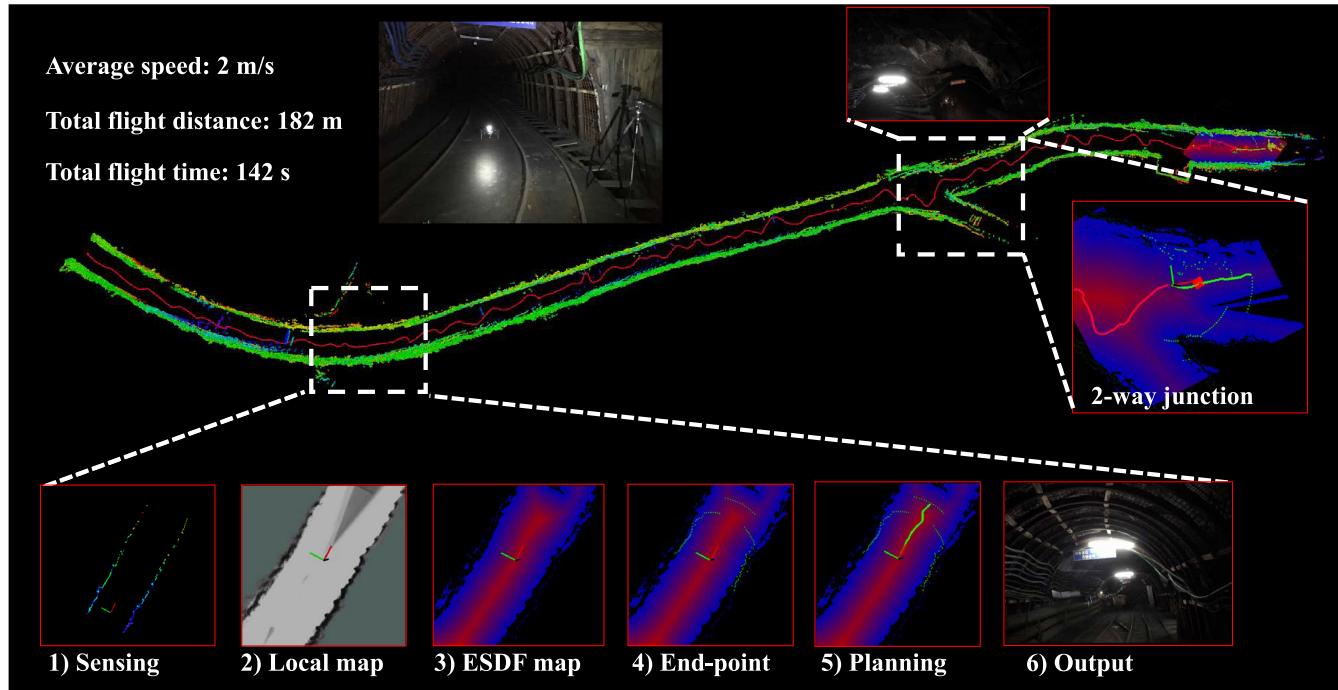


FIGURE 11 Autonomous exploration results and visualization of the proposed collision-free navigation modules at the Hwasoon coal mine in South Korea: (1) Sensing via the laser scanner; (2) local map generation at the vehicle's current position; (3) successful generation of the ESDF Map; (4) end-point selection based on the vehicle's current heading and ESDF information; (5) local path planning result from the current position to the end-point; (6) actual flying snapshot during collision-free navigation. The aerial robot explores about 182 m during 142 s flight. Location: Hwasoon mine (active mine), South Korea

TABLE 3 Characteristic of Hwasoon mine

Parameter	Value	Parameter	Value
Width	≤ 4.0 m	Name	Hwasoon coal mine
Height	≤ 2.5 m	Location	South Korea
Tunnel shape	\cap shape	Characteristic	Somewhat Dusty

courses: curved, narrow corridors with pillars, and straight. The curved course has long curved corridors with a few junctions, and the narrow pillar course has a confined corridor with several pillars. Finally, the straight course is open to the left and the end of the course is blocked by obstacles.

Figure 12 presents the curve type course test. The framework explores approximately 72 m in 50 s while passing the left fork and returning home (here, 50 s is the pre-set time limit). A narrow pillar example is shown in Figure 13. This example shows the collision-free navigation capability in confined and cluttered environment. The straight course example is presented in Figure 14. The drone reached the dead-end and returned home

approximately 0.2 m from the starting point. The specification of the Beckley coal mine is presented in Table 4.

