

## Improving coverage method of autonomous drones for environmental monitoring

Ömür YILDIRIM<sup>1,2</sup> , Revna ACAR VURAL<sup>1,\*</sup> , Klaus DIEPOLD<sup>2</sup> 

<sup>1</sup>Department of Electronics and Communication Engineering, Faculty of Electrical and Electronics Engineering, Yıldız Technical University, İstanbul, Turkey

<sup>2</sup>Department of Electrical and Computer Engineering, Technical University of Munich, Munich, Germany

Received: 23.12.2019

Accepted/Published Online: 10.08.2020

Final Version: 30.11.2020

**Abstract:** With the rapid developments of unmanned aerial vehicles (UAVs), usage of UAVs is increasing to bring autonomy for complicated processes such as environmental monitoring. Because of the complexity of the problem, environmental monitoring tasks are highly demanding in terms of time and resources. To reduce expensive costs of operations, improvements on autonomous observation capabilities has a key role. In this work, we offer coverage improvements for our autonomous environmental monitoring system. We compared different path planning approaches to find out the optimum path planning solution. Simulation results showed that required task execution time and required resources are decreased by usage of improved decomposition of the coverage field.

**Key words:** Environmental monitoring, UAV, target detection, complete coverage, path planning

### 1. Introduction

Nowadays, unmanned aerial vehicles (UAV) are increasingly being preferred for applications such as path planning, search and rescue operations and environmental monitoring for wildlife and precision agriculture. Introducing autonomy to these vehicles provides abilities such as instantaneous decision making, perceiving the environment and reacting to events without human invention. Moreover, autonomous vehicles are also capable of analyzing the collected information, communicating with each other or a ground terminal and decision making using algorithms, sensors and actuators [1–4].

Robust environmental monitoring is a demanding task that can require long periods of observation, periodical or continuous tracking several data collected from a number of sensors and access to protected, private or uncivilized areas with various sizes. In order to overcome the challenges of this task, aerial imaging techniques through the use of unmanned autonomous aerial vehicles, systems and platforms (UAVs, UASs, UAPs) are employed [5–8].

In the literature, promising tracking and observation performances of autonomous environmental monitoring approaches are reported. One of them utilized sensor networks where each sensor collects different type of measurements and exchanged them in order to track the environment and estimate the location of the event [9, 10]. The usage of reference locations in this approach leads to inhibitive deployment and time-consuming standardization. Another approach investigates aerial vehicles equipped with various sensor that can move towards the prespecified area. Robotic systems are more suitable for localization because of their dynamic and

\*Correspondence: Correspondence: racar@yildiz.edu.tr

flexible aspects [7, 11, 12]. Yet, in wild or urban environments performing consistent operations with robotic systems may be challenging due to many crucial issues such as planning, navigation and endurance. Therefore, autonomous and robust monitoring systems including artificial intelligence, obstacle detection and avoidance technology, control and communications, image processing and battery capacity measurement need to be developed [11–15]. In urban environment, maintaining persistent operation of robotic systems is troublesome even with low level of human intervention [16] where stable broadband radio link cannot be guaranteed in those environments. Limited availability of computing resources and low weight sensor usage in harsh environments for mobile systems introduces some challenges for autonomous environment monitoring. Recently, similar researches have been released for the execution of various monitoring missions. In [17, 18], researchers have concentrated to high altitude imagery with autonomous UAVs to map environments using thermal imaging and multi spectral imaging, respectively. Since, big sized data is collected and needed to be processed, huge computational power is required which is not available on-board. A possible solution is to transmit the collected data to a base station and process it after monitoring is completed. However network security should be provided during transmission and time delay should be minimized. Moreover, the ability to plan collision-free paths in complex urban environments is another task of UAV autonomy. In [3], 3D path planning of UAVs using adaptive discrete mesh is proposed and a brief bibliographic review focused on 3D trajectory planning is presented. However, planning a three-dimensional (3D) path can be impractical for some applications where the task is to sweep the surface. Yet, in 3D planning, task is usually travelling. In [19], a static 2D approach based on vertical cell decomposition is presented for path planning and identifying ground objects. This method explores and decomposes the 2D environment by constructing a finite data structure that completely encodes a solution for a given path including obstacles, such as buildings, resulting in an efficient and simple 2D path detection. Recently, in [20] a wind prediction method is used with a boustrophedon coverage path. Researchers tried to minimize the flight times by using a certain sweep angle that can benefit from the wind. In [21], a method that uses boustrophedon coverage with heuristic algorithm to find an optimal path for a vehicle with Dubins motion constraints is suggested. Researchers stated that experiments showed a promising improvement to find optimal path in less time. In [22], an improved boustrophedon coverage planning method is offered for low-altitude environment coverage in known environments with UAVs. Suggested method optimizes boustrophedon coverage by using several sweep approaches and combining them according to obstacles and required transition movements between cells.

This study is initiated with the development of a modular framework for autonomous drones in environmental monitoring. Within this framework, modules are developed asynchronous. A static path planning leads to robust navigation and environmental monitoring. The aim of this platform is to track the targets in ground (in our scenario, targets are the left-over bottles in university campus ground) and to ensure a safe flight. We presented preliminary results about vertical cell decomposition based a decision process for an unmanned aerial vehicle and discussed about the requirements that autonomy weights on the system [23]. One of the main differences of our work and previous studies is the processing aerial view of small-sized individual targets rather a group of targets. In order to perform this mission low altitude flight simulations with target and obstacle oriented maneuvers are performed and this enables target specific data gathering. Secondly, we process real-time data with on-board sensors and processors. This brings a reaction capability for complete autonomy. In this study, we offer an improved cell decomposition algorithm to increase complete coverage efficiency of our autonomous environmental monitoring system. Simulations were performed to test performances of two decomposition methods (boustrophedon decomposition and trapezoidal decomposition) for both the simple ob-

stacle map and Technische Universität München (TUM) campus map. Improvements on autonomous inspection capability of the system are validated and verified by simulation results.

Following introduction, environmental monitoring framework is presented in detail in Section 2. Layers of the proposed framework include coverage, investigation and safety modules. Experimental setup and results are provided in Sections 3 and 4, respectively. Finally, concluding remarks are discussed and possible improvements are suggested as future work.

