Autonomous Unmanned Micro Aerial Vehicles for Reconnaissance

Pratik Sen, Vivek Radhakrishnan, Abishek Krishnamoorthy,

Aman Jotwani,Nitish Anand, Vishank Bhatia, Dr. R.K Mittal

Birla Institute of Technology and Science, Pilani-Dubai Campus, Dubai, UAE

ABSTRACT

The Intelligent Flying Object for Reconnaissance (IFOR) is an autonomous aerial vehicle that has been developed by students of BITS Pilani, Dubai Campus. The vehicle is capable of localizing itself using the SLAM algorithm, stabilize its attitude (pitch, roll and yaw) and altitude using PID controllers, plan paths around obstacles and navigate an unknown indoor environment with wall following guidance. In addition, it has been designed to be capable of pattern recognition which would enable it to recognize images and signs.

In this iteration of the IFOR system a subsidiary child robot is slaved to the IFOR robot. Which will serve as a mobile sensory and manoeuver platform for the entire system. The drone will allow the entire system greater flexibility in terms of overall decisions available to the robot.

These features enable the IFOR to execute the 6th mission of the International Aerial Robotics Competition, which involves scanning an unknown indoor arena protected by laser barriers and cameras, bestrewn with obstacles, in search of a flash drive.

1. INTRODUCTION

The field of robotics is witnessing a paradigm shift in the operation and use of robots. With various robots becoming autonomous and intelligent day by day, their application and use has increased tremendously. *Aerial robots* specifically have an edge over other autonomous vehicles due to their higher degrees of freedom and agile maneuverability. To tap these features and advance the applications and versatility of the vehicle is the aim of the team.

1.1 Problem Statement

The 6th mission of the IARC requires teams to construct a fully autonomous aerial robot capable of self-controlled flight within a confined environment. The vehicle will first be required to enter a "military" facility through a one square meter or larger opening from a designated launch area 3m away. The vehicle will have to search for a target area ("Office of the Chief of Security") while avoiding unknown obstacles such as walls, columns and furniture as well as visible security systems like a video camera and a laser barrier in a hallway.

The building will contain several signs indicating the route to the target area as well as appropriate indicators allowing avoidance of the security systems. Once the target area is found the vehicle is expected to locate a target object (Black colored pen drive), pick it up and drop a decoy in its place before proceeding to fly back out of the window which it had entered through.

1.2 Conceptual Solution

We have designed an autonomous vehicle which leverages the use of an off the shelf quadrotor as the base platform. This has allowed us to focus on the autonomy of the vehicle instead of the intricacies of flight dynamics. We have used the Ascending Technologies' (Asctec) Pelican as the quadrotor of choice for its payload capacity and high structural integrity. A quadrotor, by nature, is aerodynamically unstable. This demands the use of a Stability Augmentation System (SAS) which the Pelican is already equipped with. The SAS relies on data from the Inertial Measurement Unit (IMU) to keep the vehicle stable in flight. We have equipped the quadrotor with a scanning laser range finder to enable external sensing of the surroundings. Simultaneous Localization and Mapping (SLAM) is used to calculate the coordinates of the vehicle in a global frame which are used to correct drift. The vehicle makes use of two camera's one situated in the front and the other at the bottom. The camera is used by the image processing module to detect and identify objects and words through the use of the ORB descriptors. All the modules are monitored by a Base Station which serves as the link between all systems and also allows for changes to be made dynamically to each of them. All processing of sensor data is done off board on the base station. The sensor data is pre-processed on board the vehicle itself.

