**Introduction to Image Processing (COMP2032) Coursework Report**

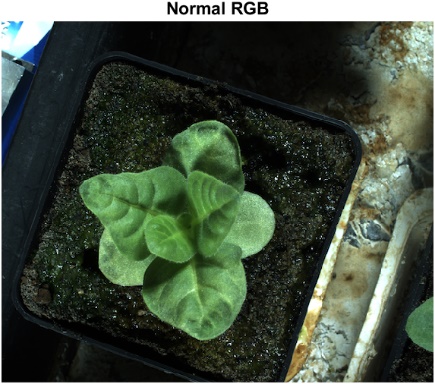
|  |  |
| --- | --- |
| Objective | Detect and Segment Leaf |
| Date | 24/03/2022 |
| Student Name | Tan Zhun Xian |
| Module Convenor | Tissa Chandesa |

**Instructions**

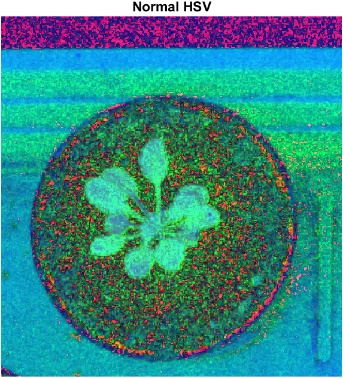
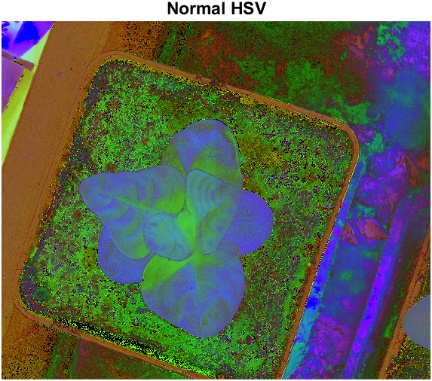
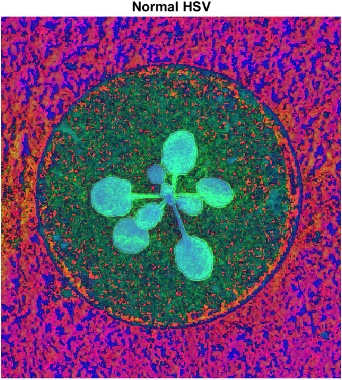
* *The filename for the Matlab file is* *Coursework.m.*
* *To run the file, just type the word ‘Coursework’ in the command window.*
* *Matlab will then prompt user for the image name.*
* *Make sure the images are in the same folder as Coursework.m.*
* *Users must enter full file name including extension (.png) to load the image.*
* *Example: plant001.png*
* *The program will return an image where each leaf is coloured differently on a black background.*

**Introduction**

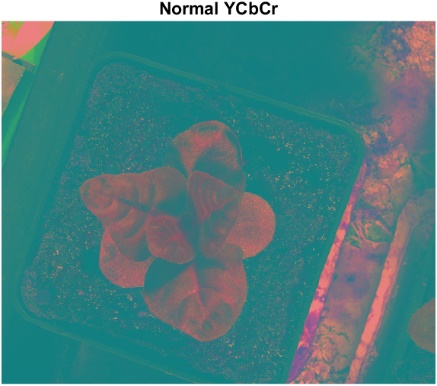
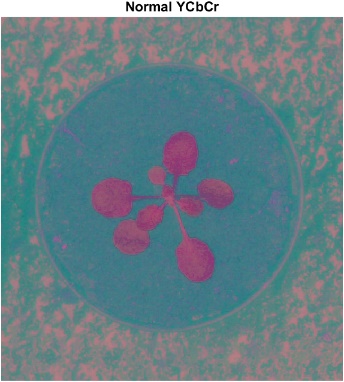
* *The images in the document will be displayed in rows of 3.*
* *The images are arranged in the order plant001.png, plant002.png and plant003.png.*
* *Below are the images used displayed in RGB, HSV and YCbCr.*
  + *RGB*

**

* *HSV*

**

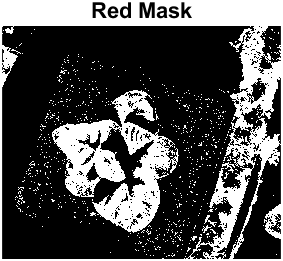
* *YCbCr*

**

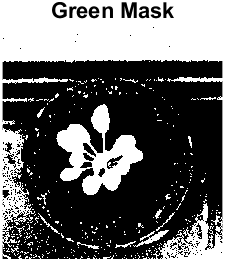
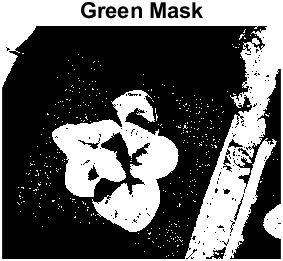
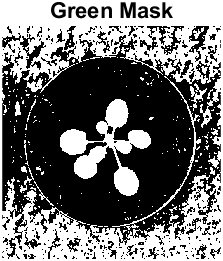
**Pre-processing technique(s)**

**Region of Interest (ROI) Masking**

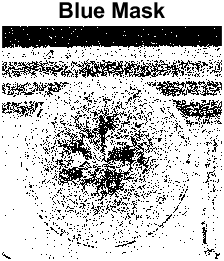
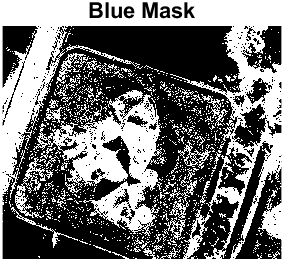
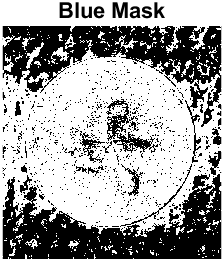
* *Using the RGB, HSV and YCbCr colour spaces, we can construct some masks to only cover the green leaves.*
* *First, we have the RGB masks.*
  + *Red Mask (red > 70 & red < 140)*

**

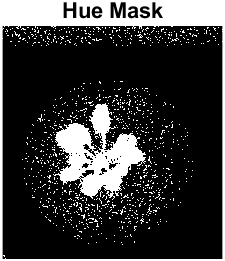
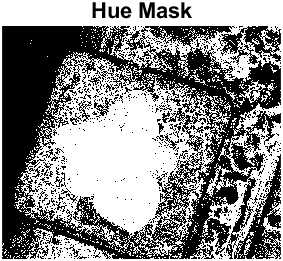
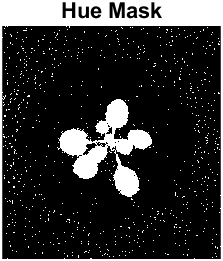
* *Although the plant can be seen in the first image, the plant cannot be clearly seen in the second image while the third image has large pieces of the plant missing. So, this mask is discarded.*
  + *Green Mask (green > 100)*

**

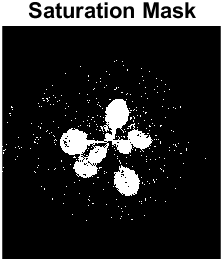
* + *Although the plant can clearly be seen in the first and second image, the third image has large pieces of the plant missing. So, this mask is discarded.*
  + *Blue Mask (blue > 30 & blue < 100)*

**

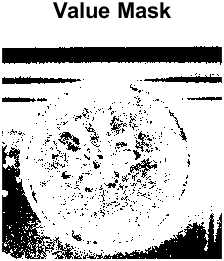
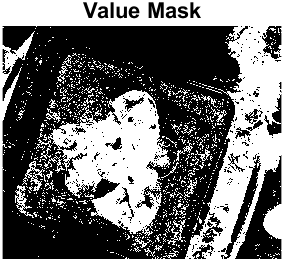
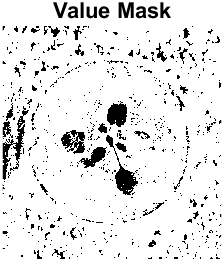
* *The plant cannot be seen clearly in all 3 images. So, this mask is discarded.*
* *Then, we have the HSV masks.*
  + *Hue Mask (hue >= 0.2 & hue <= 0.35)*

