Digital Image Processing

EE 624

Assignment-1

Tallam Vamsi – 130102070

Immidisetti Rakhil – 130102026

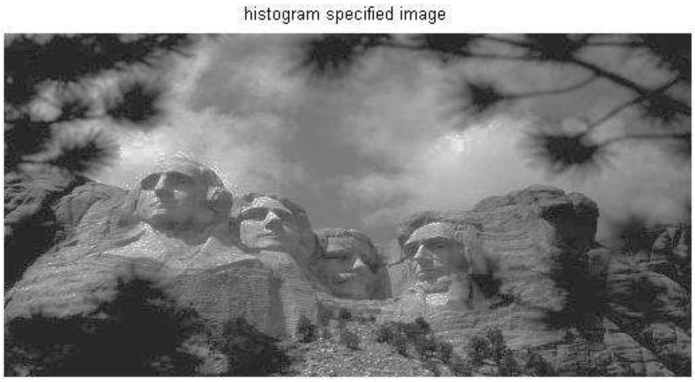
1. Histogram Specification:

Modify the histogram of the image givenhist.jpg in such a way that the resulting histogram nearly approximates the histogram of the image **sphist.jpg**. Display the histogram of the image **givenhist.jpg** after this transformation.

In image processing, **histogram matching** or **histogram specification** is the transformation of an image so that its histogram matches a specified histogram.

Given image is converted to double and histograms are obtained by the function hist. Then CDF’s are found. Then mapping was done to obtain histogram specified image.

It can be done in two ways i.e., rounding off (CDF*255) or without rounding off.



Histogram specified image with rounding off and without rounding off.

3(i). Edge preserving smoothing Filters:

You are provided with a noisy picture of Lenna. (Lenna noise.jpg). In this task, you are expected to explore a set of edge preserving filters, to address the problem of image denoising. For each filter, display both the input and de-noised image.

Anisotropic non-linear diffusion filter:
Implement the anisotropic diffusion filter for reducing the effect of noise. Consider adapting the value of the conduction coefficient, so as to stop the diffusion on the edges. Nevertheless, to get a visually pleasing result, it is suggested to iterate through the algorithm several times. You may consider either a 4 or 8 neighbour connectivity.

In image processing and computer vision, **anisotropic diffusion**, also called **Perona–Malik diffusion**, is a technique aiming at reducing image noise without removing significant parts of the image content, typically edges, lines or other details that are important for the interpretation of the image. Two functions for diffusion coefficients are used. $c\left(\|\nabla I\|\right) = e^{-\left(\|\nabla I\|/K\right)^2}$ and $c\left(\|\nabla I\|\right) = \frac{1}{1+\left(\frac{\|\nabla I\|}{K}\right)^2}$

Tested using 4 neighbours and 8 neighbours.

Original Noisy Image:

Images obtained as result of anisotropic diffusion after 5, 10 and 15 iterations using 1st formula for diffusion coefficients and 8 neighbours.

Images obtained as result of anisotropic diffusion after 5, 10 and 15 iterations using 2nd formula for diffusion coefficients and 8 neighbours.

Images obtained as result of anisotropic diffusion after 5, 10 and 15 iterations using 1st formula for diffusion coefficients and 4 neighbours.

Images obtained as result of anisotropic diffusion after 5, 10 and 15 iterations using 2nd formula for diffusion coefficients and 4 neighbours.

Using 8 neighbours worked better but fewer iterations. But blurring happens faster with 8 neighbours.

3(ii). Non-local means filter:

Denoise the image using the non-local means filter. For faster implementation, restrict the search of similar patches in a window of size 5*5 pixels round the current patch. In addition, assume each patch to be of size 7*7. To get better results, it is expected that you compute the Gaussian Weighted Sum of squares distance between the patches. The bandwidth / scale of the Gaussian may set experimentally.

Non-local means is an algorithm in image processing for image denoising. Unlike "local mean" filters, which take the mean value of a group of pixels surrounding a target pixel to smooth the image, non-local means filtering takes a mean of all pixels in the image, weighted by how similar these pixels are to the target pixel. This results in much greater post-filtering clarity, and less loss of detail in the image compared with local mean algorithms. Scale of Gaussian is found experimentally. It is observed for h=0.5, 0.9, 1, 2, 5, 25.

Given image

Images with scaling factor h= 0.5, 0.9, 1, 2, 5, 25.

