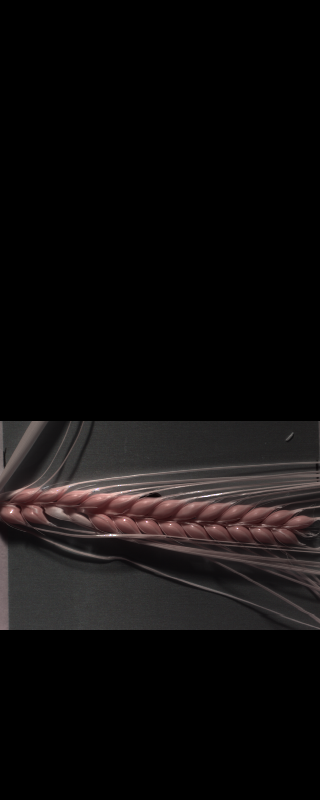
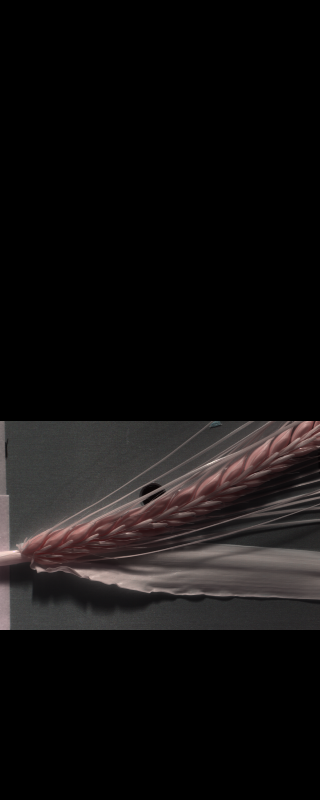
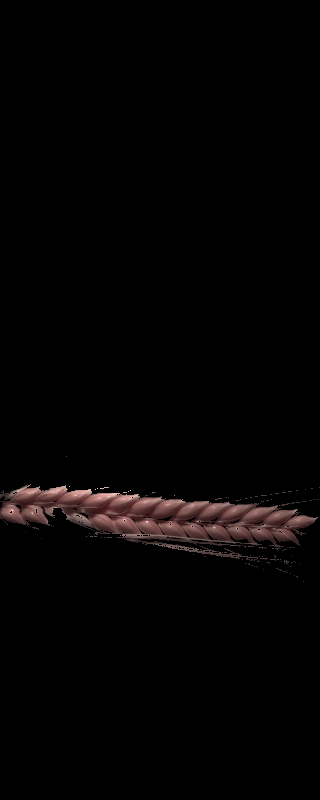
Three channels (copies of three channels pushed in to three new matrixes with in the interval of 0 and 255) are used and as it seems grains are enough different than the leaves and background. In order to save time and not do unnecessary computation, only the area where crops located, are considered for segmentation.

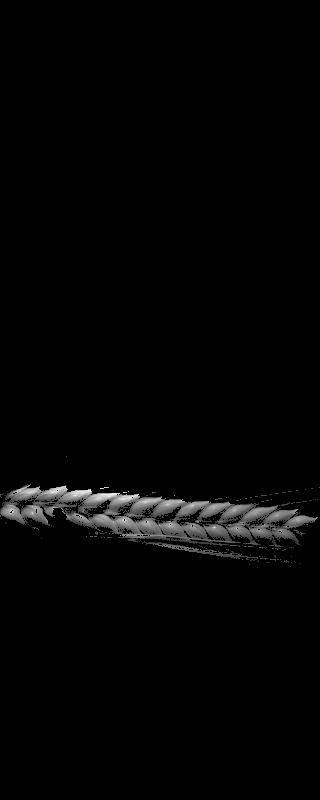
   

After applying mask using openCV library, we got below kind of images. Now, leaves are removed

* **Changing Color-space**
* **Color Conversion**
  + We use the function **cv2.cvtColor(input\_image, flag)** where flag determines the type of conversion (e. g. cv2.COLOR\_BGR2HSV).
* **Applying mask**
  + We use the function **cv2.inRange(HSV\_converted\_image, lower\_bounds, upper\_bounds)**
  + For HSV, Hue range is [0,179], Saturation range is [0,255] and Value range is [0,255]. Different softwares use different scales. So need to normalize these ranges.
  + lower\_bounds = (0, 60, 50) and upper\_bounds = (100, 255, 255) are used

If tune it more, it will affect the grains in some images. And since I don’t need three channels any more, I keep only one channel and trying to remove noises from around the images.

