Programming Assignment 1

COMPUTATIONAL PHOTOGRAPHY-EE5176

PRAVEENPRASATH B
EP21B028
INDIAN INSTITUTE OF TECHNOLOGY, MADRAS

Problem 1 : Demosaicing

Demosaicing is a process of interpolating the red, green and blue channels from the raw pixel image. Here we will try to fill in the missing pixel values from the sampled raw images through interpolation. Kindly zoom in and view RawIm- age1.mat to get an idea on how raw image looks like.
The CFA for 'RawImage1.mat' is 'RGGB' whose mask is provided in the file 'bayer1.mat'. In 'bayer1.mat' a value of 1 at a particular pixel means that only 'red' value is sampled at that pixel.Similarly values of '2' and '3' mean that only 'green' and 'blue' values are sampled at those pixels.The task here is to reconstruct the full colour image from the undersampled raw sensor data. Two popular ways of filling in missing data on a 2D grid are bilinear and bicubic interpolation. Here,we will try both the methods and compare the results.
Note that you have to interpolate only the missing values in the grid of pix- els. For instance,at a pixel where the channel 'blue' is already sampled you need to interpolate only the 'red' and 'green' values from the image grid. Use Matlab built-in function griddata to query all missing pixel values at once, instead of querying it patchwise.

Part a :

Perform bilinear interpolation of the missing pixels in each color channel R, G and B to reconstruct a full color image from RawImage1 with the CFA provided in 'bayer1.mat' [5 marks]

Figure 1: Raw image 1

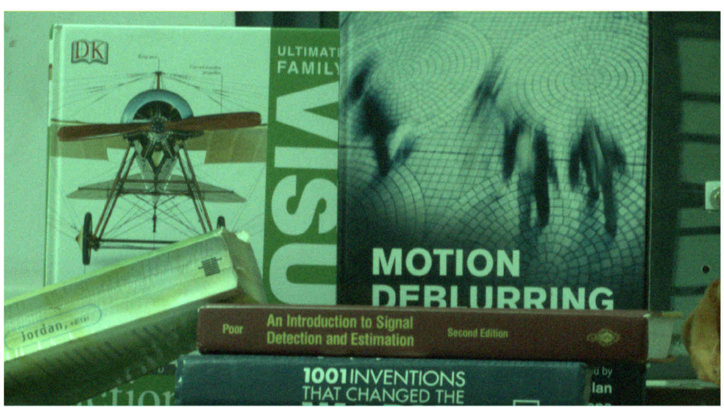


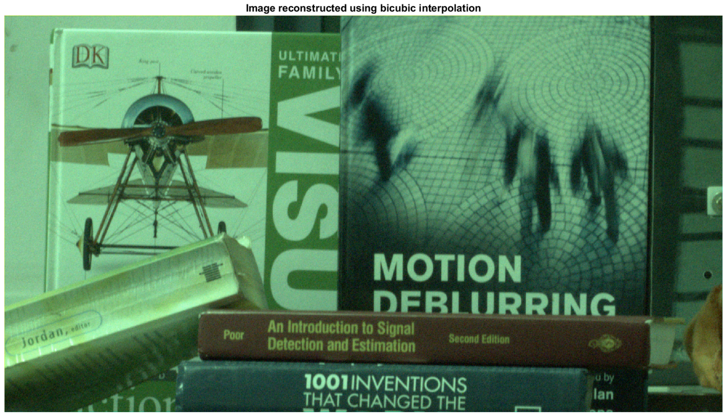
Figure 2: Image reconstructed using Bi-linear interpolation

Part b :

Perform bicubic interpolation of the missing pixels in each color channel R, G and B to reconstruct a full color image from RawImage1 with the CFA provided in ‘bayer1.mat’. Compare the reconstructed full colour image with that obtained using bilinear interpolation. [3 marks]

Solution:

Image reconstructed using bicubic interpolation

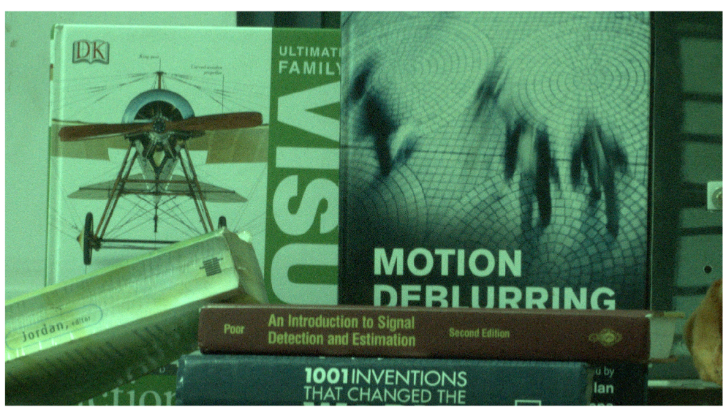


Bilinear interpolation only considers the closest 2x2(4) pixel neighbourhood for interpolation but Bicubic interpolation considers 4x4(16) pixel neighbourhood which results in a sharper image.