5. Image Restoration:

The aerial image degraded.tif has been degraded with atmospheric turbulence. The goal is to restore this image using the Pseudo inverse filter. Display the restored image.

The purpose of image restoration is to "compensate for" or "undo" defects which degrade an image. Degradation comes in many forms such as motion blur, noise, and camera mis-focus. In this method we look at an image assuming a known atmospheric turbulence.

$$H(u,v) = \exp[-k(u^2+v^2)^{5/6}]$$

Thresholds were used e=1, 0.5, .02, 0.0067, 0.005.

Original Image





2. Bilateral filter

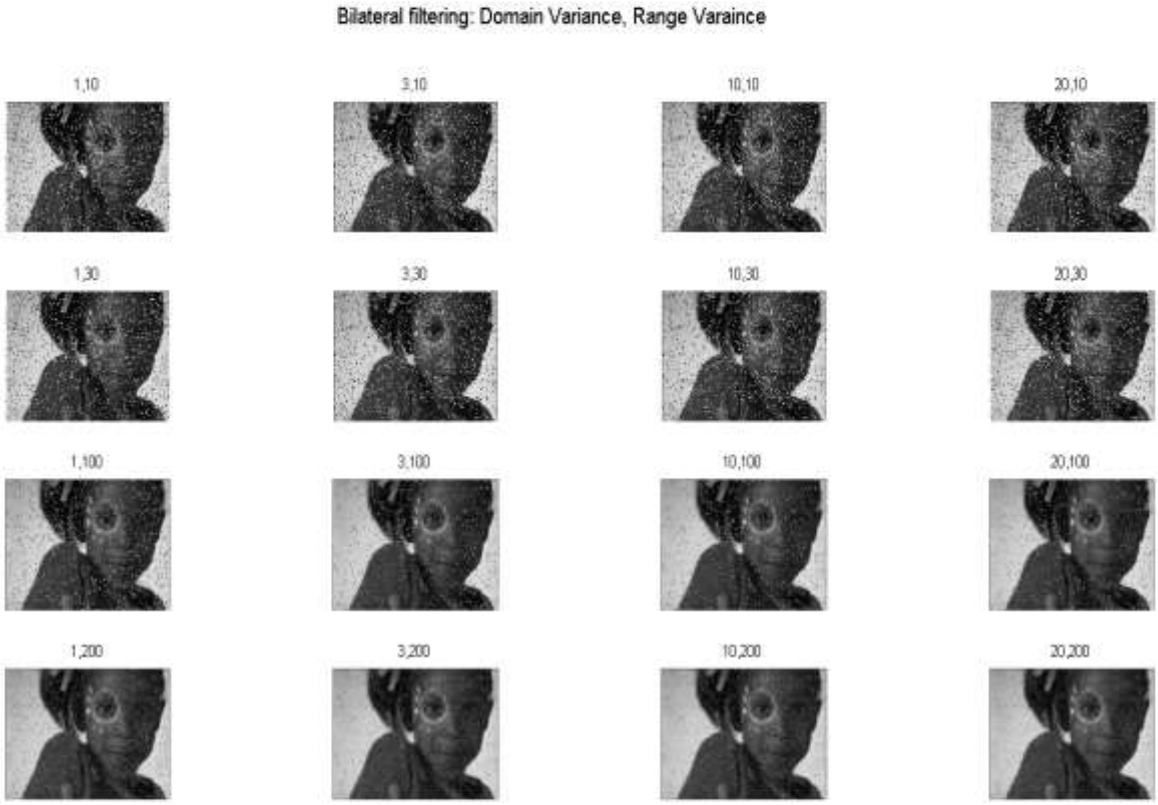
Bilateral filtering smoothens images while preserving edges, by means of a nonlinear combination of nearby image values. It combines gray levels based on both their geometric closeness and their photometric similarity, and prefers near values to distant values in both domain and range.

We find the effect of bilateral filter for different possible combinations of variance of domain and range filter. We use Prewitt filter for computing horizontal and vertical gradients of input image.