## 2. Environmental monitoring framework

In this work, we tested our proposed autonomous monitoring system for two different cell decomposition algorithms in a scenario where the task is identifying left-over beer bottles inside a university campus. To create a general structure which can be adaptable for different environmental monitoring tasks, we divided our system into modules, each of which is in charge of certain tasks and contains a specific algorithm. In this work, we will focus on the performance of decision-making module of our system with comparison of two different cell decomposition algorithms.

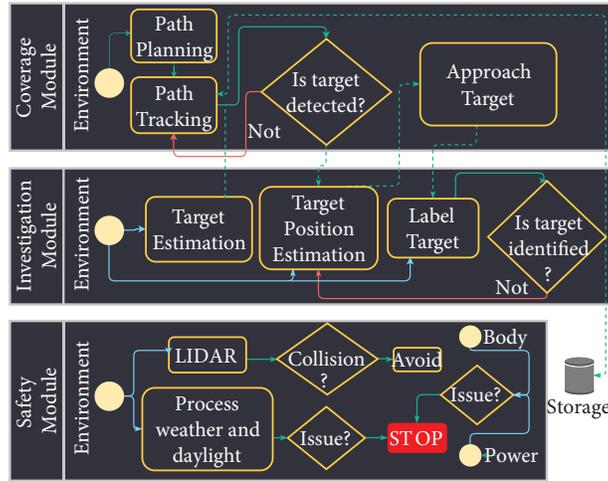
### 2.1. Autonomous system for environmental monitoring

An environmental monitoring system can be separated to two parts to identify tasks of mission clearly. In the first part, the monitoring system examines the environment and creates a coverage part and orchestrates the mission. In the second part, the system tracks the field for a potential target and tries to identify the target. Hence, for our environmental monitoring mission where the system needs to track several bottles with processing visual data, we designed our system with two connected layers, cognition and vision. The system contains a camera which feeds the vision layer. The vision layer process the raw data and estimates a possible target or identifies a detected target. The vision layer outputs the target estimation and identification to feed the cognition layer. Subsequently, the cognition layer process the output of the vision layer with other sensor data to take real time decisions and actions. The cognition layer controls movements of the UAV to achieve the complete coverage of field by considering target estimations of vision layer and events in the environment such as collision.

Environmental monitoring missions has a sectional structure in terms of the definition of subtasks. Monitoring systems are required to track and observe the field. To use the resources efficiently, systems constructs a path which ensures complete coverage. This task is often called path planning and path tracking. Further, systems are required to identify monitored targets in respect to definition of targets and environment. This task is often called observation. For the autonomy of environmental monitoring, systems required to ensure the safety of itself and the environment to ensure the continuity of mission. Safety tasks comprise resources controls to potential harm precaution. Hence, in regards to defined structure of environmental monitoring missions, proposed autonomous system is structured to fulfill environmental monitoring missions in three decision states; path tracking, target acquisition and emergency. Cognition layer is divided into three modules to conduct each decision state; coverage module, investigation module and safety module. While each module has its own action and decision logic, each module intercommunicates with other modules for their future actions-decisions as in Figure 1.

### 2.2. Complete coverage path planning

In order to cover the field complete, the vehicle is required to have a path which ensures the sweeping of the empty space of the field. A complete coverage planning algorithm constructs the complete coverage path. Path



**Figure 1.** Framework of the proposed system.

planning algorithms can be characterized by the prior knowledge that they require. As in [19, 24, 25], algorithms require prior knowledge of the accessible space of the field to calculate the path. Due to static path calculation, the environment should not have any moving obstacles so that free space of the field is not change over time. As in [26, 27], algorithms use a dynamic approach to resolve an unknown environment in real-time. With real-time approaches, it is not absolute to measure optimality of produced path(s). Yet, avoiding overlook and overlapping coverage is as important as for static approaches.

Moreover, path planning algorithms can be separated by the number of dimensions of vehicle movement. Many autonomous applications have a special type of path planning in a two-dimensional (2D) environment for tasks like sweet pepper harvesting [28], mobile cleaning [29], underwater exploration [30], lawn mowing [31]. Three-dimensional (3D) path planning is not necessary for such specific tasks since the main goal of tasks is complete coverage of a 2D field. However, 3D planning is often used for tasks which requires travelling [32].

In this work, we implement a static 2D approach for path planning. As in [33–35], we use an algorithm based on vertical cell decomposition. Further, we implemented and compared the performance of two different cell calculation approaches, boustrophedon decomposition and trapezoidal decomposition.

Algorithm 1 searches for vertical lines where obstacles has a corner and detects the cells by separating empty areas between vertical lines and constructs a graph throughout the cells and creates a path based on the graph.

System tracks the total coverage as displayed in Figures 2a and 2b. It uses the target detection area  $A_{detection}$  which is a complete circle around the UAV as in equation (1):

$$A_{detection} = \pi \tan^2(\alpha) r_{detection}^2 \varepsilon_{altitude} \tag{1}$$

where  $r_{detection}$  is target detection range,  $\alpha$  is the angle of view,  $\varepsilon_{altitude}$  is error rate of altitude sensor. After, UAV calculates complete coverage  $C_{coverage}$  as in equation (2):

$$C_{coverage} = \pi \tan^2(\alpha) r_{detection}^2 \varepsilon_{altitude} + \Delta t V \varepsilon_{acclrm} \tag{2}$$

where  $\Delta t$  is travel time,  $V$  is the velocity of vehicle and  $\varepsilon_{acclrm}$  is error rate of the accelerometer. Yet,

movement during the inspection is not included complete coverage calculation. Because UAV can return to a position that is already covered.

