6/11/2015

BALL CATCHING QUAD-COPTER

MID EVALUATION OF FYP-2

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

Table of Contents

Ball Catching Quad-copter

Introduction

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

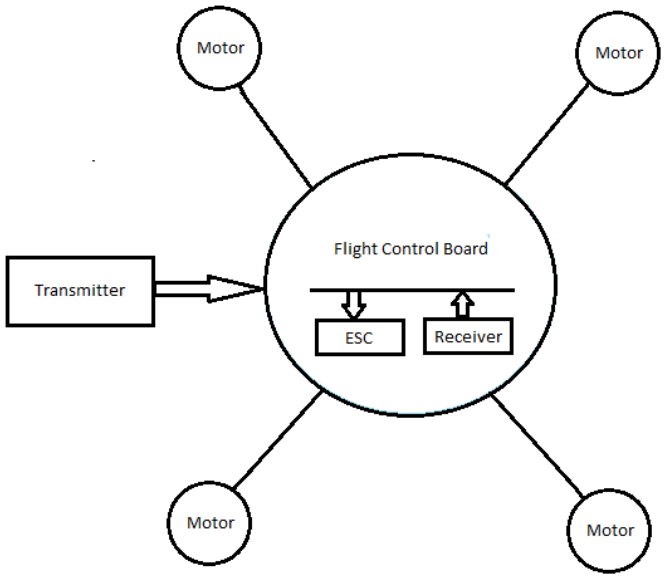
- Detection of flying quad-copter in a specified arena.
- Ball detection and trajectory drawing when thrown into arena.
- Moving quad-copter towards ball for catching it through shortest route.

Domain of project

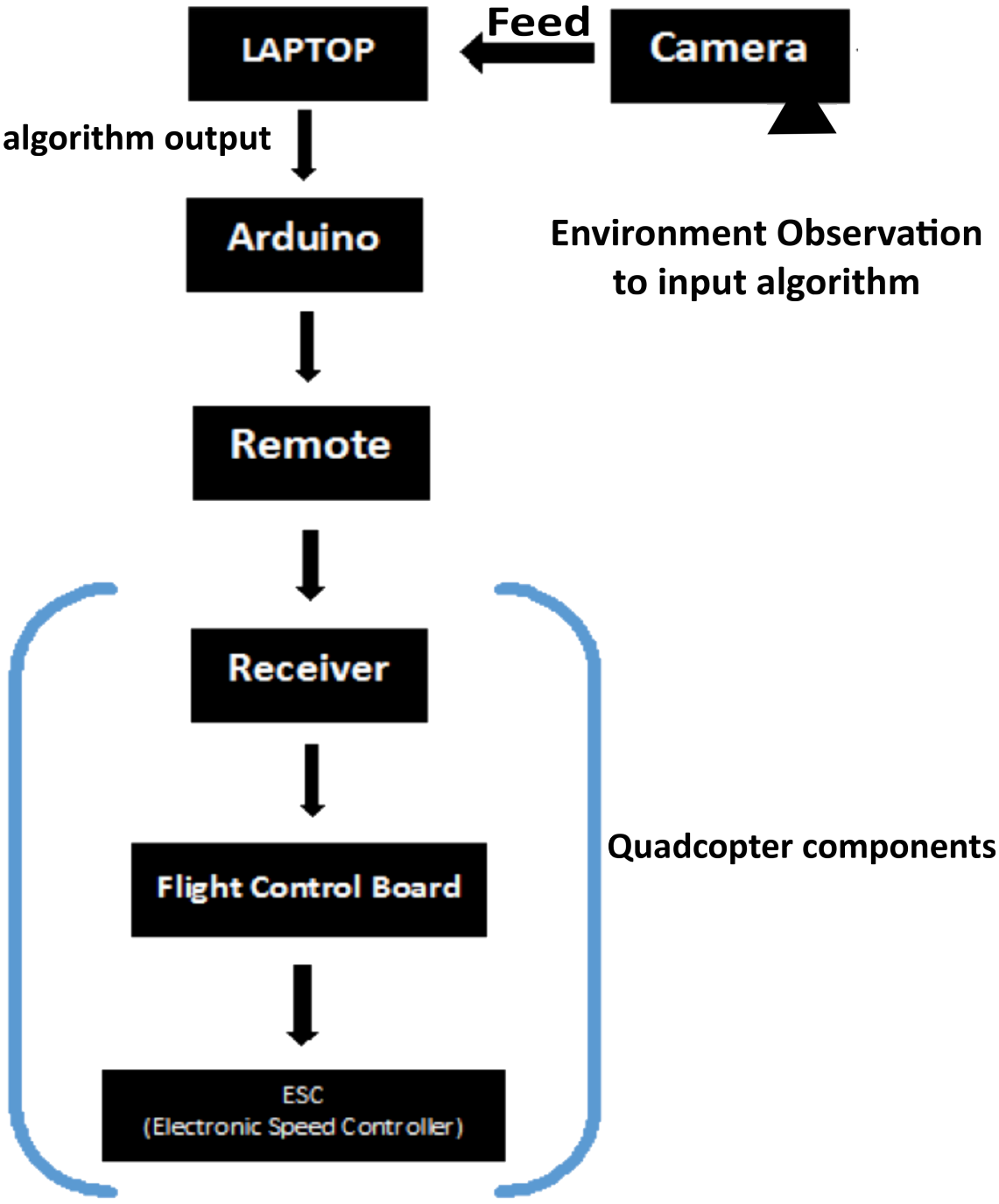
- Vision
- Robotics
- AI Planning
- Physics

