RYERSON UNIVERSITY

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

BE8105

Advance Medical Image Analysis

Lab 2: Image Representation, Quality, and Spatial Filtering

Lab Number:	2

Student Number	Name
500565465	Justin Pontalba

By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a 0 on the work, an F in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at: www.ryerson.ca/senate/current/pol60.pdf.

Before submitting your report, your TA asks questions about your report. If there is no consistency between your oral answer and your report, you will lose 50% of your total mark. For this part, Pass or fail will be circled next to your name accordingly.

BE8105 – Advance Medical Image Analysis

Lab 2 – Image Representation, Quality, and Spatial Filtering

PRELAB [10 marks]

1. Brightness and Contrast [3 marks]

a) Give a definition of brightness and contrast in images.

Brightness is a subjective descriptor that is impossible to measure, however, it embodies the achromatic notion of intensity and is a key factor in describing colour sensation.

Contrast describes the relation between the brightness values in an image or section of an image. Furthermore, contrast can be described as the perception of brightness in relation to the lightness of the background. [cite]

b)

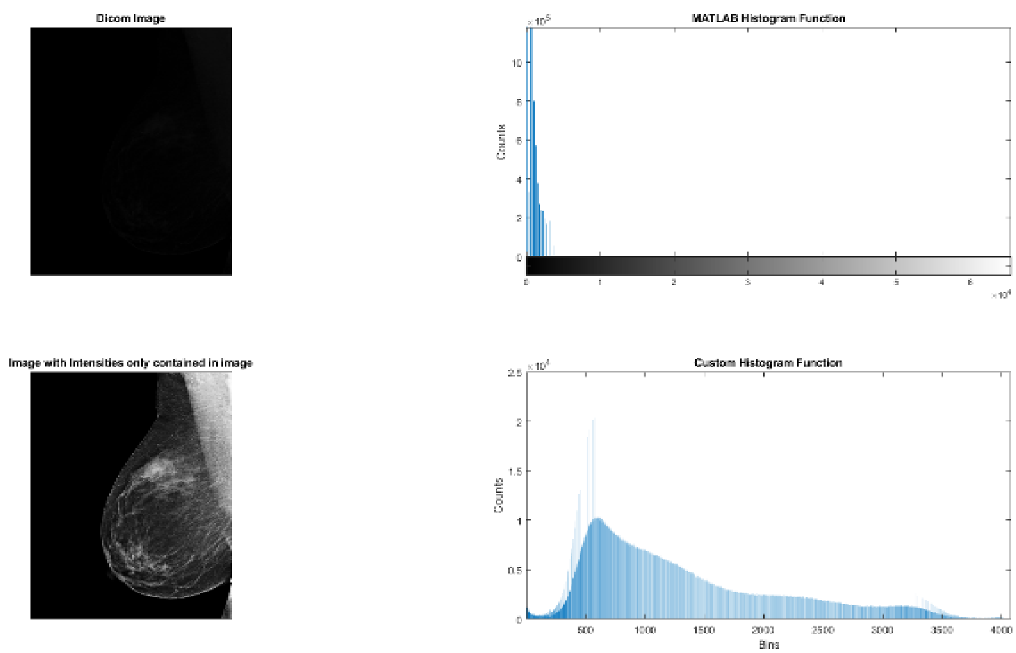


Figure 1. Image with corresponding histogram

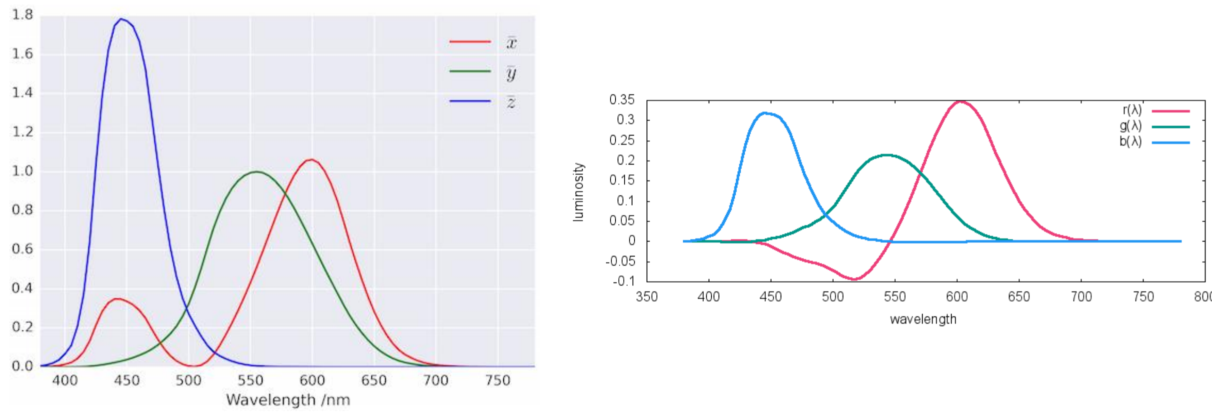
c)

Based on the histogram in part b), many pixel values are attributed to few low intensities, while there are fewer bright intensities. This indicates that the image exhibits low contrast and demonstrates low brightness. This is depicted in row 1 of Figure 1. The tissue structures are not discernable and the image appears dark. However, if the image histogram is constrained to the maximum and minimum range, the objects and tissue structures in the image are discernable and the image appears to have improved contrast.

2. Colour Spaces [2 marks]

(a) In two paragraphs, along with diagrams, explain how the XYZ space works.

