e-Yantra Robotics Competition - 2018

Theme and Implementation Analysis – Pollinator Bee

5917

Name	Atharva Rajendra Karpate
College	Shri Ramdeobaba College of Engineering and Management
Email	karpatear@rknec.edu
Date	2nd January, 2019

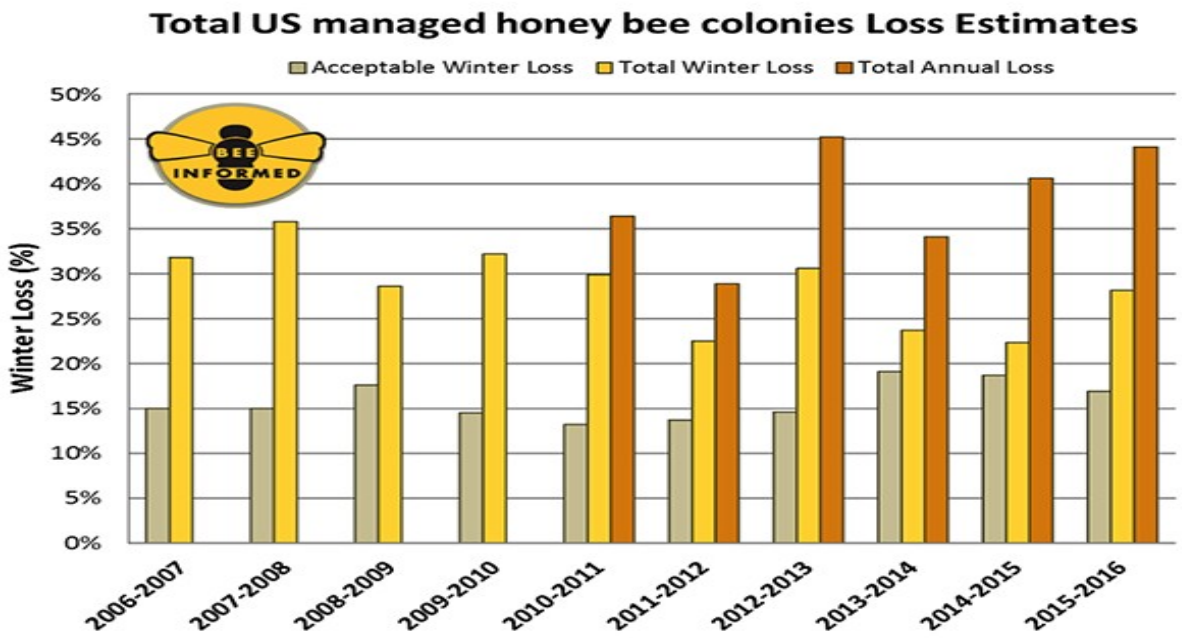
Scope and Preparing the Arena

Q1 a. State the scope of the theme assigned to you.

(5)

Pollinator-Bee is based on the 'Bio-mimicking' robotics. In this theme the behaviour of a bee is imitated by a drone performing pollination in a plant. The arena contains flowers that serve as platforms with an incomplete circuit and exposed wires. The drone with its bee stinger reaches the flower and touches it which completes the circuit and the LEDs glow, indicating successful pollination. This is similar to a bee, flying from its beehive to different flowers performing pollination.

The main purpose of such an application is to compensate the declining rate of pollination. Due to the drastic drop in bee population.



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


B
C
D
E
F
V
VI
VII


D
E
F
G

Testing your knowledge (theme analysis and rulebook-related)

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

In order to prevent the drone from any probable crashes, we plan on implementing 3 techniques.

1) Pseudo code.

(Here, xmax & xmin is the range of x ordinate of the arena.
ymax & ymin is the range of y ordinate of the arena.)

```
if((x>xmax || x<xmin) || (y>ymax || y<ymin))
        land()
        disarm()
```

2) Thread tied to the drone.

A spotter is used to prevent the drone from the crash by using a thread tied to the bottom of the drone.

3) Using a matress.

The entire arena is set up on a mattress which will ultimately give minimum impact to the drone even when it crashes.

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

- The image is taken from 'whycon/image_out' and is converted to openCv

frame via ‘ros_bridge’.

- The openCv image frame is then blurred using Median Blur to eliminate noise and smoothen the edges.

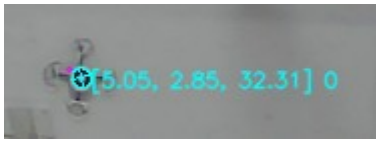
  blur=cv2.medianBlur(img,5)

- The desired color region is extracted through the image and stored in ‘mask’. ’lower’ and ‘upper’values specify the desired hsv range.

  mask = cv2.inRange(hsv, lower, upper)

- Then the image is eroded in order to remove white noise and also to stop the detection of blue whycon coordinates in the image from the whycon marker placed over drone.

  erode = cv2.erode(self.mask,kernel,iterations = 2)



- Morophological transformation closing is applied in order to join diffrent segments detected from the same plant and create one contour per plant.
  dilation = cv2.dilate(erode, kernel, iterations = 13)
  closing = cv2.erode(dilation,kernel,iterations = 1)
- Contours are drawn around the detected color region. The centroid is found and a rectangle is drawn around this contour.
  abc,contours, hierarchy =
  cv2.findContours(self.closing,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)

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)?

Considering (x,y,z) are the co-ordinates of a flower to be pollinated.
And bsh=bee stinger height, which is a constant term.

Step 1 : Drone will reach the co-ordinates (x,y,z+5) i.e. a little above the desired way point.

Step 2 : Now, the height of drone is decreased slowly till it reaches the co-ordinates (x,y,z+bsh). Ideally, LEDs should glow at this point.

Step 3 : If the LEDs don't glow, drone will be made to move a little in x direction say (x+-0.5) (assuming the error due to which the LEDs didn't glow is in x direction).

Step 4 : If the LEDs still don't glow after step 3, drone is allowed to move a little in y direction (now assuming the error was in y direction).

Step 5 : If the LEDs still don't glow, the drone is further allowed to lower its height till (z+bsh-1). Here though the height is lowered so much, it is taken care that it doesn't damage the petals or bee stinger as the petals are supported by a sponge so lowering the stinger would just compress it for a while.

Step 6 : Till step 5 the LEDs must have lit up. Even if due to some error they have not, the drone will again follow the same steps from step 1 to 5.

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

(600 - T) + (TP * 500) + LB + (CBS * 10) + (CBF * 10) - (P * 30) + B

On analysing judging and scoring parameters, according to us, the best strategy to earn maximum points would be trying to get all the 4 flowers pollinated (getting all the LEDs to glow) as 1 flower yields 500 points which would help us achieving 2000 points. As in question 5, our team is planning to adapt the mentioned algorithm to make sure the LEDs glow, hence our main focus is to get them all pollinated.

The next point that we can work on is landing bonus. For smooth landing, we can plan to land the drone when it experiences least error.

```
if(xerr<0.5 && yerr<0.5)
        land()
        disarm()
```

The other strategy is to maximise the Creative bonus points. We plan on maximising

it by making the surface area of the petal maximum so that the probability of bee stinger getting in contact with the petal increases.

In addition to it we used aluminium foil around the LEDs so that it helps in image processing.

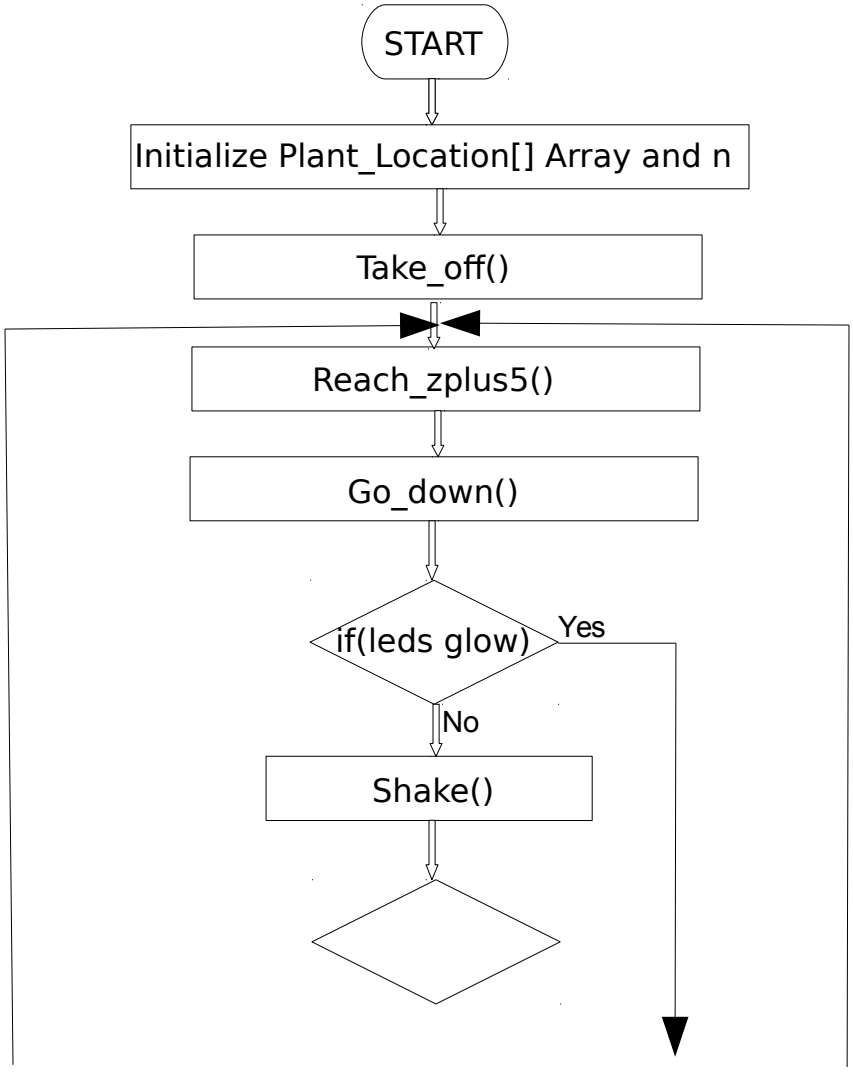
Furthermore we have made the petals with spongy foam so that it can bear the weight of the drone if there's any error in the z co-ordinate by compressing.