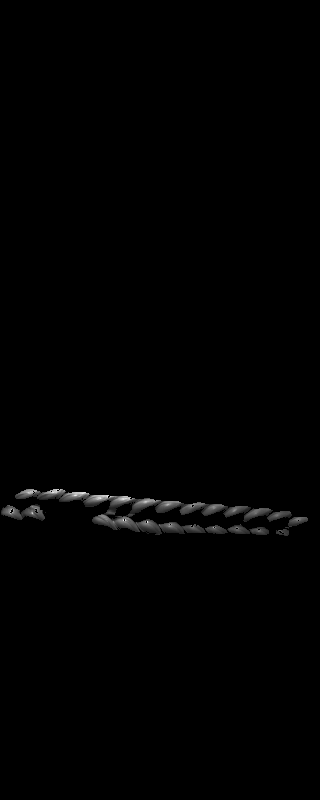
   

I wanted to make every grain well separated, so I squared the values of each pixel and again all the values pushed to the range of 0 and 255 for illustration purpose.

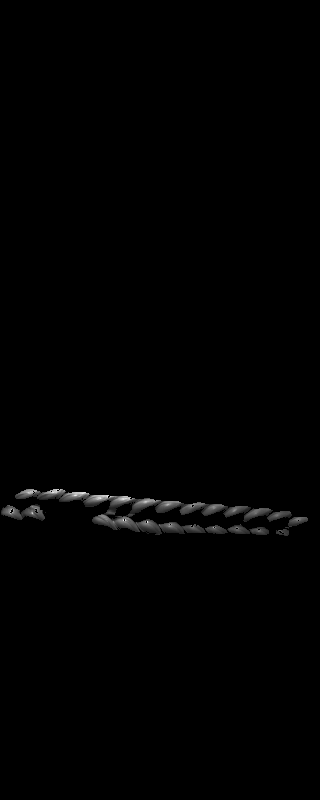
   

Structuring element used on the gray scale images to remove noises. Every pixel with in its radius (distance based on number pixels based on Moore neighborhood) and minimum number of neighbors is considered such that if number of points was less than minimum points, the point was simply valued as zero.

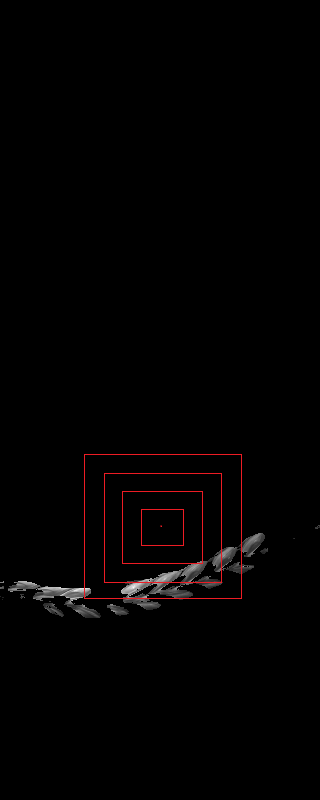
Radius = 5 and Minimum points = 50 results followings

And as a result of two iterations we got followings:

Each grain consist of around 20 X 20 pixels (SWIR) and since we need to get 5 to 6 grains from middle of each spike, so I consider that in total we need around 2000 pixels from each image. In order to get middle grains, we started from middle point of area where spikes are located and expanded its area until we cover 2000 pixels.



So as a result we would have below highlighted pixels in the following images for further steps

