Assignement EE624

Report submitted by

Afzal Ahmad
Roll no.: 150108001

Sai Bhaskar Devatha
Roll no.: 150108012

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.

Histogram equalisation is a technique used to equally spread the histogram of a given image. This technique is widely used for contrast stretching. The principles of this technique are used to perform another operation called histogram specification.

Histogram specification tries to match the histogram of a given image to a specified histogram. We first equalize the original image and the specified image. After equalisation we try matching the histogram of both the images and store the mappings. This mapping is used to generate the new image with a histogram similar to the specified image.

The results are presented below-

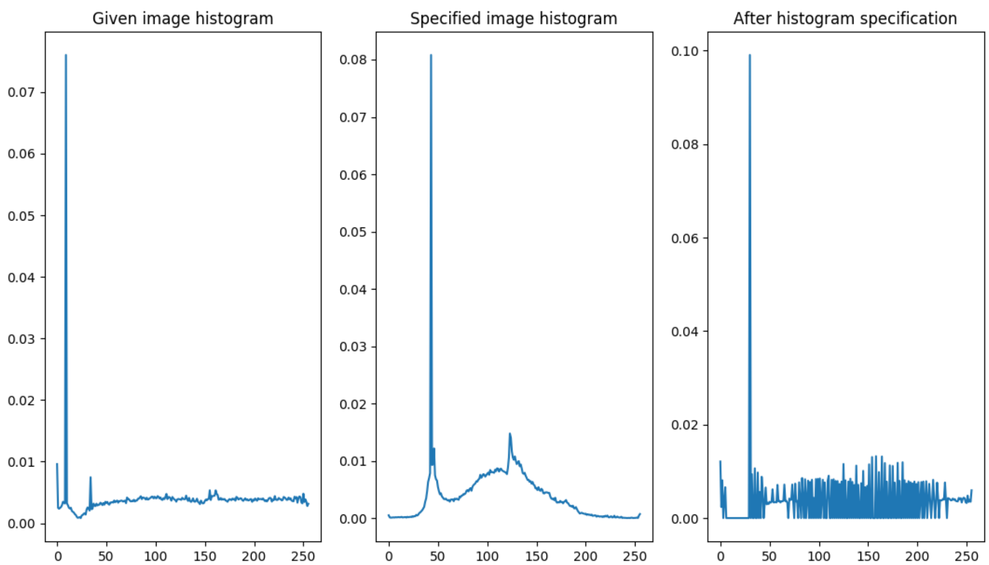

Figure 1: Histogram of the images

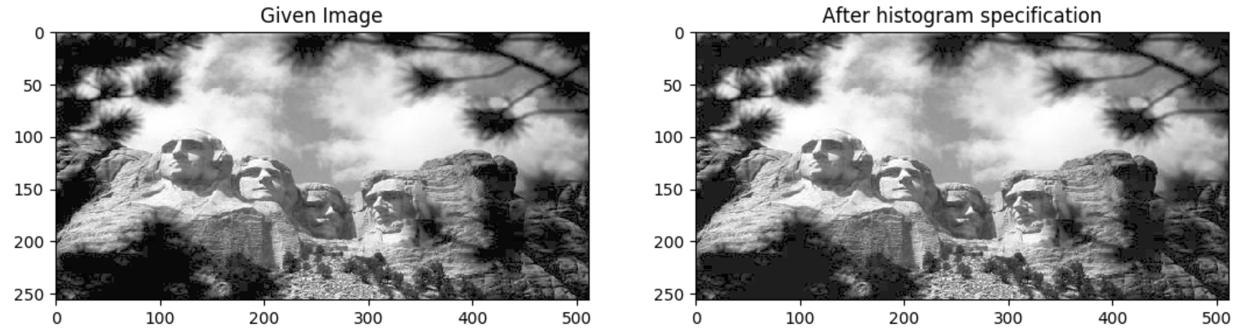

Figure 2: Images

2 Bilateral Filtering

Filtering operations can be performed using similarity in space or in intensity. Bilateral filtering technique uses a combined approach by using both the similarities together.

The effect of different values of domain and range variances is shown in following figures. We observe that at a given domain variance σ_d, increasing the range variance σ_r results in the removal of noise and a smoother image. Increasing the σ_d does give some geometric smoothing but the effect of range variance is dominant. Increasing both of them gives an image with reduced noise but the edges remain preserved.

