e-Yantra Robotics Competition - 2018

Theme and Implementation Analysis – Pollinator Bee 5492

Team leader name	Abhay Sheel Anand
College	Bharati Vidyapeeth's College of Engineering
Email	abhaysheelanand@gmail.com
Date	3rd January 2019

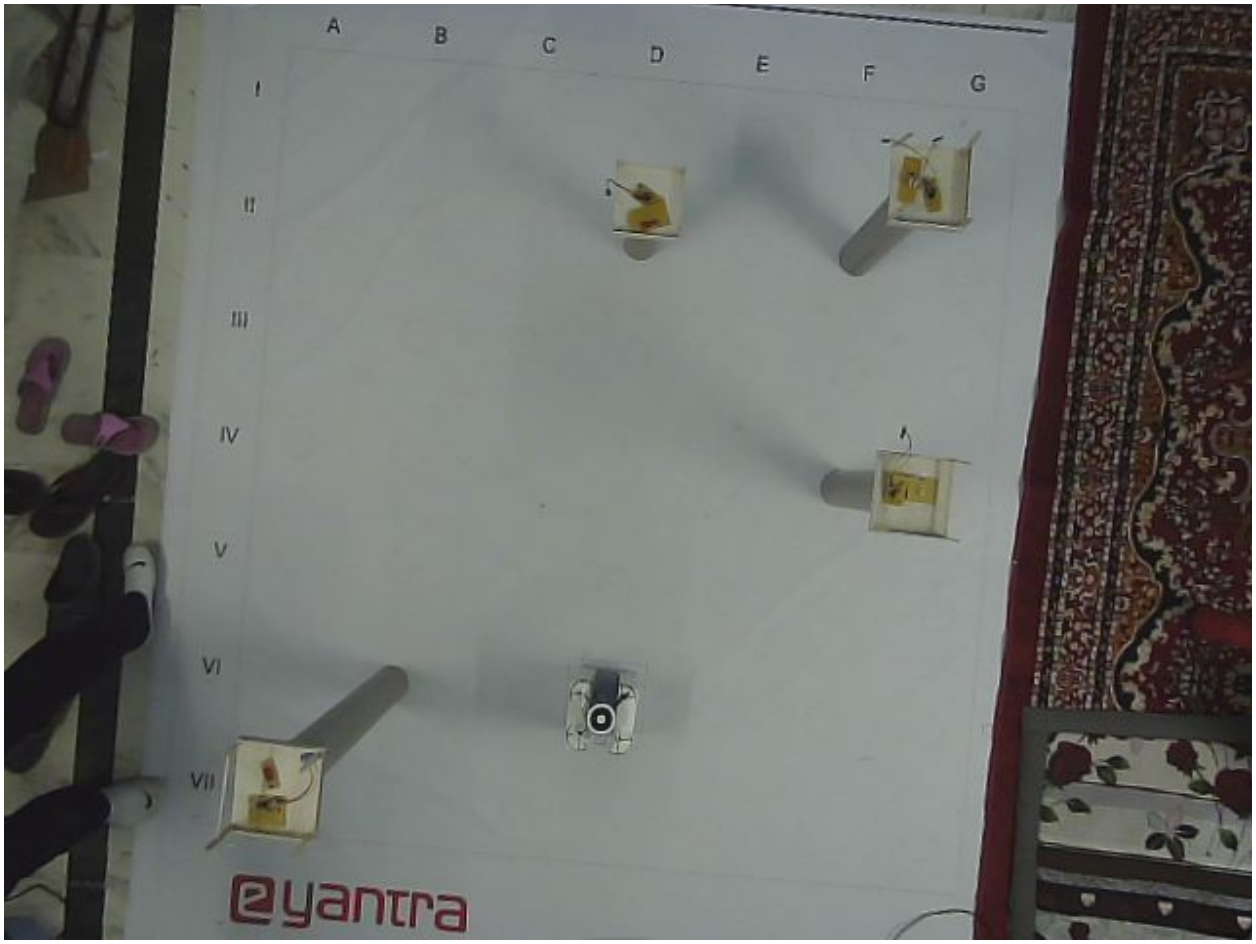
Scope and Preparing the Arena

Q1 a. State the scope of the theme assigned to you. (5)

The scope of the theme is widely varying and can be used in intervention of human activities to improve the efficiencies of processes, reducing the risk and making robust working systems. Autonomous Drones can help in fighting forest fires when it is very dangerous for firefighters, this can be done using a coordinate system, another application can be of delivering packages at different places. By using waypoint navigation we can also work on improving the efficiency of agricultural processes like spraying pesticides.

