Assignment 2: CS 663, Fundamentals of Digital Image Processing

Mikhil Gupta-203079016
Nitish Gangwar-203050069

September 1, 2021

Answer 1:

convolution takes an image $f(x)$ and kernel $w(x)$ as input and produces output $g(x)$, This 1D convolution operation can be expressed as:

$$g(x) = \sum_{u} w(u) f(x - u) \tag{1}$$

We have give the value of w as $(w_0, w_1, ..., w_6)$ as the 1D kernel, therefore $g(x)$ simplifies to,:

$$\begin{aligned} g(x) &= \sum_{u=0}^{u=6} w(u) f(x - u) && (2) \\ &= w(0)f(x) + w(1)f(x-1) + .. + w(6)f(x-6) && (3) \\ &= w_0 f(x) + w_1 f(x-1) + .. + w_6 f(x-6) && (4) \end{aligned}$$

Properties:

- similar to any convolution operatipn the 1D convolution is also time and space invariant
- As we convert the 2D convolution to 1D convolution the minimum computational comlexity is achieved
- It follows associative($(f * g) * h = f * (g * h)$)), distributive $(f * (g + h) = (f * g + f * h)$)as well as the commutative $(f * g = g * f)$ properties of the convolution.

Potential Application:

- 1D convolution is used in Real-time electrocardiogram (ECG) monitoring
- 1D CNNs have now been preferred over their 2D deep counterpart
- 1D convolutions are commonly used for time series data analysis
- Laplacian mask(-4in centre) kernel can be operated as the sum of two 1D convolution which is lot more computationally cheaper.

Answer 2:

Given the bicubic interpolation formula:

$$v(x,y) = \sum_{i=0}^{3}\sum_{j=0}^{3} a_{ij}x^i y^j \tag{5}$$

We know that, for point (x,y) around the unit square, the derivatives of $f(x,y)$ can be given as,

$$f_x = f(x+1,y) - f(x,y) \tag{6}$$

$$f_y = f(x,y+1) - f(x,y) \tag{7}$$

$$f_{xx} = f(x+1,y) - 2f(x,y) + f(x-1,y) \tag{8}$$

$$f_{yy} = f(x,y+1) - 2f(x,y) + f(x,y-1) \tag{9}$$

$$f_{xy} = f(x+1,y+1) - f(x+1,y) - f(x,y+1) + f(x,y) \tag{10}$$

Considering the 4 neighbouring points in unit square around $v(x,y)$ as (x_0,y_0),(x_0,y_1),(x_1,y_1), (x_1,y_0) .
Now matching the given points with the equation $v(x,y)$ and put the value ,
$v(x_0,y_0) = a_{00}$,
$v(x_1,y_0) = a_{00} + a_{10} + a_{20} + a_{30}$,
$v(x_0,y_1) = a_{00} + a_{01} + a_{02} + a_{03}$,
$v(x_1,y_1) = \sum_{i=0}^{3}\sum_{j=0}^{3} a_{ij}$
Now considering the first derivative with the neighbouring points in 1D
$v_x(x_0,y_0) = a_{10}$,
$v_x(x_1,y_0) = a_{10} + 2a_{20} + 3a_{30}$
$v_x(x_0,y_1) = a_{10} + a_{11} + a_{12} + a_{13}$
$v_x(x_1,y_1) = \sum_{i=1}^{3}\sum_{j=0}^{3} a_{ij}i$
$v_y(x_0,y_0) = a_{01}$
$v_y(x_1,y_0) = a_{01} + a_{11} + a_{21} + a_{31}$
$v_y(x_0,y_1) = a_{01} + 2a_{02} + 3a_{03}$
$v_y(y_1,y_1) = \sum_{i=0}^{3}\sum_{j=1}^{3} a_{ij}j$

And now considering the double derivative with respect to the cross variables,
$v_{xy}(0,0) = a_{11}$,
$v_{xy}(1,0) = a_{11} + 2a_{21} + 3a_{31}$
$v_{xy}(0,1) = a_{11} + 2a_{12} + 3a_{13}$,
$v_{xy}(1,1) = \sum_{i=1}^{3}\sum_{j=1}^{3} a_{ij}ij$.

$$A = \begin{bmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0 \\ 1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1 \\ 0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&2&3&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0 \\ 0&1&2&3&0&1&2&3&0&1&2&3&0&1&2&3 \\ 0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&2&0&0&0&3&0&0&0 \\ 0&0&0&0&1&1&1&1&2&2&2&2&3&3&3&3 \\ 0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&2&3&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&0&0&0&2&0&0&0&3&0&0 \\ 0&0&0&0&0&1&2&3&0&2&4&6&0&3&6&9 \end{bmatrix} | \tag{11}$$

