Indian Institute of Technology Bombay

EE338 : Digital Signal Processing

Application Assignment

Image enrichment for night vision

Group 2
Mayur Ware, 19D070070, Department of Electrical Engineering
Rushikesh Metkar, 19D070034, Department of Electrical Engineering
Siddhant Shingade, 19D070057, Department of Electrical Engineering

Final Report
April 17, 2022

Index

1. Abstract

Vision is one of our most important senses and images play a vital role in human perception. However, images captured in night environments have low brightness, low contrast and high noise. So, it is necessary to improve the quality of such images to extract the hidden data. We employ techniques involving Digital Signal Processing for this purpose. In Digital Image Processing, we refer to an image as the composition of a finite number of elements, each of which has a particular location and value. This fundamental discretized unit is called Pixel. Each pixel has an intensity value ranging from 0 to 255.

Enhancement of the images captured at night involves two important techniques, contrast enhancement and denoising. Intensity histogram of such images is inclined towards lower intensity levels. We manipulate this histogram data by applying processing techniques in the frequency domain on the image data. For this, we use transformations such as Discrete Fourier Transform (DFT), Log Transform, Gamma Transform. Also, convolution matrices called kernels are used to apply various effects on a digital image.

The *Atmanirbhar Bharat* mission by the Government of India is intended to make India self-reliant in all senses. With this project, we aim to build an algorithm that will process the RGB images captured at night to give the clean and desired output using the modern and efficient Digital Signal Processing techniques related to this topic to contribute to this initiative.

2. Introduction

Image enhancement refers to the process of improving the quality of images so that they become more pleasing to the human eye and can be used for further image analysis. This is done with a little attempt at the estimation of the actual image degradation process. Particularly, low light image enhancement is gaining popularity and importance due to a variety of applications including medical imaging, video surveillance and night vision. The image quality is limited in low light conditions even after significant advancements in digital cameras in terms of resolution and sensitivity.

Improving the contrast can improve the quality of images, but increasing the illumination increases the noise whereas attempting to reduce the noise decreases the contrast and brightness. Night images also impose the challenge of treatment of Poisson noise while increasing the contrast. Several attempts have been made to increase the quality of images with minimal amplification of noise. The technique of decomposing the image into illuminance and reflectance components to reproduce a natural image introduced ghosting artifacts. Implementing a de haze algorithm to the inverted input image in order to amplify the intensity resulted in the introduction of halo artifacts. Also, denoising techniques attempt to remove the noise while preserving useful information. However, too much denoising causes over-smoothening and blurry effects. Some denoising algorithms may also produce edge sharpening. Hence, special care must be taken in case of low-light images.

In this assignment, we will attempt to enhance the quality of low-light images for extracting useful information alongside taking care of minimum amplification of noise.

2.1 Digital Image Processing

In Digital Image Processing, we define an image as a two-dimensional function of pixel location and intensity. Pixel is the smallest independent unit of a Digital Image having an intensity value and a unique location. Here, we discretize the values of intensity to apply the digital techniques instead of the continuous values in conventional or analog image processing.

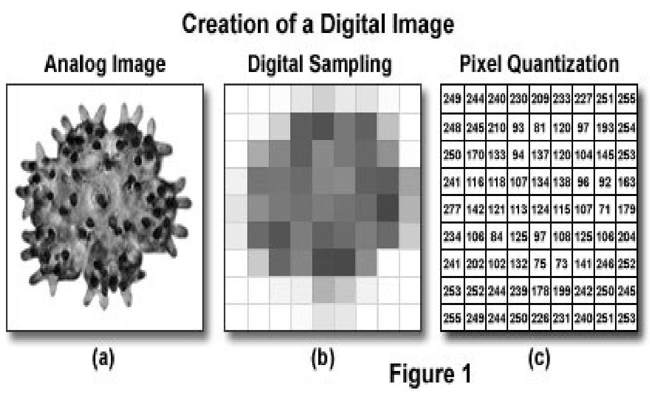


Fig. 1 : Creation of a Digital Image
Source : Hamamatsu Learning Center - Florida State University

Due to this only difference, analog and digital images can be converted into each other using simple transformation methods. As shown in Fig. 1, an analog image is first converted into a digital image by sampling it, and then, the digital image is pixelated generally using a 256-bit quantization process.
Digital Image Processing has a wide as well as varied domain of applications as it can process the whole Electro-magnetic (EM) emission spectrum, unlike the human eye whose capabilities are bounded within the visible region in the EM spectrum. We construct an RGB image by associating dynamic Red, Blue and Green color intensities to an individual pixel.