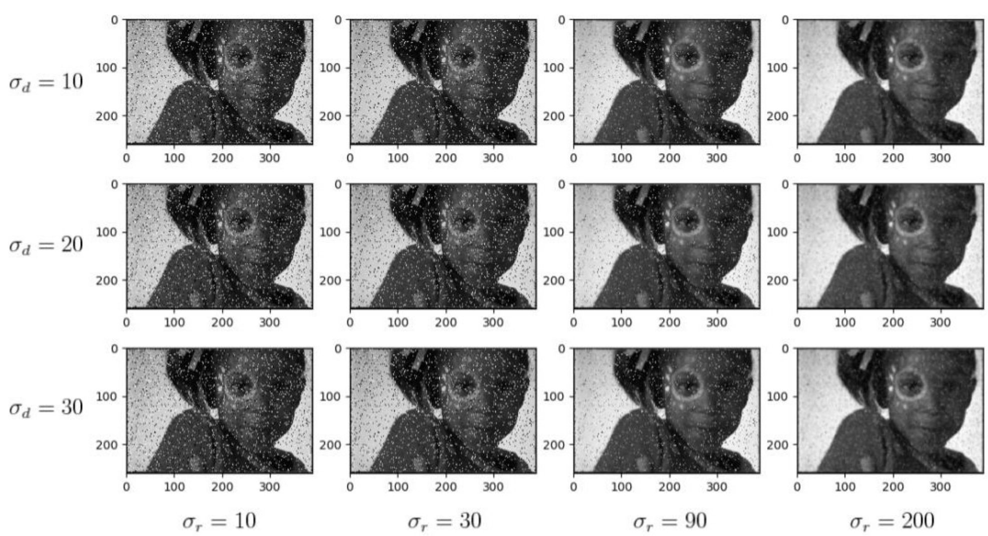

Figure 3: Results for spnoisy.jpg

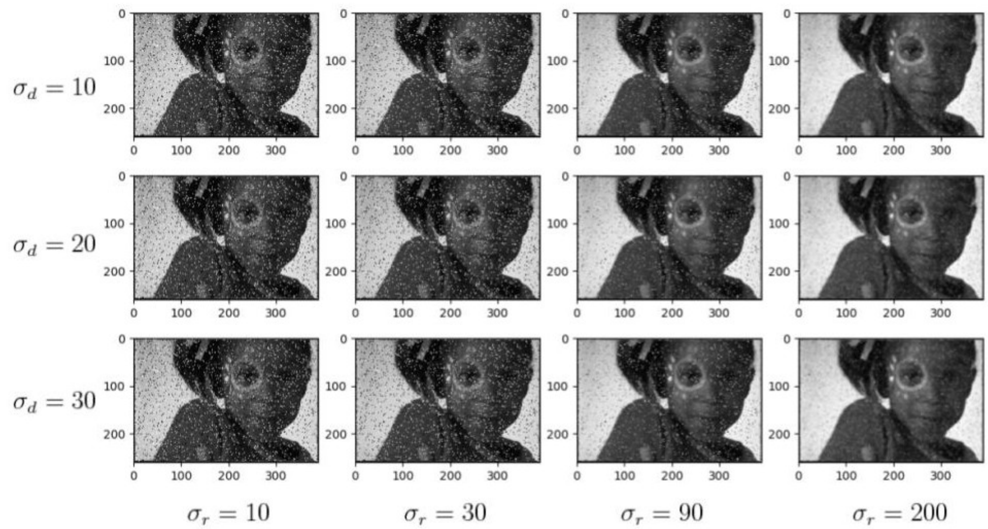

Figure 4: Results for spunifnoisy.jpg

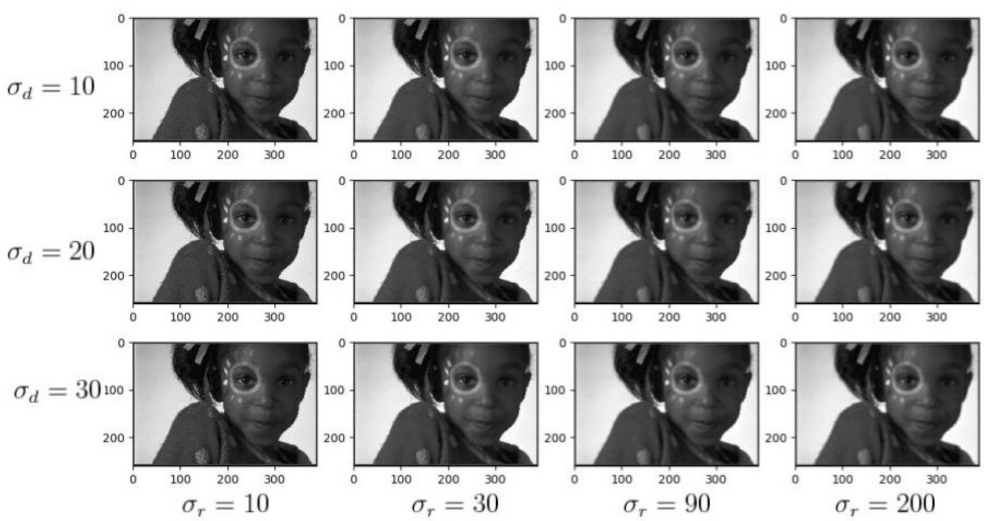

Figure 5: Results for unifnoisy.jpg

3 Diffusion filtering

3.1 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 neighbor connectivity.

Anisotropic diffusion is a filtering technique that uses directional filtering to remove noise from images. This technique was designed to preserve the image content like edges, lines etc. while filtering. It reduces image noise without removing important details that are necessary for recognizing and interpreting the image.

We are using 8 neighbours for this filtering. The results for different iterations are given below.

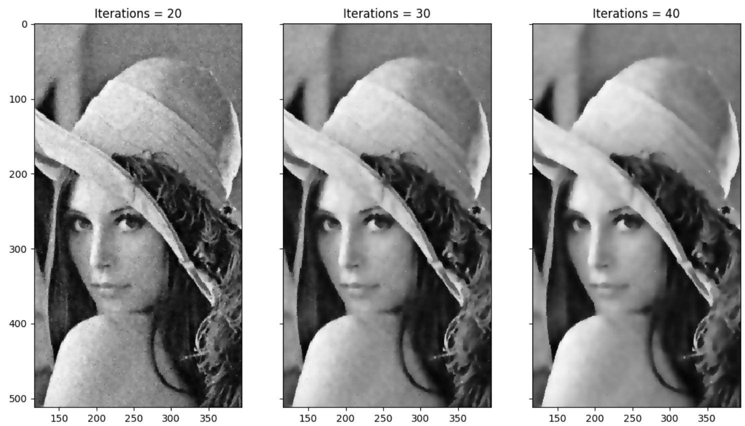

Figure 6: Anisotropic Kappa k = 15 and $\lambda = 0.1$

3.2 Isotropic linear diffusion filter

Unlike anisotropic diffusion, isotropic diffusion is a non-directional technique. It uses a much simpler way to reduce noise but it also blurs important image features like lines and edges.