Now we have the unknown variable vector,

$$X = [a_{00}, a_{01}, ..., a_{33}]$$

The corresponding B vector is given as:

$$B = [v_{x_0,y_0}, ..., v_{x_3,y_3}]$$

Writing the $AX = B$ as $X = BA^{-1}$, and writing the given in the matrix form we get the linear equation with 16 variable and 16 unknown as

$$\begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix} = A =$$

$$A =$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} v(x_0,y_0) & v(x_0,y_1) & v_y(x_0,y_0) & v_y(x_0,y_1) \\ v(x_1,y_0) & v(x_1,y_1) & v_y(x_1,y_0) & v_y(x_1,y_1) \\ v_x(x_0,y_0) & v_x(x_0,y_1) & v_{xy}(x_0,y_0) & v_{xy}(x_0,y_1) \\ v_x(x_1,y_0) & v_x(x_1,y_1) & v_{xy}(x_1,y_0) & v_{xy}(x_1,y_1) \end{bmatrix} \begin{bmatrix} 1 & 0 & -3 & 2 \\ 0 & 0 & 3 & -2 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

,

In order to know the partial derivatives v_x, v_y and v_{xy} we need to know the corresponding neighbors of the known points, that is why we will need $4+4+4+4$ total points to calculate the unknown variables., that is why we require the 16 variable to calculate the bicubic interpolation.
Bi cubic interpolation creates the interpolated surface of the 2D image, It is all about taking point on grid and Interpol

Answer 3:

Given, We have an Image $I(x, y)$ which gets corrupted by

$$AWGN = N \sim \mathcal{N}(0,\, \sigma^2)\,.$$

Our assumption is that every pair of coordinates (x , y) the noise is uncorrelated, so we will be No correlation between the noise affected points in the image so every point of the image will be affected independently by the AWGN,
Let we have the noise value getting added at the pixel location is $n(x, y)$ and the corresponding corrupted image is given as $I'(x, y)$,
therefore the corresponding noisy image model can be given as,

$$I'(x, y) = I(x, y) + n(x, y) \tag{12}$$

at each pixel location (x, y), we have $n(u)$ as

$$n(u) = \frac{exp(-(u^2)/(2\sigma))}{\sqrt{2\pi\sigma^2}} \tag{13}$$

If the noise is AWGN, it can be shown that an image $I'(x, y)$ is formed by averaging K different noisy images,

$$I'(x, y) = \frac{1}{K}\sum_{i=1}^{i=K} I_i(x, y) \tag{14}$$

Then it also follows,

$$E[(I'(x, y)] = I(x, y) = \mu_{I'} \tag{15}$$

And as considering the non-correlation the variance can be directly added without consideration of covariances of other K images,

$$\sigma^2_{I'(x,y)} = \frac{1}{K}\sigma^2_{n(x,y)}$$

$$\sigma_{I'(x,y)} = \frac{1}{\sqrt{K}}\sigma_{n(x,y)} = \frac{\sigma}{\sqrt{K}} = \sigma_{I'} \tag{16}$$

From above(equation (15)) we can say that the every pixel value will act as the mean for the noise affected image.

Since we have the multiple random uncorrelated pixel value of an Image, so by the CLT we can consider the distribution of the image as Gaussian,

Therefore the probability distribution of the pixel intensity u at each location (x, y) value can be given as:

$$\begin{aligned} p_{I'(x,y)}(u) &= \frac{exp(-(u-\mu_{I'(x,y)})^2)/(2\sigma^2_{I'(x,y)}))}{\sqrt{2\pi\sigma^2_{I'(x,y)}}} \sim \mathcal{N}(\mu_{I'},\, \sigma^2_{I'}) \\ &= \frac{exp(-(u-\mu_{I'})^2)/(2\sigma^2_{I'}))}{\sqrt{2\pi\sigma^2_{I'}}} \end{aligned} \tag{17}$$

4(a)

We have been given the laplacian mask with -4 in the centre as A

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{18}$$

Now, as we know that the matrix formed by the multiplication of the two vectors has always the matrix rank as 1,
then from the above statement we can state that in order to the kernel to be separable the matrix has to be of Rank 1.
Therefore in order to determine the rank of the matrix we will apply the row transformation in A where $R3 \leftarrow (R3 - R1)$:r therefore the final echelon form gives:

$$A \leftarrow \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{19}$$

From above we can say that the Matrix A i.e. laplacian kernel has rank=2 (not equal to 1), therefore it cannot be expressed as the separable outer product of the two vectors.