Block Diagrams of System

- **Block Diagram of Quad-copter**



- **Block Diagram of Feedback Loop**


LAPTOP
Feed
Camera
algorithm output
Arduino
Environment Observation
to input algorithm
Remote
Receiver
Flight Control Board
Quadcopter components
ESC
(Electronic Speed Controller)

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 the 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 Cartesian plain. 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=Sqrt [$(X_2-X_1)^2+(Y_2-Y_1)^2$]

VI. **Speed Calculation**

Speed has calculated by dividing the distance with 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 difference between two positions of object in two consecutive frames divided by time provided by camera. Trajectory of the ball is result of using physics formulas using simulator.

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 (top and side view) of square shaped arena. X and Y-axes of ball will be calculated from both cameras and X-axis of one camera will be treated as Z-axis of second camera.

1.4 Detection of Quad-copter

A ball against each edge of the quad-copter is attached. Each ball has different color to identify each edge of quad-copter. Position of the quad-copter has detected in the arena just like ball detection.

2. Robotics and AI Planning

2.1 Quad-copter Controls

Control wires of the quad-copter 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. Four types of motor movements obtain for quad-copter stability.

- **Radar**
- **Elevator**
- **Throttle**
- **Aileron**

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

2.2 Generation of Specific output from Remote

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

- Arduino Software will be connected with our C++ code
- Arduino board will be initially provided with test inputs to perform test maneuvers, then after that algorithm will generate actual input by using camera feed and after that AI Planning loop will be started.
- These actual inputs will then be used to control quad-copter.

2.3 Quad-copter Stability

Calculating the X and Y-axes of balls attached to its corners will attain Quad-copter stability and using this information control signals will be generated for quad-copter maneuvers. Processes detail is already given in point 2.2 and 1.4.

2.4 Quad-copter movements for ball catch

Quad-copter will fly at specific height within the arena in a stable condition. When a ball will enter into the arena we will calculate its landing point using the physics formulae. This data will be used for calculating new input signals for quad-copter. Data will contain information about expected landing point of the ball. A feedback loop will be generated that will guide quad-copter to move at new position. When ball will reach at expected landing position, flying quad-copter will already be there to catch it.

3.Physics

3.1 Calculation of landing point of Ball

We will write an algorithm using laws of physics that will find the landing point of ball. Speed of the ball, ball's height from ground, when thrown, and angle of projection of ball will be provided to algorithm. The algorithm will provide its path, x-displacement and its peak point of trajectory. X-displacement will give us landing point of ball.

System Constraints

- Camera dimensions **=>** 5x5x5 feet
- Ball color **=>** Red, Blue and Yellow
- Background color **=>**Black
- Light placement**=>** Behind each camera
- Angle of projection of ball **=>** Fixed
- Speed of ball **=>**150-190 pixels per micro sec
- Path of the ball **=>**Trajectory
- No Wind

Risk and Alternative Plans

If ball-catching goal will fail then we will make a spider cam. Ropes from quad-copter's edges to the side supports will tie the quad-copter. Then quad-copter will move along x or y-axes through the rope drawn in either of the direction.

Table of Milestones

Goal	Deadline
1. ROBOT MOVEMENT IN GRID	25-06-2015
2. ROBOT MOVENT DIRECTED WITH SPECIFIED ANGLE	12-07-2015
3. WORKING ON QUAD ASMUCH AS POSSIBLE	FINAL PRESENTATION

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)