[ECE 271] Homework-1

Yang Yue
A53301503

Problem 1.5

a) 1) By simply looking at the Cheetah.bmp and estimate the ratio of the area that cheetah covers in the whole picture.

$$P_Y(FG) = \frac{estimated\ area\ of\ cheetah}{estimated\ area\ of\ image}$$

2) By loading the TrainingSampleDCT_8.mat, from which we can possess the size of training samples' matrices of front ground and background. And then the prior probability can be estimated as:

$$P_Y(FG) = \frac{size(TraningSamplesDCT\_FG)}{size(TrainingSampleDCT\_BG) + size(TraningSamplesDCT\_FG)}$$

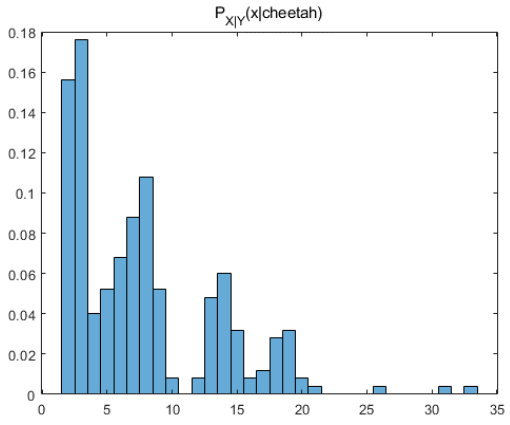
3) By loading the Cheetah_mask.bmp, then transform it from black and white pixels into a matrix consist of 0s and 1s. The prior probability can be written as:

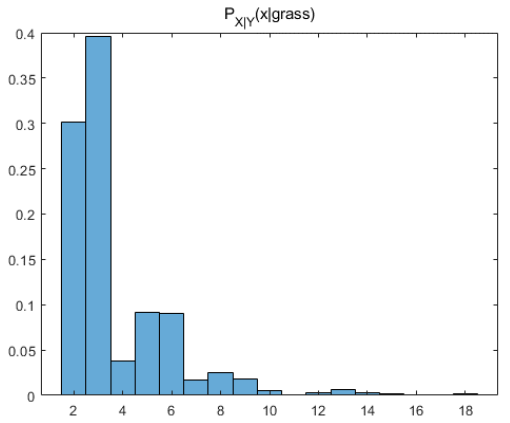
$$P_Y(FG) = \frac{number\ of\ 1}{total\ number\ of\ 1\ and\ 0}$$

All methods mentioned to estimate the prior probability of front ground, the corresponding prior probability of background is simply defined as:

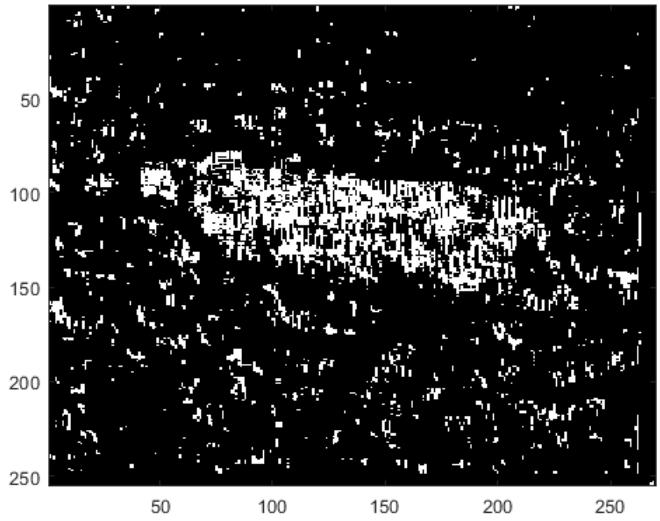
$$P_Y(BG) = 1 - P_Y(FG)$$

b) Index histograms are shown below:





c) The image file generated from array A is shown below:



d) By loading the Cheetah_mask.bmp, we can get an array, each of whose elements represent whether the pixel belongs to the front ground or the background by its 0-1 status.
An error determined by:

$$e_i = \begin{cases} 1, & A_i \neq Mask_i \\ 0, & A_i = Mask_i \end{cases}$$

Then the error rate is calculated as follows:

$$Err = \frac{|\sum A_i - Mask_i|}{number\ of\ pixels} = 17.26\%$$

Appendix

Find2ndx.mat

```
function x = find2ndX(oriMatrix)
    %To extract the second largest number's position --x
    %in TrainingSample_BG/FG
    %First, set the largest element in BG/FG matrices to 0
    oriMatrix(oriMatrix == max(oriMatrix,[],2))=0;
    %second, find the largest element in TSDCT_FG/BG, send their positions
to
    %xFG/BG.
    [~, x] = max(oriMatrix,[],2);
end
```

hw1_5.mat

```
%load training samples and zig-zag order
load('TrainingSamplesDCT_8.mat');
TSDCT_FG = TrainsampleDCT_FG;
TSDCT_BG = TrainsampleDCT_BG;
zigzag = load('Zig-Zag Pattern.txt');
zigzag = reshape(zigzag, 1, []) + 1;

%set prior prob P(X), P(Y)
lenFG = size(TrainsampleDCT_FG, 1);
lenBG = size(TrainsampleDCT_BG, 1);

xFG = find2ndX(TSDCT_FG);
xBG = find2ndX(TSDCT_BG);

%print out histographs of BG/FG
figure(1);
h_fg = histogram(xFG,'Normalization','probability');
title('P_{X|Y}(x|cheetah)');
figure(2);
h_bg = histogram(xBG,'Normalization','probability');
title('P_{X|Y}(x|grass)');

%Pxy_BG <= P(X=xi|Y=BG)
%Pxy_FG <= P(X=xi|Y=FG)
Pxy_BG = zeros([64,1]);
Pxy_FG = zeros([64,1]);
tlb_BG = tabulate(xBG);
tlb_FG = tabulate(xFG);
Pxy_BG(1:size(tlb_BG,1)) = tlb_BG(:,3);
Pxy_FG(1:size(tlb_FG,1)) = tlb_FG(:,3);
```

```
Pxy_BG(Pxy_BG == 0) = 0.0001;

py_BG = lenBG/(lenBG+lenFG);
py_FG = 1-py_BG;
PdP = (Pxy_FG.*(py_FG))./(Pxy_BG.*(py_BG)); %decision boundary

%read the test img cheeta
cheetah_img = imread('cheetah.bmp');
cheetah_dw = im2double(cheetah_img);
%set a blank padding
cheetah_pad = [cheetah_dw, zeros([size(cheetah_dw,1),7]);
zeros([7,size(cheetah_dw,2)+7])];

%initialize test set
test_set = zeros([255*270,64]);
cnt = 1;
%test set implementation

%slice the image into 8*8 blocks and run the dct2()
for col = (1:size(cheetah_pad,2))
    if (col+7) > size(cheetah_pad,2)
        break;
    end
    for row = (1:size(cheetah_pad,1))
        if (row+7) > size(cheetah_pad,1)
            break;
        end
        test_set(cnt,:) =
reshape(abs(dct2(cheetah_pad(row:row+7,col:col+7))),1,[]);
        cnt = cnt + 1;
    end
end

%find the 2nd max value's position and map it with zigzag
testMx2nd = find2ndX(test_set);
test2ndZig = zeros([size(testMx2nd,1),1]);
for i = 1:size(testMx2nd,1)
    test2ndZig(i) = zigzag(testMx2nd(i));
end

%Cheetah or not?
A =
reshape(isCheetah(test2ndZig,PdP),size(cheetah_dw,1),size(cheetah_dw,2));
figure(3);
```

```
imagesc(A);
colormap(gray(255));

%error computation
cheetah_mask = imread("cheetah_mask.bmp");
cheetah_mask_dw = im2double(cheetah_mask);
errorRate = sum(abs(A -
cheetah_mask_dw),'all')/(size(cheetah_mask,1)*size(cheetah_mask,2))
```