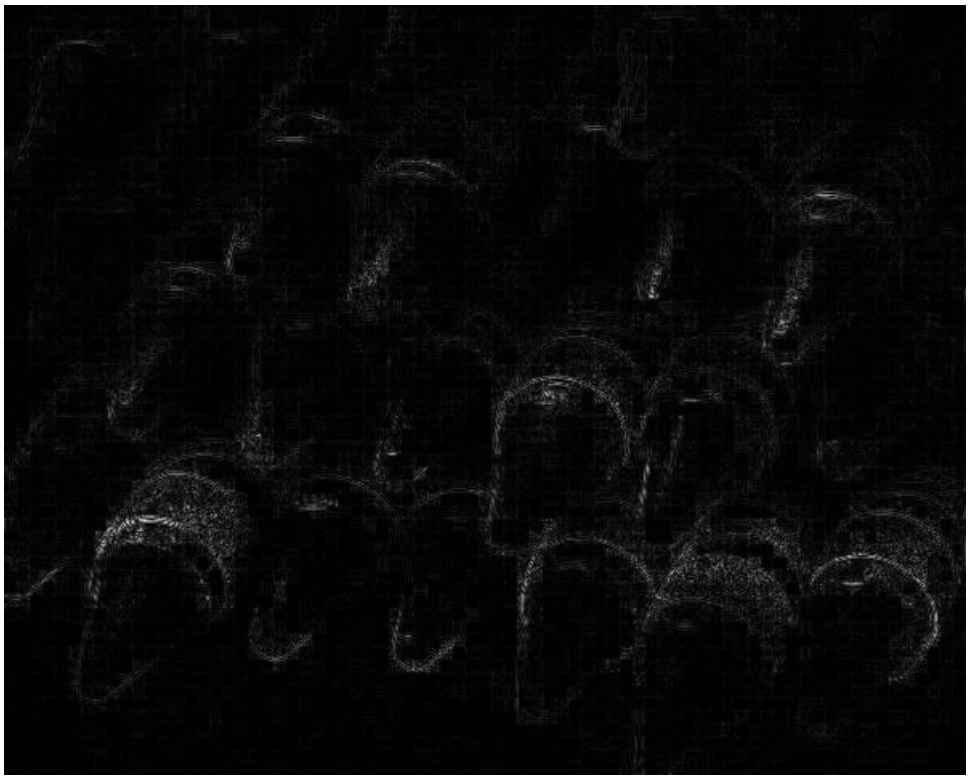
Name : Yan Zhang

Part-1 : Linear Interpolation

1) Insert your linear interpolated test image(hope.jpg) here:

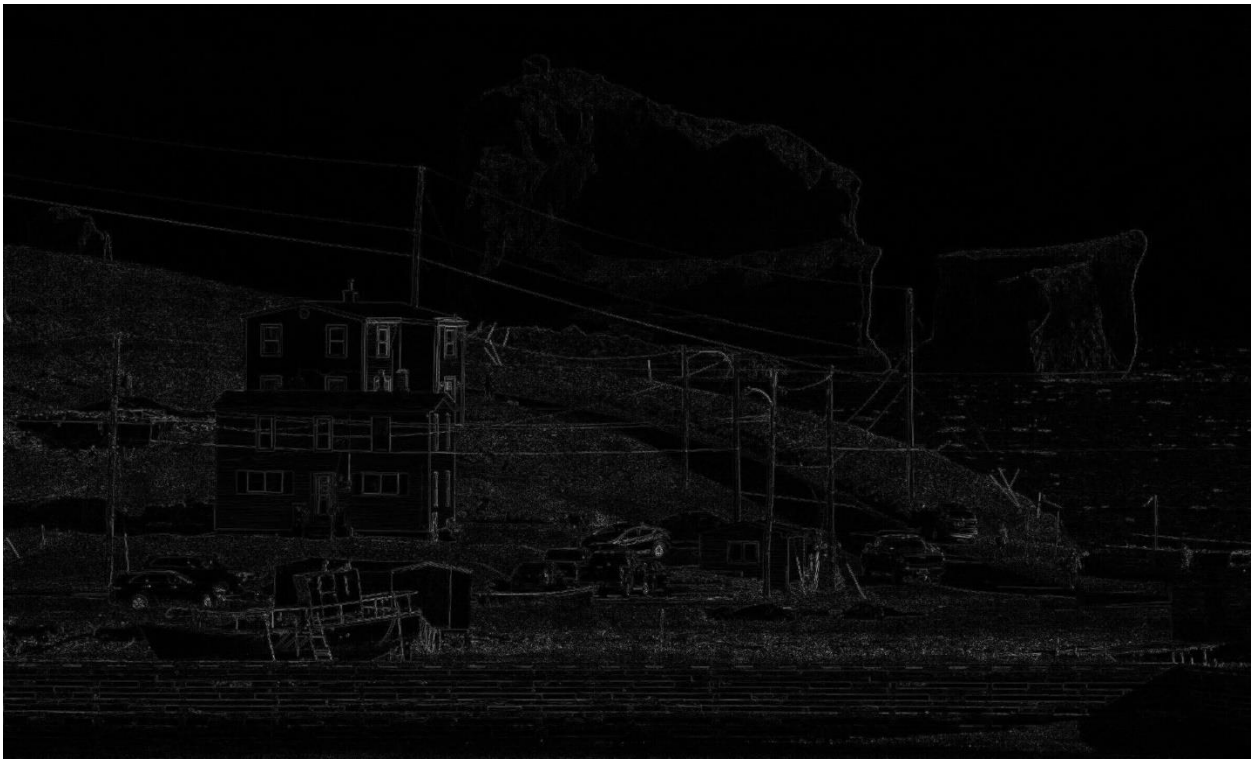
2) Display the map/plot of all the 3 training images here:

Crayons solution image



Crayons maps of errors

Iceberg solution image



Iceberg maps of errors

Tony solution image

Tony maps of errors

3) Post close-up of any artifacts you came across.

1. Crayons:

The outer edge is blurred.

Original Image

Solution Image

2. Tony:

The soil color is lighter than that in original version

Original Image

Solution Image

The hair-sky boundary is more blured in solution image than in original image.

Original Image

Solution Image

3. Iceberg:

The white vertical wall in solution image is somewhat blurred than in original image.

Original Image

Solution Image

4) Average_per_pixel error and Max_pixel_error for each of 3 training images :

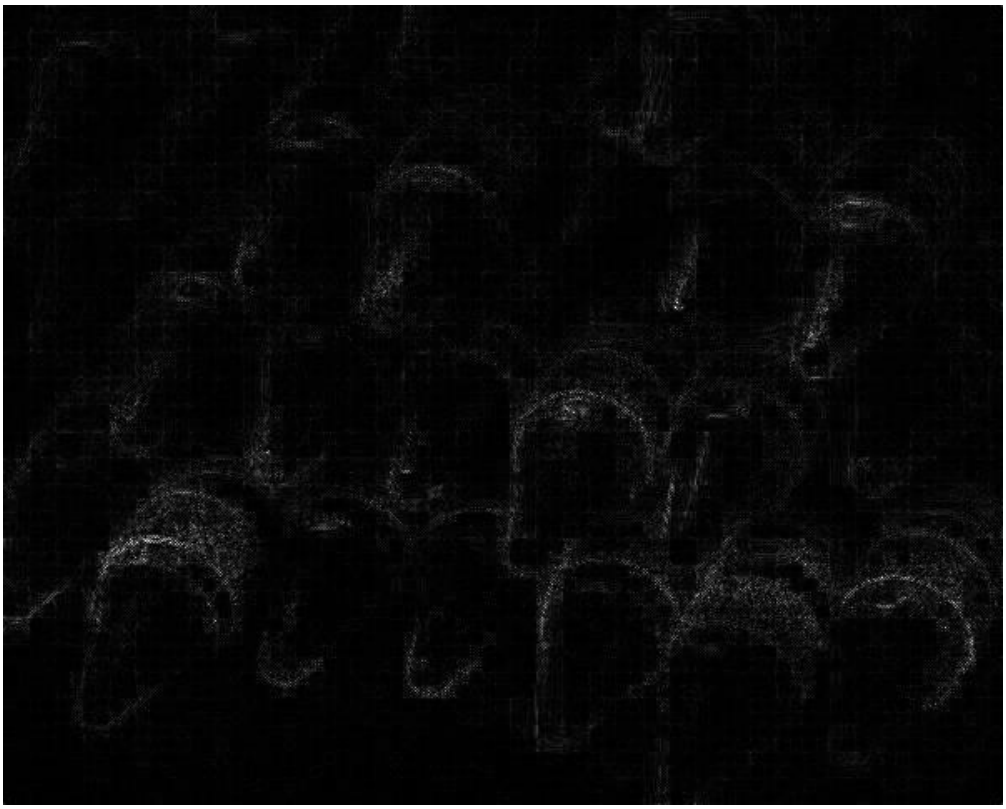
Image	Average_per_pixel_error	Max_pixel_error
Crayons	151.91146419270834	53478.125
Tony	23.362575130208334	9817.8125
Iceberg	105.2289647216683	30197.3125

