DIGITAL IMAGE PROCESSING

UNIT – III

3.1 Image Enhancement in Frequency Domain:

Frequency refers to the rate of repetition of some periodic events. In image processing, spatial frequency refers to the variation of image brightness with its position in space. Frequency domain is just the space defined by the values of the Fourier transform and its frequency variable. Image enhancement in frequency domain refers that the multiplication of each element of the Fourier coefficient of an image F(u,v) by a suitably chosen weighing function H(u,v) then we can accentuate certain frequency components and attenuate others.

$$G(u,v) = F(u,v) \cdot H(u,v) \qquad \ldots\ldots\ldots\ldots\ldots\ldots..(1)$$

The enhancement or suppression of frequency components is termed as ***Fourier filtering (or) Frequency domain filtering.***

The above figure shows the basic steps of filtering in frequency domain.

Where, f(x,y) = Input image

g(x,y) = Enhanced image

Frequency Domain Filtering:

Frequency domain filtering is simply compute the Fourier transform of the image to be enhanced, multiply the result by a filter and take the inverse transform to produce the enhanced image. The filters are classified based into three categories.

- **Notch filter** – *A filter that attenuate a selected frequency (and some of its neighbors) and leave other frequencies of the Fourier transform relatively unchanged*
- **Low pass filter** – *A filter that allows only lower frequency components that attenuates the higher frequency components. (Regions of relatively uniform gray values in an image contribute to low-frequency content)*
- **High pass filter** – *A filter that allows only higher frequency components that attenuates the lower frequency components. (Regions of edges and sharp*

transitions in gray values in an image contribute significantly to high frequency content)

3.1.1 Low pass filer / Smoothing filter in frequency domain:

Smoothing in frequency domain is the process of attenuating a specified range of high-frequency components in the transform of a given image. It's also known as **blurring.**

Need for Smoothing: Sharp transitions in the gray levels of an image such as edges and noise are present as high-frequency components in its Fourier transform. To remove these unwanted contents, Smoothening is required.

3.1.1.1 Ideal low pass filter:

Ideal lowpass filter is the simplest lowpass filter. It "cuts off" all the high-frequency component of the Fourier transform which are located at a distance greater than a specified distance D_0 from the origin of the centered transform.

The transfer function of 2D-ILPF is given by,

$$H(u,v) = \begin{cases} 1 \; if \; D(u,v) \leq D_0 \\ 0 \; if \; D(u,v) > D_0 \end{cases}$$

Where, D_0 – Cut-off frequency

D(u, v) – Distance between the point (u, v) and the origin of the frequency rectangle.

Ringing Effect:

Ringing effect so known as Gibbs phenomenon in mathematical methods of image processing is the annoying effect in images and video appeared as rippling artifact near sharp edges. ***Ringing effect is caused by distortion or loss of high frequency information in image.***

3.1.1.2 Butterworth low pass filter:

In Butterworth lowpass filter, there is no clear cutoff frequency which decides the amount of frequencies to be passed and the amount of frequencies to be filtered. When amount of high-frequency content removed decreases, the image becomes finer in texture.

The transfer function of 2D-BLPF is given by,

$$H(u,v) = \frac{1}{1 + \left[\frac{D(u,v)}{D_0}\right]^{2n}}$$

Where, n – Filter order

Ringing Effect: In BLPF ringing effect is directly proportional to the order of the filter (n). If order "n" increases the ringing also increases. Therefore, BLPF of order "n = 2" is preferred because, it provides ***Effective lowpass filtering and Acceptable ringing characteristics.***

3.1.1.3 Gaussian low pass filter:

The Gaussian smoothing operator is a 2-D convolution operator that is used to `blur' images and remove detail and noise. In this sense it is similar to the mean filter, but it uses a different kernel that represents the shape of a Gaussian (`bell-shaped') hump.

The transfer function of 2D-GLPF is given by,

$$H(u,v) = e^{-D^2(u,v)/2\sigma^2}$$

Where, σ – Measure of the gaussian curve spread.

