EPPN Summer School Image Analysis Practical – Segmentation

What we will do:

- Look at different kinds of pixel, edge and region-based segmentation methods
    - Thresholding
    - Canny
    - Watersheds
- Blob counting
- Batch macros
-

Pixel-based segmentation: thresholding

Load practical2a.jpg

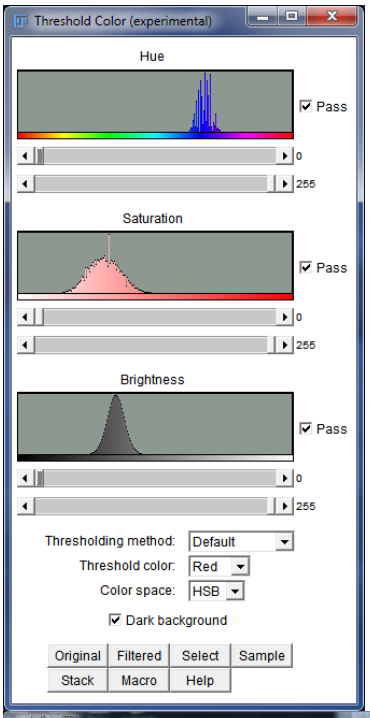
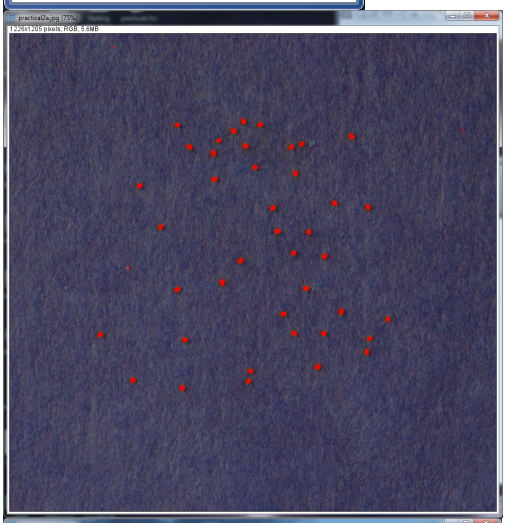
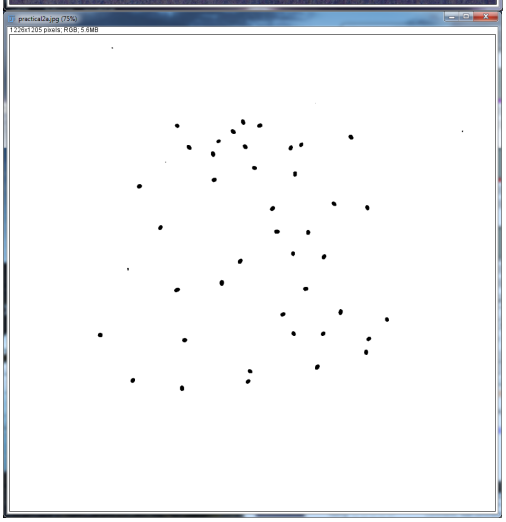
Open the colour thresholding toolbox	Image→Adjust→Color Threshold…	
Threshold the image to segment the seeds.	Trying using first only the Hue (color) settings, and then Brightness	
Create a binary image	Press Select, and then Filtered	
Invert the image	Make the black blobs white (more intuitive), using Edit→Invert, then make the	

image greyscale (we don't need colour now)
(Image→Type→8bit)

Advanced:

1. Use the Fiji object counter to count the number of seeds (Analyze→3d Object Counter). Set the options for what is measured in Analyze→3D OC options.

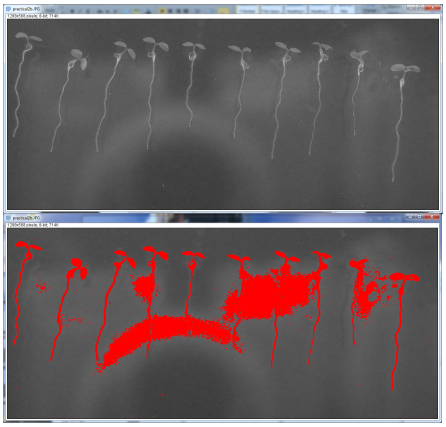
2. Write a macro to do the object counting. Use Process→Batch→Macro to run this macro on all the images in the "5 images" folder and save the result images to another folder.

Edge-based segmentation: Canny-Deriche

This practical uses this plugin

http://imagejdocu.tudor.lu/doku.php?id=plugin:filter:edge_detection:start

Load practical2b.jpg



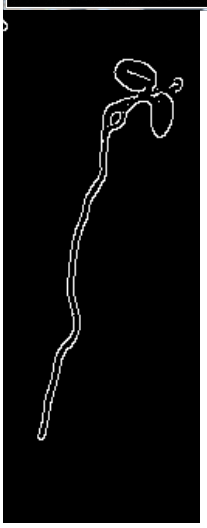
Try thresholding the image to segment the roots

Note how the background reflection causes problems

Reload the original image.
In the next few steps, we'll try replicating a Canny-Deriche edge detector, the main steps of which are:
1. Smoothing
2. Calulate image gradients
3. Non maximal suppression
4. Thresholding with hysteresis