5.3 | Discussions

The results provide a comprehensive evaluation of the performance of the proposed collision-free navigation framework in various simulation and real-world environments. Each module, such as the local-map generation, the ESDF map generation, end-point selection, and grid-search confer fully autonomous collision-free navigation capability to the real-world aerial robot. The end-point selection method selected a destination where the robot can safely navigate the local area. We demonstrated the successful exploration capability of the lightweight autonomy framework by fusing the end-point with the grid-search method, reflecting ESDF values.

A deeper consideration is that quadrotors generate more dust while flying through the mine. To address this problem, we measured the intensity value using a software de-dust method. However, we failed to detect thin objects and obstacles with less reflection in this case. Eventually, we solved the problem by replacing the hardware from Hokuyo UST-20LX, a passive laser

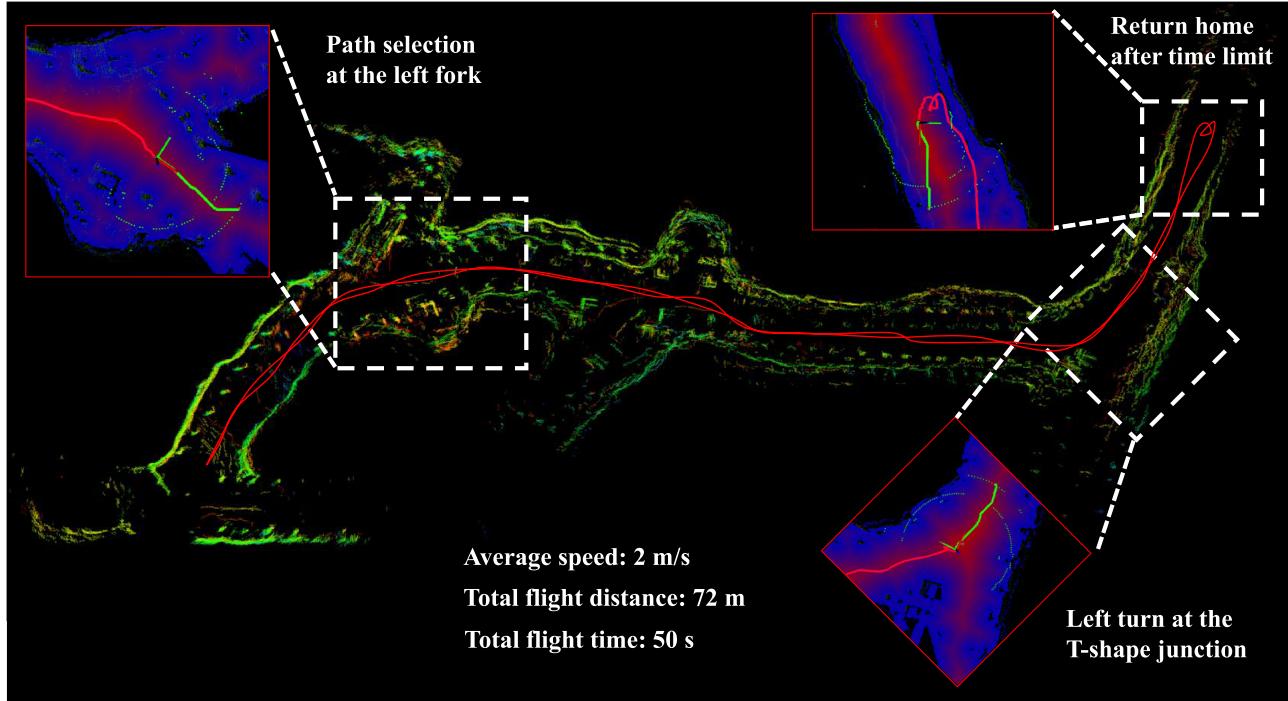


FIGURE 12 Autonomous exploration results at the Beckley coal mine. Unlike the Hwasoon mine, this mine includes a path with several forks and pillars that increase the collision risk of the vehicle. In this experiment, we also tested the return to home (RTH) functionality after the pre-set time limit. The aerial robot explores long smooth curve while passing the left fork and RTH after flying 72 m in 50 s. Location: Beckley mine, West Virginia, United States

