TP4: Gabor Filtering For Feature Extraction

April 1, 2009

1 Introduction

Feature extraction and texture segmentation has received significant attention in recent years. Several filtering systems could be applied for that goal, in this TP we will mainly focus on 2D Gabor filters.

2 Work

2.1 Work to do

You should send as back a report containing the code you wrote, the comparison of the obtained results with the different parameters values as well as your observations. Experiments have to be performed on the zebra image provided. The deadline for sending the report is by Friday 10 April. Afterwards, the solutions are going to be available and no report will be accepted.

2.2 2D Gabor Filters

Two dimensional Gabor filters were initially proposed in [1,2] and are widely used in image processing, computer vision, neuroscience as well as psychophysics. One can define a complex Gabor filter as the product of a Gaussian kernel $g(x, y, \sigma)$ (with σ the scale of the Gaussian envelope), and a complex sinusoid as shown by equation (1);

$$Gabor(x, y) = S(x, y)g(x, y, \sigma) \tag{1}$$

Our complex sinusoid would be written as the following;

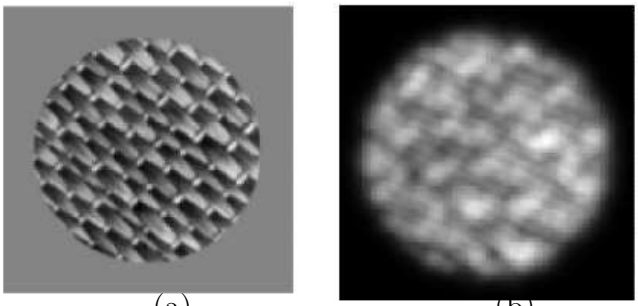
$$S(x,y) = \exp j(2\pi Freq(xcos(\theta) + ysin(\theta)) + \Phi) \tag{2}$$

Indeed we can express the frequency *Freq* as well as the direction θ in spatial coordinates, by defining U and V such as:

$$U = Freq \quad cos(\theta) \tag{3}$$
$$V = Freq \quad sin(\theta) \tag{4}$$

thus the equation 2 could be re-written this way:

$$\Re(S(x,y)) = cos(2\pi(Ux + Vy) + \Phi) \tag{5}$$
$$\Im(S(x,y)) = sin(2\pi(Ux + Vy) + \Phi) \tag{6}$$



(a) (b)

Figure 1: (a) input image (b) gabor filtered image

The spatial gabor filter should be implemented, and the main function will take as an input the original image, the frequency *Freq*, the direction θ. After calculating the product f the Gaussian mask and the sinusoid one, you have to convolve this result with the image to get the filetred one. During this TP you will be asked to work with $\Phi = 0$, and several θ directions in the range $[0, \pi[$, finally for the σ scales it would be between 4 and 10. Your experimental evaluation should be performed on the provided zebra image. You have to vary the several parameters of the gabor filters, in order to

evaluate the results obtained and draw conclusions about the performance of the method. The Gaussian filtering function implemented in the TP2 will be exploited during this TP and will be provided to you. Thus you should check the Gabor mask size and the Gaussian one, they have to be the same. Furthermore, filtered resulting images relative to each parameters should be compared. It is recommended to follow the outlines code provided in the section 3.

2.3 Bonus

A bank of Gabor filters with different chosen orientations and frequencies are applied to an image. One should note that each of these filters results on a two-dimensional array having the same size as the input image. Indeed for feature detection the outputs of the different Gabor filters used in a filter bank for every pixel of the image are combined in a vector that is associated with the relative pixel. Let's point out that the number of the vector components is equal to the number of filters bank. As a bonus you are asked to explore this association (vectors/pixels) so as to segment an image. Therefore you have to gather the vector feature results as explained before, and this for a fixed Gaussian scale width. For each pixel you have to find the strongest response, and then reassemble all the answers to form the segmented image. We will provide you a part of the necessary code with missing part that you have to fill in.

3 Implementation Outlines

- load an image from file,
- specify gabor filter parameters (direction, frequency, sigma),
- display image,
- display filtered image.
- save the result image in a file.

Regarding the implementation as well as the report, we strongly recommend to follow these steps and rename the functions as following;

```
1)GaussianMask(_sigma,  _radius)
2)SinusoidalMasks(freq, theta, phase, cos_mask, sin_mask)
3)GaborFilter( sigma, freq, theta, phase, img)
4)WriteImage(img, filename)
5)GaborSegmentation(scale_fits, n_dirs)
```

References

[1] J.G. Daugman. Uncertainty relations for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America*, 2:1160–1169, 1985.

[2] N. Petkov S.E. Grigorescu and P. Kruizinga. Comparison of texture features based on gabor filters. *IEEE Trans. on Image Processing*, 11(10):1160–1167, 2002.