IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 December 2024, accepted 28 December 2024, date of publication 31 December 2024, date of current version 8 January 2025.

Digital Object Identifier 10.1109/ACCESS.2024.3524586

RESEARCH ARTICLE

Development of Tributary Mapping System Using Multiple Unmanned Aerial Vehicles With LiDAR

JAEHWI SEOL[1,2], JEONGHYEON PAK[1,2], AND HYOUNG IL SON[1,2,3], (Senior Member, IEEE)

[1]Department of Convergence Biosystems Engineering, Chonnam National University, Gwangju 61186, Republic of Korea
[2]Interdisciplinary Program in IT-Bio Convergence System, Chonnam National University, Gwangju 61186, Republic of Korea
[3]Research Center for Biological Cybernetics, Chonnam National University, Gwangju 61186, Republic of Korea

Corresponding author: Hyoung Il Son (hison@jnu.ac.kr)

This work was supported in part by the Korea Institute of Planning and Evaluation for Technology in Food, Agriculture, and Forestry (IPET) through the Agriculture and Food Convergence Technologies Program for Research Manpower Development Program, Ministry of Agriculture, Food, and Rural Affairs (MAFRA) under Grant RS-2024-00402136; in part by the IPET through the Project on Fostering Expertise and Advancing Technologies in Digital Agriculture, MAFRA under Project RS-2024-00397026; and in part by the "Convergence and Open Sharing System" Project, Ministry of Education and National Research Foundation of Korea.

ABSTRACT This paper proposes a hybrid system comprising multiple unmanned aerial vehicles (UAVs) to monitor nature. The proposed tributary mapping system consists of perception and control systems. A perception system is established for tributary mapping using three-dimensional light detection and ranging (LiDAR) semantic segmentation for surface water recognition. The system defines a water point cloud and segments the surface water area using singular value decomposition, applying the water-absorbing property of LiDAR. A path is generated through random sample consensus by calculating the center point from the segmented area. Each UAV is modeled using a continuous-time dynamic-based low-level model and a discrete event-time dynamic-based high-level model to control the multiple UAVs system. The desired plant behavior is designed considering the control objectives of the plant, and a supervisory controller is developed accordingly. A physics-based robot simulator is employed to verify the proposed tributary mapping system in perception and path generation and the performance of the hybrid system-based supervisory controller.

INDEX TERMS Tributary mapping, semantic segmentation, hybrid system, supervisory controller, simultaneous localization and mapping.

I. INTRODUCTION

Mapping and managing tributaries is pivotal in agricultural crop production and natural ecosystem conservation. The topographical features of agricultural land and tributaries significantly influence crucial factors such as water flow dynamics, soil erosion patterns, and groundwater recharge rates. Effective tributary mapping is vital to optimize moisture management in agricultural land, mitigate erosion risks, and promote sustainable agricultural practices. In agricultural contexts, widespread tributary networks directly affect water distribution and utilization, profoundly influencing crop yields and productivity. Similarly, tributaries are vital in natural ecosystems that shape ecosystem health and biodiversity. Changes in tributary characteristics can disrupt the ecological balance and jeopardize biodiversity, highlighting the criticality of tributary mapping and management in conservation efforts. Therefore, tributary mapping and management are increasingly recognized as essential tasks for enhancing agricultural productivity and conserving natural ecosystems.

Despite the recognized importance of tributary mapping, conventional methods are often limited, particularly in the context of vast and complex agricultural landscapes. Traditional approaches typically entail manual data collection through on-site surveys, yielding highly accurate data, but are time-consuming and require extensive human involvement. Alternatively, remote sensing techniques [1], [2], such as satellite-based aerial imaging, help remotely acquire tributary-related information, reducing labor and time requirements. However, these savings often come at the

The associate editor coordinating the review of this manuscript and approving it for publication was Yang Tang.

cost of accuracy and detail of data. Robotics technology is increasingly used in diverse fields, including agriculture [3], ecosystem management [4], and environmental monitoring [5] to address these challenges.

Both unmanned ground vehicles (UGVs) and unmanned aerial vehicles (UAVs) are used to monitor the natural environment. However, these types of robots are affected by unique environmental constraints. For tributary mapping, UAVs present a distinct advantage, given their ability to access areas inaccessible to UGVs. The UAVs offer several benefits over traditional mapping methods. They can generate detailed tributary maps in remote or difficult-to-reach areas, significantly enhancing the scope of environmental monitoring. In addition, when equipped with special sensors, UAVs can efficiently and accurately capture critical data, such as water flow patterns, plant health, and soil moisture content. These capabilities are crucial in agricultural applications, where precise and timely data are essential for effective land and water management. Moreover, UAVs facilitate real-time monitoring and allow rapid responses to changes in tributary conditions, which are crucial for maintaining optimal environmental health [6]. Frequent UAV-based inspections help detect erosion, water pollution, and other environmental problems early. Such early warnings enable farmers and land managers to implement mitigation measures promptly, preventing minor problems from escalating into major ones. In these ways, UAV-based tributary mapping enhances the accessibility and scalability of environmental monitoring, reducing costs and labor requirements associated with traditional surveying methods.

However, given the large working environment, it is challenging to implement effective environmental and ecosystem monitoring using a single robot. Consequently, multi-robot system (MRS) has been researched as a viable solution to increase work efficiency and address the limitations of single-robot systems [7]. Moreover, MRS-based collaboration offers several advantages, particularly regarding system reliability and operational efficiency. These systems exhibit flexibility, allowing them to continue functioning even if a particular robot fails. Therefore, the overall reliability of MRS is higher than that of single-robot systems [8]. This characteristic of MRS addresses a critical limitation of single robots, which is restricted by operational time.

The MRS can be categorized into two primary types based on their tasks: strong and weak cooperation [9]. In strong cooperation scenarios, robots directly interact to accomplish specific tasks that a single robot cannot manage independently. An example of such tasks is the transportation of heavy objects, the weights of which exceed the payload capacity of an individual robot. Multiple robots can collaboratively manage this task by continuously coordinating their actions, overcoming the limitations imposed by the payload capacity of a single robot. In contrast, weak cooperation involves dividing and assigning distinct tasks to individual robots, allowing for parallel task execution without requiring direct interaction between the robots. This approach is beneficial in functions where the independence of the robot can be maintained while still contributing to a common objective. By employing the strengths of both types of cooperation, MRS can be optimized to handle diverse tasks more efficiently and effectively than single-robot systems. Therefore, MRS implementation represents a significant advancement in robotic systems that aprovides a robust framework for completing complex tasks with enhanced reliability and operational efficiency. Further research and development in this area are essential to harness the potential of MRS fully in diverse applications, ranging from environmental monitoring to industrial automation.

Although MRS afford significant efficiency advantages in wide-area environments, several technical challenges remain unsolved. One challenge is achieving accurate perception amid environmental constraints, particularly in unstructured, unknown, and dynamically changing environments [10], [11]. Aquatic environments—encompassing rivers, seas, tributaries, and coastal regions–are especially susceptible to variations in weather conditions, drastically affecting the operational capabilities of robotic systems [12]. Conventionally, vision-based perception systems have been used for environmental perception. However, these systems alone often lack the robustness required for reliable operation in complex aquatic environments [13]. Changes in illumination, reflections of surface water, and particulate matter in aquatic environments can severely degrade the performance of vision-based sensors. Light detection and ranging (LiDAR) sensors offer a promising solution to these challenges. LiDAR sensors offer dense and precise range measurements independent of lighting conditions, making them advantageous for environments with impaired visibility, such as urban or aquatic settings, where they are increasingly deployed in mobile robotic applications [14]. The ability of LiDAR to produce reliable data in various environmental conditions—ranging from bright sunlight to low-light scenarios–ensures consistent performance and enhances the robustness of robotic systems in aquatic environments. Therefore, a new approach based on LiDAR is needed to develop a perception system for monitoring aquatic environments.

The other primary challenge related to MRSs is that the classical control theory based on differential equations has several limitations in handling large-scale, complex dynamic systems such as nature monitoring. The scalability, maintainability, and modularity levels achievable using this theory are limited (e.g., in scenarios where robots must be added to perform additional tasks because of environmental uncertainty or when they are excluded because of communication errors) [15]. Discrete event systems (DESs) and formal systematic modeling methods for large-scale dynamic systems have been applied to overcome this challenge. Recently, many researchers have demonstrated that the automata-based supervised control theory (SCT) is efficient when applied to large-scale dynamic systems under DES

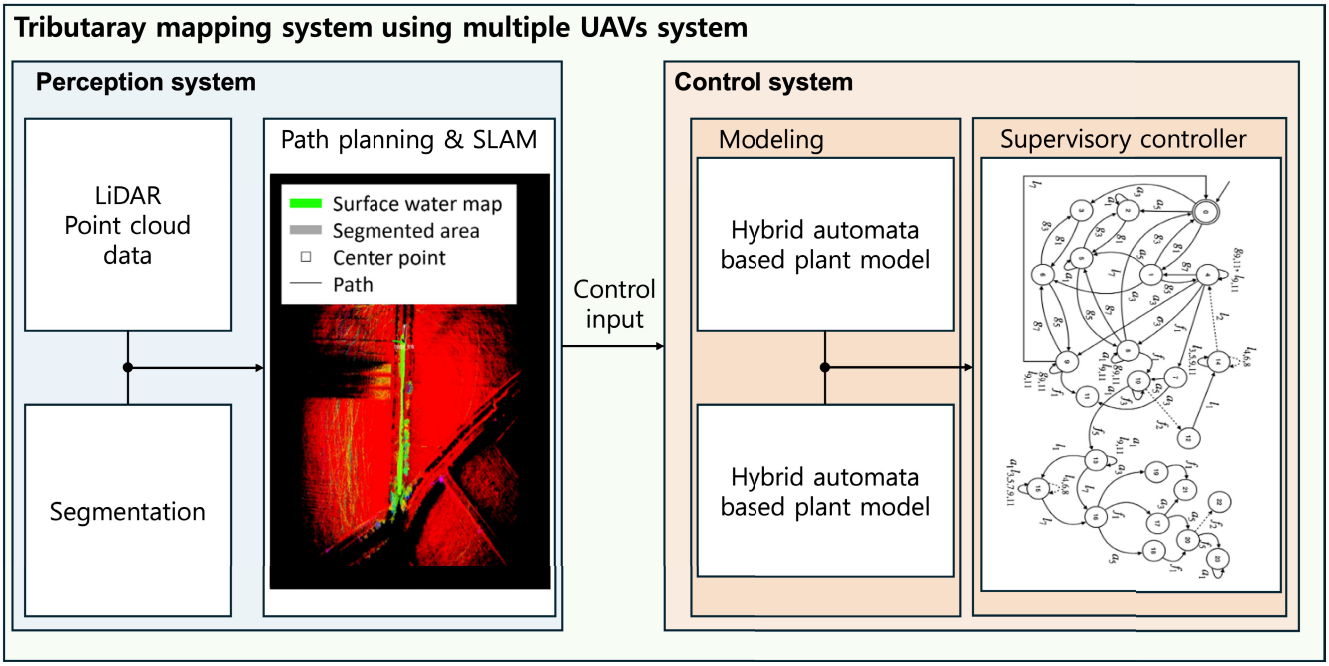


FIGURE 1. Tributary mapping framework using multiple UAVs for tributary management.

control [16]. However, model uncertainty in unstructured environments must be overcome to implement the dynamic behavior of a plant in a DES. Therefore, methods are needed to overcome the limitations of SCT-based approaches in DES.

