Section A: Getting Region of Interest

Please find *getPerspective.py* file in the folder *Resource*. This python file has the code which crops part of input image enclosed by a black rectangle. For example, when the image of the arena as shown in Figure-1 is given as input, the output image as shown in Figure-2 is returned. Note that this code crops the redundant parts of the image to get the arena image bounded by the black rectangle. Run the file *getPerspective*.py which will generate two images, input_image.jpg and output_image.jpg. **You may have to edit your code for adjusting the np array for the black color and image resolution.**

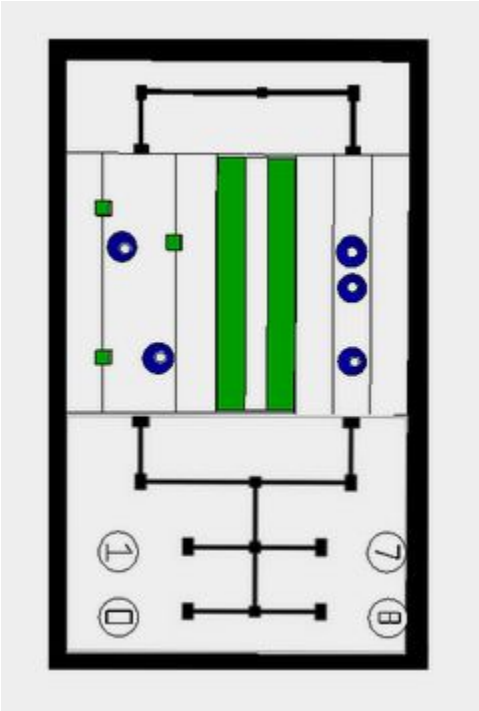

Figure-1. Input Image

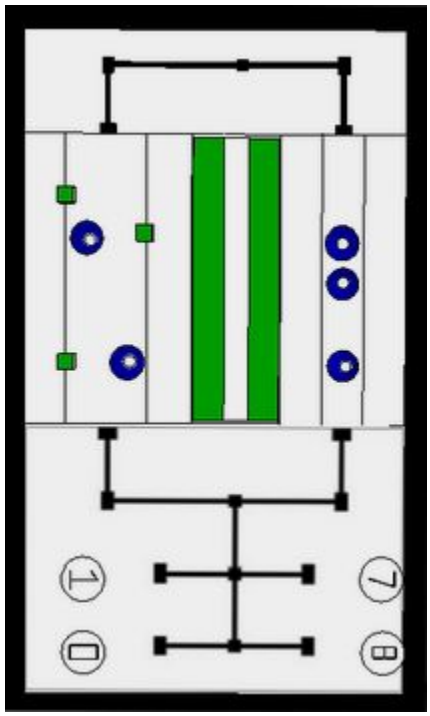

Figure-2. Output Image

Section B: HSV trackbar

Please find *hsvtrackbar.py* file in the folder *Resource*. This python file has the code which will help you to process the video frame in HSV mode for getting the np arrays for particular object of interest in the arena.

Section C: Hint for robot position and orientation detection

Consider the arena as shown in Figure-1 and Figure-2. Consider the figure to represent Top-view of arena captured by overhead camera. Assume that image is of

600x600 pixels. Robot is represented by grey circle, wheels are represented by blue rectangle and small blue circle represents castor wheel.

Task: Think about an image processing based algorithm to find position and orientation of the robot on the arena. You must try it yourself on your theme arena to get desired result. You can put the robot on the arena in two different position and orientation as shown in Figure-3 and Figure-4.

The main component of arena is the flex sheet. If you note carefully, printed part consist of logos, some text etc. These part of flex sheet is not of our interest. Our **Region of Interest (ROI)** i.e. part of printed flex sheet which is used for performing the task. In this case it is part **bounded by thick black border.** ROI may change according to theme, students are expected to find out what is the ROI for the assigned problem statement.

