2/24/2015

BALL CATCHING QUADCOPTER

MID EVALUATION OF FYP-1

Subhan Ali 11105016
Adnan Shafique 11105033
Zeeshan Afzal 11105003
Mamoona Sikander 11105014

Ball Catching Quadcopter

Introduction

Levitation is a fascinating phenomenon for human. Quadcopter research is practical example of this interest. Our project goal is catching a ball using quadcopter. The process involves:

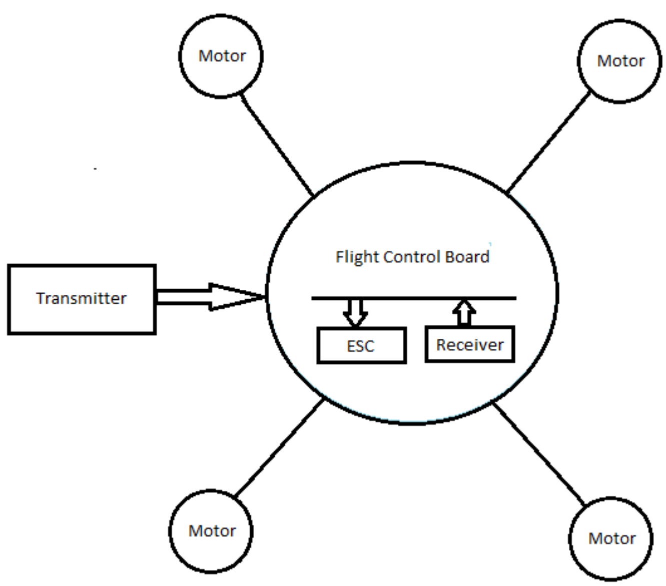
- Detection of flying quadcopter in a specified arena.
- Ball detection in same arena when thrown into arena.
- Moving quadcopter towards ball for catching it through shortest route.

