Image Enhancement

CS - 8009

By - Ashwani Devi (24MECC01)
Guide: Dr. Akshay Pandey

Code:
https://github.com/Ashwani132003/Image_Processing_Python

What is Image Enhancement?

- Image enhancement is to process any image so that the resulting image is better suited to a specific application than the original image.
- It sharpens and emphasizes image features such as edges, boundaries, and contrast to improve the usability of a graphic display for display and analysis.

Image enhancement is subjective in nature.

- Because different applications and users have different goals.
- Example:
  - **Medical Imaging**: Clarity and contrast are crucial to highlight specific structures.
  - **Astronomical Imaging**: Noise reduction and detail preservation are essential.
  - **Photography**: Aesthetics, sharpness, and color balance matter more.
  - **Forensics**: Enhancing small details without introducing artifacts is critical.

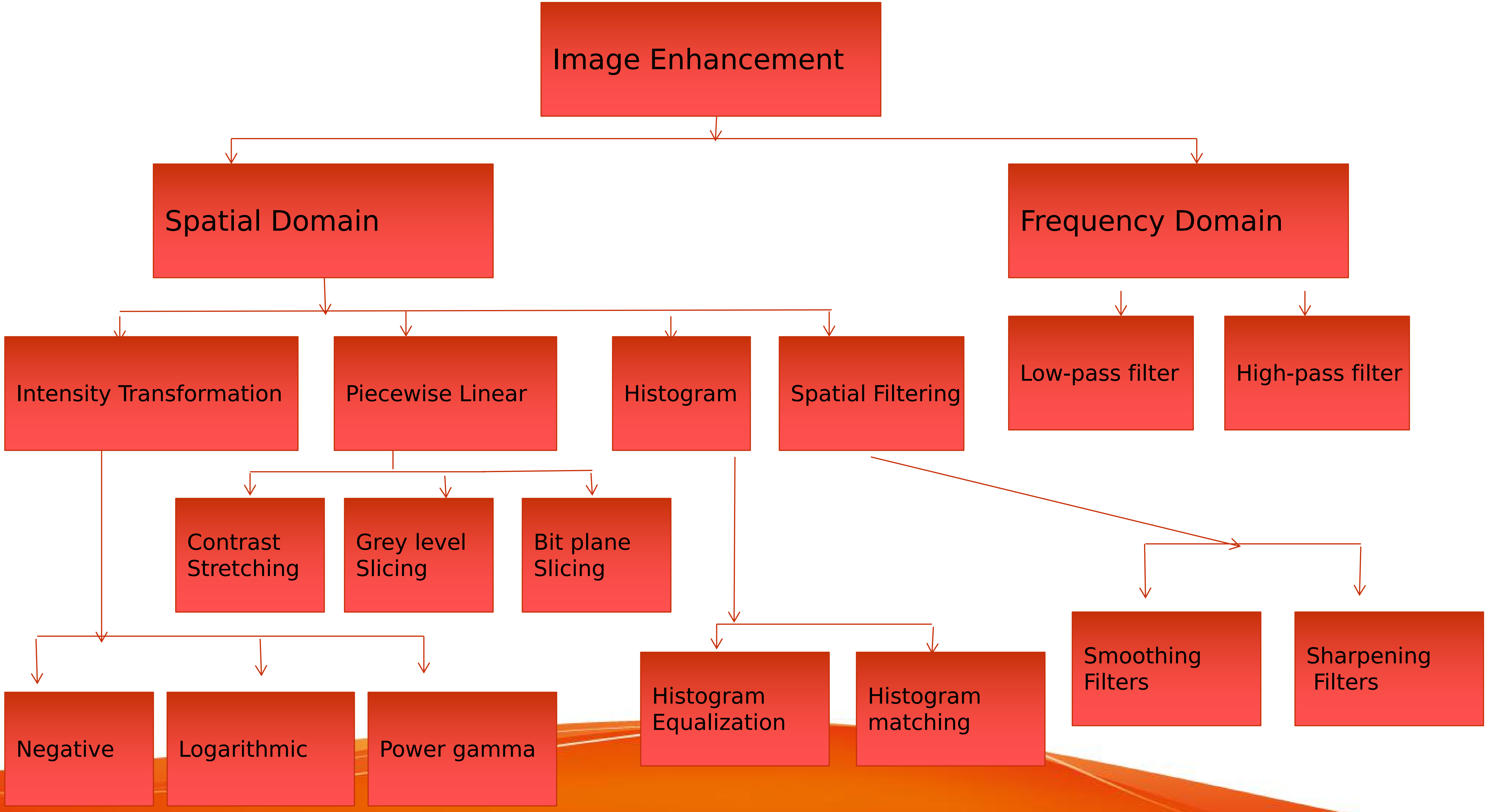


Image Enhancement
Spatial Domain
Frequency Domain
Intensity Transformation
Piecewise Linear
Histogram
Spatial Filtering
Low-pass filter
High-pass filter
Contrast Stretching
Grey level Slicing
Bit plane Slicing
Negative
Logarithmic
Power gamma
Histogram Equalization
Histogram matching
Smoothing Filters
Sharpening Filters

Spatial vs Frequency domain

- **Spatial domain** techniques manipuletes the pixels of an image directly. This process happens in the image’s coordinate system, also known as the spatial domain.
- **Frequency domain** techniques transforms an image from the spatial domain to the frequency domain. In this process, Mathematical transformations (such as the Fourier transform) are used. The image can be modified by manipulating its frequency components.

Image Negative

- The image negative transformation inverts the pixel values of an image, making dark areas appear bright and vice versa. This is useful for:
  - Enhancing contrast in medical imaging
  - Making details visible in dark regions
  - Creating artistic effects

- Let the image has an intensity level in the range [0 L-1], then the intensity after, negative transformation is given by:
  - **$s = L-1-r$**
- For 8-bit image, $s = 255-r$ (r is the original pixel value)

Original Image

Negative Image


Original Image
Negative Image


Original Image
Negative Image

Logarithmic Transformation

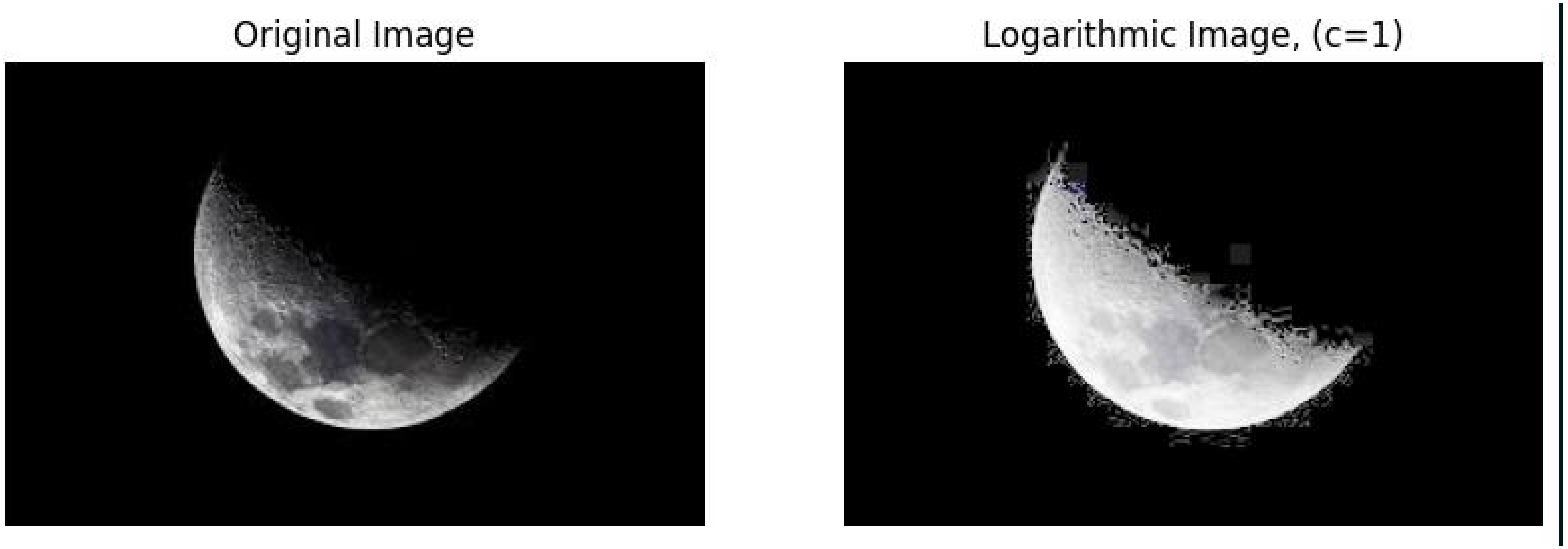
- Used in image enhancement to improve the visibility of darker regions in an image while compressing the intensity of brighter regions.

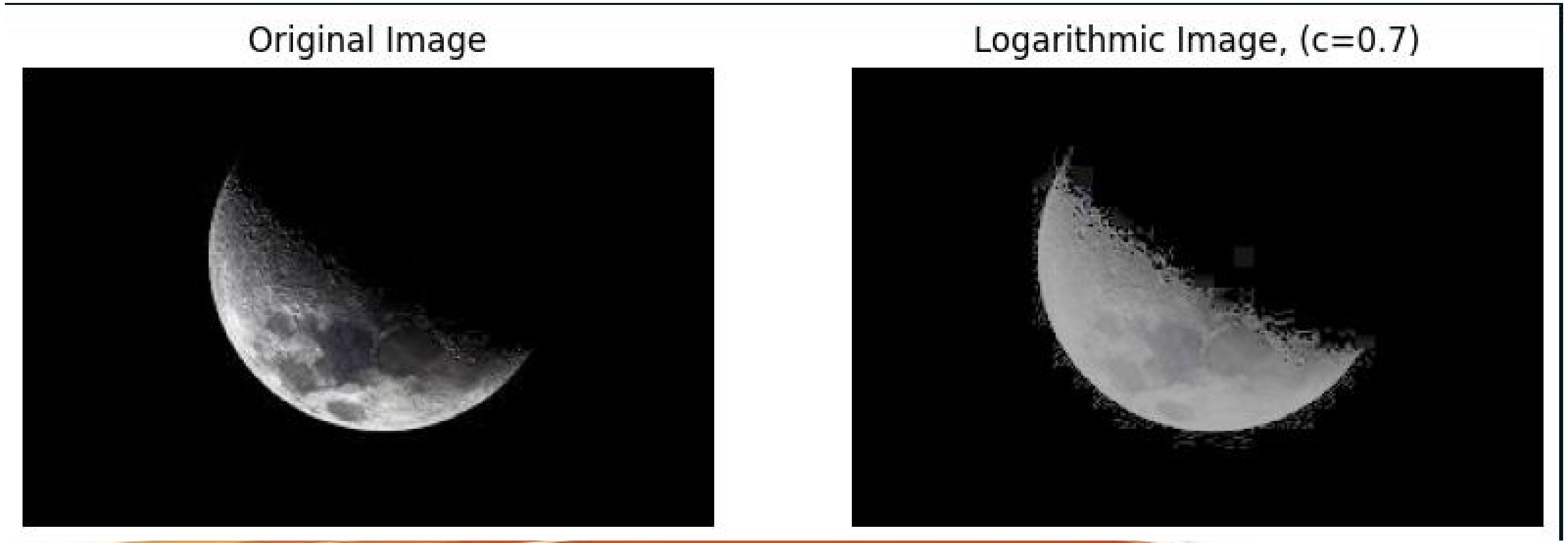
- The Log function has the important characteristic that it compresses the dynamic range of images with large variation in the pixel value.

$$s = c * \log(1+r)$$ [c=scaling factor/constant]

- When to Use?
  - Low-light images (e.g., Moon, medical X-rays, satellite images)
  - High contrast images (avoiding extreme brightness)

- Brightens Dark Pixels:
  - In many images, important details are hidden in dark areas (e.g., astronomical images, medical scans).
  - The logarithmic function expands lower intensity values to make them more visible.
- Compresses Bright Areas:
  - If an image has high-intensity values (too bright), log transformation compresses them, preventing excessive brightness.