---

**Algorithm 1** Extended vertical cellular decomposition
 

---

**Input:** vertices  $V$

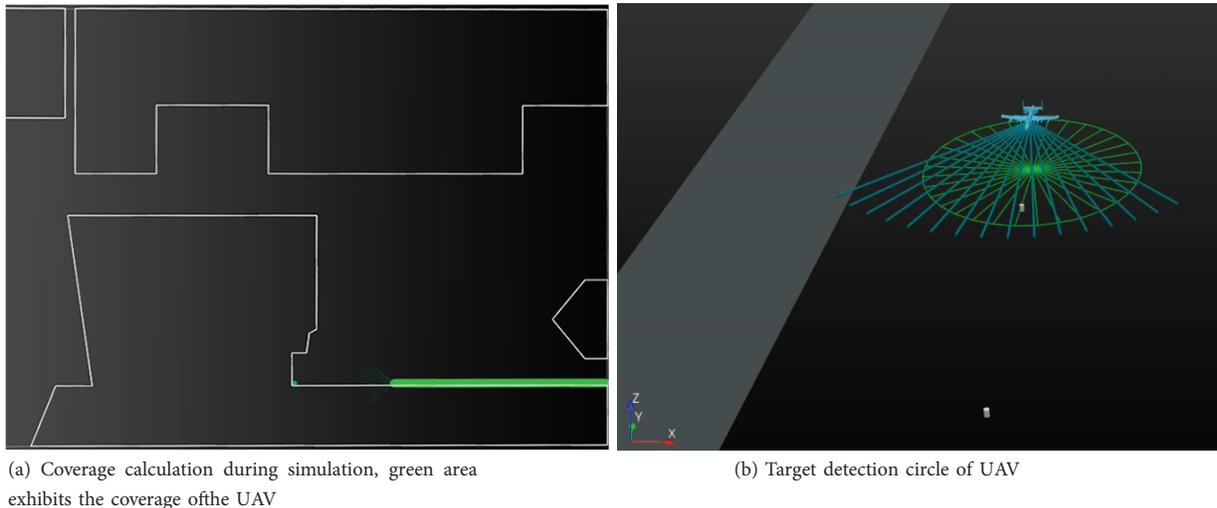
**Output:** cells  $C$ , graph  $G$ , decomposition of cells  $DC$

```

 $V_{lines} \leftarrow calculate\_vertical\_lines(V)$ 
 $L_{adaptive} \leftarrow []$ 
for  $line$  in  $V_{lines}$  do
   $L_{adaptive} \leftarrow find\_adaptive\_lines(line, V)$ 
end for
 $V_{lines} \leftarrow merge\_lines(V_{lines}, L_{adaptive})$ 
for  $line$  in  $V_{lines}$  do
   $C \leftarrow find\_cell(line, V)$ 
end for
 $G \leftarrow find\_graph(C)$ 
 $DC \leftarrow decompose(C)$ 

```

---



**Figure 2.** Captures of simulation.

**Trapezoidal decomposition:** Trapezoidal decomposition is a form of exact cellular decomposition, which is the aggregation of not intersecting adjacent zones forming the field. Each zone is termed a cell and the aggregation of cells fills the empty space of the field. Coverage of each cell can be achieved simply by back-and-forth motions. Trapezoidal cells are convex empty spaces between two successive polygon vertices. Successive vertices are generally closest neighbour vertices and an imaginary line from one vertex to another should have no intersection with given obstacles of the map. As in Figure 3a where the blue part is the obstacle, vertex A and vertex B are neighbours. An imaginary line between vertices A and B has no intersection with the obstacle. Thus, the empty space between vertex A and B is the cell 1. But vertex A and vertex C are not closest neighbours. Further, an imaginary line from vertex A to vertex C has an intersection with the obstacle.

**Boustrophedon decomposition:** Boustrophedon decomposition optimizes coverage path in order to reduce lengthwise transition movements between cells. Thus, boustrophedon decomposition reduces total cell count which increases simplicity and efficiency of cell transitions. Cells are covered with simple back-and-forth

motions as trapezoidal decomposition. Boustrophedon cells [35] merges trapezoidal cells and creates non-concex cells. In both decomposition methods, cells are created between two vertices. The difference between these two approaches is with the middle events [35]. The trapezoidal decomposition closes or opens a cell when a middle event occurs. But, the boustrophedon decomposition simply updates the current cell with new border points during the middle events.

As in Figure 3a where the blue part is the obstacle, cell 1 is defined between vertices A and B and cell 2 is defined between vertices B and C by the trapezoidal decomposition algorithm. But for the boustrophedon decomposition intersection point for vertex B is a middle event and causes the extension of the cell 1 until the vertex C. Simply, this method merges trapezoidal cells 1 and 3 and produces cell 1 as in Figure 3b.

Difference of the boustrophedon decomposition and trapezoidal decomposition can be seen better with Figure 4. For TUM map, the trapezoidal decomposition resolves map to fourteen cells where the boustrophedon decomposition resolves nine cells.

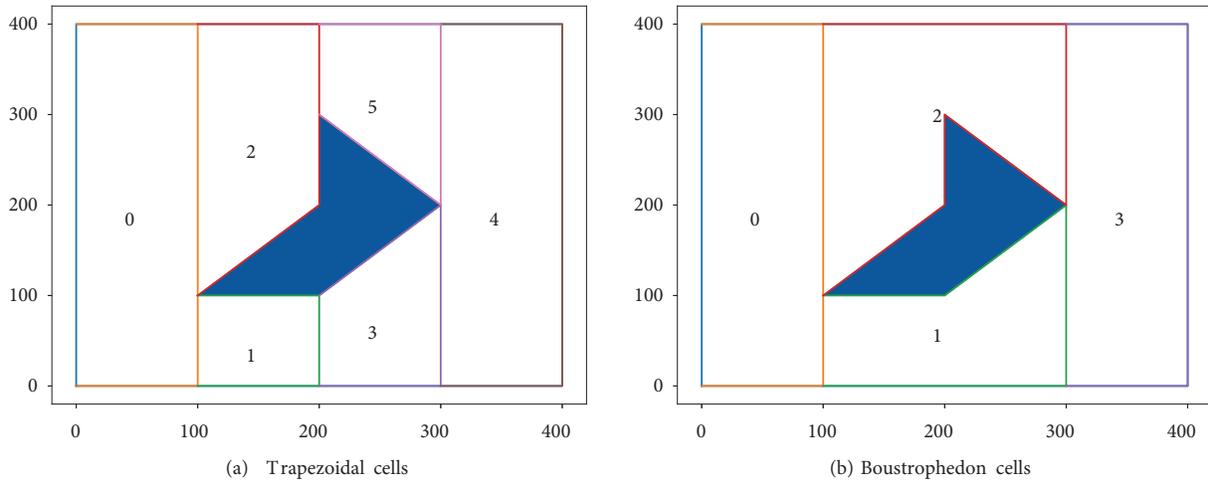


Figure 3. Cell decompositions of sample map.