Domain of project

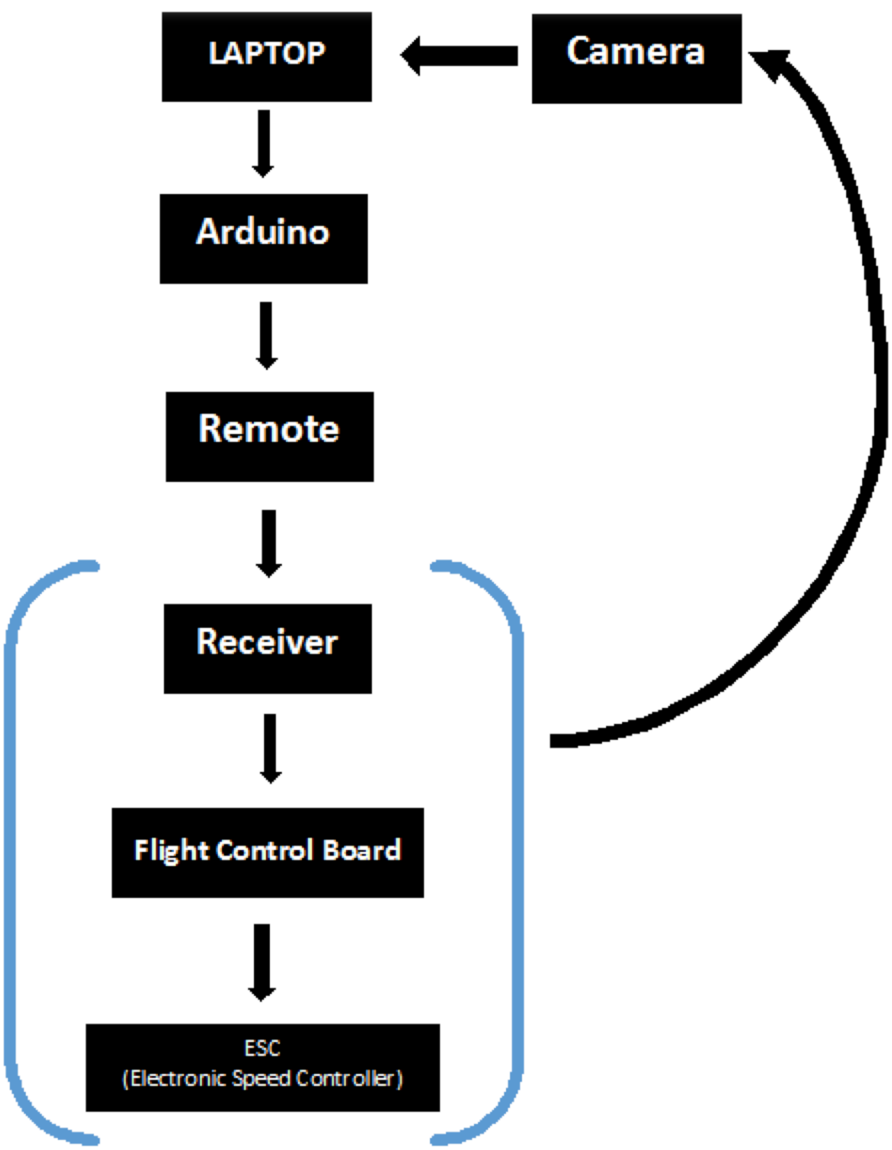
- Vision
- Robotics
- AI Planning

Block Diagrams of System

- **Block Diagram of Quadcopter**



- **Block Diagram of Feedback Loop**



1. Vision

1.1 Ball Detection in 2D

Ball detection in 2D environment has completed by using concepts of vision. Ball presence is detected with camera when it enters into arena. Camera provides this information to the system which performs computations to calculate color, shape and location of the ball. OpenCV functions have used to locate ball from information obtained by the camera. Shape of the ball is obtained in the same way. Following parameters have calculated.

I. **Calculation of X-Axis and Y-axis**

   Camera coverage area has treated as mirror image of first quadrant that is reflected on laptop screen as fourth quadrant. Position of ball along x-axis and y-axis has calculated with respect to origin of the quadrant.

II. **Calculation of radius**

   Calculation of radius has done by using Open CV function.

III. **Calculation of angle of projection**

   Values of x- and y-axis calculated before has used to calculate angle of projection of ball by using following formula:

   Angle of Projection=$\tan^{-1}(y/x)$

IV. **Time Calculation**

   System time has used for calculating time between two consecutive frames.

V. **Distance Calculation**

   Distance covered by ball has calculated by x-and y-axes of ball in two consecutive frames using distance formula.

   Distance=$\mathrm{Sqrt}[(X_2-X_1)^2+(Y_2-Y_1)^2]$

VI. **Speed Calculation**

   Speed has calculated by dividing distance by time calculated above.

   Speed=Distance/Time

1.2 Ball trajectory in 2D

Expected landing point of the ball in 2D has calculated on the basis of ball speed and angle of projection. Ball speed is the measure of distance between two consecutive frames provided by camera. Trajectory of the ball is result of using mathematical formulas.

1.3 Ball Detection and trajectory in 3D

Ball will be detected and trajectory will be drawn in 3D using two cameras placed on two adjacent sides of square shaped arena .X- and y-axes of ball will be calculated from both camera and x-axis of one camera will be treated as z-axis of second camera.

1.4 Detection of Quadcopter

A ball against each edge of the quadcopter will be attached .Each ball will have different color to identify each edge of quadcopter .Position of the quadcopter will be detected in the arena just like ball.

2. Robotics and AI Planning

2.1 Quadcopter Controls

Control wires of the quadcopter are plugged into arduino board (***an open-source electronics platform based on easy-to-use hardware and software***).This board in turn is connected with system. Quad copter stability is obtained by four types of motor movements.