If $\sigma = D_0$ then, $$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

The obtained filter will have ***no ringing effects.***

Applications of LPF:

- Character Recognition
- Printing and Publishing Industry and
- Processing satellite and Aerial Images

3.1.2 High pass filer / Sharpening filter in frequency domain:

Image sharpening refers to any enhancement technique that highlights edges and fine details in an image (or) Image Sharpening is the process of attenuating low frequency components without disturbing the information in the Fourier transform. Image sharpening is widely used in printing and photographic industries for increasing the local contrast and sharpening the images

Need for Sharpening: Human perception is highly sensitive to edges and fine details of an image, and since they are composed primarily by high frequency components, the visual quality of an image can be enormously degraded if the high frequencies are attenuated or completely removed. In contrast, enhancing the high-frequency components of an image leads to an improvement in the visual quality.

3.1.2.1 Ideal high pass filter:

Ideal high pass filter is the simplest high pass filter. It "cuts off" all the low-frequency component of the Fourier transform which are located at a distance lesser than a specified distance D_0 from the origin of the centered transform.

The transfer function of 2D-IHPF is given by,

$$H(u,v) = \begin{cases} 0 \ if \ D(u,v) \leq D_0 \\ 1 \ \ if \ D(u,v) > D_0 \end{cases}$$

Where, D_0 – Cut-off frequency

D(u, v) – Distance between the point (u, v) and the origin of the frequency rectangle.

3.1.2.2 <u>Butterworth high pass filter:</u>

The behavior of Butterworth high pass filters is smoother than ideal highpass filter. This means that the images produced by BHPF are better than IHPF produced images.

The transfer function of 2D-BLPF is given by,

$$H(u,v) = \frac{1}{1 + \left[\frac{D_0}{D(u,v)}\right]^{2n}}$$

Where, n – Filter order

3.1.2.3 <u>Gaussian high pass filter:</u>

The results produced by a gaussian high pass filter are smoother than the results produced by IHPF and BHPF.

The transfer function of GHPF with cutoff frequency at distance D_0 from the origin is expressed as,

$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

3.2 COLOR IMAGE PROCESSING:

Color image processing is divided into two broad categories namely,

1. **Pseudo Color Processing:** In Pseudo color processing, a color is assigned to a particular monochrome intensity or range of intensities.
2. **Full Color Processing:** In Full color processing, the images are acquired using full-color sensors such as a TV camera or a Color Scanner are processed.

The various color models that exist are:

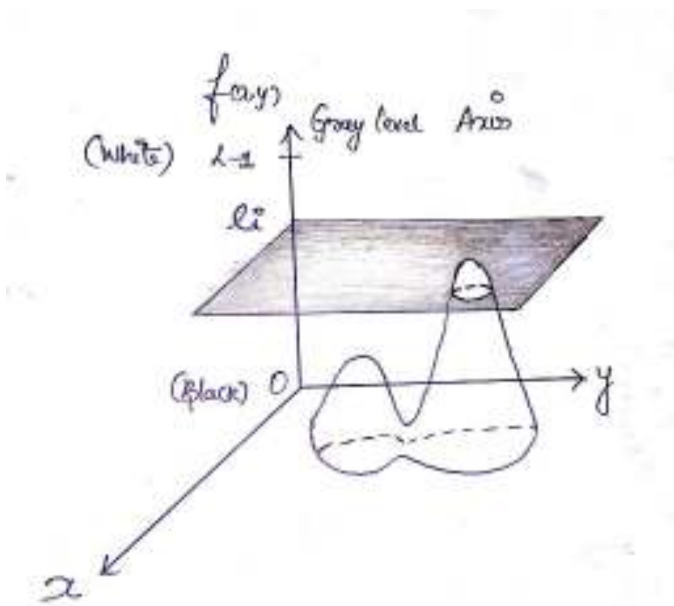
- Red Blue Green (RGB) Model (Primary color Model)
- Cyan Magenta Yellow (CMY) Model (Secondary color Model)
- Hue Saturation Intensity (HSI) Model