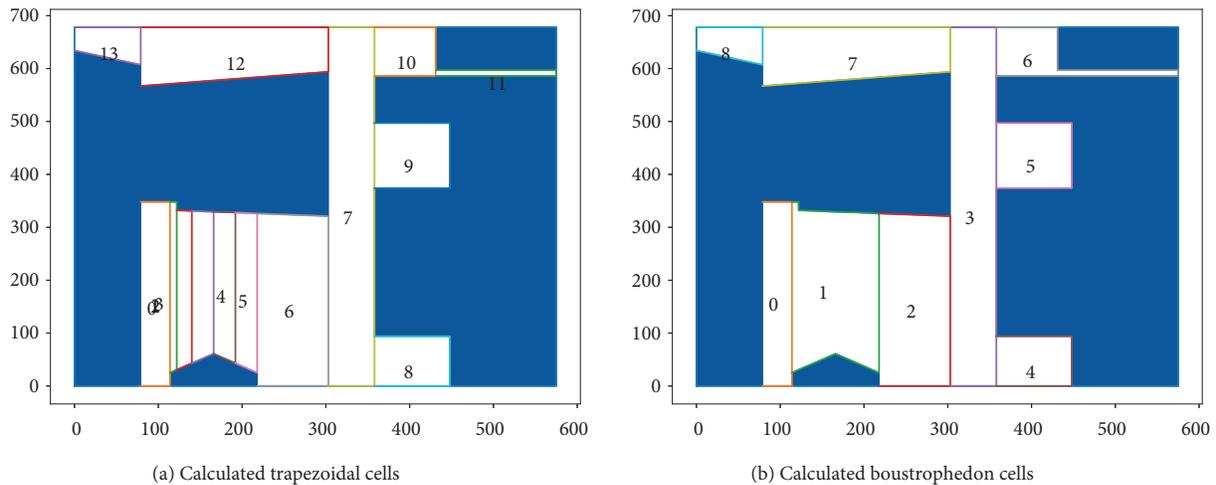


Figure 4. Cell decompositions of TUM map.

### 2.3. Autonomous inspection

An inspection task requires high-level observation skills and a robust decision making mechanism. To make the observation, systems can depend on human labor and/or machines. Yet, due to complexity and repetitiveness, execution of inspection tasks demands intensive labor and high concentration. One of the reasons for complexity of inspection systems comes from the fact that processing periodic or continuous data which needed to monitor the status of environment/targets. Moreover, the field that needs to be monitored can be in a position which is difficult to attain even with a vehicle. In respect to the dependencies and required skill set of inspection tasks, autonomous executions are advantageous compared to manual executions. In [36], autonomous aerial robots were used for transportation and deployment for search and rescue missions. UAVs contained different sensors to detect the targets and move cooperatively to tasks such as deployment of small objects. In another study [37], an autonomous monitoring application was presented to identify emergencies. They proposed a technique which combines and processes visual and infrared cameras for segmentation of fires.

In general, the requirement for the autonomous inspection task is collecting rich and reliable information of targets. Based to the target and monitoring environment, optimum circumstances for data quality differs, for example, while identifying bark beetle damage on trees [7], vehicle required high altitude flight and for chemical source localization in a small controlled aquatic environment [38], vehicle required a direct approach to the source. For our monitoring task where the system needs to identify beer bottles, we required to collect information from visual data. To gather better information, system needed to have optimum conditions for the best image quality. Thus, we implemented a policy-based approach to satisfy image quality needs with a supplying optimum perspective of the bottle as in Figure 2b. System decides for a certain movement policy after processing target position and image processing output. As explained in Section 3, the UAV assures the optimum distance between the detected bottle and itself. If observation from optimum distance does not satisfy the required image quality, UAV decides for a policy where it makes a circular movement around the target.

### 2.4. Emergencies and safety

Governments and institutions determine many flight regulations and safety standards for autonomous vehicles to ensure the safety by avoiding/reducing potential damage to UAVs and environment. Moreover, safety has a key role for systems with expensive components to ensure the continuous progress of mission. In our approach, we used a set of rules to determine and identify potential harms. Each emergency detection is engaged to a sensor measurement and a rule which sets a certain threshold. In general, our rules are fusion of the collection of previous experiences, regulations of autonomous flights and health rate of mechanical parts of the UAV.

Our system tracks and detects collision, power need to finish task, damage on mechanics (e.g., stopped engine), loss of localization, optimal daylight, adverse weather conditions. To detect emergencies, we use several data sources; Laser Imaging, Detection And Ranging (LIDAR) to calculate possible collision, forecast tracking to detect potential instabilities for flight such as instant change on wind, a photo-diode to track the daylight. Further, if we detect a violation of one of our safety merits, our system determines on a policy that ensures the safety of system. If the detected emergency violation is a collision, UAV avoids the collision by changing its route. In case of more fragile emergencies where system can not resolve without human intervention, it breaks the execution of tasks and tries to move a safe and reachable position (e.g., safety landing or returning to the base).

### 3. Experimental setup

#### 3.1. Implementation

To run the simulation, we determined certain target features. For the target features, we selected certain parameters for target detection. These parameters are used for cell decomposition algorithms and target inspection. In this section, we present details of target features and algorithms.

##### 3.1.1. Target details

The mission of simulation is to collect locations and features of beer bottles inside certain area. Beer bottles differ on size but roughly they have 0.25 m height, 0.07 m diameter.

##### 3.1.2. Target detection details

During coverage of the field, UAV is cruising in certain altitude and it captures the field from aerial perspective. In respect to defined target size, optimum cruise altitude of the UAV has set to 2.5 m. One a target detected, UAV gets closer to target to be able to identify bottle type and brand. This optimum identification distance has set 0.75 m.

##### 3.1.3. Path planning details

To determine the path, system has used two different algorithms based on vertical cell decomposition principal, trapezoidal decomposition and boustrophedon decomposition. Outcome if these two algorithms constructed cells  $C$  converted to road-maps  $DC$ . We have used back and forth motion lines to extract road-maps. Distance between motion lines is set to optimum cruise altitude which is also the optimum detection range of targets during complete coverage.

We extended vertical cell decomposition algorithms, to solve the position issues of obstacles. We improved the algorithm with the adaptive line concept to divide spaces with special obstacle types.

##### 3.1.4. Target identification details

When the system detects a target during complete coverage of the field, UAV tries to make certain movements to increase the accuracy of identification. For successful identification, UAV approaches through the target until it's as close as the optimum identification range. At the optimum identification range, if the target identification accuracy is under the threshold, UAV keeps the optimum distance and does a circular movement around the target.