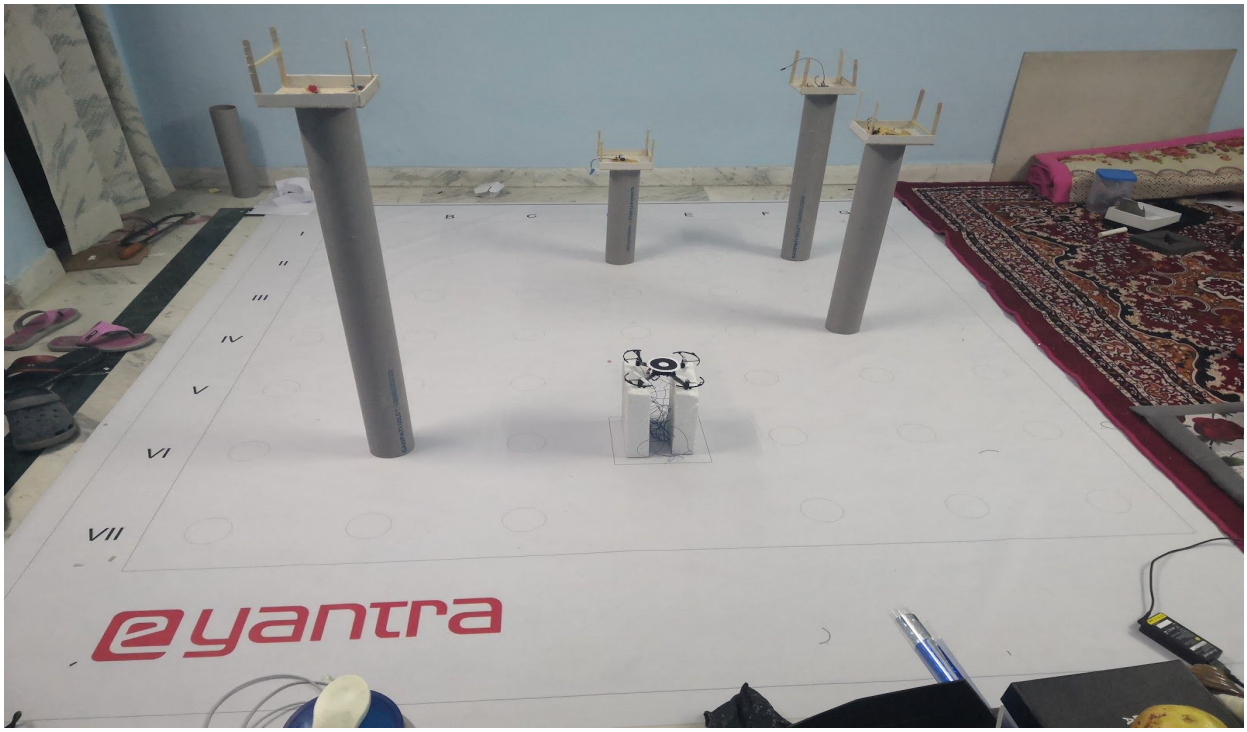
b. **Attach the Final Arena Images.** **(10)**

Top View




B
C
E
F
II
III
IV
V
VI
VII
eyantra


A B C E F
II
III
IV
V
VI
VII
eyantra


B C E F
II
III
IV
V
VI
VII
eyantra

Testing your knowledge (theme analysis and rulebook-related)

Q2. How will you ensure that while tuning the PID value, Drone will not crash? (5)

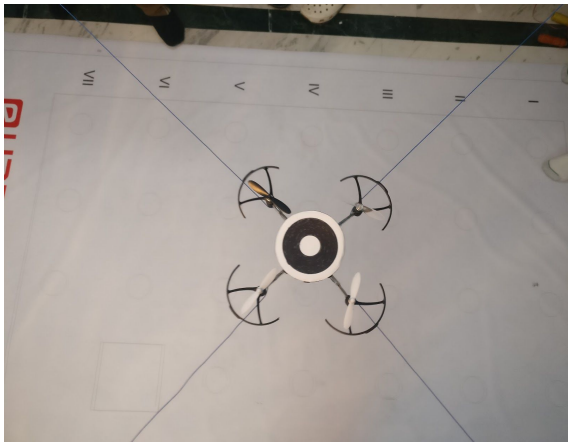
Safety of the drone is very important so we applied a few "jugaads" to ensure that our drone doesn't crash very often, hence protecting the drone from crashing to pieces

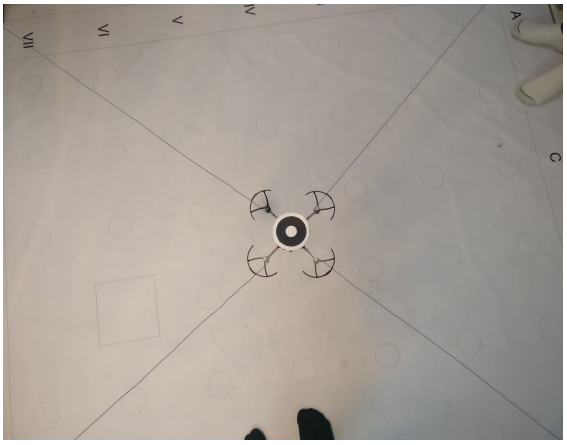
First of the "jugaads" we used is that we attached a string at the bottom of the drone and steered the drone away from walls and other obstacles by holding the string, hence preventing the drone from running amok. To make sure that our interference with the drone was not hindering the natural motion of the drone, we made sure that the string was slack when the drone wasn't under any immediate threat of crashing. We also used the string to bring the drone back in the frame of the camera so that WhyCon markers can be detected and we can tune the PID. Another thing we made sure while applying this method is that we kept the string below the level of propellers to prevent the string from getting tangled in motors and propellers from breaking.