We use 8 neighbours for this filtering. As compared to anisotroptic filtering

Figure 7: Isotropic Kappa k = 15 and $\lambda = 0.15$

4 Corner Detection

As explained in the class, to detect Harris corner the following steps are implemented.

1. Compute Derivatives
2. Compute Matrix M (Structure Tensor Matrix)
3. Compute Corner Response function (R)
4. Threshold R
5. Non Maximal Suppression

It is observed that the intensity around a corner differs a lot with neighbouring pixels. After computing derivatives, the following figures are observed.

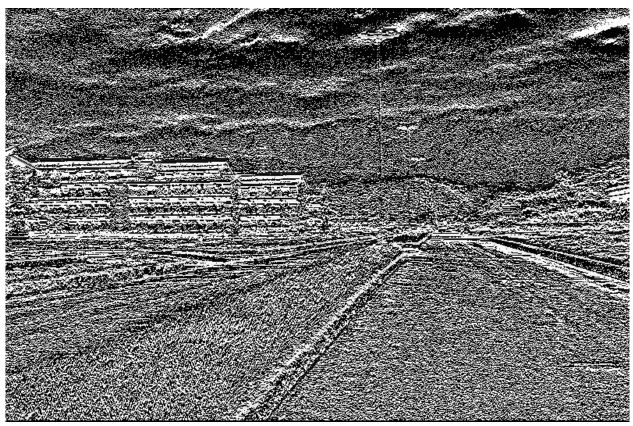

Figure 8: Derivative in X-direction

Figure 9: Derivative in Y-direction

Later, a window sized 5x5 and the above derivatives, Structure Tensor Matrix is formed. Using the M Matrix, Corner Response Function (R) is calculated. Three different values are taken for alpha in between 0.04 and 0.06. After calculating the R, it is threshold with different values in multiples of its mean (K). The threshold images are shown below.

