Unmanned Aerial Vehicle of BITS Pilani, Dubai Campus for the International Aerial Robotics Competition 2015

Syed Zeeshan Ahmed, Ganesh Ram R.K., Rochal Saxena, Aashish Dugar, Ketul Patel, Ayanava Sarkar, Shreya Jain, Vikhyat Rawla , Saptadeep Debnath, Geet Capoor, Dr.V.Kalaichelvi, Prof. Dr. R. N. Saha

Birla Institute of Technology and Science, Pilani – Dubai Campus

ABSTRACT

This paper describes IFOR (Intelligent Flying Object for Reconnaissance) an autonomous aerial vehicle that has been developed by BITS Pilani, Dubai Campus students. The vehicle can localize and navigate in environments independent of external navigation aids such as GPS or large stationary points of reference (e.g. walls) for SLAM algorithms. Downward-facing cameras and optical flow via Px4Flow is used to localize and stabilize the attitude (pitch, roll and yaw) and altitude of the vehicle using PID controllers. It also identifies the autonomous ground robots by the forward-facing camera and avoids obstacle ground robots by setting the area near the paths of obstacle ground robots as hazardous area.

INTRODUCTION

Statement of the Problem

The 7th Mission of IARC 2015 requires teams to build a full autonomous aerial vehicle capable of flight in a sterile indoor environment. The competition arena consists of a 20mx20m square which is completely GPS denied and also devoid of any physical landmarks to prevent the use of SLAM. The boundaries of the square arena will consist of wide white lines on the sides, red and green wide lines on the top and bottom respectively as shown in fig 1.

Ten programmable iRobot Creates will be placed at the center of the arena at the start and will begin to move outward towards the edges; upon collision with other robots they will re-orient themselves so as to generate randomness in their motion. Their direction of motion can be manipulated by tapping on it (which will trigger a 45° turn) or by landing in front of it (which will trigger a 180° turn).

The main objective is to herd the said robots across the green line and to prevent them from going out of bounds. This challenge requires teams to demonstrate two new behaviours, which have never been demonstrated before. Firstly, interaction between the aerial robots and the moving ground robots, which involves identification and distinguishing from the obstacle robots. Second, localization and navigation in the sterile environment. All of the said behaviours should be demonstrated under 10mins hence UAV must be smart enough to make strategic decision and.

Conceptual Solution to the Problem

Our proposed solution involves the use of a single UAV which will be performing all the necessary operations mentioned in the problem statement. The UAV will be using cameras placed in different directions and the live video feed will be relayed to an Off-board station over Wi-Fi link where all the image processing techniques have been implemented (Canny Edge Detection and Colour Detection). The Canny Edge detection algorithm is used for localization within the grid based arena and the Colour detection is used for detecting and tracking the ground robots. The On-board computer will be performing the basic operations such as flight control, position/drift control, coordination of all the on-board sensors like the Px4Flow, LIDAR etc. The Px4Flow is used for drift control and position hold while the scan data from the LIDAR is used for obstacle avoidance. Both the on-board and off-board computers will be running on the Robot Operating System (ROS). The image processing techniques and overall system architecture is discussed in detail in this paper.

