



# e-Yantra Robotics Competition Plus

(eYRC+ Pilot)

<eYRC+#925>

Team leader name	Suvrat Shankar Chaturvedi
College	JSS Academy of Technical Education Noida
e-mail	om.vedvasu@gmail.com
Date	30/11/2014

## Scope of the Task

(5)

Description and Purpose of the task:

The task1 assigned to us can be broadly divided onto two parts:

- 1) To present our theoretical knowledge through a set of questions and answers from the videos and text tutorials provided.
- 2) To show our practical understanding by the following task.

An image containing balloons and a shotgun is given. Balloons are identified by different letters. We have to display the letter of that balloon the shotgun is pointing.

Solution of this task can be applied in the practical problems of hurdle detection, decision making and in gaming consoles or creating multi-dimensional environment.

## Camera and Image Processing

(8)

Write down the answers to the following questions.

1. What is the resolution (size) of the picture taken from your camera?
2. What is the resolution (size) of the test image assigned in the task?
3. What is the use of thresholding an image?
4. What is the use of color masks?

Answers:

1. For snapshot : 640\*480 (can be varied for still image)
2. 770\*475

3. Use of thresholding an image is create image which can be analyzed easily especially by the logical systems working on high or low (either black or white). It increases the readability of the image. It also differentiates the image on the basis of intensity.
4. Use of color mask is to detect the color intensity (BGR) for any color of an image or in a video.

## Software used

(7)

Write down the answers to the following questions.

1. Write a function in python to open a color image and convert the image into grayscale. You are required to write a function *color\_grayscale(filename,g)* which takes two arguments:
  - a. filename: a color image (Test color image is in folder "Task1\_Practice/test\_images". Pick first image to perform the experiment.)
  - b. g: an integer

Output of program should be a grayscale image if g = 1 and a color image otherwise.

Answer:

```
filename="Task1_Practice/test_images/1"
def color_grayscale('filename',g):
    """
    filename-- input color image stored as file(Test color image is in folder
    "Task1_Practice/test_images". Pick first image to perform the
    experiment.)
    g -- int 0 or 1
    returns img-- grayscale of input image if g=1 else color image
    """

    if g==1:
        img = cv2.cvtColor(filename, cv2.COLOR_BGR2GRAY)
    else:
        img=filename
    return(img)
```

2. Write a function in python to return only the red portions of the image based on the appropriate HSV range.

< Answer format:

Use the snippet given below to write code for the function. Inline comments are mandatory to, explain the code. (Test color image is in folder Task1\_Practice/test\_images". Pick first image to perform the experiment.)

>

```

def red_threshold(img, hsv_low, hsv_high):

    '''
    img-- input color image stored as file (Test color image is in folder
    "Task1_Practice/test_images". Pick first image to perform the experiment.)
    hsv_low-- int list for hsv low value eg. [50,200,300]
    hsv_high--int list for hsv high value eg. [100,255,255]

    returns img--with red part only
    '''
    lower = np.array(hsv_low)
    upper = np.array(hsv_high)

    #masking is done for a any color depending upon hsv_low and hsv_high

    mask = cv2.inRange(img, lower, upper)
    img = cv2.bitwise_and(img, img, mask=mask)
    return(img)

```