2018222_HW7

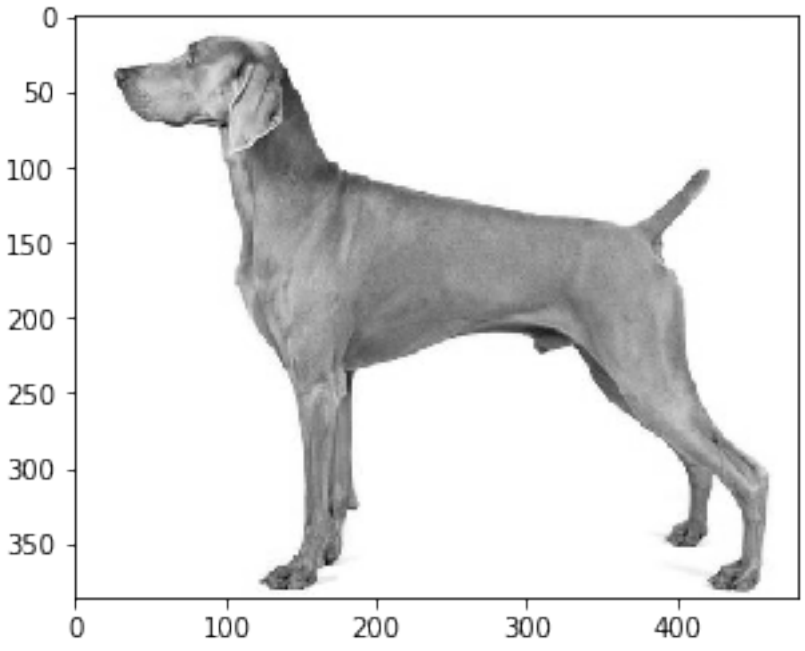
February 14, 2021

1 Computer Vision HW 7

Arka Sarkar ; 20181222

[22]:
```
import numpy as np
import cv2
import matplotlib.pyplot as plt
import copy
```

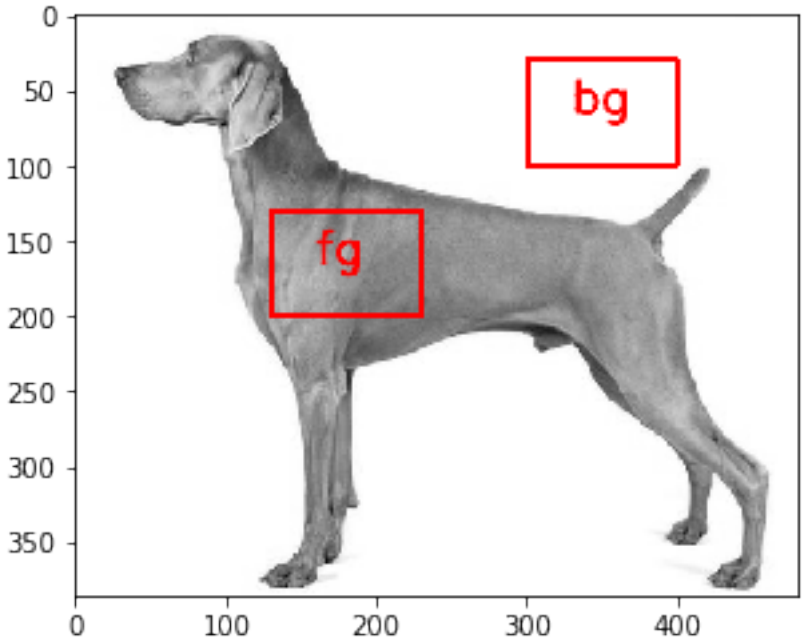
[23]:
```
image = cv2.imread("dog2.png")
plt.imshow(image)
plt.show()
```



[24]:
```
#background and foreground patch
color = (255, 0, 0)
thickness = 2
```

```
bg_patch = image[ 30:100,300:400]
fg_patch = image[130:200, 130:230 ]
bg_fg_image = cv2.rectangle(copy.deepcopy(image), (300, 30) , (400, 100) ,␣
↪color, thickness)
bg_fg_image = cv2.putText(bg_fg_image, "bg ",(330,65), cv2.
↪FONT_HERSHEY_SIMPLEX, 1, color, thickness, cv2.LINE_AA)
bg_fg_image = cv2.rectangle(bg_fg_image, (130,130) , (230, 200) , color,␣
↪thickness)
bg_fg_image = cv2.putText(bg_fg_image, "fg ",(160,165), cv2.
↪FONT_HERSHEY_SIMPLEX, 1, color, thickness, cv2.LINE_AA)
```