**

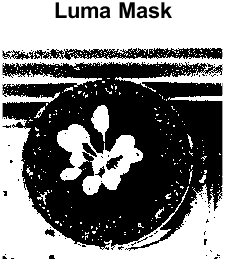
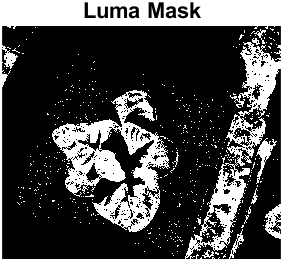
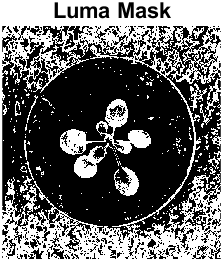
* *Although the main plant can clearly be seen in all 3 images. The side plant to the right in the third image cannot be seen clearly. So, this mask is discarded.*
  + *Saturation Mask (saturation >= 0.6)*

**

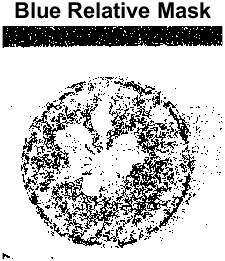
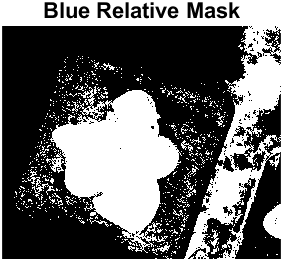
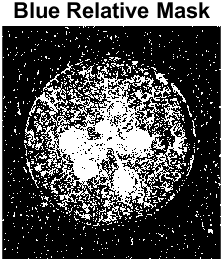
* + *The plant in the first and second image can be seen quite clearly. However, the plant in the third image is almost fully blocked. So, this mask is discarded.*
  + *Value Mask (value >= 0.2 & value <= 0.7)*

**

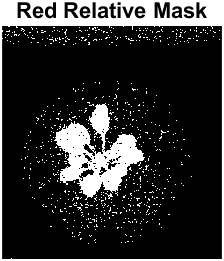
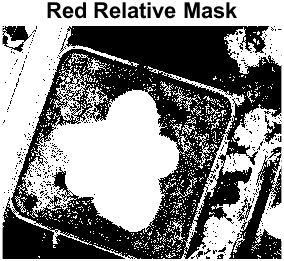
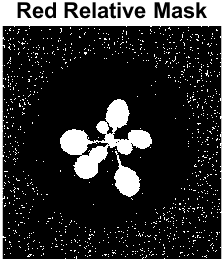
* *The plants cannot be seen clearly in any of the 3 images. So, this mask is discarded.*
* *Then, we have the YCbCr masks.*
  + *Luma Mask (luma >= 100 & luma <= 150)*

**

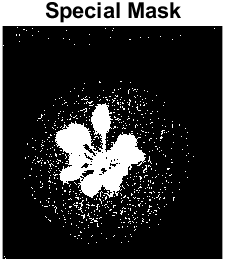
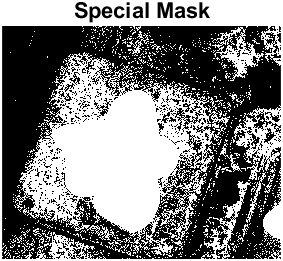
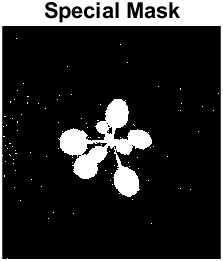
* *Although the plant in the image can be seen clearly, the other 2 images have large pieces of the plant missing. So, the mask is discarded.*
  + *Blue Relative Mask (**blueRelative >= 60 & blueRelative <= 121)*

**

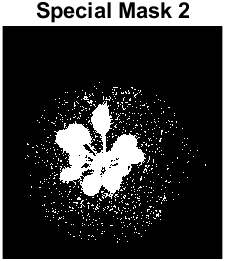
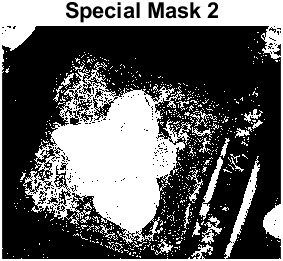
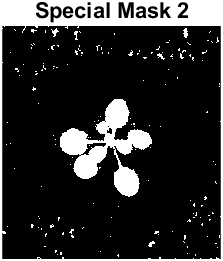
* + *The plants in all 3 images can be seen. Although there is a lot of noise in the first 2 images, it will be removed by the other masks. So, this mask is chosen.*
  + *Red Relative Mask (**redRelative >= 100 & redRelative <= 125)*

**

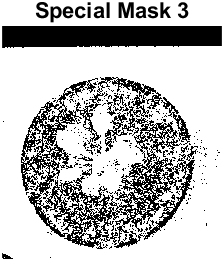
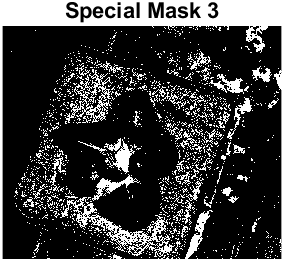
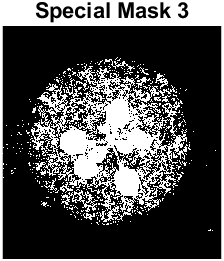
* + *The plant in all 3 images can be seen clearly. Although there is a lot of noise in the third image, it will be removed by the other masks. So, this mask is chosen.*
* *Finally, we have the special masks.*
  + *Special Mask 1 (**green > 1.1 \* red & green > 1.1 \* blue)*

**

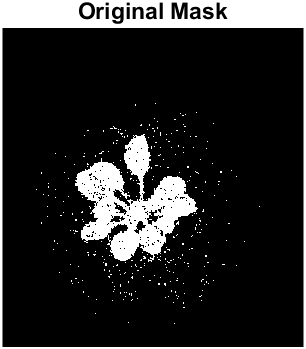
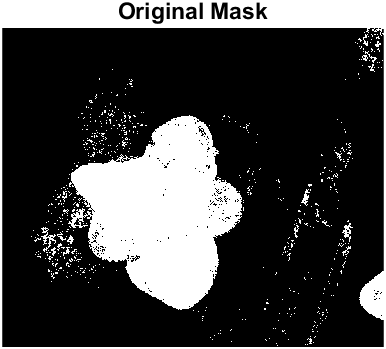
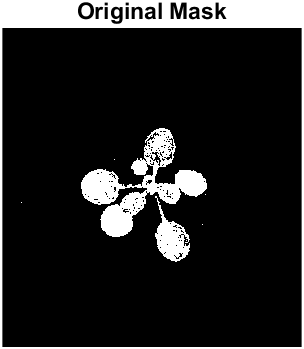
* + *The plant in all 3 images can be seen clearly. Although there is a lot of noise in the third image, it will be removed by the other masks. So, this mask is chosen.*
  + *Special Mask 2 (**green > (red + blue) / 1.4)*

**

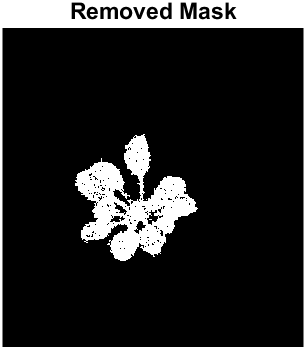
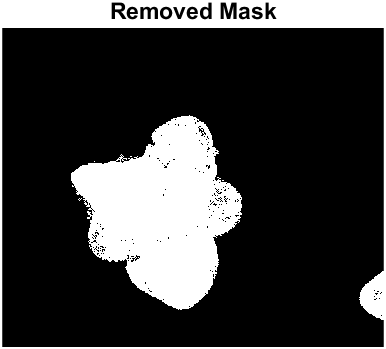
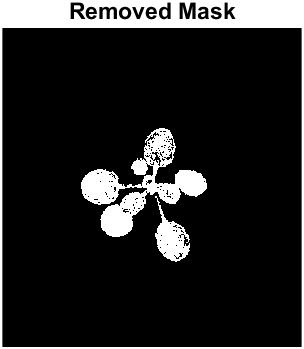
* + *The plant in all 3 images can be seen clearly. Although there is a lot of noise in the third image, it will be removed by the other masks. So, this mask is chosen.*
  + *Special Mask 3 (red > 1.5 \* blue)*

