Flipkart

GRiD 2.0

Autonomous Indoor Drone

Team Name : Team Alpha01

Institute Name: Indian Institute of Technology, Bhubaneswar

Team members details

Team Name	Team Alpha01				
Institute Name	Indian Institute of Technology, Bhubaneswar				
Team Members >	**1 (Leader)**	**2**	**3**	**4**	**5**
Name	Saswat Kumar Dash	Prajwal V. Sangam	Smarth Bansal	Gaganpreet Singh	Jamma Sathwik
Batch	2018-22	2018-22	2018-22	2018-22	2018-22
Area of expertise	Robotics Motion Planning	Control System	Robotics Motion Planning	Simulation	Image Processing

Functionalities of the Robot

- **What all can the robot do?**
  Robot can move in 3D space avoiding obstacles. It captures real-time video and detects gates and obstacle. *Minimizes time objective function* to reach goal.
- **What all activities can it perform?**
  It can take smooth turns and carry payloads. If at beginning the goal is not visible it still can reach goal by considering waypoints as interval goal and proceed forward.
- **Are there any things that the robot can do above and beyond the requirement?**
  The *length and breath of gates* along with its height from ground can *vary* (dimensions limited by drone size)
- **Are there any out of the box functionalities?**
  It can *detect gates which are at an angle* to horizontal and also detect gates which are *discontinuous* by a proposed Gate algorithm.

Robot Specifications

Technical Specification:

-> Raspberry Pi zero w.
-> 9dof IMU by Generic.
-> WaveShare BMP388 High precision Pressure Sensor.
-> Generic AX168 1600kv A2212 Brushless motor.
-> 4 x Generic AX168 1600kv A2212 Brushless motor.
-> Lipo Battery 4200mAh/11.1v.
  Weight : 0.4
  C rating : 30C
-> Camera Specifications:(Parrot BeBop Drone Quadcopter with 14-Megapixel Flight Camera (Red)
  Sensor : 1/2.3" CMOS
  Resolution : Photo: 14 MP (3800 x 3188) Video: 1080p30
  Image Stabilization : 3-axis electronic image stabilization
  Field of view : 180°
    Note: The effective field of view for video is less than 180° to accommodate the electronic image stabilization system.
-> Paparazzi open source

Physical Specifications:

->Frame size : The diagonal length of the drone will be around 40cm and thickness of frame around 2.5cm.

->Propeller Diameter: 16cm

Robot/Solution Limitations

Solution:

Obstacle Detection and Gate detection:

Cameras are used along with Image Processing to draw a map of the Environment (immediately visible on sight) and then applies gate detection algorithm **[1]** to produce a 3D map and gives it as input to Path planning module. Trajectory Planning module smoothen the path and gives its desired inputs of velocity,acceleration,torques to control module.

Controller Design:

Higher order Integral Sliding Mode controller **[6]** is used. This makes drone robust to payload variation under a limit. Integral SMC is used as it is robust to parameter uncertainty. The model doesn’t behave abruptly during loading or unloading of payload.

Trajectory Tracking:

As trajectory tracking is concerned our approach is similar to one mentioned in the reference**[2]**. The approach in the paper provides choice for following the waypoints according to the requirement. The model may be tuned to exactly follow the waypoints or to omit some of the waypoints depending upon the robot state(velocities, attitude, position).

Mechanical Hook:

Adjustable hook to carry payload.

Limitations:

What can the robot not do ?

Our design is not capable of performing acrobatics or fast maneuvers.

Payload cannot be varied widely. The range of payload has to be specified and controller has to be tuned according to it.

The controller is not designed to handle hanging payloads.

- It seems that our drone seems to meet all the mentioned requirements.

Path Planning

Algorithm:

Rapidly-Exploring Random Tree (RRT) **[3]**

Sample Based Algorithm, creates random samples that evolves into a graph.

RRT* algorithm gives an optimized path without any zig-zag routes preventing jittering of the drone.

Creates a graph and finds a path simultaneously, reduces the burden of extra code to find a path separately.

PseudoCode:

Current position of the drone starts a new Random Tree with one parent

A new node is chosen randomly, the point is validated across the following checks:

1. Node Obstacle Collision Test - Checks if the node is feasible to pass
2. Drone Objective Criteria Check - Checks to see if the new node violates the boundary criteria set by the objective.

The new node is connected to the nearest node as a child. A Path Obstacle Collision Test is performed on the path before it is added as a child.

Path Planning

Each time a new node is added to the tree, it is checked whether the goal is reachable.

New nodes are added to the graph, until the objective is reached, or a fulfilling criteria is achieved.

Path Smoothing:

Each node is checked against other nodes and reconnected to the graph based on the combined shortest length forwards. This makes the graph smooth itself as the number of nodes are continuously increased.

Running the algorithm, repeatedly, with different controls will lead the drone inevitably to the goal.