Overall System Architecture

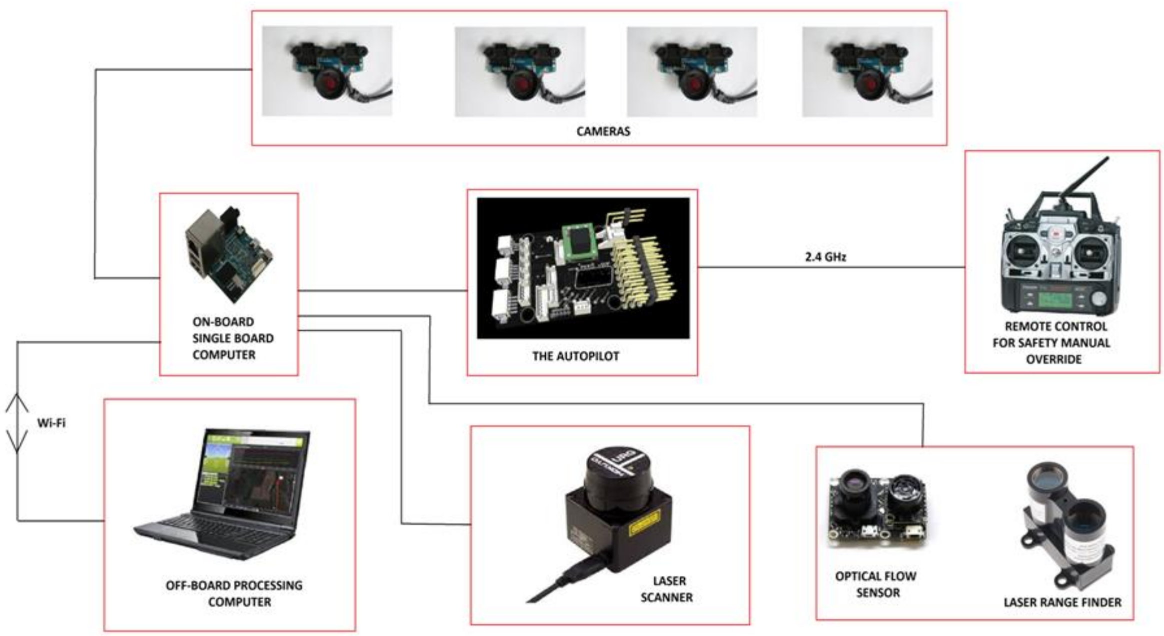


Figure 1. Overall system Architecture

Yearly Milestones

The team has been active in this competition since 2011 version. The team secured first place in the 2011 competition at the American venue. The preparation for this particular mission began in June, 2014. Since then the team has worked to develop and modify the old UAV to suit this mission. The team was divided in to small groups which will handle the various sections such as target Identification, computer vision, flight control loop, sensors, flight electronics and structures. The team has acquired a rapid prototyping machine, which has been involved in designing various custom parts for the UAV. The team has modelled the entire UAV including the sensors, motors, propellers etc., which will be used for simulation and software analysis purposes.

AIR VEHICLE

Figure 2. Modified Pelican Quadrotor

The Air vehicle is built on the Ascending Technologies' Pelican frame, which was chosen due to its light weight, payload capabilities, flexibility to add more base plates and overall structural integrity. All the other native parts from the Pelican platform which were used in the previous mission, such as the Intel Mastermind, Asctec motors, ESC etc., have been replaced with new components. The flight controller used is the Pixhawk autopilot which is an independent, open source, open hardware project. It employs 168MHzCortex M4F CPU and runs a very efficient real-time operating system (RTOS). The Odroid U2 is the on-board computer running ROS (Robot Operating System) on Linux (UBUNTU) which is the brain of the vehicle, that co-ordinates all the sensors and cameras. The flight electronics are powered from the 12V Lithium Polymer battery. Since, the voltage is too high, a voltage regulator module which provides 5V constant voltage is used.

Propulsion and Lift system

For this mission, the Quadrotor platform has been chosen. The Quadrotor is a Vertical Takeoff and Landing (VTOL) rotorcraft which is propelled by four rotors. Unlike a helicopter, a Quadrotor does not require a separate propeller to negate the angular momentum generated by the movement of the driving propeller/s; this is due to the fact that, in a Quadrotor, a set of two propellers is made to rotate clockwise while the other set is made to rotate anticlockwise. Hence the net angular momentum about the center is zero.

The Quadrotor is chosen for its stability and for the ease of availability of resources for the platform. The entire Quadrotor weighs around 1000 grams. It is propelled by four brushless actuators (880Kv AC2836-358) whose speed is controlled to perform different flight maneuvers. The motor's speed is controlled using normal 20A (rated) All-in-one Electronic speed controller (Qbrain 4in1). The actuators and the ESCs are powered by a 6000mAh 3-Cell Lithium Polymer battery.

Guidance, Navigation and Control

Stability Augmentation System

The quad copters by nature are an unstable system; hence they need external controllers in order for them to fly in the desired manner i.e. at set attitude and attitude. PID based controllers of different levels have been used to achieve stabilized flight. First level: specifying the motor speeds directly in terms of rpm; Second level: setting the desired attitude (roll, pitch & yaw) & Third level: position control by specifying the target position. The 3rd level of control is chosen along with specifying the linear velocities. As the position controller application is built into the PIXHAWK it greatly simplifies the task, as we are no longer have to design the position controller and also don't have to worry how the UAV achieves the desired position [1].

