|| 26- 28th March, 2021 ||

INTER IIT TECH MEET'21

IIT Guwahati

MID PREP EVENT

(250 Points)

DRDO DRGE'S VISION BASED OBSTACLE AVOIDANCE DRONE

Recent flash floods in Uttarakhand and similar disasters in the recent past have thrown difficult challenges to search & rescue parties. Inaccessible areas, narrow valleys, tunnels, and large swaths of affected areas have tested human capabilities in terms of perseverance and speed. Faster response rates and the ability to negotiate complex environments are the keys to successful search & rescue operations. As the world is rapidly advancing towards automation; smart and intelligent aerial vehicles are the next-gen key advancement in the field.

The ability to sense objects in real-time and avoid imminent collisions is the basic requirement of autonomous drone operations in such environments. Drone obstacle avoidance is highly contextual, and hence requires intelligent algorithms and well-designed workflows implemented in robust software.

PROBLEM STATEMENT:

The task is to design an autonomous drone that navigates in a complex static environment by avoiding any collision with the on-field obstacles and reaching the target destination after its correct detection.

- Design an algorithm that autonomously navigates a drone from point A to point B avoiding collisions with the obstacles present in the path.
- The local coordinates of Point A would be known before-hand and the drone has to detect and navigate up to Point B (that would be an Aruco Marker).
- The task is considered to be completed if the drone lands on Point B Aruco Marker) without any crash. The ROS package must publish the ID of the Aruco marker and string “Landed” after landing on to topic “/Aruco/message”

    a. When not detected: “Marker ID: none, looking for marker”
    b. When detected and landed: “ Marker ID : 0, Landed”

- Some parts of the Aruco marker must be visible to the RGB camera upon landing.
- Multiple Aruco Markers (false) may or may not be provided. The drone has to correctly identify the Aruco Marker based on the ID provided before landing. Correct Aruco ID will be ‘0’ in all the world.
- The drone model will be provided with a forward-facing depth camera and downward-facing RGB camera only (Any other sensors cannot be used).
- The flight should be strictly restricted to a height of 5m only.

The final flight will be simulated on a standardized environment and computer system of the judge.

SOFTWARE SPECIFICATIONS:

- Ubuntu 18.04
- ROS Melodic
- Gazebo 9
- Ardupilot Firmware
- Python or C++
- Any extra library used must be specified in the documentation and must be open source.

Any submission not aligning with the first 5 conditions won't be accepted since it needs to be tested out on a standardized environment and computer system.

GUIDELINES:

- Everything must be done in simulation using ROS and Gazebo with ArduPilot Flight Controller software.
- Use Ubuntu 18.04 and ROS Melodic and Gazebo9 (any others will not be accepted).
- You will be provided with installation instructions (Ardupilot) and a package with a drone model with given sensors.
- You will be given 3 test worlds to test your algorithm and the final worlds used for evaluation will be similar to the worlds given and will not be shared.
- Your submission format must be a ROS package and everything needed to run must be within a single bash script or a launch file.
- If any extra dependencies or packages are needed for your simulation, you need to provide proper documentation to install them. Every package you use must be open source.
- Three different gazebo worlds consisting of different types of obstacles will be released at a fixed time interval.
- Teams are required to work on such an algorithm that fits all the three worlds provided.
- Only one final submission will be accepted.
- The team has to submit their final ROS package along with proper documentation for its installation. Documentation must also include working principles and algorithm explanations.

RELEASE DATES:

- **World 1:** **6th March, 2021**
- **World 2:** **12th March, 2021**
- **World 3:** **18th March, 2021**

A maximum of 8 participants (per team) shall be awarded a participation /merit certificate.

EVALUATION CRITERIA:

The evaluation will be done based on the following factors:

- Ease of Installation of the ROS package along with dependent libraries.(100 marks)
- Concise documentation and presentation skills.
- The marking scheme is divided into 3 components :

COMPONENT 1 :

- Successful completion and execution of a task- "world".

WORLD NO	MARKS UPON COMPLETION OF TASK	GRACE MARKING
World 1	100	(x/L)*50
World 2	200	(x/L)*100
World 3	300	(x/L)*150

x : Drone distance from initial position
L : Total distance of Marker B from Initial point

COMPONENT 2 :

- Time of flight: For each world, the time of flight will be noted for the teams who have completed the world. Marks will be distributed according to the following chart. Grading would be relative with respect to top 5 teams wrt time of flight.

TEAM	TOF	MARKS ALLOTTED
Team 1	t1	0.5 * (M)
Team 2	t2	0.4 * (M)
Team 3	t3	0.3 * (M)
Team 4	t4	0.2 * (M)
Team 5	t5	0.2 * (M)

- 0.1 * (M) for rest of the teams who completed the world.
- t1 < t2< t3< t4< t5
- M : Max. marks allotted for particular World on completion.
  M = 100 for World 1
  = 200 for World 2
  = 300 for World 3

EVALUATION CRITERIA:

COMPONENT 3 :

- The computational power used, lesser power implies better performance (Jetson TX2 Module is the upper limit).

TEAM	COMPUTATIONAL ELEMENT UTILISED (CPU, GPU, RAM)	MARKS ALLOTTED
Team 1	f1	0.5 * (M)
Team 2	f2	0.4 * (M)
Team 3	f3	0.3 * (M)
Team 4	f4	0.2 * (M)
Team 5	f5	0.2 * (M)

- 0.1 * (M) for rest of the teams who completed the world.
- f1 < f2< f3< f4< f5
- M : Max. marks allotted for particular World on completion.
  M = 100 for World 1
  = 200 for World 2
  = 300 for World 3

TOTAL MAXIMUM POINT DISTRIBUTION PER TASK

WORLD	MAXIMUM POINTS
WORLD 1	200
WORLD 2	400
WORLD 3	600