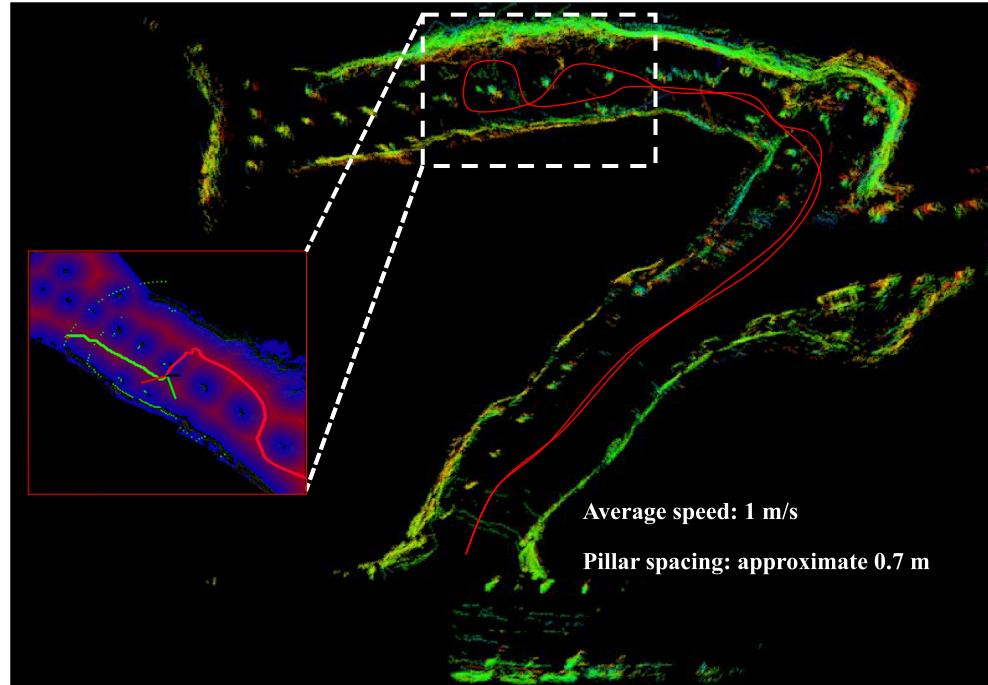


FIGURE 13 Autonomous exploration results at the Beckley coal mine. This course has a very narrow room-and-pillar structure with approximate spacings of 0.5 m to 1 m. In this experiment, our local planner explored the collision-free path in narrow spaces. Location: Beckley mine, West Virginia, United States

scanner with RPLiDAR A3, an active laser scanner. However, more fundamental dust-rejection methods, such as multi-sensor fusion should be devised. In addition, the VIO Fusion functionality provided by the

Cartographer does not perform the correction when the VIO diverges. Therefore, the localization performance depends on the VIOs. To improve the robustness of the localization performance, we propose the introduction