Original Image
Logarithmic Image, (c=1)

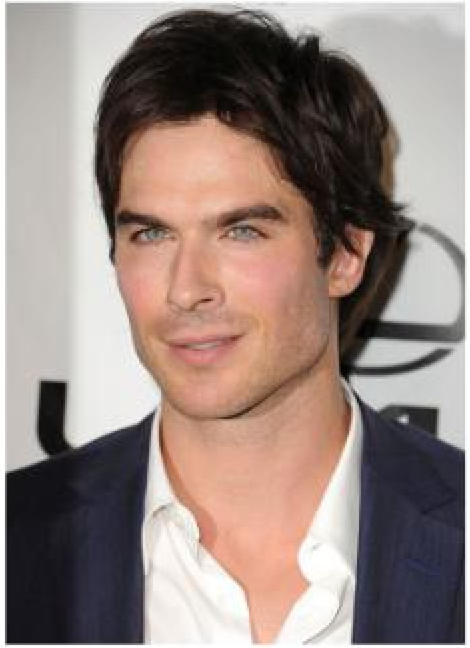

Original Image
Logarithmic Image, (c=0.7)

c=0.5

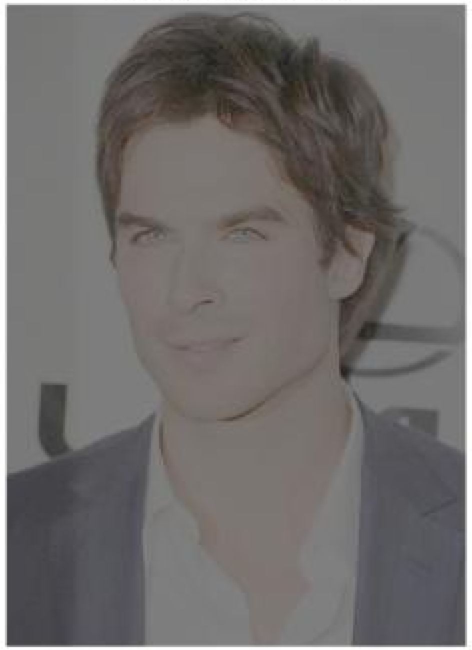
Original Image

Logarithmic Image

Original Image



Logarithmic Image



Original Image

Logarithmic Image

Power-law (Gamma) Transformations

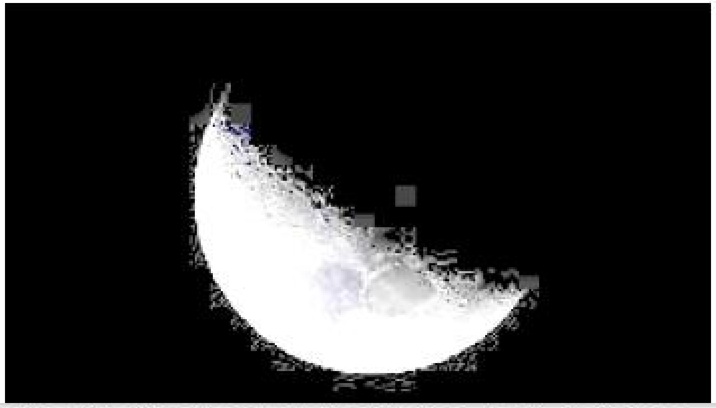
- This has the basic form $s=c*r^\gamma$,where c and γ (gamma) are positive constants.
- Fractional values of γ maps a narrow range of dark input values into a wider range of output values. Opposite of this also true for higher values of input
- levels.
- These are also called as gamma correction due to the exponent in the power law equation.

Power-law (Gamma) Transformations

- CRT device have an intensity to voltage response that is a power function with exponent varying from approximately 1.8 to 2.5.
- Such display system would produce images that are darker than intended.
- Gamma correction is very important when to reproduce an image exactly on a display system.
- gamma_h=1.5 → Enhances high frequencies (details, textures).
- gamma_l=0.5 → Suppresses low frequencies (uneven illumination)

Original Image

Gamma-Corrected Image, for c=1.5, gamma = 0.3

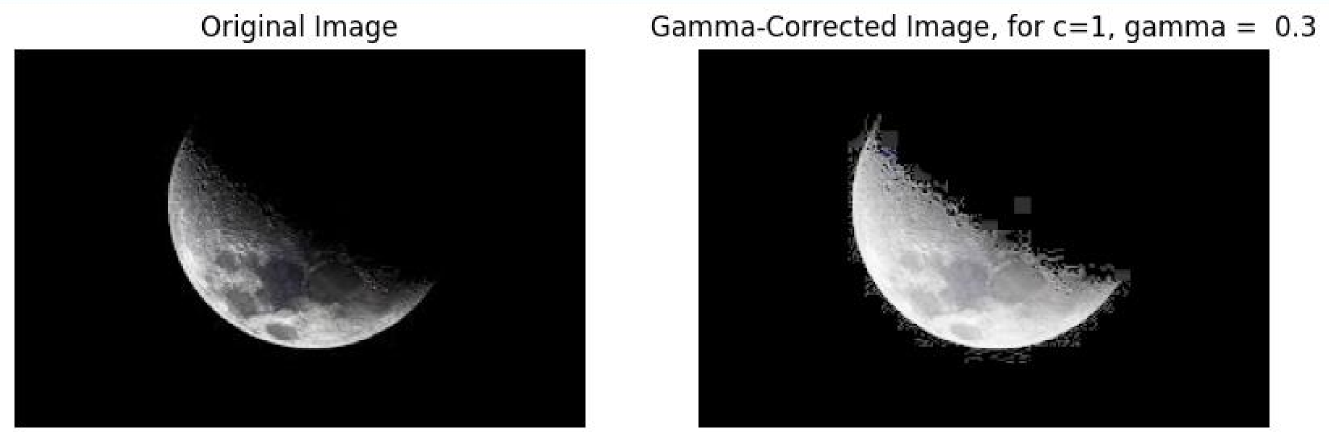


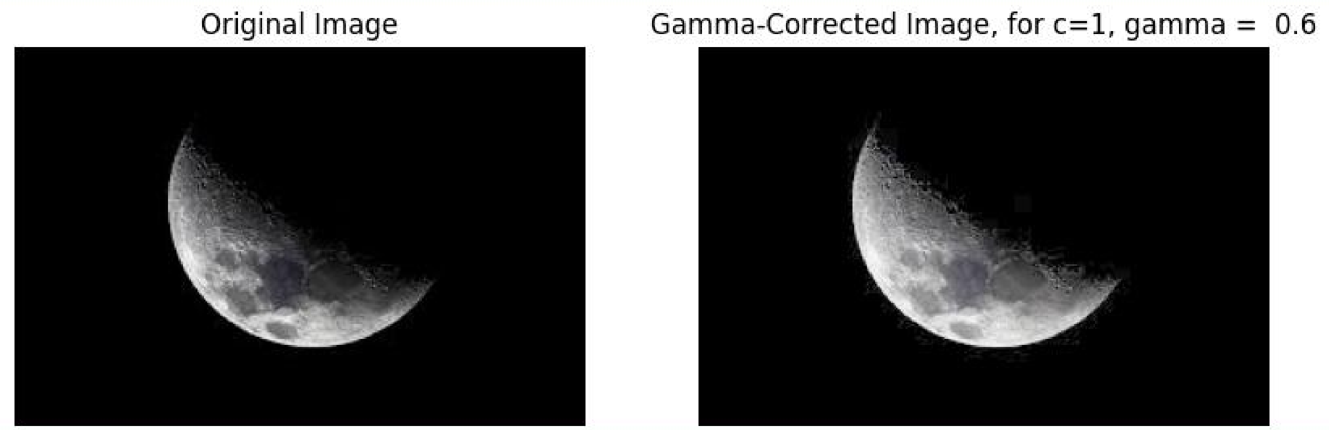
Original Image

Gamma-Corrected Image, for c=1.5, gamma = 0.6

Original Image

Gamma-Corrected Image, for c=1.5, gamma = 0.8


Original Image
Gamma-Corrected Image, for c=1, gamma = 0.3


Original Image
Gamma-Corrected Image, for c=1, gamma = 0.6


Original Image
Gamma-Corrected Image, for c=1, gamma = 0.8


Original Image
Gamma-Corrected Image, for c=1, gamma = 0.3
Original Image
Gamma-Corrected Image, for c=1, gamma = 0.6
Original Image
Gamma-Corrected Image, for c=1, gamma = 0.8

Logarithmic and power gamma transformations, for c = 3, gamma = 3:

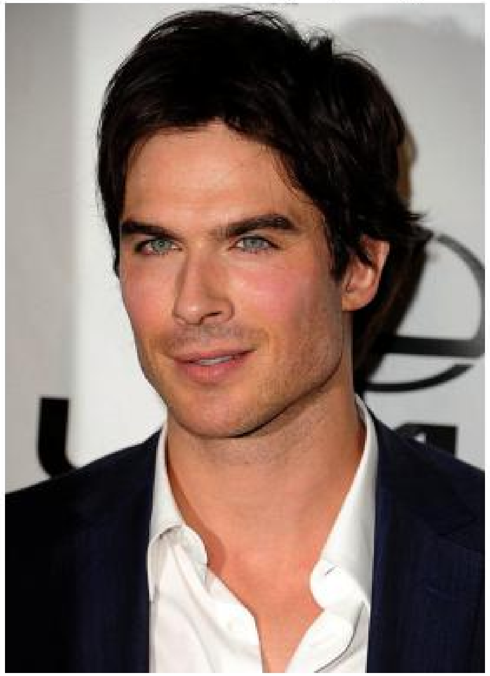
Original Image | Logarithmic + Gamma Image

