Lab 3 - Mouse callback and color segmentation

Computer Vision

Alberto Pasqualetto, 2121718

7 April 2024

Task 1

In task 1 an image is required to be loaded and displayed; the reference image is shown in figure 1.

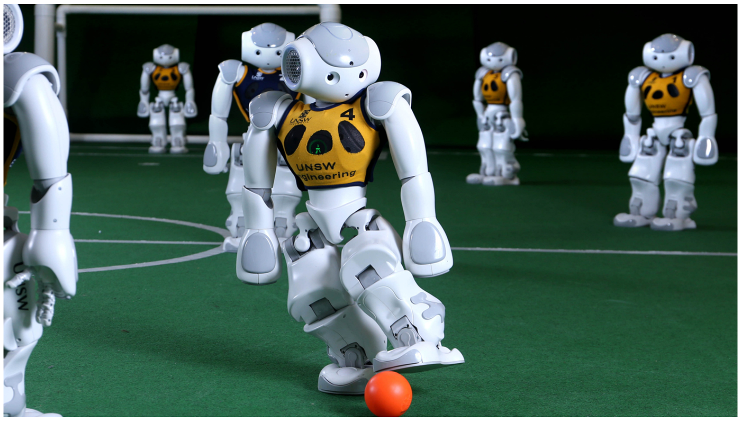


Figure 1: Original image

Task 2

Task 2 requires to write a callback function which is triggered when the user clicks on the image. The callback function should print the BGR values of the pixel in that position. A callback function is attached to a specific window using the function `cv::setMouseCallback(winname, callbackfunction)`.

The callback function is of type `cv::MouseCallback`, and it needs parameters `event`, `x`, `y`, `flags`, and `userdata`. `event` is the event that triggered the callback: in our case we need to check if the event is a left click (`cv::EVENT_LBUTTONDOWN`). `userdata` is a pointer of type `void` which can be used to pass additional data to the callback function. In our case we pass the image to the callback function, so that we can access the pixel values. `x` and `y` are the coordinates of the pixel clicked by the user which will be used to access the pixel values in the image.

Task 3

Task 3 asks to calculate the mean of the BGR values of the pixels in the 9x9 neighborhood of the clicked pixel; the mean is calculated for each channel separately.

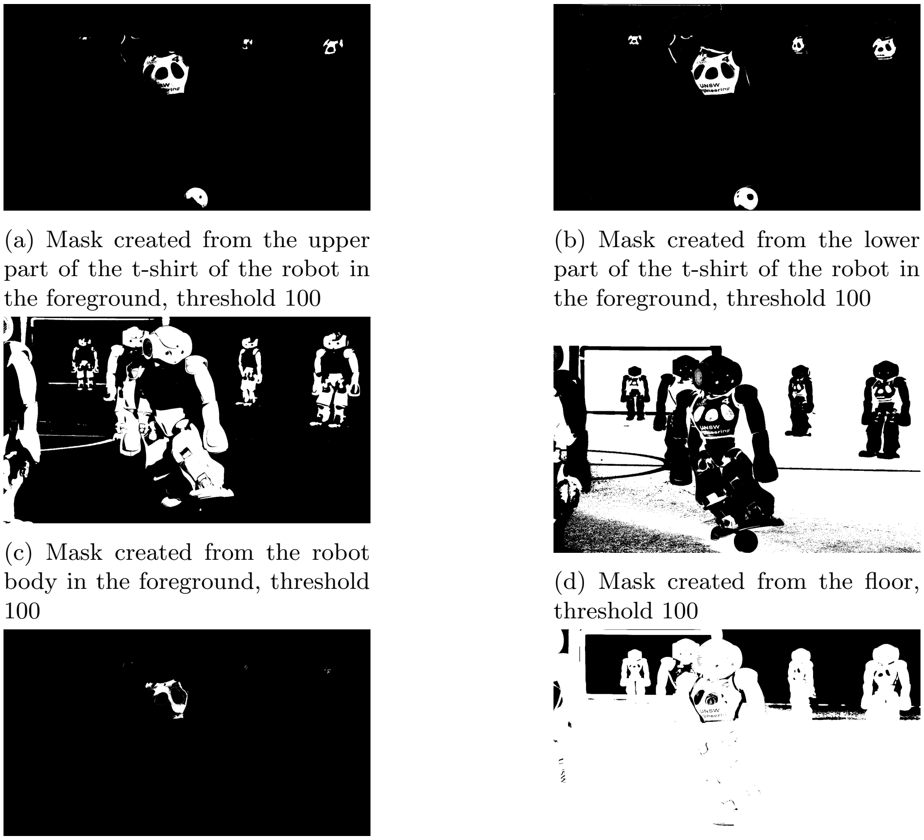
This code must be implemented in a callback function, expanding the one written for task 2.