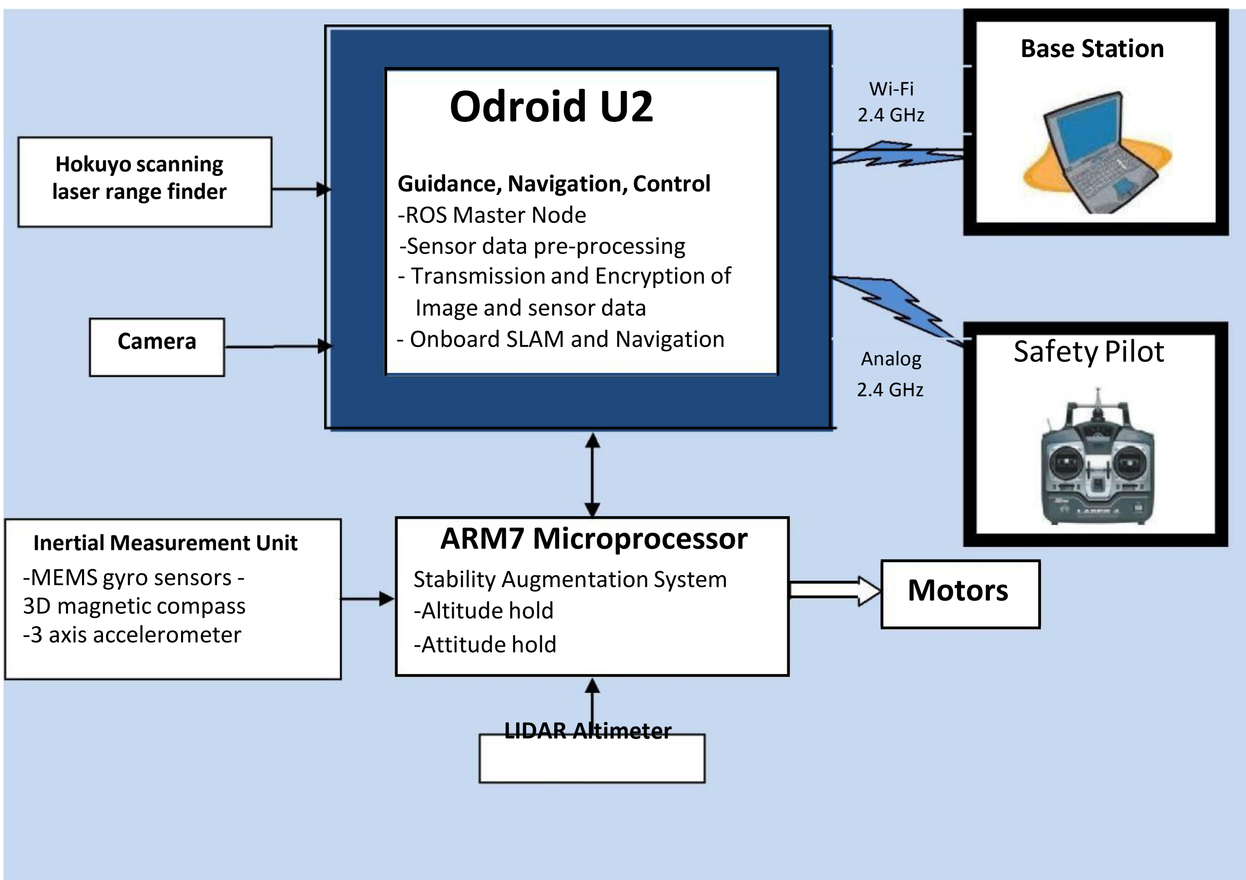

Figure1: System architecture

1.3 Yearly Milestones

In our last attempt at the 6th Mission, we designed to design a vehicle that is capable of flying autonomously by performing drift control, intelligent navigation through the arena, path planning and robust image based object detection. We also had shifted to the ROS software backbone for easy standardization of our system.

This year we present a new multi robot system so as to address problems that are difficult to attend with one robot. To cover the IFOR UAV’s inherent deficiencies in terms of maneuverability, a secondary autonomous drone is used in tandem with the IFOR robot. For this purpose the Parrot AR Drone was selected. The IFOR and Parrot AR drone will have a Parent Child Hierarchy, respectively. The IFOR drone will primarily provide its services such as SLAM, Navigation and Mission Control for both UAV’s. On the other hand, the Parrot AR will provide its camera feed, and superior flight capabilities to execute more complicated procedures such as Pen drive Search and Pickup.