4(b)

Laplacian of an image$f$ with the gaussian g can be written as $\nabla^2(f * g) = f * \nabla^2 g$, expand the del operation the above expression can be written as below due to associative and commutative nature of convolution,

$$f * \nabla^2 g = f * \left(\frac{\partial^2}{\partial x^2} g + \frac{\partial^2}{\partial y^2} g \right) = f * g_{xx} + f * g_{yy} \tag{20}$$

Where g_{xx} and g_{yy} are the second derivative g.
We have 2D convolutional LOG mask given as $A = \nabla g$, which can be expressed as the equation (18) as the sum of two 1D 2nd derivative operator as $g_{xx} = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$ and $g_{yy} = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}^T$;

Therefore we can write the 2D LOG kernel as the sum of two 1D kernels.

$$f * \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} = f * \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} + f * \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \tag{21}$$

Answer 5:

Mean filter that we can use the standard mean filter like for 3×3 is something like this

$$\frac{1}{9} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{22}$$

and for 5×5 is something like this

$$\frac{1}{25} * \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{23}$$

$$f_1 = \frac{1}{(2a+1)^2} \sum_{i=-a}^{i=a} \sum_{i=-a}^{i=a} f(x+j, y+i)$$

$$f_2 = \frac{1}{(2a+1)^2} \sum_{i=-a}^{i=a} \sum_{i=-a}^{i=a} f_1(x+j, y+i)$$

$$f_3 = \frac{1}{(2a+1)^2} \sum_{i=-a}^{i=a} \sum_{i=-a}^{i=a} f_2(x+j, y+i)$$

$$f_K = \frac{1}{(2a+1)^2} \sum_{i=-a}^{i=a} \sum_{i=-a}^{i=a} f_{K-1}(x+j, y+i)$$

So, if we represent our mean filter with g and original image as f. Then this convolution operation can be represented as:

$$f1 = f * g$$
$$f2 = f1 * g = (f * g) * g$$
$$f3 = f2 * g = ((f * g) * g) * g$$

and like this we can represent the f_k as

$$f_k = ((f) * g) * g)..... * g$$

Since, we know that convolution operation is associative in nature.So, we can formulate this as we can convolve the **g** with itself and at last we can convolve it with original image **f**.

Answer 6:

i

Given 1D image:

$$I(x) = cx + d$$

,

The Gaussian distribution in 1-D has the form

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

1D image J is filtered Gaussian kernel with filter length w,The convolution of I with a kernel g is

$$\begin{aligned}
I'(x) &= I(x) * g(x) \\
&= \sum_{u \in w} g(u) I(x-u) \\
&= \sum_{u \in w} (a(x-u)+b) \frac{exp(-u^2/(2\sigma^2))}{\sqrt{2\pi\sigma^2}} \\
&= (ax+b) \sum_{u \in w} \frac{exp(-u^2/(2\sigma^2))}{\sqrt{2\pi\sigma^2}} \\
&\text{since the sum of the weights of the filter is approximately unity} \\
&= (ax+b) \\
&= I(x)
\end{aligned}$$

Hence we got the same image as original image.

ii.

When image I is filtered with the bilateral filter with parameters σ_s and σ_r the resultant image J is,

$$BF[I]\mathbf{p} = \frac{1}{W\mathbf{p}} \sum_{\mathbf{q} \in S} G_{\sigma_s}(\|\mathbf{p} - \mathbf{q}\|) G_{\sigma_r}(|I_{\mathbf{p}} - I_{\mathbf{q}}|) I_q$$

Where,

$$W_{\mathbf{p}} = \sum_{\mathbf{q} \in \mathcal{S}} G_{\sigma_s}(\|\mathbf{p} - \mathbf{q}\|) G_{\sigma_r}(|I_{\mathbf{p}} - I_{\mathbf{q}}|)$$

let the pixel location be p and the neighbourhood location be q, therefore putting the values of the intensities at the location p and q

$$
\begin{aligned}
BF[I]\mathbf{x} &= \frac{1}{W\mathbf{x}} \sum_{\mathbf{q} \in S} G_{\sigma_s}(\|\mathbf{x} - \mathbf{q}\|) G_{\sigma_r}\left(|I_{\mathbf{x}} - I_{\mathbf{q}}|\right) I_{\mathbf{q}} \\
&= \frac{1}{W\mathbf{x}} \sum_{\mathbf{q} \in S} G_{\sigma_s}(\|\mathbf{x} - \mathbf{q}\|) G_{\sigma_r}\left(|(a\mathbf{x} + b) - (a\mathbf{q} + b)|\right)(a\mathbf{q} + b) \\
&= \frac{1}{W\mathbf{x}} \sum_{\mathbf{q} \in S} G_{\sigma_s}(\|\mathbf{x} - \mathbf{q}\|) G_{\sigma_r}\left(|a\mathbf{x} - a\mathbf{q}|\right)(a\mathbf{q} + b) \\
&= \frac{\sum_{\mathbf{q} \in S} exp(-(\mathbf{x} - \mathbf{q})^2/2\sigma_s^2) exp(-a^2(\mathbf{x} - \mathbf{q})^2/2\sigma_r^2)(a\mathbf{q} + b)}{\sum_{\mathbf{q} \in S} exp(-(\mathbf{x} - \mathbf{q})^2/2\sigma_s^2) exp(-a^2(\mathbf{x} - \mathbf{q})^2/2\sigma_r^2)} \\
&= k(ax + b) \\
&= I_D(x)
\end{aligned}
$$

$I_D(x)$ is the denoised intensity of pixel x

Answer 7:

To prove that laplacian operator is rotationally invariant,

Given (x, y) original coordinate, whereas (u, v) is rotated coordinate, i.e.
$u = x\cos\theta - y\sin\theta$ and $v = x\cos\theta + y\sin\theta$]

TPT: $f_{xx} + f_{yy} = f_{uu} + f_{vv}$
Proof:-

$$\frac{\partial f}{\partial x} = \frac{\partial u}{\partial x}\frac{\partial f}{\partial u} + \frac{\partial v}{\partial x}\frac{\partial f}{\partial v} \tag{24}$$
$$= \cos\phi\frac{\partial f}{\partial u} + \sin\phi\frac{\partial f}{\partial v} \tag{25}$$
$$= f_x = Z \tag{26}$$

Now again taking derivative of the $Z = f_x$ with respect to x we get,

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial Z}{\partial x} \tag{27}$$
$$= \frac{\partial u}{\partial x}\cdot\frac{\partial Z}{\partial u} + \frac{\partial v}{\partial x}\cdot\frac{\partial Z}{\partial v} \tag{28}$$
$$= \cos\phi\frac{\partial Z}{\partial u} + \sin\phi\frac{\partial Z}{\partial v} \tag{29}$$
$$= \cos\theta\frac{\partial}{\partial u}\left(\cos\theta\frac{\partial f}{\partial u} + \sin\theta\frac{\partial f}{\partial v}\right) + \sin\theta\frac{\partial}{\partial v}\left(\cos\theta\frac{\partial f}{\partial u} + \sin\theta\frac{\partial f}{\partial v}\right). \tag{30}$$
$$= \cos^2\theta\frac{\partial^2 f}{\partial u^2} + 2\cos\theta\sin\theta\frac{\partial^2 f}{\partial u\partial v} + \sin^2\theta\frac{\partial^2 f}{\partial v^2} \tag{31}$$
$$= \cos^2\theta f_{uu} + 2\cos\theta\sin\theta f_u f_v + \sin^2\theta f_{vv} \tag{32}$$
$$= f_{xx} \tag{33}$$

Similarly the expression can be derived for f_{yy} ,

$$\frac{\partial^2 f}{\partial y^2} = \sin^2\theta\frac{\partial^2 f}{\partial u^2} - 2\cos\theta\sin\theta\frac{\partial^2 f}{\partial u\partial v} + \cos^2\theta\frac{\partial^2 f}{\partial v^2} \tag{34}$$
$$= \sin^2\theta f_{uu} - 2\cos\theta\sin\theta f_u f_v + \cos^2\theta f_{vv} \tag{35}$$
$$= f_{yy} \tag{36}$$

adding the above equation and we get,

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f_{uu}(\cos^2\theta + \sin^2\theta) + 2\sin\theta\cos\theta f_u f_v - 2\sin\theta\cos\theta f_u f_v + f_{vv}(\sin^2\theta + \cos^2\theta) \tag{37}$$

$$= f_{uu} + f_{vv} = f_{xx} + f_{yy} \tag{38}$$

Hence proved

Answer 8

Summary for the result is like this:
When an noise of **zero mean and standard deviation 5** is added to the **barbara image and kodak image** then with **sigma_s=2 and sigma_r=2**. The results are not much different, the only difference which is not much significant but when observed carefully shows a very slight difference in intensity values at places and a slight amount of blur at places.The result for the case when **sigma_s=3 and sigma_r=15** the results are slightly more observable.
Whereas when the **standard deviation** has been increased to 10 then as amount of noise has been increased the results can be seen from the images at places where there was an noise, in output image that place has been blurred so to manage the noise.The results in this pdf are not that much clear.So, please check the results from the directory folder.

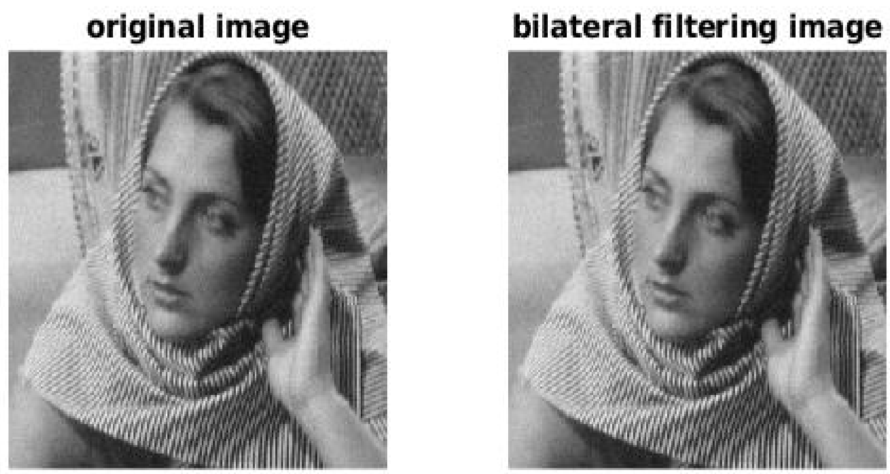


Figure 1: left is noisy and right is an filtered image using bilateral filter. Sigma_s = 2, Sigma_r = 2, Mean = 0 and Standard Deviation = 5