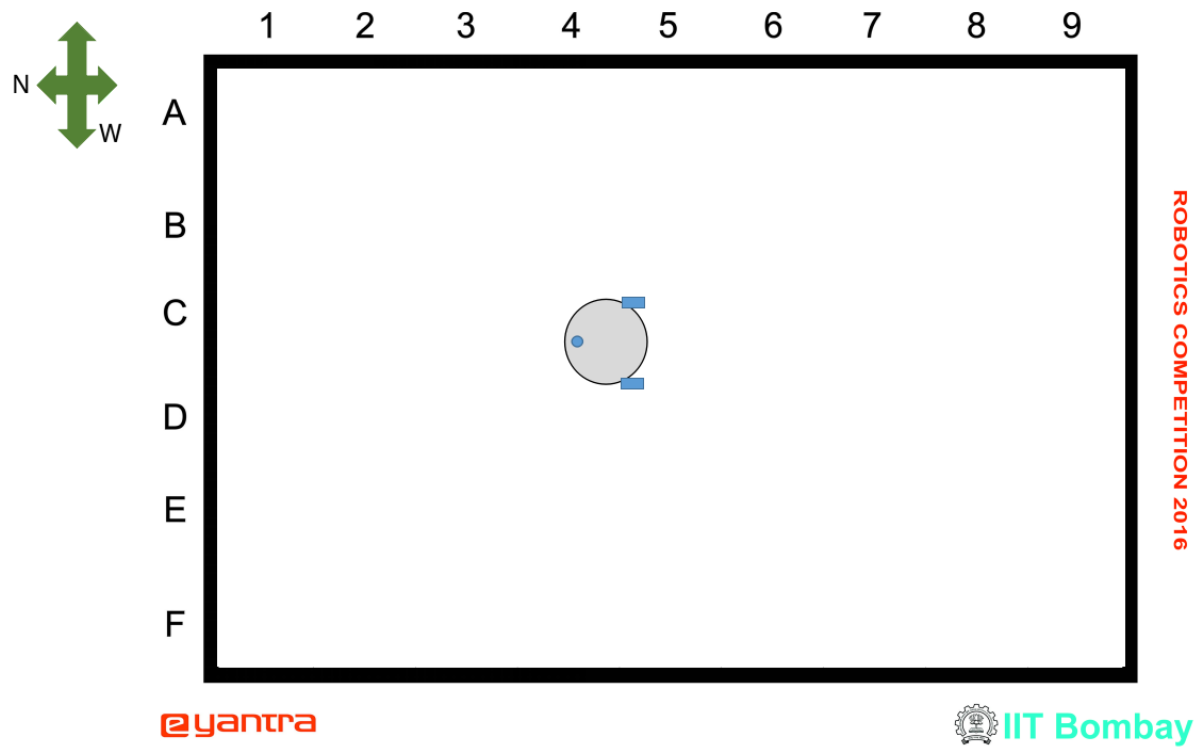

Figure-3 Robot Position and Orientation - case-1

In order to find the position and orientation of robot your first task should be to extract out ROI.

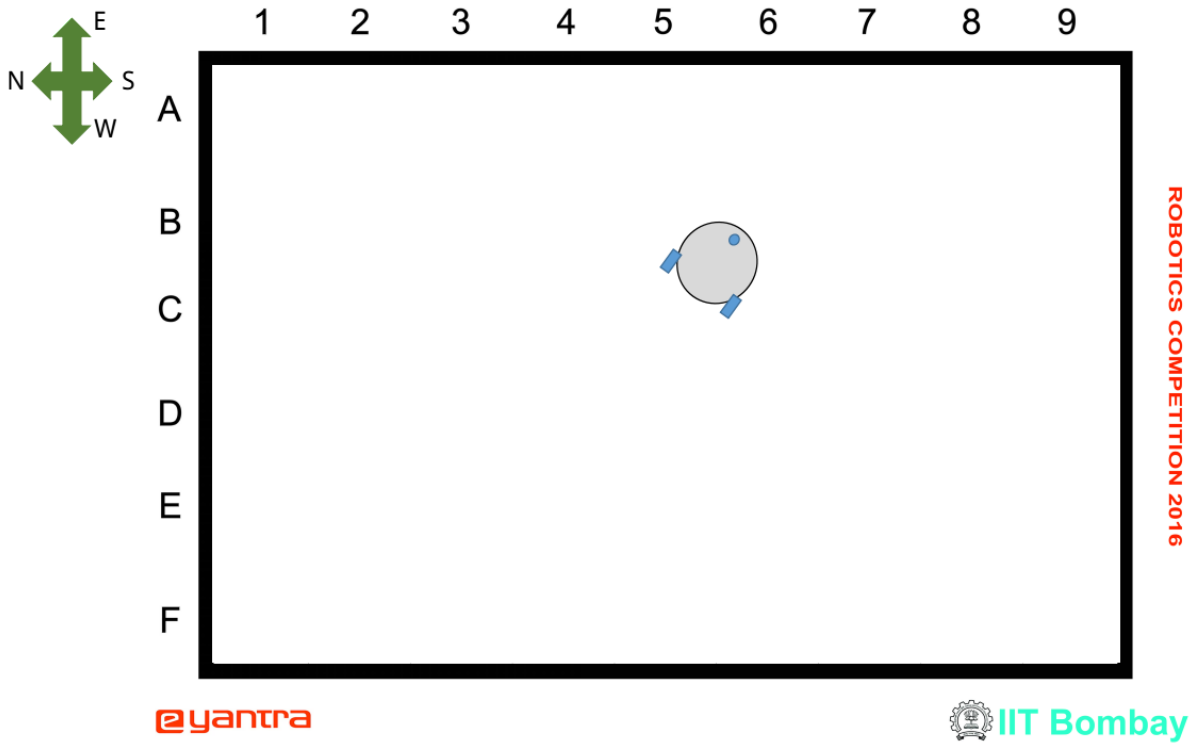

Figure-4 Robot Position and Orientation - case-2

Next task is to find position and orientation of the robot. Note that there is an direction arrow in top left corner of image for your reference. Orientation can be in any form, be it an angle or a direction.

Position:
You have to design an algorithm to find the position of robot (i.e. find pixel values of center of robot. Probably you would have guessed, center pixel should given position in 2D i.e. distance on x-axis and ,y-axis). Find out position on robot for two cases shown in Figure-1 and Figure-2

Orientation:
As is evident for Figure-1 and Figure-2, robot orientation has changed, how can you find the correct orientation. You can use the direction arrow (it is 4 direction shown, you can choose to add more direction in between to get finer resolution). You aim should be to tell how much and in which side the robot is aligned at any point of time.

Guidelines: You may identify the bot using image processing techniques mentioned on portal's resource page. You are free to customise your bot to aid your image processing algorithms.