Navigation

The sterile nature of the arena i.e. lack of GPS and any physical landmarks or stationary points of reference completely rule out the use of LIDAR based SLAM or simple GPS. All the elements that can be used for localization and navigation are lying flat on the floor, thus vision based navigation is the obvious choice.

The vehicle is equipped with three cameras- The PX4FLOW and two Logitech HD webcams.
The PX4FLOW is used for the optical flow calculations. This is our main component for Odometry. The Logitech webcams used are the "Logitech HD Webcam C310". The reason for selecting these cameras is primarily sourced upon their value, with the specifications and the ability to handle light conditions and some forms of noise independently [2-3].

Now the cameras have been integrated into the UAV in such a way that one of them is fitted along the x-axis of the vehicle and the other along the y-axis of the vehicle. This will help us to receive information from each of our assumed axes and therefore, help in the mobility of the UAV with respect to the environment.

Our approach to the problem statement includes vision based localization inside the grid structured environment. Since image processing requires a lot of processing power, hence it is done off-board on a ground-station. The ground station runs with the help of OpenCV 3.0 (Open-Source Computer Vision) and ROS (Robot Operating System) running in Ubuntu(Linux). The live video feed from the on-board cameras is relayed onto the ground-station which is connected via Wi-Fi to the ODROID.

Initially we consider the position of the drone to be on the borderline of the arena. This gives us an initial starting point for the mission. Now based on the position, the drone has been fed hard coded boundaries for the arena. This is a good option as the arena is static and it eliminates substantial amount of on-board processing. Now after the boundaries have been set, the drone can begin its task, i.e., identifying the targets, the obstacles, and acting with necessity. For this we have implemented a few algorithms.

CANNY EDGE DETECTION

The “Canny Edge Detection” algorithm is implemented on the live video to extract the edges. Now there are x and y-axis parameters to keep track of the current position of the UAV in the grid. The camera fitted along the x-axis detects any horizontal line and hence the x-coordinate counter of the vehicle is updated as it passes a horizontal line. Similarly, the y-axis camera detects vertical lines and y-coordinate is updated as it passes a vertical line. Hence, this provides a 2-d coordinate system for the vehicle to be localized inside the grid-based environment [4].

The OpenCV function implemented here is:

- ***cv2.Canny(inp,minVal,maxVal)***

Where:

inp-represents the input image given to the program.
minVal-a threshold, showing that any value below this considered to be a non edge.
maxVal- represents a limit above which any value is considered to be a sure edge.

Now within function, there naturally exists the possibility of edges falling in between the minimum and maximum limits. In that case the algorithm detects lines that are intersecting with, or combined with the lines considered as sure edges. These lines will then be considered as a part of the final mapping.

Also, this algorithm is considered a lot more efficient and relevant, as its predictions are based on the intersections as well. It also has a Gaussian Filter inclusive, as the edges are easily prone to noise, internal or external.

