COC202 Computer Vision
Lab 1 – Getting started with Matlab

In the lab sessions, we will use Matlab, a matrix-based programming language, which is well suited for image processing/computer vision tasks. While Matlab will likely constitute a new programming language for you, the labs are not designed to actually teach you Matlab. Rather, assuming you "know how to program" and that the learning of a new language is hence mainly syntax, you are expected, following a brief introduction in this lab, to "pick things up on the way" and, if necessary, consult the (extensive) built-in help, searching the web etc.

Note that you can install Matlab on your own computer; go to

http://www.lboro.ac.uk/services/it/student/software/

for more information.

1. Follow the tutorial on

   http://www.mathworks.co.uk/help/matlab/getting-started-with-matlab.html

   which covers the basics of Matlab and Matlab programming. Make sure to go through the exercises in there in detail, so that you understand how Matlab differs from other programming languages (in particular, that virtually everything in Matlab is matrix-based and hence programming fundamentally involves operations on matrices rather than cycling individually through its elements using loops).

2. Now, let's get started with some simple image processing. Download the *Lenna* image from learn, use the `imread()` function to read it into Matlab, inspect how it is internally stored in Matlab, and use the `imshow()` function to display it on screen.

3. Run the `impixelinfo()` function which will allow you to inspect the actual pixel values in the image at the location of the mouse pointer. Not surprisingly, since we have a colour image, you will see three values for each pixel (for the red, green, and blue channel).

   Create three grayscale images that contain the red, green and blue colour channels of the *Lenna* image (for this, you will need to extract the appropriate parts of the *N*x*M*x*3* image matrix and assign them to new variables). Display the three channel images.

4. Use the `size()` function to obtain the dimensions of the image (matrix). Now, through appropriate addressing, create another image that crops the *Lenna* image by 50 pixels from each border, and display it.

5. Based on 4. write a Matlab function that takes two parameters, an image and a number *n*, and returns the image cropped by *n* pixels from all borders. Test the function.

Once you have finished all exercises you may leave the lab.