**

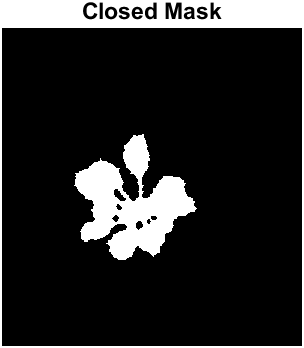
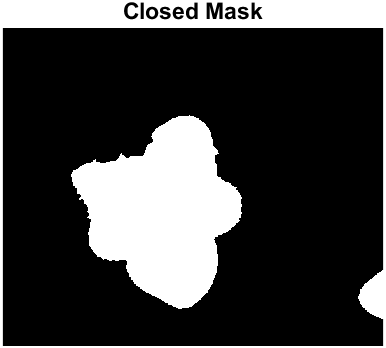
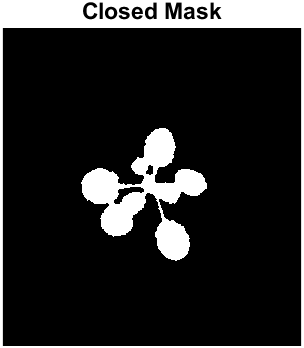
* *The plants in the first 2 images can be seen. However, the plant in the third images is almost fully blocked. So, this mask is discarded.*
* *In the end, the combination of Special Mask 1, Special Mask 2, Blue Relative Mask and Red Relative Mask was found to produce the best mask for separating the leaves from the original image.*
  + *Combined Mask*

**

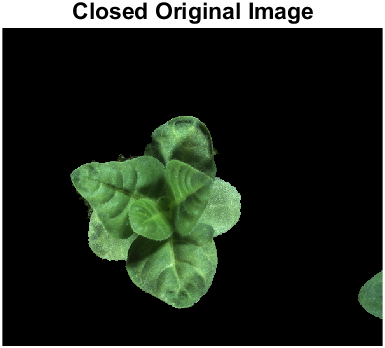
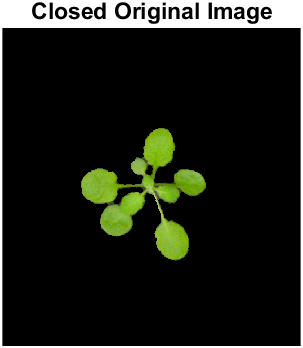
* *The original combined mask is quite dirty. We clean up the mask by first removing connected components with less than 400 pixels.*
  + *bwareaopen (originalMask,400)*

**

* *Then, we morphologically close the mask to remove the holes using a* *disk structuring element with radius 4.*
  + *imclose (originalMask, strel ('disk',3))*

**

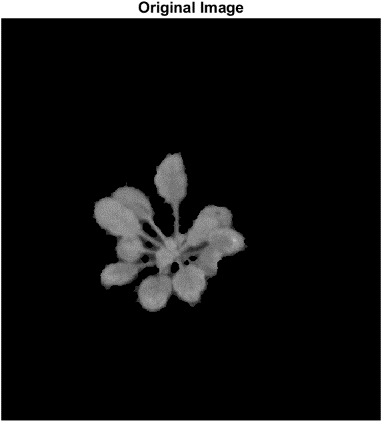
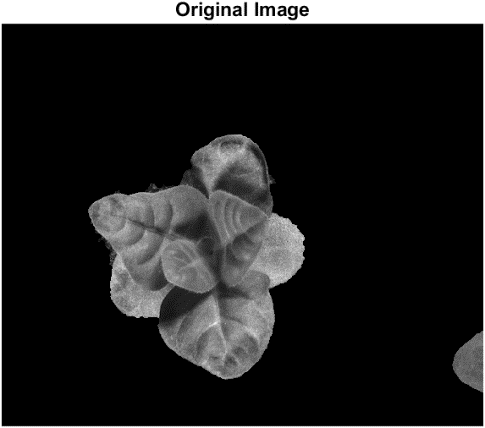
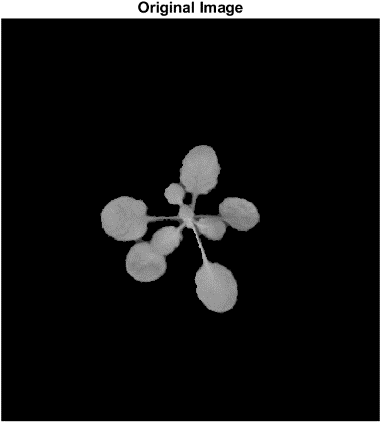
* *Finally, the mask is ready. We overlay the mask on the original image to separate the leaves from the background.*
  + *Mask overlayed on original image*

**

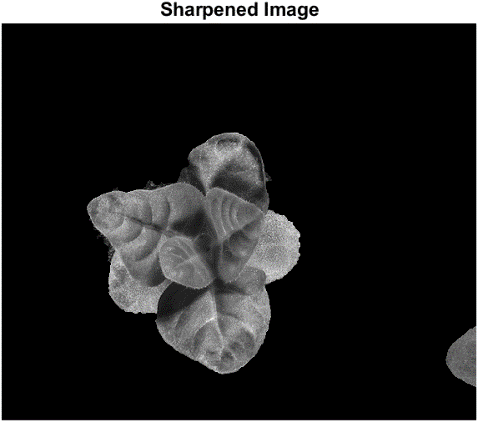
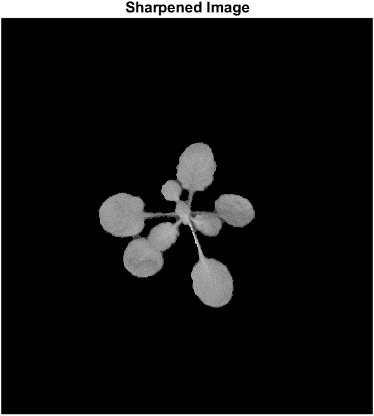
* **Reason for choosing Region of Interest (ROI) Masking**
  + We are only interested in the leaves of the images. So, we use ROI masking to cover the rest of the image so that we may focus solely on the leaves in the following operations.
  + Edge-based methods are not suitable as the images are very diverse so lots of noise will be introduced.
* **Critical Evaluation**
  + **Strengths**
    - It is trivially easy to implement once the common properties of the leaves in all 3 images have been identified which are:
      * green > 1.1 \* red & green > 1.1 \* blue
      * green > (red + blue) / 1.4
      * blueRelative >= 60 & blueRelative <= 121
      * redRelative >= 100 & redRelative <= 125
    - As seen above, we can only see the leaves after ROI masking as we are only interested in the leaves and nothing else.
  + **Weaknesses**
    - The mask properties to be used may be hard to pin down and may require close scrutiny of the images.
    - As seen above, the mask properties such as **green > 1.1 \* red & green > 1.1 \* blue** and **green > (red + blue) / 1.4** may seem arbitrary but are actual results based on close studying of the plant images.

**Unsharp Masking**

* *Using unsharp masking, we sharpen the image for further segmentation.*
* *First, we convert the images to grayscale.*
  + *rgb2gray (originalImage)*

**

* *Now, we sharpen the images with a radius of 0.5 and sharpening effect of 1.5.*
  + *imsharpen (grayScaleImage,'Radius',0.5,'Amount',1.5)*