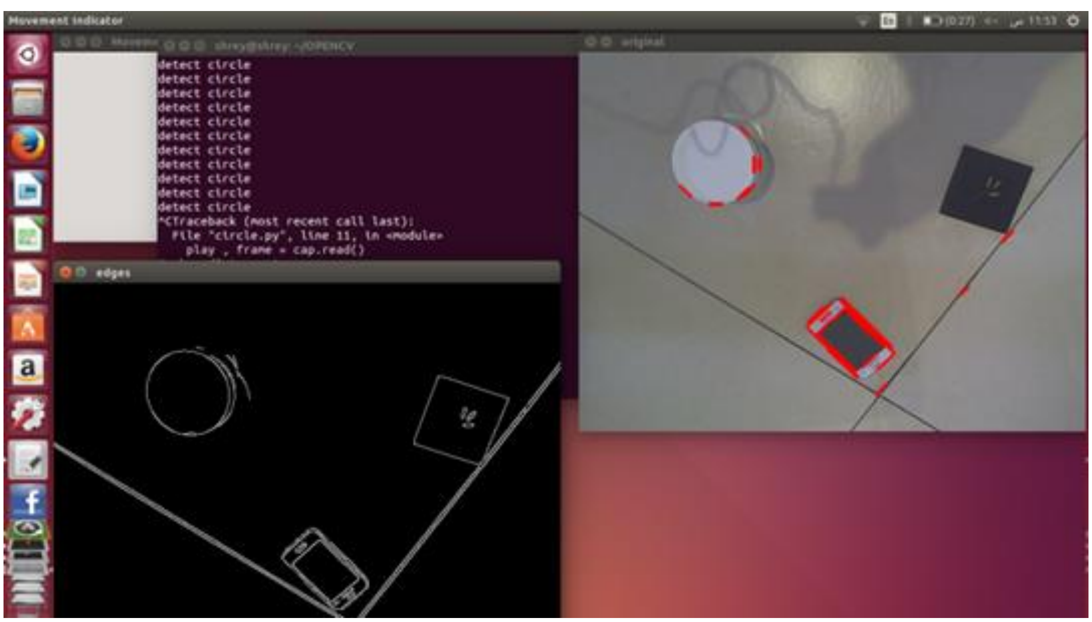


Figure 3. Image depicting edge detection of a grid.

This has helped to chart out the area accordingly, in our case the internal arena grid.
We have further implemented a color-based tracking for the ground robots. The iRobots are either plain or marked either red or green. The objective for us is to move the ground robots to their boundaries with the color corresponding to them. For this reason, only the green and red colors have been implemented in our algorithm.

COLOUR DETECTION (COLOUR SPACES)
As mentioned above, the algorithm enables us to have an identification scheme on the ground robots based on their colour.
Here we have implemented 2 of the OpenCV functions. The first function is to convert the input frame into an HSV scale.
cv2.cvtColor(frame_input, format)
Where,

frame_input- represents the input taken but the drone.
format- this represents the format in which we want the output to show.

The other function implemented is to select the range of the colour which can be considered true or valid if a drone detects a ground robot in the field of perception -
cv2.inRange(format,upperVal, lowerVal)
Where,

format-the same variable that holds the output format.
upperVal-the upper limit to the selected colour's range.
lowerVal- the lower limit to the selected colour's range.

Now initially the input is taken by the drone. This input then undergoes conversion to a HSV (Hue, Saturation and Value) scale. This is done because the representing and splitting or isolating colours is a lot easier on this scale, although the cost of implementing this is a minor increase in the processing carried out. After the input is taken, and the respective upper and lower colour limits have been coded in, the drone then gains an output visual of just the colours regarding the ground robots.
The colour ranges we have implemented here are [0,100,100] to [10,255,255] for "Red" and [50,100,100] to [60,255,255] for "Green". As mentioned above, these colour ranges are represented on the HSV scale.
This gives us the amenity of identifying the objects or targets to then carry out further tasks in the mission.

Flight Termination System
Our system is equipped with a human- operated kill switch disables all power to the motors in scenarios where a quad rotor enters into an unresponsive and dangerous state. A safety kill switch has been integrated between the power source and the power distribution board. A channel in the 7CH Remote control has been allocated for manual override over Radio Frequency for emergency safety purposes.

PAYLOAD

Sensor Suite

The UAV is equipped with the following sensors:

- Hokuyo URG-04LG-UG01 Laser Rangefinder, that has a range of up to 4m and 270° field of view
- Two Logitech HD Webcam C310, that are used for providing the live video feed for vision based navigation.
- One Px4Flow, which used for optical flow calculations
- Odroid U2 which is the main on-board processor and the PIXHAWK auto-pilot

Threat Avoidance

The Quadrotor detects and avoids threats by analysing feedback from the Hokuyo laser range finder. The laser provides information on where any obstacles may be within a 2 m bubble around the vehicle [5]. Paths that require the vehicle to go near any obstacles will have a higher cost than paths that do not, and paths that intersect with any obstacles will have an "infinite" cost.

Communications

The vehicle communicates with a base computer via mavros protocol in order to deliver telemetry data. The other communication links include a Wi-Fi link to deliver real time video to the base station. Both of these links operate at 2.4 GHz. Finally, a safety pilot can take control over the vehicle at any time using a Futaba radio controller operating at 72 MHz.