The following pictures clearly depict the aforementioned jugaad:

The second "jugaad" that we applied was to attach a thin inextensible string under each motor, and extending the string to and holding them to a certain height while extending the string to prevent the drone from crashing. By using four strings and making sure that the string isn't completely stretched we can prevent the drone from crashing and bound its movement to a specific region that prevents the drone from going out of the camera frame, while preserving the infinite degrees of freedom of the drone.

The following two pictures clearly depict the aforementioned jugaad:





Finally, we added the disarm functionality to disarm the drone when it is mid air on the press of a key. We achieved this using "key_command.py" and "key_handling.py" python scripts in the scripts folder of "plutoserver".

We made a subscriber that subscribes '/input_key' and made a function that matches the input key with the key we have chosen to disarm our drone, if the keys match the drone is disarmed.

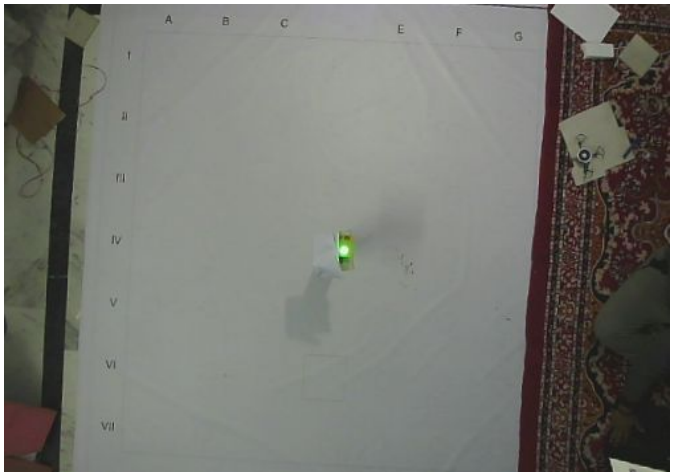
The Subscriber

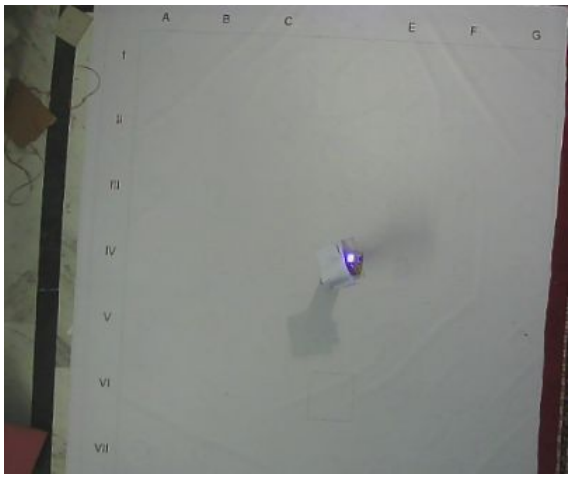
```
rospy.Subscriber('/input_key',Int16, self.our_disarm)
```

The Function

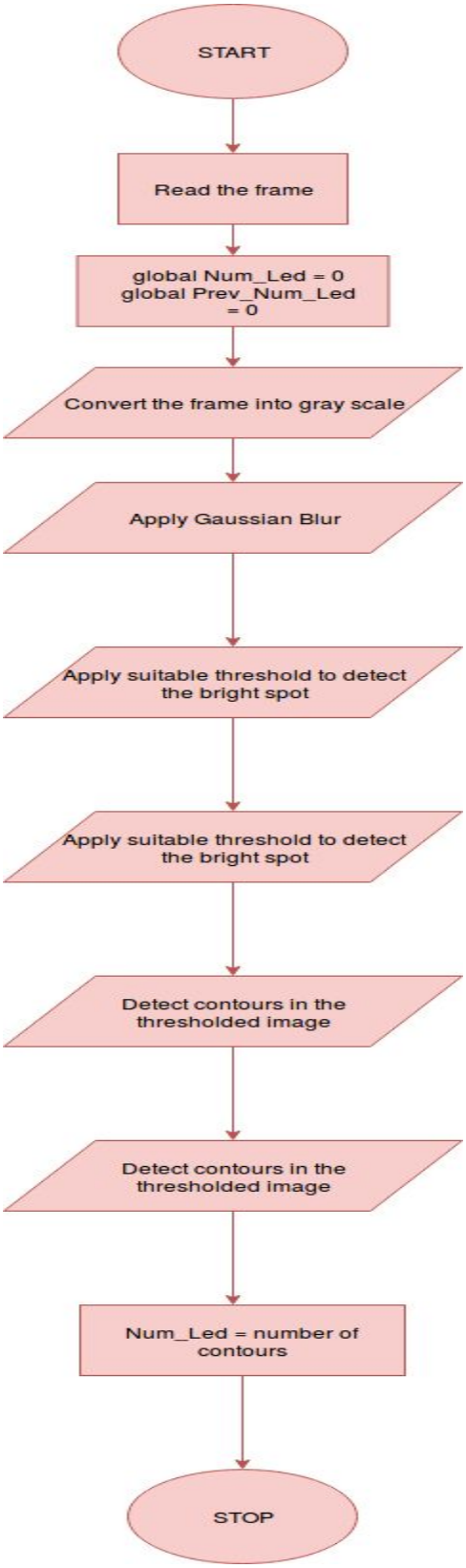
```
def our_disarm(self,inpv):
    if (inpv==5):
        self.disarm()
        print "drone disarmed"
```