3.2.1 Pseudo Color Image Processing:

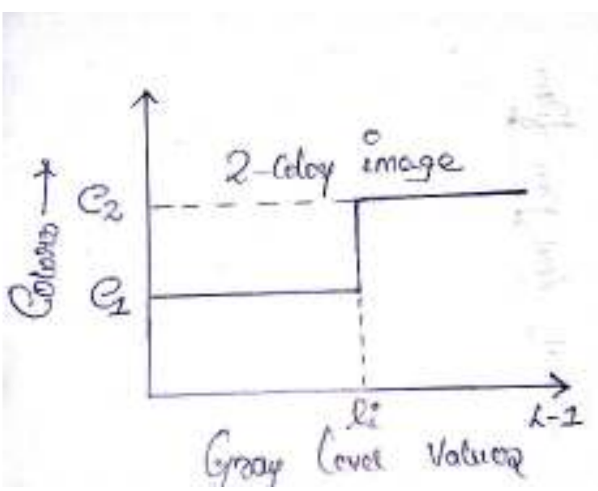
- Pseudo or False image processing is the term used to differentiate the process of assigning colors to monochrome images from the processes associate with true color images.
- Changing a black & white image or grayscale image into color image helps human to distinguish / read the images properly.

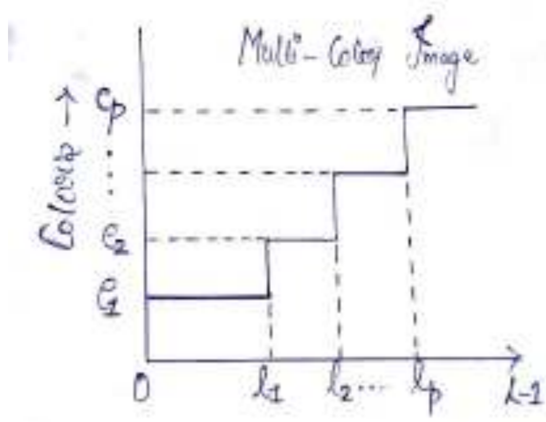
3.2.1.1 Intensity Slicing:

- It's one of the simplest methods of pseudo-color processing, here the image is represented in a 3-D manner or function.
- The X and Y axes have the pixels of the image f(x,y) and the gray level axis consist of the gray level values.
- The 3-D graph shows the various Gray level values that the image pixels have.
- ***If the graph is sliced by using a particular Gray level value (l_i), then we can assign two different colors to the pixels having Gray level values smaller than l_i and Gray level values greater than l_i.***



- This this technique converts a monochrome image into a bicolor image.



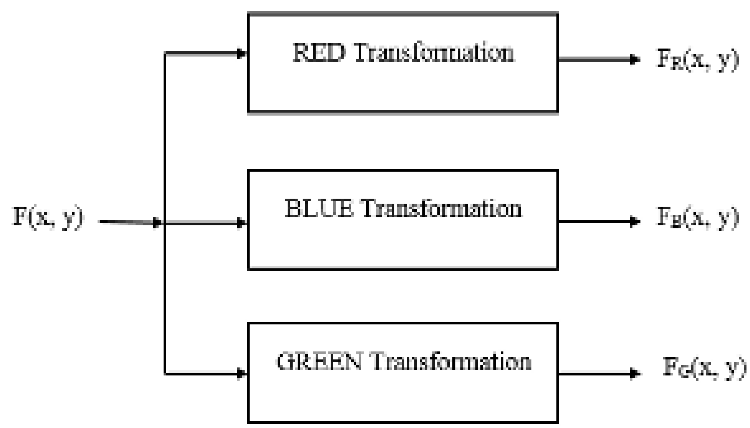


- If more than on Gray level value is used to slice an image graph. Then the slicing plane are given by $l_1, l_2, \ldots\ldots l_p$.