Power Management System

The energy source of autonomous aerial vehicles is a critical aspect of operations. Quadrotors most essentially depend on lithium polymer batteries as their major souce of power due to their high energy density instead of the conventional sources used by other RC vehicles. A brushless motor's rotation speed is related to its kV rating and the input voltage it is important to choose a battery allowing for a good capacity to weight ratio while providing an appropriate voltage. To add, a higher voltage creates a higher rpm but a larger propeller requires a lower rpm. Looking at all aspects and testing, 11.1V Lithium Polymer Battery was considered most suitable. A power distribution board is used to distribute power and communication lines to all motor controllers and other systems on board. This provides the Pixhawk with 5 volts and allows for battery voltage monitoring.

OPERATIONS

Flight preparation

The flight test is carried out with utmost precaution following the appropriate safety procedure to prevent any technical mistakes from taking place. The functioning of the switch activating the emergency manual override is also checked.

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 LEDs on the Pixhawk, power module and all the sensors.

Man/Machine interface

The vehicle communicates over the Wi-Fi link to the control station in order to relay real time video feeds from the onboard cameras, status of the vehicle, commands sent by the onboard computer to the autopilot and the telemetry data in real time. This forms the live HMI of the system which facilitates continuous monitoring of the vehicle.

RISK REDUCTION

Vehicle Status

Status LEDs are located on the Autopilot, Odroid and Telemetry set. The different status LEDs and its meaning are shown in the figure below. The status LEDs help confirm the working of all the sensors inside the autopilot.

Flashing red and blue: INITIALIZING

Double flashing yellow: ERROR. Sensor Calibration needed.

Solid green plus single long tone: ARMED: Ready to Takeoff

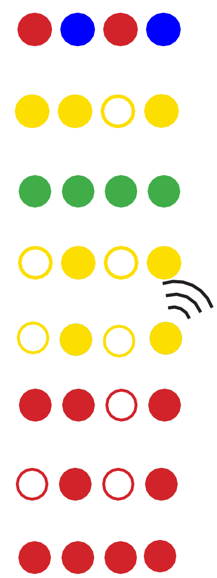
Flashing yellow: Remote Control failsafe activated.

Flashing yellow plus quick repeating tone: Battery failsafe activated.

Quick, constant blinking: performing system check. Please wait.

Intermittent blinking: system ready. Press the safety button to activate.

Solid: ready to arm. Proceed to the arming procedure.



In case of sensor failure in the autopilot, the exact name of the sensor shows up on the screen of the control station via the telemetry link. This can be corrected by recalibrating the sensor. Other flight electronics such as the Odroid, Telemetry and the sensors have a status LED on them that indicates if it is functioning.

In addition to this, the ground station receives the status of the Quadrotors's current orientation and also the flight commands that are being executed.

Vibration Isolation

The Autopilot has the IMU unit which is prone to vibrations. It can give false readings which can lead to a bad flight. Since the orientation of the UAV is essential information that needs to be as precise as possible, the IMU needs to be secured properly. Vibration needs to be less than 0.3 G in the X and Y axes and less than 0.5 G in the Z axis.

3D printed Anti-Vibration Platform for the IMU
The vibration damper is shown in the figure on the right. It incorporates 4 rubber dampers under the autopilot platform. The asctec pelican's base plate's model was modified to accommodate the anti-vibration damping mechanism and then 3D printed to be incorporated into the main frame.