Q3. How will you detect the LEDs lighting up using image processing? You may use your pseudo code to explain your approach. (5)





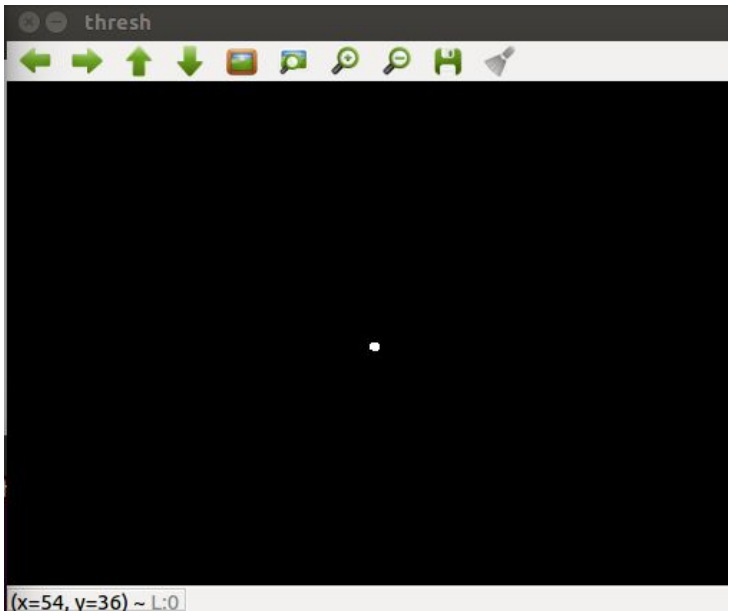
To detect the LEDs lighting up, the bright spots in the frame are detected. This is compared with the previous number of bright spots. If previously detected number of blind spots is less that indicates that LED is switched on. For detection of bright spots, the following algorithm is used:-



Bright-spot detection

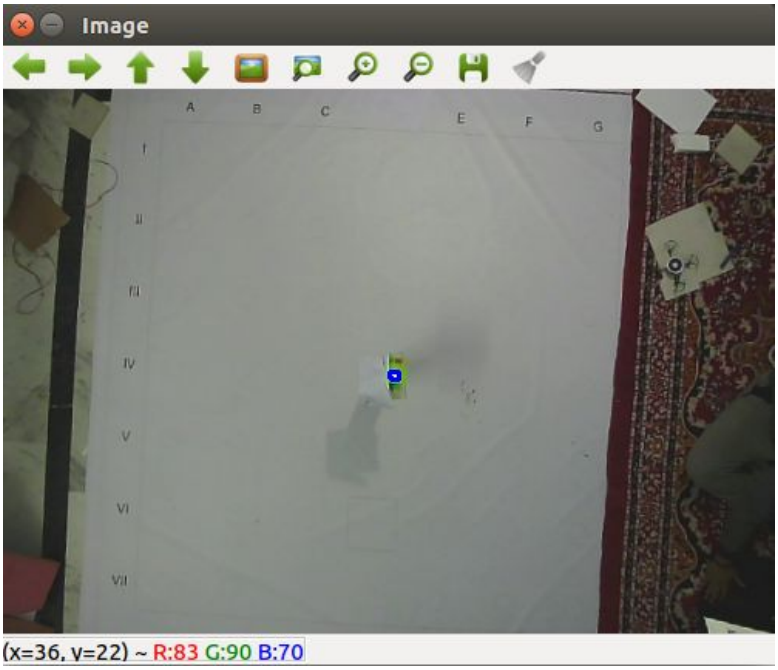
Gaussian blur reduces the noise and details, thus making the detection facile. After applying gaussian blur, the following output is obtained.

After gaussian blur, the bright spot is detected by applying a suitable threshold in the image. After thresholding the following output is obtained.



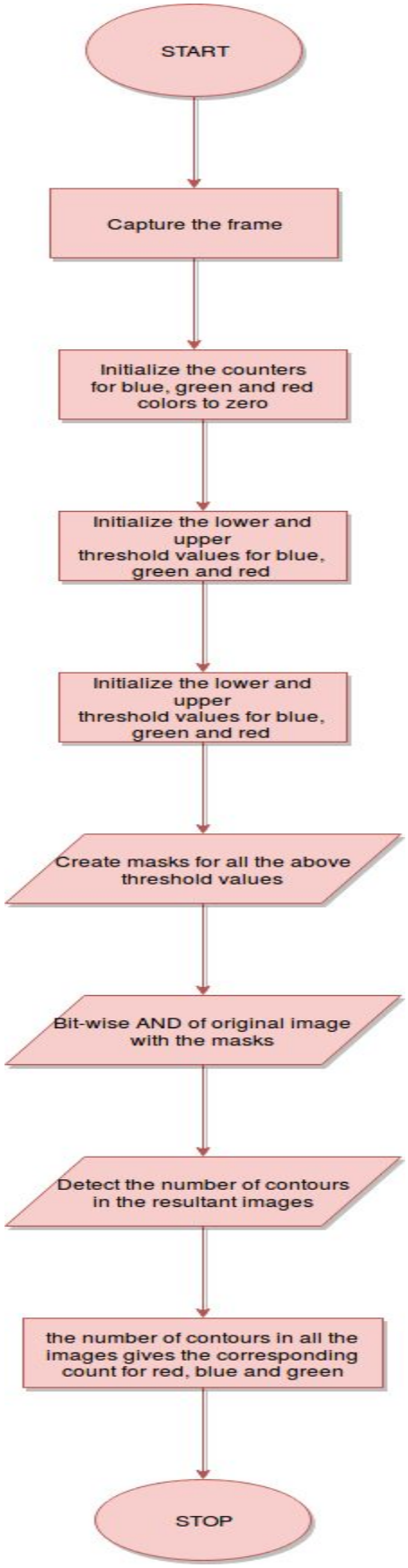
Now, the contours are detected in this image. These contours will indicate the number of bright