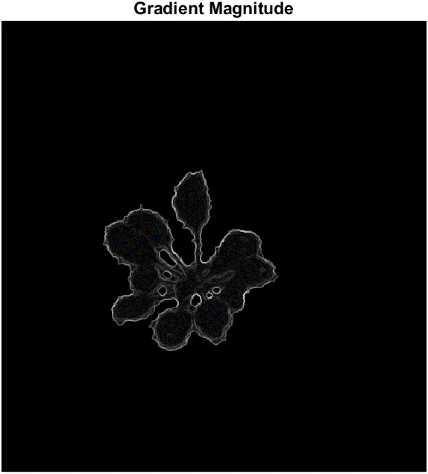
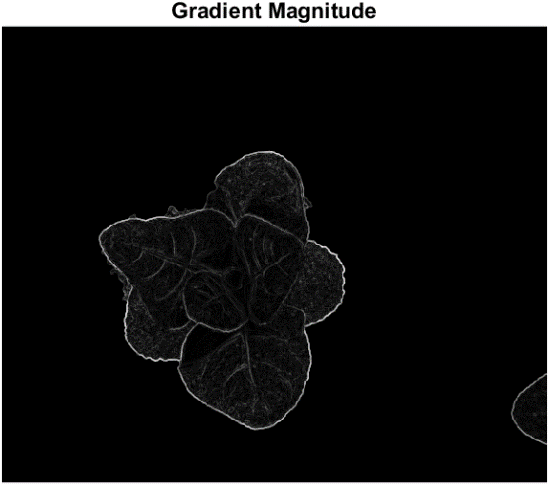
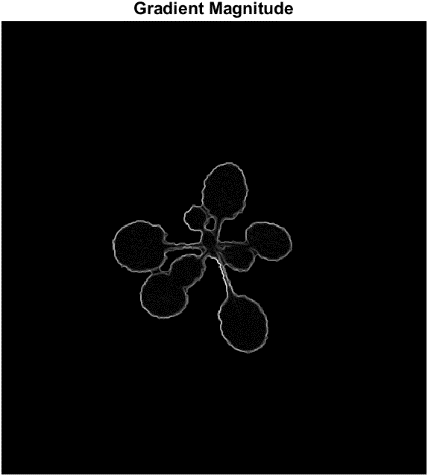
**

* **Reason for choosing Unsharp Masking**
  + The original grayscale image is not very clear, so we increase the contrast and enhance the edges so that the following segmentation may be clearer.
* **Critical Evaluation**
  + **Strengths**
    - Unsharp masking enhances edges and fine detail in the images so that the watershed segmentation may be more accurate.
  + **Weaknesses**
    - Unsharp masking increases noise in the filtered images.

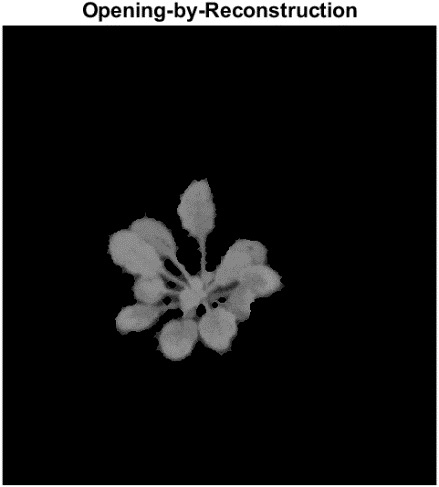
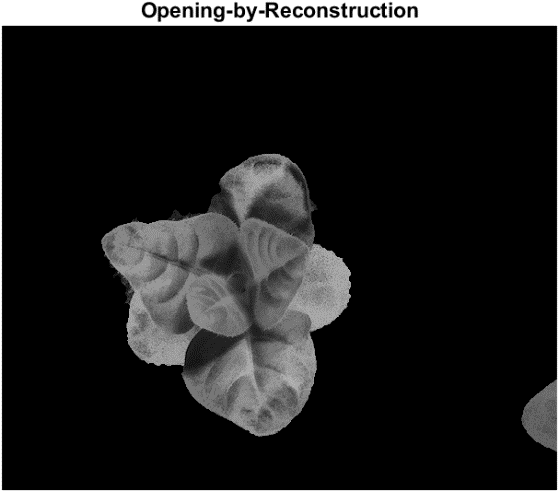
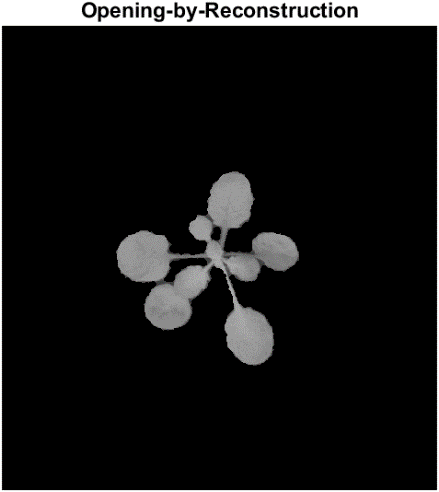
**Segmentation Technique**

**Watershed Segmentation**

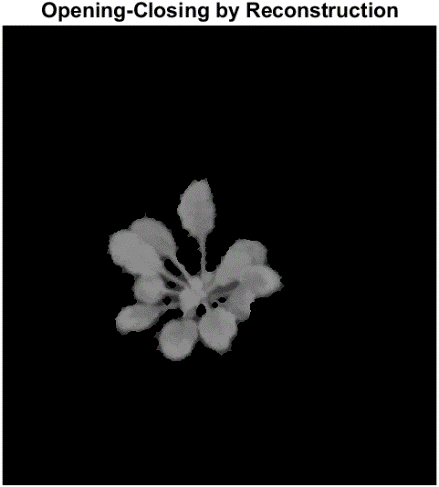
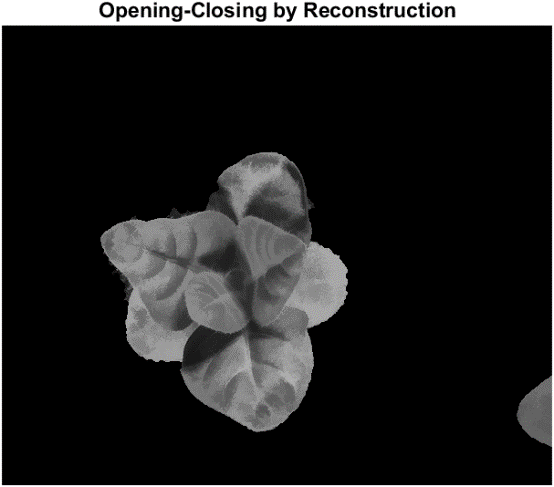
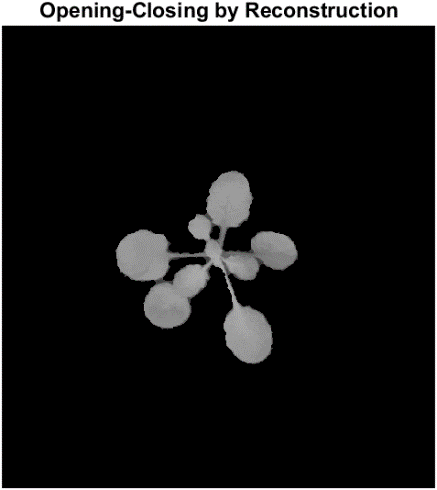
* *Using watershed segmentation, we can separate individual leaves as catchment basins and give each leaf a unique colour by colouring their labels.*
* *We can get the gradient magnitude of the leaves using the function below.*
  + *gmag = imgradient (I)*

**

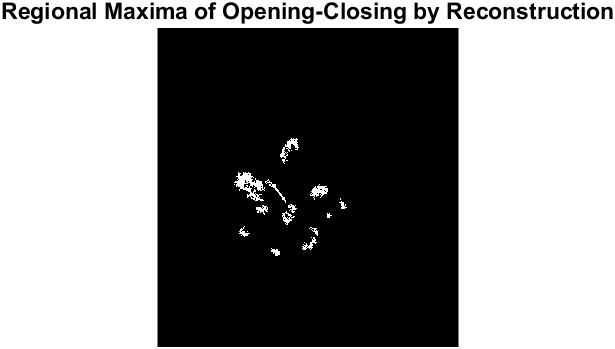
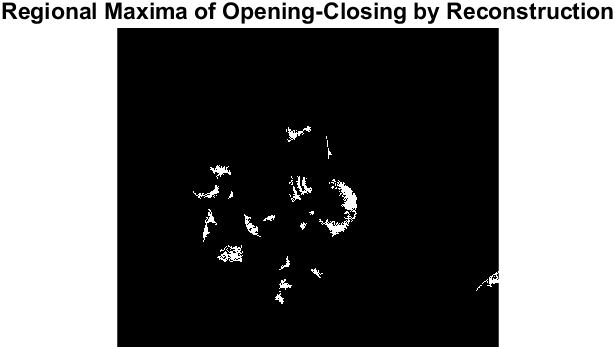
* *However, the gradient magnitude alone is not enough for watershed segmentation. We will find the foreground markers of the images by using morphological techniques called "opening-by-reconstruction" and "closing-by-reconstruction" to "clean" up the image. These operations will create flat maxima inside each object that can be located using the function imregionalmax.*
* *First, we perform opening-by-reconstruction by performing an erosion followed by a morphological reconstruction. The erosion is performed using a disk structuring element with radius 3.*
  + *Ie = imerode (I, strel ('disk',3))*
  + *Iobr = imreconstruct (Ie,I)*