Part c :

Do demosaicing for RawImage1.mat using pattern ‘rggb’ with matlab built-in function ‘demosaic’ and compare with previous two reconstructed im- ages [2 marks]

Solution :



Demosaiced image used inbuilt function

Raw image and Demosaiced images using various techniques

Part d :

What assumptions are you making while interpolating the missing pixel values and what will happen when the assumptions do not hold? [2 marks]

Solution:

We assume that the chrominance of the image changes transiently not abruptly and channel intensity of each pixels depends on its neighbourhood pixels. When assumption does not hold we can see moire patterns in which fine black and white details are interpolated as colour information by interpolation.We can use median filter to help with colour artifacts.

Part e

Perform bicubic interpolation on'kodim19.mat' which has been sampled with the CFA pattern 'RGGB' (provided in kodim cfa.mat) to reconstruct a full color image. [2 marks]

solution:

Bicubic interpolated image

Part f:

Now convert the RGB image obtained in part (e) to YCrCb color space and median filter (choose appropriate filter size) the chrominance channels. Next,Convert back the image into RGB space. [3 marks]

Solution:

Ycbcr kodim image and Median filter applied Image

Part g:

For reference the true colour image 'kodim19.png' is given in the data folder. Compare the demosaiced images obtained in part (e) and (f) with the actual image and comment your observations. (The image 'kodim19.png' is obtained from Kodak dataset) [3 marks]

solution:

Left:Original image, Center: Only Demosacied image, Right: Median filtered image

The Only Demosaiced image has a prominent moire pattern throughout the image especially noticeable on the white fence when compared to the original image. Median filtered image has a less prominent moire pattern compared to only demosaiced

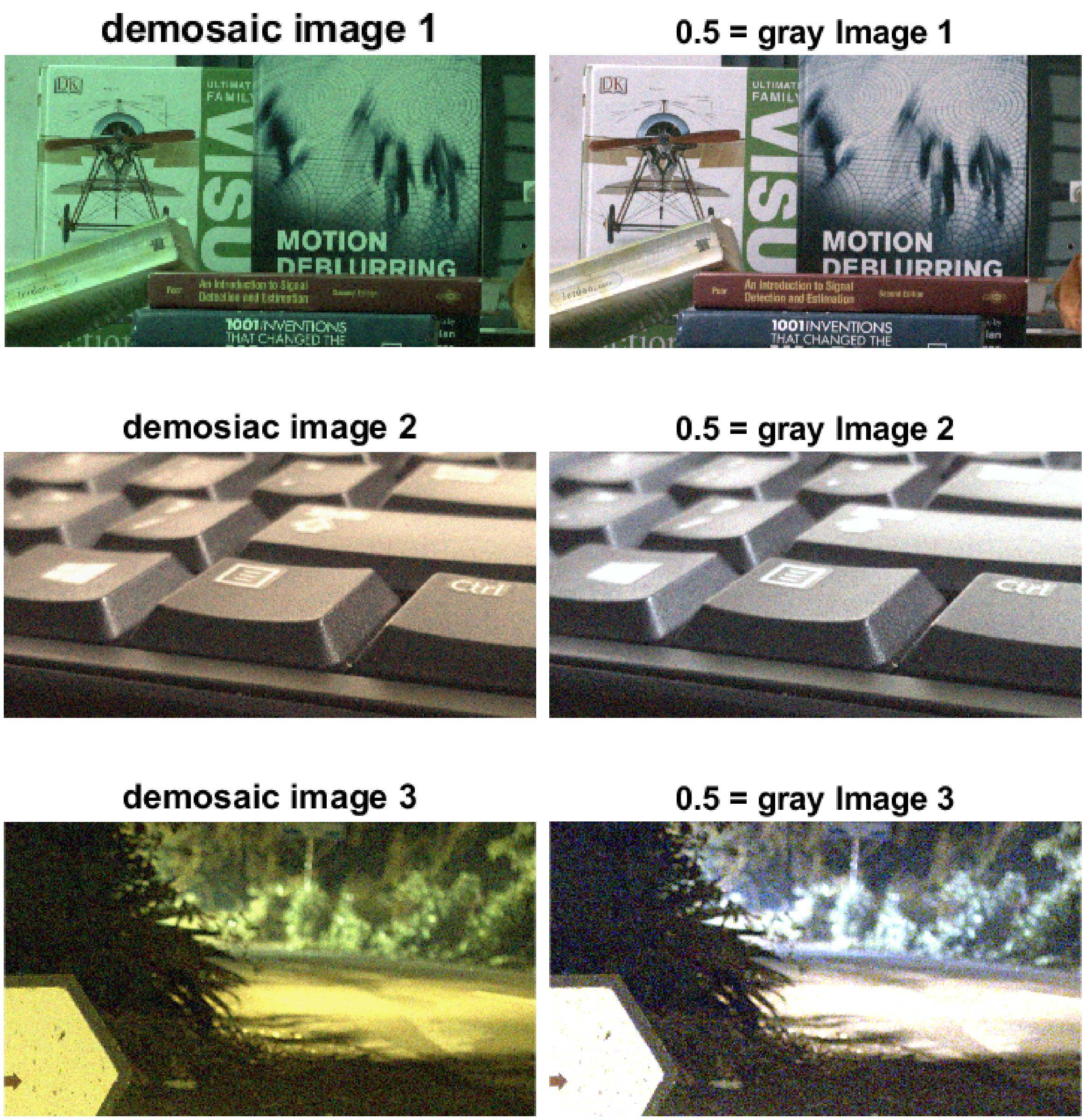
Problem 2: White Balancing and Tone Mapping