The XYZ colour space is a CIE standard that is designed to define a color model that covers the entire visible wavelength area. The X, Y, and Z values however are virtual values that must be converted to real spectral values using color matching functions. The 2° or 10° CIE standard observer standards dictates which colour matching functions are used. 2° or 10° refers to the visual field that a coloured object is viewed at which relates to specific colour sensitivity of the eye, as colour sensitivity of the eye changes according to the angle of view. $\tilde{x}, \widetilde{y},\ and\ \tilde{z}$ represents the linear transformation [cite].



a) CIE $\tilde{x}, \widetilde{y},\ and\ \tilde{z}$ colour matching function *adapted from https://scipython.com/blog/converting-a-spectrum-to-a-colour/*

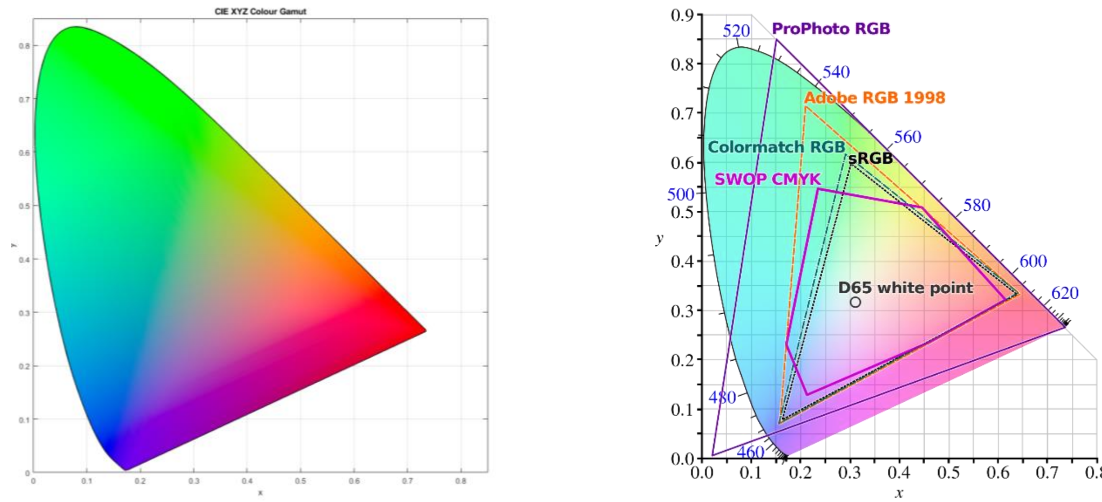
b) CIE r, g, b colour matching function adapted from https://medium.com/hipster-color-science/a-beginners-guide-to-colorimetry-401f1830b65a

Figure 2. Colour matching functions

The following equation defines the linear transformation:

$$\begin{pmatrix} \widetilde{x\ (\lambda)} \\ \widetilde{y\ (\lambda)} \\ \widetilde{z(\lambda)} \end{pmatrix} = \begin{pmatrix} 0.49000 & 0.31000 & 0.20000 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00000 & 0.01000 & 0.99000 \end{pmatrix} \cdot \begin{matrix} \widetilde{r\ (\lambda)} \\ \widetilde{g\ (\lambda)} \\ \widetilde{b(\lambda)} \end{matrix}$$

(b) Show a colour gamut and describe how it can be used in electronic displays and printing devices.



a) XYZ Chromaticity Diagram

b) Various gamuts adapted from https://en.wikipedia.org/wiki/Color_space

Figure 3. Chromaticity diagram and gamut

Figure 3 above depicts the XYZ chromaticity diagram. The x-axis represents the *x* component and the y-axis depicts the *y* component. From the chromaticity diagram, a colour gamut can be derived. Colour gamuts are used to define colour ranges of colour monitors and colour printers.

3. HSV Space [3 marks]

(a) Describe the HSV space, and discuss what each channel (H,S,V) represents.

The HSV colour space is a colour space where colours are defined according to a hexagonal prism or commonly in a conical shape where H represents *hue,* S represents *saturation,* and V represents *value.* This colour space decouples the intensity component and colour-carrying component of colour [cite]

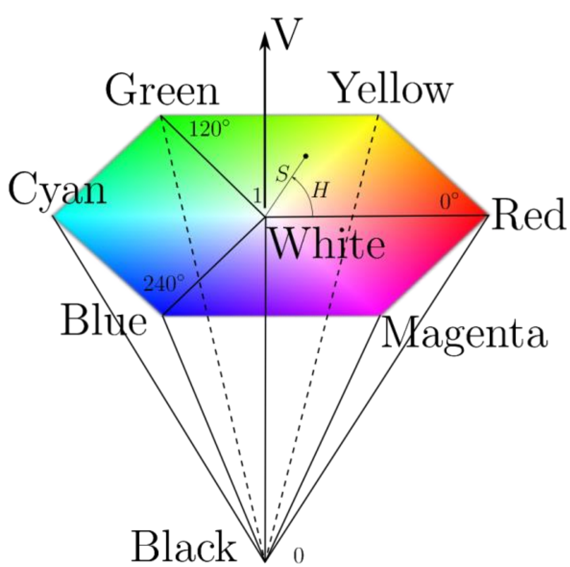


Figure 4. HSV colour space representation adapted from http://argos.vu/devnotes-9-16-16-ui-work-initialization-serialization/

The hue channel defines the apparent colour or dominant wavelength and is valued at 0^{o} to 360^{o}. Major colours and their angle can be observed from Figure 4. Saturation is defined on the central axis (going into the page) and describes the *concentration* or *purity* of colour. Saturation is defined between 0 to 1 or 0% to 100%. For S = 0, hue is undefined. The value component is based on brightness and lies between 0 on the apex of the prism and 1 on the base.

(b) Convert "malignant.tiff" to the HSV space. Show the original image, and each one of the channels.