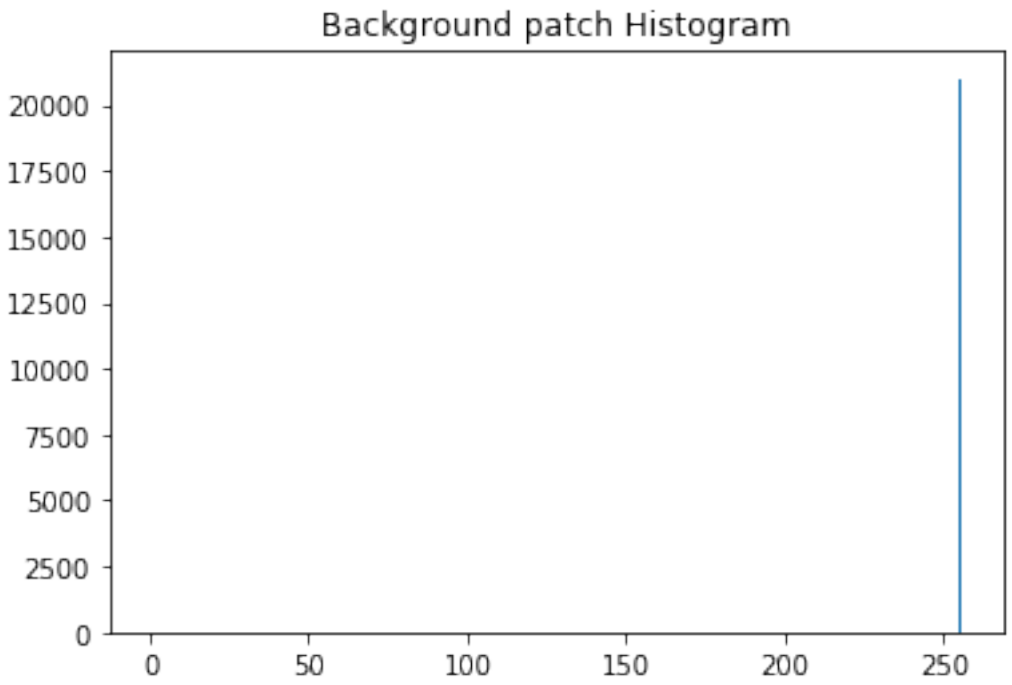
[25]:
```
plt.imshow(bg_fg_image)
plt.show()
```