2.2 Steps involved in Image enrichment for night vision

In contrast to normal images, low light images have a low Signal-to-Noise Ratio (SNR) and hence high noise. Apart from this, the also have low illumination and low contrast. They might also suffer from the problem of non-uniformity in scene luminescence. Intensity histogram of low brightness images is very much concentrated towards 0 intensity level. Also the average luminescence is very as compared to the normal images in case of low light images.
We will apply noise filtering algorithm to filter out the noise as much as possible. To increase the contrast of the image, we will do plot the histogram versus intensity graph and then apply histogram equalization technique that is mentioned in the later sections. Low pass spatial filtering/high pass spatial filtering will be applied to make the image smooth/sharp as per the requirement.

3. Intensity Transformations and Spatial Filtering

The image plane itself is referred to as the spatial domain, and image processing performed in this area is based on direct pixel manipulation. This is in contrast to image processing in a transform domain which we will discuss in later sections. For image transformation, you must first transform it into the transform domain, process it there, and then acquire the inverse transform to return the results to the spatial domain.
Intensity transformations and spatial filtering are the two main types of spatial processing. Intensity transformations work on single pixels of an image. Spatial filtering operates on the neighborhood of each pixel in the image. Examples of intensity transformation include contrast manipulation and image thresholding and spatial filtering consist of image smoothing and sharpening.

3.1 Intensity Transformation Functions

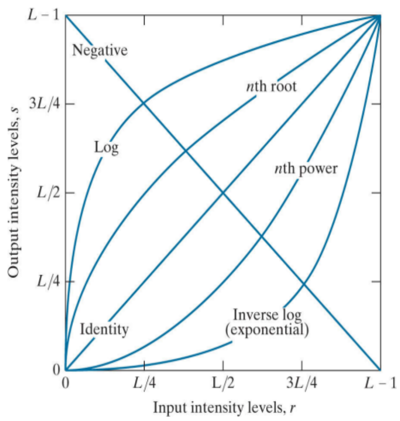


Fig. 2 : Intensity Transformation Functions
Source : GeeksforGeeks.com

Let us denote the values of pixels as r before processing and s after processing. A transformation T connects these values by mapping a pixel value$r$ to a pixel value s. Because we're dealing with digital data, intensity transformation function values are often kept in a table, and the mappings from r to s are accomplished via table lookups. A lookup table storing the values of T for an 8-bit picture will have 256 (L) entries. Intensity transformations that we are going to use in our project are shown in Fig. 2.

Fig. 3 : Original Reference Image

We will explain all the transformations using a reference image shown in Fig. 3

3.1.1 Linear Transformation

Linear Transformation is the most basic intensity transform in Digital Image Processing. It includes Identity Transformation and Negative Transformation. The relationship between *r* and *s* in these transformations is similar, which is a straight line.

a) Identity Transformation

$$s = r$$

Identity transition is indicated by a straight line. In this transition, each value of the input image is directly mapped to each other value of the output image. That results in the same input image and output image. And hence is called identity transformation.

Fig. 4 : Identity Transformation

b) Negative Transformation

$$s = (L - 1) - r$$

We have L = 256, which implies,

$$s = 255 - r$$

The second linear transformation is negative transformation, which is the invert of identity transformation. In negative transformation, each value of the input image is subtracted from the L-1 and mapped onto the output image.

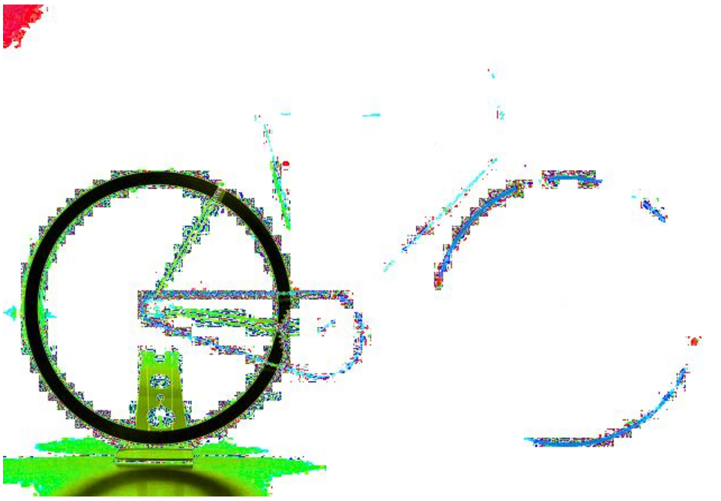


Fig. 5 : Negative Transformation

3.1.2 Logarithmic Transformation

The equation for Logarithmic Transformation is given by

$$s = c(1 + r)$$

[where c is a constant which is chosen appropriately]
This transformation maps a narrow range of low intensity values in the input into a wider range of output levels. Conversely, higher values of input levels are mapped to a narrower range in the output. The log function has the important characteristic that it compresses the dynamic range of pixel values which can be used to expand the values of dark pixels in an image.