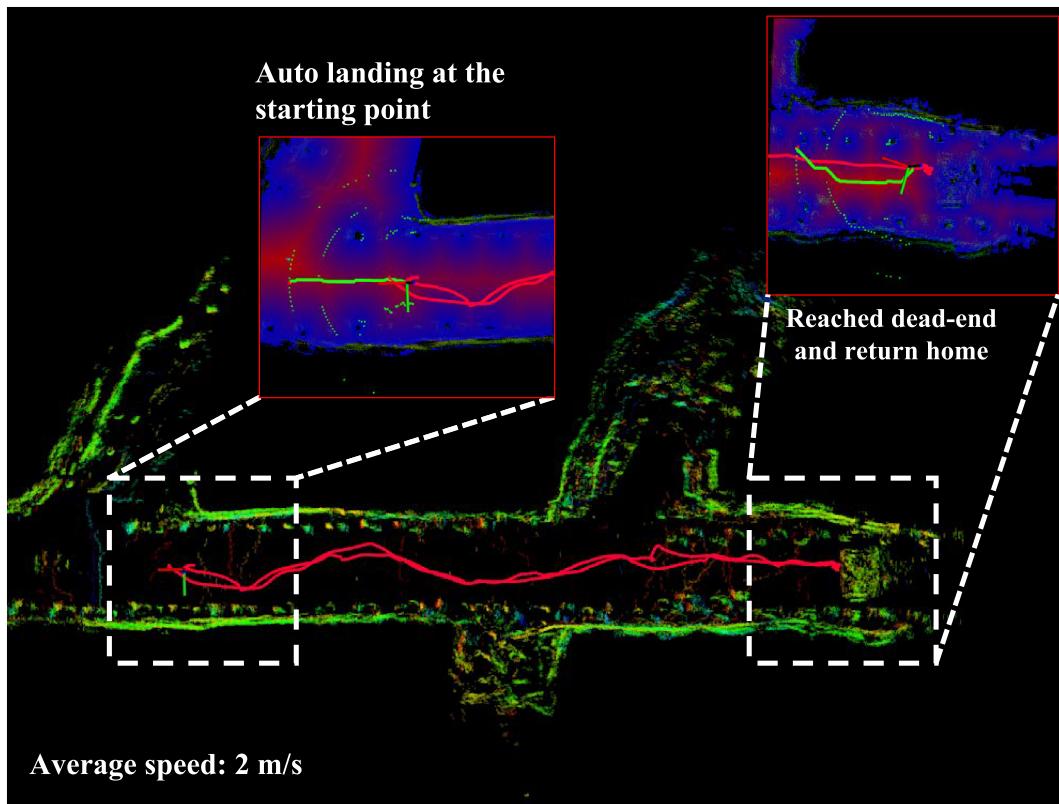


FIGURE 14 Autonomous exploration results at the Beckley coal mine. Successful autonomous flight with turn-around action after detecting the dead-end. A video of this experiment can be found at https://youtu.be/MdYXu_ImXik Location: Beckley mine, West Virginia, United States

TABLE 4 Characteristic of the Beckley mine

Parameter	Value	Parameter	Value
Width	≤ 3.0 m	Name	Beckley coal mine
Height	≤ 1.5 m	Location	West Virginia
Tunnel shape	Square shape	Characteristic	Muddy

of estimation techniques and covariance check algorithms such as an error-state Kalman filter and a chi-squared test, which would properly fuse the advantages of SLAM and VIO.

6 | CONCLUSIONS

This study proposes a detailed autonomy framework for exploring unknown environments for a lightweight aerial robot. The proposed method utilizes the ESDF map to fly through a collision-free flight path, followed by an endpoint selection method. The proposed lightweight autonomy framework has been experimented with various

simulations and real-world environments, and successfully demonstrated the performance of the algorithm.

In the future, we will extend our 2.5D mapping approach to 3D using the recently developed lightweight solid-state-LiDAR [37], to achieve full-scale planning and exploration in citified areas such as urban areas, which require a more agile vertical planning method. The method can be directly implemented in a 3D method when a 3D occupancy grid map is available. In this manner, the drone can fly a horizontal course and vertical passage, such as stairways. Finally, this study focuses on local planners and excludes information about global planners. By introducing a hierarchical route planning method to effectively and meticulously observe the area under exploration, we will plan a global road map and research the avoidance of local obstacle collisions while moving along the road map.

ACKNOWLEDGEMENT

This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government (2017-0-00067, Development of ICT Core Technologies for Safe Unmanned Vehicles).