2. Air vehicle

2.1 Asctec Pelican

The Asctec Pelican, is a development platform for the design of a quadrotor based UAV’s. It’s a robust platform running attitude control algorithms at 1000Hz thus providing a very stable vehicle for the sensor suite. The quadrotor also provided filtered IMU data for use in other algorithms hence allowing for more robust pose and heading estimation.

The Pelican weighs about 980 g and has a payload capacity of 500 g. The payload capacity of the Asctec Pelican meets the flight time required, when loaded with the sensor suite.

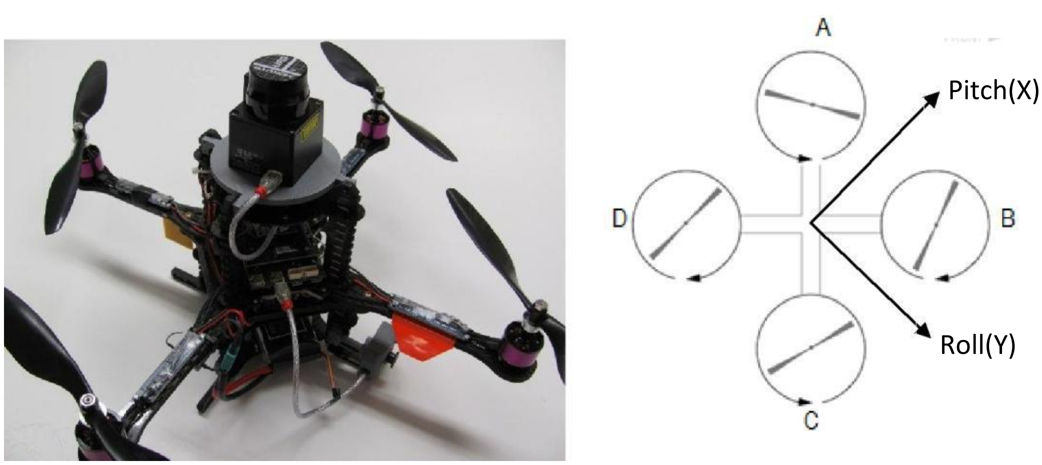

Figure 2: The Asctec Pelican quadrotor and the propeller action.

2.2 Parrot AR Drone

The Parrot is a relatively cheap of the shelf quadrotor drone, weighing less than 500 grams in total. The parrot is far more agile and lighter than the Asctec Pelican, however although it does contain an internal computer, it is closed off to developers. Access is only possible via the provided API. Hence, no custom onboard algorithms can be run on the ArDrone. It is for this reason the Parrot is slaved to the Asctec Pelican.

The Parrot contains a variety of sensors inbuilt into the quadrotor, including a 6DOF MEMS, Optical Flow Sensor, a Sonar Altimeter and two cameras. The drone provides accurate quadrotor state via sensor fusion between the altimeter, IMU and optical flow sensor. The ARDone has exceptional stability and is used as faux appendage for the Asctec Pelican.

Navigation & Path Planning

SLAM

SLAM is an acronym for the Simultaneous Localization and Mapping algorithm. It consists of multiple parts; landmark extraction, data association, state estimation, state update and landmark update. The algorithm estimates the apparent change in position of landmarks (unique regions) between the current scan and the map assuming the surroundings to be stationary. The vector opposite to this one represents the change in position of the vehicle. “Global” coordinates hence calculated are used to generate a map of the environment. Therefore, the map and coordinates are simultaneously estimated from each other.