**

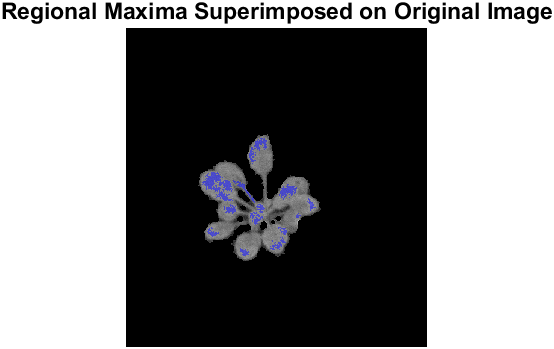
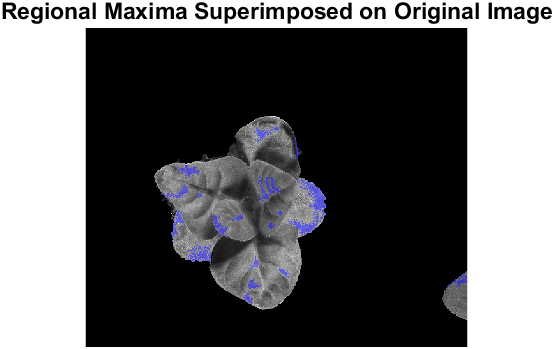
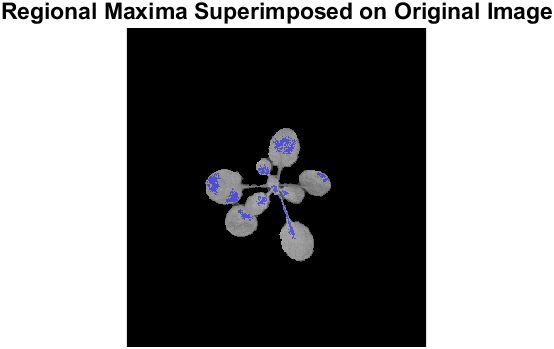
* *Then, we perform closing-by-reconstruction by performing a dilation followed by a morphological reconstruction. The dilation is performed using a disk structuring element with radius 3. We must remember to complement the image inputs and output of the morphological reconstructions.*
  + *Iobrd = imdilate(Iobr, strel ('disk',3))*
  + *Iobrcbr = imreconstruct(imcomplement(Iobrd),imcomplement(Iobr))*
  + *Iobrcbr = imcomplement(Iobrcbr)*

**

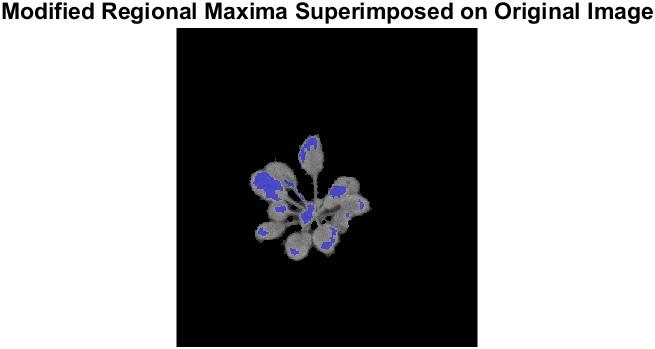
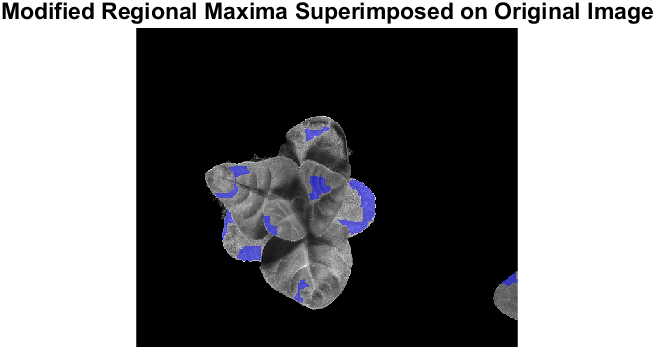
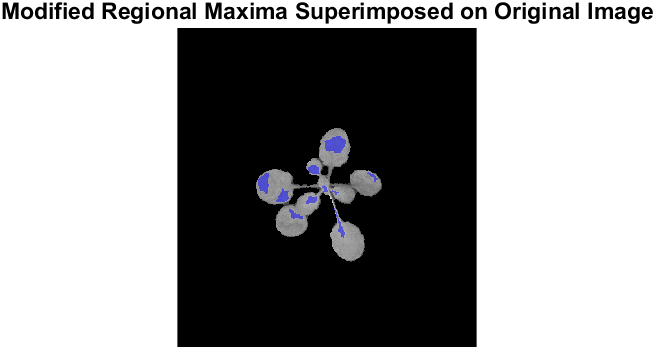
* *Then, we get the regional maxima of the images.*
  + *fgm = imregionalmax (Iobrcbr)*

****

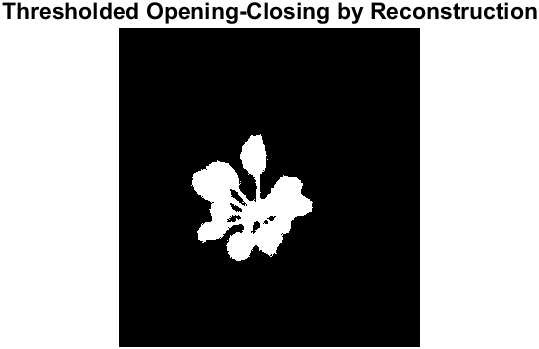
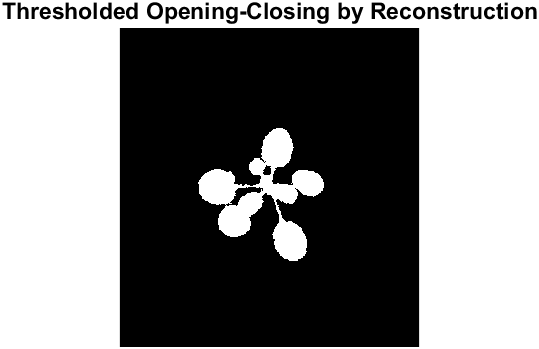
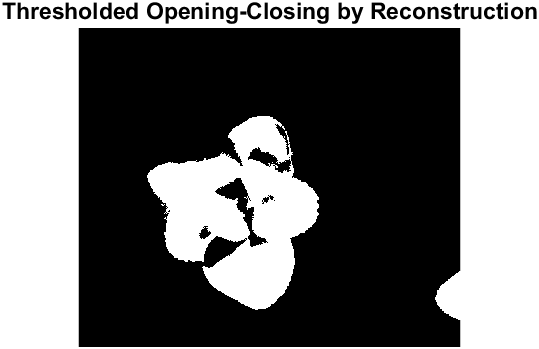
* + *Regional Maxima Superimposed on Original Image*

