Homework 0 - Alohomora CMSC 733

Abhilash Pravin Mane
Master of Engineering Robotics
Univeristy of Maryland
College Pqrk, Maryland 20740.
Email: amane@umd.edu
Phase 1

I. Introduction

EDGE detection has been a classical problem in computer vision from a long time. Researcher have been performing edge detection using sobel and Canny edge detector by analyzing intensity discontinuity. The Pb lite outperforms these algorithm as it consider textures, Color and brightness to create pixelwise weightage factor. This algorithm has 4 steps. 1) Generating Filters - DOG, LM, Gabor and Half disc filters. 2) Creating a texton, brightness, color map by clustering. 3) Computing gradients of the maps 4) Boundary detection.

A. Generating Filters

Generation of filters is done to detecting various texture on an image. This is the first step of Pb lite. This step is done to distinguish between edges and texture.

1) Derivative of Gaussian (DOG) Filters): Derivative of gaussian is generally used to detect edges in a given image. Generally this is done by convolving a sobel kernel with gaussian. This resulting kernel is then convolved with the image to find out edges in the images. These generated derivative kernel is then oriented into multiple angle. Angles are at increment of 45 ranging from 0 to 360 of total 8 orientaions. The gaussiasn is co-ccenttric (σ=σ_x= σ_y) with 2 scales i.e. σ=[1,3]. So the total DOG kernel are 8x23 = 16 kernels. The DOG filters can be seen in Fig 1.

2) Leung Malik (LM) Filters: Leung Malik filter are the filters which are deduced to detect textures in the image. They consist of 1st order derivative 2nd order derivative and laplacian of elongated or ellipsoidal gaussain kernel of different scales and orientation. Also the filter consists of symmetric gaussian kernel with different scales. The first order and second order derivative is calculated at elongation of 3 times of the gaussian σ_x = 3σ_y and the σ_x have 4 scales which are 1,$\sqrt{2}$,2 (σ_x=[1,$\sqrt{2}$,2$\sqrt{2}$]). These kernels are then oriented at an angle increment of 30 degrees over 0 to 180. Hence 6 orientation. So the number of kernels are 2x3x6=36. Along with this Leung Malik filter bank consists of coccentric gausssian and laplacian of thee gaussian with scale same as above, total 8 filters. The total number of filters Leung malik are 36+12=48 filters. Filters can be scene in Fig 2.

3) Gabor Filters: Gabor filter are the filters that detect specific frequency in an image. It is similar to how human visual system works. The filter tries to detect if there is any frequency λ in the given image. The filter is then oriented with an angle increment of 45/2 in the range of [0,180] The gaussian scale used for generating these filters is σ is 3,5,10,15. Fig3 depicts Gabor filters.

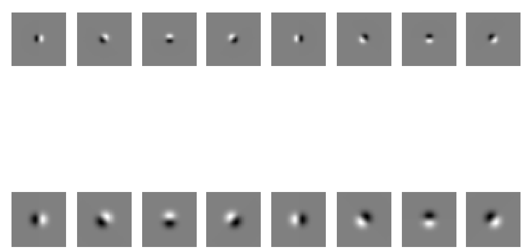

Fig. 1: Derivative of gaussian filter with σ =1,3 and 16 different orientation

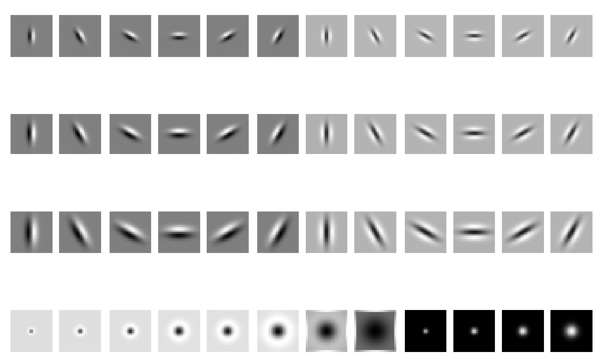

Fig. 2: Leung Malik filters: 1st 6 colums are 1st derivative of elongated gaussian and last 6 columns are 2nd order derivative of elongated gaussian, half of the last row consist of gaussian and laplacian of gaussian,σ =1,$\sqrt{2}$,2,2$\sqrt{2}$‘ .