Original Image | Logarithmic + Gamma Image

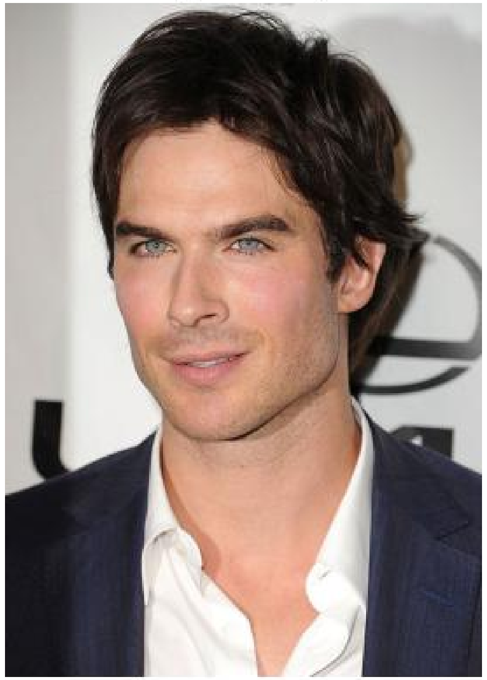
Original Image

Logarithmic + Gamma Image

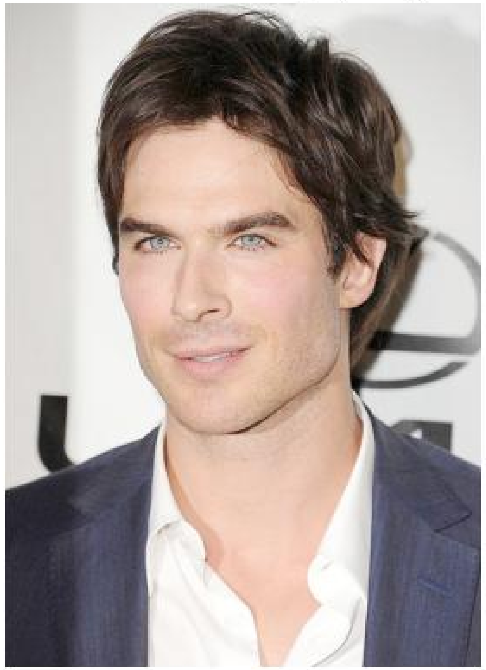
Logarithmic + Gamma Image (c=1,gamma=7.5)



Original Image



Logarithmic + Gamma Image (c=1,gamma=3)



Piecewise Linear Transformation

- It is used for the manipulation of an image so that the result is more suitable than the original for a specific application.

- a. Contrast-Stretching
- b. Gray-Level Slicing / Intensity Level Slicing
- c. Bit Plane Slicing

Contrast stretching / Normalization

- We stretch the minimum and maximum intensity values present to the possible minimum and maximum intensity values.

- Low contrast images result from the following:
  - Poor illumination
  - lack of dynamic range in the imaging sensor
  - Wrong settings of the lens aperture during acquisition

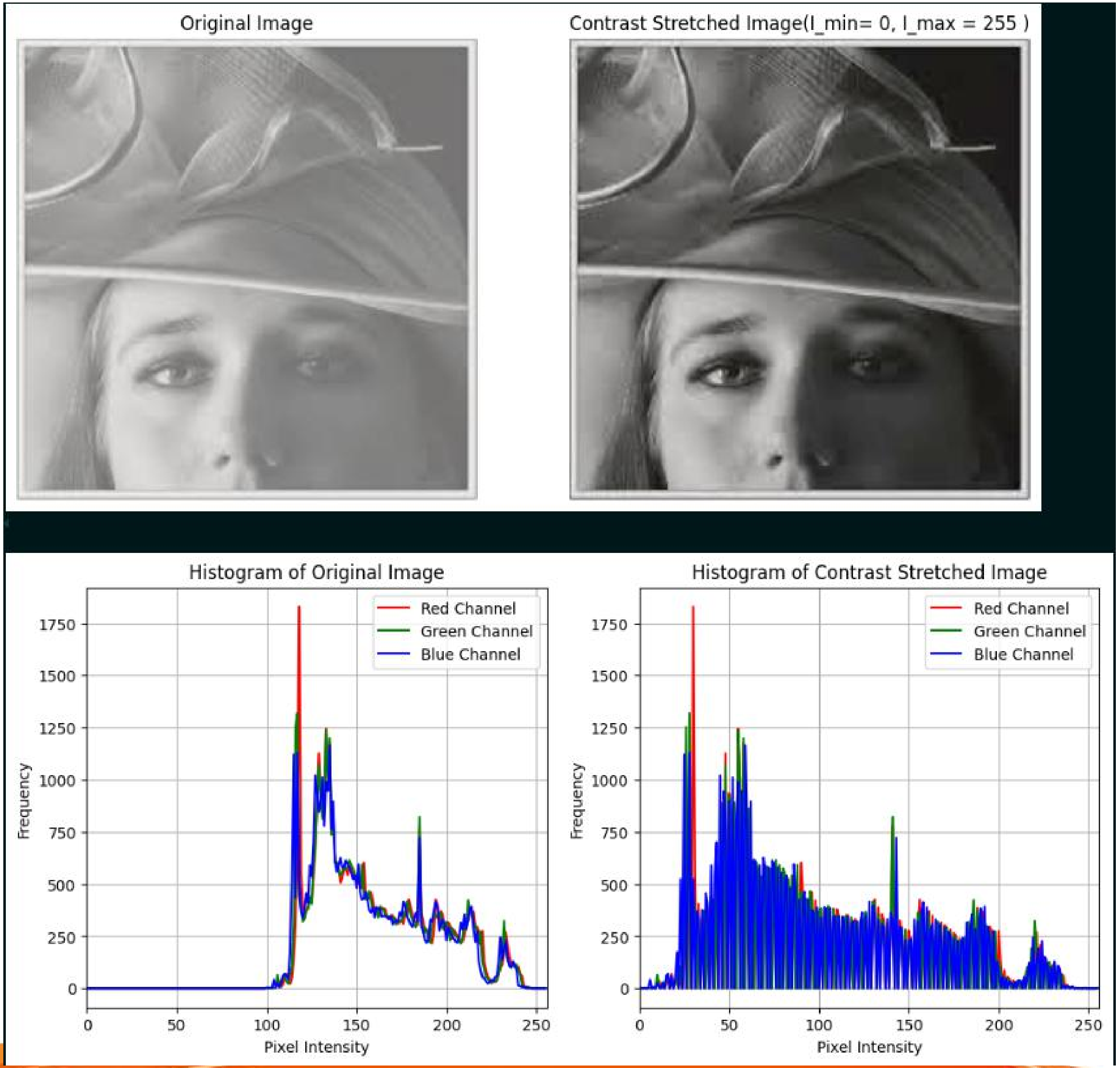
- Example: If the minimum intensity value(r_{min}) present in the image is 100 then it is stretched to the possible minimum intensity value 0. Likewise, if the maximum intensity value(r_{max}) is less than the possible maximum intensity value 255 then it is stretched out to 255.(0–255 is taken as standard minimum and maximum intensity values for 8-bit images)

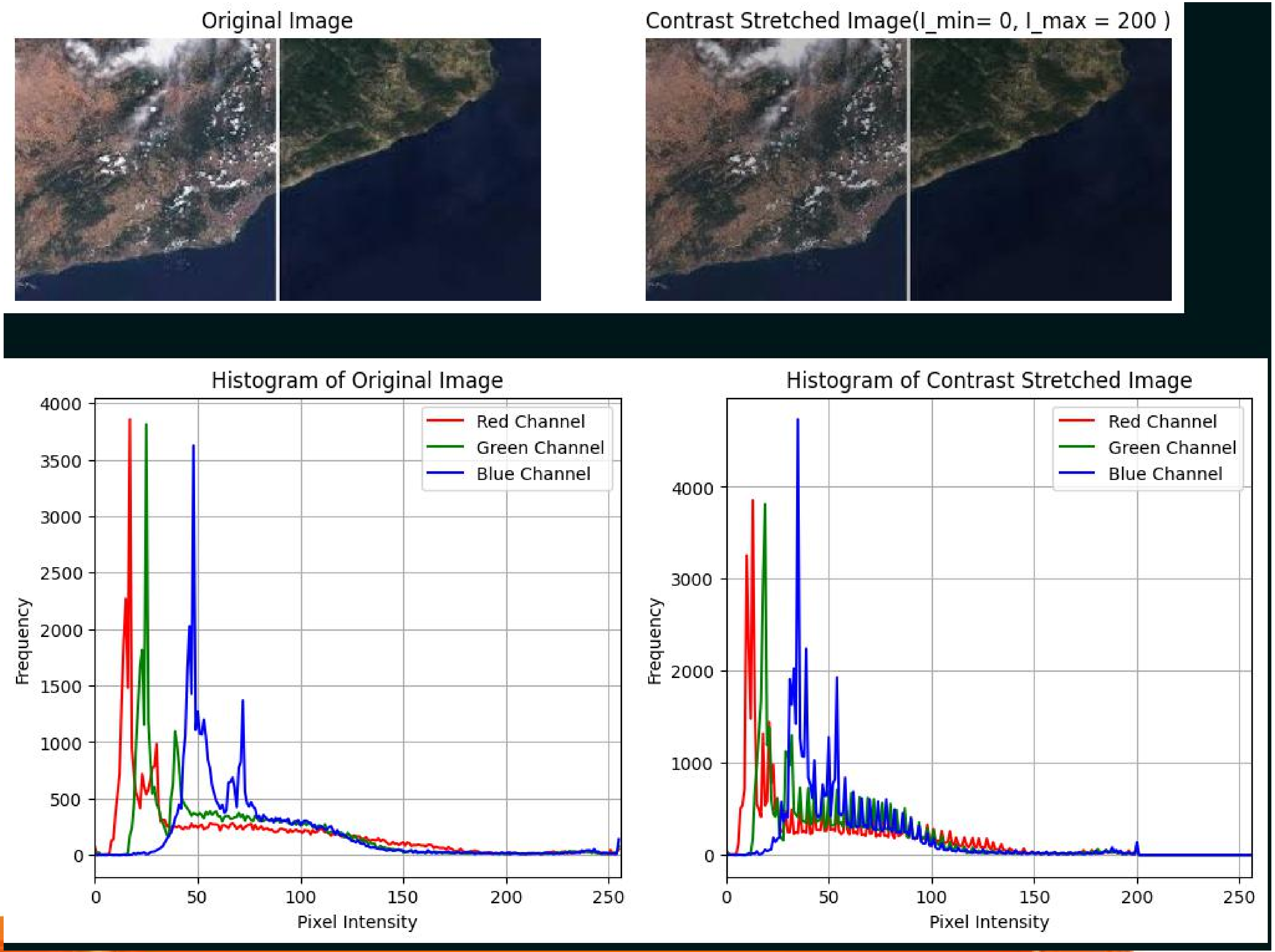
- $$s = \frac{(r - r\,min)\,(I\,max - I\,min)}{(r\,max - r\,min)} + I\,min$$

where,
r = current pixel intensity value,
r min = minimum intensity value present in the whole image,
r max = maximum intensity value present in the whole image