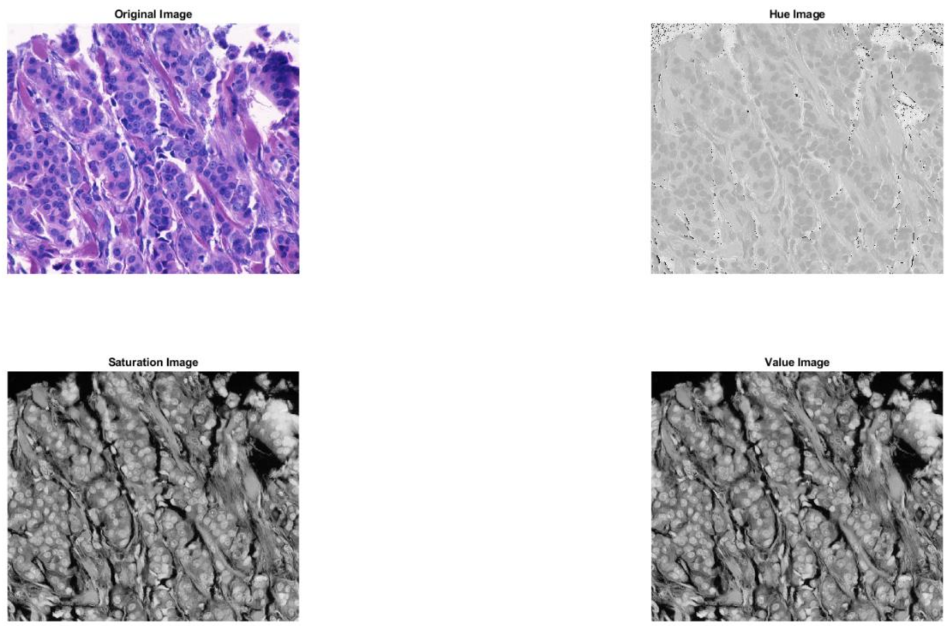
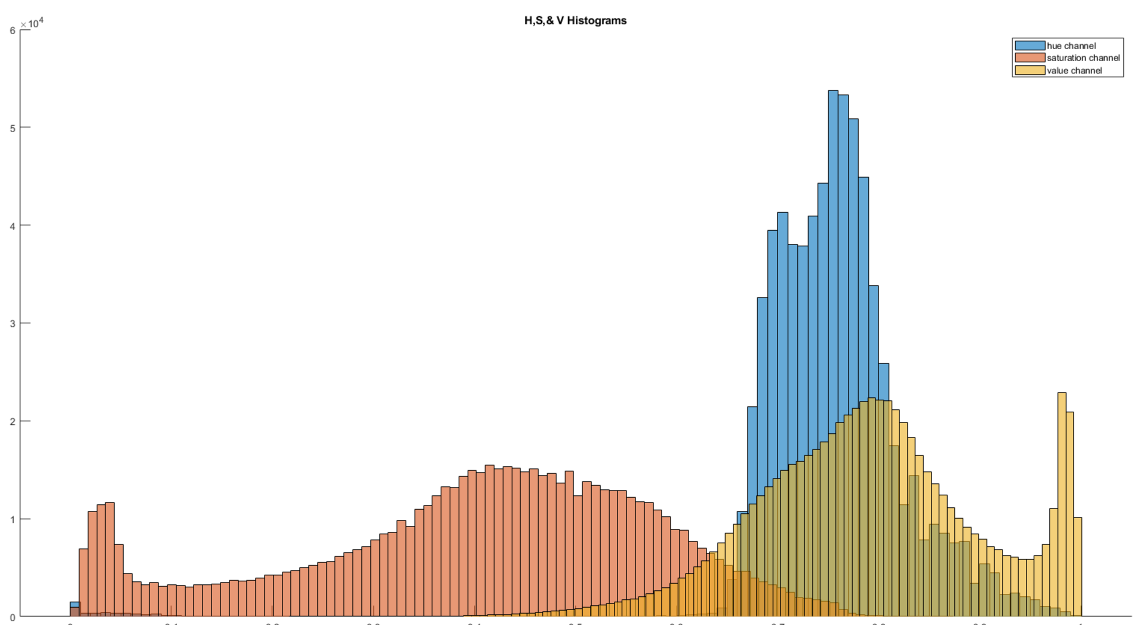


Figure 5. Original vs. H, S, V Channel Images

(c) Using a single figure, show the histogram of each channel with labels and legends.



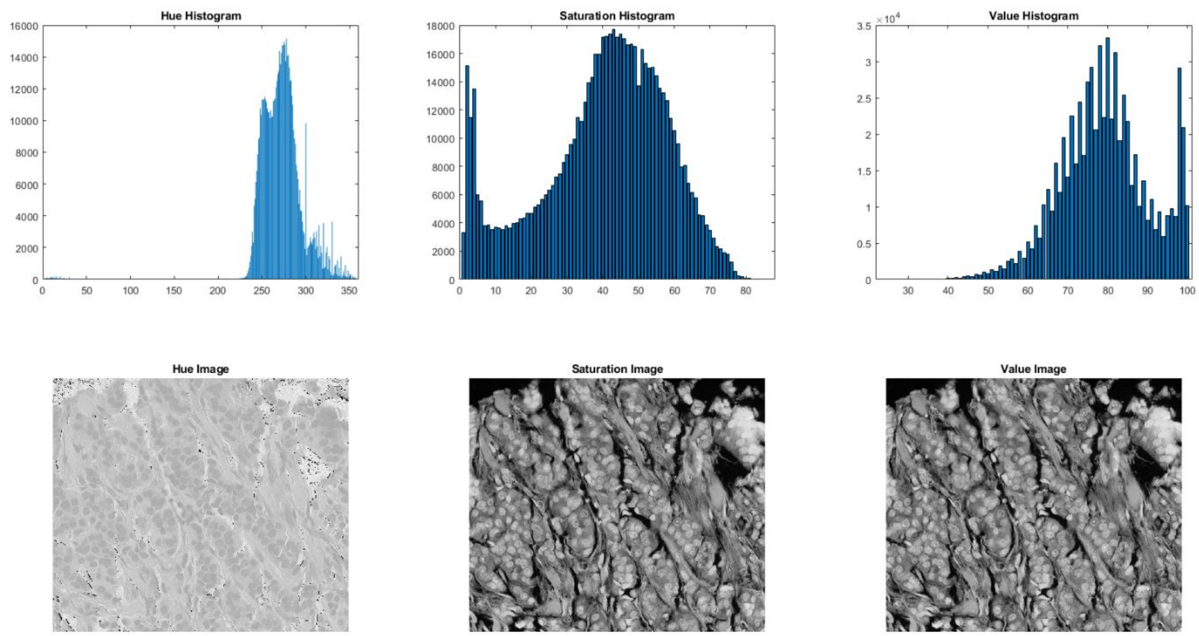


Figure 6. HSV Images and corresponding histograms

(d) Based on the visual appearance of each channel, as well as the histograms, discuss some advantages of HSV in terms of detecting nuclei, i.e. which channel would you choose to process and why?