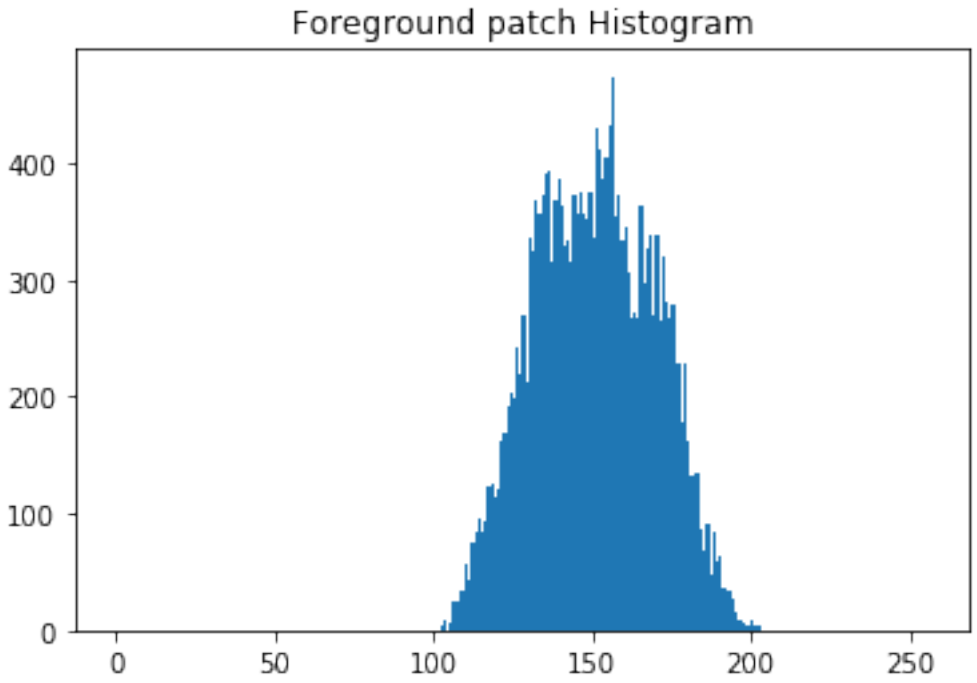


1.0.1 fg : foreground patch

1.0.2 bg : background patch

[26]:
```
plt.hist(bg_patch.ravel(),256,[0,256])
plt.title("Background patch Histogram")
plt.show()
plt.hist(fg_patch.ravel(),256,[0,256])
plt.title("Foreground patch Histogram")
plt.show()
```