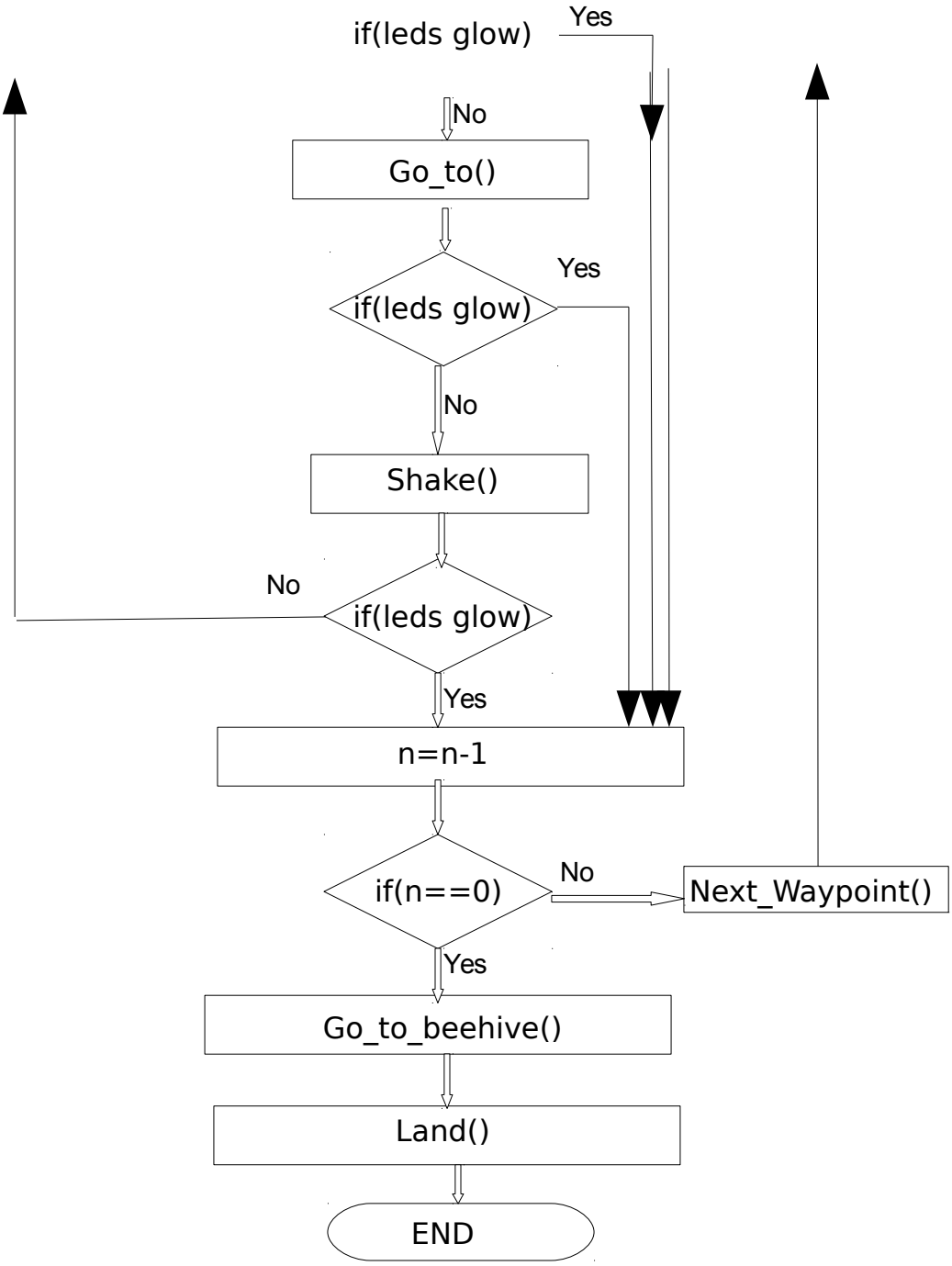
We have also strategized the sequence at which the plants will get pollinated. The sequence is in the descending order of the plant’s height to ensure that the drone does not collide with the stem of the flower. As it’s easier to come from more height to less than vice versa. This ultimately will take less time to complete the task as it prevents any probable crashes.