Next process in the pipeline is white balancing. Here you will do white-balancing for all three raw images RawImage1, RawImage2 and RawImage3 in each sub- section. Note that you have to perform demosaicing for RawImage2 and RawImage3 before white-balancing. The CFA for RawImage2 is 'grbg' (bayer2.mat) and for RawImage3 it is 'rggb'(bayer3.mat). Use the code from the previous problem to demosaic the images RawImage2 and RawImage3.

Part a :

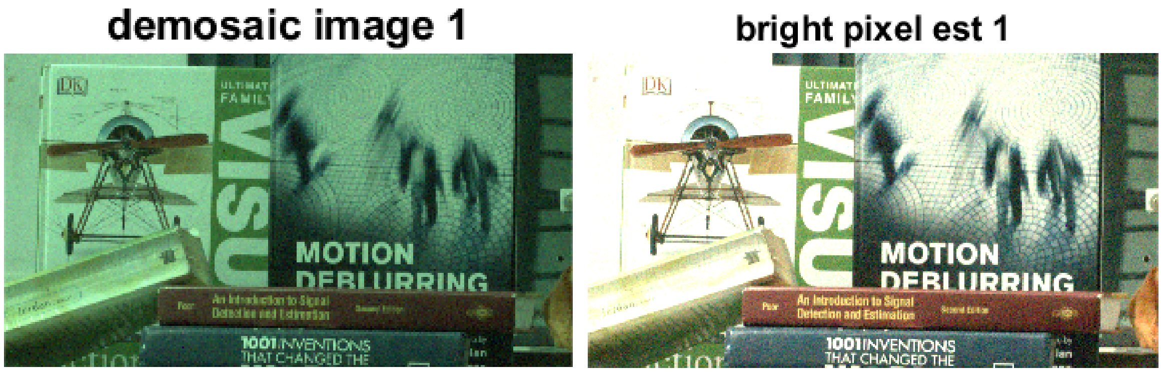
Assume the average color of the scene to be gray and then do white balanc- ing. [2 marks]

Solution:

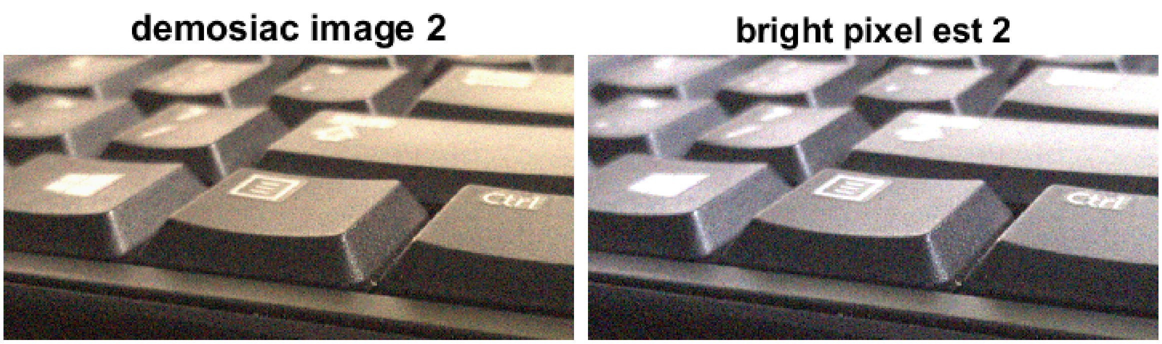


Part b:

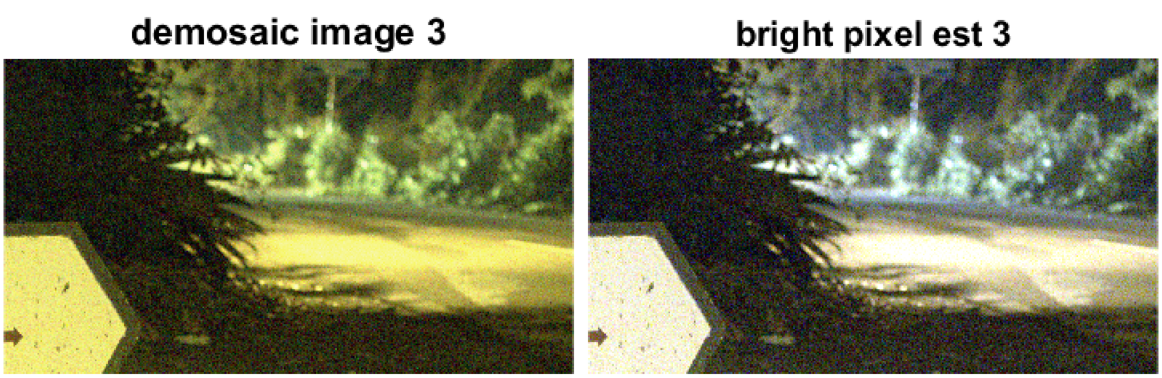
Assume that the brightest pixel to be specular highlight and should therefore be white. Use pixel coordinate (x, y) = i) (830, 814) for RawImage1 [1 mark]



ii) (1165, 280) for RawImage2. [1 mark]

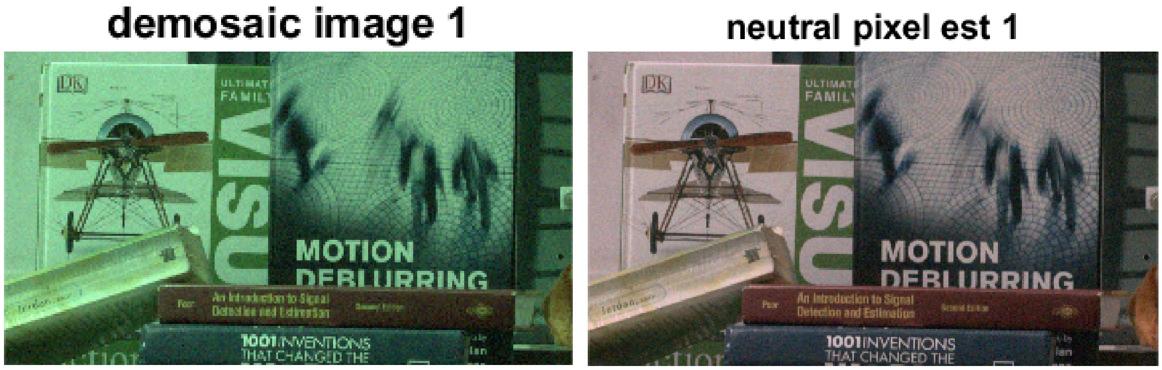


iii) (175, 675) for RawImage3. [1 mark]

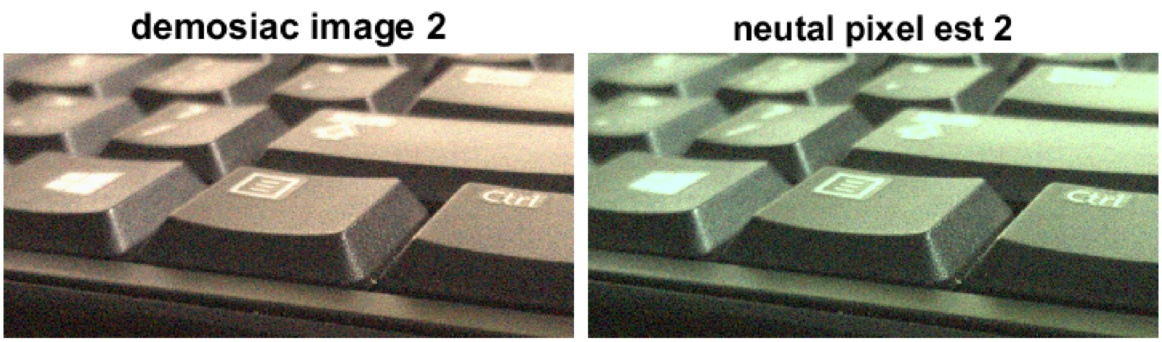


part c:

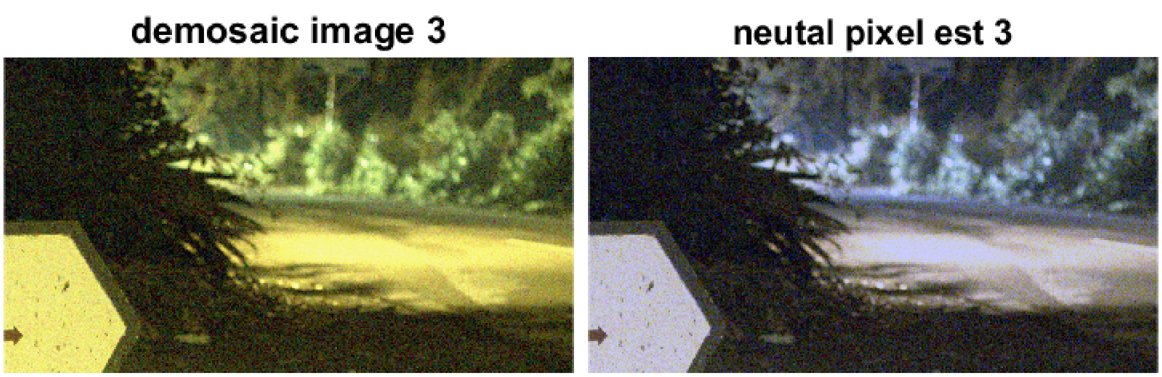
Assume that some part of the scene to be neutral. Use pixel coordinate (x, y) =
i) (2000, 435) for RawImage1 [1 mark]