spot in the image. The detected contour is drawn with blue color as shown below.



The status of the LED is not detected continuously as it will consume a lot of processing power. It is detected only when the drone has visited a particular waypoint. After the drone has visited a particular waypoint, the bright spot detection code is executed. This code compares the result with the previously detected bright spots to check whether the LED has light up or not. In case the LED has not light up, the drone will traceback its path to visit that waypoint again.

After detecting that the LED is switched on, the color of LED is detected. To detect the color of the LED, the image is masked with the suitable threshold mask. Then, the For detection of the color of LED, the following algorithm is used:-


START
Capture the frame
Initialize the counters for blue, green and red colors to zero
Initialize the lower and upper threshold values for blue, green and red
Initialize the lower and upper threshold values for blue, green and red
Create masks for all the above threshold values
Bit-wise AND of original image with the masks
Detect the number of contours in the resultant images
the number of contours in all the images gives the corresponding count for red, blue and green
STOP

Q5. Let us consider a scenario: **(5)**

The Pollinator Bee has reached a desired waypoint, but the LEDs at the waypoint have not lit up.

What will happen according to your algorithm (Consider the theme rules specified in the rulebook)?

When a drone reach a waypoint and if light does not lit up then we would traceback to that same path to reach that same waypoint . This method has two advantages one is that battery is reserved and other reason that total time taken for the completion of the task is reduced. We would use image processing to make sure that light is lit up, if not then a type of interrupt would be called and make drone to reach the last waypoint.

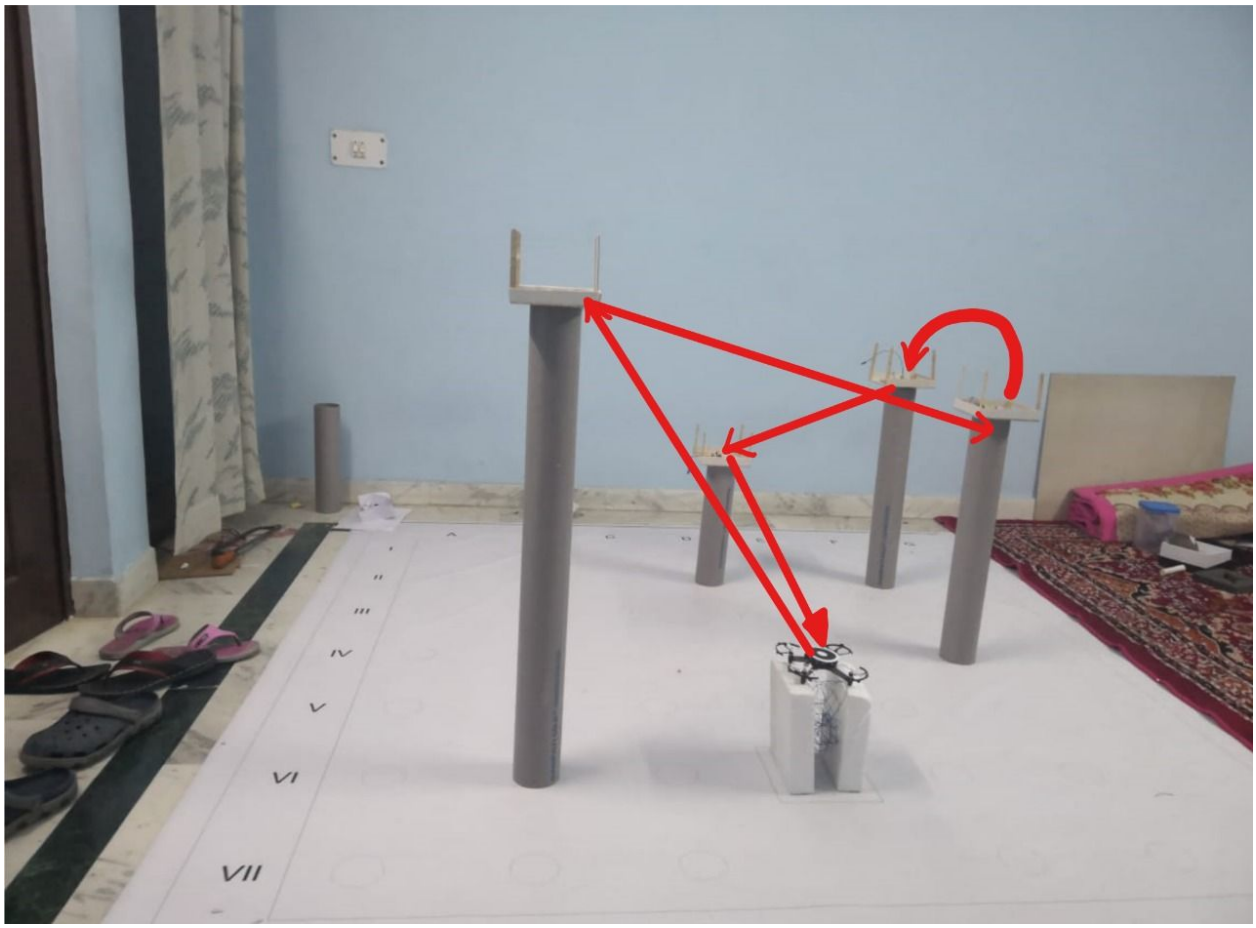
Q6. What will be your strategy to earn maximum points in a run? **(5)**