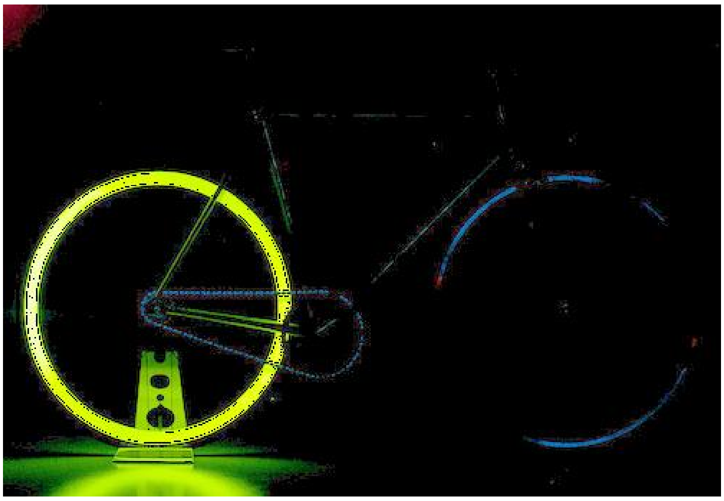


Fig. 6 : Logarithmic Transformation

3.1.3 Power Law Transformation

Power Law Transformation has the form

$$s = cr^{\gamma}$$

[where c and γ are positive constans]
Power law curves are pretty much similar to the logarithmic transformation. But, the main difference between them is γ. Variation in the value of γ varies the enhancement of the images. This type of transformation is used for enhancing images for different types of display devices. The γ of different display devices is different.

Fig. 7 : Power Law Transformation

3.1.4 Use of Intensity Transformations in low light image processing

For the night images, as they are dark, the intensity values they contain are very low. Contrast stretching increases the range of intensity levels in an image such that it covers the recording medium's or display device's optimal complete intensity range. Contrast enhancement can be achieved using power-law intensity transformations with a fractional exponent. Thus, we can get high contrast images using intensity transformations even in low light.

3.2 Histogram Processing

Let r_k denote the intensities of a digital image, f(x, y). The unnormalized histogram of f is defined as $h(r_k) = n_k$ for k = 0,1,2,3,. . . .,L-1, where n_k is the number of pixels in f with intensity r_K, and the subdivisions of the intensity scale are called histogram bins. To normalize the histogram divide n_k by MN where M and N are the number of image rows and columns, respectively.

3.2.1 Histogram Equalization

For the intensity transformation s = T(r) 0<=r<=L-1, we assume that, T(r) is a monotonic increasing function in the interval 0<=r<=L-1 and 0<=T(r)<=L-1 for 0<=r<=L-1.

Let p_r(r) and p_s(s) denote the PDFs of intensity values r and s in two different images. The subscripts on p indicate that p_r and p_s are different functions. A fundamental result from probability theory is that if pr(r) and T(r) are known, and T(r) is continuous and differentiable over the range of values of interest, then the PDF of the transformed (mapped) variable s can be obtained as

$$p_s(s) = p_r(r)|\frac{dr}{ds}|$$

Consider the transformation function,

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

We can clearly see that it is monotonically increasing and satisfies 0<=T(r)<=L-1 for 0<=r<=L-1. For discrete values, we work with probabilities and summations instead of probability density functions and integrals (but the requirement of monotonicity stated earlier still applies).

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(j) \quad k = 0, 1, 2, ..., L-1$$

This is called a histogram equalization and has the general tendency to spread the histogram of the input image so that the intensity levels of the equalized image span a wider range of the intensity scale. The net result is contrast enhancement.