**

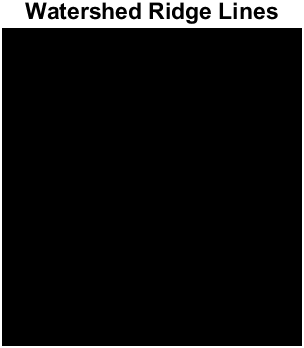
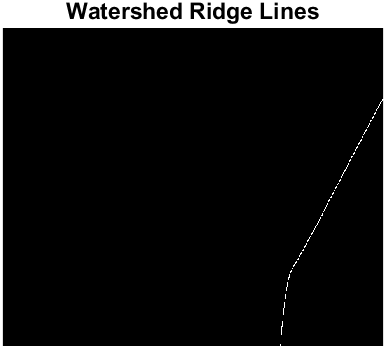
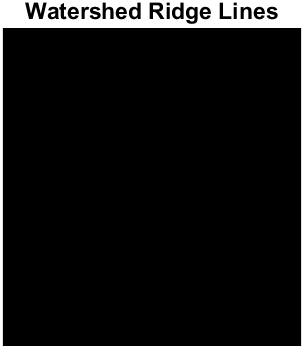
* *We see that the regional maxima are very dirty. So, we must first clean the maxima to prepare it for the watershed segmentation. This can be done by performing a* *morphological closing followed by an erosion to clean the edges of the regional maxima and shrink them.*
* *However, we regrettably reach a point where the 3 images cannot be handled by using the same code. Using the same code for all 3 images will result in an over-segmentation of the third image.*
* *We detect this issue by using the function bwconncomp(fgm4).NumObjects to count the number of connected components and use an if else statement to handle this issue. If the code detects more than 12 connected components, as in the case of the third image, it will use alternative settings for the morphological closing and erosion.*
* *The alternative functions are listed below the original functions. As seen below, the morphological closing of the third image is done using a disk structuring element of radius 7 instead of 3. Then, we find that all 3 images still share the same settings for the erosion operation. Finally, we remove connected components in the third image with less than 150 pixels as opposed to 20 pixels for the first 2 images.*
  + *fgm2 = imclose (fgm, strel ('disk',3))* [1st and 2nd image]
  + *fgm2 = imclose (fgm, strel ('disk',7))* [3rd image]
  + *fgm3 = imerode (fgm2, strel (ones (2,2)))* [1st, 2nd and 3rd image]
  + *fgm4 = bwareaopen (fgm3,20)* [1st and 2nd image]
  + *fgm4 = bwareaopen (fgm3,150)* [3rd image]
* *We then overlay the modified regional maxima on the original images.* 
  + *Modified Regional Maxima Superimposed on Original Image*

**

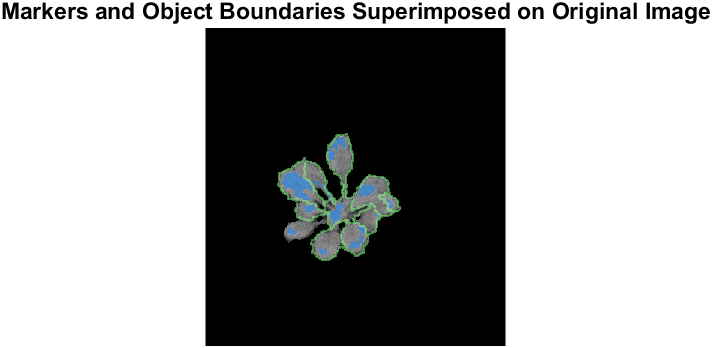
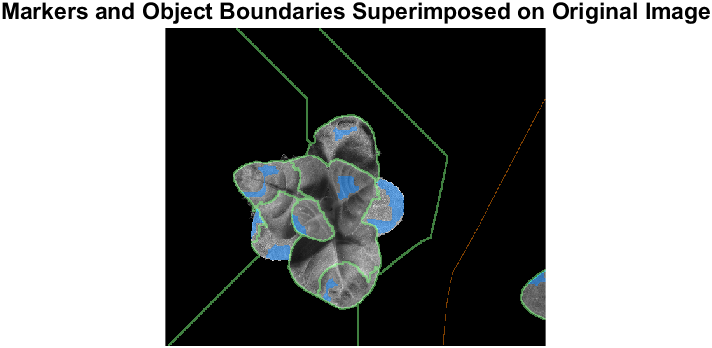
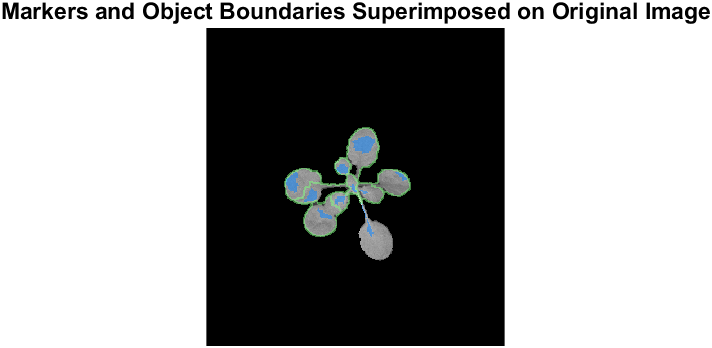
* *Now, we mark the background by binarizing the cleaned-up image of Iobrcbr.*
  + *bw = imbinarize(Iobrcbr)*

**

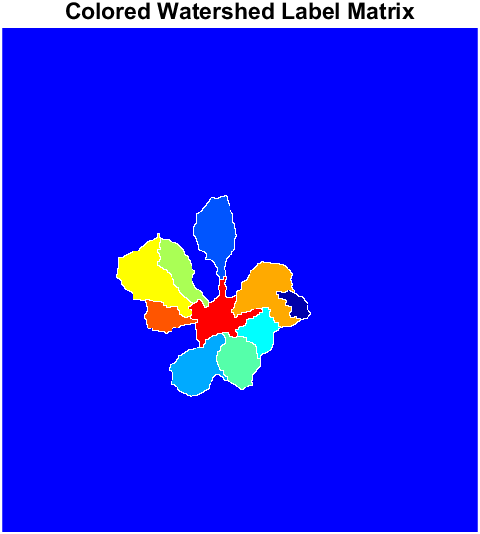
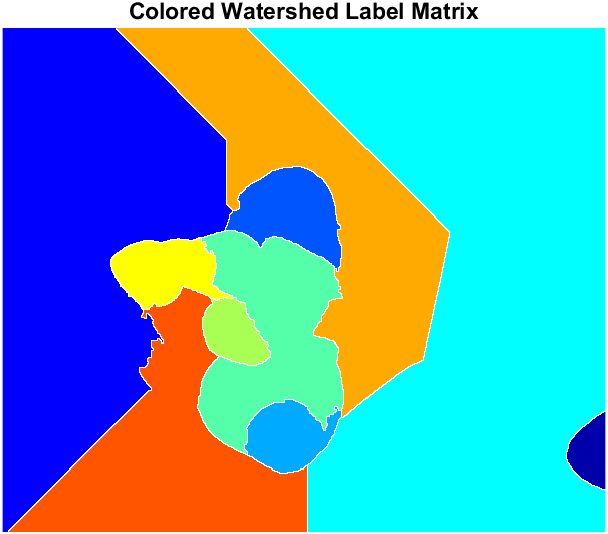
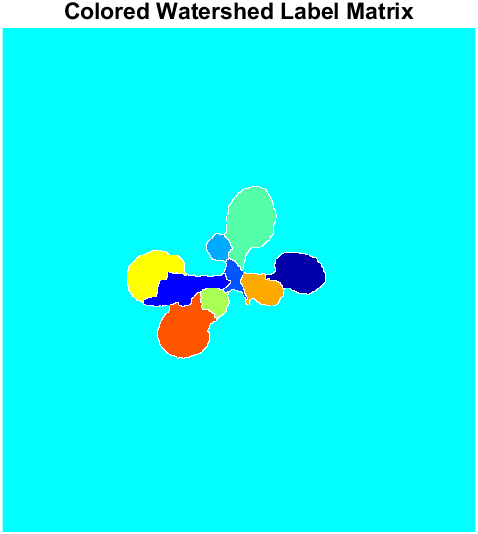
* *Then, we want to thin the background by computing the "skeleton by influence zones", or SKIZ, of the foreground of bw. We do this by computing the watershed transform of the distance transform of bw, and then looking for the watershed ridge lines (DL == 0) of the result.*
  + *D = bwdist(bw)*
  + *DL = watershed(D)*
  + *bgm = (DL == 0)*