- Then p slicing planes will slice the graph into p+1 intervals. For p+1 intervals the pixels can be assigned C_p+1 colors, changing the input image into a multi-color image.

3.2.1.2 Gray Level to Color Transformation:

- In this method of pseudo color image processing, the input image f(x,y) is split into 3 separate channels.



- These 3 channels carry the input image to Red, Green, Blue Transformation devices.
- Here the transformation is done to each and every pixel independent of their gray level values.
- The final image that is produced is a composite made up of the 3 primary colors i.e. Red, Blue & Green

3.2.2 Full Color Image Processing:

There are two categories of Full-color image processing,

1. First approach is applied to the whole image color components individually and produce a composite color image.
2. The second approach the processing is applied to the color pixels directly.

A pixel of a colored image is a vector quantity and they process 3 color components in it,

$$C(x,y) = \begin{bmatrix} C_R(x,y) \\ C_G(x,y) \\ C_B(x,y) \end{bmatrix}$$

3.2.2.1 Color Transformation:

- Generalized transformation equation is given by,

$$g(x, y) = T[f(x, y)]$$

- Color transformation is used to transform pixel to a single-color model. It is not implemented to converts a HIS image to RGB image.

$$S_i = T_i(r_1, r_2 \dots \dots \dots ., r_n)$$

Where, S =Output pixel, T= Transformation, r = Input pixel and n = depends on color model (i.e: For RGB n=3)

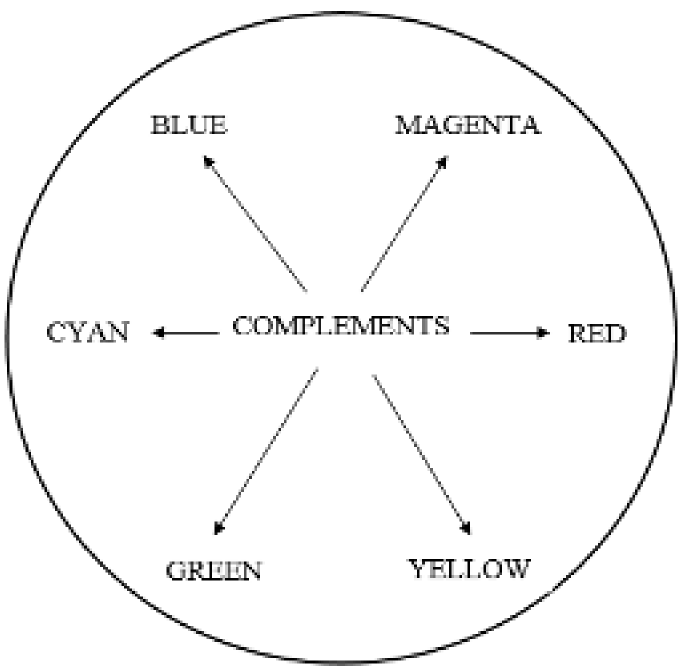
- Cost of conversion varies according to the color model in which the conversion is being done.
- For example, to modify the intensity of an image using;

$$g(x, y) = kf(x, y) \qquad 0 < k < 1$$

- For HIS model, the formula is: $S_3 = kr_3; S_2 = r_2; S_1 = r_1$
- For RGB model, the formula is: $S_i = kr_i$; Where i = 1,2,3.

3.2.2.2 Color Components:

- The colors lying directly opposite to one-another in a color circle are called components.
- Color complements are analogous to gay scale negatives.