This paper proposes a novel tributary mapping approach that integrates a surface water mapping system and a hybrid system combining a DES and a continuous time system (CTS) with multiple UAVs for tributary management. Fig. 1 depicts the proposed tributary mapping framework for a multiple UAVs system. The contributions of this work are as follows. To segment tributary areas, we develop a perception system that employs LiDAR and is robust against varying external lighting conditions. This segmented information is applied for path planning and mapping. In the proposed control system, CTS-based low-level control is employed to overcome unstructured, dynamic environments, and SCT-based high-level models and controllers are applied to overcome scalability and modularity problems in large-scale systems. This approach is ideal for large, unstructured, and unknown environments with complicated controller design and modeling. We propose a tributary mapping system combining these perception and control systems using multiple UAVs. This system can detect objects floating in tributaries, enhancing agricultural productivity and conserving natural ecosystems.

II. RELATED WORK

To our knowledge, the literature on tributary management using UAVs is limited. No existing study has employed multiple UAVs systems for this purpose. This section presents a concise overview of the methods for acquiring surface water data and controlling UAVs for mapping applications.

In tributary mapping, advanced technology, such as high-resolution cameras [17], [18], multispectral sensors [19], [20], and LiDAR sensors, are employed to gather detailed information about water bodies. The existing studies have focused on single-UAV systems with vision sensors for capturing surface water data. For example, the authors in [21] developed a localization and mapping algorithm for riverine environments. They employed a multiview geometrical formulation with initial- and current-view projections of point features from actual objects surrounding the river and their reflections. The correspondences of these features, combined with the positional and altitude information of the UAV, enhanced the observability of the estimation system. In this approach, visual data were applied to improve the accuracy of mapping and localization in river settings. However, the approach depends on the clarity and quality of the captured reflections. Similarly, in [22], the authors developed an algorithm to segment water from land in river environments containing specular reflections and distinguishable symmetric feature points. They formulated and solved a graph-based optimization problem using dense three-dimensional (3D) stereo data and Dijkstra's algorithm to determine the water boundary. However, these methods are constrained by the need for symmetrical shadows reflected on the water surface, limiting their applicability in more complex or asymmetrical scenes. Further advancing the field, the authors of [23] explored various model architectures (U-Net and DeepLab-V3+) in combination with encoder backbones (MobileNet-V3, ResNet-50, and EfficientNet-B4)

to delineate inundated areas under varied environmental conditions and data availability scenarios. They highlighted the significant effort required to prepare high-quality training data, noting that the target domain is influenced by varying scene and image properties, depending on sensor characteristics (e.g., radiometry and spatial resolution), atmospheric conditions, land use/land cover of the background class, and appearance of the water class. These findings highlight the complexity of developing robust models that accurately map water bodies under diverse conditions.

LiDAR sensors have been used extensively in robotics and environmental monitoring because they deliver dense and accurate range measurements. These sensors are especially useful in high-precision data applications, such as generating precise surface water maps. Moreover, LiDAR sensors provide reliable data under various environmental conditions that are challenging for visual and inertial data integration systems. While effective in controlled or indoor environments, these systems often struggle with the varying illumination and dynamic conditions that characterize outdoor settings. Despite the broad applicability of LiDAR, thus far, its use in water environments has focused on measuring river water levels (stages), as in [24] and [25]. These studies underscore the efficacy of LiDAR in capturing accurate water level data, which is crucial for flood monitoring and water resource management. However, the applications of LiDAR extend beyond mere water level measurements. For example, in [26], the authors described the complex challenges associated with applying LiDAR to detect navigable regions. They applied a customized deep learning network to segment LiDAR point clouds into bridges, river banks, and plants. They designed a Kalman-filter-based tracker to track objects on river banks and compensate for missing parts of river banks. Additionally, they used a wave frontier detection method to model navigable regions using objects on a river bank. Although LiDAR for water level measurement is well-documented, its potential in other aspects of environmental monitoring and robotic navigation is vast but largely untapped.

The control strategies for UAVs in water-mapping applications include autonomous navigation systems capable of real-time data processing and decision-making. These systems must be robust to environmental variations and maintain stability and accuracy under diverse weather conditions. In [27], the authors studied optimal UAV path planning for coverage search in river areas to maximize the probability of finding lost targets. Their approach entailed extracting river subregions using a Gaussian mixture model based on a probability distribution map and empirically determining priorities using approximation insertion. The positive/negative greedy method was applied to fulfill the terminal time constraints, and the Gaussian mixture model-approximation insertion method was employed to generate the optimal search path heuristically under the constraints. Their study highlighted the importance of using probabilistic models and heuristic methods to increase the efficiency of UAV search and coverage missions in riverine environments. More recently, the authors of [28] proposed an autonomous coverage path-planning algorithm using deep reinforcement learning to maximize success rates in a limited timeframe. In their approach, each unit navigates directly to the grid with the highest probability in its block using an environmental map established based on the drift simulation results obtained at a given time. However, they assumed a static search environment during path planning, a significant limitation given the dynamic nature of riverine and aquatic environments. This assumption can lead to suboptimal performance in real-world scenarios where environmental conditions change rapidly.

To the authors' knowledge, multiple UAVs systems for water management have not yet been reported in the literature. In [7], an SCT-based controller was proposed for swarm UAVs, and the performance of the proposed controller was verified via an experiment in which a group of UAVs was clustered using open-source software. In [29], UAV control was realized using a model predictive control scheme for precise stabilization during water sampling. Moreover, the authors of [30] proposed a general system architecture with onboard vision-based navigation and decision-making for remote sensing and precision agriculture. In this architecture, the onboard computer running a detection algorithm transmits the target's coordinates (local position (x, y, z)) to a Pixhawk device based on the target's position relative to the rotation matrix node (x, y) and its height relative to the ultrasonic node (z). Waypoints are created and driven in this control architecture. However, this architecture does not account for complex scenarios, such as creating multiple goals; therefore, it can create only fragmented situations.

Additionally, to date, SCT-based multiple UAV systems for ecosystems and natural systems have not been reported in the literature. Therefore, this research reviews related studies on formal methods for multirobot control, particularly SCT. SCT has traditionally been applied to manufacturing automation [31]; however, its application has recently been integrated into dynamic cyber-physical robotic systems. For instance, in [32], the authors implemented SCT in agricultural UAVs. Moreover, in [33], the authors developed a control architecture for deploying heterogeneous multirobot teams in urban search and rescue operations and presented the results of their experiments. Most studies have used simple supervisory control based on an automata model without addressing the challenges of integrating an MRS with a CTS. Studies that have developed application-focused DES-based supervisory control and dynamic systems have not described guaranteed solutions to heterogeneity and scalability problems nor proposed a systematic approach to complex dynamic systems. Therefore, additional research that focuses on approaches beyond the traditional control methods for field robots is required to control, implement, and maintain large-scale dynamic systems using DES and SCT.

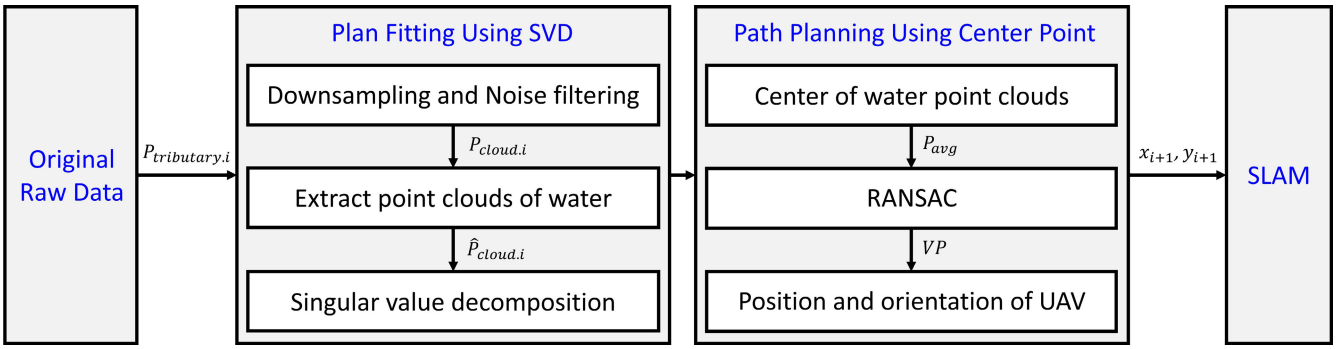


FIGURE 2. Tributary perception flowchart.

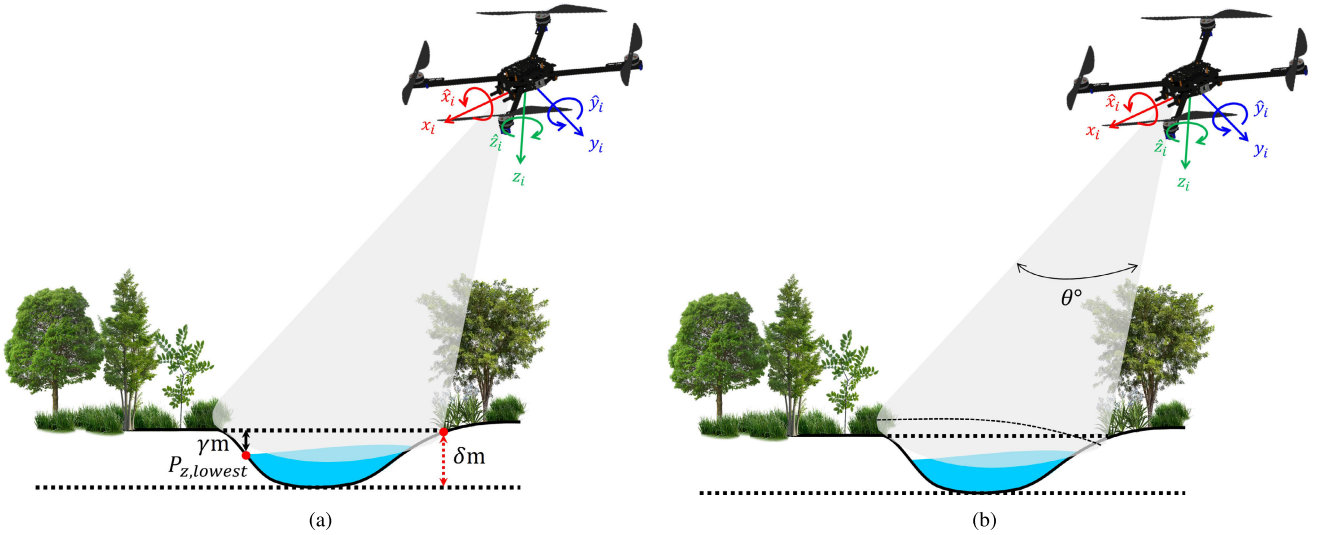


FIGURE 3. Side view of UAV path planning and navigation for a tributary: (a) tributary point cloud parameter and (b) spatial constraints with $\pm\theta°$.