We use the Hector SLAM approach to the Simultaneous Localization and Mapping problem, a robust and efficient algorithm with good integration with the ROS framework. The Hector SLAM solution uses a 2D grid map representation which is updated using a scan matching approach. The algorithm has low runtime requirements and can run with an update rate of 40Hz all the while reducing the computation required as compared to other solutions to the SLAM problem. The system uses odometry data, along with the scan matching approach to result in a very robust estimation of pose of the vehicle. The input used for solving the SLAM problem are laser scans from the lidar as well as odometry data from the IMU and optical flow from the PX4FLO; the robot state as estimated by the navigation filter. Subsequently all sensor readings are fused using a 6DOF EKF, so as to provide an accurate guess of the quadrotors state. Data provided by the navigation filter is used for transformation of laser scans to take into account the attitude of the laser scanner and vehicle during acquisition of scans.

To enable autonomous cumulative mapping so as to take into account earlier made maps, a feature based map merging system has been developed. The robot detects SURF features for the estimated map, after which a registration approach is then used to arrive at a common coordinate frame for all pose states.

We have used the same basic SLAM algorithm as our last attempt, however with further improvements and modifications to address the changes to our overarching system. Unlike our previous attempt the SLAM algorithm will be working onboard, hence sensor data needed to be slightly filtered so as to reduce the load on our processor. Secondly , due to insufficient computation available to the ARDrone , all slam modules were run on the Asctec, however since no LIDAR was provided to the ARDrone, localization was done relative to the Asctec Pelican via visual markers and accurate state information provided by the Drone.

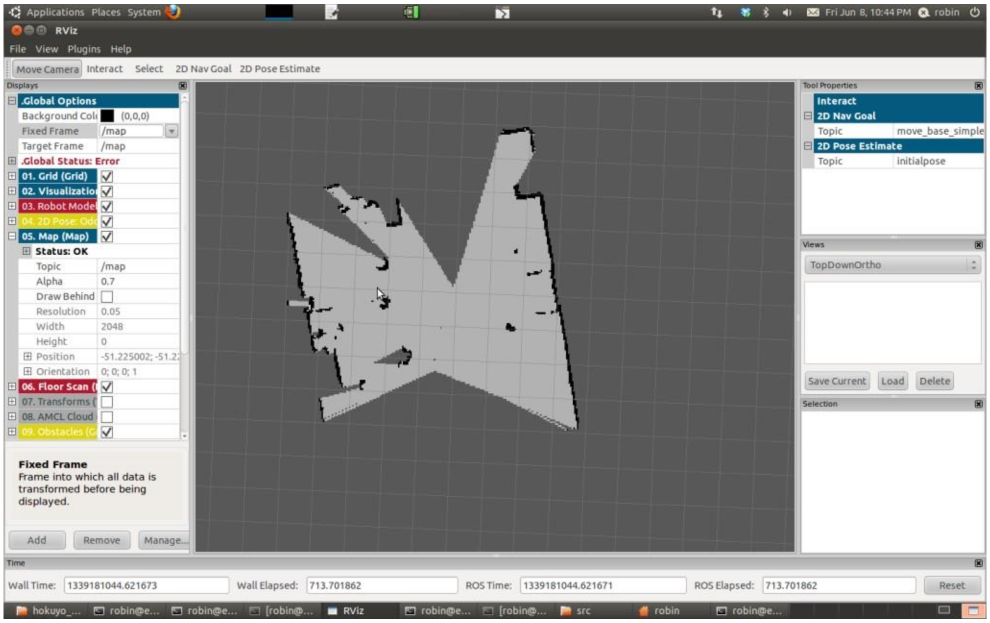

Figure 3 : Hector Mapping

Path Planning

The path planning module uses a gradient planner which uses the cost map method to calculate the optimal path to the goal from the robot. The gradient planner is a wave front planner that computes the cost of getting to the goal at every cell (20x20cm) in the map. If the cost of a cell is higher than the cost of a neighbor's value added with the local transit cost, then its value is updated. The planner takes into account the local neighborhood of the cell to update the cost of the cell. The algorithm initializes the goal with zero cost and everything else with a very large cost. All goals are put into an open list, which is then popped from the list and then each of the cell's neighbors are updated. The neighbor cells with a lowered cost are then pushed back onto the list. The algorithm completes when the open list is empty.