Analyzing the hue channel first, the histogram exhibits a multimodal shape which indicates multiple dominating colours in the image. Knowing the hue range of the hsv space it is clear that there are colours ranging from blue (240^{o}), magenta (300^{o}), and red (0^{o}, 360^{o}), but mainly dominated by a mix of blue and magenta. This is also reflected in the hue image, where nuclei and stroma are discernable because of the hues they display. From the saturation and value channel histograms, it is difficult to attribute where the values are localized. When observing the saturation and value images, the nuclei regions exhibit high saturation and value values while stroma and background are low and close to zero respectively. For detecting nuclei, the hue channel would be used as there is greater discernibility between nuclei and surrounding structures. If a simple threshold method is applied for nuclei detection the nuclei objects should be captured easily.

4. Contrast and Edges [2 marks]

(a) Explain the concepts of contrast and edges in images and discuss why these image features are important when considering image quality.

As previously mentioned contrast describes the relation between the brightness values in an image or as the perception of brightness in relation to the lightness of the background. For the ideal contrast, the distribution of intensities contained in an image are equally spread through its full range. Overexposed or bright images tend to have higher intensity pixels while underexposed images have more pixels of lower intensity. Image contrast will affect how discernable certain objects are in an image. Understanding the concept of contrast, and being able to identify *good* and *bad* cases, is useful when applying enhancement techniques. Technically speaking edges in images are formed from pixels with derivative values that exceed a defined threshold. Edges are based on a measure of intensity-level-discontinuity at a point. Edges are an important feature that give detail to images as they infer texture, form shapes, and create boundaries. High edge content in an image may give a *sharp* appearance, which means that the image highlights transitions in intensity. On the other hand, low sharpness may correspond to low edge content and may indicate the image is blurry or objects are not as discernable as in the previous care. Image of good quality could have equalized contrast and moderate sharpness. Note any extreme level of contrast or sharpness can be of *bad* or *good* quality. These descriptors are subjective to the observer and purpose of the image.

(b) Show (mathematically) how to compute the edge strength for a grayscale image. Show how you would approximate the 2D gradient. Use proper mathematical notation.

As previously mentioned edges are formed from pixels with derivative values, and because we are applying edge detection to digital images, derivatives of a digital function are defined in terms of differences. A basic first-order derivative of a one dimensional function *f(x)* is:

$$\frac{\partial f}{df} = f(x+1) - f(x),$$

whereas for a second-order derivative is:

$$\frac{\partial f^2}{\partial x} = f(x+1) + f(x-1) - 2f(x).$$

Now in the 2-dimensional case, a common second order operator is the Laplacian which is defined as:

$$\nabla^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y).$$

The filter masks used to implement this operation is shown below:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

EXPERIMENTS [20 marks]

5. Grayscale Edge Detection and Denoising. [10 marks]

(a) Using a gradient filter, perform grayscale edge detection on "mri.dcm". Compute the magnitude, and choose an appropriate threshold to generate an edge map. Describe (and justify) your algorithm and show all results.

The algorithm to perform edge detection was implemented as follows:

I. Image read into MATLAB using *dicomread*
II. The horizontal and vertical gradients were computed by convoluting (custom function) the image with the respective filter masks (*sobel, prewitt* horizontal and vertical filter masks).
III. The gradient magnitude was computed by: $G = \sqrt{G_x^2 + G_y^2}$
IV. Otsu's thresholding was applied to find the global optimal threshold to separate the edge gradients and background

```
function [result] = customConv(img,kernel,
stride,zeropad,n_size)
%CUSTOMCONV performs spatial convolution
given an input image and kernel
% stride defines number of pixels to shift by
% zeropad defines the number of padding to apply
% n_size defines the number of channels
  [m,n,d] = size(img);
  [m2,n2,d2] = size(kernel);
  new_array = zeros(m,n);

  img = padarray(img,[zeropad zeropad],0,'both');
  [m1,n1,d1] = size(img);

  for i =1:stride:m1
    for j=1:stride:n1
      s = subim(img, n_size, n_size, i, j);
      value =
getConvNeighbourhood(double(s),double(kernel));
      new_array(i,j) = value;
%         fprintf("\nrow:%d,col:%d\n", i,j);
    end
  end

  result = new_array;

end
```

a) Convolution Function

```
function s = subim(f, m, n, rx, cy)
%SUBIM Extracts a subimage, s, from a given
image, f.
% The subimage is of size m-by-n, and the
coordinates of its top,
% left corner are (rx, cy).
s = zeros(m, n);
[d1,d2,d3] = size(f);
  for r = 1: m

    for c = 1:n

      row = r + rx - 1;
      col = c + cy - 1;
      if row > d1 || col > d2
        continue;
      else
        s(r,c) = f(row,col);
      end

    end

  end
end
```

b) Neighbourhood extraction function

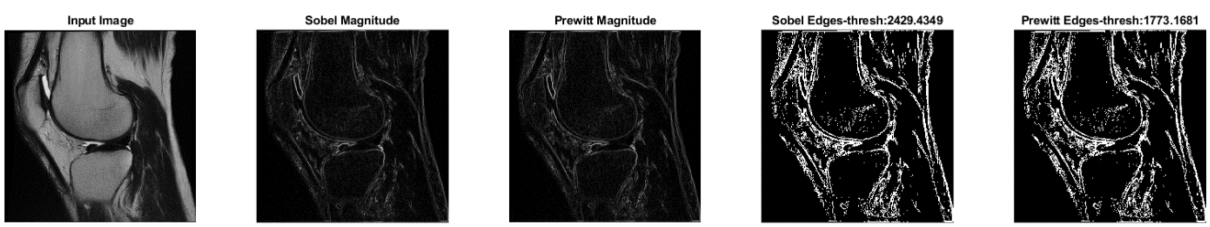


Figure 7. Edge detection

(b) Use Gaussian filters to denoise “mri.dcm” with the following filter sizes: 3x3, 5x5 and 7x7 and 9x9. Show how you chose the correct parameters for the Gaussian, and what the values were. Display the original and four filtered images, along with the corresponding histograms. Comment how the image quality, and histograms change for each filtered image.

