1. How well do you expect this to work on other images?

It doesn’t work very well on these images. Without the blurring and canny edge detection, there was too much noise, and it found non-existing circles. The blurring and canny edge detection give a reasonably clear picture of circles, but hough circles doesn’t seem to detect many of those.

Also, I didn’t have time for properly calculating the pip\_index, so that’s bad as well (doesn’t account for the bar in the middle).

I expect this to work on other images just as bad as it does on these images: it will probably not found many circles, but it will find some of them.

1. What are possible fail cases of this approach and how would you address them?

* Doesn’t work on .png files
* Doesn’t work in bad lighting conditions
* Doesn’t work when the circles are not placed exactly on pips
* Doesn’t work when we don’t follow the naming conventions
* Don't support linux style file paths
* Doesn’t work when the circle sizes are different

1. How would you implement finding the colors of the checkers and distinguishing which player the checker belongs to?

In the original image (before the canny edge detection), I would take the average of the brightness inside the found circle. If it is below a given threshold, then it’s the dark player, if it’s above, then it’s the light player.