Homework 0: Alohomora

Dushyant Patil

I. Introduction

Boundary detection is an important, well-studied computer vision problem. Clearly it would be nice to have algorithms which know where one object transitions to another. But boundary detection from a single image is fundamentally difficult. Determining boundaries could require object-specific reasoning, arguably making the task hard. A simple method to find boundaries is to look for intensity discontinuities in the image, also known of edges. This report shows the steps taken to generate filter banks which is the initial step in the problem of edge detection. The report shows the filter bank generation for Difference of Gaussian (DoG) filters, Leung-Malik (LM) filters, and Gabor filters. The report also states the challenges faced and the next steps for implementing the pb lite boundary detection.

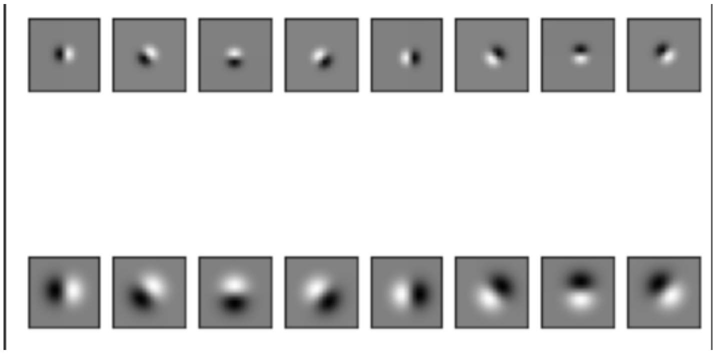

Fig. 1: DoG filter bank

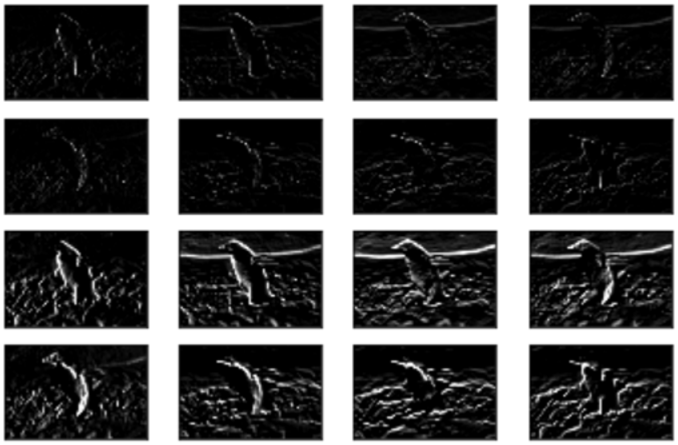

Fig. 2: DoG filter bank output

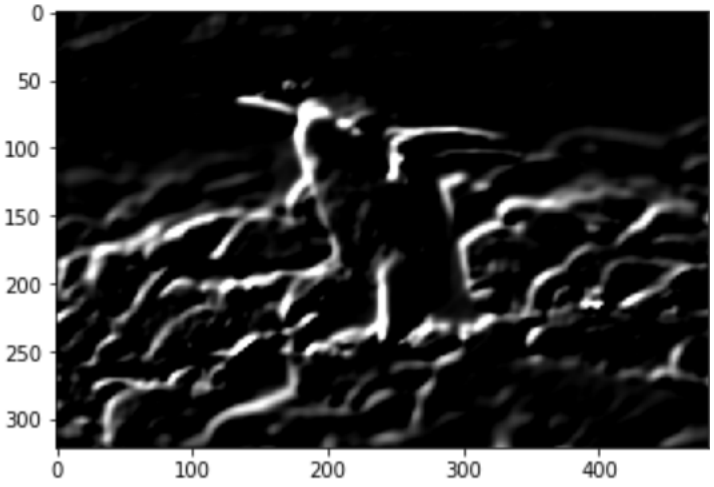

Fig. 3: DoG filter bank output

II. Phase1

The objective of Phase 1 was to implement the pb lite edge detection with the help of Texton map, Color map, Brightness map. The edge detection is implemented in 4 steps:

1) Generate and implement filter banks
2) Implement half-disc masks
3) Create texton, brightness, and color map using kmeans clustering
4) Implement pb lite output using Texton, brightness, color gradients, and Canny, Sobel operators

Filter banks were created using different parameters which include standard deviation, mean, angle, wavelength, and phase for sinusoidal waves. Sobel operators were used for performing discrete differentiation. The filter banks created were as follows:

1) Oriented DoG filter: The filter takes kernel size and standard deviation as an input (mean for uni-variate normal distribution is assumed zero in all filter bank generations). The Gaussian kernel generated for different Std Dev values were convolved with Sobel operator to get first order derivative of the operator. The output was rotated in 8 different direction (in steps of 45 degrees). The equation used for normal distribution is as follow:

$$\phi(x) = 1/\sqrt{[2\pi\sigma^2]} * e^{(}\frac{x^2+y^2}{2\sigma^2})(1)$$

2) Leung-Malik filter: LM filter bank has 48 filters. It consists of first and second order derivatives of Gaussians at 6 orientations and 3 scales making a total of 36; 8 Laplacian of Gaussian (LOG) filters; and 4 Gaussians. We consider two versions of the LM filter bank. In LM Small (LMS), the filters occur at basic scales σ = 1, $\sqrt{2}$, 2, $2\sqrt{2}$. The first and second derivative filters occur

