

**DMET 901: Assignment #3**  
**Due: Sunday, December 21<sup>st</sup>, 2008 (11:59 PM)**

**Important:**

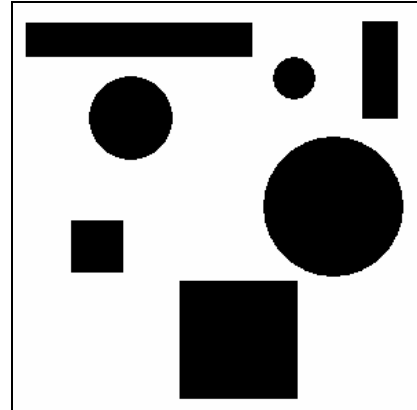
☞ The assignment is to be solved **individually**. You may discuss your work with your colleagues but you have to submit your own copy. Cheating cases will be dealt with firmly.

**Question 1:**

Create a dialog-based application that uses OpenCV to implement the Canny edge detector. The user must be able to enter 3 parameters as inputs to the algorithm. Use a gray scale image of your choice. Compare your result with the result of applying the OpenCV built-in function **cvSobel**.

## Question 2:

Create a dialog-based application that allows the user to enter a binary image file name. Your application should use OpenCV to read and display the binary image. Apply the Connected Components Labeling algorithm discussed in class using 8-connectivity to detect connected black pixels. For simplicity, assume that the connected black components form circles, squares or rectangles only (i.e., no other shapes). Your application should determine the number of each of these objects their locations and sizes. The output should be a text file.







For the **test.bmp** image provided on our Web site, your output file should look like the following:

```
2 Rectangles [x, y, sideX, sideY]
10, 15, 164, 24
255, 14, 25, 70

2 Squares [x, y, side]
43, 159, 37
122, 203, 85

3 Circles: [x, y, radius]
205, 55, 15
86, 84, 30
233, 148, 50
```

-  Assume that all rectangle and square sides are parallel to the x- and y-axes.
-  A rectangle is represented by its upper left corner and the lengths of horizontal and vertical sides.
-  A square is represented by its upper left corner and its side length.
-  A circle is represented by its center and radius.

## Submission:

1. Submit a typed report, written as an MS Word file (or in LaTeX), including all results along with applicable theory and discussion including sample images and snapshots for the dialog.
2. Your code completed with useful comments.