DIP Homework Assignment #2

姓名：鍾毓安

學號：B01902040

系級：資訊四

Email: [iamyuanchung@gmail.com](mailto:iamyuanchung@gmail.com)

繳交日期：2016.4.12

**Source Code**

The following script and functions were implemented for this assignment. More details of implementations were commented in each corresponding file.

1. README.m: This is the main script that acts like the main function. All the required tasks (with best tuned parameters), including Problem 1, 2 and 3 will be completed one by one when README.m is executed.
2. sobelEdgeDetection.m: Given an image and threshold , this function computes the 1st order gradients by convoluting with Sobel mask and takes those pixels whose gradients are greater or equal to as edges. This function returns the row gradients map , column gradients map , and the final edge map .
3. laplacianOfGaussian.m: Given an image and filter size , this function performs the Laplacian of Gaussian (LoG) technique for detecting edges. Hinted by its name, LoG uses a Gaussian smoothing filter for reducing the noise in before Laplacian is applied. However, this results in two rounds of convolution, including the Gaussian smoothing process and the Laplacian operation, which causes heavy computation. Since the convolution operations are associative, one can convolve the Gaussian smoothing filter with the Laplacian filter first, and then convolve this “hybrid filter” with to achieve the required result. We will refer to this hybrid filter as LoG filter. By using the LoG filter, only one round of convolution operation with the image is needed. Argument is used to determine the size of the LoG filter. This function returns the resulting edge map.
4. cannyEdgeDetection.m: Canny algorithm contains five steps and are too long to describe here, please see the function file for details of implementations.
5. stitchFourFigures.m and warpToGourdShape.m: These two functions are hard-coded and only suitable for taking sample4 ~ sample7 as inputs. The details of stitching and the design of warping functions are described in Problem 2.
6. lawsFeatureExtraction.m
7. classifyPixels.m and kMeansCluster.m
8. attachTexture.m

**Problem 1: Edge Detection**

Three images are given in this problem. For each given image, you are required to generate several edge maps using the following methods. [Please mark the edge points with intensity value 1 and background points with intensity value 0.]

1. Sobel edge detection
2. 2nd order edge detection
3. Canny edge detection

For each method, please apply different parameters and provide some discussions on how they would affect the resultant edge maps. From the observations of your results, list pros and cons of each method, respectively.

**Problem 2: Geometrical Modification**

The goal of this problem is to register the given four images and perform proper geometrical modification on the overlapped square image to obtain a desired shape.

1. Please stich these four images into one complete image and paint the residual regions in black. Denote the result as R.
2. Crop the largest square image of image R and denote it as S. (Hint: the size of S is 512 x 512.)
3. Segment the image S into three parts with predefined size and design three warping functions to convert the image to a gourd-shaped image. Output the result as G.

**Problem 3: Texture Analysis**

The attached figure demonstrates a gray-level image which is compose of several animals with different texture patterns.

1. Perform Law’s method on the given image to obtain the feature vector of each pixel.
2. Use k-means to classify each pixel and label same kind of texture with same intensity. Please specify the intensity levels you adopt and output the result as L.
3. Based on image L, try to attach the correct texture to each animal as best as you can. Output the result as C.

Please provide the details of each step and discussions for each part in the report.