Algorithm Analysis

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

(10)





1. n : Here n is number of flowers. Initializing n means setting n=4.
2. Plant_location[] : This array stores information about the location of these flowers i.e. co-ordinates of these flowers. These co-ordinates are sorted in descending order of the heights of the plants.
3. Take_off() : This function arms the drone and increases its height till it becomes equal to the maximum height of a plant.
4. Reach_zplus5() : This function makes the drone attain the height equal to (z+5) given (x,y,z) are the co-ordinates of desired waypoint and hence reach at position (x,y,z+5).

5. Go_down() : This function slowly deacreases the height of drone to (z+bsh) where bsh=bee stinger height and hence drone acheives the position(x,y,z+bsh).
6. if(leds glow) : Here status of leds is checked whether they light up or not.
7. Shake() : This makes the drone change its position slightly in x and y directions (assuming error in those directions, hence as a correction factor).
8. Go_to() : If shake function doesn't work, this function is called which further lowers the height of drone to (z+bsh-1).
9. Next_Waypoint() : After successful pollination of one flower, next waypoint is given to the drone by using Plant_location[] array.
10. Go_to_beehive() : This functions makes drone return to the position (x,y,z+bsh) where (x,y,z) are co-ordinates of beehive.
11. Land() : This function lands drone to beehive when it experiences least error.

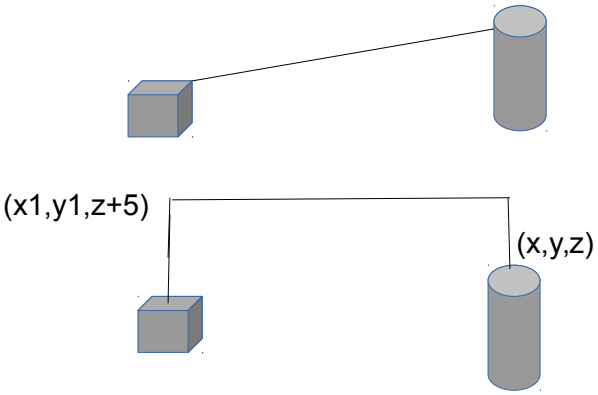
Challenges

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

1. Color detection in image processing.
Due to the closeness of LEDs, it appeared as 'White' colour when viewed through the camera. So to detect Red, Blue and Green colors effectively, we found the upper range and lower range of the HSV of all those colors. Then we performed erosion followed by dilation and then again erosion. Then the centroid of the cluster of LEDs was found and a contour was drawn around the detected LEDs.

2. Contact of bee stinger and flower's petal.
It is possible that the bee stinger may not come in contact with the petal due to some error in the whycon co-ordinates.If the error exists in the x&y co-ordinates so the drone moves in (+-0.5) distance in the x-y plane till the LEDs lit up. If the LEDs still do-not lit then the error may exist in 'z' co-ordinate, then the height of drone is decreased by 0.5 distance of the petal's height. The petal is made up of spongy foam to bear the pressure of drone's weight if this condition arises.

3. Drone crashes with the flower.
It is possible that the drone travels diagonally from its co-ordinates to the flower's co-ordinates and may hit the flower in order to get to the desired co-ordinates. So to avoid this scenerio, we coded to make the drone travel in the upward direction first starting from the bee-hive till (z+5) where z is z co-ordinate of the flower to be pollinated and then allow it to move in x-y plane to reach to its desired co-ordinates rather than allowing it to travel diagonally.



Here the cube shown is the beehive and the cylinder shown is the plant to be pollinated.

4. Collapse of the flower.
It is possible that when the drone hovers over the flower so it may come in contact with it inappropriately resulting in the collapse of the flower as the stem of the flower is made of a light weighted hollow PVC pipe. So in order to avoid that we have fixed the bottom of the stem with cellotape, plus we have filled the hollow PVC pipe with some material to make it more heavy comparatively which will make it less probable to fall.