CHUCK State Estimation

Jonathan Sudiaman, Michael Welch

April 11, 2017

1 About CHUCK

CHUCK is a cornhole-playing robot that was designed by Professor Derbinsky and one of his students, Tyler Frasca. Cornhole is a competitive game between two teams, (teams of one or two), each with their own set of colored beanbags, and two boards with a cornhole near the top of the board. The way it is played is that each team alternate turns throwing a beanbag at the board that is across from them. Each bag that lands on the board counts as one point for that team. Any bag that is thrown through cornhole on the board counts as three points for that team. Any other bag that is not on the board does not factor into the score. At the end of each round of throwing, you subtract the lower score from the higher score and the difference is added to the team's total score that had the higher score. If both teams have points that cancel each other out, no team gains any points for that round.

2 Project Motivation

CHUCK is able to throw a beanbag with given parameters. CHUCK is not able to tell what the score is or what the current state of the game is. The goal of the project was to accurately locate the cornhole board, the cornhole hole and any beanbags in play within CHUCK's vision and use that to let CHUCK know what the score was and what the state of the game is.

3 Methods

3.1 Detecting Board & Beanbags

To accomplish this, OpenCV was utilized to be able to take a picture or video and be able to extract relevant data from the data feed. In this case, OpenCV was used to be able to detect colors of different objects, e.g. the beanbags, the board, and create a contour around the corresponding object. In adjunction to this, HSV was used to more accurately distinguish different colors from each other. HSV is another way to detail information from a color by breaking it up into hue, saturation and value, or brightness. To do so, we created a mask of an image from the provided dataset that only displayed objects of a certain HSV range, displaying it as a solid white color. Any other color that did not match up with this was changed to black. This was done for both colors of beanbags and for the border of the board. This was accomplished by utilizing the cv2.rectangle() function. By utilizing this

function, along with the mask of the image, it was possible to create a bounding contour around the white spots from the image and return those boxes.

3.2 Detecting Cornhole

To be able to detect the cornhole, we utilized another OpenCV function called cv2.circle(). It was used in conjunction with Circular Hough Transform, which is an edge-detection algorithm. Circular Hough Transform takes an image that was passed in and blurs it. It will then try and detect where a circle is located. From there, we then were able to create a circular contour at the location that the Hough Transform returned.

4 Results

In order to obtain a measure for how well CHUCK can estimate board states, we compared our estimates to actual values from an annotated dataset. The dataset contains 198 images along with respective information about positioning and dimensions of the beanbags, cornhole, and board. Our measurements are done in two ways – first, an abstract comparison which compares our estimated score to what the score actually is, and secondly, a fine-grained comparison against X/Y coordinates and width/height or radii of key objects.

4.1 Score Comparison

Processing images from the dataset, we were able to estimate scores with an average absolute (i.e. not percent) error of .5 and a median error of 0.

4.2 Positional/Dimensional Comparison

Results for the fine-grained comparisons were measured by percent error, and are as follows:

Object (Position and Dimensions)	**Average % Error**	**Median % Error**
Board	1.1	0.7
Cornhole	9.2	4.2
Red Beanbag	5.5	1.9
Blue Beanbag	7.1	3.0

5 Issues

5.1 Handling Clumped Beanbags

There were some issues with detecting the beanbags, as the program was not accurately able to distinguish multiple beanbags of the same color that were too close in proximity

to each other. As a simplistic solution, we attempted to set a certain size that a single contour could be before it was considered multiple beanbags. This caused some issues where a very large beanbag was split into two. It also did not solve any issues with more than one beanbag being on top of each other while still being under the size range.

5.2 Misplaced Cornhole

Circular Hough Transform unfortunately detected many other circles in addition to the cornhole. To try and counteract this, we utilized the size and location of the board to more accurately detect the location of the cornhole by using a constant distance from the top of the board to the center of the hole. Another issue with the cornhole detection was that sometimes the cornhole detected was not located at the correct location, usually due to at least one beanbag overhanging the hole while still on the board. The algorithm would then place the circle over the beanbag, and in one extreme case, at the bottom of the board.

6 Conclusion

Notwithstanding our progress, there is still much to be done in order for CHUCK to make it into the big leagues. In the future, CHUCK can be improved by giving it more information to use in its state estimations, such as how many beanbags have been tossed in the current round. A learning-based detection algorithm, akin to Haar Cascades, may also prove to be valuable.