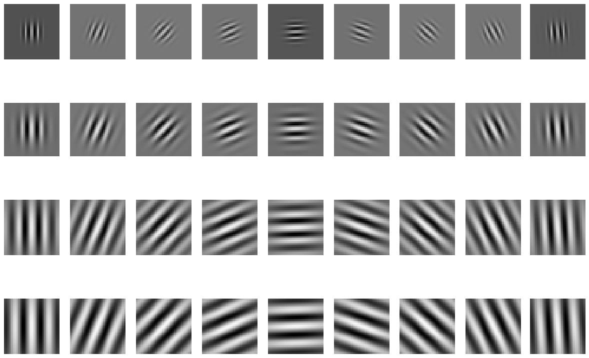

Fig. 3: Gabor filter bank σ=[3,5,10,15] and λ=[3,5,7,9] respectively

B. Creating a texton, brightness, color map by clustering

After generating all the filters the input image is convolved with these filters to obtain a pixel wise filter response. We obtain an N value vector for each pixel. This vector has all the texture information corresponding to the filters. This N dimension vecxtor of images is passed through Kmeans algorithm to cluster which assigns a texton ID for image depending on the vector value. A K clusters are formed where K for texture is 64. This clustering algorithm gives us a texton map. This clustering is done to group together similar objects and reduce the computation.
Similarly, the above proces of clustering is carried out for brightness and color to obtain texton brightness and color map. For generation texton brightness map the image is converted to grayscale and for texton color map image is directly used. The number of cluster for brightness and color are N=16.