The strategy to earn maximum points will be to pollinate the flower in very less or no time. For this purpose, we have designed the flower with a mesh of wires and aluminium foil will be tied to the drone. Thus, when the drone will pass through a particular flower, the foil will short circuit the wires of the latch circuit, causing the LEDs to glow. This way we have highly increased the probability of pollination i.e. glowing of the LED. Since there are creativity bonus points for the most creative flower platform and bee stinger, these designs have been proposed. We have reduced the chances of pollination failure by the proposed mechanism as there are maximum marks for number of flowers pollinated. This way we have also reduced the time taken in the run.

As there are bonus marks for landing on the beehive, so we have planned that we will store the initial position of the drone and at last we will take drone to that position in the x-y coordinates and then we

will reduce the throttle slowly, so that it lands at the beehive.

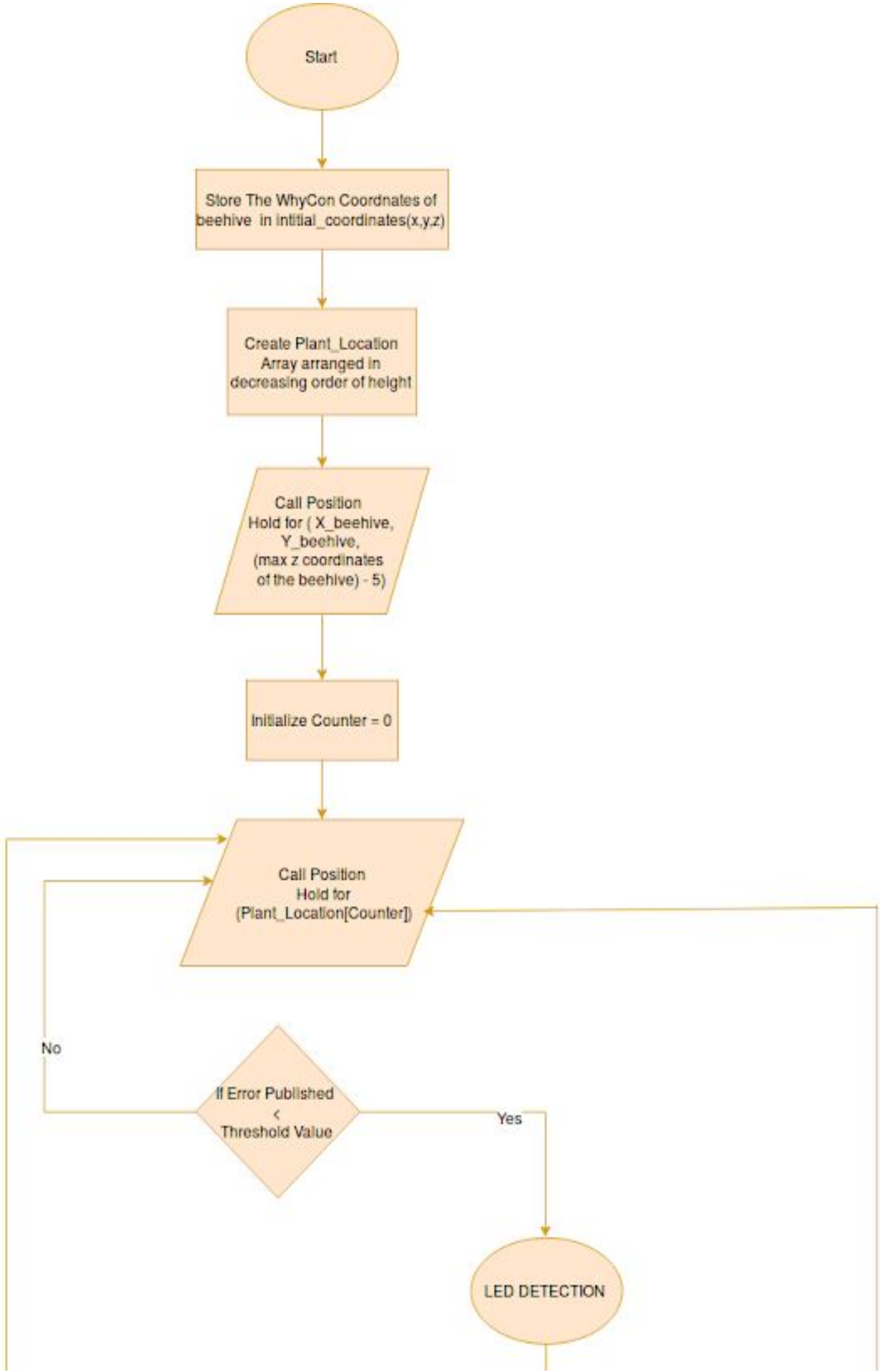
Another strategy is that we have chosen the order of traversal in an optimized way so that there is no need to give throttle again and again, thus saving the battery. The order of traversal will be in descending order.

