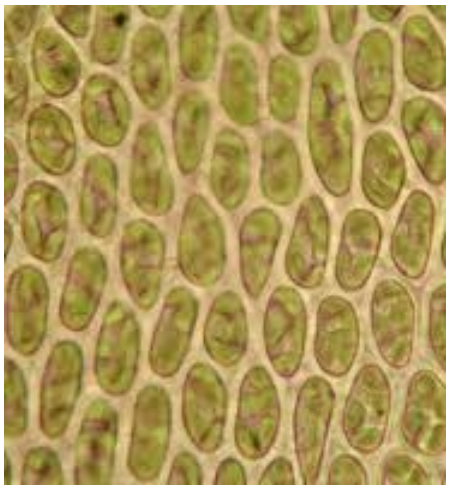
Exercise 1: Region labelling and ellipse fits



bladcellen.png

For the image above (original file on Blackboard: bladcellen.png):

a) Write a program to separate the cells from the background using Otsu's method, and calculate the number of cells using a 'two pass' or 'flood fill' algorithm.

b) For each cell which is NOT touching the border of the picture, calculate:

   I. Area (number of pixels) of the cell

   II. Center of mass of the cell

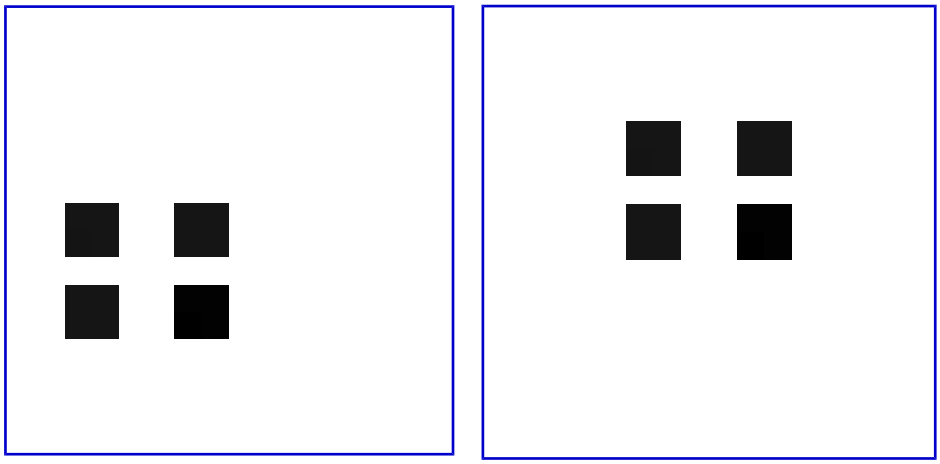
   III. Length of major and minor axes of fitted ellipse

   IV. Eccentricity of fitted ellipse

   Plot the fitted ellipses with the original picture in the background. List all found values in a table.

NB: The Flood Fill algorithm was not explained during the lecture. For the adventurous, you can find more info about this algorithm on Wikipedia: http://en.wikipedia.org/wiki/Flood_fill.

Exercise 2: Cross-correlation and displacement



pattern1.tif **pattern2.tif**

Pictures 'pattern1.tif' and 'pattern2.tif' represent images at times (t) and (t+Δt), respectively.

Load both pictures into Matlab, and write a program to:

a) Calculate the cross-correlation between:

   I. pattern1 and itself,

   II. pattern2 and itself.

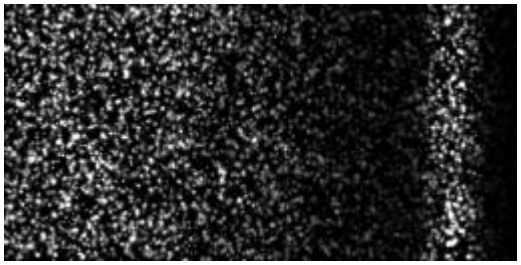
   (N.B.: you are supposed to write your own cross-correlation function here; use of built-in correlation functions is not allowed)

b) Plot the cross-correlation outputs as a 3D surface, and explain the obtained result.

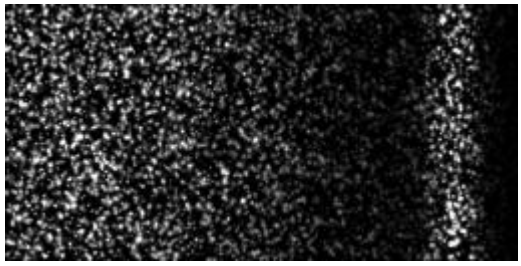
c) Calculate the cross-correlation between pattern1 and pattern2, plot again as a 3D surface and explain the result.

d) From the result of c), calculate the displacement between both images.

Exercise 3: Particle Image Velocimetry [For grades up to 9 and 10]



E001_1.tif



E001_2.tif

Images 'E001_1.tif' and 'E001_2.tif' represent two snap shots from a shear sliding experiment of granular materials at time (t) and (t+Δt), respectively.

a) Eliminate the effects of inhomogeneous lighting by applying Adaptive Histogram Equalization.

b) Divide both images into sub-images of equal size – for example, 32x32 or 16x16 pixels.

c) Use the cross-correlation function written in Exercise 2 to calculate the displacement within each sub-image. Apply the technique explained in the lecture to obtain sub-pixel accuracy.

d) Visualize the obtained velocity field by drawing arrows, e.g. by using quiver().