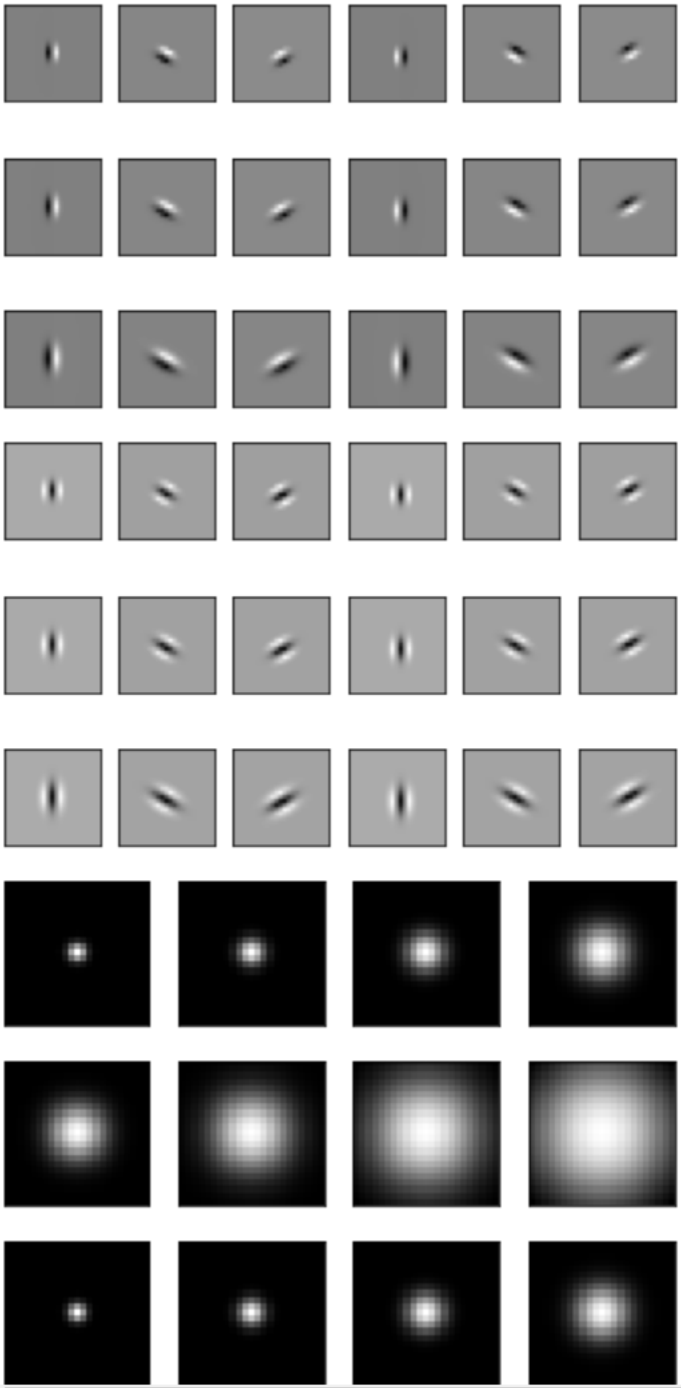

Fig. 4: LM filter bank

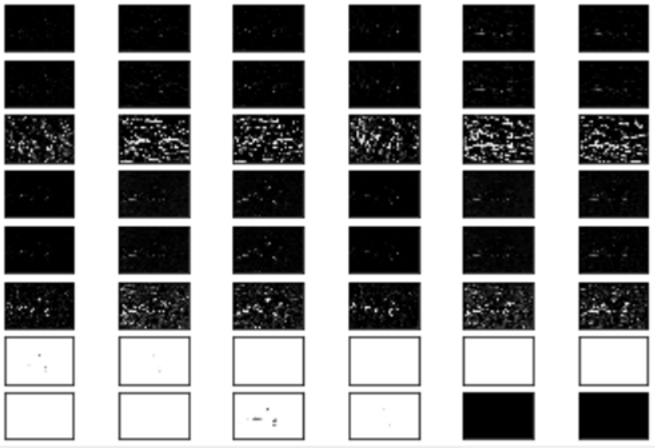

Fig. 5: LM filter bank output

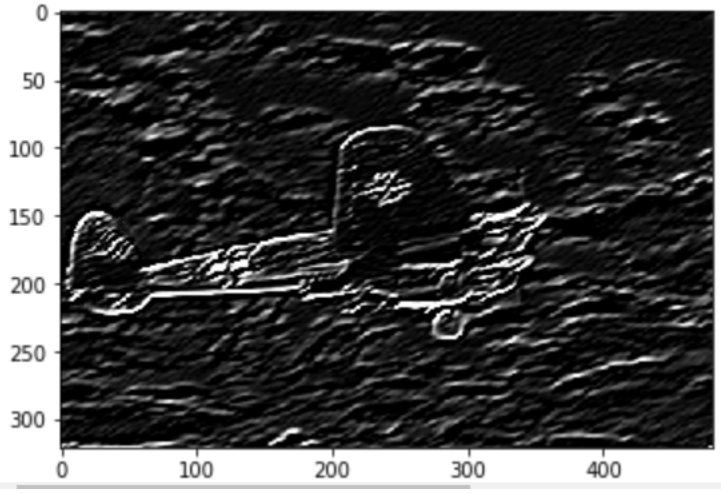

Fig. 6: LM filter bank output

at the first three scales with an elongation factor of 3. 8 LoG filters are created with 2 scales of σ and 3σ. The first and second derivatives were calculated using convolution of Gaussian kernel and Sobel operator

3) Gabor filter: The Gabor filter is Gaussian kernel modulated by a Sinusoidal wave. The filter needs input values of kernel size, elongation factor, phase shift value, wavelength and angle of input for sinusoidal wave, etc. The multiple orientations can be achieved either by changing the input angle value or rotating output of the Gabor kernel generation filter. The equation used for calculating the Gabor kernel was as follows:

$$g(x,y,\lambda,\theta,\phi,\sigma,\gamma)=e^{(}\frac{-x'^2+y'^2}{2\sigma^2})*cos(k) \tag{2}$$