III. PERCEPTION SYSTEM

A. LIDAR SEGMENTATION

LiDAR point cloud segmentation spatially groups points with similar properties into homogeneous regions. Although surface water typically absorbs the near-infrared beams emitted by LiDAR, we focus on this characteristic to propose a novel approach to segmenting surface water in aquatic environments.

Tributary segmentation based on LiDAR point cloud data follows the sequence depicted in Fig. 2. A LiDAR point cloud is a set of points spread out in 3D space. A 3D LiDAR point cloud is downsampled for efficient processing. After voxelization, the downsampled 3D LiDAR point cloud is defined as $P_{cloud,i} = [P_{x,i}, P_{y,i}, P_{z,i}]$, where $P_{x,i}, P_{y,i}, P_{z,i}$ with the coordinates x, y, z. Then, we established a grid size of α m and removed any point clouds if their count in a voxel was below a threshold β for the minimum number of point clouds required in one voxel. However, the noise-filtered point cloud still contains obstacles, such as the bushes around the tributaries. Therefore, we defined a surface water point cloud to segment the surface water and generate paths accordingly.

We set specific parameters using the LiDAR point cloud data, as depicted in Fig. 3, to segment the tributary area, which is defined as follows:

$$P_{tributary,i} = \begin{cases} \hat{P}_{cloud,i} & \text{if } P_{z,i} < P_{z,lowest} + \gamma \ \ and \ \ P_{z,i} < \delta \\ (0,0,0) & otherwise, \end{cases} \tag{1}$$

where the tributary area $P_{tributary,i}$ is estimated by adding γ m to the lowest point in the point cloud data $P_{z,i}$. In this manner, the tributary region is defined with the $P_{z,i}$ values of the point cloud plus γ m. Additionally, the height of the tributary area cannot exceed δ m. These parameters are empirical values, and we assume that the point clouds satisfying these two conditions represent surface water. Then, the tributary regions are segmented via singular value decomposition (SVD)-based plane fitting [34] of the filtered point cloud. The SVD

is a matrix decomposition method that can decompose an arbitrary $m \times n$-dimensional matrix M, where M is defined as follows:

$$M = U\Sigma V^t, \tag{2}$$

where U is an mxm-dimensional orthogonal matrix, Σ denotes an mxn-dimensional diagonal matrix, and V^T represents an nxn-dimensional orthogonal transpose matrix. In addition, U contains left singular vectors, providing information about the column space of the original matrix. The diagonal components of Σ represent singular values and play a crucial role in the SVD of the original matrix. Because Σ is a diagonal matrix, all its off-diagonal elements are zero. Further, V^T contains right singular vectors, providing information about the row space of the original matrix. The projection matrix obtained through the SVD is as follows:

$$\begin{cases} P_{tributary,i}P^T_{tributary,i} & = (USV^T)(VS^TU^T) \\ & = USS^TU^T, \\ P^T_{tributary,i}P_{tributary,i} & = (VS^TU^T)(USV^T) \\ & = U^TUSS^T, \end{cases} \tag{3}$$

where the cross product and inner product of $P_{tributary,i}$ are obtained using SVD. The SVD demonstrates the projection of a 3D vector into a two-dimensional (2D) vector using Eq. (3).

Some point clouds may not be acquired owing to occlusion caused by external factors (e.g., wiring). Based on the heading direction of UAV x_i, we searched for the drivable area in the range of $\pm\theta^\circ$ within the horizontal view of the LiDAR sensor, as depicted in Fig. 3(b).

B. NAVIGATION AND SIMULTANEOUS LOCALIZATION AND MAPPING

Semantic simultaneous localization and mapping (SLAM) extends the capabilities of traditional SLAM by incorporating geometric and semantic information, providing detailed insight into the objects and spaces in an environment. In tributary perception, a LiDAR-equipped UAV detects tributaries in a dynamic environment based on the path generated based on the segmentation results as the control input. Path planning for UAVs applies the median value of the tributary area. The median value of a tributary point cloud is defined as follows:

$$P_{avg} = \frac{1}{m}\sum_{j=0}^{m} P_{cloud.i}, \tag{4}$$

where m represents the number of segmented point cloud data. Path planning based on tributary segmentation is executed in real time, and multiple control inputs are input indiscriminately. Therefore, the estimated tributary area is calculated in the range r_1 m to ensure a stable search, and the segmentation range is r_2 m, as depicted in Fig. 4. In addition, stable control inputs are generated using the random sample consensus (RANSAC) algorithm. A threshold T is defined, and the points in this threshold are considered inliers. These inliers form a stable path, whereas the points beyond the threshold or those identified as duplicates are excluded to ensure the path is devoid of distant or redundant points. The probability values used in the RANSAC algorithm are as follows:

$$\rho = 1 - (1 - \epsilon^q)^Q, \tag{5}$$

where Q denotes a RANSAC iteration, q represents the number of samples drawn at one time, ϵ indicates the ratio of inliers in the input data, and ρ denotes the probability that a sample is selected from an inlier at least one out of Q times. The success of RANSAC relies on drawing sample data exclusively from inliers in at least one out of Q attempts. The probability of achieving this success increases as Q increases. However, the number of iterations is typically determined probabilistically by considering the practical constraints on indefinite execution owing to the finite nature of RANSAC iterations. The number of iterations of RANSAC is expressed in Eq. (5).

$$Q = \frac{log(1-\rho)}{log(1-\epsilon^q)}. \tag{6}$$

For control inputs, the UAV position is denoted as $P_i = [U_x, U_y, U_z]$. The UAV movement direction is represented by the vector $VP = [\hat{x_i}, \hat{y_i}]$, which provides crucial information guiding the UAV along a specific trajectory. The proposed system estimates the direction of UAV movement in 2D space. The components of VP are defined as follows:

$$\begin{cases} \hat{x_i} = P_{avg,x} - x_i, \\ \hat{y_i} = P_{avg,y} - y_i, \end{cases} \tag{7}$$

where $P_{avg,x}$ and $P_{avg,y}$ denote the x and y coordinates of VP, respectively, and VP provides a directional reference based on the current coordinates of the UAV relative to the average target position. The distance d from the current UAV position P_i to VP is calculated using the Euclidean distance formula derived from the Pythagoras theorem, as depicted in Fig. 4(b) and defined below:

$$d = \sqrt{(\hat{x_i})^2 + (\hat{y_i})^2}. \tag{8}$$

For $d > 0$:

$$\begin{cases} \hat{x_i} = \hat{x_{i-1}}/d, \\ \hat{y_i} = \hat{y_{i-1}}/d, \end{cases} \tag{9}$$

where vector normalization is performed to scale the vectors to fit the moving distance. The UAV's position is updated, as defined in the following equation:

$$\begin{cases} x_{i+1} = x_i + C\cdot\hat{x_i}, \\ y_{i+1} = y_i + C\cdot\hat{y_i}, \end{cases} \tag{10}$$

where C denotes a constant value representing the UAV's speed, which can be adjusted by modifying C, helping the UAV avoid obstacles or move toward a specific point.

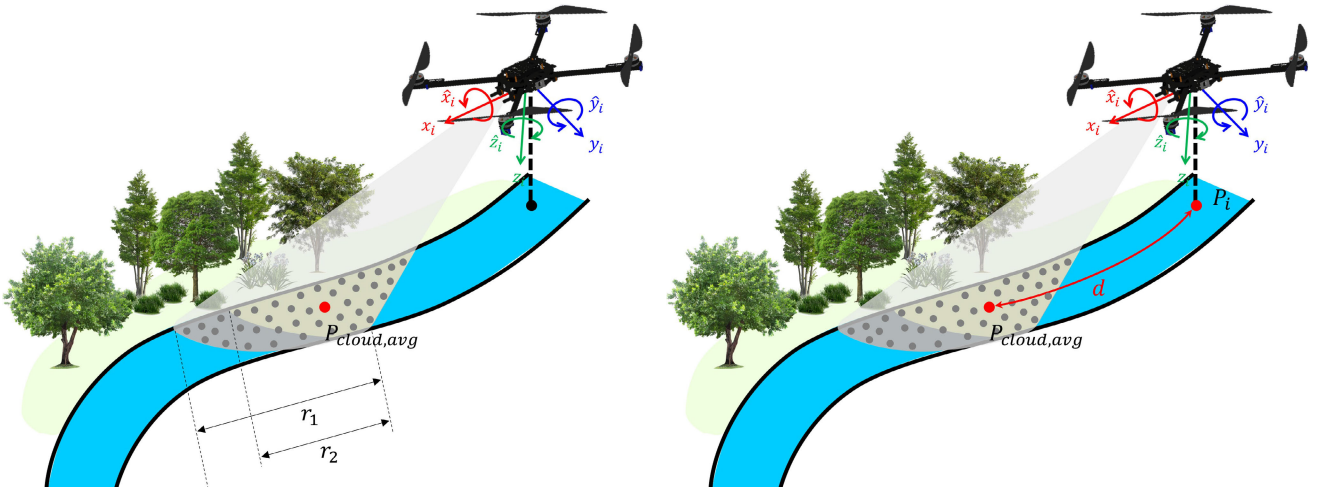


FIGURE 4. Top view of UAV path planning and navigation for tributary: (a) extraction of the average point cloud from spatial constraints of r_1 and r_2 and (b) distance calculation.

Over the decades, SLAM has been developed to be a robotics capability. This study employs fast-LiDAR odometry and mapping (F-LOAM) [35] owing to its high computational efficiency and localization accuracy. The F-LOAM algorithm formulates a sensor model and extracts features. The extracted features are calibrated via distortion compensation. Then, the general and global feature extraction results are obtained, and the mapping process and outcomes of laser odometry calculations are explained. With noniterative two-step distortion compensation, F-LOAM facilitates robust, real-time path planning for UAVs at a lower computational cost than that of traditional LOAM [36].

IV. MULTIPLE UAVS CONTROL SYSTEM

A. SUPERVISORY CONTROL THEORY

This section provides brief overviews of SCT and DES (see [37] for more details). A DES is a dynamic system with continuous-time and discrete state space characteristics. In addition, a DES is an event-driven system whose state is thoroughly transitioned by eligible events over time. In this study applies the automata theory as the discrete event modeling formalism to explain system behavior.

The finite state automaton G for modeling a DES is a tuple consisting of the following five elements [37]:

$$G = \{Q, \Upsilon, \zeta, q_0, Q_m\}, \tag{11}$$

where A denotes the set of all states, Υ represents the set of all events, ζ is the state transition function of G ($\zeta : Q \times \Upsilon^* \mapsto Q$), a_0 indicates the initial state of G, and Q_m denotes the subset of marker states, indicating a goal state or the final state ($Q_m \subset Q$). In the transition function ζ, Υ^* represents a sequence (string) of events containing the null event ε. Moreover, the event set Υ is divided into a set of controllable events Υ_c and a set of uncontrollable events Υ_{uc}.