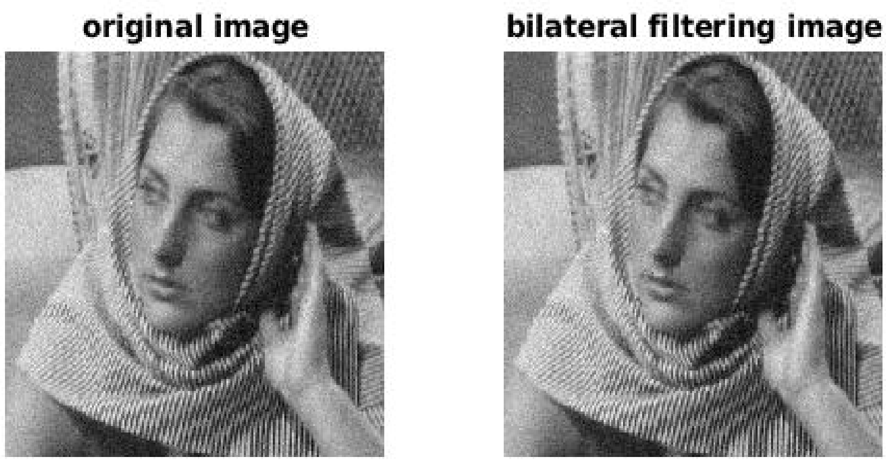


Figure 2: left is noisy and right is an filtered image using bilateral filter. Sigma_s = 2, Sigma_r = 2, Mean = 0 and Standard Deviation = 10

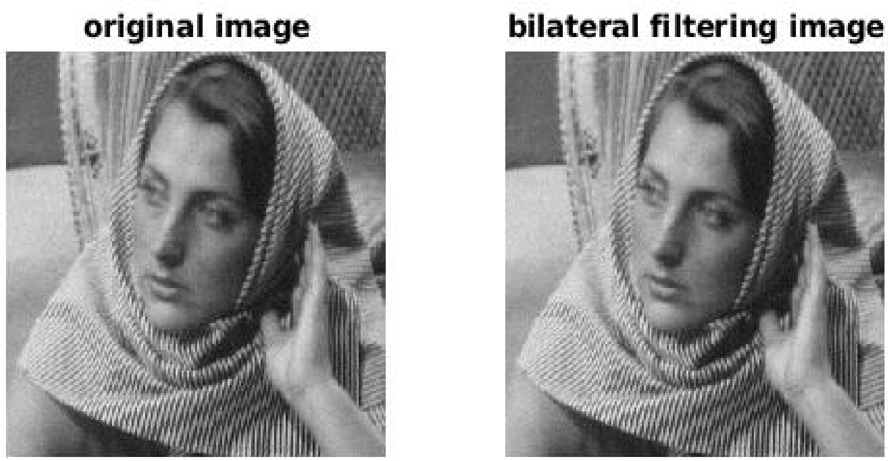


Figure 3: left is noisy and right is an filtered image using bilateral filter.Sigma_s = 0.1, Sigma_r = 0.1, mean = 0 and Standard Deviation = 5

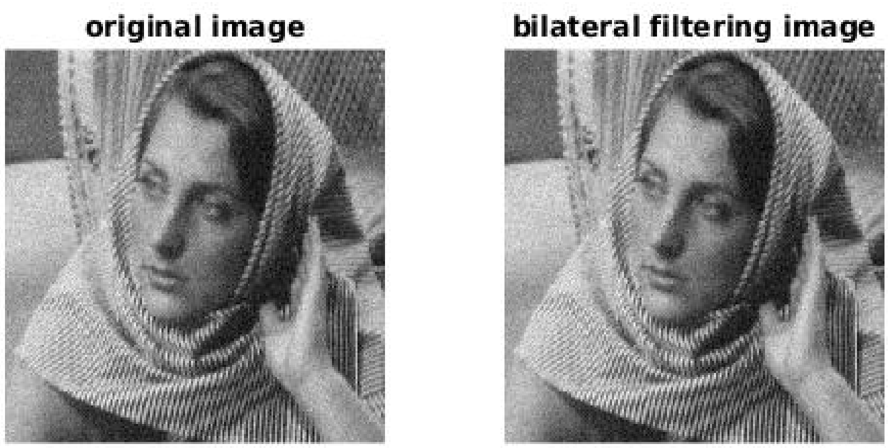


Figure 4: left is noisy and right is an filtered image using bilateral filter.Sigma_s = 0.1, Sigma_r = 0.1, Mean = 0 and Standard Deviation = 10