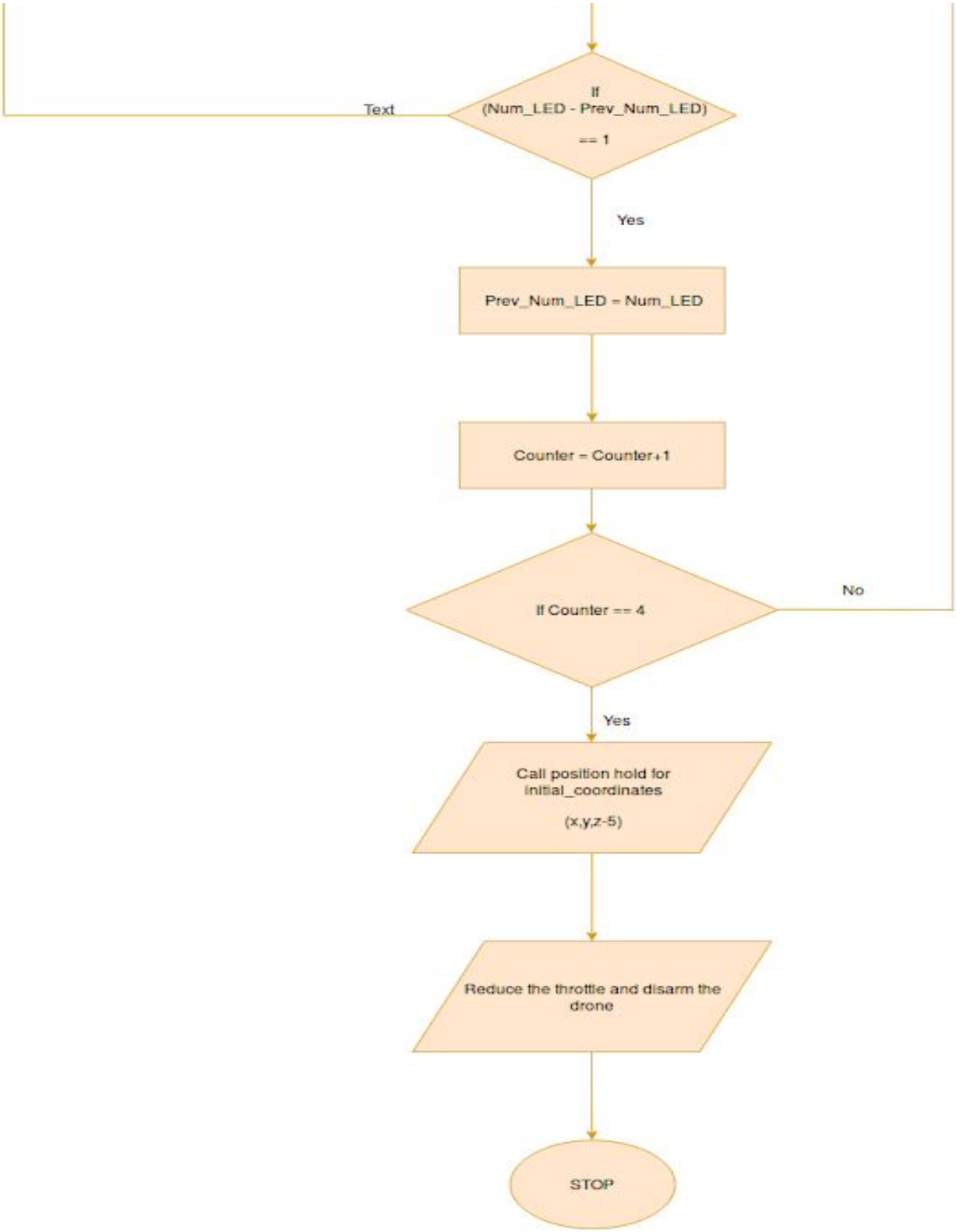


Algorithm Analysis

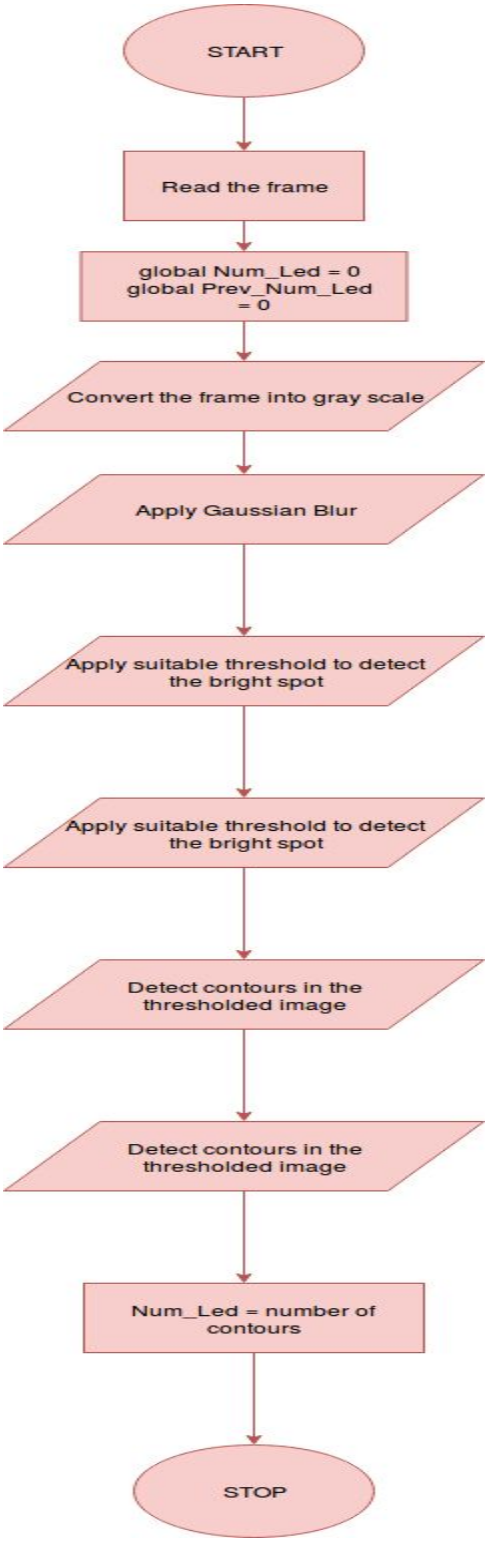
Q5. Draw a flowchart illustrating the algorithm you propose to use for theme implementation.
(10)

The flowchart of the proposed algorithm is shown below. The discontinuity is due to its large size. The flow chart is in the link FLOWCHART


Start
Store The WhyCon Coordnates of beehive in initial_coordinates(x,y,z)
Create Plant_Location Array arranged in decreasing order of height
Call Position Hold for (X_beehive, Y_beehive, (max z coordinates of the beehive) - 5)
Initialize Counter = 0
Call Position Hold for (Plant_Location[Counter])
If Error Published < Threshold Value
No
Yes
LED DETECTION


If
(Num_LED - Prev_Num_LED)
== 1
Text
Yes
Prev_Num_LED = Num_LED
Counter = Counter+1
If Counter == 4
No
Yes
Call position hold for
initial_coordinates
(x,y,z-5)
Reduce the throttle and disarm the
drone
STOP

In this flowchart, the LED detection function is:-



The position hold function is the function that uses PID to stabilize the drone at the particular coordinates.