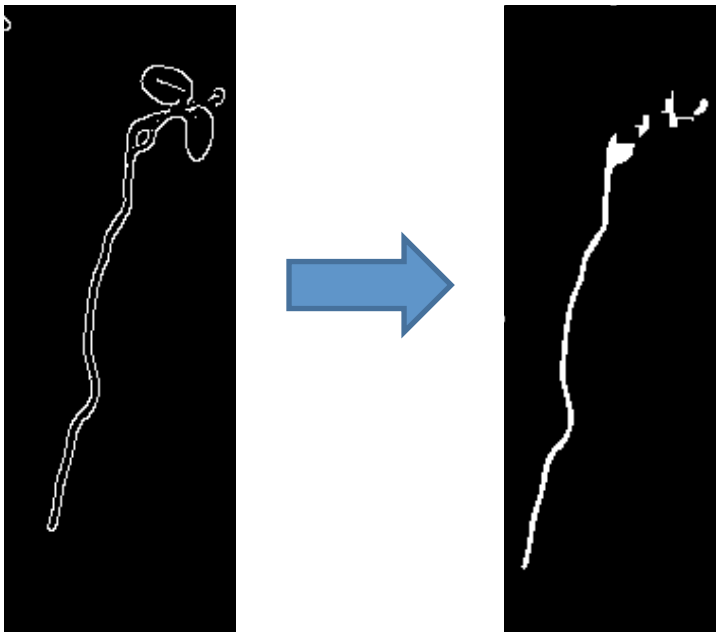
Perform a median filter

Process→Filters→Median
Suggest radius of 1 or 3 for these images, but feel free to play around

Calculate image gradients using the Deriche approach	Plugins→Image Edge→Deriche… Alpha = 0.5. This also performs non-maximal suppression.	
Threshold with Hysteresis	Do this to the maximally suppressed image. Plugins→Image Edge→Hysteresis.. High threshold: ? Low Threshold:? Try setting the low and high thresholds by examining the maximally suppressed output image	

<u>Advanced:</u>

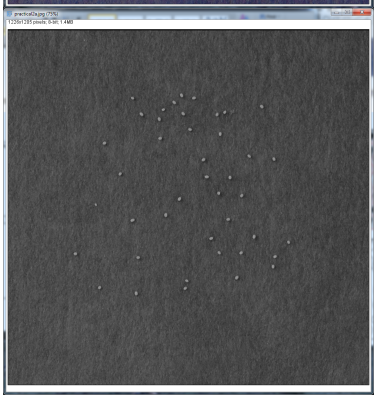
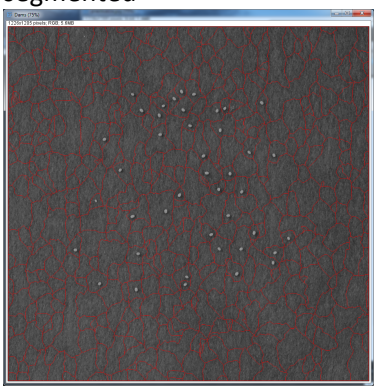
Notice how the edge segmentation truly identifies the edges of the root: what we ideally want is a midline. One way to try and find this is using *morphological operators.* You can find these in the Process→ Binary menu. Try and process the edge image to produce a segmentation of the root itself (don't worry about the leaves):



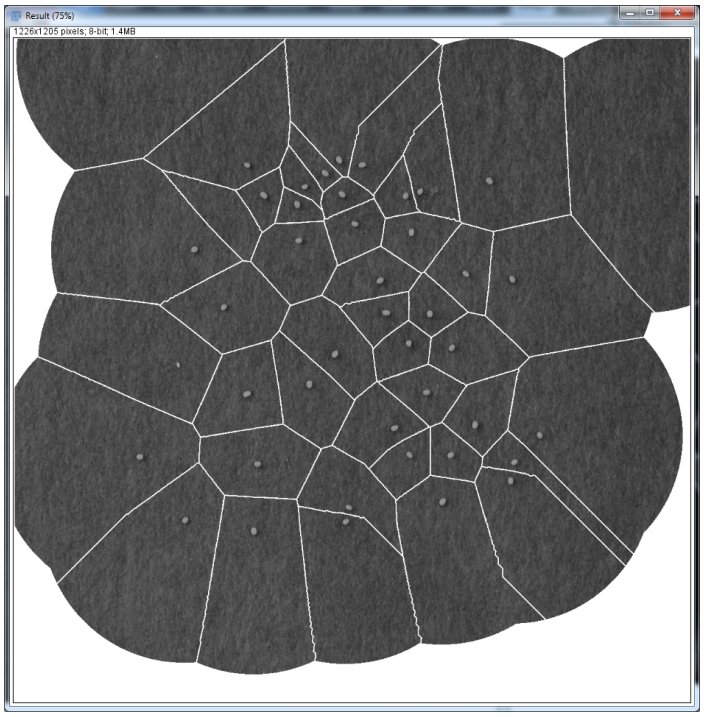
Region-based segmentation: Watershed

This practical uses Daniel Sage's Watershed plugin

http://bigwww.epfl.ch/sage/soft/watershed/

Load practical 2a.jpg		
Make greyscale	Image→Type→8-bit	
Try a watershed segmentation	Plugins→Watershed→Watershed segmentation. Try the default settings, but change to Bright Objects/Dark Background. Select Overlaid Dams for display over the original image.	Notice the amount of oversegmentation due to the texture on the background paper.
Close the output image, and run watersheds again	This time first try blurring the image using the Gaussian blur option. Try different radii. Re-run the watershed algorithm.	The result is still over-segmented 

Watershed segmentation may not be suited here because of the over-segmentation present due to the highly-textured background. However, there is a way to use watershed segmentation in combination with the threholded segmentation we looked at earlier to divide the image into regions defined by distance to the seeds. This is the result we want to achieve:



Try to replicate this. You will need some or all of these functions

-the original threshold method we tried earlier

-creating a 'distance map' (Process→Binary→Distance map). This gives each pixel an intensity equivalent to its distance to the nearest 255 (White) pixel

-using the Watershed plugin

-Combining two images (using the image calculator in the Process menu?) to create the final output image.