ORCID

Sunggoo Jung  <https://orcid.org/0000-0001-7672-6885>

REFERENCES

1. H. Balta et al., *Integrated data management for a fleet of search-and-rescue robots*, J. Field Robot. **34** (2017), no. 3, 539–582.
2. A. Bircher et al., *Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots*, Auton. Robots **40** (2016), no. 6, 1059–1078.
3. M. J. Milford and G. F. Wyeth, *SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights*, in Proc. IEEE Int. Conf. Robot. Autom. (Saint Paul, MN, USA), May 2012, pp. 1643–1649.
4. T. N. Yap and C. R. Shelton, *Slam in large indoor environments with low-cost, noisy, and sparse sonars*, in Proc. IEEE Int. Conf. Robot. Autom. (Kobe, Japan), May 2009, pp. 1395–1401.
5. A. Agha et al., *Nebula: Quest for robotic autonomy in challenging environments; team costar at the darpa subterranean challenge*, arXiv preprint, CoRR, 2021, arXiv: 2103.11470.
6. B. Yamauchi, *A frontier-based approach for autonomous exploration*, in Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom. (IEEE, Monterey, CA, USA), July 1997, pp. 146–151.
7. P. E. Hart, N. J. Nilsson, and B. Raphael, *A formal basis for the heuristic determination of minimum cost paths*, IEEE Trans. Syst. Sci. Cyber. **4** (1968), no. 2, 100–107.
8. C. Connolly, *The determination of next best views*, Proceedings in Proc. IEEE Int. Conf. Robot. Autom. (St. Louis, MO, USA), Mar. 1985, pp. 432–435.
9. W. A. Kamal, D.-W. Gu, and I. Postlethwaite, *Milp and its application in flight path planning*, IFAC Proc. **38** (2005), no. 1, 55–60.
10. T. Schouwenaars, J. How, and E. Feron, *Receding horizon path planning with implicit safety guarantees*, in Proc. Am. Control Conf. (Boston, MA, USA), 2004, pp. 5576–5581.
11. Y. Kuwata et al., *Robust constrained receding horizon control for trajectory planning*, in Proc. AIAA Guid., Navig., Control Conf. Exhibit (San Francisco, CA, USA), Aug. 2005, pp. 1–12.
12. D. Shim et al., *Autonomous exploration in unknown urban environments for unmanned aerial vehicles*, in Proc. AIAA Guid., Navig., Control Conf. Exhibit (San Francisco, CA, USA), Aug. 2005.
13. R. Prazenica et al., *Vision-based geometry estimation and receding horizon path planning for uavs operating in urban environments*, in Proc. Am. Control Conf. (Minneapolis, MN, USA), June 2006, pp. 2874–2879.
14. J. Bellingham, A. Richards, and J. P. How, *Receding horizon control of autonomous aerial vehicles*, in Proc. Am. Control Conf. (Anchorage, AK, USA), May 2002, pp. 3741–3746.
15. A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, *Receding horizon “next-best-view” planner for 3d exploration*, in Proc. IEEE Int. Conf. Robot. Autom. (Stockholm, Sweden), May 2016, pp. 1462–1468.
16. R. Allen and M. Pavone, *A real-time framework for kinodynamic planning with application to quadrotor obstacle avoidance*, in Proc. AIAA Guid., Navig., Control Conf. (San Diego, California, USA), Jan. 2016, pp. 1–18.
17. F. Gao, Y. Lin, and S. Shen, *Gradient-based online safe trajectory generation for quadrotor flight in complex environments*, in Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (Vancouver, BC, Canada), Sept. 2017, pp. 3681–3688.
18. X. Peng and D. Xu, *Intelligent online path planning for uavs in adversarial environments*, Int. J. Adv. Robot. Syst. **9** (2012), no. 1, 3.
19. S. Lai et al., *A robust online path planning approach in cluttered environments for micro rotorcraft drones*, Control Theory Technol. **14** (2016), no. 1, 83–96.
20. M. Popovic et al., *Online informative path planning for active classification on uavs*, arXiv preprint, CoRR, 2016, arXiv: 1606.08164.
21. <https://www.darpa.mil/program/darpa-subterranean-challenge>, [Online Accessed: 8 Febuary, 2021].
22. T. Dang et al., *Graph-based subterranean exploration path planning using aerial and legged robots*, J. Field Robot. **37** (2020), no. 8, 1363–1388.
23. T. Dang, C. Papachristos, and K. Alexis, *Visual saliency-aware receding horizon autonomous exploration with application to aerial robotics*, in Proc. IEEE Int. Conf. Robot. Autom. (Brisbane, QLD, Australia), May 2018, pp. 2526–2533.
24. R. Reinhart et al., *Learning-based path planning for autonomous exploration of subterranean environments*, in Proc. IEEE Int. Conf. Robot. Autom. (Paris, France), May 2020, pp. 1215–1221.
25. P. De Petris et al., *Collision-tolerant autonomous navigation through manhole-sized confined environments*, in Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (Abu Dhabi, United Arab Emirates), Nov. 2020, pp. 84–89.
26. C. H. Tan et al., *Design optimization of sparse sensing array for extended aerial robot navigation in deep hazardous tunnels*, IEEE Robot. Autom. Lett. **4** (2019), no. 2, 862–869.
27. <https://auvidea.eu/j120/>, [Online Accessed: 8 February, 2021].
28. W. Hes et al., *Real-time loop closure in 2d lidar slam*, in Proc. IEEE Int. Conf. Robot. Autom. (Stockholm, Sweden), May 2016, pp. 1271–1278.
29. H. Oleynikova et al., *Signed distance fields: A natural representation for both mapping and planning*, in RSS 2016 Workshop: Geometry and Beyond-Representations, Physics, and Scene Understanding for Robotics, University of Michigan, St. Ann Arbor, MI, USA, 2016.
30. I. Pohl, *The avoidance of (relative) catastrophe, heuristic competence, genuine dynamic weighting and computational issues in heuristic problem solving*, in Proc. Int. Joint Conf. Artif. Intell. (San Francisco, CA, USA), Aug. 1973, pp. 12–17.
31. S. Agarwal et al., *Ceres Solver*, available at <http://ceres-solver.org>
32. <https://chrony.tuxfamily.org>, [Online Accessed: 8 Febuary, 2021].
33. L. Meier et al., *Pixhawk: A system for autonomous flight using onboard computer vision*, in Proc. IEEE Int. Conf. Robot. Autom. (Shanghai, China), May 2011, pp. 2992–2997.
34. S. Karaman et al., *Anytime Motion Planning using the RRT*, in Proc. IEEE Int. Conf. Robot. Autom. (Shanghai, China), May 2011, pp. 1478–1483.
35. J. Zhang, et al., *Falco: Fast likelihood-based collision avoidance with extension to human-guided navigation*, J. Field Robot. **37** (2020), no. 8, 1300–1313.
36. J. Zhang and S. Singh, *Loam: Lidar odometry and mapping in real-time*, Robot. Sci. Syst. **2** (2014).

