

# Segmentation and Thresholding

---

Seperate the rabbit from the background using thresholding.

- `threshold = graythresh(grayImg)`
- `binrayImg = im2bw(grayImg, threshold)`

```
% Separate the rabbit from the background
img = imread('b2.jpg');
subplot(2,2,1), imshow(img), title('Orignial Image');

% Convert the image to grayscale
grayImg = rgb2gray(img);

% find the best threshold for binarization using Otsu's method
threshold = graythresh(grayImg);

% Binarize the grayscale image
% pixels below threshold become black, and pixels above become white.
binrayImg = im2bw(grayImg, threshold);

subplot(2,2,2), imshow(binrayImg), title('Thresholded Image');
```

**Orignial Image**



**Thresholded Image**



---

## Color-based segmentation (Intensity slicing)

---

**Extract light brown chairs**

- Loop through each pixel in the image
- Check if the pixel color corresponds to a light brown chair

```
% Extract light brown chairs only
img = imread('2.jpg');
```

```

% Get the height, width, and number of channels of the image
[h, w, c] = size(img);
% Initialize an empty matrix to store the result
result = zeros(h,w,c);

% Loop through each pixel in the image
for i=1:h
    for j=1:w
        % Check if the pixel color corresponds to a light brown chair
        if img(i,j,1) > 100 && img(i,j,1) > (img(i,j,2)+20) && img(i,j,2) >
(img(i,j,3)+20)
            result(i,j,:) = img(i,j,:);
        end
    end
end

figure;
subplot(1,2,1), imshow(img), title('Original Image');
subplot(1,2,2), imshow(uint8(result)), title('Extracted Chairs');

```

**Original Image**



**Extracted Chairs**



## Segment based on the object boundaries

```
% Segment based on the object boundaries
img = imread('3.jpg');

% Convert the original image to grayscale
grayImg = rgb2gray(img);

% Apply edge detection using the Canny operator
edgeImg = edge(grayImg, 'canny');

% If boundaries of each object is not fully connected apply dilation
% to increase the thickness of edge
edgeImg = imdilate(edgeImg, ones(5,5))

% Group pixels together
[L num] = bwlabel(~edgeImg);

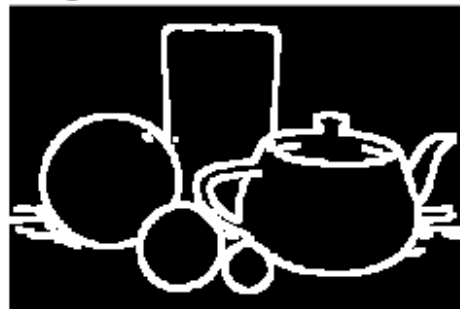
% Colorize each group
RGB = label2rgb(L);

subplot(2, 2, 1), imshow(img), title('Original Image');
subplot(2, 2, 2), imshow(edgeImg), title('Edge Detection - Black BG');
subplot(2, 2, 3), imshow(~edgeImg), title('Edge Detection - White BG');
subplot(2, 2, 4), imshow(RGB), title('Edge Detection - RGB');
```

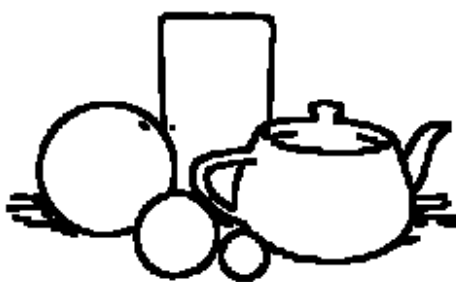
**Original Image**



**Edge Detection - Black BG**



**Edge Detection - White BG**



**Edge Detection - RGB**