$$k=cos(2\pi/\lambda+\phi) \tag{3}$$

$$x'=xcos(\theta)+ysin(\theta) \tag{4}$$

$$y'=-xsin(\theta)+ycos(\theta) \tag{5}$$

III. Challenges Faced

1. I being new to programming caused me to have slow speed 2. Less knowledge of in built functions 3. Inefficient coding

IV. Next Steps

1. Generate T, C, B maps 2. Create pb lite output 3. Work on phase 2 - deep learning

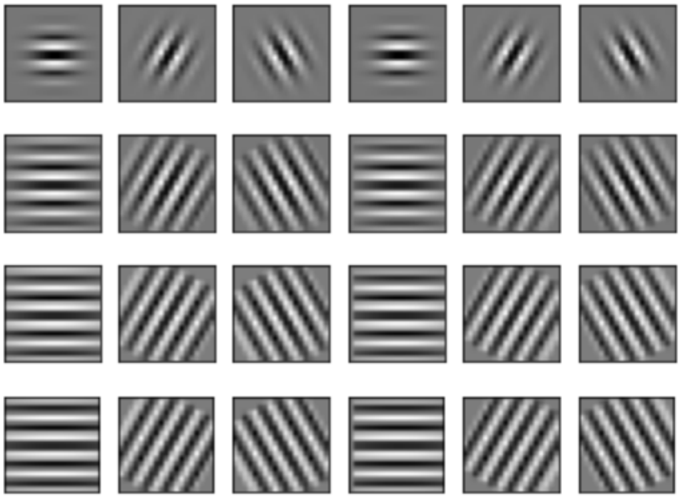

Fig. 7: Gabor bank

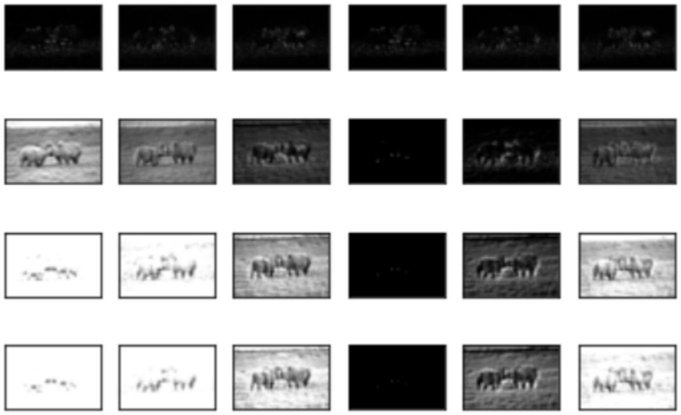

Fig. 8: GB filter bank output

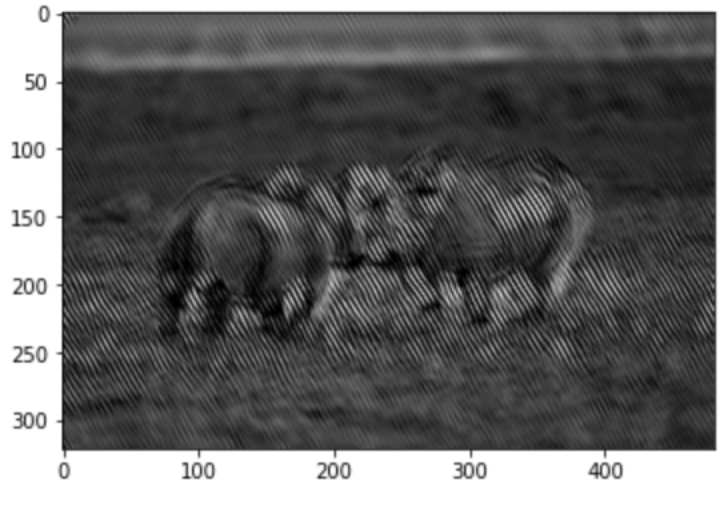

Fig. 9: GB filter bank output