In simulation, the target detection precision is calculated  $P_{target}$  with Gaussian distribution as in equation (3);

$$P(x) = \frac{100}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} \quad (3)$$

where  $\mu$  is 0.75 m,  $x$  is the parametric distance between UAV and target,  $\sigma$  is 1 m. Further, we applied a random function using python's random uniform function which returns a value between 0 and 100. This random number is compared with complementary of target detection precision to 100.

Algorithm 2 describes the method, which estimates the possibility of target detection and determines target detection by using a threshold. Calculation of target detection precision is based on the Gaussian probability of target distance. Complementary of detection precision is being compared with a randomly

generated number which is between 0 and 100. If the random number is bigger than complementary of detection precision then the target is marked as detected.

---

**Algorithm 2** Determination of target detection

---

**Input:** target precision  $P_{target}$

**Output:** detection  $D$

$possibility \leftarrow random.uniform(0, 100)$

$comp_P \leftarrow 100 - P_{target}$

**if**  $possibility > comp_P$  **then**

$D \leftarrow True$

**else**

$D \leftarrow False$

**end if**

---

Simulations were performed for the simple obstacle map and TUM map to measure the effectiveness of the purposed system in the environments with different complexity. Further, to compare performance of trapezoidal decomposition and boustrophedon decomposition algorithms, two simulations were performed for each map. Tables 1 and 2 show the results for trapezoidal and boustrophedon decomposition algorithm on the simple obstacle map, respectively while Tables 3 and 4 show the results for trapezoidal and boustrophedon decomposition algorithm on the TUM map, respectively. For all the tables, travel distance and mission time are provided by means of meter and minute, respectively.

**Table 1.** Results of the simple obstacle map simulation for trapezoidal decomposition.

Run	Detected targets	Labeled targets	Location precision	Travel distance	Mission time
1	8	8	92.4%	1808	7.1
2	8	8	93.4%	1821	7.37
3	9	7	92.6%	1903	7.9
4	6	5	93.8%	1783	7.11
5	9	9	92.3%	1813	7.27
6	7	7	92.9%	1788	6.99
7	9	8	93.7%	1823	7.35
8	8	8	93.4%	1853	7.88
9	10	10	92.7%	1855	7.18
10	10	10	92.4%	1863	7.31

For each simulation, random placement of 10 bottles which stand vertically placed was done. Further, a quad-rotor rotary wing UAV with a 4500 milliampere hour battery and a LIDAR sensor with 10 m sensitivity was used as the task executor vehicle. UAV had limit speed as 10 m/s and constant acceleration as 5 m/s<sup>2</sup>.

In the simple obstacle map as in Figure 5a, a square shaped obstacle was placed to the center of field. The TUM campus has 4 buildings and they represented with obstacles as in Figure 5b.

#### 4. Experimental results

The system was tested for two different maps. For simulation of both maps, ten bottles were placed to the environment. In simulations, following data were recorded to measure monitoring capability of proposed system;

**Table 2.** Results of the simple obstacle map simulation for boustrophedon decomposition.

Run	Detected targets	Labeled targets	Location precision	Travel distance	Mission time
1	9	9	92.9%	1849	7.20
2	9	9	93.6%	1822	7.31
3	9	8	92.4%	1887	7.4
4	7	7	93.5%	1771	7.21
5	10	10	92.7%	1863	7.32
6	8	8	92.4%	1794	7.03
7	8	8	93.1%	1812	7.16
8	10	10	92.9%	1893	7.82
9	9	9	93.9%	1837	7.61
10	6	6	93.8%	1749	6.91

**Table 3.** Results of TUM campus map simulation for trapezoidal decomposition.

Run	Detected targets	Labeled targets	Location precision	Travel distance	Mission time
1	10	9	93.5%	36608	82.79
2	8	8	92.4%	36253	78.84
3	10	9	92.7%	36547	80.33
4	8	7	92.5%	36276	80.82
5	10	10	94.0%	36353	79.67
6	10	10	93.5%	36304	82.76
7	9	9	93.7%	36126	79.61
8	10	10	93.8%	36226	79.71
9	10	10	92.7%	36280	78.78
10	9	8	93.7%	36380	80.20
11	10	10	92.7%	36289	78.75
12	9	9	92.1%	35932	79.29
13	8	8	92.8%	35697	78.89
14	9	9	93.5%	36493	79.69
15	9	7	93.9%	36235	80.27

detected target count, labeled target count, location precision, total travel distance, total mission execution time and total interruption due to emergencies.

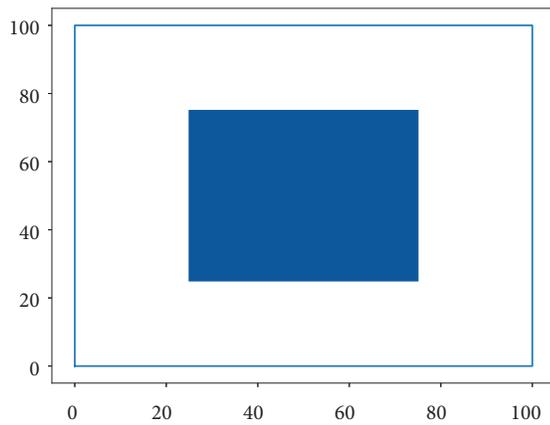
#### 4.1. The simulation of the simple map

A simple map as in Figure 5a was used as the first environment for simulations. The map consists one square shaped obstacle. The map has 100 m width to 100 m length. The obstacle in the middle of the field is 50 m wide and 50 m long. Total coverage space is 7500 m<sup>2</sup>. We randomly placed 10 bottles to the empty spaces around the obstacle.

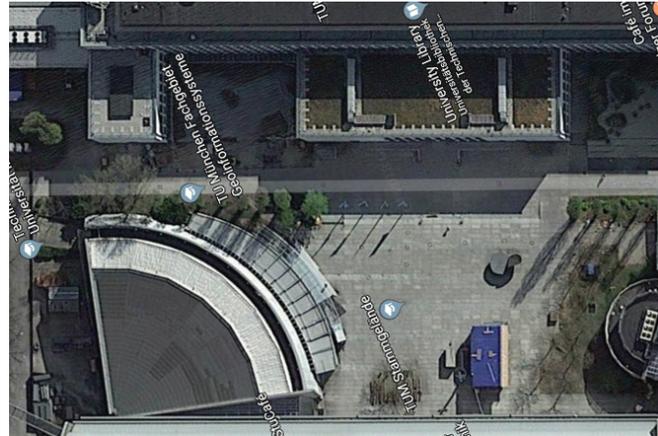
The approximated minimum travel distance for complete coverage of the simple map was calculated as 1500 m from Figure 2. The cruise speed of the drone is 10 m/s. To travel 1500 m, the drone needs to fly at least 150 s (2.5 min).

