

Computer Vision & Pattern Recognition

Assignment 7

Arslan Ali, Massimo Parisi

June 2, 2022

Exercise 1 & 2

The attached code follows these 'high-level' steps for each of the requested videos:

1. Read the first frame of the video and compute the boundaries of the snooker table and the six pockets
2. For each frame, detect the balls
3. Check if a ball is in a pocket; if found, save the frame and highlight the aforementioned pocket.

In the following paragraphs we will explain how these steps were technically performed.

Detect boundaries and pockets

We detect boundaries and pockets for the first frame in the following way:

1. Preprocess the frame using different techniques (blurring, thresholding, erosion, dilation, Canny edge detection)
2. Use `cv2.findContours` to find the contours of the table
3. Take only the contour with the minimum area
4. Use `cv2.minAreaRect` to find the (rotated) rectangle with minimum area enclosing the aforementioned contour
5. Use `cv2.boxPoints` to find the four vertices of the aforementioned rectangle. We now know the table's boundaries
6. Check the lengths of the sides of the rectangle and find the side pockets as the medium point of the two longest sides



Detect balls

We detect balls for each frame in the following way:

1. Preprocess the frame using different techniques (saturation enhancement, thresholding, erosion, dilation, Canny edge detection)
2. Find circles using `cv2.HoughCircles`



Find potted balls

We find potted balls for each frame in the following way:

1. In order to avoid duplicates of the same potted ball, check if a ball has been potted recently, if yes ignore the current frame
2. Compute the distance between each ball-pocket combination and if one of them is shorter than a certain error, we consider it as a potted ball
3. After detecting a potted ball, we save the frame and draw a circle on the pocket in which the ball has been potted



All the code and results for this exercise are in the attached file (`snooker_table.py`, `functions.py`, `results.zip`).