**

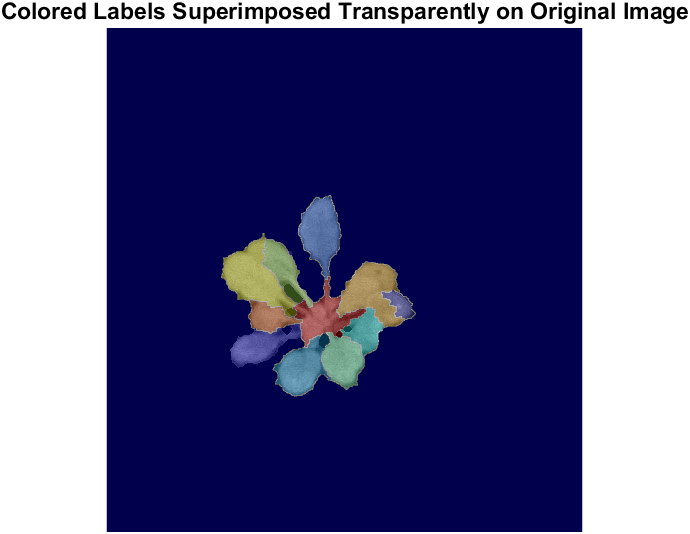
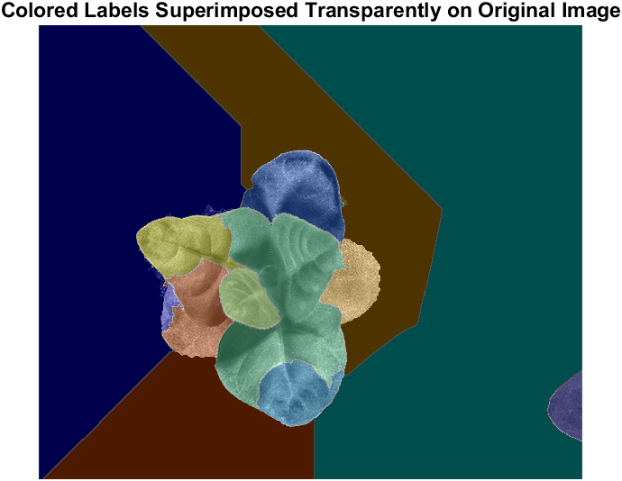
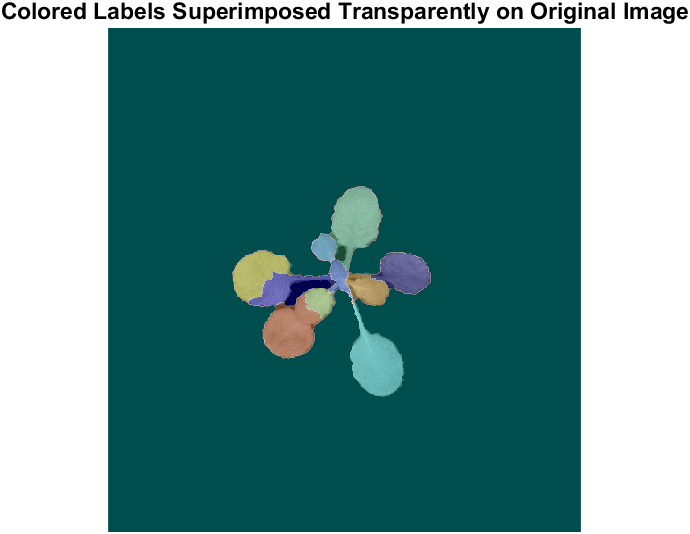
* *Finally, we use imimposemin to modify the gradient magnitude image so that its only regional minima occur at foreground and background marker pixels. Then, we compute the watershed-based segmentation.*
  + *gmag2 = imimposemin(gmag, bgm | fgm4)*
  + *L = watershed(gmag2)*
* *We can visualise the results by superimposing the foreground markers, background markers, and segmented object boundaries on the original image.*
  + *labels = imdilate (L==0, ones (3,3)) + 2\*bgm + 3\*fgm4*

**

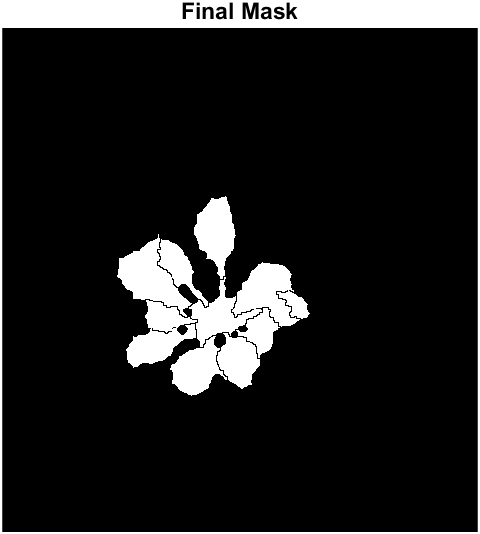
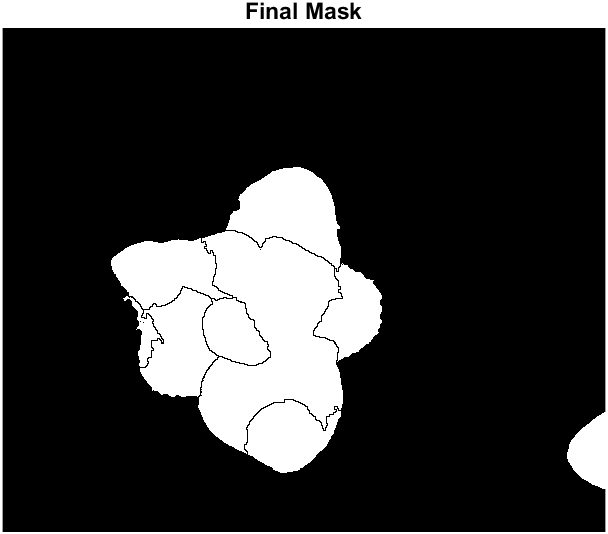
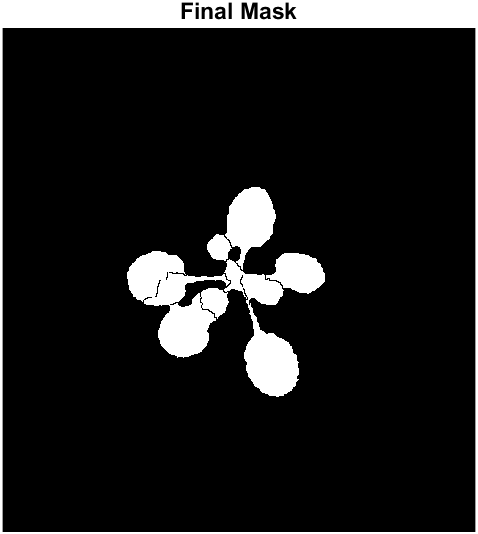
* *We can also visualise the results by displaying the label matrix as a colour image.*
  + *Lrgb = label2rgb(L,'jet','w','shuffle')*

**

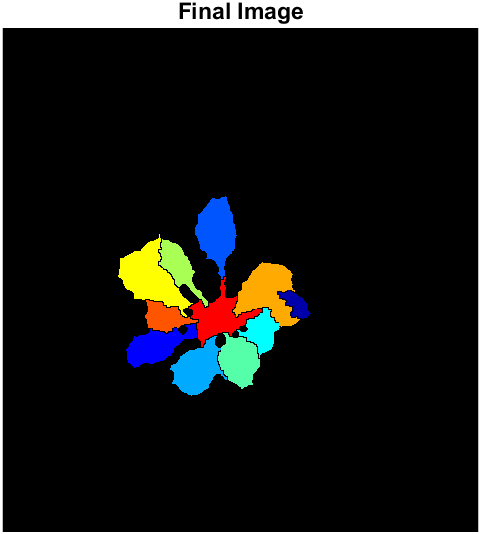
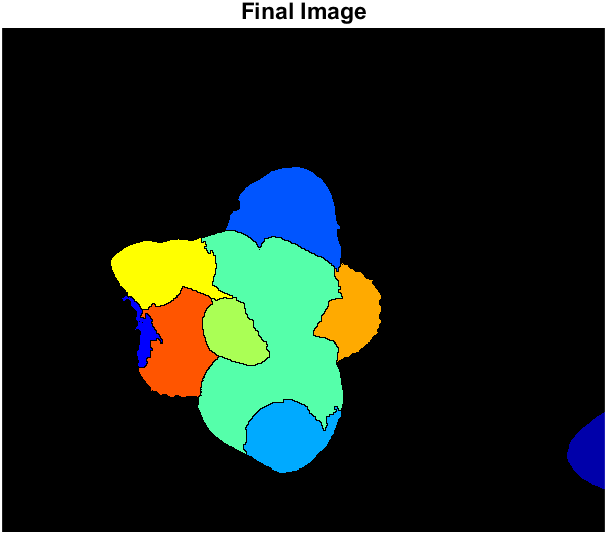
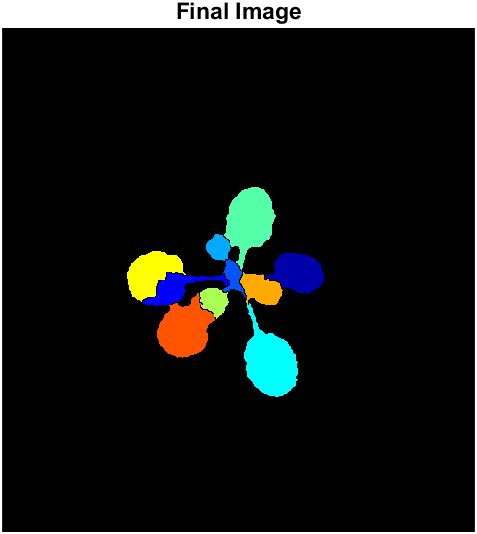
* + *Coloured Labels Superimposed Transparently on Original Image*