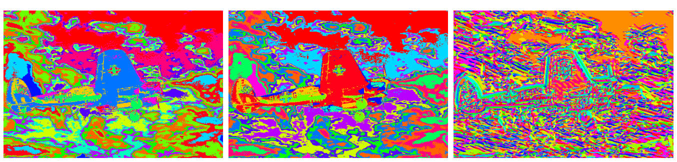

Fig. 4: B, C, T , Image 1

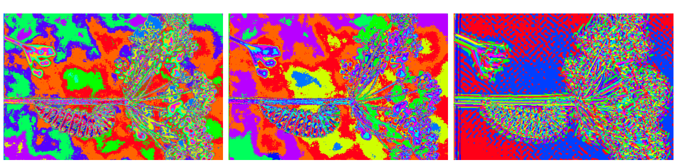

Fig. 5: B, C, T, Image2

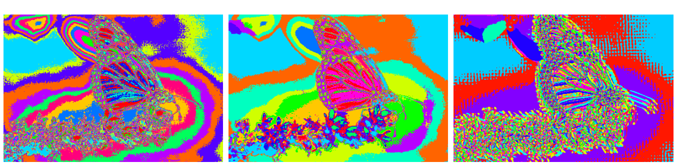

Fig. 6: B, C, T, Image 3

Fig. 7: B, C, T, Image 4

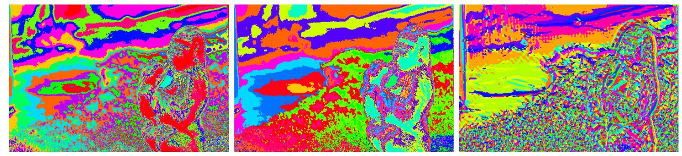

Fig. 8: B, C, T, Image 5

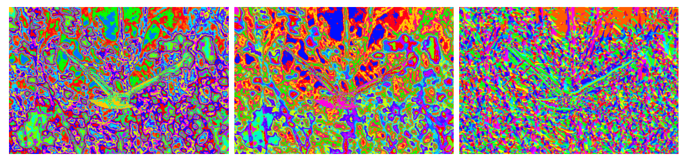

Fig. 9: B, C, T , Image 6

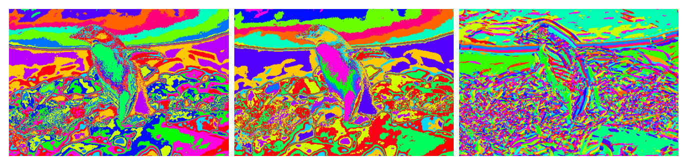

Fig. 10: B, C, T , Image 7

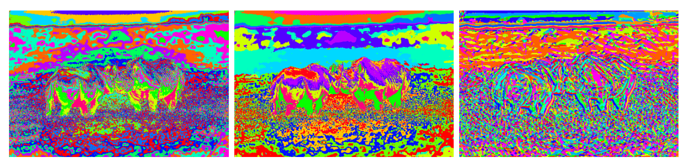

Fig. 11: B, C, T, Image 8

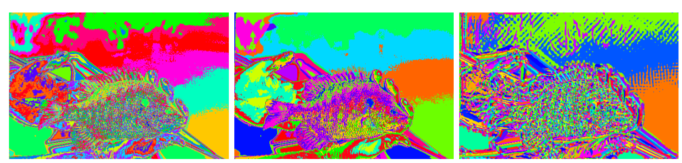

Fig. 12: B, C, T , Image 9

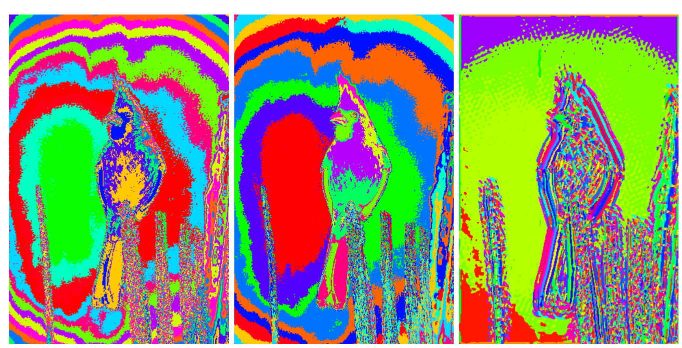

Fig. 13: B, C, T, Image 10

C. Computing gradients of the maps

We calculate the texture, brightness and color gradient using half disc masks.

1) Half disc: Half disk mask is used to decrease the computation time as traditional gradient method(looping over each pixel) has more computational time. For similar distribution the gradient acquired from half disk mask is small. For dissimilar distribution the gradient value is large. Half disc mask consists of pairs of half disc binary images. The half disc are generated by creating a circle using equation of circle. Then the generated circle part is divided into 2 halfs. These 2 halfs are then rotated with angle of 45/2 from 0 to 180 to get 8 orientation. 4 different radius are used to obtain the circles. Total number of filters are 4 x 8 =32 pairs. Refer fig 14 for Half disc diagram.

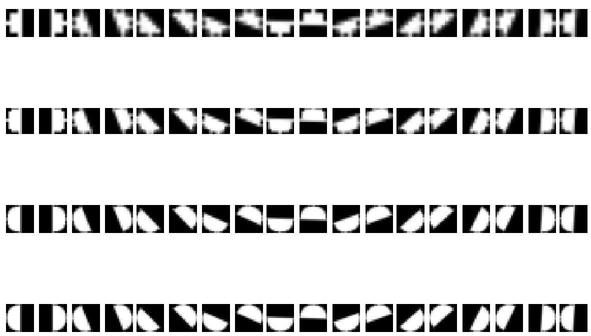

Fig. 14: Half disc with radius = [3,5,10,15] and 16 different orientation

2) chi square distance : χ^2 using the pair of images obtained by convolving half disc pairs. It is generally used to compare 2 histograms.

$$\chi^2(g,h) = \frac{1}{2}\sum_{i=1}^{K}\frac{(g_i - h_i)^2}{g_i + h_i}$$

Here K are the index of the bins. For texton gradient K used is 64 and for color and brightness gradient K is 16.