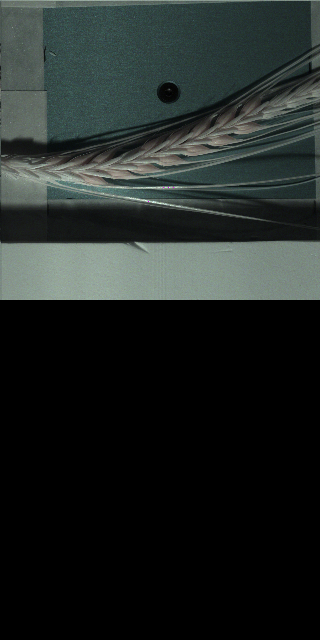
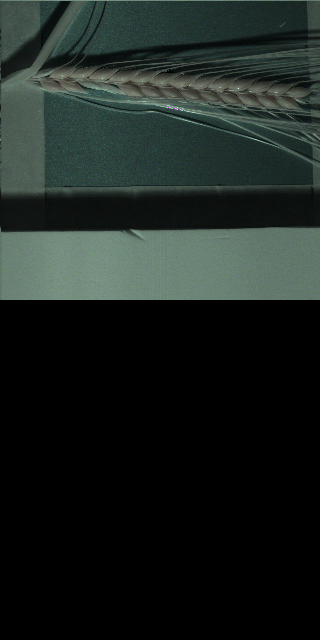
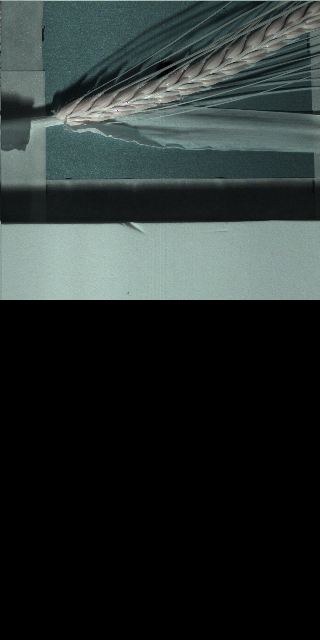
   

So for every image (SWIR) we would have 2000 pixels to act as 2000 variables for 256 values. And regarding VNIR since it has wider size and each grain is consist of around 2500 pixels, I would like to apply same procedure but again take more grains into consideration (e.g. 5 to 6 grain) and randomly pick 2000 of them to match our SWIR size. And during merging we will make it as 256 + 160 values for each variable.

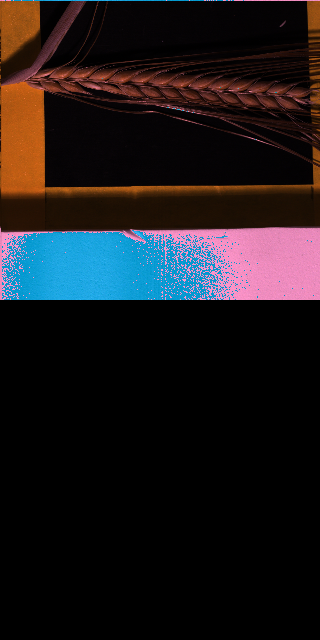
VNIR

Original images: images resized (from 1600 x 3200 to 320 x 640) and only upper side of the images is taken for further process to minimize calculations.

Three channels: combination of 159, 152, 120 channels

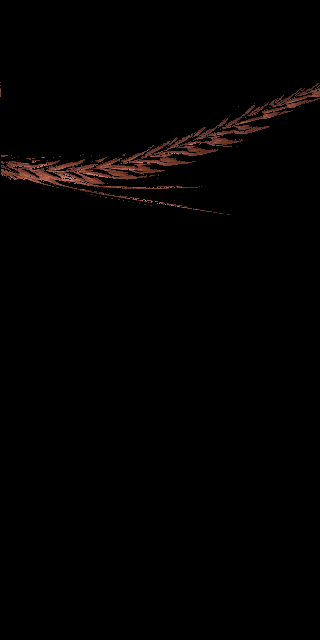
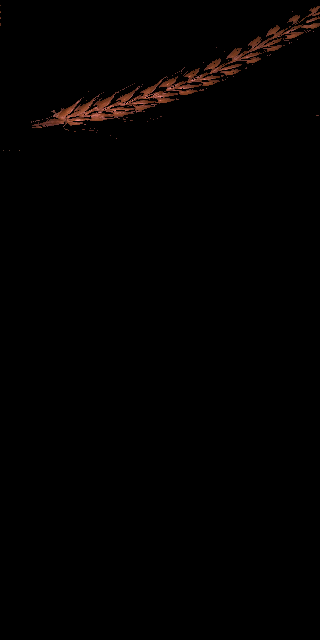
   