In order to choose the filtering parameters for the Gaussian signal-to-noise ratio (SNR), peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) were used. High values of SNR and SSIM are desired whereas low values of PSNR are desired. The figure below depicts these metrics as a function σ and kernel size. As depicted by the figure, as sigma and neighbourhood is increased from 0.6 to 9.6 and 3 to 9, SNR is not largely affected. However, PSNR and SSIM are. This is expected as larger neighbourhoods and a larger sigma value would cause greater blurring effects. For this reason, a middle sigma value was used to over exaggerate the results of neighbourhood size. A sigma value of 5.1 was

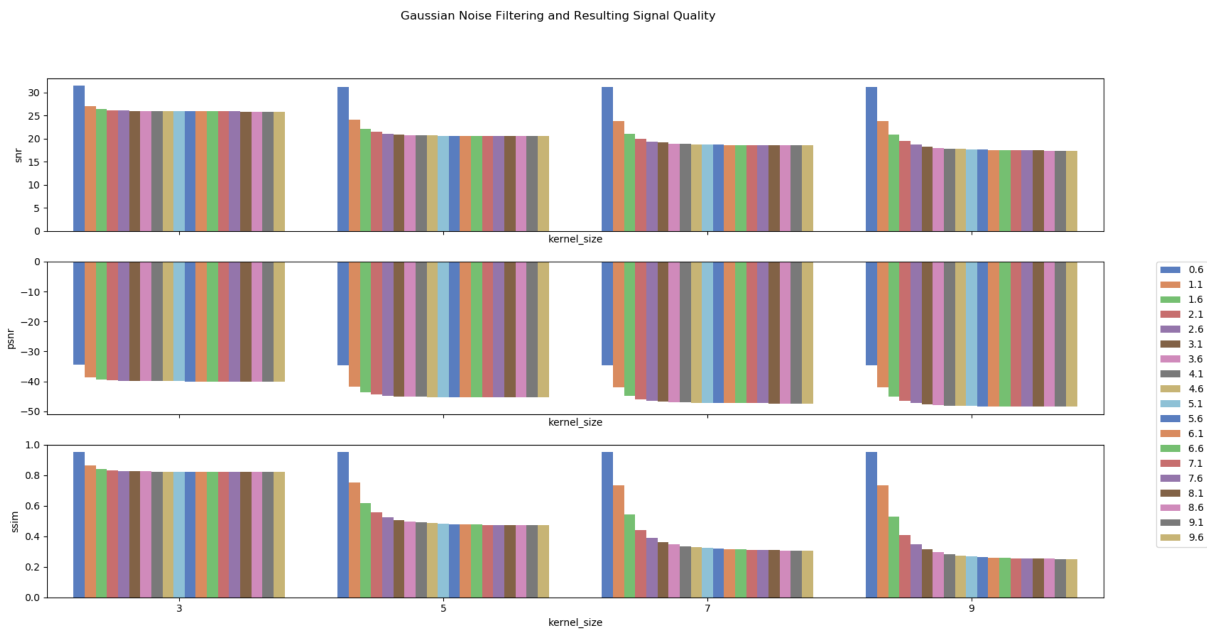


Figure 8. Choosing Filter Parameters

used for the results in Figure 9. Results based on filtering do not differ significantly between histograms of the filtered images. However, the 9x9 based filtered image is more noticeably blurry compared to the images filtered with smaller neighbourhoods.

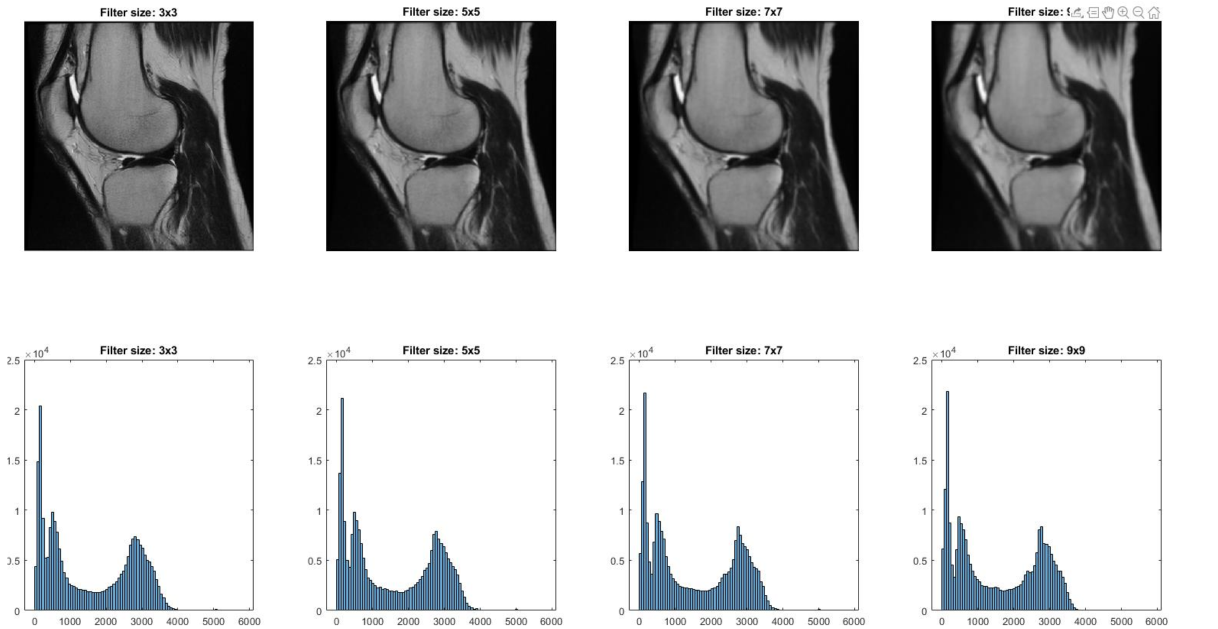


Figure 9. Filter Results

(c) Choose an image quality metric from class and plot it for the original and filtered images. Show and justify the metric you chose, and explain how it describes what is happening.

The quality metrics that were chosen to quantity image quality were SSIM and SNR. SSIM quantifies the perceptual difference between the original image and the filtered one whereas SNR measures the level of the original image to the sums of the squared differences between the original and filtered image. Looking at the specific sigma value, the graph below depicts the SSIM and PSNR as kernel size changes. For smaller neighbourhoods such as 3 x 3 and 5 x 5, the SSIM and SNR value are greater. As the kernel size increases, more global information is being filtered by the Gaussian due to the larger kernel. For smaller neighbourhoods the kernel is being applied more locally. This is why as kernel size increases the SSIM and SNR decreases.