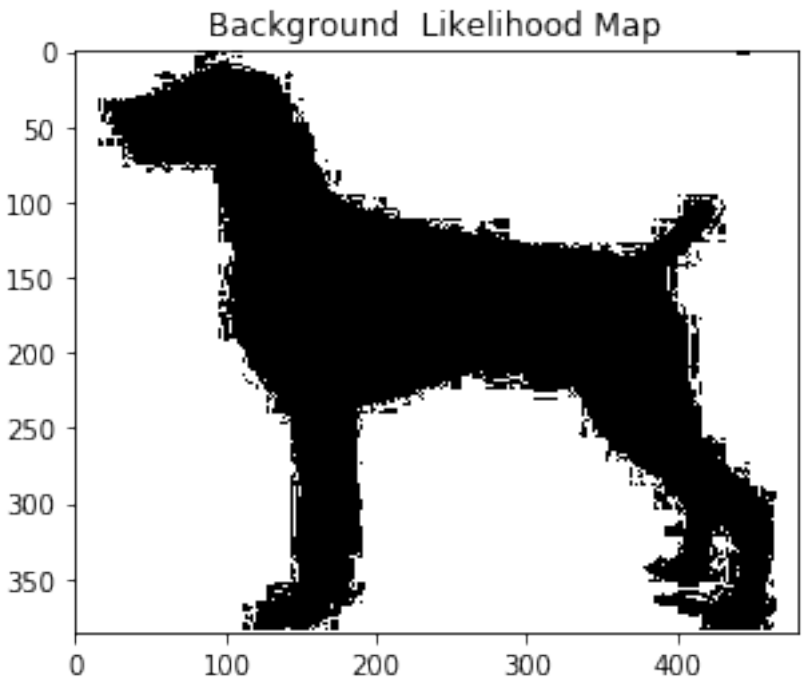


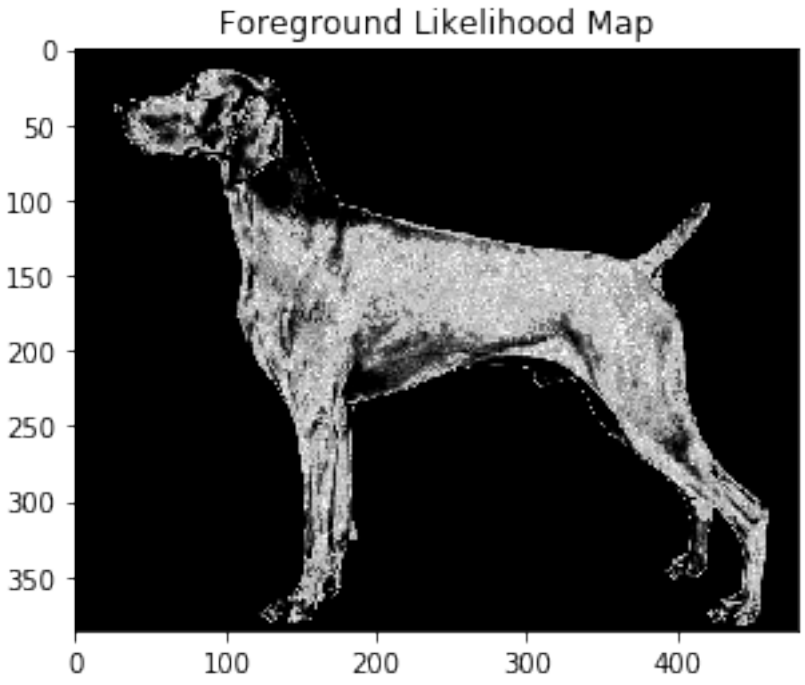


[76]:
```
def fg_bg_likelihood_map(image,fg_patch, bg_patch):
    bg_pixels = bg_patch.ravel()
```

```
    fg_pixels = fg_patch.ravel()
    bg_prob_map = np.array([0 for i in range(256)])
    fg_prob_map = np.array([0 for i in range(256)])

    for i in range(len(fg_pixels)):
        fg_prob_map[fg_pixels[i]]+=1
    for i in range(len(bg_pixels)):
        bg_prob_map[bg_pixels[i]]+=1

    bg_prob_map = bg_prob_map/np.max(bg_prob_map)
    fg_prob_map = fg_prob_map/np.max(fg_prob_map)
    fg_likelihood_map = np.zeros((image.shape[0], image.shape[1]))
    bg_likelihood_map = np.zeros((image.shape[0], image.shape[1]))
    for i in range(image.shape[0]):
        for j in range(image.shape[1]):
            fg_likelihood_map[i,j] = fg_prob_map[int(np.mean(image[i,j]))]*255
            bg_likelihood_map[i,j] = bg_prob_map[int(np.mean(image[i,j]))]*255

    plt.imshow(bg_likelihood_map , cmap = 'gray')
    plt.title("Background  Likelihood Map")
    plt.show()
    plt.imshow(fg_likelihood_map, cmap = 'gray')
    plt.title("Foreground Likelihood Map")
    plt.show()
    return fg_prob_map, bg_prob_map
```

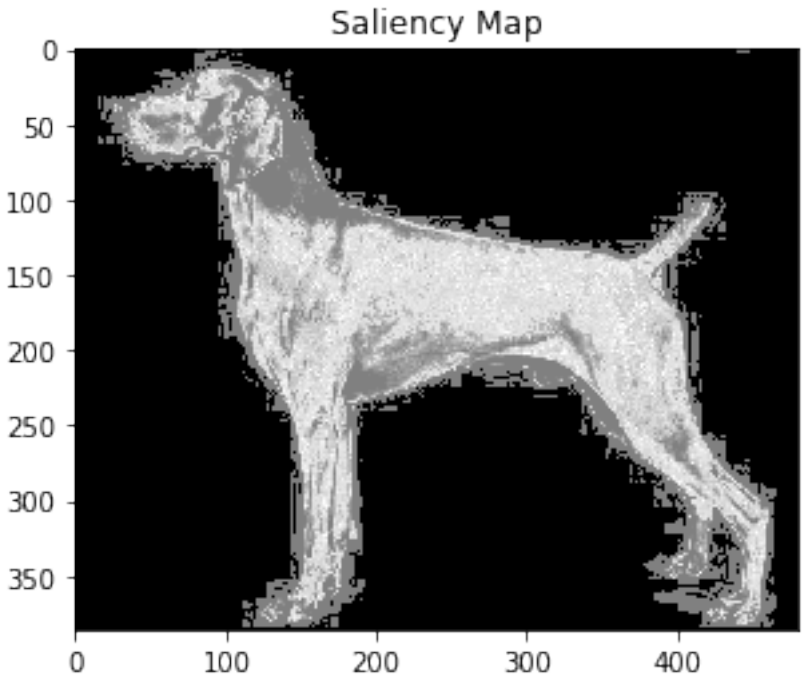
[77]:
```
fg_prob_map, bg_prob_map = fg_bg_likelihood_map(image,fg_patch, bg_patch)
```





[81]:
```
saliency_map = (fg_prob_map + ( 1 - bg_prob_map))/2 #computing saliency map
saliency_image = np.zeros((image.shape[0], image.shape[1]))
```

```
for i in range(image.shape[0]):
    for j in range(image.shape[1]):
        saliency_image[i,j] = saliency_map[int(np.mean(image[i,j]))]*255

plt.imshow(saliency_image, cmap = 'gray' )
plt.title("Saliency Map")
plt.show()
```



[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: