Manipulating images

Introduction

- In this exercise you will manipulate an image using Python, Numpy and Matplotlib.
- You should try to conform your code to the PEP 8 style guide and try to encapsulate the code for these exercises in functions so that the code is not one long blob of code.
- Cite the documentation you used in comments in your code.

Steps

1. Choose a color .PNG image to work with.
2. Use either the Python Imaging Library (PIL) or Matplotlib to load the image, and then turn it into an array.
3. (For the report): what is the the shape of the array?
4. Using `pyplot.imshow` and `pyplot.subplots` create a figure showing the red, green and blue channels individually. Add this figure to your report and hand in the code you used to create the figure (including the image loading).
5. Create a grayscale image out of your full color image using the techniques mentioned on http://www.johndcook.com/blog/2009/08/24/algorithms-convert-color-grayscale/
   (implemented using Numpy). Create a figure for your report and hand in the code you used.
6. Using a Sobel filter detect the edges in the image. See the Wikipedia page for this process: http://en.wikipedia.org/wiki/Edge_detection you can implement this using a 2d convolution of the Sobel filter with the image (convolutions are part of Numpy and Scipy). Image to be included in your report, and code handed in separately.
7. Now you will create a blurred version of the original image by using convolving the 2d image array (on a per color channel basis) with a small "Gaussian kernel". See Wikipedia: http://en.wikipedia.org/wiki/Gaussian_blur Image in report, code handed in separately.
8. As an little extra, try to sharpen your image using the Unsharp Mask technique: http://en.wikipedia.org/wiki/Unsharp_masking Image in report, code handed in separately.