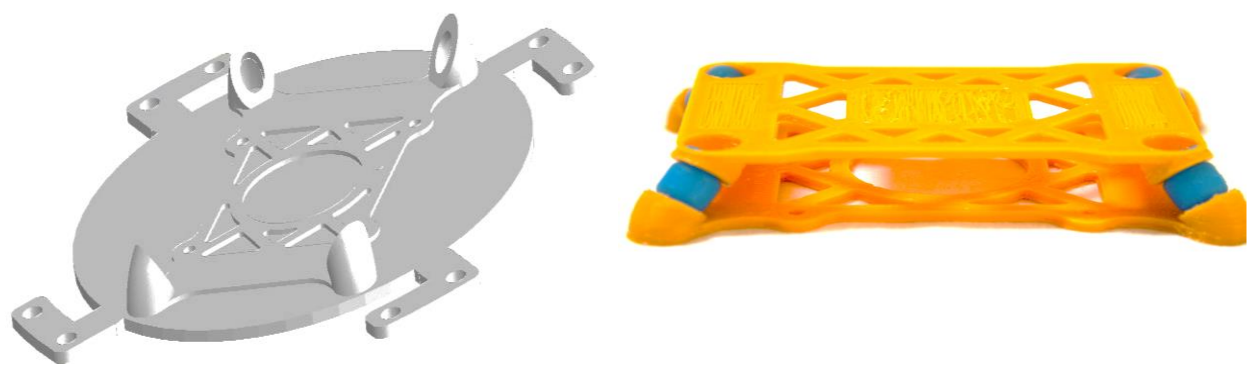


Figure 4. Modified base plate

3D printed modified base plates
The original Asctec base plates have hole provisions for the native set of electronics. The design was modified such that the autopilot, Odroid can be bolted safely with appropriate plastic damping columns in between. Two support columns were added to improve the strength of the base plates and at the same time keeping it light weight. The figure below shows the base plate design.

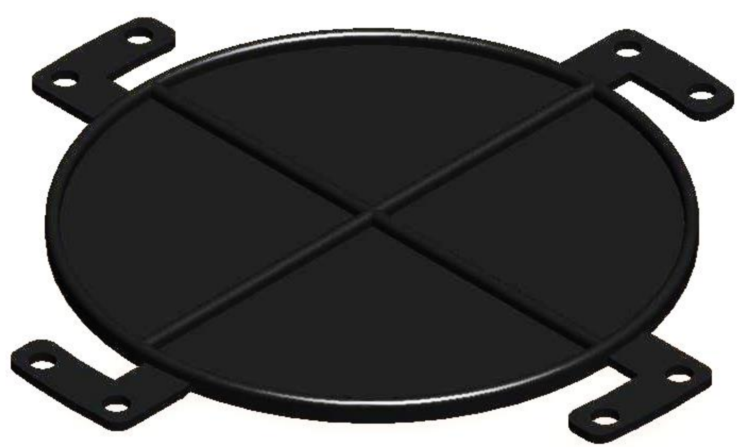


Figure 5. Modified base plate.

EMI Solution
The magnetometer is very sensitive to EMI, hence it is mounted above all the sources of EMI such as the motors and ESC. Other sensors prone to EMI have been positioned appropriately. The setup of all the components is shown in the figure below.

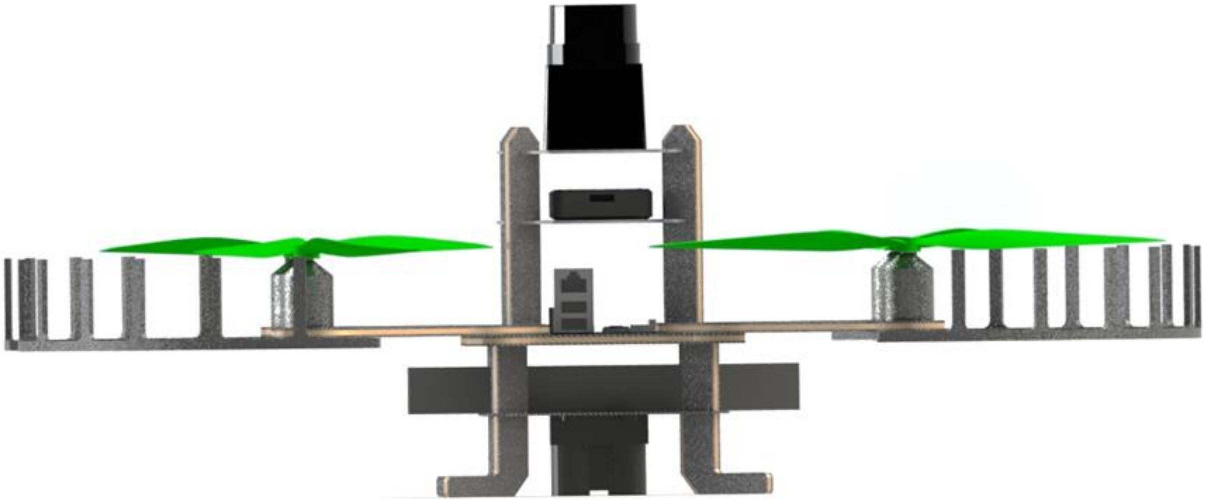


Figure 6. Side view showing all the components position.

