1. Table detection

In order to take detect the table, we have to take advantage of its features. The table edges are long straight lines. It is also blue and stands out from its environment. So our approach consists of finding the lines present in the input image and then filter them according to different criteria. After that, the lines are grouped depending on which side of the table they belong to. From this group of lines, the four sides of the table is calculated

1.1 Lines detection

Using Hough peaks and Hough lines, the algorithm start by detecting the lines that are present in the input image. We consider as a valid lines the ones that are on the edge of the table. The lines are therefore filtered on the following criteria:

Orientation

Given the point of view of the camera, we don't expect to have an edge of the table showing vertically on the image. Hough lines algorithm uses a polar coordinates system, therefore we filter the lines directly from θ of the polar coordinates.

Color separation

A valid line will have from one side the blue table color and then from the other side any other color (floor green, wall white…). To decide which lines are valid, two sample pixels are taken from each side of the line. The midpoint of the line along with the orthogonal line going through the midpoint are calculated. The sample pixels are equally distant points from the centroid on the orthogonal lines. The HSV values of these points are considered as it is a more robust way to deduce the color.

Given two points $p_1 = (h_1, s_1, v_1)$ and $p_2 = (h_2, s_2, v_2)$, a color is considered blue if

$blue_1 = (h_1 > BLUEMIN) \wedge (h_1 < BLUEMAX) \wedge (h_1 > VLIGHTMIN)$

where the thresholds (BLUEMIN, BLUEMAX and VLIGHTMIN) are defined from the hue values of the blue color. A minimum value of brightness is also imposed as blacks may be considered as blue.

A line is considered valid if

$line = blue_1 \veebar blue_2$

1.2 Lines grouping

Still needs to be developed

After detecting the lines on the edge of the table we need to cluster them using kmeans ….

Because of the distortion caused by the wide angle lens, the lines detection algorithm sometimes return many lines for one table edge.

we have multiple lines per edge => weighted average grouping

then grouping => calculate interescts and then finished