**Table 4.** Results of TUM campus map simulation for boustrophedon decomposition.

Simulation run	Detected targets	Labeled targets	Location precision	Travel distance	Mission time
1	10	10	92.8%	35974	80.03
2	10	9	94.0%	35571	82.58
3	9	9	93.3%	36128	79.36
4	7	7	92.5%	35910	78.58
5	9	8	92.3%	35483	79.46
6	9	9	92.6%	36081	79.51
7	10	10	93.0%	35896	79.81
8	8	8	92.4%	36143	79.73
9	9	9	93.8%	35724	79.00
10	6	6	92.5%	35502	78.67
11	9	9	93.9%	35859	80.08
12	10	8	94.0%	36150	80.61
13	9	9	93.6%	36127	79.14
14	6	5	92.2%	35743	78.66
15	9	7	93.3%	36235	80.27



(a) The simple obstacle map



(b) TUM campus from aerial view

**Figure 5.** Simulation maps.

Simulations of the simple map were performed for two decomposition methods the boustrophedon decomposition and trapezoidal decomposition as described in Section 2.2. For the simple map, the boustrophedon decomposition method calculated same cells as trapezoidal decomposition. Hence, they should produce similar results. For each method, ten simulations were carried out.

Results of the simple obstacle map simulation for trapezoidal decomposition method were gathered in Table 1. Following averages derived from ten simulation executions: 84% target detection rate, 80% target labelling rate, 92.96% location precision, 1831 total travel distance and 7.34 min of mission execution time. These findings suggest that the drone travelled 331 m more than minimum travel distance for complete coverage 1500 m. Thus, the drone needs 22% more power for travelling than expected. The drone spent 4.86 more minutes

for entire mission execution where minimum time was 2.5 min. Hence, the drone needs 193% more time than expected. Through the runs, no interruption is observed.

Results of the simple obstacle map simulation for boustrophedon decomposition method were gathered in Table 2. Following averages derived from ten simulation executions: 85% target detection rate, 84% target labelling rate, 93.12% location precision, 1827.7 m total travel distance and 7.29 min of mission execution time. These findings suggest that the drone travelled 327.7 m more than minimum travel distance for complete coverage 1500 m. Thus, the drone needs 21.8% more power for travelling than expected. The drone spent 4.79 min more for entire mission execution where minimum time was 2.5 min. Hence, the drone needs 191% more time than expected. Through the runs, no interruption is observed.

Results of both methods do not suggest conspicuous differences. Hence, results prove that usage of boustrophedon decomposition method for simple maps is not efficient, since it demands more computational power than trapezoidal decomposition. Further, it is observed that the suggested system is highly successful for identification and labelling of targets and their positions.

#### 4.2. The simulation of the TUM map

TUM's main campus as in Figure 5b were used as the second environment for simulations. The TUM campus map has 575 m width and 678 m length. The map contains four obstacles which represent buildings of TUM campus. The first obstacle is on the west part of the campus, and it has 104 995 m<sup>2</sup> surface. The second obstacle is on the south part of the campus, and it has 3536 m<sup>2</sup> surface. The third obstacle is on the east part of the campus, and it has 107 632 m<sup>2</sup> surface. The fourth obstacle is on the north part of campus, and it has 11 664 m<sup>2</sup> surface. Total surface of obstacles covers 227 827 m<sup>2</sup> space of the map. Thus, available free space of the TUM map is 162 023 m<sup>2</sup>. Ten beer bottles were randomly placed to free space of the TUM campus.

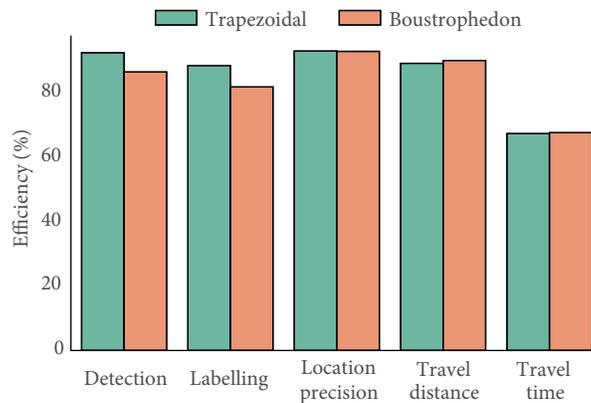
The approximated minimum travel distance for complete coverage of the TUM campus map was calculated as 32 404.6 m from equation 2. The cruise speed of the drone is 10 m/s. To travel 32 404.6 m, the drone needs to fly at least 3241 s (54.01 min).

Results of the TUM campus map simulation for trapezoidal decomposition method were gathered in Table 3. Following averages derived from fifteen simulation executions: 92.66% target detection rate, 88.66% target labelling rate, 93.17% location precision rate, 36 266.6 m total travel distance, 80.02 min of mission execution time. These findings suggest that the drone travelled 3862 m more than minimum travel distance for complete coverage 32 404.6 m. Thus, the drone needs 11.9% more power for travelling than expected. The drone spent 26.01 more minutes for entire mission execution where minimum time was 54.01 min. Hence, the drone needs 48.15% more time than expected. Further, the drone faced three emergency interruptions due to battery outage in each simulation run. For each emergency interruption, the drone returned to base station and recharged its battery to continue to the mission. The complete recharge of the battery takes approximately 4 h. Hence, three complete recharge of the battery caused 12 h delay for completion of the mission.

Results of the TUM campus map simulation for boustrophedon decomposition method were gathered in Table 4. Following averages derived from fifteen simulation executions: 86.66% target detection rate, 82% target labelling rate, 93.08% location precision rate, 35901.7 m total travel distance, 79.69 min of mission execution time. These findings suggest that the drone travelled 3497.1 m more than the minimum travel distance for complete coverage 32404.6 m. Thus, the drone needs 10.79% more power for travelling than expected. The drone spent 25.68 more minutes for entire mission execution where minimum time was 54.01 min. Hence, the drone needs 47.54% more time than expected. Further, the drone faced three emergency interruptions due to

battery outage in each simulation run. As explained before, three complete recharge of the battery caused 12 h delay for completion of mission.