Safety
Some of the safety features, that the IFOR is equipped with, include the custom 3D printed Propeller guards, emergency safety manual override channel and the manual kill switch.

3D printed propeller guards
The propellers are the only moving part of the UAV that needs to be secured from the environment. We have designed lightweight propeller guards such that it covers the entire side and at the same time provide very little restriction to air movement downwards. These guards can be 3D printed at any time if it gets damaged during the testing phase.

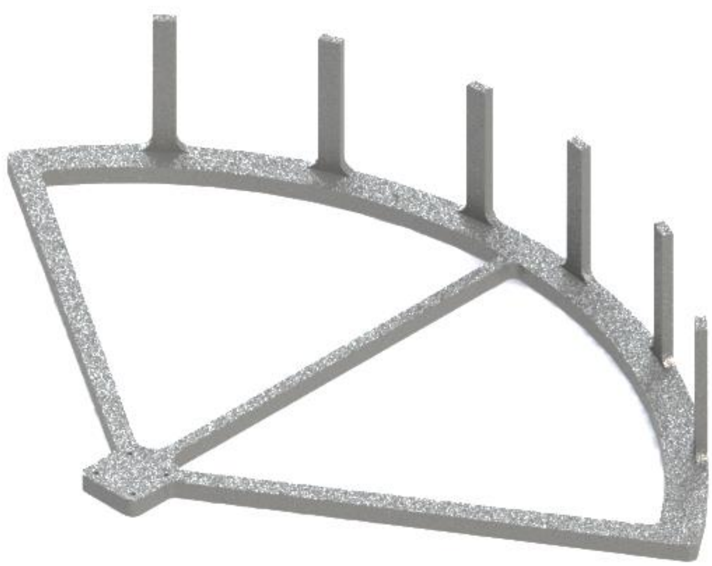


Figure 7. Propeller guard.

Modelling and Simulation

Figure 7. A CAD model of the Quadrotor

An overall CAD model of the entire UAV was created for simulation and analysis purposes. The model was constructed as close as possible to the actual model. It even includes all the electronics on board the UAV. All the rotating components have been given the appropriate Degrees of freedom. For example, the propeller is fixed to the actuator and rotates about its central axis as it does in the actual model. This model is being used for simulation purposes as it is made according to actual dimensions.

Testing

PROCESS/COMPONENT	TESTING
Basic Remote based control of UAV	*Tested and Working*
Communication between the Ground Control Station to Onboard computer	*Tested and Working*
Offboard Take-off and Landing	*Tested and working*
Waypoint input	*Tested and working*
HOKUYO Laser Scanner	*Tested and Working*
Ground Robot Detection	*Tested*
Grid based Localization	*Tested*

TABLE 1. PROCESSES AND COMPONENTS TESTED/WORKING

CONCLUSION

In this paper, we present the details of Team IFOR's Quadrotor that is capable of autonomously navigation through unknown environments where it can avoid dynamic obstacles using laser range finder and also interact with autonomous ground robots. Four cameras are mounted on the vehicle to detect targets for interaction later. The vehicle can self-localise using optical flow on downward facing cameras and stabilise itself using PID controllers.

REFERENCES

[1] A. Tayebi and S. McGilvray, "Attitude stabilization of a vtol quadrotor aircraft," Control Systems Technology, IEEE Transactions on, vol. 14, no. 3, pp. 562-571, may 2006

[2] Lorenz Meier et al., "A System for Autonomous Flight using Onboard Computer Vision", Proc. of The IEEE International Conference on Robotics and Automation (ICRA), 2011.

[3] M Achtelik et al., Stereo vision and laser Odometry for autonomous helicopters in gps-denied indoor environments. Proceedings of the SPIE Unmanned Systems Technology XI, 7332, 733219, Jan 2009.

[4] Haghi, A.; Sheikh, U.U.; Marsono, M.N. "A Hardware/Software Co-design Architecture of Canny Edge Detection", *Computational Intelligence, Modelling and Simulation (CIMSiM), 2012 Fourth International Conference on*, On page(s): 214 – 219

[5] J. E. Gomez-Balderas G. Flores L. R. Garcia Carrillo R. Lozano, Tracking a Ground Moving Target with a Quadrotor Using Switching Control, J Intel Robot Syst (2013) 70:65-78, DOI 10. 1007/s10846-012-9747-9.