Part-2 : Freeman Method

5) Insert your Freeman Method test image(hope.jpg) here:

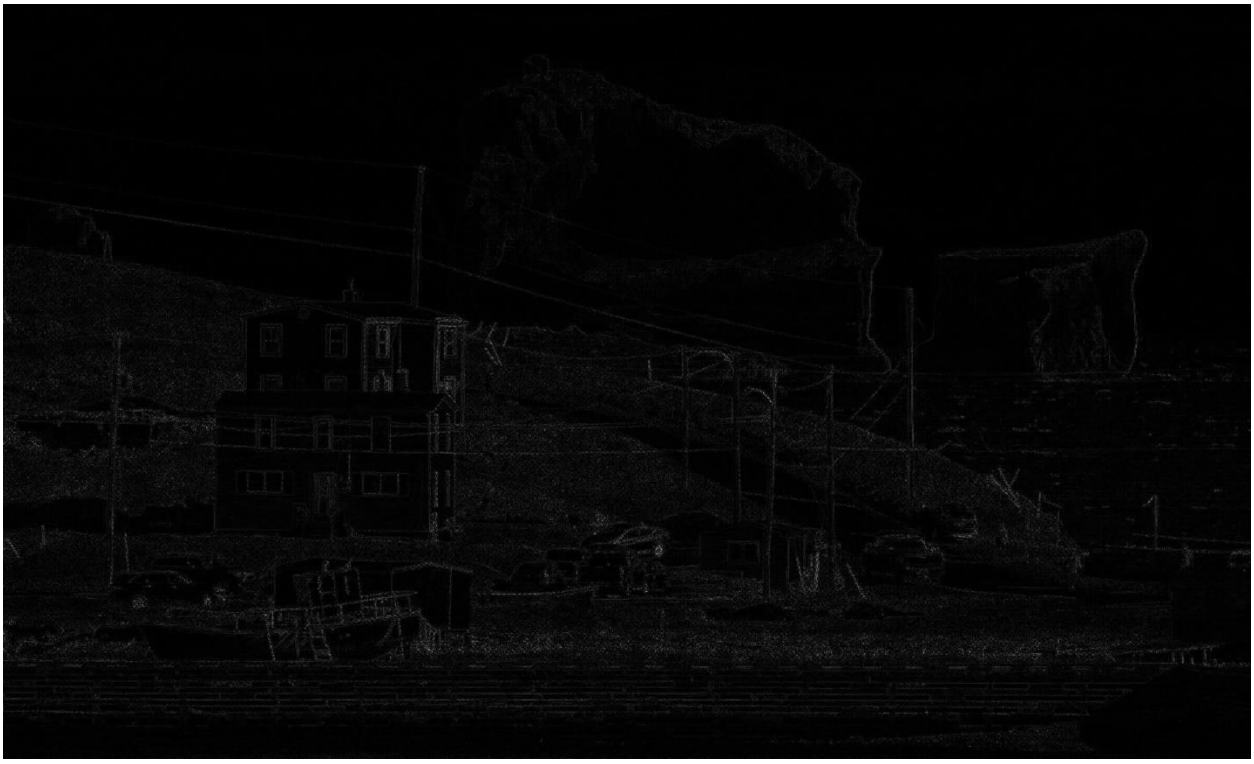
6) Display the map/plot of all the 3 training images here:

Crayons Freeman solution image



Crayons maps of errors

Iceberg Freeman solution image



Iceberg maps of errors

Tony Freeman solution image

Tony maps of errors

7) Post close-up of any artifacts you came across.

1. Crayons:

The crayon head color appears more vibrant in original image than in solution image.

Original Image

Solution Image

2. Iceberg:

The color contrast of the vertical wall between the shade side and bright side is larger in original image than in solution image.

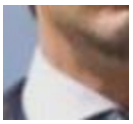
Original Image

Solution Image

3. Tony:

The collar-neck boundary is more blurred in solution Image than in original image.