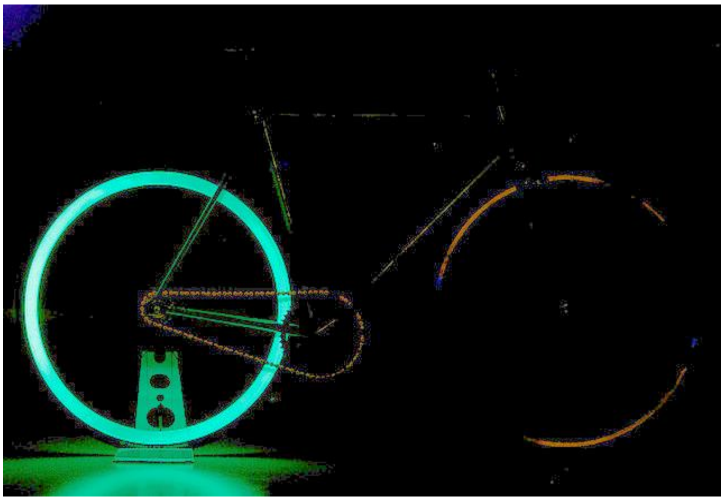


Fig. 8 : Histogram Equalization

3.2.2 Histogram Matching (Specification)

It's quite important to be able to select the shape of the histogram that the processed image should have on occasion. Histogram matching, also known as histogram specification, is a technique for creating images with a given histogram.

Consider the continuous intensities *r* and *z*, which we represent as random variables with PDFs p_r(r) and p_z(z) respectively, as we did before. The intensity levels of the input and output images are denoted by *r* and *z*, respectively. p_r(r) can be guessed from the input image and p_z(z) is given to us as the specified PDF of the output image.

Consider the following set of equations,

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

$$G(z) = (L-1)\int_0^z p_z(v)dv = s$$

$$z = G^{-1}(s) = G^{-1}[T(r)]$$

The above equations determine Histogram matching.

Now, in the discrete domain,

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) \quad k = 0,1,2,...,L1$$

$$G(z_q) = (L-1)\sum_{i=0}^{q} p_z(z_i)$$

$$z_q = G^{-1}(s_k)$$

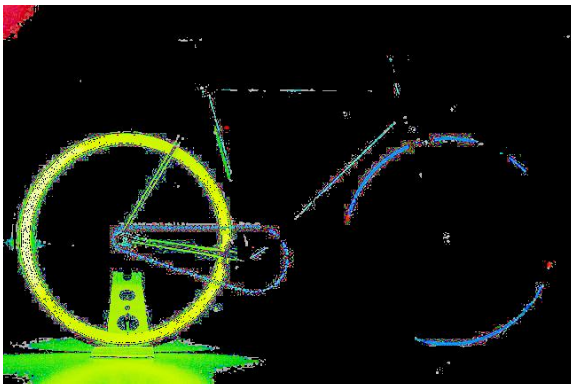


Fig. 9 : Histogram Matching

.

3.3 Fundamentals of Spatial Filtering

Spatial filtering alters an image by replacing each pixel's value with a function of the pixel's and its neighbors' values. The filter is termed a linear spatial filter if the operation performed on the image pixels is linear. The filter is otherwise a nonlinear spatial filter. A low pass filter, for example, is a filter that passes low frequencies.Its overall effect in this regards is to blur an image and smooth it out.
A sum-of-products operation is performed by a linear spatial filter between an image f and a filter kernel, w. The kernel is an array whose size determines the operation's locality and whose coefficients determine the filter's type. Spatial correlation is given as

$$g(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b}w(s,t)f(x+s,y+t)$$

Where size of the image is M x N with a kernel of size m x n. x and y are varied so that the center (origin) of the kernel visits every pixel in f (image) once.
The Fourier transform is the link between spatial and frequency-domain processing. To travel from the spatial to the frequency domain, we use the Fourier transform; to return to the spatial domain, we use the inverse Fourier transform. Convolution, which is the basis for filtering in the spatial domain, is equivalent to multiplication in the frequency domain, and vice versa.

3.3.1 Low Pass Spatial Filters

Smoothing (also known as averaging) spatial filters are used to smooth out abrupt intensity shifts. Because random noise often has rapid intensity changes, noise reduction is a natural use of smoothing. Smoothing is a technique for reducing irrelevant detail in an image, where "irrelevant" refers to pixel areas that are tiny in comparison to the filter kernel's size. Convolving a smoothing kernel with an image blurs the image. The degree of blurring is governed by the kernel's size and coefficient values. Lowpass filters are basic in the sense that they may be used to derive other significant filters such as sharpening (highpass), bandpass, and bandreject filters.

3.3.2 High Pass Spatial Filters

Sharpening highlights transitions in intensity. We showed how pixel averaging (smoothing) in a neighborhood may be used to achieve image blurring in the spatial domain. Because averaging and integration are analogous, it follows that sharpening may be achieved by spatial differentiation. The strength of the response of a derivative operator is proportional to the magnitude of the intensity discontinuity at the point at which the operator is applied. Thus, image differentiation enhances edges and other discontinuities (such as noise) and de-emphasizes areas with slowly varying intensities.