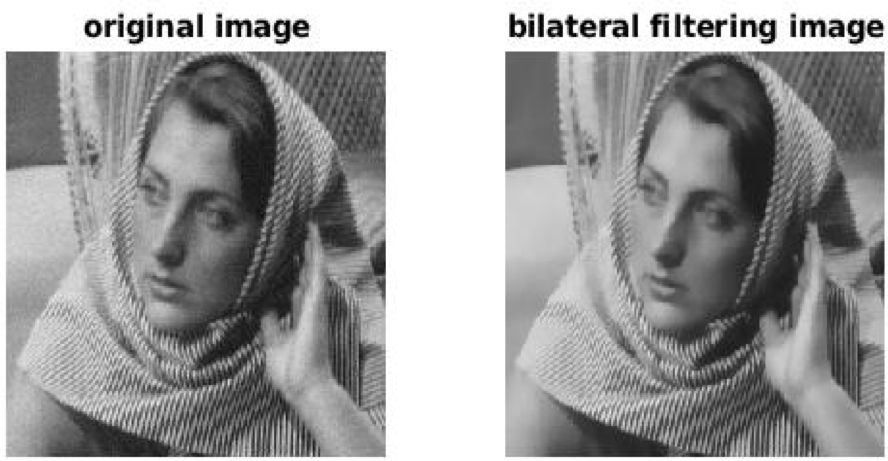


Figure 5: eft is noisy and right is an filtered image using bilateral filter. Sigma_s = 3, Sigma_r = 15, Mean = 0 and Standard Deviation = 5

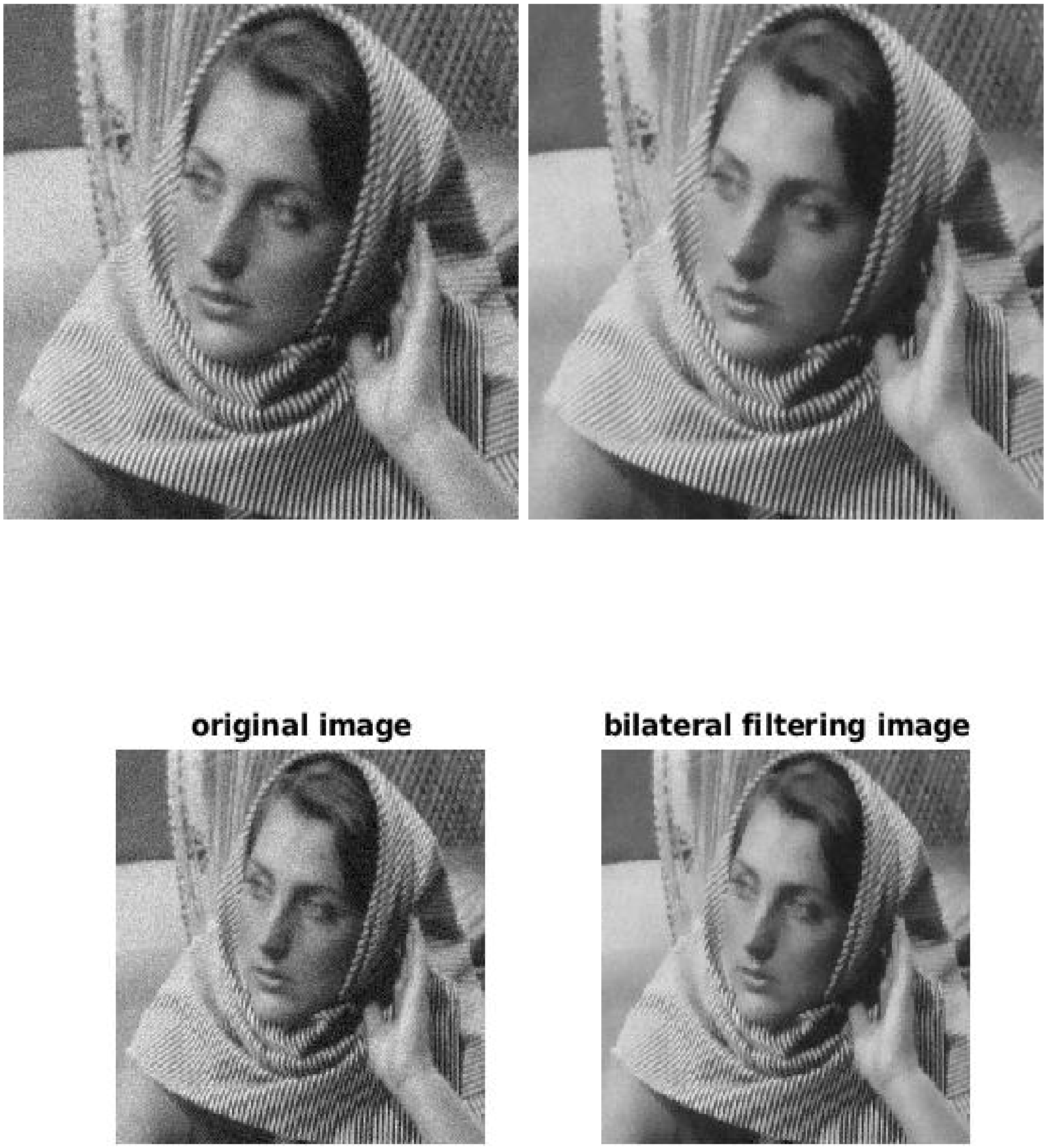


Figure 6: Left is noisy and right is an filtered image using bilateral filter. Sigma_s = 3, Sigma_r = 15, Mean = 0 and Standard Deviation = 10