Simulation results in TUM campus map shows that boustrophedon decomposition reduces the travel distance and travel time compared to trapezoidal decomposition as in Figure 6. The main reason for increased travel efficiency is reduced transition movements due to reduced cell count in boustrophedon decomposition. Further, target detection and target labelling percentage is lower in boustrophedon decomposition compared to trapezoidal decomposition. A reduce in target detection and labelling performance is expected due to fact that total investigation time is reducing with boustrophedon decomposition. However, loss in target detection and labelling performance is greater compared to gain in travel time. Randomness of bottle placements can cause such a fact if bottles are placed in cell transition paths. Yet, we expect performance gain in travel time should converge with performance loss in target detection and labelling.



**Figure 6.** Efficiency comparison of two decomposition methods in TUM map.

## 5. Conclusion

In this paper, two cell decomposition approaches were implemented to a modular software framework of an UAV for autonomous environmental monitoring missions. The performance of the system is measured in different environments with simulations for each decomposition approaches. Simulations were performed to test performance of two decomposition methods (boustrophedon decomposition and trapezoidal decomposition) for two environments (simple obstacle map and TUM campus map).

Results of the simple obstacle map simulations provided that the system collected truthful information for 8.2 bottles of every 10 bottles. Further, results provided that system travelled 21.9% more compared to the minimum travel distance for complete coverage of the simple map.

Results of the TUM campus map simulations provided that the system collected truthful information for 8.5 bottles of every 10 bottles. Further, results provided that system travelled 11.3% more compared to the minimum travel distance for complete coverage of the simple map.

Simulations results of both environments proved robust monitoring capability of the proposed system. However, the proposed system performed the simple obstacle map simulation with 78.1% efficiency and the TUM campus map simulation with 88.7% efficiency. Main reason behind efficiency loss is the investigation of targets. In each simulation, ten targets were placed to the field. After detection of targets, the drone reduces its speed and investigates the detected target for detailed information which is causing time loss and additional power usage. Another reason was the coverage path. Transition between cells caused unwanted resources usage.

In TUM campus map simulation, due to large size of the field, the drone returned to the base station to recharge itself for three times. In regards to approximate 4 h recharge time of the battery, the completion of mission is delayed for 12 h. These findings show that the proposed system needs a better capacity for flight time to cut interruptions, and multirotor drones for environmental monitoring in large fields are inefficient due to power capacity of multirotor drones.

This study used two offline path planning algorithms to calculate coverage path. Results showed that both approaches calculated paths that are suitable for complete coverage of the field. However, for complicated maps, boustrophedon decomposition method improves the efficiency of the system due to reduced cell number. It reduces the total execution time of the mission.

At present, the implementation of boustrophedon decomposition and trapezoidal decomposition to the suggested autonomous monitoring system is tested and compared with simulations. To reinforce our simulation results, we will focus on the hardware integration of our system. Further, to monitor the environments with dynamic obstacles, a dynamic path planning approach may be used, such as reinforcement learning based dynamic path planning.

## Acknowledgment

The authors would like to thank the ABBY-Net for providing the idea of environmental monitoring project.

## References

- [1] Sebbane YB. Smart Autonomous Aircraft: Flight Control and Planning for UAV. Boca Raton, Florida, USA: CRC Press, 2015.
- [2] Zhang R, Zhang J, Yu H. Review of modeling and control in UAV autonomous maneuvering flight. In: IEEE 2018 International Conference on Mechatronics and Automation; Changchun, China; 2018. pp. 1920-1925.
- [3] Samaniego F, Sanchis J, Garcia-Nieto S, Simarro R. Recursive rewarding modified adaptive cell decomposition (RR-MACD): a dynamic path planning algorithm for UAVs. *Electronics* 2019; 8 (3): 306. doi: 10.3390/electronics8030306
- [4] Liu X, Yang T, Li J. Real-time ground vehicle detection in aerial infrared imagery based on convolutional neural network. *Electronics* 2018; 7 (6): 78. doi: 10.3390/electronics7060078
- [5] Bayat B, Crasta N, Crespi A, Pascoal AM, Ijspeert A. Environmental monitoring using autonomous vehicles: a survey of recent searching techniques. *Current Opinion in Biotechnology* 2017; 45: 76-84. doi: 10.1016/j.copbio.2017.01.009
- [6] Tripolitsiotis A, Prokas N, Kyritsis S, Dollas A, Papaefstathiou I et al. Dronesourcing: a modular, expandable multi-sensor UAV platform for combined, real-time environmental monitoring. *International Journal of Remote Sensing* 2017; 38 (3): 2757-2770. doi: 10.1080/01431161.2017.1287975
- [7] Nasi R, Honkavaara E, Lyytikainen-Saarenmaa P, Blomqvist M, Litkey P et al. Using UAV-based photogrammetry and hyperspectral imaging for mapping bark beetle damage at tree-level. *Remote Sensing* 2015; 7 (11): 15467-15493. doi: 10.3390/rs71115467
- [8] Manfreda S, McCabe M, Miller PE, Lucas R, Madrigal VP et al. On the use of unmanned aerial systems for environmental monitoring. *Remote Sensing* 2018; 10 (4): 641. doi: 10.3390/rs10040641
- [9] Barrenetxea G, Ingelrest F, Schaefer G, Vetterli M, Couach O et al. Sensorscope: out-of-the-box environmental monitoring. In: IEEE 2008 International Conference on Information Processing in Sensor Networks; Missouri, USA; 2008. pp. 332-343. doi: 10.1109/IPSIN.2008.28