4. Frequency Domain Filtering

After defining and transforming the image in the spatial domain, we move towards the frequency domain filtering of the image. Image processing methods that are very complicated to implement in the spatial domain can be implemented in simpler ways using frequency-domain methods. Signals are easy to handle in the frequency domain using the popular Fourier transforms. Moreover, these methods can be reversed easily to obtain the original signal.
In this section, all the frequency domain transformations which are used in our night image enhancement algorithm are explained. As we are processing discrete signals, we will deal with the discrete versions of all transformations.

4.1 Discrete Fourier transform (DFT)

The Discrete Fourier Transform (DFT) is a similar version of the continuous time fourier transform applied on finite sequence data of a continuous signal (N instants separated by sample times T).
Let f(t) be the continuous signal which is the source of the data. Let N samples be denoted by f[0], f[1], f[2],...f[k],...,f[N-1].
Then, we define the DFT as,

$$F(j\omega)=\sum_{k=0}^{N-1}f[k]e^{-j\omega kT}$$

.

This can be normalized upto frequencies 0 to 2π

$$F[n] = \sum_{k=0}^{N-1} f[k] e^{-j\frac{2\pi}{N}nk}$$

F[n] coefficients are complex.

4.2 Inverse Discrete Fourier Transform (IDFT)

The Inverse Discrete Fourier Transform (IDFT) is the discrete version of continuous inverse fourier transform. It is applied to the fourier transformed signal to recover the original signal.
For a normalized DFT F[n], IDFT is defined as,

$$f[k] = \frac{1}{N} \sum_{n=0}^{N-1} F[n] e^{+j\frac{2\pi}{N}nk}$$

One can notice that, the IDFT is 1/N times the complex conjugate of the DFT.

4.3 Fast Fourier Transform (FFT)

A Fast Fourier Transform (FFT) is an algorithm to compute the DFT of a sequence. This operation is used in many fields and computing from the direct definition happens out to be slow for practical usage. An FFT method rapidly computes such transformation by factorizing the DFT matrix into a product of sparse (mostly zero) factors. The output of the FFT and DFT algorithms are the same, but the FFT has a much faster execution time than the DFT (proportional to $N\log_2(N)$ versus N^2 operations).

5. Night image enhancement algorithm

5.1 Adaptive Histogram Equalisation

The traditional method of histogram equalisation though simple doesn't take into account the local image information. This method is not suitable when the image is not homogeneous. Hence advances in the traditional method are done so as to obtain more accurate and informative results.
One such advancement is called the Adaptive Histogram Equalisation(AHE). It is also used to enhance the contrast of the image. It differs from the traditional histogram equalisation method as it computes several histograms each for a different section of an image. It divides the image into unique blocks locally and then carries out the histogram equalisation for each of the blocks. Hence this method becomes more suitable for improving the local contrast and bringing out the details of a low light image. The transformation function used is directly proportional and depends on the CDF of pixel values in the region nearby. It is derived the same as the traditional histogram equalisation.
This method, though advantageous, needs to be executed efficiently. The time taken or the time complexity of execution is proportional to the size of the image and hence for large images it would be computationally expensive. Hence there is a need to speed up the computations and reduce the time complexity.

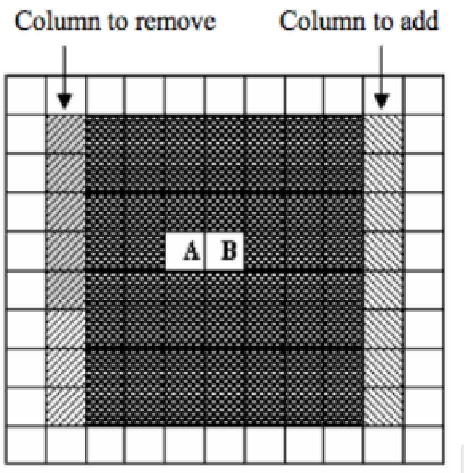


Fig. 10 : Faster Adaptive Histogram Equalisation
Source : [3]

This is done by allowing only the first block of the image to process every pixel in the block. Therefore when the centre of the window function is displaced from position A to position B, we remove the pixels on the left column and make an addition of pixels on the right column.