ii) (445, 715) for RawImage2. [1 mark]



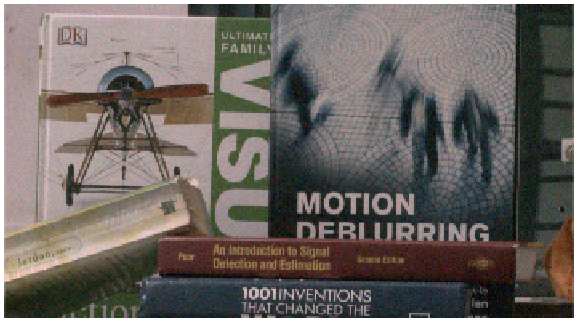
iii) (1550, 565) for RawImage3. [1 mark]



0.1 Part d:

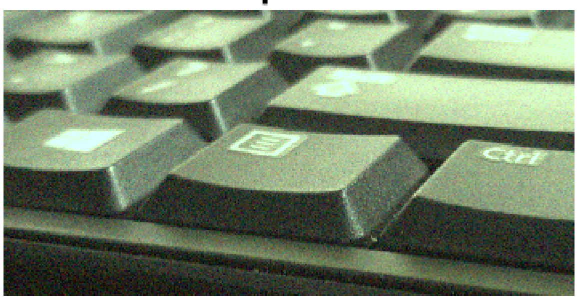
Perform tone mapping on each of the above images using the following two methods:
i) Histogram equalization [2 marks]