Task 4

Task 4 requires to implement a color segmentation algorithm which creates a mask of the image where only the pixels with a color similar to the one clicked by the user are shown as white pixels (255 value for a `uchar` type), the others are set to black, so we can start from a `Mat::zeros` and only color the white pixels.

Similarity is defined by a threshold (`T`) on the Euclidean distance between the BGR values of the pixel (for all pixels) and the mean BGR values in the 9x9 neighborhood of the clicked pixel from task 3.

Some results are shown in figure 2: masks change depending on the clicked pixel even if the click is performed on the same object since it can have (slightly) different color values as shown in figures 2a and 2b; masks obviously change by clicking on different objects in the image as shown in figures 2c and 2d and when the threshold is changed as shown in figures 2e and 2f.



(a) Mask created from the upper part of the t-shirt of the robot in the foreground, threshold 100

(b) Mask created from the lower part of the t-shirt of the robot in the foreground, threshold 100

(c) Mask created from the robot body in the foreground, threshold 100

(d) Mask created from the floor, threshold 100

(e) Mask created from the t-shirt of the robot in the foreground, threshold 25

(f) Mask created from the t-shirt of the robot in the foreground, threshold 250

Figure 2: Masks

Task 5

In task 5 the same operation as in task 4 is performed, but the mask is created using the HSV color space instead of the BGR color space. The HSV color space is used because it is more intuitive to define a color range in the HSV color space than in the BGR color space.

Changing the color space (using the function `cv::cvtColor(src, dst, cv::COLOR_BGR2HSV)`) the results are usually better and sharper, as shown in figure 3.

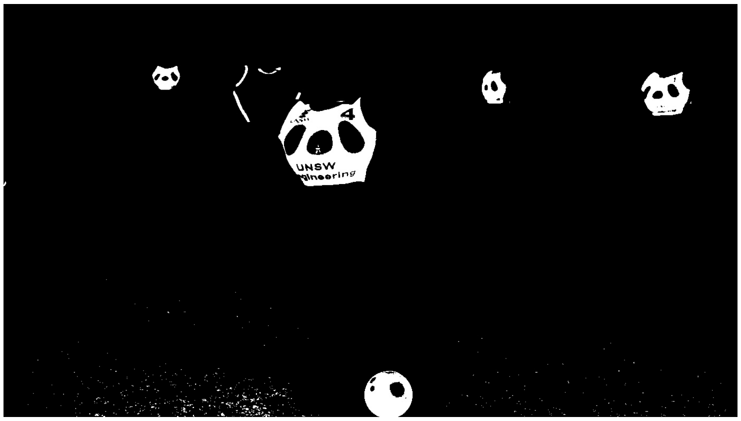


Figure 3: Mask created from the t-shirt of the robot in the foreground, threshold 100, using the HSV color space and Euclidean distance

In order to exploit the HSV color space, the distance can also be calculated using only the hue channel, since we want to isolate a limited color range, giving us better result: a better segmentation can be seen in figure 4 (false positives can be removed afterwards since they are outliers).