An attempt to add Non-holonomic characteristics:

- The proposed skeleton-biased locally-seeded RRT (skilled-RRT)**[4]** is faster than the other baseline planners (RRT, RRT*, A*-RRT, Theta*-RRT, and MARRT) through experimental tests
- To minimize the planning time, we adopt the idea of biasing the RRT tree-growth in more focused ways
- Extended from 2D to 3D space

Robot Visualization -3D Diagram/Sketch

Brief on Programming Module

MODULES BUILT DURING THE PROTOTYPE:

Programming Language: **[5]**

All of the algorithms will be coded in Python.

Obstacle Detection:

Image Processing using OpenCV module in python. Detection of Gates and then creating a map of environment and feeding its output to Path Planning Module.

Path and Trajectory Planning:

Use RRT* algorithm in python and smoothen the curve using Cubic Spline Algorithm.

Controller and Trajectory tracking:

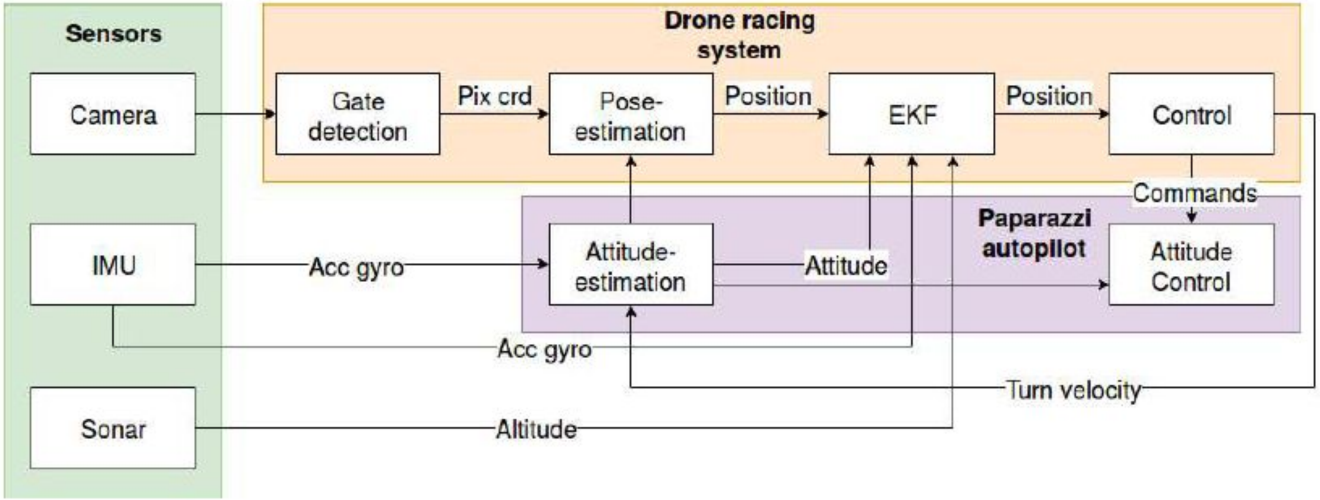
The controller design will be done in MATLAB SIMULINK APP and CONTROLLER TOOLBOX is also used. The resulting design is converted into [c c++ python].

IMU Calibration and installation:

For IMU calibration open-source code from github is used. The used repository is MomsFriendlyRobotCompany/mpu9250.

Execution Plan

High level action items in terms of what will be the steps from the drawing board to the actual prototype.



References

1. Shuo Li, Christophe De Wagter, Guido de Croon- Autonomous drone race: A computationally efficient vision-based navigation and control strategy, September 2018 https://www.researchgate.net/publication/327717731_Autonomous_drone_race_A_computationally_efficient_vision-based_navigation_and_control_strategy
2. Péter Bauer, József Bokor - Tuning and Improvements in a Waypoint and Trajectory Tracking Algorithm, August 2012 https://www.researchgate.net/publication/268557808_Tuning_and_Improvements_in_a_Waypoint_and_Trajectory_Tracking_Algorithm
3. Mechali Omar,Limei Xu,Mingzhu Wei-A Rectified RRT* with Efficient Obstacles Avoidance Method for UAV in 3D Environment,May 2019 https://www.researchgate.net/publication/332934508_A_Rectified_RRT_with_Efficient_Obstacles_Avoidance_Method_for_UAV_in_3D_Environment
4. Yiqun Dong,Faster RRT-based Nonholonomic Path Planning in 2D Building Environments Using Skeleton-constrained Path Biasing https://link.springer.com/article/10.1007/s10846-017-0567-9
5. Codes of python Resource : https://pythonrobotics.readthedocs.io/en/latest/
6. https://www.researchgate.net/publication/319987946_Adaptive_Robust_Control_of_Quadrotor_Helicopter_towards_Payload_Transportation_Applications

Flipkart

GRiD 2.0