The algorithm computes the configuration space for a circular robot and includes safety distances to obstacles. The obstacle points are entered as goal points, and the update algorithm is run over the list, generating a new open list. This is done till the distance of the obstacles has been determined to the resolution of cycles spent on the algorithm. After which a cost is associated with the distance; infinite cost within a specified safety radius and decreasing cost moving away from the radius.

Rapid switching of global paths are avoided by including hysteresis by lowering the cost along the path. This allows the algorithm to switch paths only when there is substantial evidence of the profitability of the new path.

After creating a map in navigating to a goal the map can be re-used so as to be more efficient at the same task.. A better global path can be computed for the same task if a map of the environment is already readily available. This allows for the algorithm to plan through shorter paths and even make use of shortcut, which may have been detected and validated on an earlier run.

Figure 4 Obstacle Growth and Path Planning

Exploration

To achieve autonomous operation and added to the fact that no previous map of the environment is available we need an exploration module to determine where the quadrotor has to fly next .This is done based on the partial maps available from the SLAM module.

Our exploration module uses the Frontier-based exploration approach [2]. Frontiers are points on the boundary between explored and the unexplored space. The main idea behind this approach is that we move to the edge of the line separating known and unknown area. Once the quadrotor has reached the frontier, it can look into the unexplored space and add these areas to its map.

We have used occupancy grids as our spatial representation. Occupancy grids contain cells, where each cell stores the probability of the corresponding region of space being occupied. Initially, all the cells of the grid are set to a pre-determined probability of occupancy. Each time a sensor reading is obtained from the LIDAR, the corresponding probability is updated in the grid. After the development of the occupancy grid, each cell is classified by comparing its probability to the initial probability it was assigned. Each cell is classified as:

a) Free: current probability <Initial probability.
b) Unknown: current probability =Initial Probabilty
c) Occupied: current probability > Initial Probability

A free cell that is next to an unknown cell is classified as a frontier. Any frontier region that is greater than a predefined size is considered as a frontier.

Once frontiers have been established, the quadrotor attempts to reach the nearest unvisited, accessible frontier. To determine the nearest frontier, we used a cost based approach. Here, the exploration planner module calculates the cost to reach every frontier .The frontier with the least cost is chosen first. A search algorithm is used on the occupancy grid to find the shortest, obstacle free path to the selected frontier.

Once the quadrotor has reached its destination, the frontier is added to the list of visited frontiers. Again, readings of the LIDAR are used to update the occupancy grid. The quadrotor again determines frontiers in the updated occupancy grid and tries to visit the nearest, accessible frontier. If a particular frontier has been unvisited for some period of time i.e. the quadrotor is unable to reach that frontier then that particular frontier is blacklisted and added to the list of all the inaccessible frontiers. The quadrotor then updates the occupancy grid using the LIDAR and tries to visit the next unvisited frontier.

Drift Control

Drift control is most important to the performance of the vehicle in the mission since milligram imbalances in weight, apart from other factors, cause a quadrotor to drift unintentionally. The coordinates which are output by SLAM are used by the Drift control algorithm to estimate the difference between the desired position and the actual position. The error in meters, for pitch and roll respectively, is then used in the following PID loops which are used to command the pitch and roll of the vehicle:

$$\text{Command Pitch} = \text{Pitch Bias} + e(t) + \int dT + e(t)$$
$$\text{Command Roll} = \text{Roll Bias} + e(t) + \int dT + e(t)$$

K_c, K_p, Kd is the controller parameter. These constants are obtained as per the Zeigler-Nicholls tuning rule.

