

Digital Image Processing

Vusuvandla Khagesh Kumar

170102070

Electronics and Communication Engineering, B.Tech.

9963696940

khage170102070@iitg.ac.in

No need to load images in matlab files. The folder as a whole will take care of path. Just run the file.

1. Histogram Specification

The aim of this problem is to compute a transformation function in such a way that the transformed image histogram is close to that of the target image histogram.

Input – Original Image, Target Image.

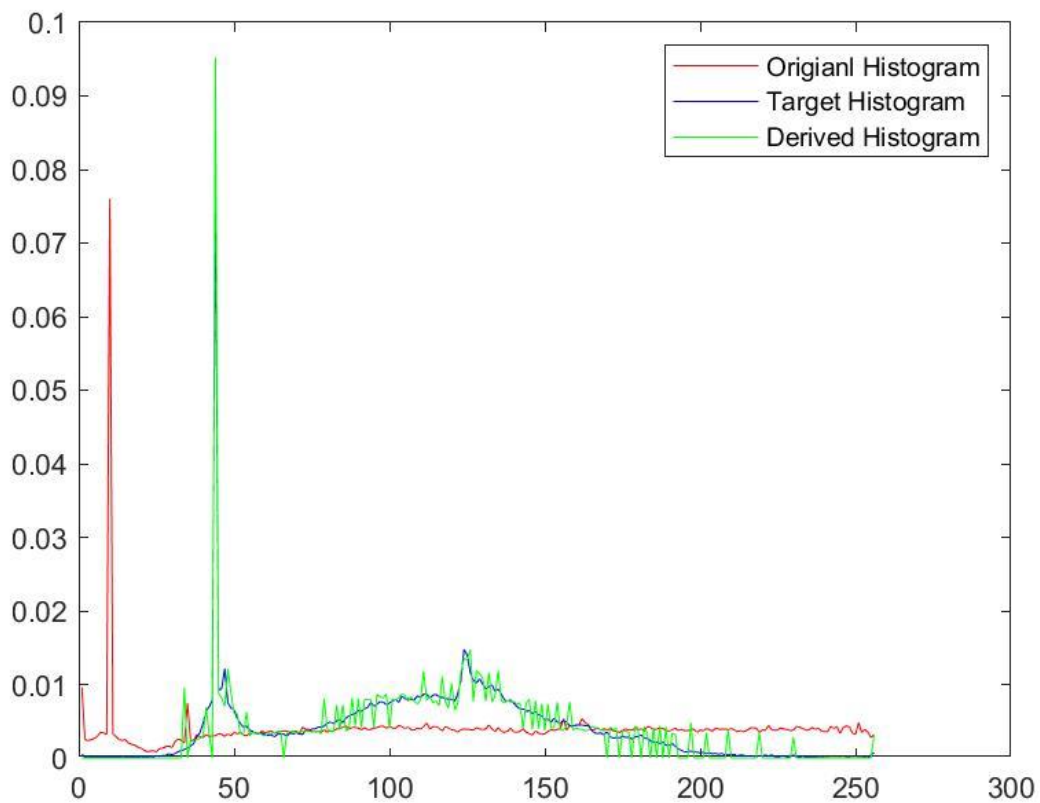
Output – Histogram comparison between original, target and transformed image histograms.



Original Image (givenhist.jpg)



Target Image(Equitized Image) (sphist.jpg)



Histogram for original, target and transformed images with specified colours.

2. Bilateral Filtering

Bilateral filter is used to preserve edges with the fact that it depends largely on intensity(range filter) variation rather than spatial(domain filter) only. The filter is more sensitive to standard-deviation of range filter(σ_r), and less sensitive to standard-deviation of domain filter(σ_d). Let us experiment on some images and discuss some observation.

White noise in dark background: White pixels correspond to high pixel intensity and dark pixel corresponds to low pixel intensity. Let us consider only one center white pixel in between dark pixels. In this case the coefficients in range filter are very low because the difference is very high(white pixel intensity – dark pixel intensity) and exponent of -ve of this high value is very low. So the overall filter doesnot vary much and the final pixel intensity is almost same earlier, because the dark pixels are nullified because low weight is given to them and they add to negligibl amount to the filter calculation. **Result:** So, very bright pixels are retained and pixels relative to background dark pixels are filtered.

Black noise in white background: The procedure is similar to that in the above case but the main difference is here the bright pixels are nullified retaining the dark pixels.

- i) Input and output for **spnoisy.jpg**
Mask size = 5
Sigma_r = 80 (more than 80 will cause blurring and less than 80 will not remove noise)
Siggm_d = 40 (any value between 10+ works same).



- ii) Input and output for **spunifnoisy.jpg**
Mask size = 5
Sigma_r = 70 (more than 80 will cause blurring and less than 80 will not remove noise).
Siggm_d = 40 (any value between 10+ works same).
Remark: The output is almost as expected and the noise iss removed with much less sigma_r when compared with previous example.