- For I min = 0 and I max = 255 (for standard 8-bit grayscale image)

$$s = 255 * (r - r\,min) / (r\,max - r\,min)$$


Original Image
Contrast Stretched Image(I_min= 0, I_max = 255)
Histogram of Original Image
Histogram of Contrast Stretched Image
Red Channel
Green Channel
Blue Channel
Frequency
Pixel Intensity
0
250
500
750
1000
1250
1500
1750
0
50
100
150
200
250

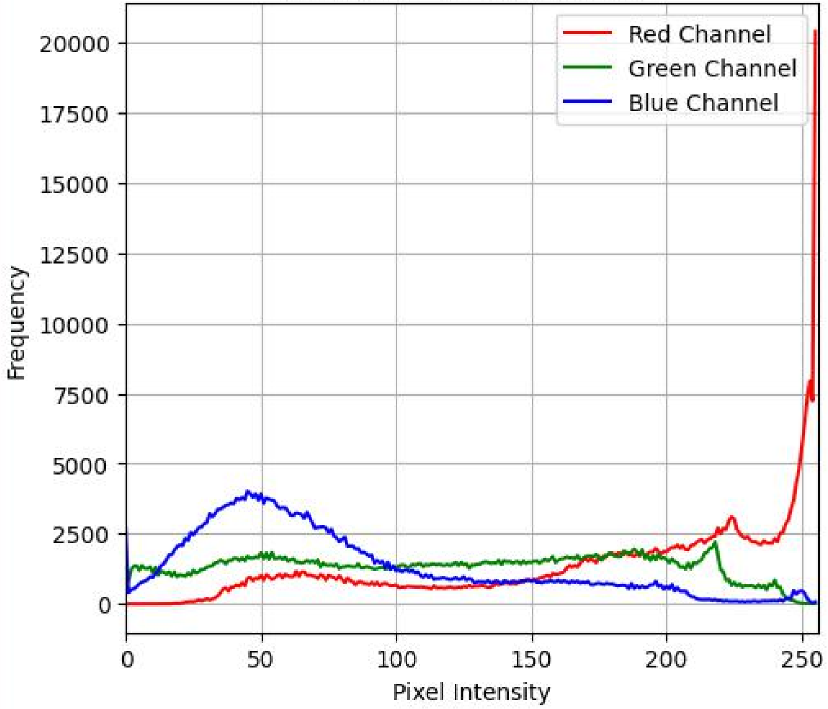


Original Image
Contrast Stretched Image(I_min= 0, I_max = 200)
Histogram of Original Image
Histogram of Contrast Stretched Image
Red Channel
Green Channel
Blue Channel
Frequency
Pixel Intensity

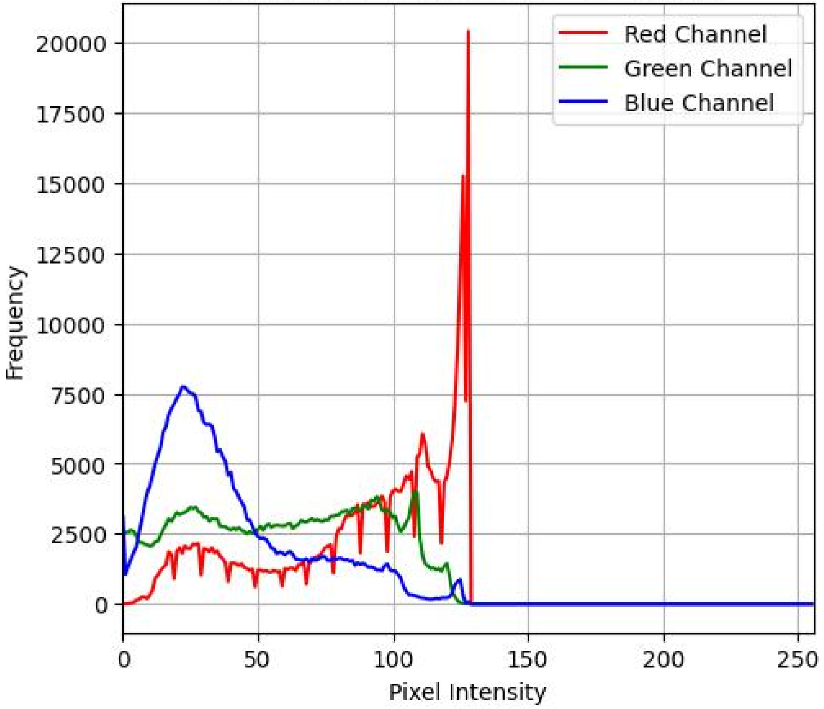
Original Image

Contrast Stretched Image(I_min= 0, I_max = 128)

Histogram of Original Image



Histogram of Contrast Stretched Image

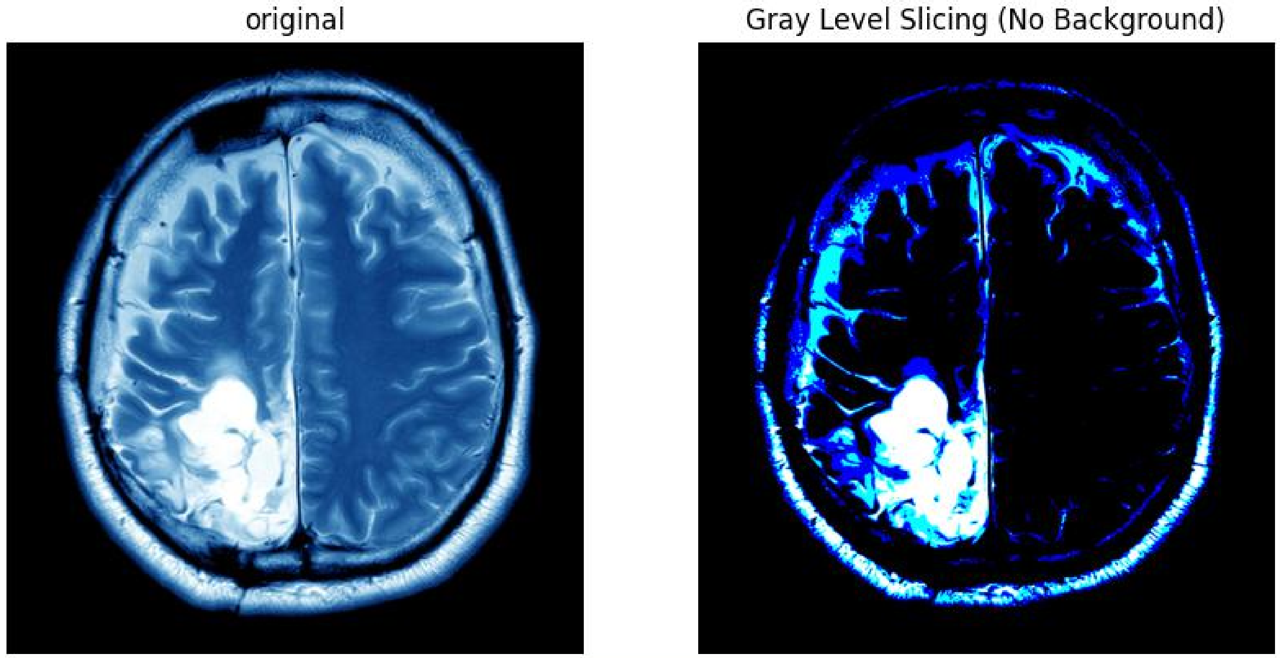


Gray-Level Slicing / Intensity Level Slicing

- Used to enhance certain ranges of pixel intensities while suppressing others.
- There are two types:
  - Without Background Preservation → Sets all other intensities to 0.
  - With Background Preservation → Keeps other intensities unchanged.
- Example :
  - Enhancing features such as masses of water in the satellite imagery.
  - Enhancing flaws in X-ray images.

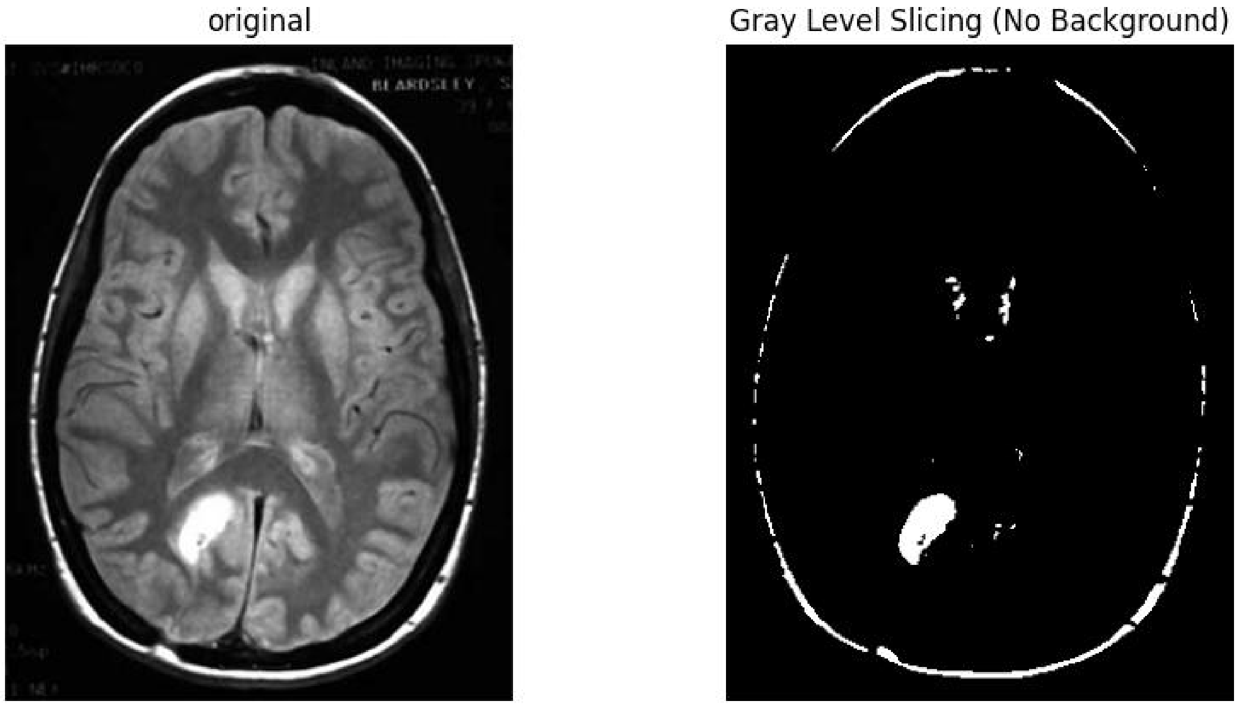
- if I_min <= pixel intensity <= I_max then enhance
- else keep the pixel intensity as it is or make it 0.