Altitude Control

The altitude of the quadrotor is estimated by deflecting the 9 beams of the LIDAR at 90 degrees, so that the beam hits the ground vertically. The average of the 9 beams is considered as the true altitude of the quadrotor. This value of the height is used in a PID control algorithm to provide the value for estimating the value of the thrust required to reach a specified altitude. The Throttle Bias value provides an option to the user to reach different setpoints with variable payload.

Figure 5: LIDAR mirror mount for height estimation

3. Payload

3.1 Sensor suite

GNC Sensors

- *Asctec Pelican Inertial Measurement Unit (IMU):* The IMU consists of MEMS gyro sensors, a triple axis accelerometer and triple axis magnetic compass. This allows for the calculation of tilt angles which can be polled up to 100 times per second.
- *Hokuyo UTM 30 LX Light Detection and Ranging Device:* A scanning laser range finder of weight 160 gm., angular resolution 0.36 degree, range 5.6 m and a frequency of 10 Hz
- *Point Grey Firefly MV cameras*: The cameras provide data to the *IP* module which allow for detection of the Blue LED and laser barriers in addition to recognition of signboards
- *PX4FLO*: The sensor provides accurate odometry data via optical flow, allowing for a much more efficient localization of the robot.

3.2 Mission sensor

Image Processing (IP) module

We have equipped the vehicle with two high resolution cameras that provide the input for the image processing module. The cameras have been placed on the front and bottom of the quadrotor. The front facing camera will provide the inputs for the word recognition algorithm in order to identify the signboard of the "Office of the Chief of Security". The competition involves recognizing a given set of Arabic letters from a set of words; the target room can be identified by this particular word. We do word identification by calculating a shape descriptor for the entire work via ORB (Oriented Robust Brief), and then matching these descriptors with the those found in the target sign board via the RANSAC brute forcing method. We identify the words by their shape instead of the individual letters contained within. This approach is similar to how humans reading works, and provides a better performance in matching and identification than normal OCR based methods.

As a initial step of filtering for the pendrive detection process, we begin with a a K-Means based segmentation for the background and foreground. The K-Means algorithm picks out the picks out the objects of interest from the image. Afterword's we use a silhouette based matching to find the interest objects that match the shape of the pendrive. For this we have used Curve Space descriptors, which are a good way to find the affine transformation between two contours. After the previous shape are found we then use those portions of the image and affirm the pendrive using ORB based feature matching. We also use the Affine transform between the contour and the silhouette to calculate and then apply the required perspective transform, for better matching performance.

Figure 6 : SURF Bruteforce Matching via RANSAC

3.3 Communications

The vehicle makes use of multiple communication links with the base-station via WiFi and Xbee modules. The Xbee module is used for sending the control input to the vehicle from the base station through the reliable and fast Zigbee protocol. We use the Wi-Fi link to transmit both the LIDAR and the video data, the handling of the sensor data is managed be the ROS Messaging Framework. A safety pilot can take control over the vehicle at any time using a Futaba radio controller operating at 2.4 Ghz.

3.4 Power system

The quadrotor is powered by an 11.1V Lithium Polymer Battery. A power board is used to distribute power and communication lines to all motor controllers and other systems on board. The power board comprises of a switching power regulator to generate a stable 6V supply for the Auto Pilot board and a high power MosFET to switch current ON and OFF.

4. Operation

4.1 Flight preparation

Each flight test is performed with utmost precaution following the mentioned safety procedure which ensures a safe and smooth flight of the quadrotor.

4.2 Checklist

1. Double Check LiPo battery voltage using voltmeter
2. Examine the propellers, safety mounts, nuts and screws for any damage
3. Test communication link between the quadrotor and the Ground station
4. Enable safety pilot and check kill switch action before flight
5. Check status LED"s

5. Risk reduction

5.1 Vehicle status

Two status LEDs allow for a check on certain critical vehicle states. The blinking of a Red LED on the processor board of the Pelican indicates the initialization and calibration of the sensors. Once the sensors are calibrated a green LED blinks rapidly indicating the flight control software is running. In case the battery voltage drops under 9.8 volts a loud tone is emitted, with the beeping becoming faster as the voltage drops.