neutral pixel est 1



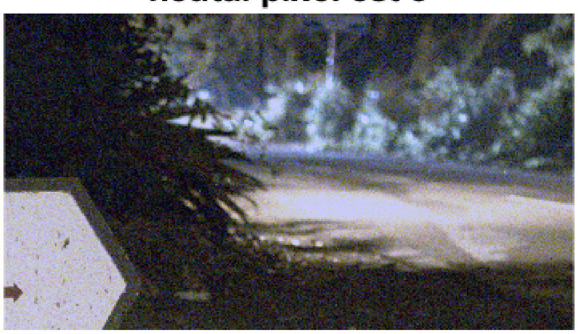
Histogram Equalization 1

neutal pixel est 2



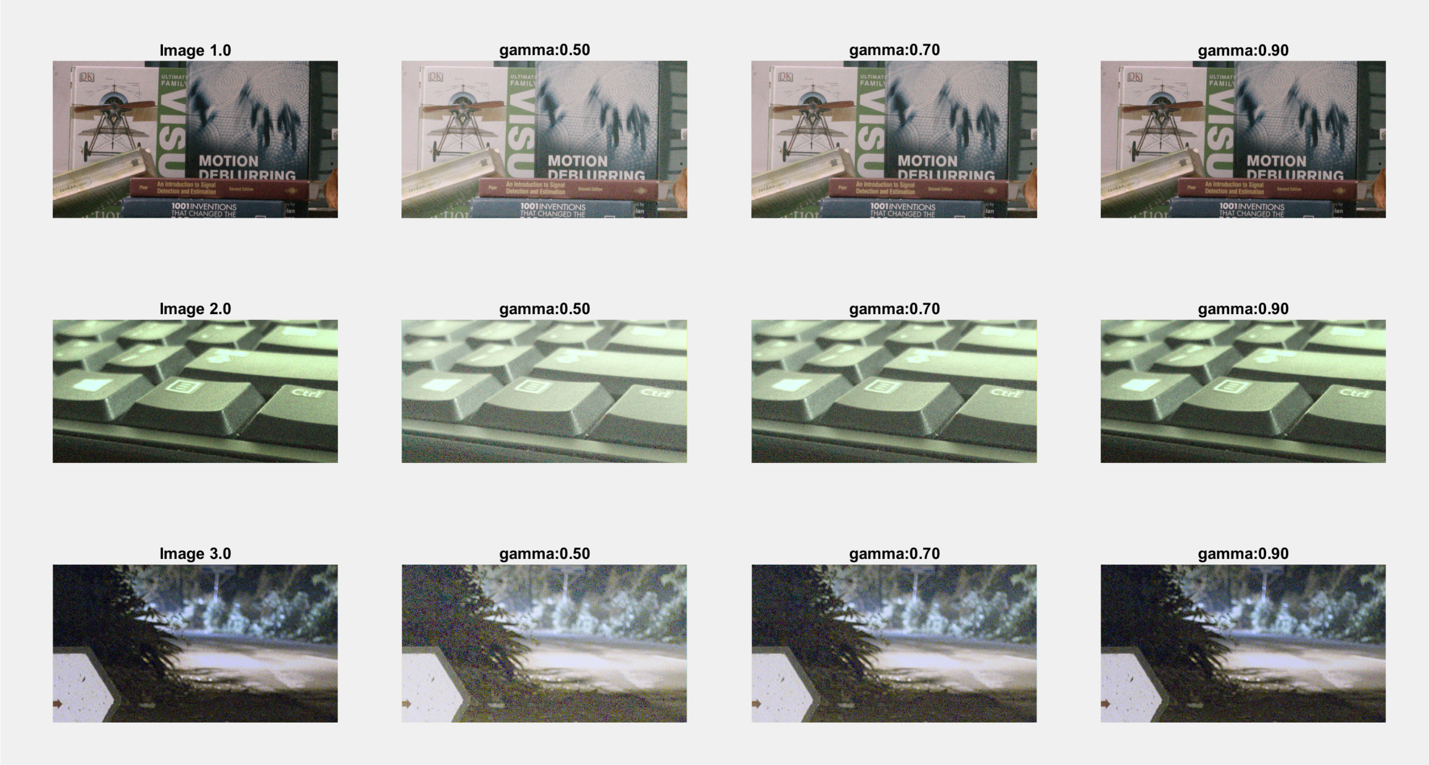
Histogram Equalization 2

neutal pixel est 3



Histogram Equalization 3

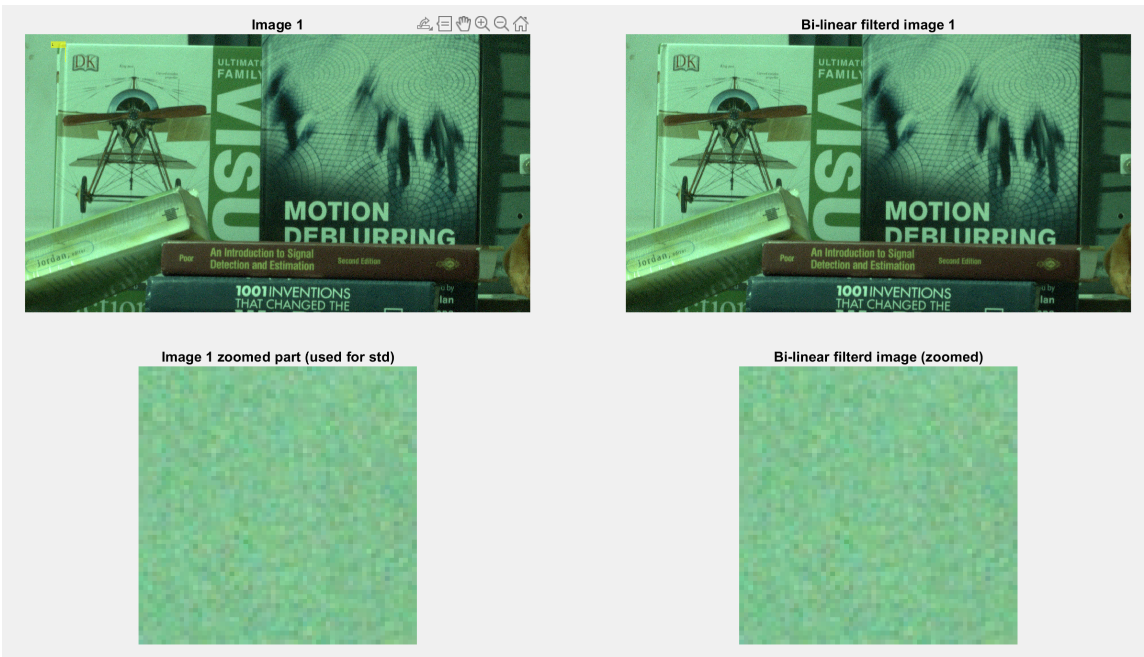
ii) Gamma correction for the gamma values of 0.5, 0.7 and 0.9 [3 marks]