- enhancement could be making the pixel 255(for 8 bit) or scaling the pixel (ex. new_pixel = pixel*2)



min_intensity = 200 # Lower threshold
max_intensity = 255 # Upper threshold

min_intensity = 200
max_intensity = 255



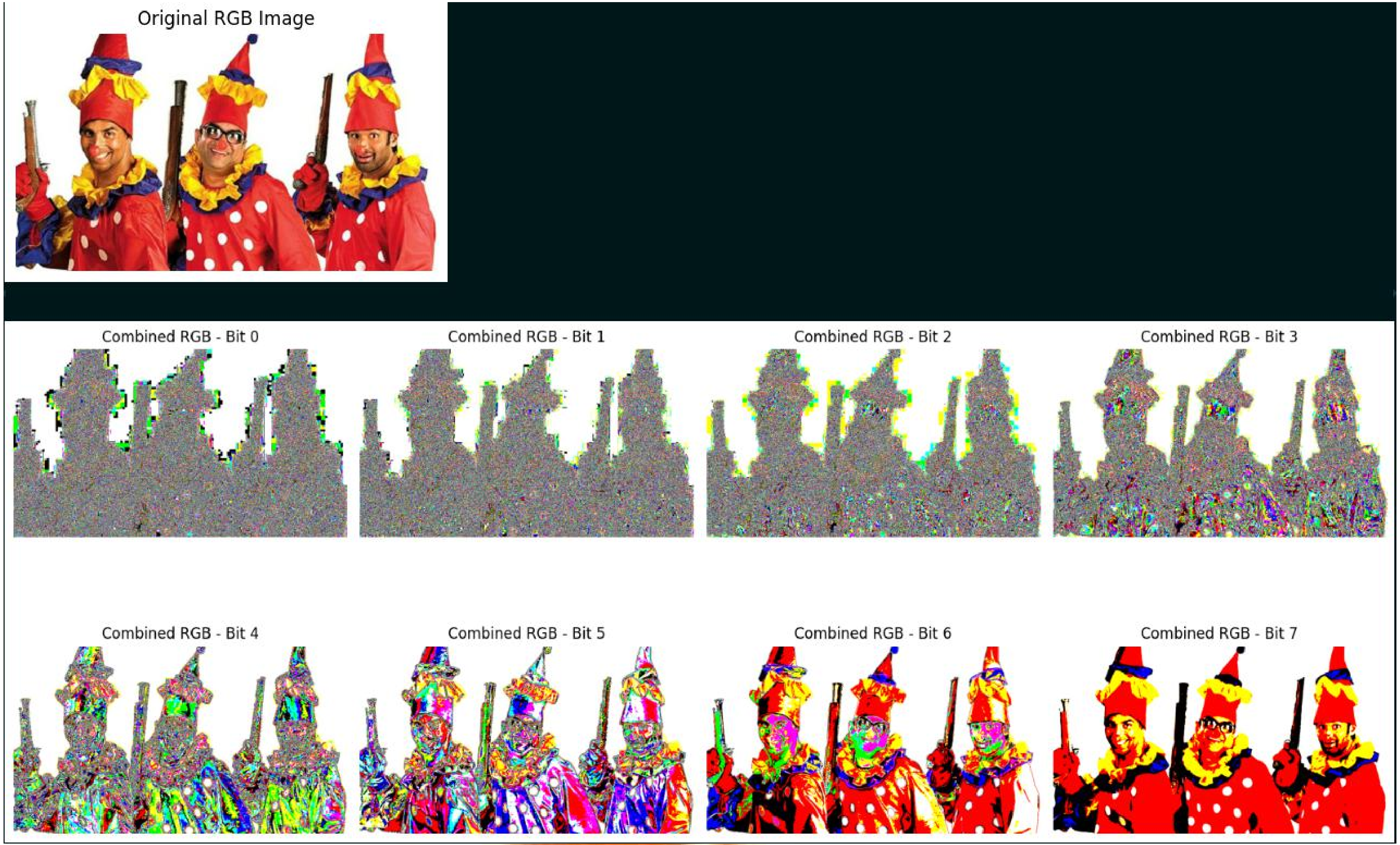
Bit Plane slicing

- Each pixel is composed of bits, intensity of each pixel in 256 grayscale image is composed of 8 bits.
- Instead of highlighting grey level range, we could highlight the contribution made by each bit.
- By isolating particular bits of the pixel values in an image we can highlight interesting aspects of that image.

Bit Plane slicing

- Higher-order bits usually contain most of the significant visual information.
- Lower-order bits contain subtle details.
- Separating a digital image into its bits plane is useful for analyzing the relative importance played by each bit of image.
- It helps in determining the adequacy of the number of bits used to quantize each pixel.

- Example:
  - Each pixels are digital number comprising of bits
  - For a 256 level gray-scale image there are 8 bits for each pixel
  - We can highlight the contribution of these bits to total image appearance


Original RGB Image
Combined RGB - Bit 0
Combined RGB - Bit 1
Combined RGB - Bit 2
Combined RGB - Bit 3
Combined RGB - Bit 4
Combined RGB - Bit 5
Combined RGB - Bit 6
Combined RGB - Bit 7

Histogram Processing

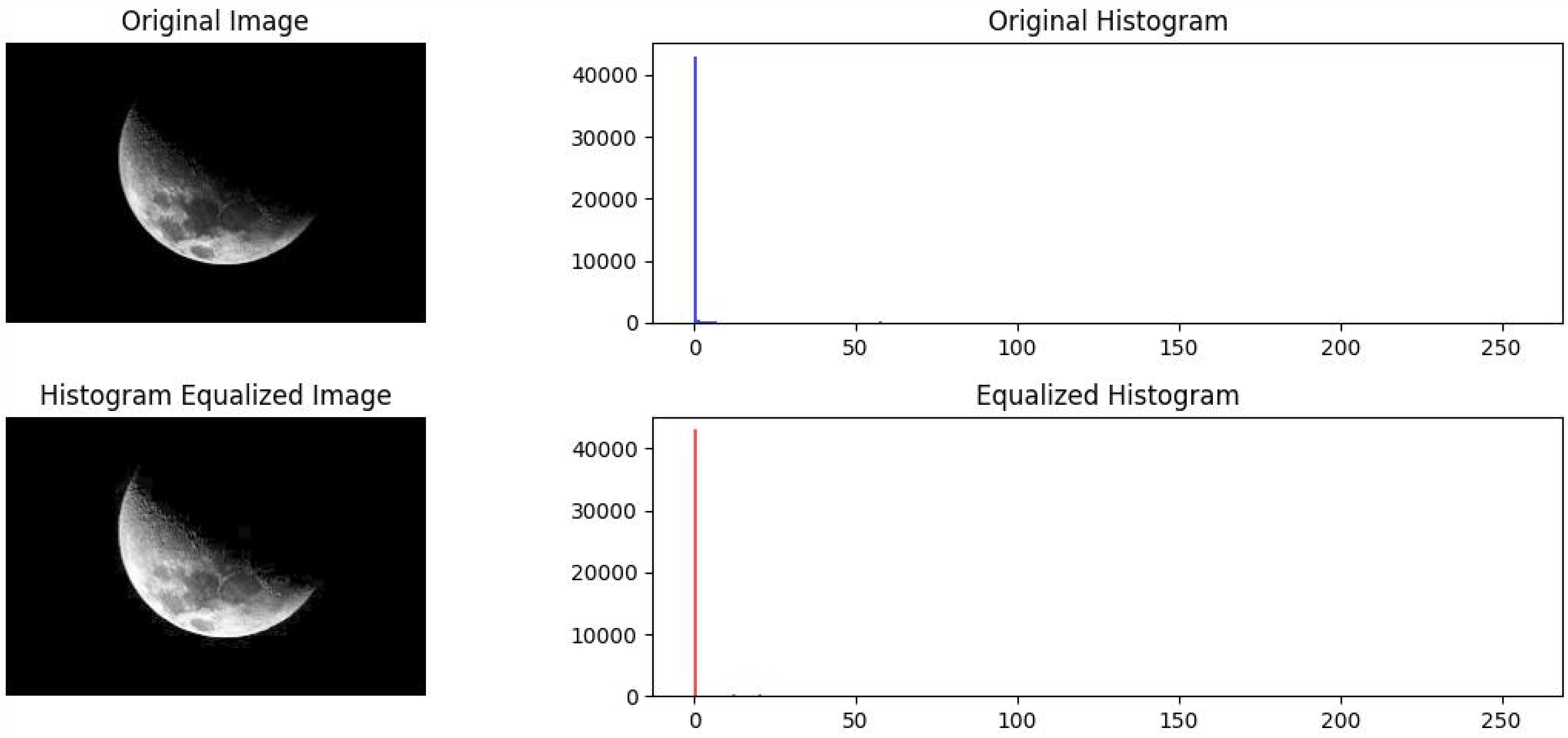
Involves analyzing and manipulating the distribution of pixel intensities, represented by an image's histogram, to enhance visual quality and interpretability.

- Histogram Equalization
- Histogram Matching

Histogram Equalization

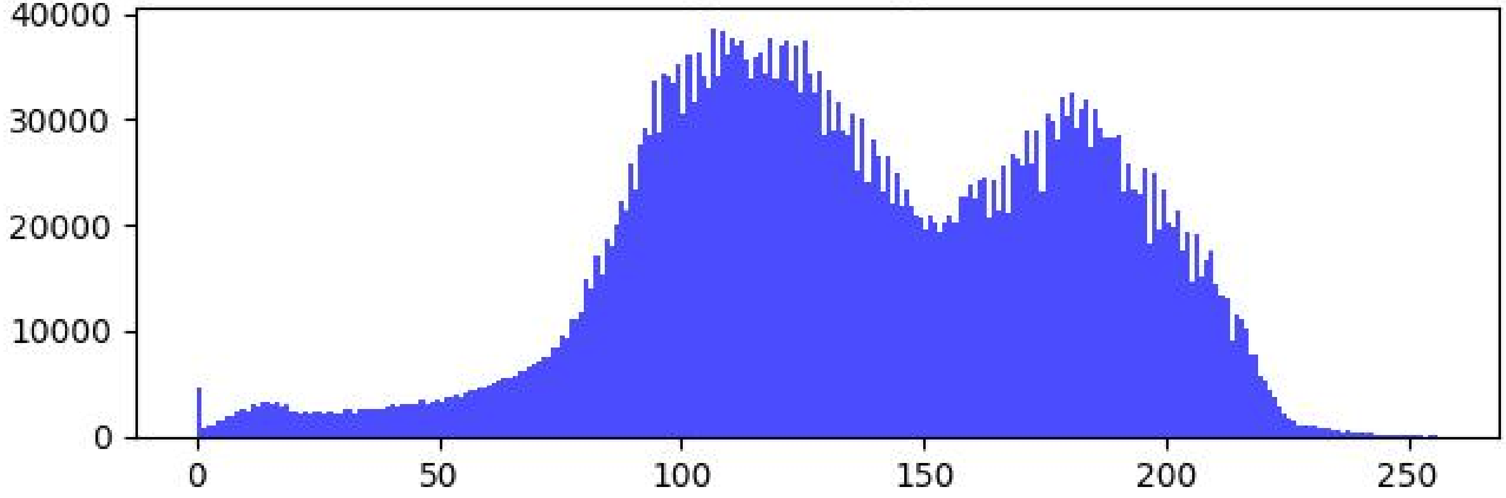
- Histogram equalisation is a nonlinear process aimed to highlight image brightness in a way particularly suited to human visual analysis.