As we increase the variance of range filter we observe that it only changes the color map of the image. While increasing the variance of domain filter we observe geometric smoothing of image. So when we increase both of them we observe that the filtered output image gets better and edges are preserved, enforcing both geometric and photometric locality.

unifnoisy.jpg

spnoisy.jpg

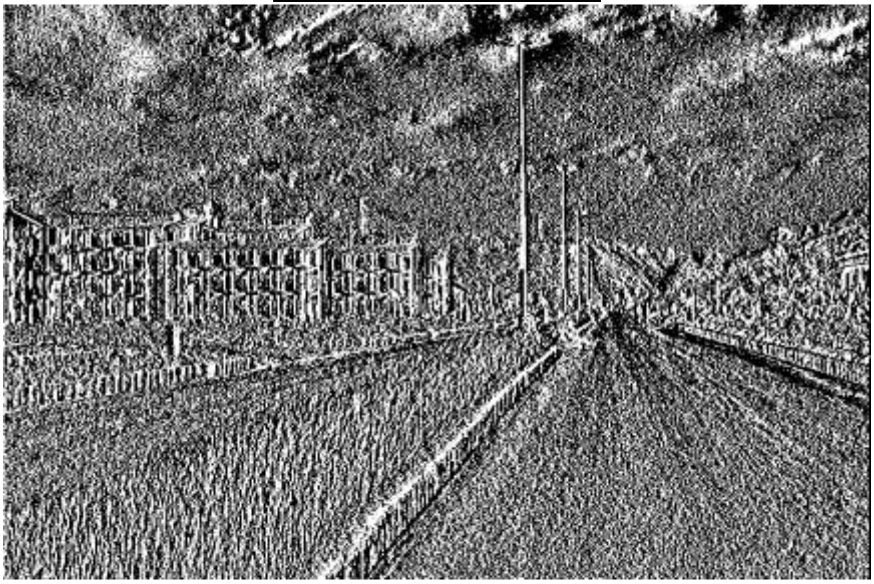


spunifnoisy.jpg

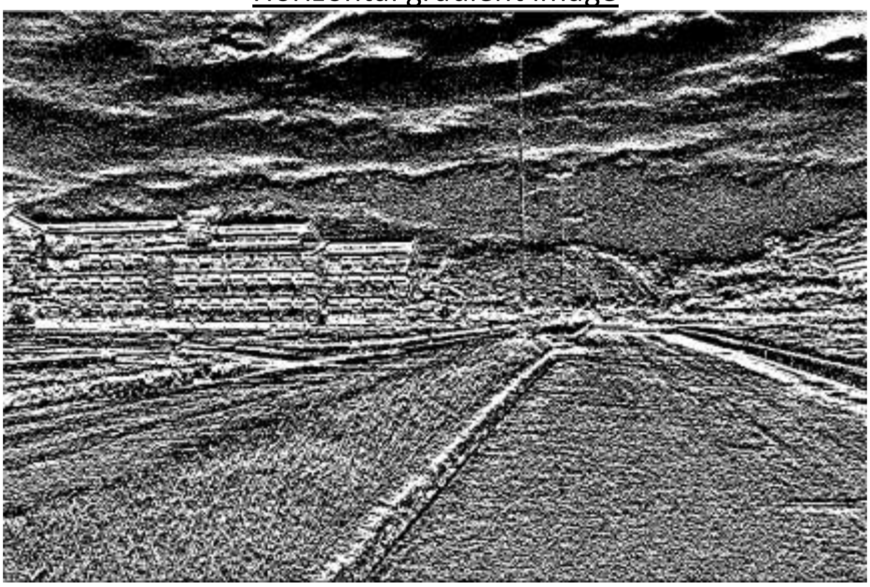
4. Harris corner detection

At a corner there is significant change in intensity in all directions. In Harris corner detection, the structure tensor matrix captures the intensity structure of local neighborhood. We compute the corner response function from the eigenvalues of structure tensor matrix. The structure tensor matrix is computed for each pixel by considering a window centered on it of size 5x5. We then performed threshold on corner response with a value of 10^7 which was suitably selected. Then we performed non maximal suppression considering a neighborhood of 5x5.