**

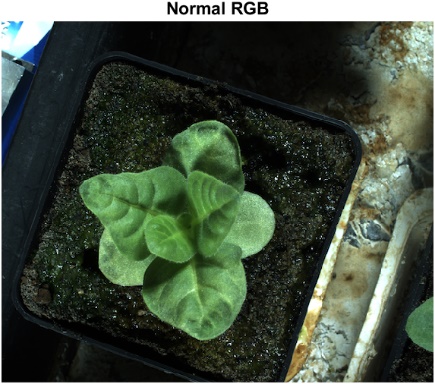
* *The coursework wants coloured leaves on a black background. To accomplish this, we can create another mask.*
* *We take the ROI leaves image and mark non-zero pixels as foreground and the rest as background. We also take the coloured label matrix and mark the white border lines as background. We combine these 2 masks then remove any connected components with less than 200 pixels to remove any remaining noise. The final mask produced is shown below.*
  + *Final Mask*

**

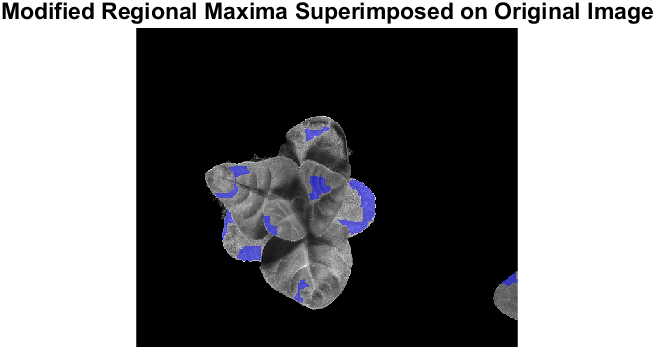
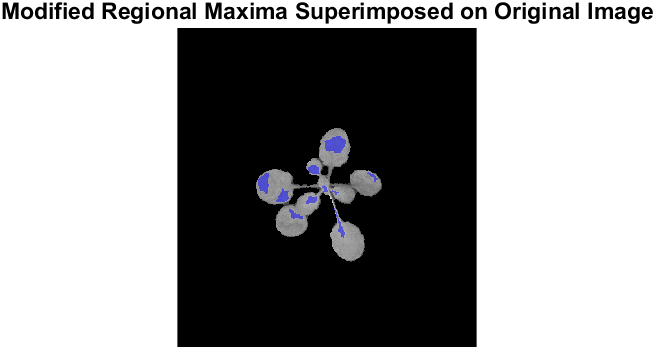
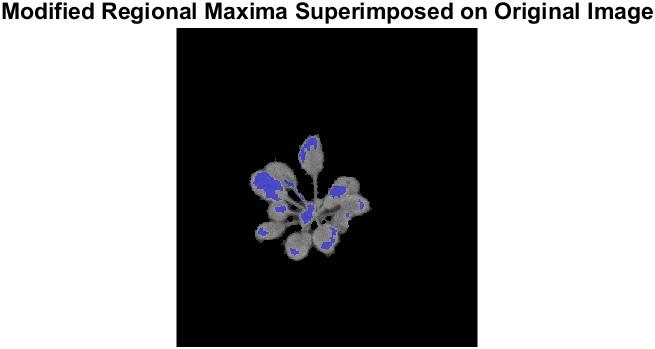
* *By using this mask on the coloured label matrix, we arrive at our final answer.* 
  + *Final Images*

**

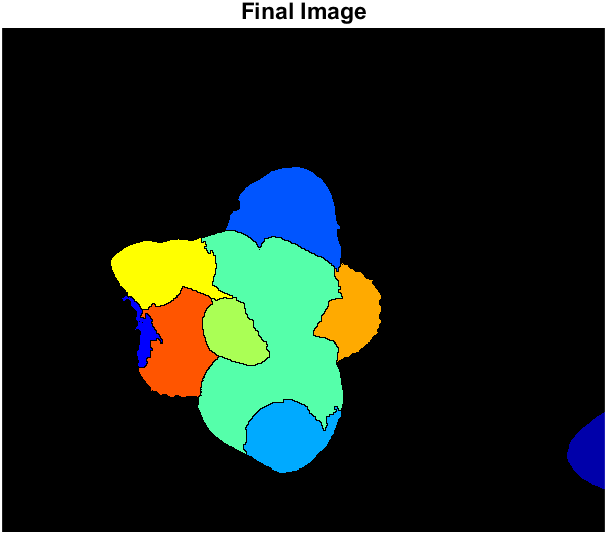
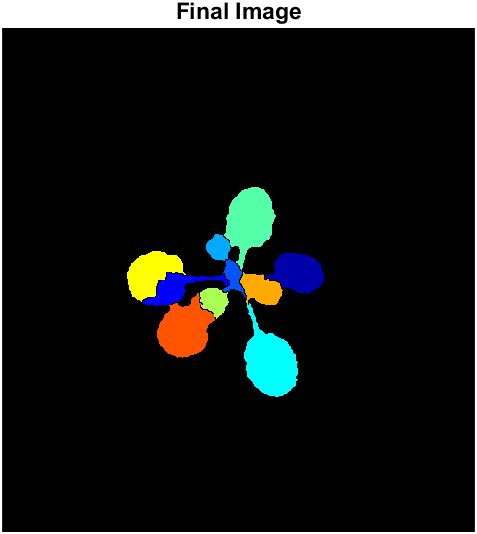
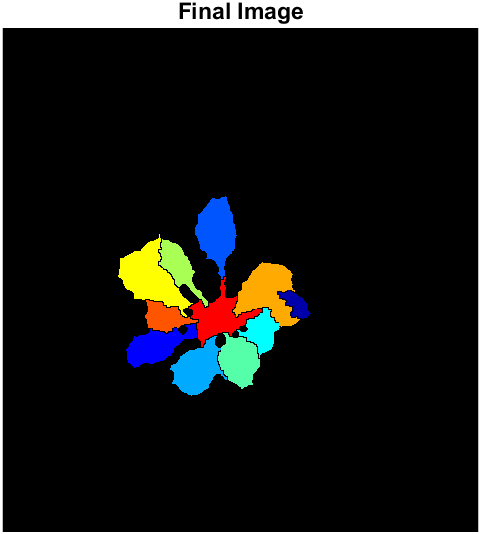
* + *Original Images for comparison*

**

* **Reason for choosing Watershed Segmentation**
  + The coursework requires that each plant is segmented into individual leaves so watershed segmentation is the perfect tool for this job.
* **Critical Evaluation**



* + **Strengths**
    - Watershed segmentation is very good at segmenting the plant into individual leaves based on their local maxima.
    - As seen above, almost each leaf has only one maximum with the exception of a leaf in the 2nd and 3rd image.



* + **Weaknesses**
    - The following segmentation is not perfect as the object borders are not perfectly defined even though the maxima is well defined.
    - This issue is seen the most prominently in the 3rd image. Even with the maxima in the correct position, the borders of the leaves are not well defined so the resulting watershed looks horrible.
    - Each leaf may also have more than one maximum and this may affect the watershed as seen in the 2nd and 3rd image.