The language occurred by the automaton G is defined as follows:

$$L(G) := \{s \in \Upsilon^* | \zeta(q_0, s)!\}, \tag{12}$$

where $\zeta(a_0, s)!$ indicates that the next state in which string s occurs at a_0 is defined in G. The prefix closure of language $L(G)$ is defined as follows:

$$\overline{L(G)} := \{t \in \Upsilon^* \mid t \leq s \quad \exists\, s \in L(G)\}, \tag{13}$$

where $L(G)$ is defined as prefix-closed when $L(G) = \overline{L(G)}$.

The marked language of automaton G is defined as follows:

$$L_m(G) := \{s \in L(G) \mid \zeta(a_0, s) \in Q_m\} \subseteq L(G), \tag{14}$$

if G satisfies $\overline{L_m(G)} = L(G)$, and $L(G)$ is nonblocking. The marked state can be reached after any string occurs in any state of G [37]. This nonblocking condition is necessary for designing a proper supervisor in SCT, because the DES may fall into a deadlock or livelock when $L(G)$ is blocking.

The supervisor is defined as the automaton $S = (X, \Upsilon, \delta, x_0, X_m)$, where X, Υ, δ, x_0, and X_m represent the sets of states and events and the state transition function, initial state, and marker state, respectively. The plant is defined as DES G, and the behavior, generated language, and marked language of plant G under supervisor S are defined as follows:

$$S/G = \{X \times Q, \Upsilon, \delta \times \zeta, (x_0, a_0), X_m \times Q_m\}, \tag{15}$$

$$\begin{aligned} L(S/G) : \epsilon \in L(S/G), \forall s \in \Upsilon^*, \\ \epsilon \in \Upsilon : s \in L(S/G), \\ s\upsilon \in L(G), \upsilon \notin \Theta \Rightarrow s\upsilon \in L(S/G), \end{aligned} \tag{16}$$

where υ is an eligible event, and Θ represent a control mapping function defined as $\Theta : L(G) \mapsto 2^{\Upsilon_c}$. The supervisory control problem (SCP) used to design the supervisor is defined as follows:

Definition 1: For a given $K \subseteq G$, find a supremal language S that is controllable *w.r.t.* (G, Υ_{uc}), and satisfies $L(S/G) = K$ and $L(S/G) = \overline{L_m(S/G)}$.

Thus, if K is defined as the specification for G, the supervisory control problem is to find a supervisory controller that satisfying $L(S/G) = K = \overline{L_m(S/G)}$ and is nonblocking, and controllable *w.r.t.* G. In this case, many supervisors may satisfy the specifications and be controllable. Among these K, the supremal controllable sublanguage of K is identified as the solution to the supervisory control problem. Therefore, S can allow for the maximal occurrence of the eligible language in G.

B. HYBRID SYSTEM OF MULTIPLE UAVS

This section presents a control method for multiple robots from the perspective of this hybrid system. This approach allows for integrating continuous control dynamics and discrete event-driven logic. The proposed hybrid system is modeled by integrating a CTS with the DES framework. A hybrid automaton was proposed in [38] as a formal framework for modeling hybrid systems. A hybrid automaton provides a mathematical structure that represents the combined continuous and discrete dynamics of a system, thereby facilitating precise analysis and control design. We aim to enhance collaboration and coordination between multiple UAVs in the proposed tributary mapping system by adopting this hybrid system approach. This method applies the strengths of both continuous control and discrete event logic to provide a robust and flexible framework for managing complex tasks in dynamic environments.

With the proposed approach, we facilitate collaboration between field robots by designing a controller for a tributary mapping system comprising multiple UAVs. At this stage, we configure a controller based on a hybrid system of multiple UAVs to achieve this objective. The advantages of using a hybrid system in multirobot collaboration are manifold. Integrating continuous-time dynamic-based low-level models with discrete event-time dynamic-based high-level models, which define the hybrid system, ensures overall system performance via a bidirectional interaction. High-level models, governed by discrete-timed dynamics, issue commands, such as waypoints or task assignments, whereas low-level models process these commands based on continuous-time dynamics and provide real-time state feedback. This interaction enables control in dynamic environments by employing the decision-making capabilities of high-level models and the precise execution of low-level models. A modular supervisory controller facilitates decentralized control, allowing dynamic role assignment and scalable integration of additional UAVs without major architectural modifications. This approach enhances robustness, efficiency, and scalability, making it suitable for complex multiple UAVs systems.

This study explains the concept of a hybrid system of robots using the following example. Typically, UAVs incorporate a low-level controller that manages the continuously tuned dynamics. A typical example of such a low-level controller is the PID controller. For instance, the continuous-time aspect of a UAV hybrid system includes motor control using low-level controllers. The PID controller continuously adjusts motor speeds for a quadrotor to maintain stability and achieve the desired flight dynamics. The high-level control aspect involving DES includes events, such as power-off triggered by specific conditions, sensor inputs, or state transitions. A hybrid system combines these two components, featuring a structure that allows real-time control based on event triggers that can occur in continuous time. This integration enables the system to handle complex scenarios by applying continuous adjustments and discrete events, providing robust and efficient control over UAV operations.

The hybrid automaton G_h is a tuple consisting of the following elements [39]:

$$G_h = (E, X, \Omega, U, F, \phi, Inv, Guard, \rho, E_0, X_0), \tag{17}$$

where E denotes the set of discrete states, X represents the set of continuous states, Ω indicates the set of events, U is the set of admissible controls, F represents the vector field of G_h ($F : E \times X \times U \to X$), ϕ is the discrete state transition function of G_h ($\phi : E \times X \times \Omega \to E$), Inv denotes the set defining an invariant condition ($Inv \subseteq E \times X$), $Guard$ represents the set defining a guard condition ($Guard \subseteq \Omega \times E \times X$), ρ is the reset function ($\rho : \Omega \times E \times X \to E \times X$), E_0 denotes the initial discrete state and X_0 represents the initial continuous state.

The UAVs are modeled by including high-level (i.e., DES) and low-level (i.e., CTS) controllers and using kinematic and dynamic equations. The detailed low-level and high-level controller design for the hybrid system is discussed in the following sections.

1) LOW-LEVEL UAV MODELING

Low-level UAV models are formulated using CTS. We consider N low-level UAVs with three degrees of freedom. The position of each UAV is denoted by $p_i \in \mathbb{R}^3$, where $i = 1, 2, \ldots, N$. The flight control input of each UAV in terms of dynamics and kinematics can be expressed as follows:

$$m_i \ddot{p}_i = -\tau_i \xi_i e_3 + m_i g e_3 + \mathrm{h}_i, \tag{18}$$

$$\mathcal{J}_i \dot{w}_i + S(w_i)\mathcal{J}_i w_i = \lambda_i + \mathrm{a}_i, \tag{19}$$

$$\dot{\xi}_i = \xi_i S(w_i), \tag{20}$$

where $m_i > 0$ denotes the mass of each UAV, $p_i := [p_1; p_2; \ldots;_N] \in \mathbb{R}^3$ is the Cartesian center-of-mass position in the north-east-down (NED) inertial frame, $\tau_i \in \mathbb{R}$ indicates the thrust control input along e_3 (indicating the N, E, and D directions), ξ_i denotes the rotational matrix describing the body frame of the UAV with respect to the inertial frame, and g represent the gravitational constant. In addition, $\mathcal{J}_i$ denotes the inertia matrix *w.r.t.* the body frame. Further, w_i represent the angular speed of the UAV relative to the inertial

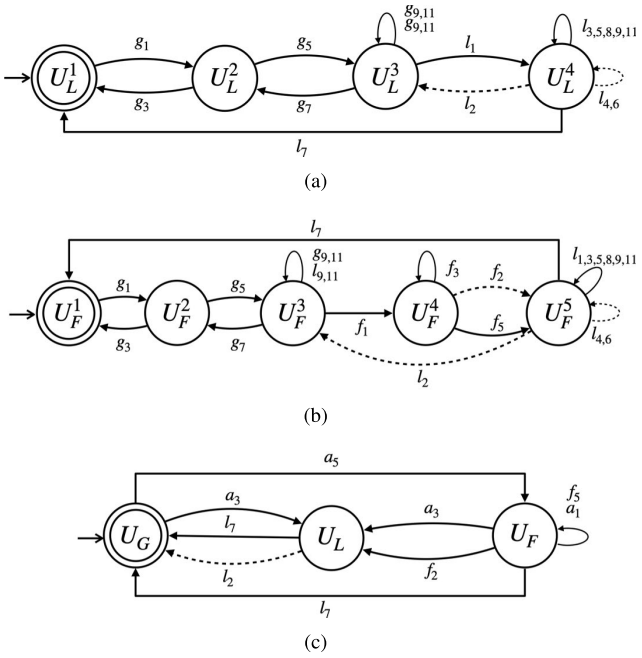


FIGURE 5. Plant model: (a) leader UAV model, U_L^i, (b) follower UAV model U_F^n, and (c) assign model G_A.

frame, and $\lambda_i \in \Re^3$ indicates the attitude torque control input. Moreover, H_i, $a_i \in \Re^3$ denotes the aerodynamic perturbations of a typical UAV in flight, where H_i, $a_i \approx 0$.

The multiple UAVs system is controlled using a leader-follower formation control strategy suitable for unstructured environments. Formation control is implemented based on the desired leader-follower distance $d_{i,j}$. In formation control, the leader and follower are assumed to know each other's states and that communication between them is always maintained. Each robot i, where $i = \{1, \ldots, n\}$, can estimate its position p_i using a GPS module and inertial measurement unit (IMU). From p_i and p_j, robots can calculate the relative distance $l_{i,j} = p_i - p_j$. This leader-follower formation can cope with unstructured environments by assigning new leaders at a split tributary.

2) HIGH-LEVEL UAV MODELING

High-level UAVs are modeled using DESs (hybrid automata). In detail, a multiple UAVs system consists of leader, follower, and assign models, depending on the control strategy, as illustrated in Fig. 5. The double circles represent marked states, and the dotted lines represent uncontrollable events. The proposed hybrid automaton model includes states that represent different UAV operation modes, such as flying, hovering, and landing, and transitions between these states are triggered by discrete events, such as reaching a waypoint or detecting an obstacle. The low-level PID controllers govern the continuous dynamics in each state, ensuring responsive UAV control.

The high-level leader UAVs are modeled as the automata U_L^i. The leader UAV states are defined as $U_L^i = \{U_{L,i}^1, U_{L,i}^2, U_{L,i}^3, U_{L,i}^4\}$, for $i \in \{1, 2, \ldots, n\}$ (where $U_{L,i}^1$: ideal, $U_{L,i}^2$: arming, $U_{L,i}^3$: hovering, and $U_{L,i}^4$: performing task). Each state includes a low-level model that describes the UAV altitude, h_i, velocity v_i, and distance between the leader UAV and a follower UAV, d_i. For example, we consider the state set $h_i, v_i, d_i = 0$ of $U_{L,1}^1$, where $d_i = 0$ indicates that the role to be played by this UAV is not assigned.