Different combinations: 23, 31, 39 channels

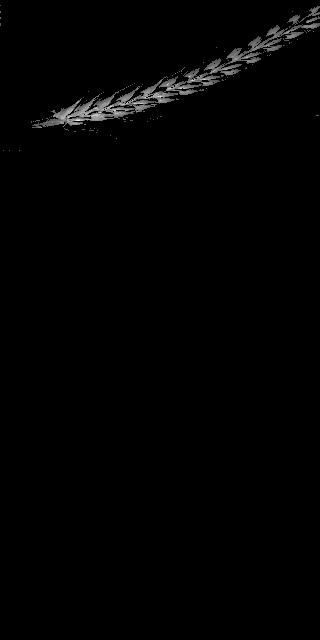
   

After applying mask with:

* lower\_bounds = (0, 0, 80)
* upper\_bounds = (10, 250, 255)

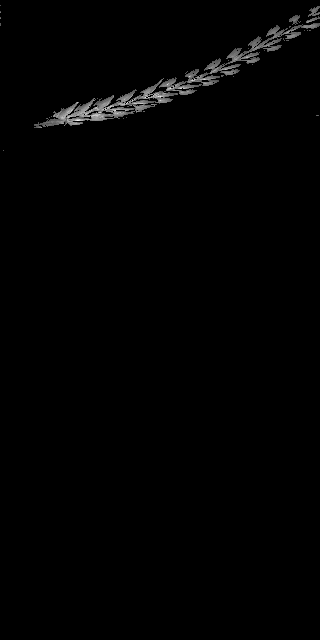
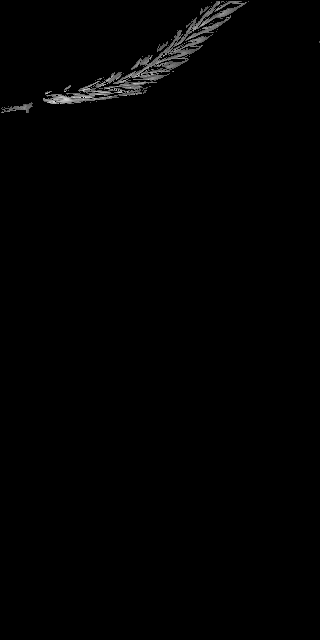
   

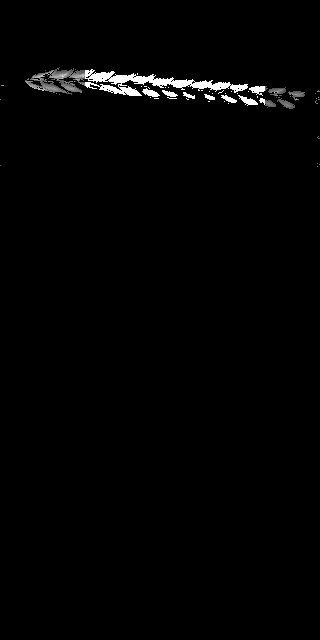
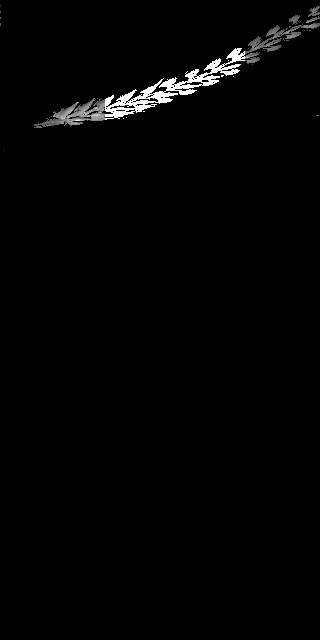
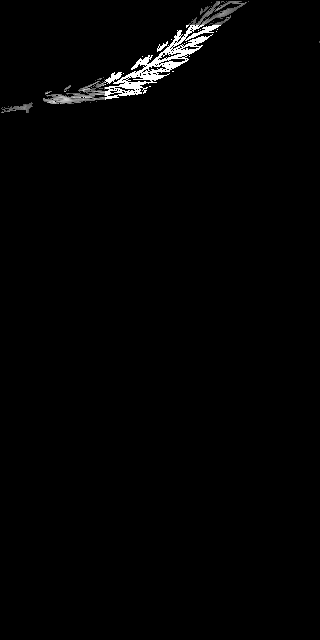
Keeping only single channel for cleaning grains:

Here grains are well separated so we only need to remove the noises. Radius = 5

* Minpoints = 20
* Number of iterations: 2

**Merging both images:**

Finding few points to capture coordinates of each images and level of zooming plus rotation if there is.