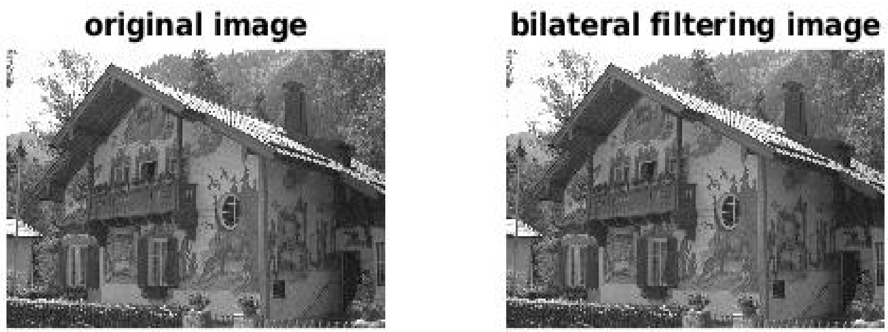


Figure 7: Left is noisy and right is an filtered image using bilateral filter. Sigma_s = 2, Sigma_r = 2, Mean = 0 and Standard Deviation = 5

Figure 8: left is noisy and right is an filtered image using bilateral filter. Sigma_s = 2, Sigma_r = 2, Mean = 0 and Standard Deviation = 10

Figure 9: Left is noisy and right is an filtered image using bilateral filter.Sigma_s = 0.1, Sigma_r = 0.1, mean = 0 and Standard Deviation = 5

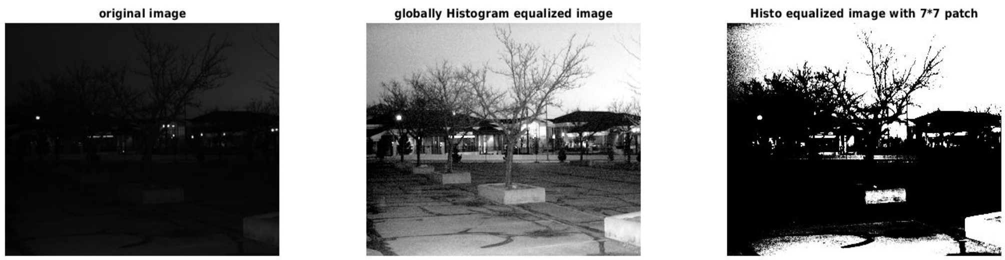
Figure 10: Left is noisy and right is an filtered image using bilateral filter.Sigma_s = 0.1, Sigma_r = 0.1, Mean = 0 and Standard Deviation = 10

Figure 11: Left is noisy and right is an filtered image using bilateral filter. Sigma_s = 3, Sigma_r = 15, Mean = 0 and Standard Deviation = 5

Figure 12: Left is noisy and right is an filtered image using bilateral filter. Sigma_s = 3, Sigma_r = 15, Mean = 0 and Standard Deviation = 10

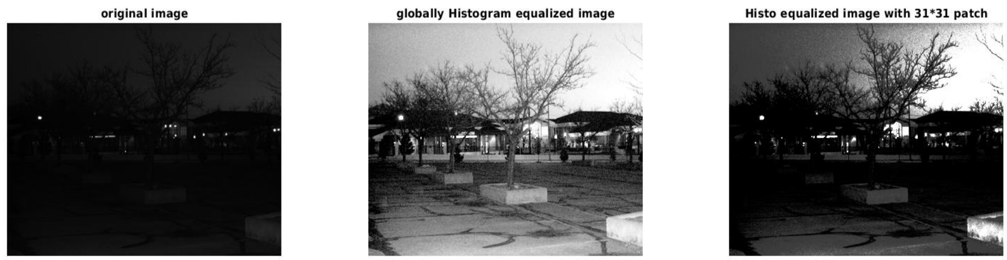
Answer 9

Global(Top) vs 7x7 HE(Bottom)