- [10] Hakala I, Tikkakoski M, Kivela I. Wireless sensor network in environmental monitoring-case foxhouse. In: IEEE 2008 International Conference on Sensor Technologies and Applications; Cap Esterel, France; 2008. pp. 202-208. doi: 10.1109/SENSORCOMM.2008.27
- [11] Quaritsch M, Kuschnig R, Hellwagner H, Rinner B. Fast aerial image acquisition and mosaicking for emergency response operations by collaborative UAVs. In: International Conference on Information Systems for Crisis Response and Management; Lisbon, Portugal; 2011. pp. 1-5.
- [12] Tomic T, Schmid K, Lutz P, Domel A, Kassecker M et al. Toward a fully autonomous UAV: research platform for indoor and outdoor urban search and rescue. IEEE Robotics and Automation Magazine 2012; 19 (3): 46-56. doi: 10.1109/MRA.2012.2206473
- [13] Alsalam BHY, Morton K, Campbell D, Gonzalez F. Autonomous UAV with vision based on-board decision making for remote sensing and precision agriculture. In: IEEE 2017 Aerospace Conference; Montana, USA; 2017. pp. 1-12.
- [14] Hung C, Xu Z, Sukkariieh S. Feature learning based approach for weed classification using high resolution aerial images from a digital camera mounted on a UAV. Remote Sensing 2014; 6 (12): 12037-12054. doi: 10.3390/rs61212037
- [15] Xiang H, Tian L. Method for automatic georeferencing aerial remote sensing (RS) images from an unmanned aerial vehicle (UAV) platform. Biosystems Engineering 2011; 108 (2): 104-113. doi: 10.1016/j.biosystemseng.2010.11.003
- [16] Casper J, Murphy RR. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 2003; 33 (3): 367-385. doi: 10.1109/TSMCB.2003.811794
- [17] Ludovisi R, Tauro F, Salvati R, Khoury S, Mugnozza GS et al. UAV-based thermal imaging for high-throughput field phenotyping of black poplar response to drought. Frontiers in Plant Science 2017; 8: 1681. doi: 0.3389/fpls.2017.01681
- [18] Arnold T, De Biasio M, Fritz A, Letiner R. UAV-based multi-spectral imaging. In: IEEE Sensors Conference; Hawaii, USA; 2010. pp. 1-4. doi: 10.1109/ICSENS.2010.5690923
- [19] Lozano-Perez T, Wesley MA. An algorithm for planning collision-free paths among polyhedral obstacles. Communications of the ACM 1979; 22 (10): 560-570. doi: 10.1145/359156.359164
- [20] Coombes M, Chen WH, Liu C. Flight testing boustrophedon coverage path planning for fixed wing UAVs in wind. In: IEEE 2019 International Conference on Robotics and Automation; Montreal, Canada; 2019. pp. 711-717. doi: 10.1109/ICRA.2019.8793943
- [21] Lewis JS, Edwards W, Benson K, Rekleitis I, O’Kane JM. Semi-boustrophedon coverage with a dubins vehicle. In: IEEE 2017 International Conference on Intelligent Robots and Systems; Vancouver, Canada; 2017. pp. 5630-5637. doi: 10.1109/IROS.2017.8206451
- [22] Bahnemann R, Lawrence N, Chung JJ, Pantic M, Siegwart R et al. Revisiting boustrophedon coverage path planning as a generalized traveling salesman problem. In: 12th Conference on Field and Service Robotics; Tokyo, Japan; 2019. pp.1-14.
- [23] Yildirim Ö, Diepold K, Vural RA. Decision process of autonomous drones for environmental monitoring. In: IEEE 2019 International Symposium on Innovations in Intelligent Systems and Applications; Sofia, Bulgaria; 2019. pp. 1-6. doi: 10.1109/INISTA.2019.8778341
- [24] Wang C, Soh Y, Wang H, Wang H. A hierarchical genetic algorithm for path planning in a static environment with obstacles. In: IEEE 2002 Canadian Conference on Electrical and Computer Engineering; Manitoba, Canada; 2002. pp. 1652-1657. doi: 10.1109/CCECE.2002.1013004
- [25] Chu K, Lee M, Sunwoo M. Local path planning for off-road autonomous driving with avoidance of static obstacles. IEEE Transactions on Intelligent Transportation Systems 2012; 13 (4): 1599-1616. doi: 10.1109/TITS.2012.2198214
- [26] Ramirez-Serrano A, Boumedine M. Real-time navigation in unknown environments using fuzzy logic and ultrasonic sensing. In: IEEE 1996 International Symposium on Intelligent Control; Michigan, USA; 1996. pp. 26-30. doi: 10.1109/ISIC.1996.556172

- [27] Elshamli A, Abdullah HA, Areibi S. Genetic algorithm for dynamic path planning. In: IEEE 2004 Canadian Conference on Electrical and Computer Engineering; Ontario, Canada; 2004. pp. 677-680. doi: 10.1109/CCECE.2004.1345203
- [28] Lehnert C, English A, McCool C, Tow AW, Perez T. Autonomous sweet pepper harvesting for protected cropping systems. *IEEE Robotics and Automation Letters* 2017; 2 (2): 872-879.
- [29] Hofner C, Schmidt G. Path planning and guidance techniques for an autonomous mobile cleaning robot. *Robotics and Autonomous Systems* 1995; 14 (2-3): 199-212.
- [30] Kunz C, Murphy C, Camilli R, Singh H, Bailey J et al. Deep sea underwater robotic exploration in the ice-covered arctic ocean with AUVs. In: IEEE 2008 International Conference on Intelligent Robots and Systems; Nice, France; 2008. pp. 3654-3660.
- [31] Shiu BM, Lin CL. Design of an autonomous lawn mower with optimal route planning. In: IEEE 2008 International Conference on Industrial Technology; Chengdu, China; 2008. pp. 1-6. doi: 10.1109/ICIT.2008.4608497
- [32] Belkhouche F, Bendjilali B. Reactive path planning for 3-D autonomous vehicles. *IEEE Transactions on Control Systems Technology* 2011; 20 (1): 249-256. doi: 10.1109/TCST.2011.2111372
- [33] Acar EU, Choset H, Rizzi AA, Atkar PN, Hull D. Morse decompositions for coverage tasks. *International Journal of Robotics Research* 2002; 21 (4): 331-344. doi: 10.1177/027836402320556359
- [34] Barraquand J, Latombe JC. Robot motion planning: a distributed representation approach. *International Journal of Robotics Research* 1991; 10 (6): 628-649.
- [35] Choset H. Coverage of known spaces: the boustrophedon cellular decomposition. *Autonomous Robots* 2000; 9 (3): 247-253.
- [36] Bernard M, Kondak K, Maza I, Ollero A. Autonomous transportation and deployment with aerial robots for search and rescue missions. *Journal of Field Robotics* 2011; 28 (6): 914-931.
- [37] Martínez-de Rios JR, Merino L, Ollero A. Fire detection using autonomous aerial vehicles with infrared and visual cameras. In: IFAC World Congress; Prague, Czech Republic; 2005. pp. 660-665.
- [38] Ohashi M, Kagawa Y, Nakatsuka T, Ishida H. Crayfish robot that generates flow field to enhance chemical reception. *Journal of Sensor Technology* 2012; 2 (4): 185. doi: 10.4236/jst.2012.24026