The leader UAV model consists of a set of the general and leader UAV states, where $U_{L,i}^1$, $U_{L,i}^2$, and $U_{L,i}^3$ are general states (i.e., the set of states before an event *start mission*). For example, if an event *start mission* occurs in state $U_{L,i}^3$, this state transitions from the general to the leader UAV state. At this time, because all UAVs are in the general state, multiple leader UAVs can be created, making it possible to conduct searches over split tributaries.

The high-level follower UAVs are modeled as automata, where U_F^n represents the UAV states and events. The states of the follower UAVs are defined as $U_F^n = \{U_{F,n}^1, U_{F,n}^2, U_{F,n}^3, U_{F,n}^4, U_{F,n}^5\}$, for $n \in \{1, 2, \ldots, k\}$, in state $U_{F,n}^1$ (where $U_{F,n}^1$: ideal, $U_{F,n}^2$: arming, $U_{F,n}^3$: hovering, $U_{F,n}^4$: following leader UAV, and $U_{F,n}^5$: task performing). The follower UAV model includes the set of general states, leader UAV states, and follower UAV states.

We included the leader UAV state in the follower UAV model to ensure that a follower UAV can transition to being the leader UAV to manage scenarios in which a split occurs. If the leader UAV detects a split zone during a mission, a follower UAV transitions to being a leader UAV to sustain the exploration process. Similarly, if a follower UAV loses communication with its leader UAV, the follower is transitioned to being a leader UAV to cover a tributary. For this reason, the follower UAV model consists of the general state, follower UAV state, and leader UAV state. The general state is the state before the assignment of follower UAVs. The follower UAV states $U_{F,n}^1$, $U_{F,n}^2$, and $U_{F,n}^3$ represent the general state, and $U_{F,n}^4$ denotes the follower state. Finally, $U_{F,n}^5$ represents the leader state.

The assign model states are $G_A = \{U_G, U_F, U_L\}$ (where U_G: general state, U_F: follower UAV state, and U_L: leader UAV state). In addition, G_A assigns a state to each UAV, meaning that the state of each UAV is determined as the general state. The general state is defined as the state before allocating the leader or follower model to each UAV.

We can obtain the plant model corresponding to a high-level UAV model via the following simple synthesis:

$$G_{plant} = U_L \,||\, U_F^n \,||\, G_A. \tag{21}$$

where G_{plant} is determined by the number of UAVs initially consisting of a leader j, followers $n \in 1, \ldots, n$, and an assign model. The plant model can be obtained relatively simply through this parallel synthesis process, even when UAVs are added to G_{plant}.

TABLE 1. Events and states of each UAV.

State	Event	Description	Controllable
General	g_1	Arming	○
	g_3	Disarming	○
	g_5	Take off	○
	g_7	Landing	○
	g_9	Hovering	○
	g_{11}	Move to tributary	○
Leader	l_1	Start mission	○
	l_2	Finish mission	×
	l_3	Move to narrow tributary	○
	l_4	Tributary detected	×
	l_5	Send message to follower UAV	○
	l_6	Tributary not detected	×
	l_7	Return to home	○
	l_8	Tributary detected more than two	×
	l_9	Altitude up	○
	l_{11}	Altitude down	○
Follower	f_1	Following leader UAV	○
	f_2	Lost leader	×
	f_3	Following sub-leader UAV	○
	f_5	Received message from leader UAV	○
Assign	a_1	Assigned sub-leader UAV	○
	a_3	Assigned leader UAV	○
	a_5	Assigned follower UAV	○

C. SUPERVISORY CONTROLLER

To design the supervisory controller, we first design its specifications (control objectives) and then identify the supremal controllable sublanguage of the multiple UAVs system, G_{plant}. The control objectives corresponding to the design specifications (S_p^i) are as follows:

- S_p^1: Performing mission
- S_p^2: Navigation
- S_p^3: Grouping

where S_p^1 implies that the leader UAV moves to a tributary in the hovering state before the mission, and S_p^2 implies that the leader UAV can increase its altitude to explore whether the entire tributary is undetected or path planning is not achieved during a mission. Finally, S_p^3 denotes the division of UAVs into groups and subgroups for assigning leaders to each group and the attempt by a UAV to reorganize if it loses communication with the leader UAV.

Fig. 6 depicts obtaining the supervisory controller for the proposed hybrid control system. First, by synthesizing the plant (G_{plant}) and specification (S_p^i), numerous sublanguages (K_i) are created, each undergoing a process to check its property, ultimately deriving a supervisor. The supervisor is a controllable, supremal sublanguage defined by SCT among the created sublanguages. A modular supervisory controller S_i is depicted in Fig. 7, corresponding to the obtained K_i that satisfies controllability. The controller is designed to avoid conflict to satisfy the control objective. This supervisory controller controls the system through control actions. For example, Fig. 7(c) considers the occurrence of events, such that g_1, g_5, f_1, and a_3 respond to states q_0, q_1, q_4, q_7, and q_{11}, respectively. Then, q_{11} is a deadlock state in which the desired state cannot be reached regardless of which event occurs. Therefore, the supervisory controller implements a control action to ensure system stability by disabling the transition of event f_1 from q_7 to q_{11}.

TABLE 2. Hardware specifications.

Hardware	Specifications
Quadcopter	F660
LiDAR	VLP-16
Companion board	Jetson AGX Xavier
Flight controller	Pixhawk 6C
Telemetry	SiK Telemetry Radio V3
GPS	M9N

TABLE 3. Specification of 3D LiDAR.

Velodyne VLP-16	Specifications
Channels	16
Measurement range	up to 100 m
Accuracy	± 3 cm (typical)
Field of view (vertical)	2°
Field of view (horizontal/azimuth)	360°
Angular resolution (vertical)	2°
Angular resolution (horizontal/azimuth)	0.1 – 0.4°
Rotation rate	5 – 20 Hz

V. EXPERIMENTAL SETUP

The general architecture of the multiple UAVs system in the experiments is depicted in Fig. 9. We apply 4G LTE for multiple UAVs systems because communication delays can occur depending on the environment. Communication between UAVs based on a cellular network ensures reliable data exchange. This ensures wide coverage, high-speed transmission, low latency, and stable connectivity for real-time UAV coordination in dynamic environments. Middleware built on the robot operating system facilitates seamless integration and coordination among UAVs, enabling real-time computation of critical state information, including current positions and inter-UAV distances. Additionally, the architecture is configured with MAVLink to support lightweight and efficient communication, further enhancing the ability of the system to manage state information and enabling robust and precise multiple UAVs collaboration.

We employed a UAV platform, specifically a quadcopter, comprising a carbon-fiber frame integrated with onboard computers and various sensors, as depicted in Fig. 8 and Table. 2. In unstructured outdoor environments, the efficacy of visual inertial odometry systems is compromised due to fluctuating illumination conditions from sunlight variation. In contrast, 3D LiDAR sensors (Table. 3) offer superior performance by providing dense and precise range measurements over long ranges while requiring minimal computational resources compared with other sensors, such as high-resolution cameras and multispectral sensors. This advantage makes LiDAR indispensable for such tasks as autonomous navigation and precise semantic mapping in surface water environments. Notably, the platform boasts a payload capacity (excluding the battery) of 3.5 kg and can

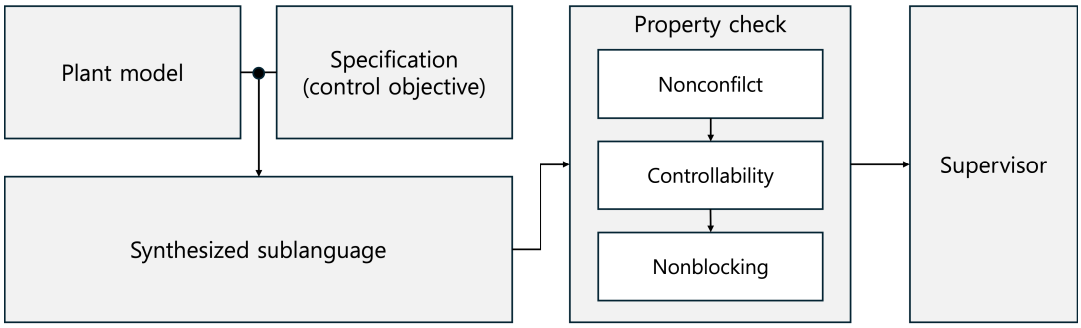


FIGURE 6. Process of designing the supervisor.

TABLE 4. Results of the perception field test.

	Environment 1		Environment 2	
	Width	Height	Width	Height
Ground truth	7.10 m	3.10 m	7.10 m	3.10 m
Results	7.21 m	3.15 m	7.84 m	3.21 m
Error	0.11 m	0.05 m	0.74 m	0.11 m

be flexibly reconfigured to accommodate additional onboard computers and mission-specific sensors.

VI. EXPERIMENTAL RESULTS

A. EXPERIMENTAL RESULTS OF THE PERCEPTION SYSTEM

We set the operational parameters as follows. The grid resolution was established as $\alpha = 1$ m, and point clouds were filtered from the grid if their count fell below $\beta = 3$. Additionally, the tributary areas were estimated by adding a constant $\gamma = 0.5$ m to the lowest point in the point cloud data, and the height of the tributary area did not exceed $\delta = 1$ m. The calculated tributary area was bound in the range $r_1 = 20$ m. Moreover, the segmentation area was confined in the variable range of $r_2 = 10$ m. Finally, using the UAV's heading point as a reference, we searched drivable regions of the angular range of $\pm\theta = 4°$.

Experiments were executed in two canal environments at various lighting conditions. The first environment was assessed on a sunny morning (Fig. 10(a)), and the second environment was evaluated during challenging conditions, characterized by backlighting during a cloudy evening (Fig. 10(b)). From the semantic SLAM result map, we can generate a surface water map in real time, which is represented by the green plane, and can classify floating objects on the surface water as obstacles, indicated in red. As summarized in Table. 4, the width and height of both environments are 7.1 m and 3.1 m, respectively. In addition, the length of the environment is about 300 m and 600 m, respectively. Errors occur in the presence of obstacles such as bushes or sharp curves on the water surface, but the overall average error is close to the ground truth. This adaptability is crucial for addressing the diverse canal widths in agricultural environments. Moreover, the performance of the proposed system under various lighting conditions with strong sunlight and backlighting highlights its robustness in less-illuminated scenarios, ensuring consistent mapping capabilities even in suboptimal weather conditions.

The results demonstrate that the proposed LiDAR-based system effectively enables real-time mapping and navigation in complex agricultural environments. The system allows for real-time detection and monitoring of blockages or obstructions in water flow by identifying floating objects. Additionally, the system displays the potential for flood and drought preparedness by accurately measuring the width and height of waterways. Its ability to address variable lighting conditions and accurately detect obstacles on the water surface highlights its robustness and reliability, making it a practical solution for precision agriculture and surface water management.