5.2 Retinex Algorithm

The Retinex theory takes into account the functioning of the human retinal cortex. It has it's basis on the perception of the colour by the eye and how it models the colour invariance. The human vision only retains the important information regarding the characteristics of the object such as the reflection coefficient etc. and it discards the factors it is unsure of like the intensity and the unevenness of the light source. This is therefore useful in determining the reflective characteristics of the object.
We can equate the image as

$$I(x, y) = R(x, y)\ L(x, y)$$

where L(x, y) depicts the illumination factor which depends on the environmental factors. This gives information regarding the dynamic range of the image. R(x, y) depicts the reflection factor which depicts the reflection characteristics of the surface. This gives information regarding the dynamic range of the image The product of these two gives the Image I(x, y).

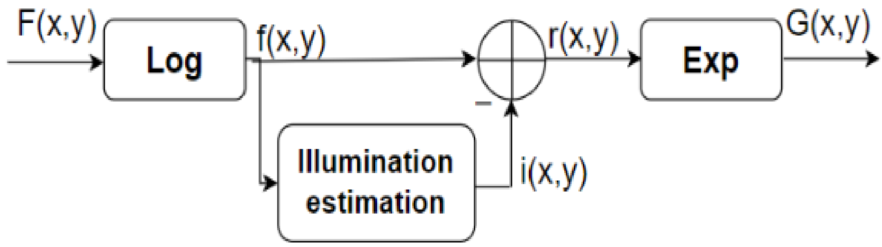


Fig. 11 : Block Diagram of RETINEX Algorithm

If L(x, y) can be estimated from the image then R(x, y) can be separated from the total amount of light, thus enhancing the image by reducing the impact of L(x, y). The algorithm is depicted in the above figure. This algorithm has the characteristics of colour constancy and has a great dynamic range compression. It also has a good sharpening capability.

5.2.1 Single Scale Retinex (SSR)

The formula for this algorithm is as follows :

$$logR_i(x, j) = logI_i(x, y)log[G(x, y) \times I_i(x, y)]$$

Here, (x,y) denote the x and y coordinates of the position of pixel in the image. The i denotes the colour channel of the reflected image i.e. R(x, y) and of the input image i.e. I(x, y). * denotes the convolution operation between a colour channel and the gaussian channel function G(x, y).
G(x, y) can be arrived at by using the relation :

$$G(x, y) = Ke^{(-\frac{x^2+y^2}{\sigma^2})}$$

Where σ is the scaling parameter. More the value of σ, lesser would be the dynamic compression and hence lesser would be the clarity in the local values. The normalisation factor K ensures that the double integral of the gaussian channel function over the range of x and y is always 1.
The mathematical relation is

$$R_i^{'}(x, y) = 255 \times \frac{(R_i(x, y) - R_{min})}{(R_{max} - R_{min})}$$

Is used to map the output image in the interval [0,255]. By doing this automatic gain compensation is achieved and the functions of the human visual system are more accurately represented. Rmin and Rmax denote the minimum and maximum gray levels in the image and the left hand side denotes the output obtained by gray stretching the ith channel.

.

5.3 Homomorphic Filtering (HF)

Image enhancement methods have been expanded from the spatial domain to the frequency domain as a result of the development of multiscale image analysis technologies. Image enhancement techniques that use the frequency domain convert an image into the frequency domain for filtering using Fourier analysis, and then inversely transform the finished image back into the spatial domain.
The properties of the illumination-reflection model are used in Homomorphic Filtering based enhancement approaches to turn the illumination and reflection components into a sum in the logarithmic domain rather than a product. In the Fourier transform domain, a high-pass filer is employed to boost the high-frequency reflection component while suppressing the low-frequency illumination component.
The steps involved in HF process are as follows :
1) The illumination component is multiplied by the reflection component in the illumination-reflection model, which cannot be converted into the frequency domain. As a result, the logarithmic transformation needs be used to convert these multiplicative components into additive components before they may be treated independently. Taking log on both sides of equation (x.xx) we get,

$$lnI(x,y) = lnL(x,y) + lnR(x,y)$$

2) The Fourier transform is used to convert the image from the spatial domain to the frequency domain.

$$F[lnI(x,y)] = F[lnL(x,y) + lnR(x,y)]$$

It can also be written as,

$$I(u,v) = L(u,v) + R(u,v)$$

where I(u, v), L(u, v) and R(u, v) are the Fourier transforms of I(x, y), L(x, y) and R(x, y), respectively.
3) A suitable high-pass filter is chosen for contrast enhancement, and the transfer function H(u, v) enhances the R(u, v) component in the frequency domain.