Zoom in into the final image(The noise intensity similar to background intensity is removed)
Only the opposite pixel intensity noise is not removed as this type of noise cannot be removed using bilateral filter. Higher values of sigma_r will further remove noise.

iii) Input and output for **unifnoisy.jpg**

Mask size = 5

Sigma_r = 20

Sigma_d = 10 (any value between 10+ works same).

Remark: The output is almost as expected and the noise is removed with much less sigma_r when compared with previous example.



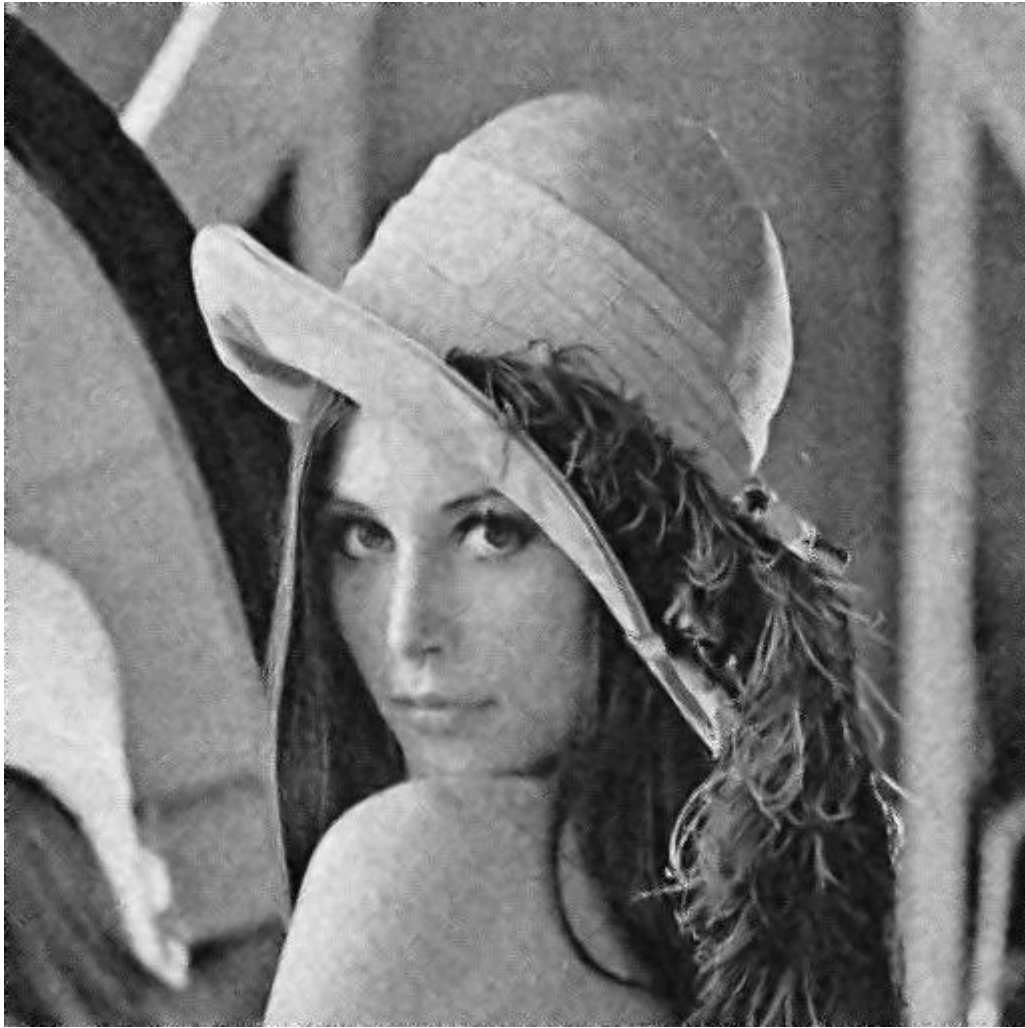
3. Edge preserving Smoothing Filters

Non-local means: This technique preserves the edges (which is also expected from bilateral filtering), which uses the concept of similarity between two neighbourhood patches (unlike bilateral filtering which uses the concept of similarity between two neighbourhood pixels). So, for similarity between patches we use **Gaussian Weighted Sum of Square Differences (GSSD)**, with standard-deviation = σ_d .

Observation: $\sigma_d = 10$, $h = 30$ gives good filtering for **lenna.noisy.jpg** image. The value of 'h' should be large so as to decrease the weight of GSSD which can be large. Even here σ_d does not have much significance.



Original Image



$h = 30, \sigma_d = 10$

4. Corner Detection

Harris Corner Detection: The technique here used is based on the change of function of sum of difference in neighbouring patch pixels. This approach is computationally very expensive, so we can reduce the time complexity by calculating the second moment matrix whose time complexity is the order of number of pixels in that particular patch(patch size is taken to be constant).

Here are some images with edges included.

i) Threshold = 10^{10}

Harris Corner Detection followed by Non-maximul Supression



Harris Corner Detection followed by Non-maximul Supression



ii) Threshold = 10^9

Harris Corner Detection followed by Non-maximul Supression



Harris Corner Detection followed by Non-maximul Supression