$$I' = \mathrm{round}\left(\frac{\mathrm{CDF}(I) - \mathrm{CDF}_{\min}}{\text{Total Pixels} - \mathrm{CDF}_{\min}} \times 255\right)$$


Original Image
Original Histogram
40000
30000
20000
10000
0
0
50
100
150
200
250
Histogram Equalized Image
Equalized Histogram
40000
30000
20000
10000
0
0
50
100
150
200
250

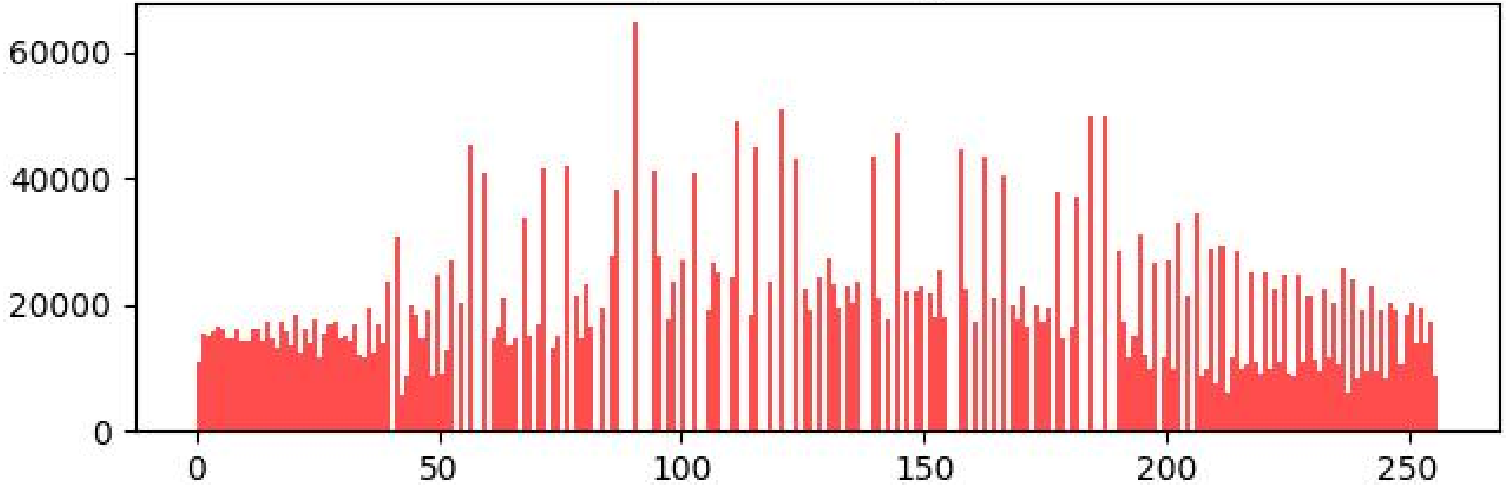
Original Image

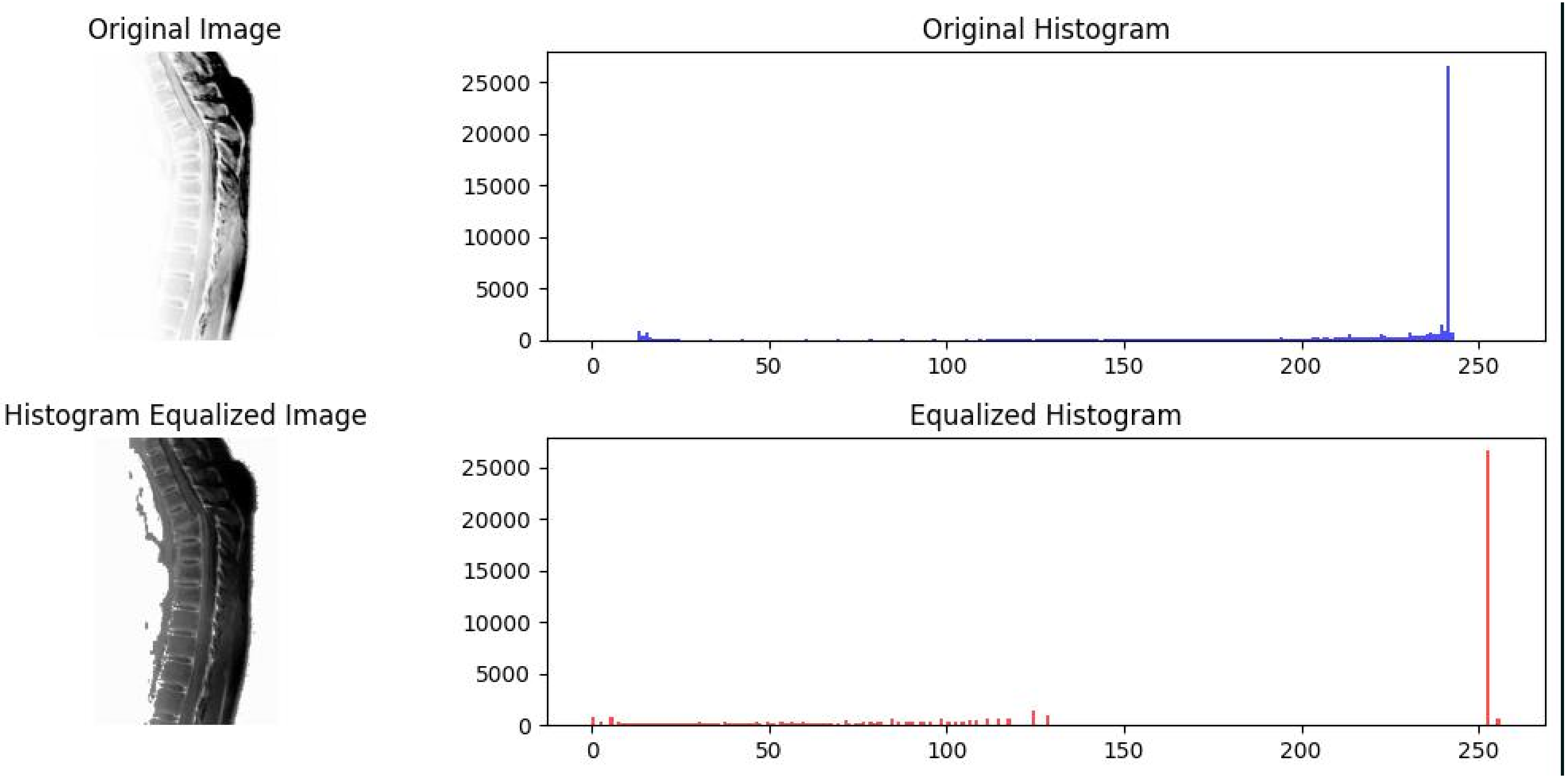
Original Histogram



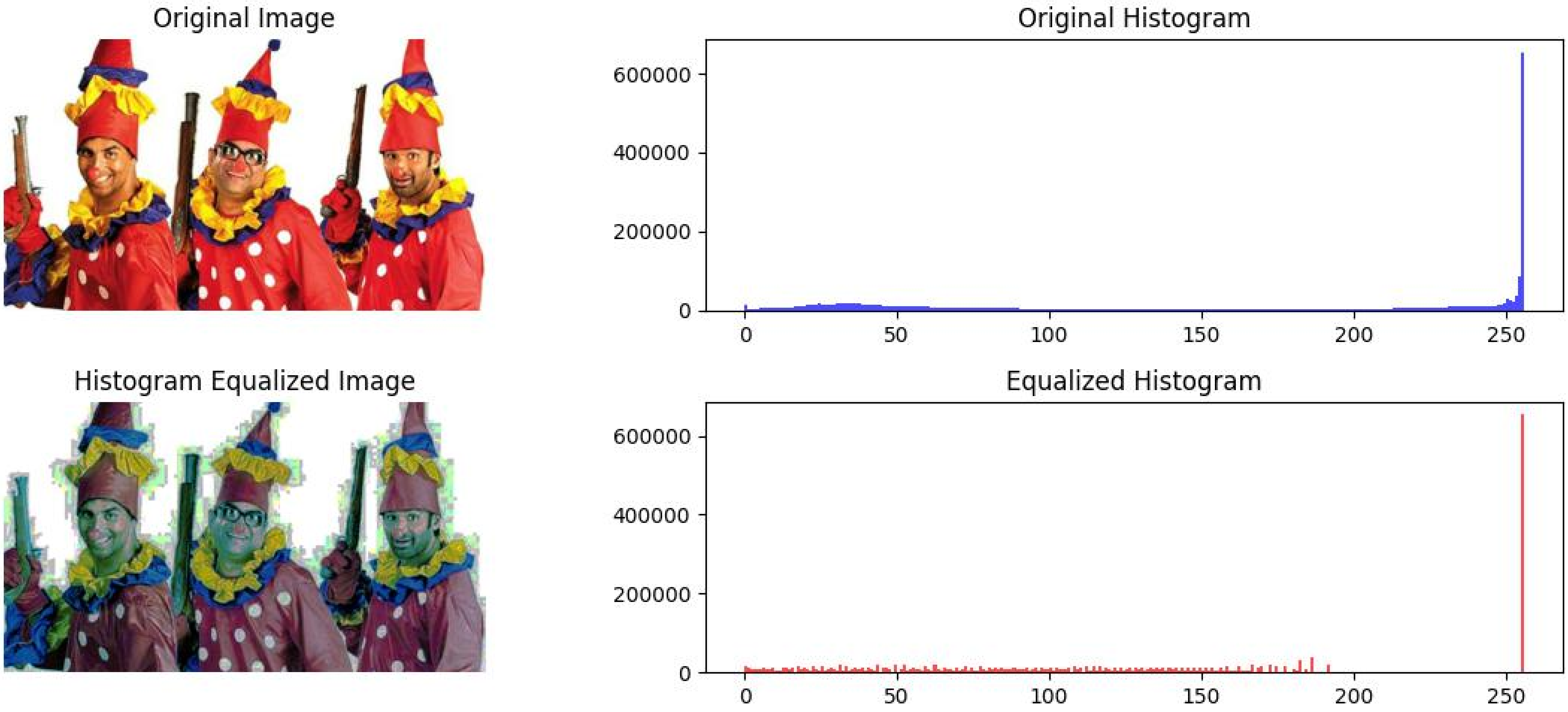
Histogram Equalized Image

Equalized Histogram





Original Image
Original Histogram
25000
20000
15000
10000
5000
0
0
50
100
150
200
250
Histogram Equalized Image
Equalized Histogram
25000
20000
15000
10000
5000
0
0
50
100
150
200
250


Original Image
Original Histogram
600000
400000
200000
0
0
50
100
150
200
250
Histogram Equalized Image
Equalized Histogram
600000
400000
200000
0
0
50
100
150
200
250

- Contrast Stretching differs from the more sophisticated histogram equalization in that it can only apply a linear scaling function to the image pixel values. As a result the `enhancement' is less harsh in contrast stretching.