$$S(u,v) = H(u,v)I(u,v) = H(u,v)L(u,v) + H(u,v)R(u,v)$$

4) The image is transformed from the frequency domain to the spatial domain using the inverse Fourier transform.

$$s(u,v) = F^{-1}(H(u,v)L(u,v)) + F^{-1}(H(u,v)R(u,v)) = h_L(x,y) + h_R(u,v)$$

where s(u, v) denotes the inverse Fourier transform corresponding to S(u, v). As a result, the improved image is the result of superimposing the illumination and reflection components.
5) The inverse logarithmic transform G(x, y) = exp[s(x, y)] is applied to obtain the final corrected image.

$$G(x,y) = exp|h_L(x,y)| \times exp|h_R(x,y)|$$

As a result, the HF technique's core is to build an appropriate filter H(u, v) based on the image qualities specified by the lighting and reflection components, as well as a frequency filter and a grey transformation, to compress the dynamic range and enhance the contrast.
A homomorphic filter has the following general form:

$$H(u,v) = (\gamma_H - \gamma_L)H_{hp}(u,v) + \gamma_L$$

[where $\gamma_L < 1$ and $\gamma_H < 1$ and H_{hp} is a high-pass filter]
If a Gaussian HPF is used, then

$$H_{hp} = 1 - exp[-c \times (D^2(u,v)/D_0^2)]$$

[where c is a constant that controls the form of the filter]

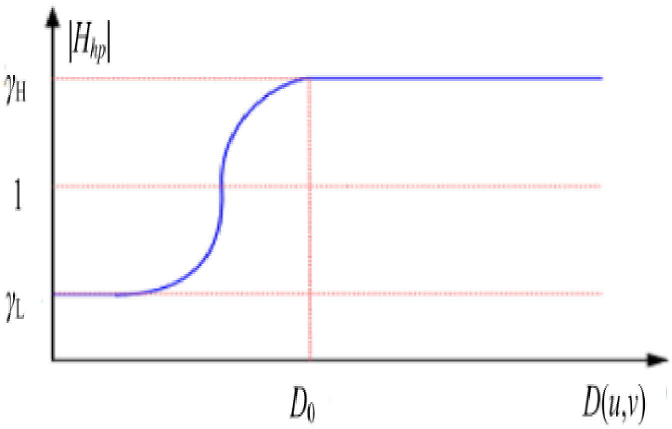


Fig. 12 : Amplitude response curve of a homomorphic filter
Source : [4]

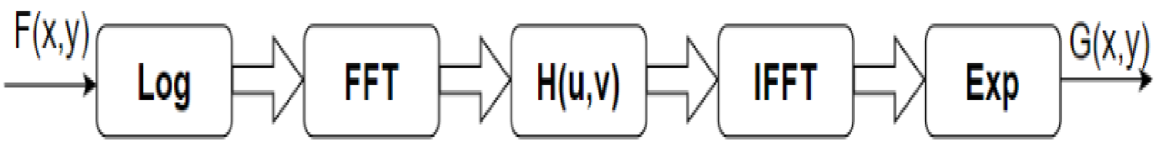


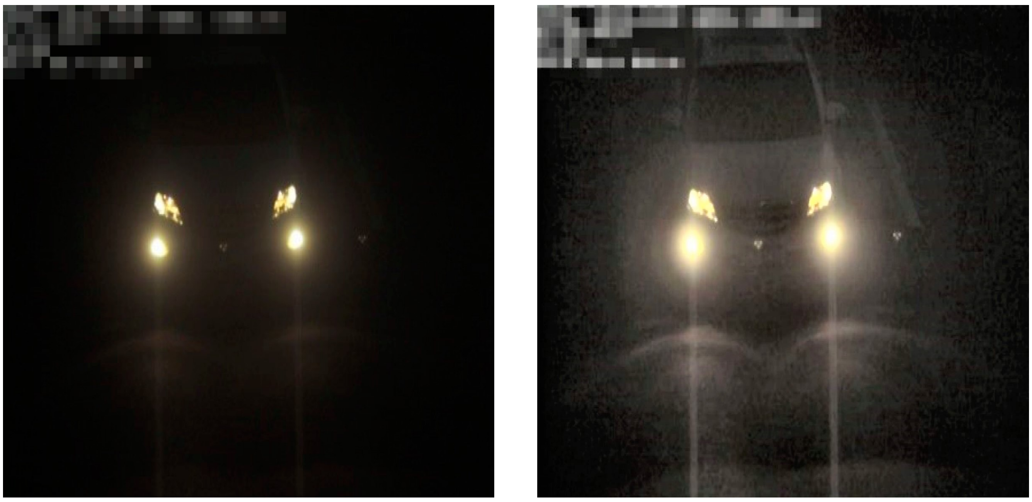
Fig. 13 : Flowchart of HF algorithm

After the HF procedure, the image brightness is enhanced. Light-induced uneven patches can be removed using HF algorithms, while the image's contour information is preserved.