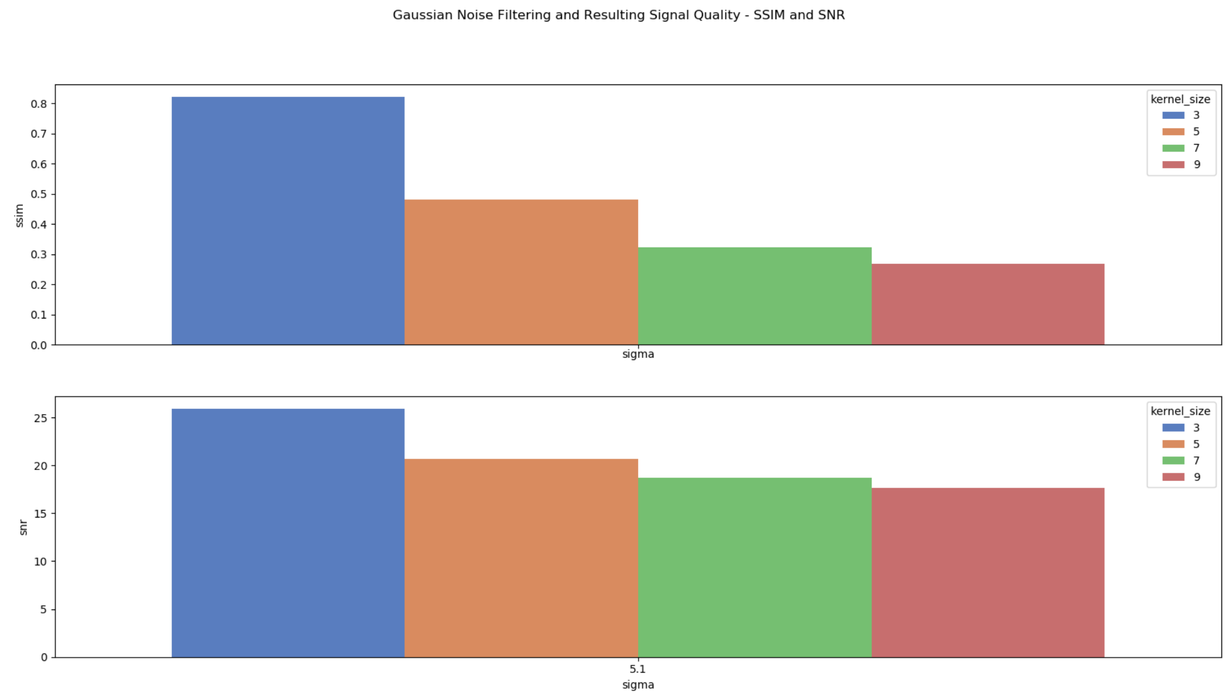


Figure 10. Quality Metrics

(d) Apply the edge detector you developed on (a) on each of the filtered images. Show and discuss your results.

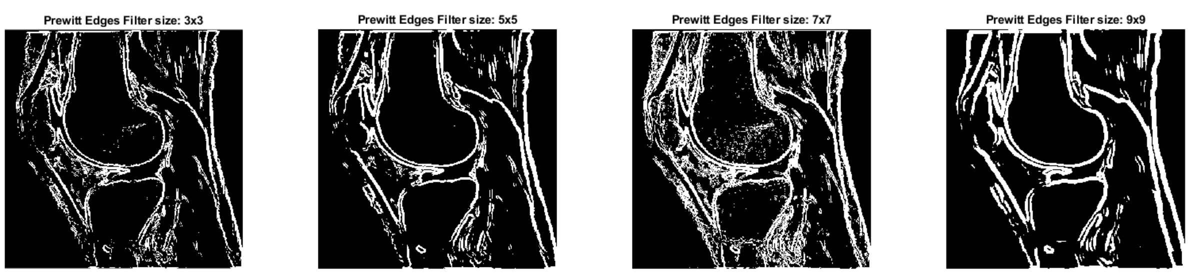


Figure 11. Edge Detection Using Prewitt Mask

The figure above depicts the edges of the images filtered at various neighbourhood sizes. For the smaller neighbourhood because of the fine grain edge detection the edges show more detail of the image. For larger neighbourhoods, the edges are thicker due to the smoothing effect the larger filter has on the input image. Because Otsu's threshold is used to binarize the images, a different threshold is used dependent on the gradient magnitude. For the 7x7 filtered image, some fine edge content or high frequency information is still captured compared to the 9x9 filtered edges despite the larger neighbourhood size.

(e) Using "path.tif" (RGB image), perform edge detection on each of the grayscale channels, and choose an appropriate method to generate an edge map. Describe (and justify) your algorithm and show all results. Is it adequate to detect edges in colour images this way? Why or why not?