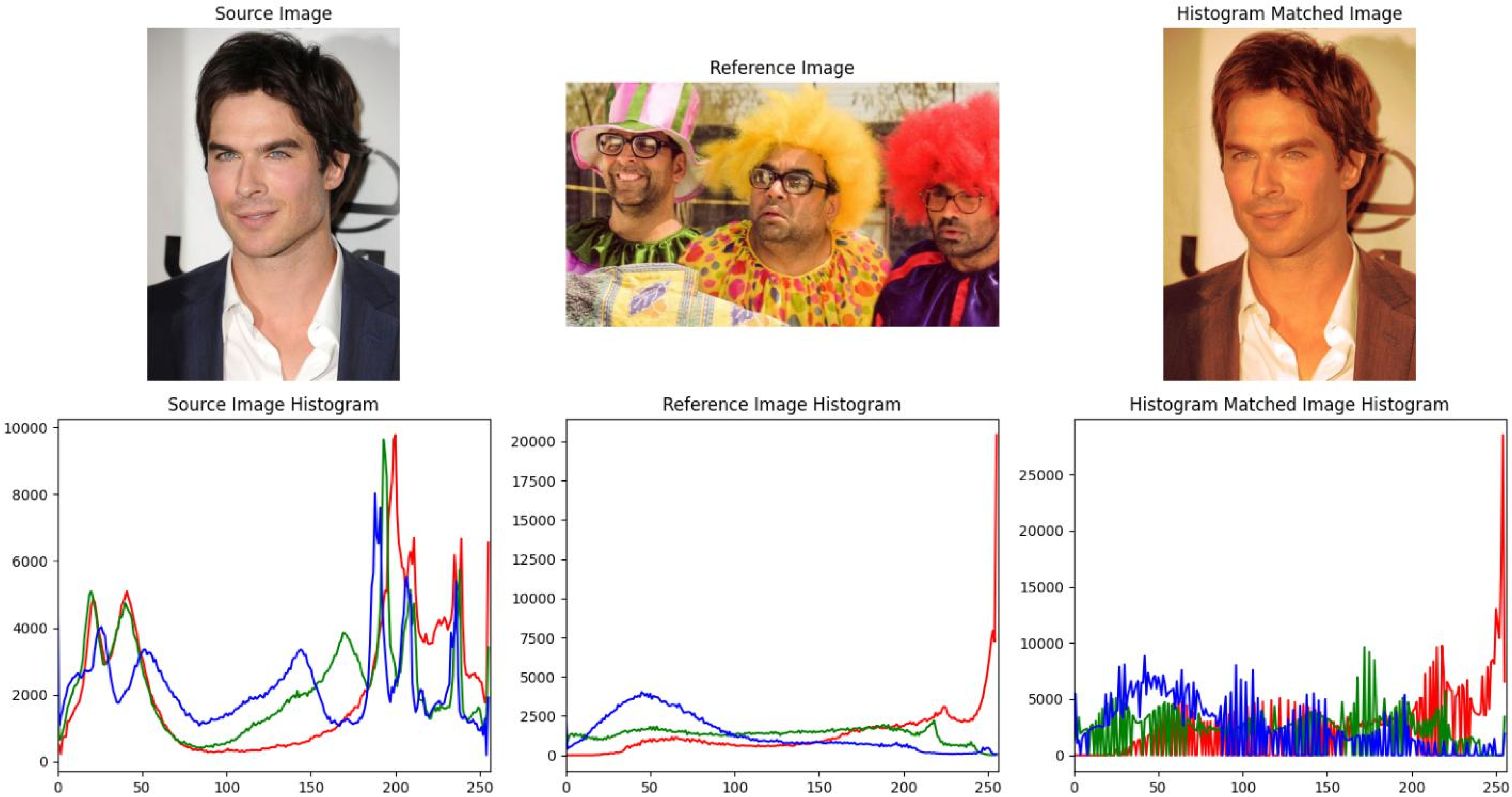
- Use Histogram Equalization if:
  - The image has low contrast or uneven lighting.
  - You want to enhance dark or bright details.
  - Example: X-ray images, satellite images, medical imaging.

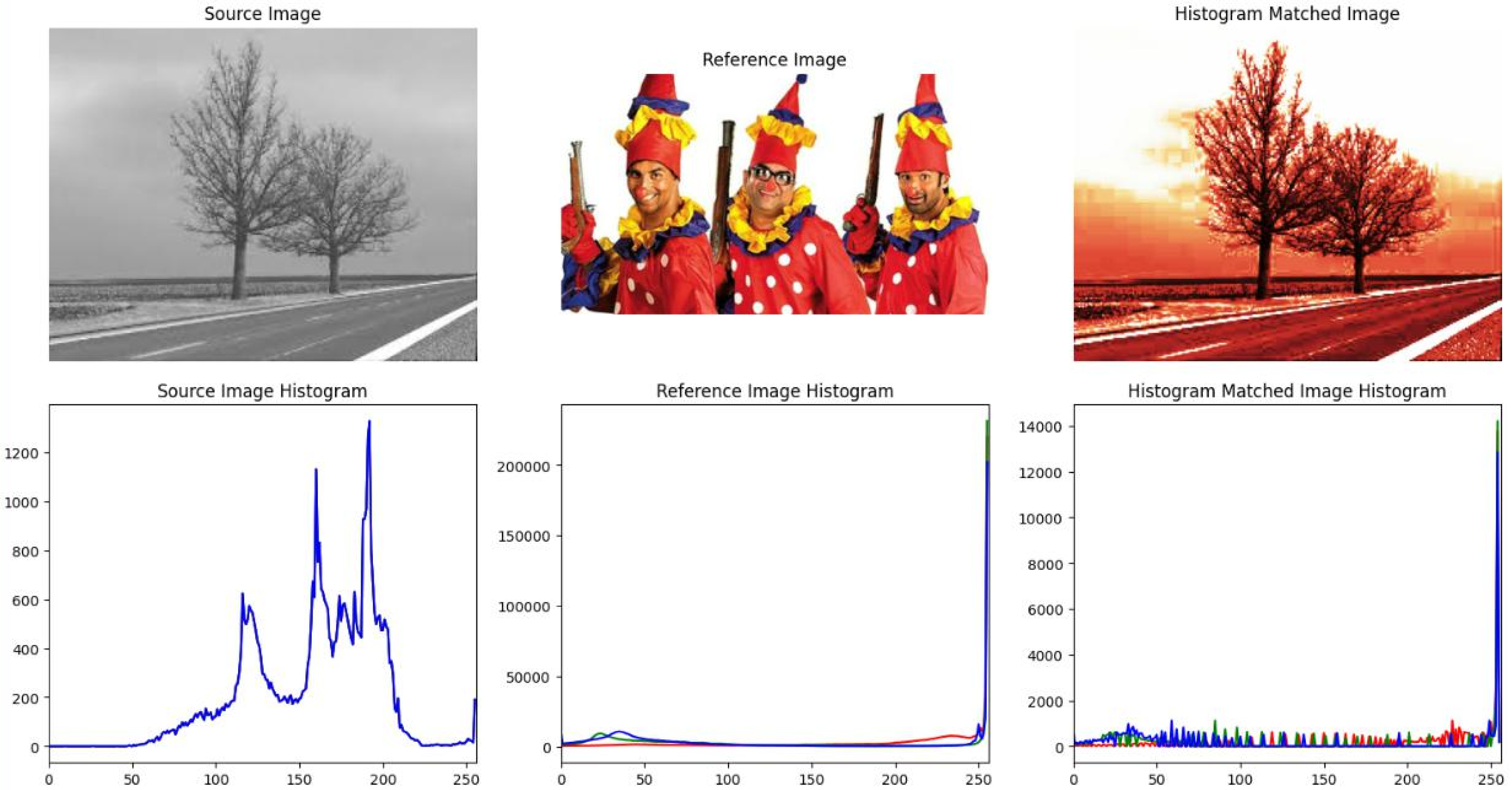
- Use Contrast Stretching if:
  - The image already has contrast but needs better distribution.
  - You want a controlled enhancement without histogram changes.
  - Example: Photos taken in foggy conditions.

Histogram Matching

- Histogram matching is a technique used to adjust the pixel intensity distribution of one image to match another image (reference). This is useful in:
- ✔ Image normalization
- ✔ Enhancing details
- ✔ Making images look visually similar

- Steps for Histogram Matching
  - Load the source & reference images.
  - Compute the cumulative distribution function (CDF) of both images.
  - Create a mapping function to adjust pixel intensities.
  - Apply the transformation.


Source Image
Reference Image
Histogram Matched Image
Source Image Histogram
Reference Image Histogram
Histogram Matched Image Histogram
0 2000 4000 6000 8000 10000
0 2500 5000 7500 10000 12500 15000 17500 20000
0 5000 10000 15000 20000 25000
0 50 100 150 200 250


Source Image
Reference Image
Histogram Matched Image
Source Image Histogram
Reference Image Histogram
Histogram Matched Image Histogram
0
200
400
600
800
1000
1200
50000
100000
150000
200000
2000
4000
6000
8000
10000
12000
14000
50
100
150
200
250

Spatial Filtering

- This Technique is used directly on pixels of an image. Mask is usually considered to be added in size so that it has a specific center pixel. This mask is moved on the image such that the center of the mask traverses all image pixels. This type is further classified into 2 types:

- 1. Smoothing Spatial Filters

- 2. Sharpening Spatial Filters

Smoothing Spatial Filters

- Smoothing (or blurring) filters are used to reduce noise, remove unwanted details, and improve image quality before further processing.

- 1️⃣ Mean Filter → Averages pixel values in a neighborhood, reducing sharp changes.
- 2️⃣ Gaussian Filter → Uses a Gaussian kernel to smooth the image while preserving some edges.
- 3️⃣ Median Filter → Replaces each pixel with the median of surrounding pixels, effective for salt-and-pepper noise removal.