B. EXPERIMENTAL RESULTS OF THE CONTROL SYSTEM

The approach of simulating virtual environments to replicate the physical environment and assess system performance under controlled conditions has been widely applied, as demonstrated in [40]. This approach effectively identifies and mitigates potential problems before real-world implementation. Therefore, in this study, physics-based simulations were formulated to validate the proposed supervisory controller for a multiple UAVs-based tributary mapping system. This system comprises three UAVs, and the virtual environment was modeled to include real split tributaries (Fig. 11). The system was fully implemented in CoppeliaSim, a robot simulator. The state and event transitions of the multiple UAVs were received and recorded using the CTS-to-DES interfaces in MATLAB and CoppeliaSim.

The experiment was conducted considering the following scenarios. We focused on tributaries divided into three to evaluate the control performance of the MRS. This experiment focused on verifying the supervisory controller; therefore, tributary recognition was not performed. To this end, a predefined path was set for each UAV. These predefined paths were transmitted to the leader UAV. Upon receiving the trajectory, the leader UAV commenced its

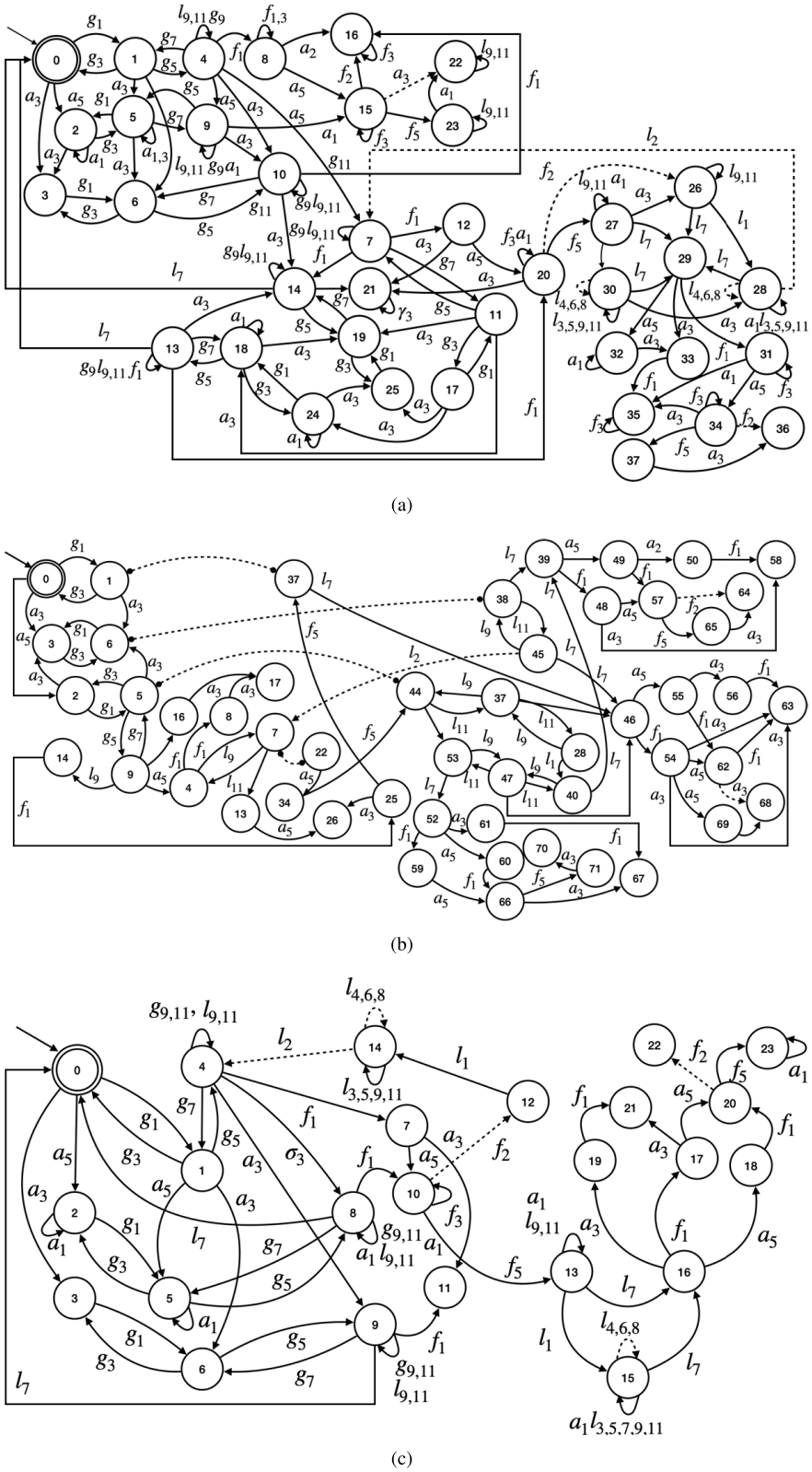


FIGURE 7. Synthesized modular supervisor according to specification: (a) modular supervisor 1 (S_1), (b) modular supervisor 2 (S_2), and (c) modular supervisor 3 (S_3).

path-following task and mapped the tributaries. The supervisory controller, designed based on the three established control goals, coordinated the movements and interactions of the UAVs.

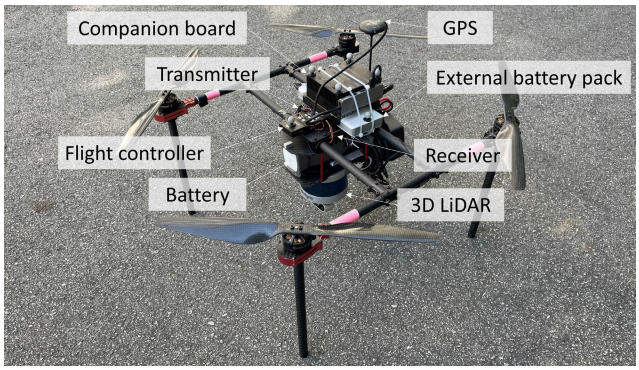


FIGURE 8. Hardware configuration.

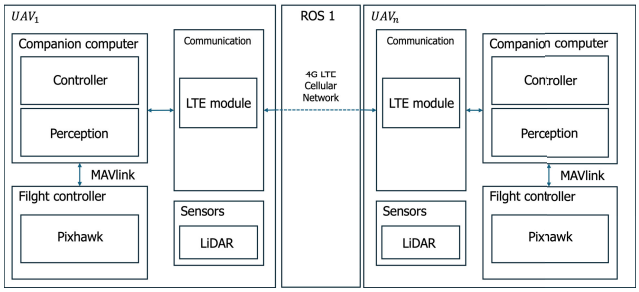


FIGURE 9. Proposed system architecture.

A limitation of this scenario is that the supervisory controller cannot independently identify branching points along tributaries. An experiment was conducted assuming that the separation points of the tributaries were known a priori to address this limitation. According to the experimental results, the leader UAV moved along the tributary and encountered a branching point. A specific event (l_8) was triggered upon reaching a predetermined splitting point. Per the predesigned S_p^3, the UAVs were divided into new groups by a supervisor, ensuring that the control specifications of each group were met. The follower UAVs mapped tributaries by responding to the branching points that occurred when an event triggered the assignment of a new leader. The results, illustrated in the accompanying figure, were evaluated based on events and state transitions over time, confirming that the behavior of the multiple UAVs system was consistent with the specified behavioral requirements.

C. EXPERIMENTAL RESULTS OF THE INTEGRATED SYSTEM

These experimental scenarios were employed to assess whether the MRS could successfully perform mapping in an unknown environment while meeting the design specifications. This experiment evaluated the performance of the proposed perception system and the designed supervisory controller. The computational complexity of the controller used in a traditional differential equation-based MRS increases rapidly as the number of robots increases. We tested and verified the scalability of the proposed system in three cases to address this scalability problem:

- Case 1: single UAV,
- Case 2: three UAVs, and
- Case 3: eight UAVs.

Figs. 12-14 depict the experimental results. The tributary perception and SLAM performance of the system were evaluated based on the UAV's trajectories. Perception and SLAM were performed independently by the leader carrying out the mission, and the follower responded to scenarios involving a split tributary.

The performance of the supervisory controller and scalability of the proposed hybrid system were evaluated regarding the time-varying state transitions and event strings. Experiments were conducted with different numbers of UAVs to demonstrate that the system based on the proposed method has adequate scalability. The entire plant model can be obtained easily via additional simple synthetic calculations and controlled using the designed supervisor. As the number of UAVs increases, the state space, events, and transitions of the overall plant increase exponentially. Case 1 (single UAV) consists of 5 states, 20 events, and 23 transitions; Case 2 (three UAVs) has 13 states, 23 events, and 67 transitions; and Case 3 (eight UAVs) expands to 8,125 states, 23 events, and 42,185 transitions. This exponential growth in the state space increases computational complexity; however, the proposed supervisory controller, implemented via a physics simulation, demonstrates sufficient scalability. Using a modular supervisor for decentralized control efficiently adapts to varying cases, reducing computational complexity and highlighting its scalability in large-scale MRSs.

Based on each case of the entire plant, the computational complexity of the proposed multiple UAVs system was analyzed based on its leader-follower control strategy. In this system, only the leader UAV performs path planning and SLAM algorithms in a 2D grid framework with a computational complexity of $O(n^2)$. The follower UAVs are limited to tracking the leader without performing additional computations. The modular supervisory controller enables decentralized control by dynamically assigning roles based on environmental conditions. Thus, the overall computational complexity of the system is expressed as $O(n^2 \cdot m)$, where m represents the number of leader UAVs. This approach highlights the system's structural efficiency and scalability.

Fig. 12 depicts the results of Experimental Case 1. In Case 1, the system could not cope with a split tributary; therefore, only the main tributary was selected and mapped. At this time, a single UAV could not cover the split zone, and consequently, it explored only one tributary. Fig. 13 presents the results for Case 2, where the leader and followers were assigned before the split occurred at l_1. At split Zone 1, the leader UAV (UAV1) detected two tributaries l_8. Then, the closest UAV among the followers was selected and assigned as the leader, l_1. Next, UAV1 moved to a narrow tributary and restarted its mission l_3 and l_1. Similarly, the new leader (UAV2) detected the second split tributary. Its mission was restarted at this point, and UAV3 was assigned as the leader

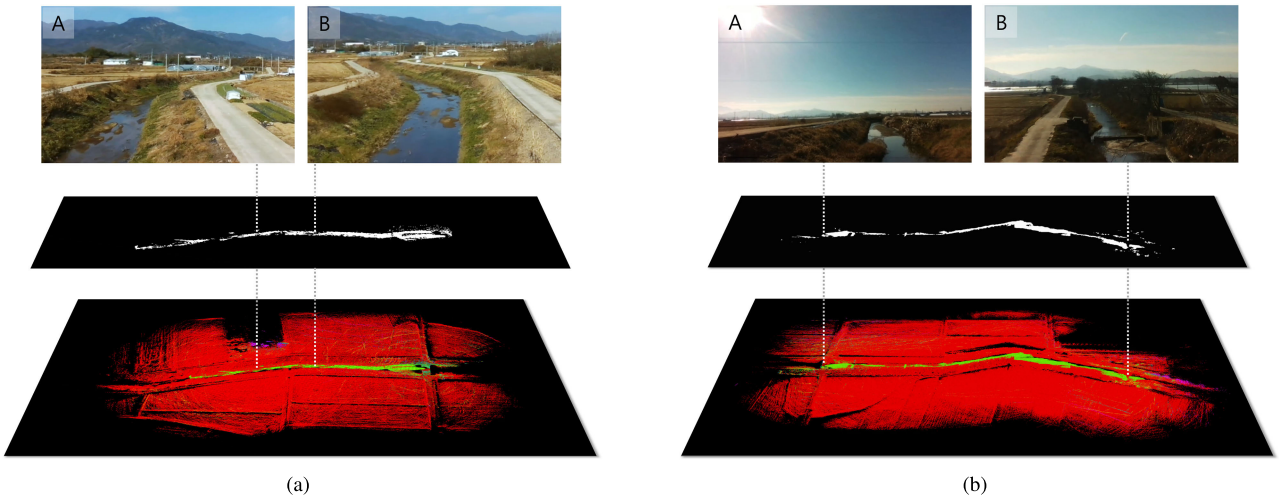


FIGURE 10. Evaluation of tributary perception system: experimental results for (a) a sunny day and (b) a cloudy evening with backlighting. The first layer illustrates the obstacles A and B of the tributary in a real environment. The middle layer provides a detailed surface water map. The final layer presents the tributary map achieved through simultaneous localization and mapping, allowing us to detect obstacles floating on the surface water.