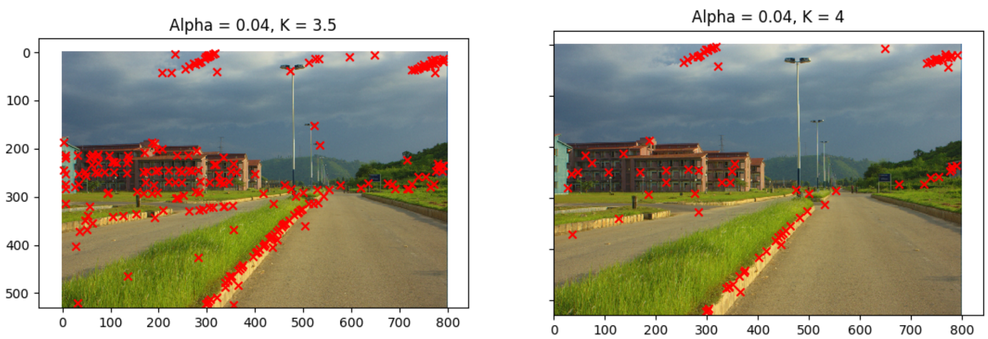

Figure 10: For alpha value 0.04

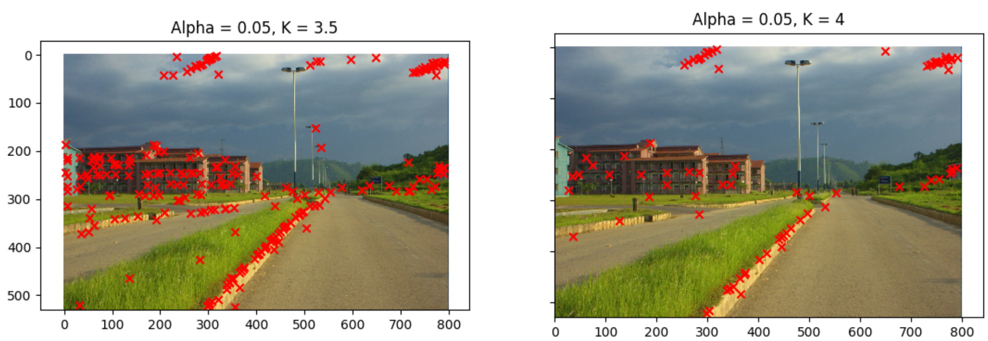

Figure 11: For alpha value 0.05

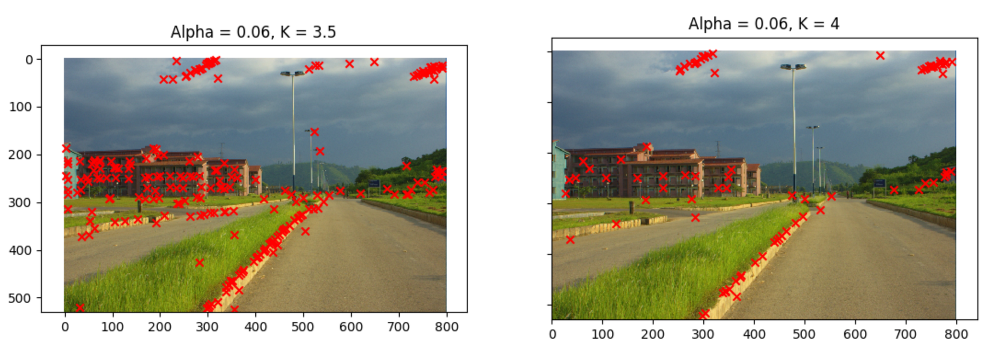

Figure 12: For alpha value 0.06

We can observe that as Alpha increase, the errors reduced a little and thresholding to higher values worked in some areas, didnt work in some areas.