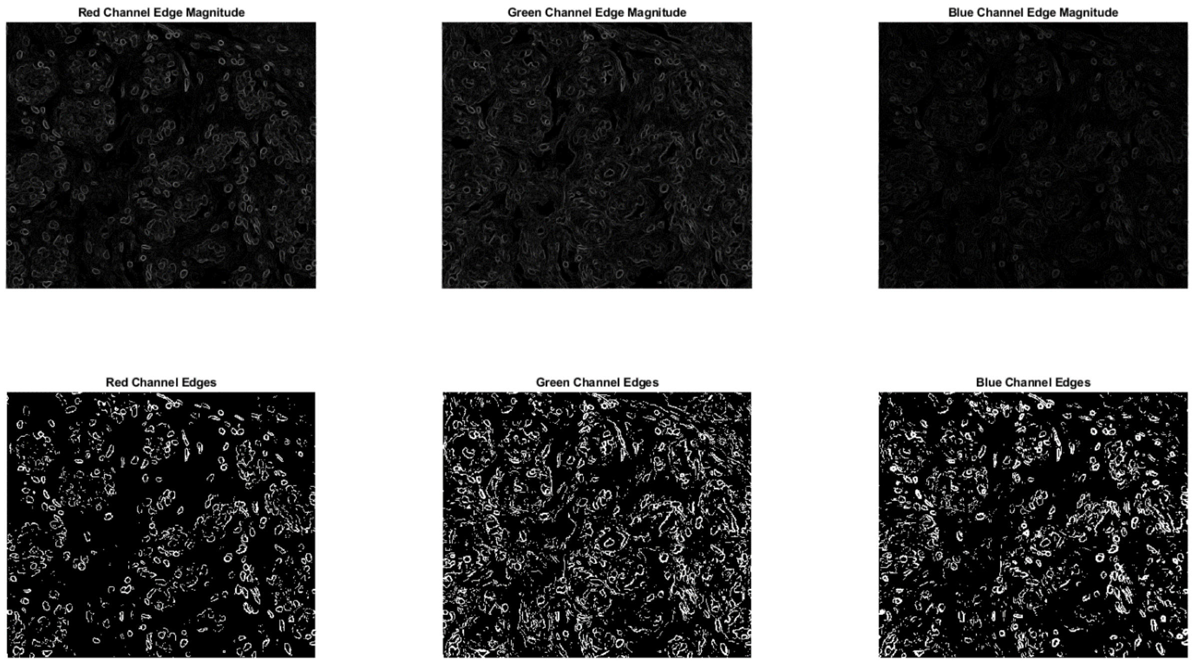


Figure 12. Edge Detection Using Prewitt Mask on Pathology Image

To obtain the images above, the algorithm in question 5a was applied to the R, G, and B, channels individually. The *best* edge map depends on the goal of visualization. The red channel gradient depicts gradients related to the nuclei, whereas the green channel gradient depicts gradients related to both the nuclei and stroma. Dependent on the goal, various thresholds can be used to capture the edges of individual structures or all structures. If the goal was to capture edges related to both nuclei and stroma, the best channel could be the green, as it provides gradient information on both. If the goal was specifically for nuclei edges, the red channel would be best as nuclei gradients are more prominent. In general, finding gradient magnitudes for individual channels may not be the best method to obtain edges, as RGB images contain correlated colour. This means that colour information is correlated across channels and edge information can be found through this correlation – "colour edges".

6. Vector Denoising for Colour Images [10 marks]

Write your own Matlab functions to perform colour vector denoising on "path.tif" using two vector methods of your choice. Choose a method to measure image quality.

(a) Explain briefly each filter and the pseudocode to implement the filters.

Vector Median Filter #1

for i = 1: horizontal dimension of image
  for j = 1: vertical dimension of image
    extract region in neighbourhood, N x N
    iterate through rgb vectors of neighbourhood:
      replace triplet with $\overrightarrow{rgb}(\text{argmin} \sum \|\|f_i - f_j\|\|)$ (L2 Norm)

(b) Apply the filters to the image using filter parameters of your choice. Show the resultant images and list the filter parameters used.

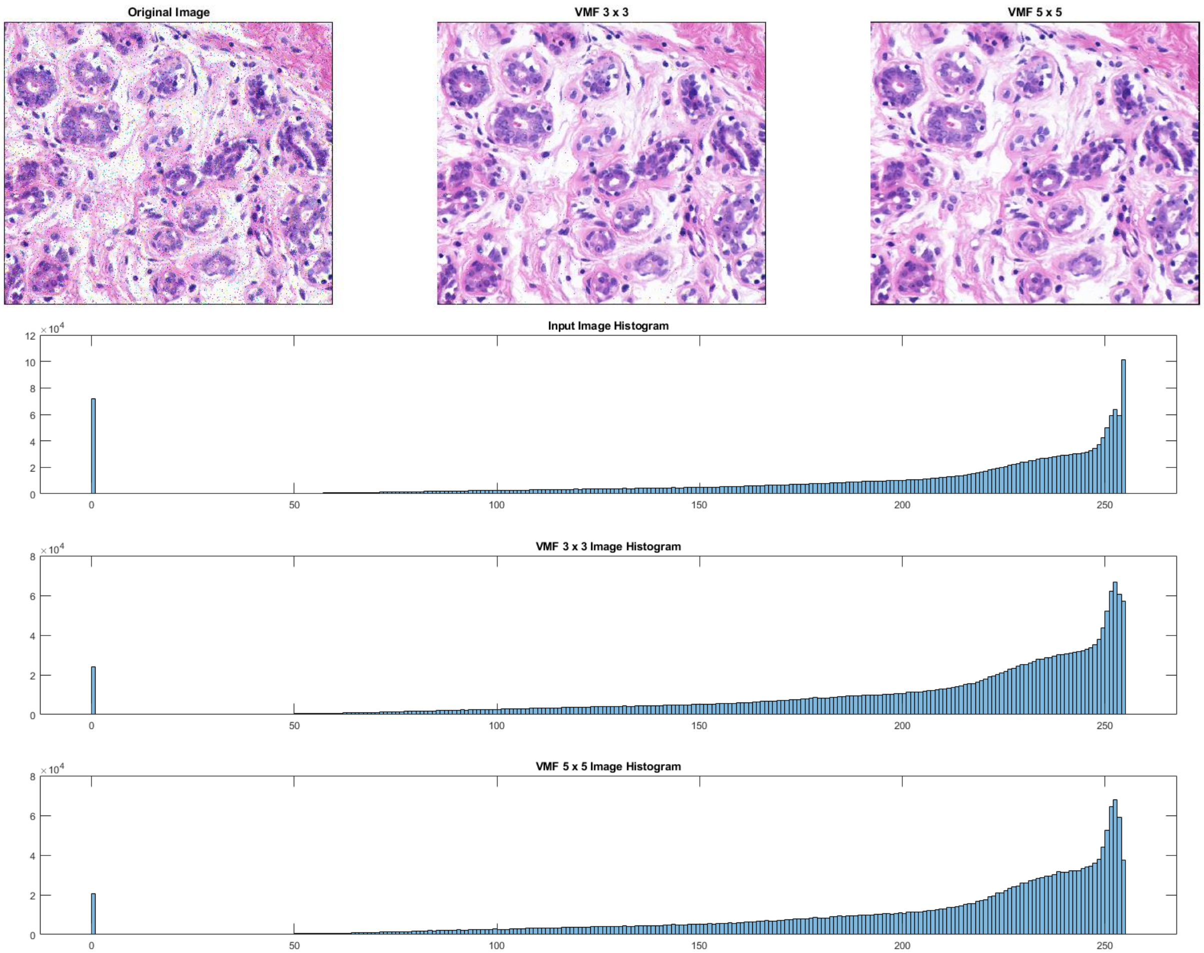


Figure 13. VMF Filtering Results

Vector Median Filter Alpha trim
for i = 1: horizontal dimension of image
for j = 1: vertical dimension of image
extract region in neighbourhood, N x N
iterate through rgb vectors of neighbourhood:
Find the magnitude of each vector
Sort the magnitude vector
Trim the magnitude vector by N samples at the front and tail of the sorted vector
Return the median of the remaining vectors