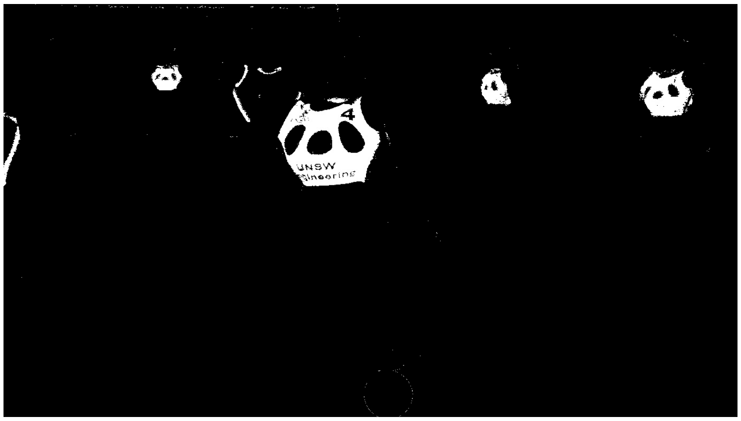


Figure 4: Mask created from the t-shirt of the robot in the foreground, threshold 5, using the HSV color space and hue distance

Task 6

In task 6 it is required to implement the color segmentation from task 4 (using BGR color space) and apply the mask to the original image recoloring the pixels in the mask with a different color: BGR(92, 37, 201) 🟥 . The result is shown in figure 5.

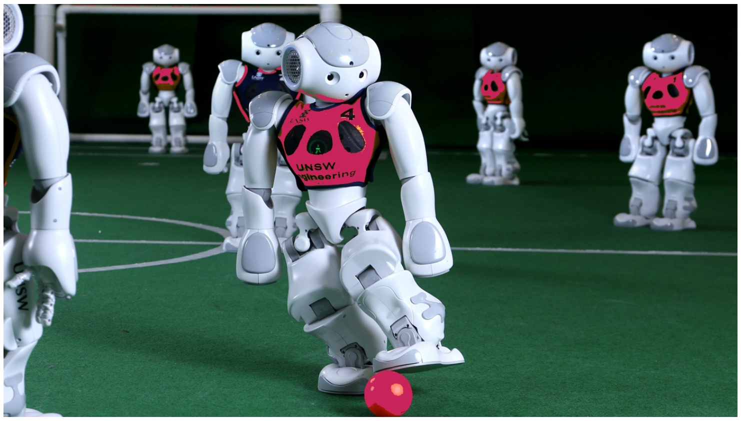
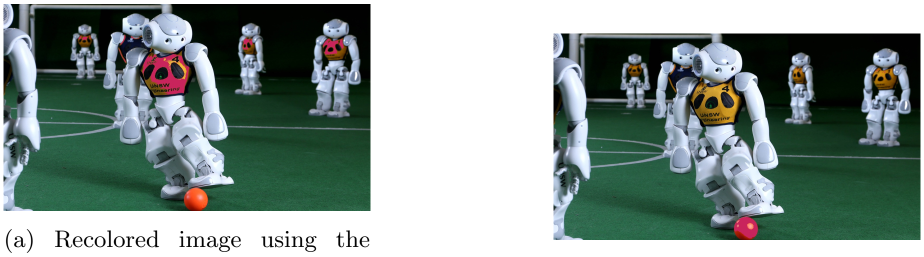


Figure 5: Recolored image using the mask created from the t-shirt of the robot in the foreground, threshold 100

Distinguishing pretty similar colors like the yellow of the t-shirts and the orange of the ball is very hard using the BGR color space, as shown in figure 5 where the ball is also recolored.

Overall, adjusting the thresholds, the results can be pretty good as shown in figure 6.



(a) Recolored image using the mask created from the t-shirt of the robot in the foreground, threshold 60

(b) Recolored image using the mask created from the ball, threshold 75

Figure 6: Recolored images with adjusted thresholds