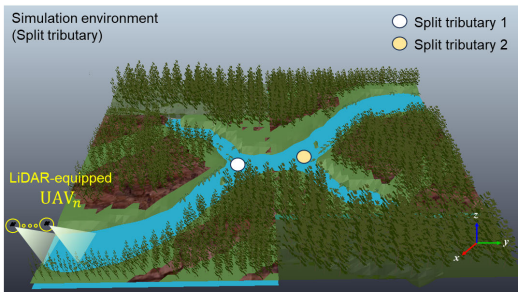


FIGURE 11. Experimental environment designed using a robot simulator similar to the actual environment.

UAV. Finally, in Case 3, as illustrated in Fig. 14, the multiple UAVs system achieved the behavior specifications despite the increase in the number of UAVs.

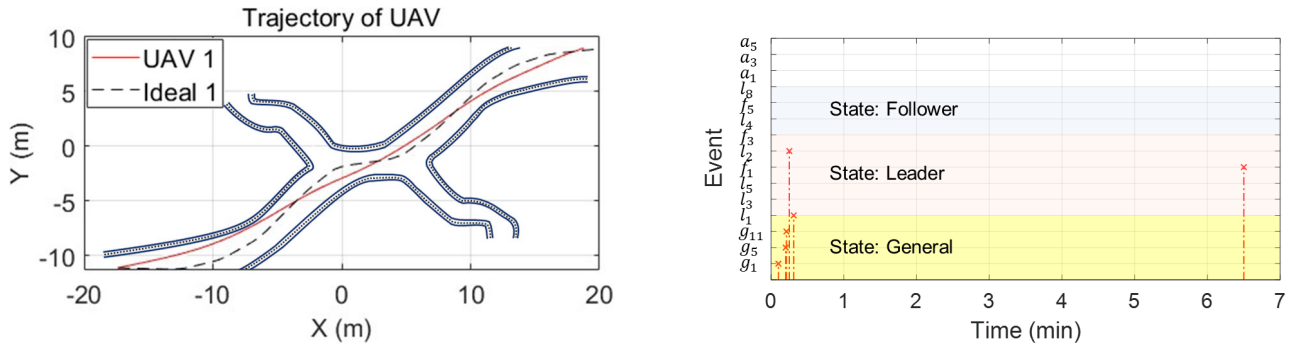
VII. DISCUSSION

This study did not include an assessment of the points at which tributaries branch in the tributary recognition system. In simulations, the divergence points were known beforehand, allowing for their identification. However, recognizing these divergence points is crucial in an entirely unknown environment. Future studies should focus on developing and integrating methods for real-time identification of tributary branching points to enhance the applicability of the system in real-world scenarios. Additionally, improving the accuracy and resolution of the mapping data further supports the effective navigation and mapping of complex river systems.

A simulation environment that rigorously mimics real-world conditions has been developed to facilitate field experiments in actual environments. However, because not all real-world variables can be accounted for in a simulation, uncontrollable factors may be encountered when the system is applied to real-world scenarios. For instance, power lines may be present around the tributary environment, requiring the system to navigate around them. Moreover, the system must be able to respond to unexpected incidents, such as bird strikes. Furthermore, ensuring reliable energy consumption management for the UAV in the field is essential. Although a fully autonomous tributary mapping system has yet to be realized, we anticipate that the proposed system can be improved via ongoing field tests.

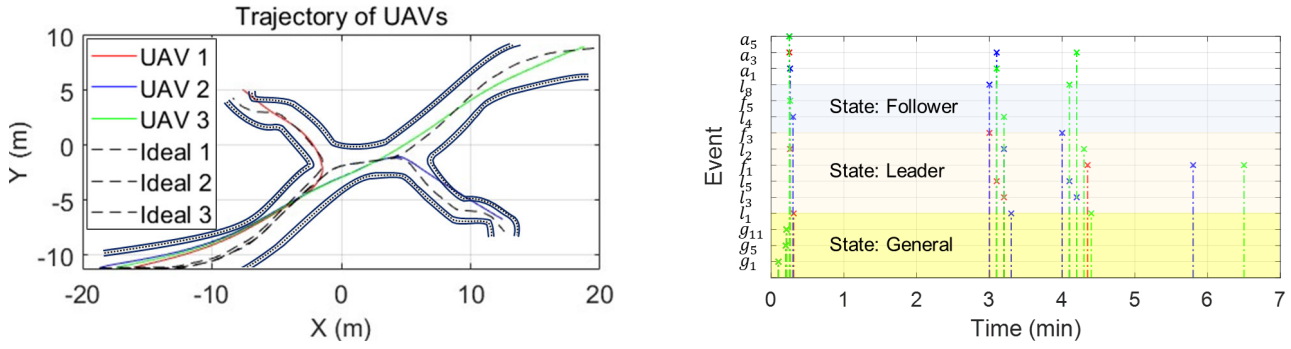
LiDAR sensors are susceptible to noise from environmental conditions such as rain, fog, and dust, which can introduce random errors in distance measurements. These disturbances can lead to point cloud anomalies, affecting the accuracy and fidelity of the generated maps. Potential errors or uncertainties in LiDAR data caused by sensor noise, reflections, or occlusions can be mitigated via preprocessing in the SLAM stage, using various techniques, such as statistical outlier removal and sensor fusion (e.g., with IMU) to ensure high-quality inputs for mapping and planning. For path planning, RANSAC enhances robustness by filtering out outliers and isolating valid inliers, even in challenging conditions, such as partial occlusions or highly reflective surfaces. If combined with an initial noise-filtering step before applying RANSAC, it can be expected to minimize the influence of LiDAR data uncertainties, ensuring accurate and reliable path generation in unstructured environments.

Furthermore, the proposed hybrid system-based tributary mapping approach is scalable depending on application requirements. It can be extended beyond homogeneous robots to include heterogeneous robots, making it suitable for applications in various domains beyond water management. For example, a representative method in water management involves sampling: UAVs conduct remote sensing from the air, whereas UGVs and unmanned surface vehicles collect more detailed local information. Moreover, this method can be adapted to agricultural tasks, where UAVs gather data



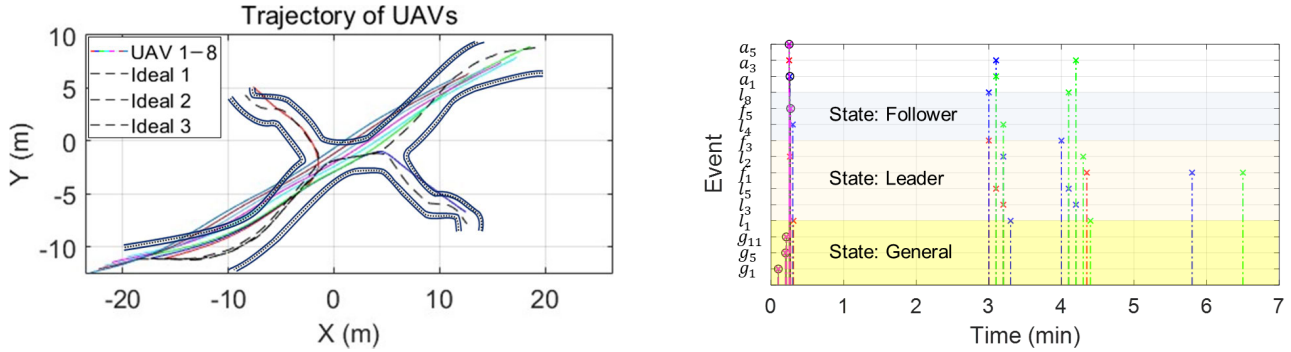
(a) (b)

FIGURE 12. Experimental results for a single UAV: (a) trajectory and (b) event and transitions.



(a) (b)

FIGURE 13. Experimental results for three UAVs: (a) trajectories and (b) event and transitions.



(a) (b)

FIGURE 14. Experimental results for eight UAVs: (a) trajectories and (b) event and transitions.

for yield prediction models from the air, and UGVs perform localized tasks. These expansions highlight the versatility and potential of the proposed system in diverse applications.

VIII. CONCLUSION

This paper presents a hybrid system employing multiple-UAVs for effective monitoring in nature. The designed tributary mapping framework integrates perception and control components. The perception system employs 3D LiDAR semantic segmentation to recognize surface water, exploiting the water-absorbing properties of LiDAR to define a water point cloud and segment surface water areas via SVD. Paths are created using RANSAC by determining the center points of these segmented areas. Each UAV is modeled with a continuous-time dynamic-based low-level model and a discrete event-time dynamic-based high-level model. We devised the desired plant behavior by addressing the control objectives and developed a corresponding supervisory controller. The effectiveness of the tributary mapping system, including perception accuracy and

path generation, and the hybrid system-based supervisory controller was validated via a physics-based robot simulator. The experimental results demonstrate that the limitations of a single-UAV system can be overcome by expanding to a multiple-UAVs approach, improving robustness across lighting conditions via a LiDAR-based perception system. Compared with single-UAV systems, the proposed system reduces exploration time by nearly 50%, addressing the challenge of limited flight duration due to a restricted UAV battery life. This approach reveals significant potential for effectively managing complex, unstructured natural environments and demonstrates substantial potential for managing complex, unstructured natural environments.

ACKNOWLEDGMENT

(Jaehwi Seol and Jeonghyeon Pak contributed equally to this work.)

REFERENCES

[1] M. Becker, F. Papa, F. Frappart, D. Alsdorf, S. Calmant, J. S. da Silva, C. Prigent, and F. Seyler, "Satellite-based estimates of surface water dynamics in the Congo river basin," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 66, pp. 196–209, Apr. 2018.

[2] K. N. Markert, F. Chishtie, E. R. Anderson, D. Saah, and R. E. Griffin, "On the merging of optical and SAR satellite imagery for surface water mapping applications," *Results Phys.*, vol. 9, pp. 275–277, Jun. 2018.

