Assignments are due at the beginning of the due date. You should submit a printed hard copy of the report. Please upload the report and all the source code and executable on the class canvas website. In the report, you should explain your methods, some illustration figures of the results, as well as any remaining issues and potential solutions.

**Assignment 3: Canny Edge Detector (Due November 07, 2017)**

Implement the Canny edge detection algorithm. This consists of three phases:

* **Filtered gradient:**
  1. Load an image
  2. Convolve the image with a Gaussian
  3. Find the x and y components of the gradient Fx and Fy at each point
  4. Compute the edge strength F (the magnitude of the gradient) and edge orientation D = arctan(Fy/Fx) at each pixel

Recall that steps 2 and 3 may be combined by convolving the image with the derivative of a Gaussian ("help conv2").

* **Nonmaximum suppression:**  
  Create a "thinned edge image" I(x,y) as follows:
  1. For each pixel find the direction D\* in (0, 45, 90, 135) that is closest to the orientation D at that pixel.
  2. If the edge strength F(x,y) is smaller than at least one of its neighbors along D\*, set I(x,y) = 0, else set I(x,y) = F(x,y)
* **Hysteresis thresholding:**  
  Repeatedly do the following:
  1. Locate the next unvisited pixel (x,y) such that I(x,y) > T\_h
  2. Starting from (x,y), follow the chain of connected local maxima, in both directions, as long as I(x,y) > T\_l
  3. Mark each pixel as it is visited

Test your alogrithm on images of your choosing, experimenting with different values of the parameters sigma (the width of the Gaussian), T\_h (the "high" threshold), and T\_l (the "low" threshold). Also run your algorithm on the following images: 1) the [lena](https://en.wikipedia.org/wiki/Lenna) image; 2) [example images](http://www.imageprocessingplace.com/downloads_V3/dip3e_downloads/dip3e_book_images/DIP3E_CH10_Original_Images.zip) of chapter 10 (Figure 1001, and Figure 1016).