Noisy image

Filtered by Mean Filter

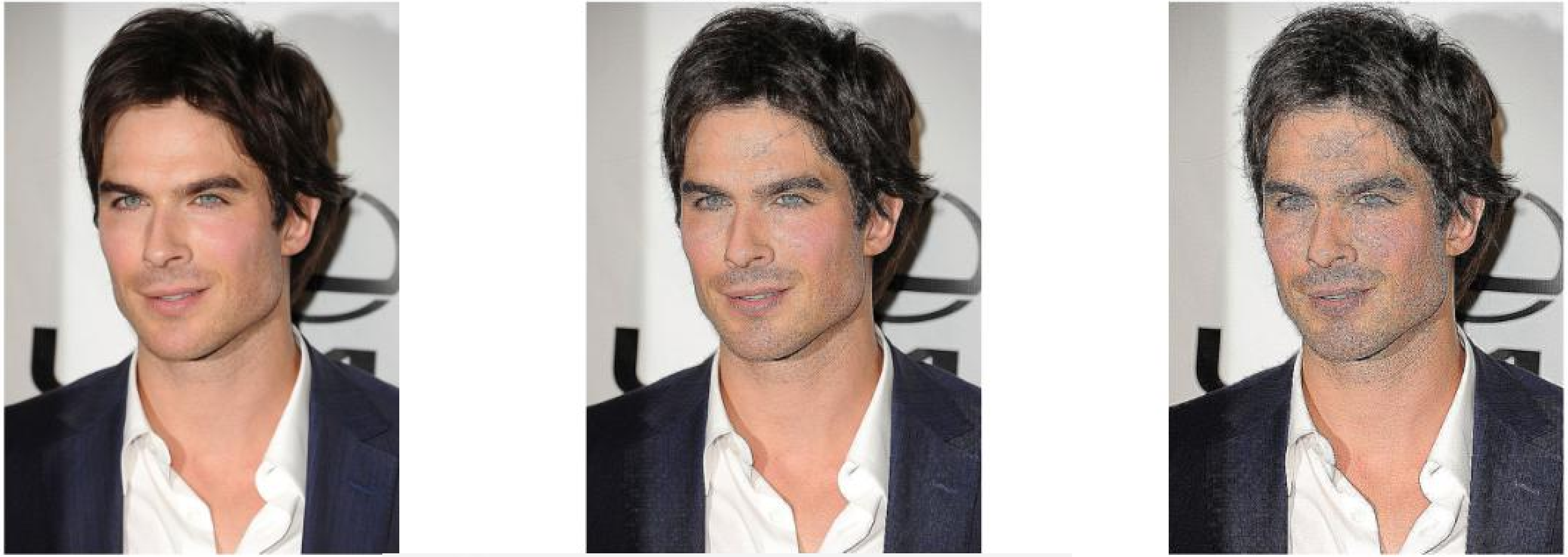
Filtered by Median Filter

Sharpening Spatial Filters

- Sharpening filters enhance edges and details by highlighting high-frequency components in an image.

- 1️⃣ Laplacian Filter → Uses the second derivative to enhance edges and sharpness.
- 2️⃣ Unsharp Masking → Creates a sharper image by subtracting a blurred version from the original.

Original | Laplacian Sharpening | Unsharp Masking



Original Image | Sharpened Image


Original
Laplacian Sharpening
Unsharp Masking

Frequency Domain

- The original image is transferred to the frequency domain in this technique, so the Fourier transform of the image is computed first. The images are processed based on their frequency content. The pixel value of the output image changes depending on the transformation function applied to the input values.
- The enhancement operation is used to change the brightness, contrast, or distribution of gray levels in an image. All enhancement operations are performed on the image’s Fourier transform, and then the inverse Fourier transform is applied to the resultant image.

Frequency Domain

- Frequency domain image enhancement refers to modifying an image by processing its frequency components rather than directly manipulating pixel values in the spatial domain. This is done using Fourier Transform (FT) to represent the image in terms of its frequency content.
- It is classified into 3 types:
  1. Low Pass Filtering
  2. High Pass Filtering
  3. Homomorphic Filtering

Low Pass Filtering

- Allows low-frequency components (smooth variations and coarse details) to pass while attenuating high-frequency components (sharp edges and noise).
- Key Characteristics:
  - Preserves smooth regions and large objects in an image.
  - Reduces sharp transitions, making images look smoother or blurred.
  - Removes high-frequency noise, making it useful for denoising.
- Use Cases:
  - Denoising: Removing random noise from an image.
  - Smoothing: Creating a softened version of the image.
  - Reducing Aliasing: Preparing images for downsampling.

Original Image

Low-Pass Filtered Image

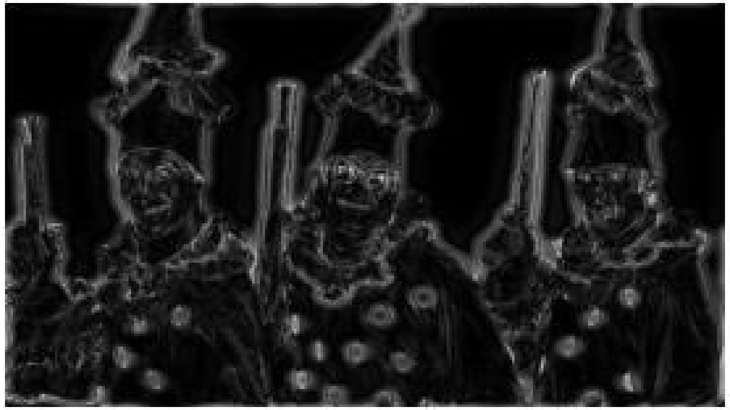
High-Pass Filter

- Allows high-frequency components (sharp edges, fine details, and noise) to pass while attenuating low-frequency components (smooth regions and gradual variations).
- Key Characteristics:
  - Enhances edges and fine details in an image.
  - Removes smooth variations, making images appear sharper.
  - Boosts high-frequency components, which include edges, textures, and noise.
- Use Cases:
  - Edge Detection: Extracting object boundaries.
  - Image Sharpening: Enhancing details in images.
  - Feature Enhancement: Highlighting fine structures in medical or satellite images.

Original Image

High-Pass Filter (Cutoff = 20)

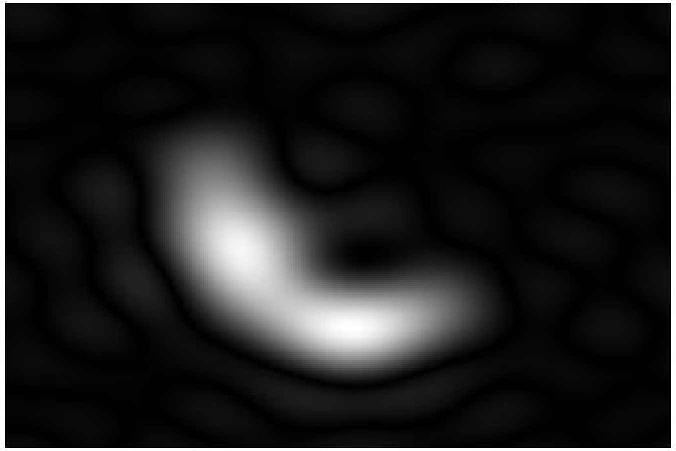
Filtered Image



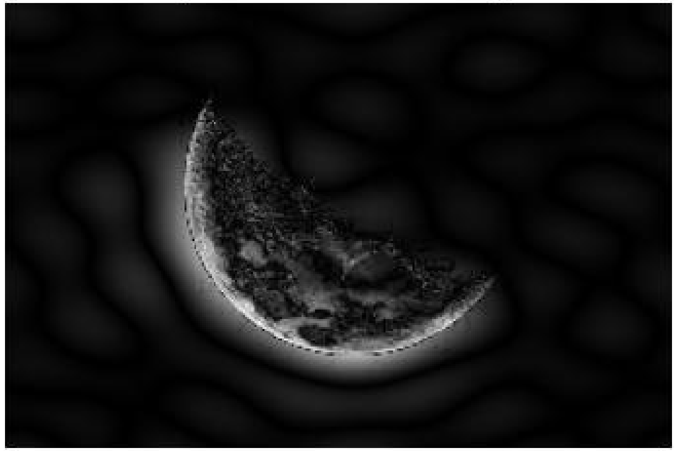
for cutoff frequency = 5

Original Image

Low-Pass Filtered Image



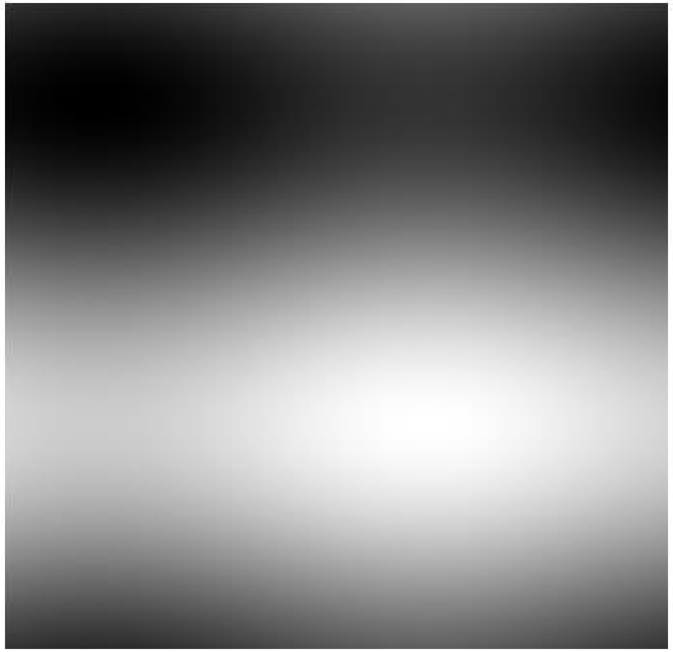
High-Pass Filtered Image



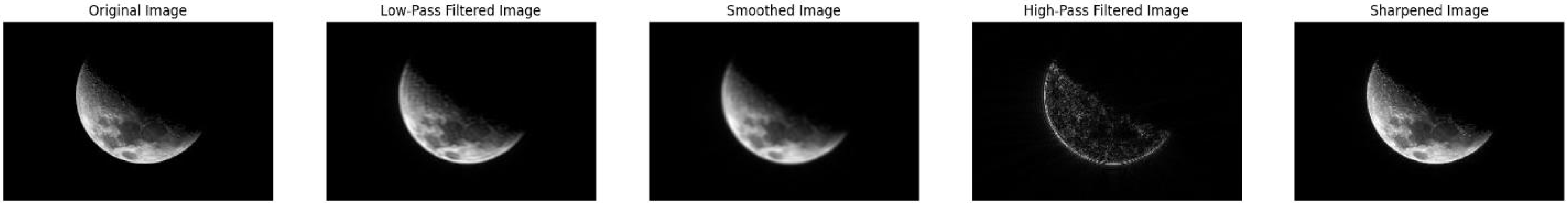
for cutoff frequency = 1

Original Image

Low-Pass Filtered Image



High-Pass Filtered Image


Original Image
Low-Pass Filtered Image
Smoothed Image
High-Pass Filtered Image
Sharpened Image

Some other techniques for Image enhancement:

- Contrast-limited adaptive histogram equalization (CLAHE) [Local Histogram Localization]
- Shearlet Transform - detect high freq. / edge,

  wavelet transform +CNN for superresolution

Code:

https://github.com/Ashwani132003/Image_Processing_Python

References

- https://medium.com/@gokcenazakyol/what-is-image-enhancement-image-processing-3-32a813087e0a
- https://medium.com/@raufjamadar113/image-enhancement-techniques-1cb79f5e5556
- https://www.slideshare.net/slideshow/image-enhancement-117439830/117439830
- https://universe.bits-pilani.ac.in/uploads/Image%20Enhancement%20in%20Spatial%20Domain.pdf
- https://samirkhanal35.medium.com/contrast-stretching-f25e7c4e8e33
- https://www.tutorialspoint.com/dip/histogram_equalization.htm