[3] J. Pak, J. Kim, Y. Park, and H. I. Son, "Field evaluation of path-planning algorithms for autonomous mobile robot in smart farms," *IEEE Access*, vol. 10, pp. 60253–60266, 2022.

[4] J. Seol, C. Ju, and H. I. Son, "Leader–follower control of multi-unmanned aerial vehicle based on supervisory control theory for a broad tributary area mapping scenario," *Proc. Inst. Mech. Eng., I, J. Syst. Control Eng.*, vol. 237, no. 10, pp. 1765–1776, Nov. 2023.

[5] L. Wei, Z. Wang, C. Huang, Y. Zhang, Z. Wang, H. Xia, and L. Cao, "Transparency estimation of narrow rivers by UAV-borne hyperspectral remote sensing imagery," *IEEE Access*, vol. 8, pp. 168137–168153, 2020.

[6] J. Pak and H. I. Son, "Semantic SLAM-based autonomous tributary navigation system using 3D LiDAR point cloud for UAV," in *Proc. 22nd Int. Conf. Control, Autom. Syst. (ICCAS)*, Nov. 2022, pp. 1380–1382.

[7] Y. K. Lopes, S. M. Trenkwalder, A. B. Leal, T. J. Dodd, and R. Groß, "Supervisory control theory applied to swarm robotics," *Swarm Intell.*, vol. 10, no. 1, pp. 65–97, Mar. 2016.

[8] C. Wei, Z. Ji, and B. Cai, "Particle swarm optimization for cooperative multi-robot task allocation: A multi-objective approach," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2530–2537, Apr. 2020.

[9] Z. Zhou, J. Liu, and J. Yu, "A survey of underwater multi-robot systems," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 1, pp. 1–18, Jan. 2022.

[10] J. Hu, H. Niu, J. Carrasco, B. Lennox, and F. Arvin, "Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 14413–14423, Dec. 2020.

[11] M. Saboia, L. Clark, V. Thangavelu, J. A. Edlund, K. Otsu, G. J. Correa, V. S. Varadharajan, A. Santamaria-Navarro, T. Touma, A. Bouman, H. Melikyan, T. Pailevanian, S.-K. Kim, A. Archanian, T. S. Vaquero, G. Beltrame, N. Napp, G. Pessin, and A.-A. Agha-Mohammadi, "ACHORD: Communication-aware multi-robot coordination with intermittent connectivity," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 10184–10191, Oct. 2022.

[12] S. Rathinam, P. Almeida, Z. Kim, S. Jackson, A. Tinka, W. Grossman, and R. Sengupta, "Autonomous searching and tracking of a river using an UAV," in *Proc. Amer. Control Conf.*, Jul. 2007, pp. 359–364.

[13] Y. Jang, C. Oh, Y. Lee, and H. J. Kim, "Multirobot collaborative monocular SLAM utilizing rendezvous," *IEEE Trans. Robot.*, vol. 37, no. 5, pp. 1469–1486, Oct. 2021.

[14] Y. Wang, Q. Chen, Q. Zhu, L. Liu, C. Li, and D. Zheng, "A survey of mobile laser scanning applications and key techniques over urban areas," *Remote Sens.*, vol. 11, no. 13, p. 1540, Jun. 2019.

[15] X. An, C. Wu, Y. Lin, M. Lin, T. Yoshinaga, and Y. Ji, "Multi-robot systems and cooperative object transport: Communications, platforms, and challenges," *IEEE Open J. Comput. Soc.*, vol. 4, pp. 23–36, 2023.

[16] C. Ju and H. I. Son, "Modeling and control of heterogeneous agricultural field robots based on Ramadge–Wonham theory," *IEEE Robot. Autom. Lett.*, vol. 5, no. 1, pp. 48–55, Jan. 2020.

[17] F. Isikdogan, A. Bovik, and P. Passalacqua, "RivaMap: An automated river analysis and mapping engine," *Remote Sens. Environ.*, vol. 202, pp. 88–97, Dec. 2017.

[18] O. Özcan and O. Özcan, "Multi-temporal UAV based repeat monitoring of rivers sensitive to flood," *J. Maps*, vol. 17, no. 3, pp. 163–170, Jun. 2021.

[19] D. G. Leckie, E. Cloney, C. Jay, and D. Paradine, "Automated mapping of stream features with high-resolution multispectral imagery," *Photogrammetric Eng. Remote Sens.*, vol. 71, no. 2, pp. 145–155, Feb. 2005.

[20] X. Luo, X. Tong, and Z. Hu, "An applicable and automatic method for earth surface water mapping based on multispectral images," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 103, Dec. 2021, Art. no. 102472.

[21] J. Yang, A. Dani, S. Chung, and S. Hutchinson, "Vision-based localization and robot-centric mapping in riverine environments," *J. Field Robot.*, vol. 34, no. 3, pp. 429–450, May 2017.

[22] K. Meier, S. Chung, and S. Hutchinson, "River segmentation for autonomous surface vehicle localization and river boundary mapping," *J. Field Robot.*, vol. 38, no. 2, pp. 192–211, Mar. 2021.

[23] M. Wieland, S. Martinis, R. Kiefl, and V. Gstaiger, "Semantic segmentation of water bodies in very high-resolution satellite and aerial images," *Remote Sens. Environ.*, vol. 287, Mar. 2023, Art. no. 113452.

[24] J. D. Paul, W. Buytaert, and N. Sah, "A technical evaluation of LiDAR-based measurement of river water levels," *Water Resour. Res.*, vol. 56, no. 4, pp. 1–9, Apr. 2020.

[25] S. García-López, M. Vélez-Nicolás, P. Zarandona-Palacio, A. C. Curcio, V. Ruiz-Ortiz, and L. Barbero, "UAV-borne LiDAR revolutionizing groundwater level mapping," *Sci. Total Environ.*, vol. 859, Feb. 2023, Art. no. 160272.

[26] Y. Shan, X. Yao, H. Lin, X. Zou, and K. Huang, "LiDAR-based stable navigable region detection for unmanned surface vehicles," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–13, 2021.

[27] P. Yao, Z. Xie, and P. Ren, "Optimal UAV route planning for coverage search of stationary target in river," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 2, pp. 822–829, Mar. 2019.

[28] J. Wu, L. Cheng, S. Chu, and Y. Song, "An autonomous coverage path planning algorithm for maritime search and rescue of persons-in-water based on deep reinforcement learning," *Ocean Eng.*, vol. 291, Jan. 2024, Art. no. 116403.

[29] F. Panetsos, G. C. Karras, K. J. Kyriakopoulos, O. Oikonomides, P. Kolios, D. Eliades, and C. Panayiotou, "A motion control framework for autonomous water sampling and swing-free transportation of a multirotor UAV with a cable-suspended mechanism," *J. Field Robot.*, vol. 40, no. 5, pp. 1209–1230, Aug. 2023.

[30] B. H. Y. Alsalam, K. Morton, D. Campbell, and F. Gonzalez, "Autonomous UAV with vision based on-board decision making for remote sensing and precision agriculture," in *Proc. IEEE Aerosp. Conf.*, Mar. 2017, pp. 1–12.

[31] H. I. Son and S. Lee, "Failure diagnosis and recovery based on design framework," *J. Intell. Manuf.*, vol. 18, no. 2, pp. 249–260, Jul. 2007.

[32] C. Ju and H. I. Son, "Discrete event systems based modeling for agricultural multiple unmanned aerial vehicles: Automata theory approach," in *Proc. 18th Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2018, pp. 258–260.

[33] Y. Liu and G. Nejat, "Multirobot cooperative learning for semiautonomous control in urban search and rescue applications," *J. Field Robot.*, vol. 33, no. 4, pp. 512–536, 2016, doi: 10.1002/rob.21597. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1002/rob.21597

[34] G. H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," in *Handbook for Automatic Computation: Volume II: Linear Algebra*, Berlin, Germany, 1971, pp. 134–151.

[35] H. Wang, C. Wang, C.-L. Chen, and L. Xie, "F-LOAM: Fast LiDAR odometry and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Prague, Czech Republic, 2021, pp. 4390–4396, doi: 10.1109/IROS51168.2021.9636655.

[36] J. Zhang et al., "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot., Sci. Syst.*, Berkeley, CA, USA, vol. 2, no. 9, 2014, pp. 1–9, doi: 10.15607/RSS.2014.X.007.

[37] W. M. Wonham and K. Cai, *Supervisory Control of Discrete-Event Systems*, 2019.

[38] T. A. Henzinger, "The theory of hybrid automata," in *Proc. 11th Annu. IEEE Symp. Log. Comput. Sci.*, Apr. 1996, pp. 278–292.
[39] C. Ju and H. I. Son, "A hybrid systems-based hierarchical control architecture for heterogeneous field robot teams," *IEEE Trans. Cybern.*, vol. 53, no. 3, pp. 1802–1815, Mar. 2023.
[40] N. Sehad, X. Tu, A. Rajasekaran, H. Hellaoui, R. Jäntti, and M. Debbah, "Towards enabling reliable immersive teleoperation through digital twin: A UAV command and control use case," in *Proc. IEEE Global Commun. Conf.*, Dec. 2023, pp. 6420–6425.

JAEHWI SEOL received the B.S. and M.S. degrees from the Department of Rural and Biosystems Engineering, Chonnam National University, South Korea, in 2020 and 2023, respectively, where he is currently pursuing the Ph.D. degree with the Department of Convergence Biosystems Engineering. His research interests include field robotics, supervisory control, and discrete event systems.

JEONGHYEON PAK received the B.S. degree from the Department of Rural and Biosystems Engineering, Chonnam National University, Republic of Korea, in 2022, and the M.S. degree from the Department of Convergence Biosystems Engineering, Chonnam National University, in 2024, where she is currently pursuing the Ph.D. degree. Her research interests include multi-agent systems, field robotics, machine learning, and SLAM.

HYOUNG IL SON (Senior Member, IEEE) received the B.S. and M.S. degrees from the Department of Mechanical Engineering, Pusan National University, Republic of Korea, in 1998 and 2000, respectively, and the Ph.D. degree from the Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea, in 2010. He also had several appointments in both academia and industry as a Senior Researcher at LG Electronics, Pyungtaek, Republic of Korea, from 2003 to 2005, and Samsung Electronics, Cheonan, Republic of Korea, from 2005 to 2009, as a Research Associate at the Institute of Industrial Science, The University of Tokyo, Tokyo, Japan, 2010, and as a Research Scientist at the Max Planck Institute for Biological Cybernetics, Tübingen, Germany, from 2010 to 2012. Before joining Chonnam National University, from 2012 to 2015, he was at the Telerobotics Group, Central Research Institute, Samsung Heavy Industries, Daejeon, Republic of Korea, as a Principal Researcher. In 2015, he joined as a Faculty Member with the Department of Convergence Biosystems Engineering, Chonnam National University, Gwangju, Republic of Korea, where he is currently working as a Professor. He is also working as an Adjunct Professor with the Department of Robotics Engineering, Chonnam National University. His research interests include field robotics, hybrid systems, haptics, teleoperation, and agricultural robotics.