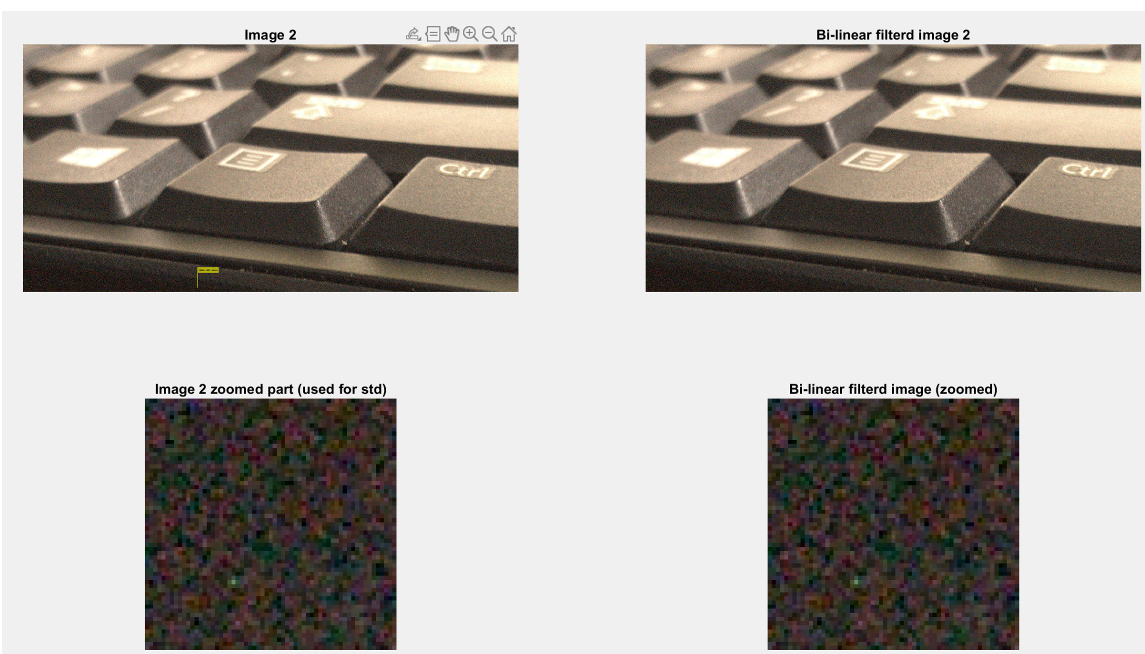


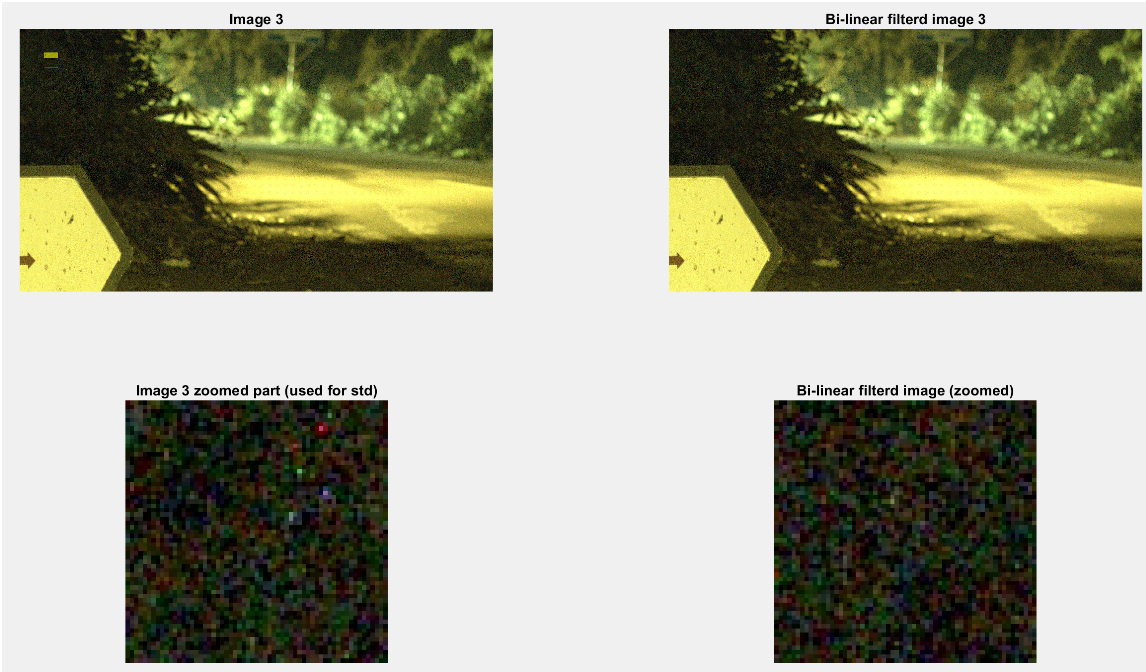
Problem 3: Image Denoising

Here in this part you will use bilateral filtering operation for denoising. The bi- two parameters s and r controls lateral filter. As we know, s controls the spatial weighing whereas the r controls the alignment of image intensi- ties. Ideally we would like to locally estimate the noise and adjust both the parameters accordingly but given the cost here we would go with single set of parameters. To estimate the noise we will choose a constant region in the image (coordinates are given below) and calculate the standard deviation along all the three channels R, G and B. Now denoise individual channels with window size as 11, s as 2.5 and for r = 1.95×n, where n is noise standard deviation in the corresponding colour channel. For the bilateral filtering, use the supplied function bfilter2.m.