Original Image



Solution Image

8) Average_per_pixel error and Max_pixel_error for each of 3 training images :

Image	Average_per_pixel_error	Max_pixel_error
Crayons	112.34364149305556	47857.625
Tony	15.508873828125	10873.5
Iceberg	67.79525049342872	33720.0625

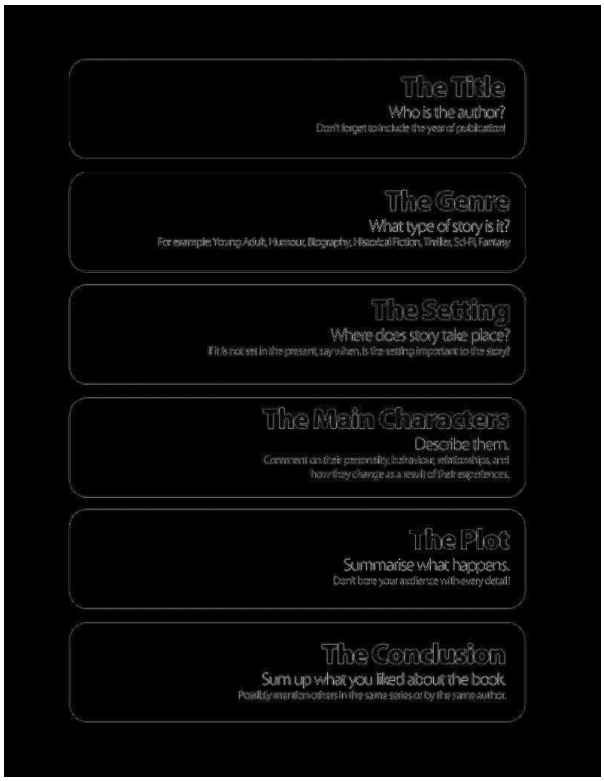
Part-3 : Images of your choice

1) Post 2 images your choice here and the corresponding error maps of your outputs with the Freeman method.

Original image 1

Fake "mosaic" image 1

Demosaiced solution image 1



Error maps

Original image 2

Fake "mosaic" image 2

Demosaiced solution image 2

Error maps

2) Any image that breaks the method and why do you think so?

The image 1 of my choice as above turns out to be blurred after demosaicing. To further clarify such effect, the red strips from both original and demosaiced images are zoomed in as below:

Original

Demosaicing

We can easily find that the black characters within the red background is blurred, compared with the original ones. If we inspect all strips more closely, besides the red one, all the boundaries between the black character outlines and colorful background are somewhat blurred, and such effect appears most obvious within red strips.

I believe the underlying mechanism could be revealed by the Fake "mosaic" image 1. Because of the averaging effect by median filter, the pixel values contrast would be smoothed. Hence, around the character boundary regions with large pixel value contrast, the contrast would be decreased by the median averaging and thus the characters appear blurred.

Part-4 : Bonus

Post any extra credit details/images/references used here.

From the above experiments, we might find that, although both methods bilinear interpolation and Freeman can obtain good results in homogenous image regions, they are prone to severe demosaicing artifacts in regions with edges and details. Some more complex methods that interpolate independently within each color plane **include bicubic interpolation [1] , spline interpolation[2]**, and **Lanczos resampling**[3].

[1] R. Keys (1981). "Cubic convolution interpolation for digital image processing". IEEE Transactions on Acoustics, Speech, and Signal Processing. 29 (6): 1153–1160. doi:10.1109/TASSP.1981.1163711.

[2] https://tools.timodenk.com/?p=cubic-spline-interpolation

[3] Turkowski, Ken; Gabriel, Steve (1990). "Filters for Common Resampling Tasks". In Glassner, Andrew S. *Graphics Gems I*. Academic Press. pp. 147–165. CiteSeerX 10.1.1.116.7898. ISBN 978-0-12-286165-9.

More sophisticated demosaicing algorithms exploit the spatial and/or spectral correlation of pixels within a color image[4], including:

Adaptive Homogeneity-Directed (AHD) selects the direction of interpolation so as to maximize a homogeneity metric, thus typically minimizing color artifacts[5].

Variable Number of Gradients (VNG) interpolation computes gradients near the pixel of interest and uses the lower gradients (representing smoother and more similar parts of the image) to make an estimate[6].

Here I implemented **Bicubic interpolation method** for improvement:

Difference between Bi-linear and Bi-cubic:

1. Bi-linear uses 4 nearest neighbors to determine the output, while Bi-cubic uses 16 (4×4 neighbourhood).
2. Weight distribution is done differently.

As can be seen from the performance comparison as below, bicubic interpolation method produces a sharper image than the other 2 methods, as illustrated by the black characters in the red background. This method balances processing time and output quality fairly well.

[4] Chang, Lanlan. Hybrid color filter array demosaicking for effective artifact suppression[J]. Journal of Electronic Imaging, 2006, 15(1):013003.

[5] Hirakawa, Keigo, and Thomas W. Parks. "Adaptive homogeneity-directed demosaicing algorithm." IEEE Transactions on Image Processing 14.3 (2005): 360-369.

[6] Ting Chen. "Interpolation using a Threshold-based variable number of gradients".

Bilinear Interpolation

Freeman Interpolation

Bicubic Interpolation