

ECE 271 Homework 1

Reya Sadhu

- a) There are two samples given, which consists of the training set of vectors for each of the class i.e cheetah (foreground) and the background. So, it is safe to assume that the no of samples in the training set of cheetah are the no of pixels in the image which forms the cheetah.

```
numCheetahSamples = size(TrainsampleDCT_FG, 1);  
numGrassSamples = size(TrainsampleDCT_BG, 1);
```

```
totalSamples = numCheetahSamples + numGrassSamples;
```

```
PY_cheetah = numCheetahSamples / totalSamples;  
PY_grass = numGrassSamples / totalSamples;
```

So the prior probabilities are,

```
PY(cheetah) = 0.19186  
PY(grass) = 0.80814
```

```
fx >>
```

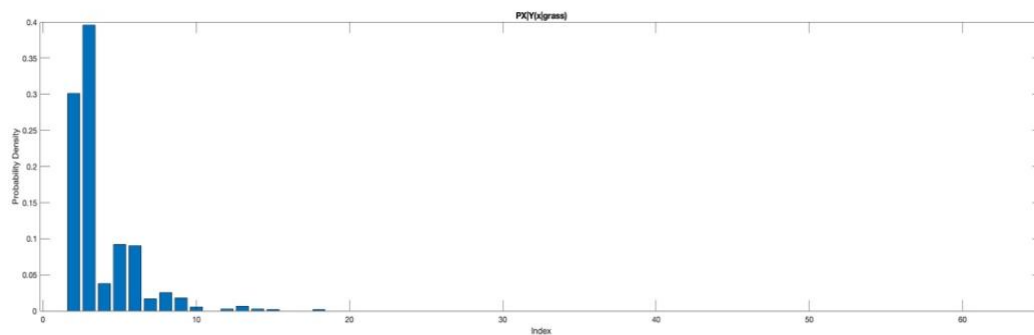
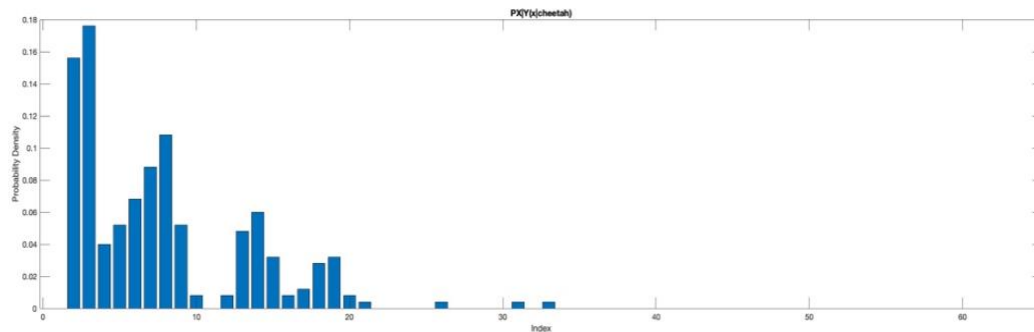
b)

```
cheetahSecondLargestIndices = zeros(size(TrainsampleDCT_FG, 1), 1);
grassSecondLargestIndices = zeros(size(TrainsampleDCT_BG, 1), 1);

for row = 1:size(TrainsampleDCT_FG, 1)
    absValuesf = abs(TrainsampleDCT_FG(row, :));
    [~, sortedIndicesf] = maxk(absValuesf, 2);
    cheetahSecondLargestIndices (row) = sortedIndicesf(2);
end
for row = 1:size(TrainsampleDCT_BG, 1)
    absValuesb = abs(TrainsampleDCT_BG(row, :));
    [~, sortedIndicesb] = maxk(absValuesb, 2);
    grassSecondLargestIndices (row) = sortedIndicesb(2);
end

cheetahHistogram = hist(cheetahSecondLargestIndices, 1:64);
grassHistogram = hist(grassSecondLargestIndices, 1:64);

cheetahPDF = cheetahHistogram / sum(cheetahHistogram);
grassPDF = grassHistogram / sum(grassHistogram);
```



c)

After calculating the dct on 8*8 blocks on the image, we get the index with the second highest energy from the dct coefficients. Using this as the value of x, we use this decision function to estimate if it falls in cheetah class or the background class.

If, $P_{Y|X}(\text{cheetah} | x) > P_{Y|X}(\text{background} | x)$, then we classify it as cheetah, otherwise as a background.