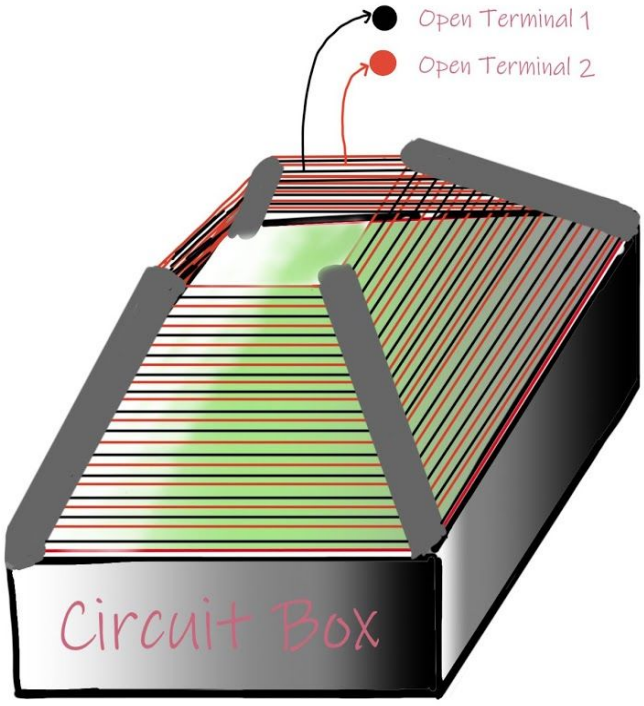
Challenges

Q15. What are the major challenges that you can anticipate in addressing this theme and how do you propose to tackle them? (5)

1. Challenge 1 - The major challenge in addressing this theme was PID tuning. Choosing the values of Kp, Ki and Kd by hit and trial method was a tedious method. This was a time-consuming task. To resolve this problem, we used auto-tuning and adaptive PID.

2. Challenge 2 - Another challenge was color detection of LED as the color of all the LEDs came out to be as white. To resolve this problem, we used array of LEDs passing their light through cellophane sheet. To reflect the light of LED, we covered the array of LEDs with aluminium foil..

3. Challenge 3 - Another challenge was pollinating the flower in very less time. To tackle this challenge, we designed the flower in such a way so that the drone just has to pass above it and the flower will get pollinated. The design of the flower is shown in the figure below.



Design of the flower