For RawImage2 use the rectangular region with coordinates: top-left (705,924) and bottom-right (765,984) in (row, column) order. For other raw images choose a region on your own and do the denoising as mentioned above. [5 marks]







Problem 3.1

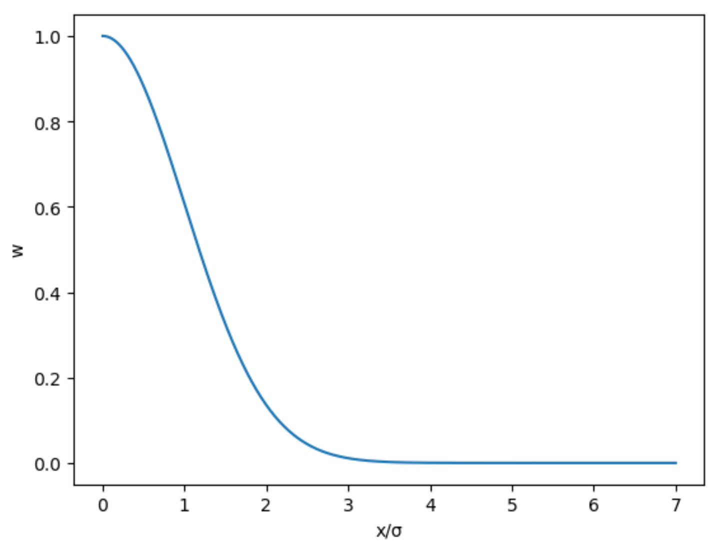
1. You have to construct a Gaussian blurring kernel with a variance for denoising an image. As you know, the Gaussian function has infinite support. In order to implement tractability, you need to truncate the Gaussian function to a finite size. For a given Gaussian kernel with variance , at what point will you choose to truncate the function and why? [2 marks].

solution:

Gaussian Blurring kernel has a weightage of the form

$$W = e^{x^2/2\sigma^2}$$

This function has infinite support. The plot of w vs ratio of x/σ follows.



At $x/\sigma = 3$ then w = 0.011 which is very small and pixels in this range will not contribute much to the pixel value, so we can truncate at $x = 3\sigma$.

Part 2 :

For a given n how would you choose the value for r ? What would happen if r n or r n? [3 marks]

Solution:

Parameter σ_r estimates weightage given for pixels by the difference of their intensities.We should choose σ_r slightly larger than σ_N so that the filter perform smoothing noise while preserving feature in the image.

When $\sigma_r << \sigma_n$, The noise is still very much prevalent in the image and filter does not do good job of suppressing the noise.

When $\sigma_r >> \sigma_n$. The filter blurs out even the features in the image resulting in loss of large o amount of information.

1 Problem 4: Depth of Field

Depth of field (DOF) is the distance between the nearest and the farthest ob- jects that are in acceptably sharp focus in an image i.e. depth range around the focal plane that produces a circle of confusion. We are working with the Depth of field (DOF) is the distance between the nearest and the farthest ob- jects that are in acceptably sharp focus in an image i.e. depth range around the focal plane that produces a circle of confusion. We are working with the FLIR BFS-U3-88S6C-C camera which has a Sony IMX267,CMOS, 1" sensor of resolution of 4096 2160 and each pixel is of size 3.45 m. Camera is attached with a lens of focal length 16mm. Let the diameter of the circle of confusion be 3 pixels. [10 marks]

a) For each of the following settings calculate the depth of the Field and its boundary distances:

$$\text{Depth of Field} = \frac{2WNv^2}{f^2}$$

where W = width of sensing = width of each pixel × Diameter of circle of confusion = 3.45 × 3 = $10.35\mu m$
N is f/#
v is object distance
f is the focal length

i) Aperture = f/2.2 and the lens focused at a distance of 3m

N = 2.2,v = 3

Depth of field $= \frac{2*10.35*10^{-6}*2.2*3^2}{(16*10^{-3})^2} = 1.6010$ m

ii)Aperture = f/4 and the lens focused at a distance of 3m

N = 4,v = 3

Depth of field $= \frac{2*10.35*10^{-6}*4*3^2}{(16*10^{-3})^2} = 2.9109$ m

iii) Aperture = f/4 and lens focused at a distance of 1m

N = 4,v = 1

Depth of field $= \frac{2*10.35*10^{-6}*4*1^2}{(16*10^{-3})^2} = 0.32343$ m

b) Compare the results obtained in part 'a' and comment your observations.

When we decrease the Aperture from (a) to (b), the Depth of field increases linearly.

2

When we decrease the object distance in (b) to (c). The Depth of field decreases.