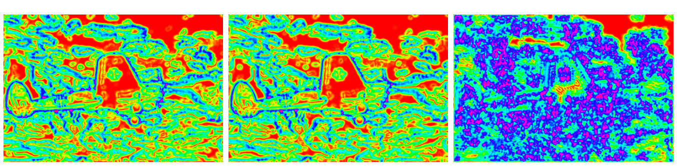

Fig. 15: Bg, Cg, Tg , Image 1

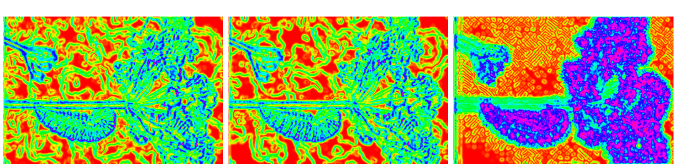

Fig. 16: Bg, Cg, Tg , Image 2

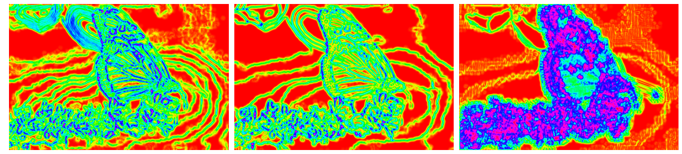

Fig. 17: Bg, Cg, Tg , Image 3

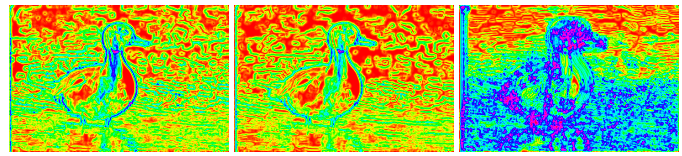

Fig. 18: Bg, Cg, Tg , Image 4

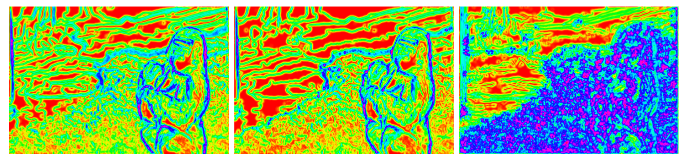

Fig. 19: Bg, Cg, Tg , Image 5

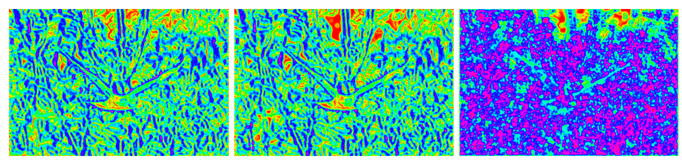

Fig. 20: Bg, Cg, Tg , Image 6

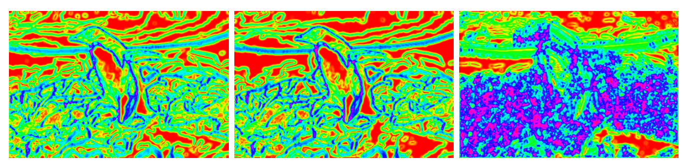

Fig. 21: Bg, Cg, Tg , Image 7

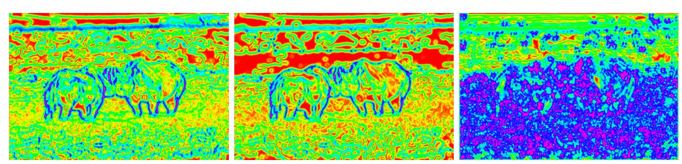

Fig. 22: Bg, Cg, Tg , Image 8

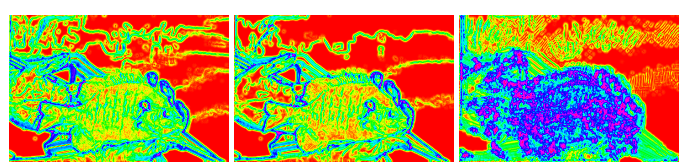

Fig. 23: Bg, Cg, Tg , Image 9

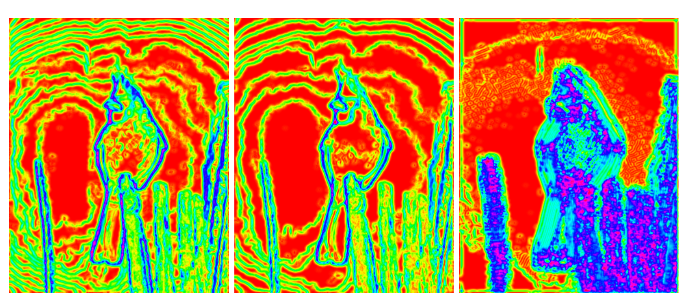

Fig. 24: Bg,Cg,Tg, Image 10

D. Boundary detection

These calculated gradients maps are used to calculate boundary detection by fusing output of canny and sobel.The canny and sobel are combined using the weightage factor. This weight act as percentage of sobel output and canny output that should be contributed in the boundary detection. The sum of this weights should be one. The gradients act as a weight for each pixels with respect to texture, color and intensity changes. The fusion is obtained by taking Hadamard product with the fusion of canny and sobel operator. Hadamard product is nothing but pixel-wise multiplication.The formula is given below.