37. <https://www.intelrealsense.com/lidar-camera-l515/>, [Online Accessed: 8 Febuary, 2021].

AUTHOR BIOGRAPHIES



Sunggoo Jung received the BS degree in mechanical and control engineering from Handong Global University, Pohang, South Korea, in 2014, and the PhD degrees in aerospace engineering from KAIST, Daejeon, South Korea, in 2020. He is currently a aerial robotics researcher at Electronics and Telecommunications Research Institute (ETRI). His research interests lie in model based reinforcement learning and agile indoor autonomous flight.



Hanseob Lee received his BS degree in aerospace engineering from Korea Aerospace University, Goyang, South Korea, in 2015, and his MS and PhD degrees in aerospace engineering from KAIST, Daejeon, South Korea, in 2017, and 2021 respectively. His research interests are continuous reinforcement learning and autonomous applications in aerial robotics.



David Hyunchul Shim received his BS and MS degrees in mechanical design and production engineering from Seoul National University, Seoul, Korea, in 1991 and 1993, respectively, and his PhD degree in mechanical engineering from the University of California Berkeley, Berkeley, CA, USA, in 2000. He was Transmission Design Engineer at The Hyundai Motor Company, Young-in, Korea, during 1993 1994. From 2001 to 2005, he was with Maxtor

Corporation, Milpitas, CA, USA, as a Staff Engineer working on advanced control systems for hard disk drives. From 2005 to 2007, he was Principal Engineer in charge of the Berkeley Aerobot Team at University of California, Berkeley. In 2007, he was employed as an Assistant Professor at the Department of aerospace engineering, School of Mechanical Aerospace and Systems Engineering, College of Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea. He is currently a full-time Professor at the Electrical Engineering Department and the Director of KI Robotics, KAIST. His research interests include dynamics and control, unmanned vehicle systems, and field robotics.



Ali-akbar Agha-mohammadi received his PhD from Texas A&M University (2013). Dr. Agha is Group Leader of the Aerial Mobility Group at JPL's Autonomous and Robotic Systems Division. His research focuses on the autonomy of robotic systems and spacecraft. At JPL, Dr. Agha leads the Team CoSTAR and the development of the NeBula autonomy solution. He also leads several other robotic autonomy projects at JPL with a dual focus on planetary exploration and terrestrial applications. Previously, Dr. Agha was a Postdoctoral Researcher at MIT and later joined Qualcomm Research's Robotic and Autonomy Division. His research interests include artificial intelligence, autonomous decision making, and perception for robotic systems, with applications to drones, rovers, and legged robots. Dr. Agha was named a NASA NIAC fellow in 2018.