- **Radar** (horizontal rotation)
- **Elevator** (rotation around side-to-side axis and vertical axis also known as **pitch** and **yaw**)
- **Throttle**(speed controller)
- **Aileron**(movement of flap of wings that control rolling and banking movement)

Quadcopter activity for catching ball is dependent upon forward ,backward, left, right, upward, downward and left and right radar.**KK2.1.5** (flight control board) is present on quad copter with **auto-level function** which reinforces quadcopter stability and motion smoothness. Input data is sent through from laptop through arduino board towards quadcopter remote.

Remote modifies this input data into signals and these signals are received by the receiver of quadcopter. Flight control board takes decision about motors rotation in rpm and gives output to ESC. Calibrations are made by selecting craft type, check motor layout/propeller direction on flight control board.**ESC** (electric speed controller) sends current signals to motor which rotates and quadcopter flies. Smoothness in motion is enhanced by trims given through remote.

2.2 Generation of Specific output from Remote

Results from detection of ball and quadcopter will be used as parameters by system to generate output .The output will be fed to remote through arduino board to generate signals for quadcopter control. The process will disintegrate into following steps:

- Arduino Software will be connected with our C++ code
- Arduino board will be initially provided with random inputs which will generate actual input by starting AI Planning loop.
- These actual inputs will then used to control quadcopter.

2.3 Quadcopter Stability

Quad copter stability will be attained calculating angle of projection of ball attached to its edges and using this information for generation of specific output from remote. Processes detail is already given in point 2.2 and 1.4.

2.4 Quadcopter movements for ball catch

Quadcopter will fly at specific height within arena in stable condition. When a ball will enter arena with specific peak height of 6 feet its expected landing position will be calculated through vision. This data will be used for calculating new input signals for quadcopter. Data will contain information about expected trajectory of the ball. A feedback loop will be generated that will guide quadcopter to move at new position. When ball will reach at expected landing position, flying quadcopter will already be there to catch it.

System Constraints

- Camera dimensions **=>** 5x5x5 feet
- Ball color **=>** Red
- Background color **=>**White
- Light placement**=>** behind each camera
- Angle of projection of ball **=>** dynamic
- Speed of ball **=>**150-187 px per microsec
- Path of the ball **=>**trajectory
- No Wind
- Peak point of trajectory **=>** 6 feet above quadcopter

Alternative System Constraints

- Camera dimensions **=>** according to camera selected
- Ball color **=>** White or may or may not random
- Background **=>**Black
- Type of lights **=>** Random
- Angle of projection of ball **=>**Fixed at any angle

Risk and Alternative Plans

- **Risk and Alternative Plan Number 1**
  If we will fail to achieve quadcopter stability (goal mentioned in point 2.3) then we will replace our quadcopter with any auto stable quadcopter. We will try to achieve all other mentioned goals with auto stable quadcopter.
- **Risk and Alternative Plan Number 2**
  If ball catching goal will fail (goal mentioned in 2.4) then we will introduce another flying object within arena instead of ball at a height above quadcopter. Our quadcopter will move towards that object and will stay one foot below that object.

Table of Milestones

Goal	Deadline	Alternative Plan
1.1 **(a)** Ball detection at speed of 150-187 px/microsec	24-02-2015	
1.1 (b) Ball detection at high speed	09-03-2015	
1.2	16-03-2015	
1.3	16-03-2015	
1.3	16-03-2015	
1.4	23-03-2015	
2.1(part 1)	24-02-2015	
2.1(part 2 final)	16-03-2015	
2.2,3.1	After finals of P1	
2.3	23-03-2015	1
2.4	After finals of P1	2

Current Explorations

- Kalman Filter (http://en.wikipedia.org/wiki/Kalman_filter)
- Machine Learning
- Feedback Loops

Acknowledgements

- Dr Syed Zeeshan Mehfooz(King Fahad University, KSA)
- Amir Zahir (Vision Expert,LUMS)
- Online tutorials by Jitendra Malik (Stanford University, USA)