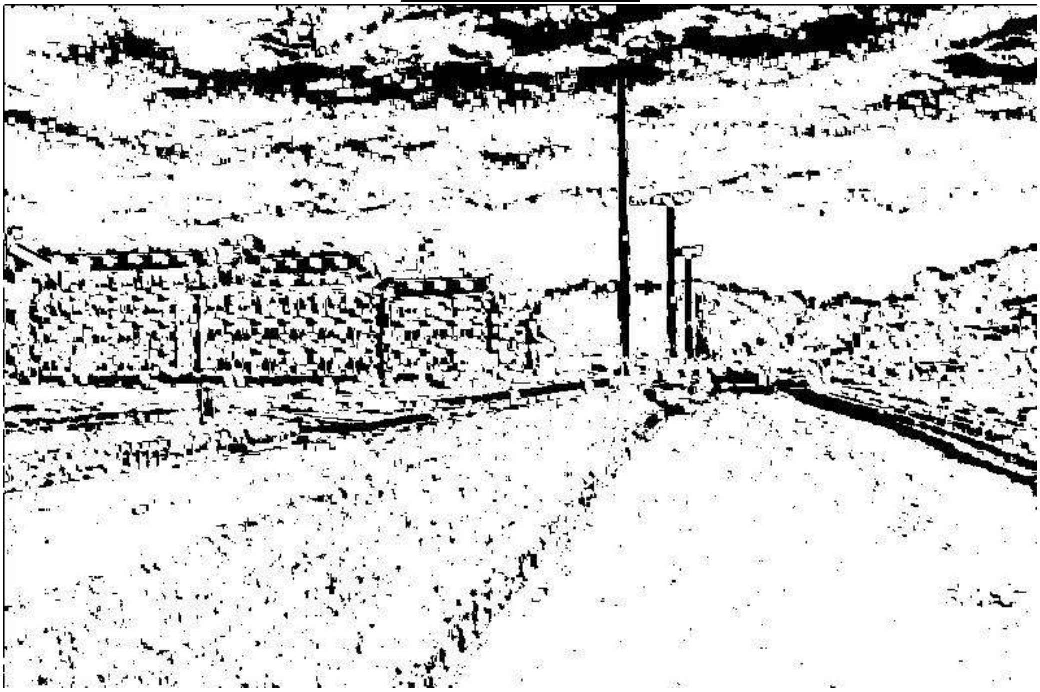
Vertical gradient image



Horizontal gradient image



Corner Response

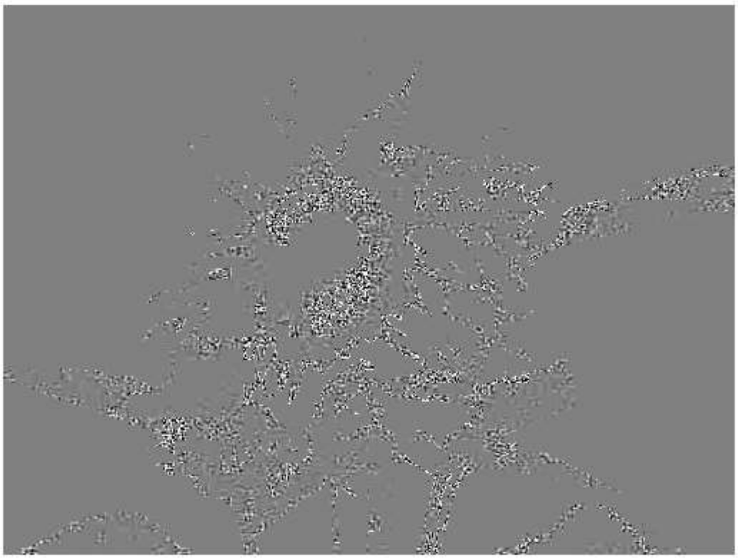


Corners after threshold and non-maximal suppression

6(i). Discrete Cosine Transform

We performed threshold coding by retaining highest 25% coefficients with respect to magnitude. Even though we are retaining only 25% of information we see that the image is of high quality. Hence, we can conclude that DCT has good compression capability. The error image intensity is scaled appropriately for proper visualization. The total Euclidean error between reconstructed and original image is 448.677

Difference between Reconstructed and Original image



Reconstructed Image

6(i). Wavelet Transform

The wavelet transform is separable. They are the product of two 1-D functions. We first perform wavelet transform column wise then row wise.

Scaling function coefficients = [0.7071, 0.7071]

Wavelet function coefficients = [0.7071, -0.7071]

After performing convolutions they are properly scaled to [0,255]

$$\phi(x, y) = \phi(x)\phi(y)$$

Variations along columns $\quad \psi^H(x, y) = \psi(x)\phi(y)$

Variations along rows $\quad \psi^V(x, y) = \phi(x)\psi(y)$

Variations along diagonals $\quad \psi^D(x, y) = \psi(x)\psi(y)$

(a)Approximate (b) Horizontal variation

(c) Vertical variation (d) Diagonal Variation