Using BDR, this can be written as,

$$P_{X|Y}(x|\text{cheetah}) P_Y(\text{cheetah}) > P_{X|Y}(x|\text{background}) P_Y(\text{background})$$

Using this,

```
for i = 1:size(I, 1) - 7
    for j = 1:size(I, 2) - 7
        row_start = i;
        row_end = i + 7;
        col_start = j;
        col_end = j + 7;
        block = I(row_start:row_end, col_start:col_end);

        block_dct = dct2(block);
        block_dct_flat = reshape(block_dct.', 1, []);
        [~, index] = maxk(abs(block_dct_flat), 2);

        if PY_cheetah * cheetahPDF(zigzag_flat(index(2))+1) >= PY_grass * grassPDF(zigzag_flat(index(2))+1)
            % Assigning it to the central pixel of the block
            A(i+3,j+3)=1 ;
        else
            A(i+3,j+3)= 0;
        end
    end
end

% Padding the image with zeros
A_resized=zeros(255,270);
for i=5:252
    for j=5:267
        A_resized(i,j)=A(i-4,j-4);
    end
end
```

The mask we generate is,



d) Comparing with the given mask, the probability or error is coming as 16.63%.

```
im_test = imread('/Users/reyasadhu/Downloads/homework1/cheetah_mask.bmp');  
im_test=im2double(im_test);  
err=abs(im_test-A_resized);  
prob_err=sum(err,"all")/(255*270);  
disp(prob_err);
```

MATLAB Code

```
numCheetahSamples = size(TrainsampleDCT_FG, 1);
```

```
numGrassSamples = size(TrainsampleDCT_BG, 1);
```

```
totalSamples = numCheetahSamples + numGrassSamples;
```

```
PY_cheetah = numCheetahSamples / totalSamples;
```

```
PY_grass = numGrassSamples / totalSamples;
```

```
disp(['PY(cheetah) = ' num2str(PY_cheetah)]);
```

```
disp(['PY(grass) = ' num2str(PY_grass)]);
```

```
cheetahSecondLargestIndices = zeros(size(TrainsampleDCT_FG, 1), 1);
```

```
grassSecondLargestIndices = zeros(size(TrainsampleDCT_BG, 1), 1);
```

```
for row = 1:size(TrainsampleDCT_FG, 1)
```

```
    absValuesf = abs(TrainsampleDCT_FG(row, :));
```

```
    [~, sortedIndicesf] = maxk(absValuesf,2);
```

```
    cheetahSecondLargestIndices (row) = sortedIndicesf(2);
```

```
end
```

```
for row = 1:size(TrainsampleDCT_BG, 1)
```

```
    absValuesb = abs(TrainsampleDCT_BG(row, :));
```

```
[~, sortedIndicesb] = maxk(absValuesb, 2);  
grassSecondLargestIndices (row) = sortedIndicesb(2);  
end
```

```
cheetahHistogram = hist(cheetahSecondLargestIndices, 1:64);  
grassHistogram = hist(grassSecondLargestIndices, 1:64);
```

```
cheetahPDF = cheetahHistogram / sum(cheetahHistogram);  
grassPDF = grassHistogram / sum(grassHistogram);
```

```
figure;  
subplot(2, 1, 1);  
bar(1:64, cheetahPDF);  
title('PX|Y(x|cheetah)');  
xlabel('Index');  
ylabel('Probability Density');
```

```
subplot(2, 1, 2);  
bar(1:64, grassPDF);  
title('PX|Y(x|grass)');  
xlabel('Index');  
ylabel('Probability Density');
```

```
zigzag=importdata('/Users/reyasadhu/Downloads/homework1/Zig-Zag Pattern.txt') ;  
zigzag_flat=reshape(zigzag.',1,[])  
disp(zigzag_flat);
```

```
I = imread('/Users/reyasadhu/Downloads/homework1/cheetah.bmp');
```

```
l=im2double(l);
```

```
A = zeros(size(l, 1) - 7,size(l, 2) - 7);
```

```
for i = 1:size(l, 1) - 7
```

```
    for j = 1:size(l, 2) - 7
```

```
        row_start = i;
```

```
        row_end = i + 7;
```

```
        col_start = j;
```

```
        col_end = j + 7;
```

```
        block = l(row_start:row_end, col_start:col_end);
```

```
        block_dct = dct2(block);
```

```
        block_dct_flat = reshape(block_dct.',1,[]);
```

```
        [~, index] = maxk(abs(block_dct_flat), 2);
```

```
        if PY_cheetah * cheetahPDF(zigzag_flat(index(2))+1) >= PY_grass *  
grassPDF(zigzag_flat(index(2))+1)
```

```
            % Assigning it to the central pixel of the block
```

```
            A(i+3,j+3)=1 ;
```

```
        else
```

```
            A(i+3,j+3)= 0;
```

```
        end
```

```
    end
```

```
end
```

```
% Padding the image with zeros
```

```
A_resized=zeros(255,270);
```

```
for i=5:252
```

```
    for j=5:267
```

```
        A_resized(i,j)=A(i-4,j-4);
```

```
    end
```

```
end
```

```
figure;
```

```
imshow(A_resized);
```

```
im_test = imread('/Users/reysadhu/Downloads/homework1/cheetah_mask.bmp');
```

```
im_test=im2double(im_test);
```

```
err=abs(im_test-A_resized);
```

```
prob_err=sum(err, "all")/(255*270);
```

```
disp(prob_err);
```