5.2 Shock / Vibration isolation

The Asctec Pelican is built on a carbon fiber frame which has a very high Ultimate Tensile Strength (UTS); thus is it can withstand a heavy impact without necking. It is also fitted with soft pads below the arms to cushion impacts.

5.3 EMI/RFI Solutions

The magnetometer is very sensitive to EMI, hence it is mounted above all electronics such as IMU and the processors.

Safety

The quadrotor though autonomous in its flight can also be manually commanded to abort. A safety kill switch mechanism has been developed in order to attain this feature. The Pelican has a built in termination system due to which, the vehicle lands the moment the transmitter falls out of range. This feature is used for flight termination via a kill switch which shuts the transmitter off and hence causes the vehicle to descend.

The IFOR is equipped with landing gear designed in a manner to deflect shock from the electronic system. The propellers are also covered, which ensures safety to both bystanders and the vehicle in case of a mishap.

The ON/OFF switch on board is designed active low, so if for some reason the mechanical switch breaks or loses connection the vehicle will remain ON. However this mechanism is overridden by the safety kill switch.

Testing

Device / Routine	Testing
HOKUYO LIDAR	Tested while running the intelligence software.
SLAM	Real time execution and experimental Determination of localization accuracy
Cameras	Real time execution and experimental Determination of localization accuracy.
Drift Control	Onboard as well off board testing for suitability to tuning

Conclusion

We have designed IFOR to be a fully autonomous quadrotor that would be able to successfully accomplish the tasks of the IARC. The IFOR system comprises of Simultaneous Localization and Mapping which has been optimized for real-time localization, drift control using simple PID controllers and a detailed set of Intelligence algorithms. The navigation of the quadrotor is handled by the path planning module ensuring that indoor environments can be explored regardless of the presence of unknown obstacles. Finally, image processing enables the quadrotor to scan for patterns, edges and symbols and make corresponding control outputs through a master program in order to guide the vehicle to its target. We have also have managed to perform the coordinated flight of two independent flight systems .

References

1 . Johnson, E. N. and Schrage, D. P., "System Integration and Operation of a Research Unmanned Aerial Vehicle," *AIAA Journal of Aerospace Computing, Information and Communication, Vol. 1, No. 1, Jan 2004, pp. 5 –18.*

2. Stefan Kohlbercher , Oskar von Stryk et al " A Flexible and Scalable SLAM system with Full 3D Motion Estimation.

4. Andrew J. Davison. Real-time simultaneous localization and mapping with a single camera. *Ninth IEEE International Conference on Computer Vision (IC CV'03) - Volume 2* , 2003.

5 . Chowdhary, G. V . and Johnson, E. N., "Theory and Flight Test Validation of Long Term Learning Adaptive Flight Controller ", *Proceedings of the AIAA Guidance Navigation and Control Conference, Honolulu, HI, 2008.*

6. Michelson, R., Rules for the International Aerial Robotics Competition 6th Mission, http://iarc.angel-strike.com/IARC_6th_Mission_Rules.pdf

7. Hoffmann, G. M., Huang, H., Waslander, S. L., & Tomlin, C. J. (2007). Quadrotor Helicopter Flight Dynamics and Control Theory and Experiment. *AIAA Guidance, Navigation and Control Conference and Exhibit*, *20*(23), 1-20.

8. Abraham Barach ; Nicholas Roy Et al. RANGE – Robust Autonomous Navigation in GPS-denied Environments. Published in the Journal of Field Robotics (JFR 2011)

9. Brian P Gerkey , Kurt Konolige. Planning and Control in Unstructured Terrain

10. Robin Singh, Aditya Jain, Pratik Sen, Vivek Radhakrishnan, Dr Anand Kumar, Autonomous Unmanned Aerial Vehicle for Reconnaissance based on Robotic Operating System(2012)