6. Results

6.1 Adaptive Histogram Equalization

Performing Adaptive Histogram Equalization (AHE) on the original image, we get the transformed image as shown below



(a) (b)

Fig. 14 : (a) Original image, (b) AHE transformed image

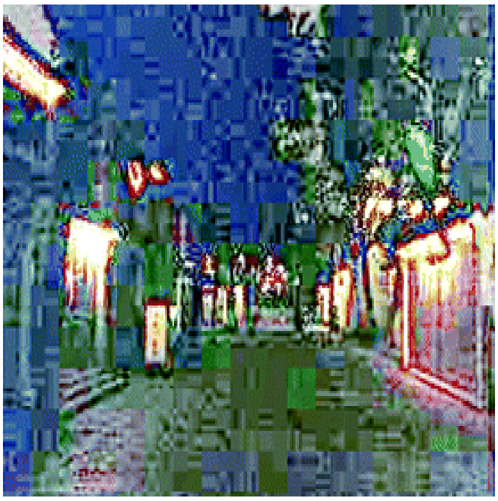
6.2 Single Scale Retinex

Varying the scale factor σ in the Gaussian filter

$$G(x,y) = Ke^{(-\frac{x^2+y^2}{\sigma^2})}$$

we get the results as

(a)



(b)

(c)

Fig. 15 : (a) Original image, (b) $\sigma = 15$, (c) $\sigma = 120$

6.3 Homomorphic Filtering

After applying a homomorphic filter,

$$H(u, v) = (\gamma_H - \gamma_L)H_{hp}(u, v) + \gamma_L$$

for different values of γ_H and γ_L, we get the results as follows

(a)

(b)

(c) (d)

Fig. 16 : (a) Original image, (b) $\gamma_H = 1.3, \gamma_L = 0.4$, (c) $\gamma_H = 1.6, \gamma_L = 0.3$, (d) $\gamma_H = 2.0, \gamma_L = 0.2$

7. User Interface

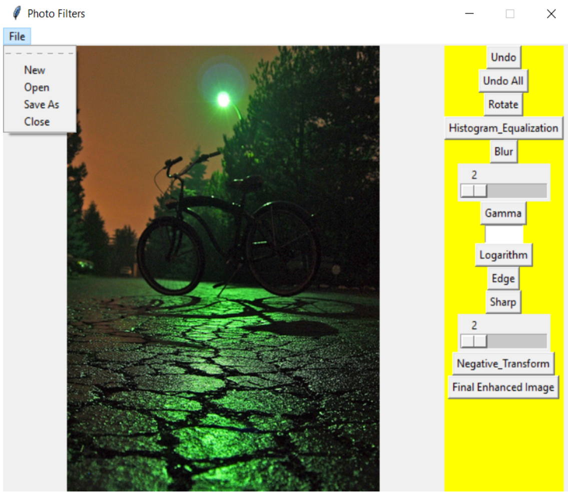


Fig. 17 : User Interface

Fig. 17 shows the User Interface (UI) generated by us in which one can directly apply all the transforms mentioned above including the night image enhancement algorithm as well as some extra commands such as Blurring, Sharpening, Undo, Undo All and Rotate.

8. Appendix

The code files used by us to generate the results are there in the GitHub repository link given below : `https://github.com/MayurWare/EE338_Application_Assignment`

9. References

1) SA Priyanka, YK Wang, "Low-Light Image Enhancement by Principal Component Analysis", in 2018 IEEE Conf., pp. 3082-3092, Vol. 7, 2019

2) HJ Tung, YK Wang, "CONTRAST ENHANCEMENT OF NIGHT IMAGES", in 2016 Intl. Conf. on ML and Cybernetics, pp. 380-385, July 2016

3) `https://github.com/arunnthevapalan/day-night-classifier/tree/600a4c7b8a9a166f049\`
`\f7ed39d38f74283080f1f/images/test/night`

4) W Wang, Xi Wu, Z Gao, "An Experiment-Based Review of Low-Light Image Enhancement Methods", pp. 87884-87917, vol. 8, 2020

5) Digital Image Processing | Fourth Edition | Rafael C. Gonzalez, Richard E. Woods