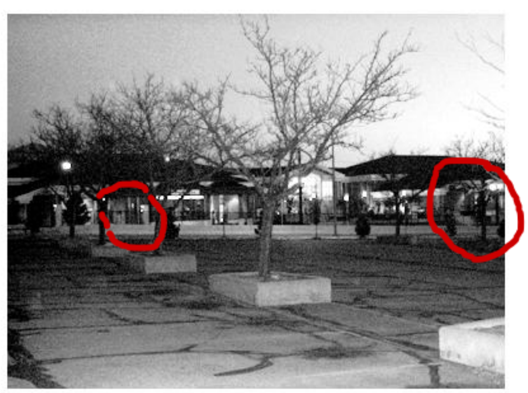


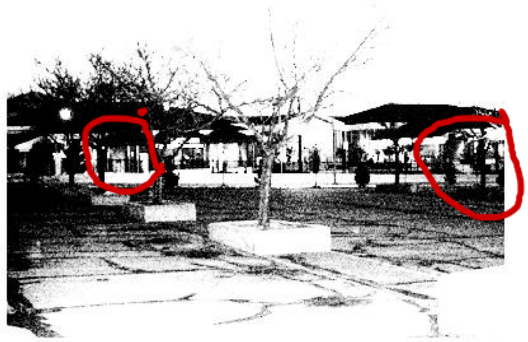
Above two images are Comparison between the Global histogram equalisation and local 7x7 histogram equalisation,
we can observe that in the small patch size the texture of the ground surface is fairly highlighted.
Whereas in larger patch size some distant object details are getting highlighted.

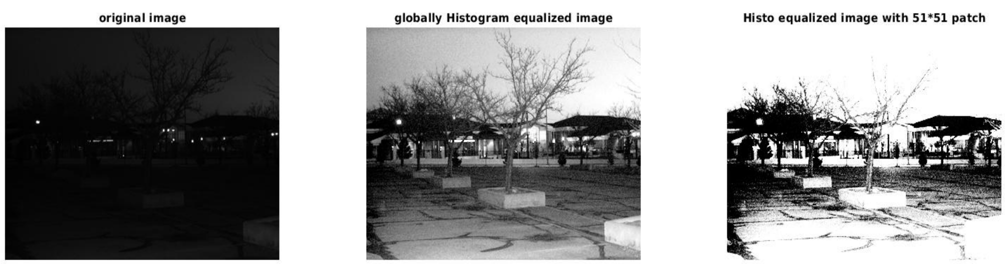
Global(Top) vs 31x31 local(Bottom)



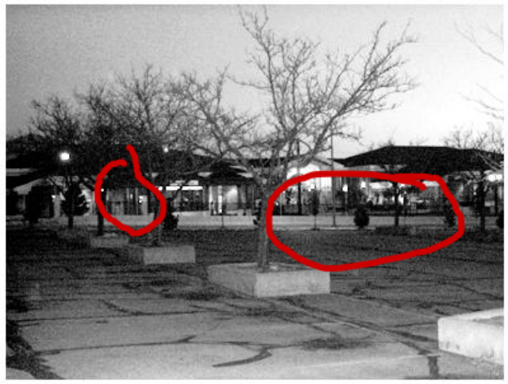
Global(Top) vs 51x51 local(Bottom)

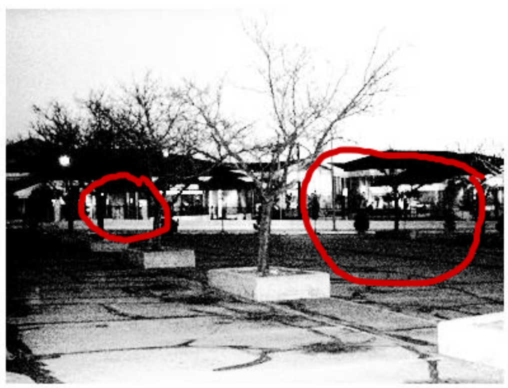


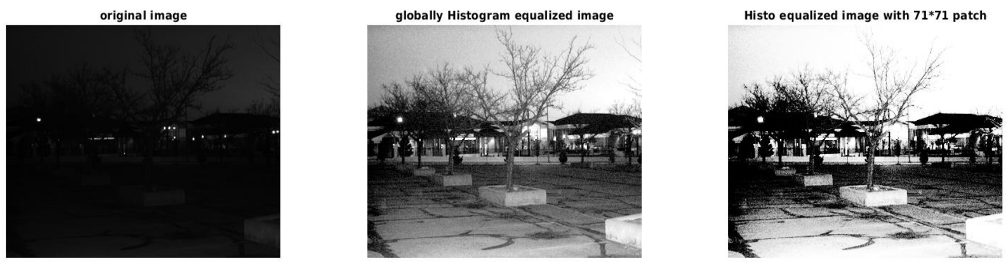



original image
globally Histogram equalized image
Histo equalized image with 51*51 patch

Global(Top) vs 71x71 local(Bottom)






original image
globally Histogram equalized image
Histo equalized image with 71*71 patch

REFERENCES

https://cecas.clemson.edu/ stb/ece847/internal/cvbook/ch03$_{f}iltering.pdf$

https : //www.ph.ed.ac.uk/ wjh/teaching/dia/documents/noise.pdf

www.cs.usask.ca/ spiteri/M211/notes/chapter3.pdf

$www.giassa.net/?page_{i}d = 371)(page_{8}6_{g}onsalves_{a}nd_{w}oods$

ref-https://cecas.clemson.edu/ stb/ece847/internal/cvbook/ch03*filtering.pdf*

$https : //en.wikipedia.org/wiki/Bilateral_{f}ilter$