$$PbEdges = \frac{\mathcal{T}g + \mathcal{B}g + \mathcal{C}g}{3} \bigodot (w1 \times cannyPb + w_2 \times sobelPb)$$

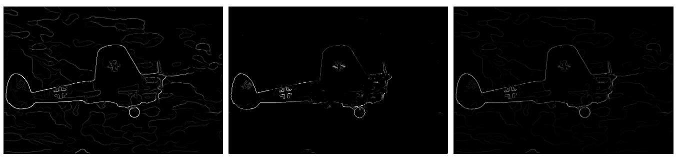

Fig. 25: canny, sobel, Pb-lite , Image 1

Fig. 26: canny, sobel, Pb-lite , Image 2

Fig. 27: canny, sobel, Pb-lite , Image 3

Fig. 28: canny, sobel, Pb-lite , Image 4

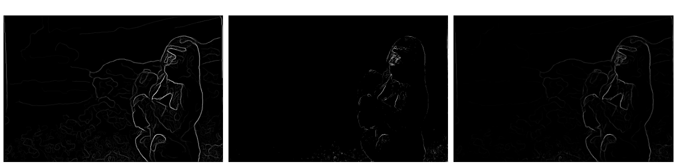

Fig. 29: canny, sobel, Pb-lite , Image 5

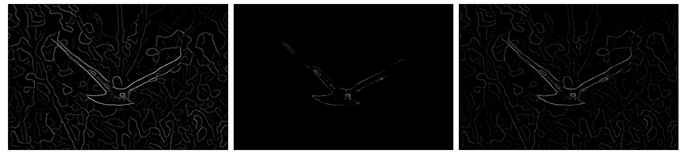

Fig. 30: canny, sobel, Pb-lite , Image 6

Fig. 31: canny, sobel, Pb-lite , Image 7

Fig. 32: canny, sobel, Pb-lite , Image 8

Fig. 33: canny, sobel, Pb-lite , Image 9

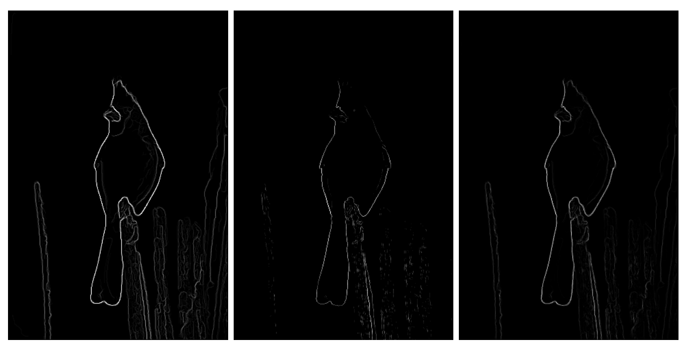

Fig. 34: canny, sobel, Pb-lite , Image 10

E. Analysis

Canny and Sobel does fail when there is lot amount of texture in the image. As it displays edges where there is not actually a edge but texture. Pb lite detects the texture and tries to suppress them. Hence it outperforms canny and sobel edge detectors. As mentioned in above section there is a weight factor for contribution of the sobel and canny this can be calculated by an optimization algorithm. Output can be improved by using large filter banks with different scales. The drawback of Pb lite has high computational time to calculate as compared to canny and edge detectors. So using Pb lite for run time application is not an ideal option. Higher the factor more influence on the image of sobel or Canny.

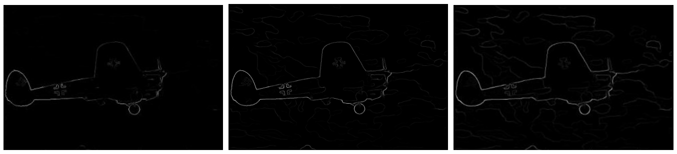

Fig. 35: Figure with different weight-age factor from right to left the influence of canny increases and sobel decreases

REFERENCES

[1] Arbeláez P, Maire M, Fowlkes C, Malik J. Contour detection and hierarchical image segmentation. IEEE Trans Pattern Anal Mach Intell. 2011 May;33(5):898-916. doi: 10.1109/TPAMI.2010.161. PMID: 20733228