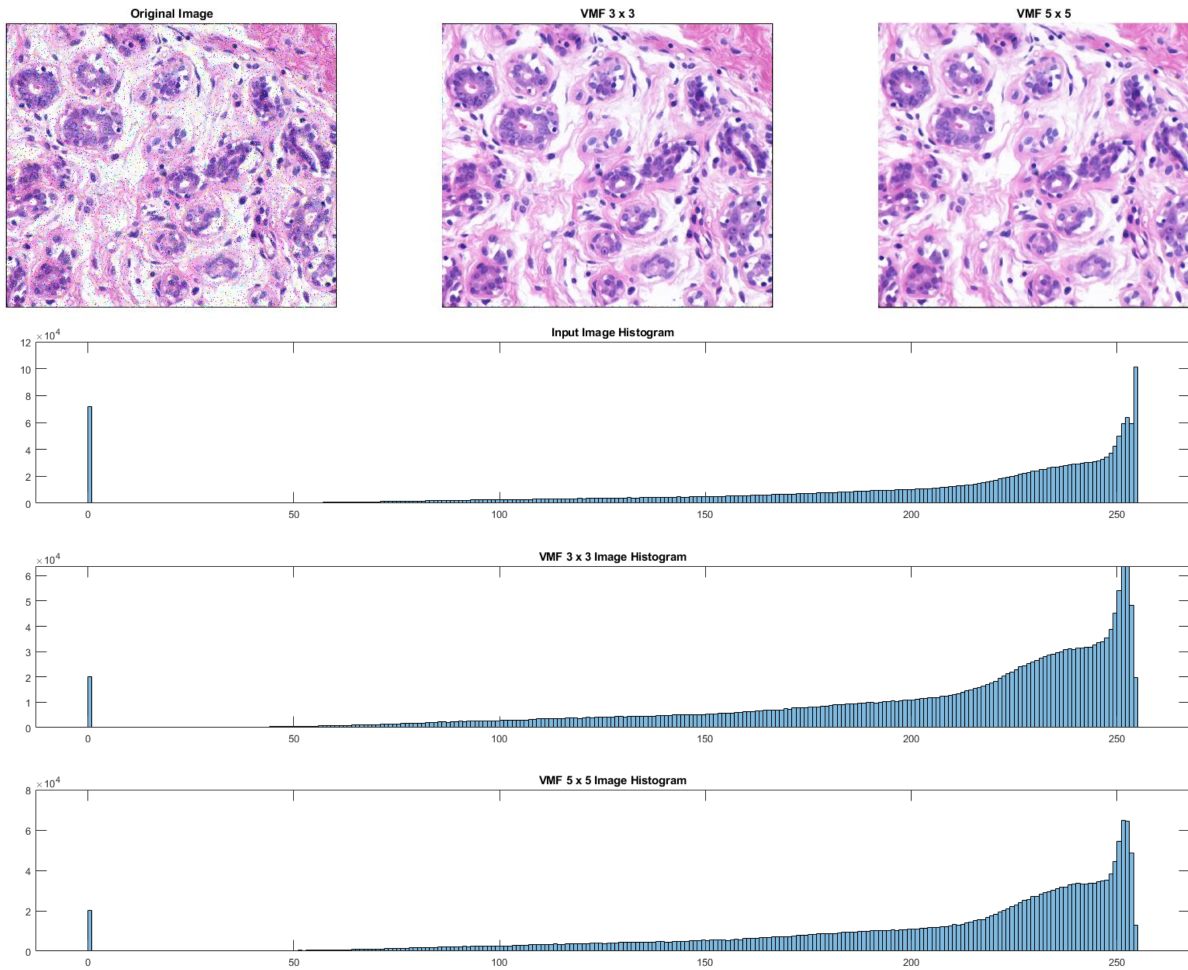


Figure 14. VMF- Alpha Trim Filtering Results

(c) Based on visual and quantitative analysis (image quality), which filter is performing better? Why?

Vector Median Filter			
Filter Size	**SNR**	**PSNR**	**SSIM**
3 x 3	22.0256	-24.5009	0.8540
5 x 5	19.9455	-26.5810	0.8204
Vector Median alpha trim			
3 x 3	21.0880	-25.4385	0.8822
5 x 5	17.4765	-29.0500	0.7726

Based on the qualitative results, the alpha trim vector for the 3 x 3 and 5 x 5 neighbourhoods removed more *salt-and-pepper* noise than the 3 x 3 and 5 x 5 VMF. Quantitatively based on the metrics in the table above, the VMF alpha trim 3 x 3 filter maintained the highest similarity to the noiseless image as indicated by the SSIM, and achieved comparable results to the *vanilla* VMF filter. Even though the SNR and PSNR are better for the VMF, obvious salt-and-pepper noise is still present in the resulting VMF image, whereas, the alpha trim VMF depicts minimal noise.

(d) Vary the window size for one of the vector filters and show the results. What happens to noise and edges when you increase the size of the filter? What are the advantages and disadvantages? Suggest a way to choose optimal parameters via pseudocode.

The results above depict results between window sizes of 3 x 3 and 5 x 5. As the window size gets larger, *more global* image information is included at each window calculation. If the window size is too large, this can cause blurring at edges even though noise is removed. At smaller window sizes, more local information is applied at the filter location, which can attenuate noise, but at the same time maintain edge content. To choose optimal parameters, a quality metric can be chosen to evaluate the effect of the filter. Similar to my method in question 5, neighbourhood size is chosen by iterating over various sizes and choosing sizes which minimize or maximize the chosen quality metric. The choice of the minimum or maximum value is dependent on what the quality metric is evaluating, ie. similarity, difference, error etc.

Selecting Neighbourhood size
Given neighbourhood size in *N = [N1, N2, N3, N_n]* for neighbourhood, index in enumerate(N):   value (index) = quality_metric(filtered_N, original) optimal_idx = argmin (value) N = corresponding neighbourhood(optimal_idx)