3.2.2.3 Color Slicing:

- Color slicing is used to highlight a specific range of colors. This process can be used to display the colors of interest so that they stand out from the background.
- The basic principle behind slicing a color image is to map the colors outside the region of interest to an average neutral color.
- If the colors of interest are enclosed by a cube; the pixel transformation formula is given by,

$$S_i = \begin{cases} 0.5\,, if \ \left[\left|r_j - a_j\right| > \frac{W}{2}\right] for\ any\ 1 \le j \le n \\ r_i \quad , \qquad\qquad\qquad\qquad otherwise \end{cases}$$

Where, W- Width of the cube

a_1, a_2, ……a_n are the color components

S_i – o/p pixel and r_i – i/p pixel.

- If the colors of interest are enclosed by a sphere; the pixel transformation formula is given by

$$S_i = \begin{cases} 0.5\,, if \ \sum_{j=1}^{n} \left(r_j - a_j\right)^2 > R_0^2 \\ r_i\,, \qquad\qquad Otherwise \end{cases}$$

Where, R_0 – Radius of the sphere.

3.3 IMAGE RESTORATION:

Image restoration is defined as the process of reconstructing or recovering an image which is in the degraded or distorted state. But a knowledge of the degradation function is needed for a successful restoration.

Aim:

The aim of restoration is to improve the appearance of an image. This is same as the enhancement process. But, unlike image enhancement, restoration is an ***objective*** process, which means that the process is based on mathematical or probabilistic models of image degradation.

3.3.1 Image Restoration / Degradation model:

The figure 3.1 shows that the degradation process is modeled as a degradation function that together with an additive noise term operates on an input image f(x,y) to produce degraded image g(x,y). Given g(x,y), some knowledge about the degradation function H, and some knowledge about the additive noise term η(x,y), the objective of restoration is to obtain an estimate f^(x,y) of the original image.

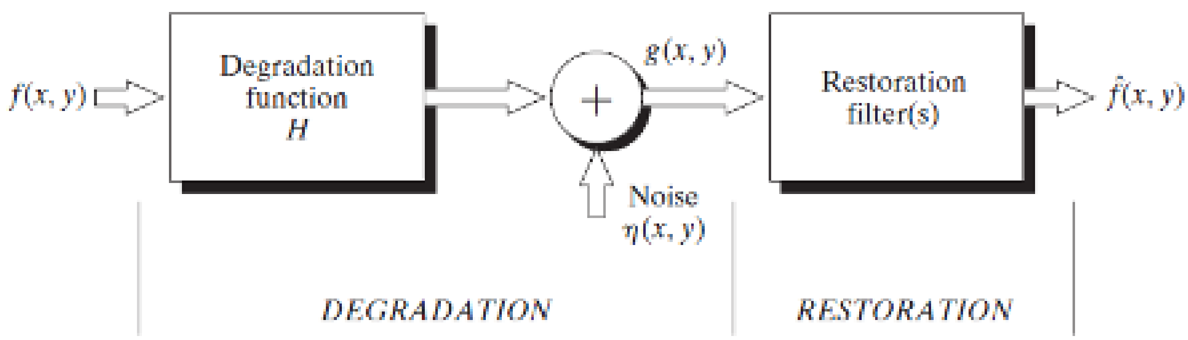


Fig: 3.1 A model of the image degradation / restoration process.

Degradation Process:

The Degradation unction here operates with an additive noise term. It works on the input image f(x,y) and produces a degraded image g(x,y)

Restoration Process:

The objective of restoration process is to obtain an estimate or approximation f^(x,y) of the original image. This approximation should be as close as possible to the original input image. When knowledge about H and n increases, the approximation f^(x,y) gets more closer to f(x,y).

Spatial Domain Representation:

Assuming that H is linear, position invariant process, the degraded image can be represented in the spatial domain as

$$g(x,y) = h(x,y) * f(x,y) + n(x,y)$$

Where, g(x,y) – Degraded image in spatial domain

h(x,y) – Spatial representation of the degradation function

f(x,y) – Original image

n(x,y) – Additive noise

Frequency Domain Representation:

Convolution in spatial domain is equal to the multiplication in the frequency domain, therefore, the degraded image in the frequency domain is represented by,

$$G(u,v) = H(u,v)F(u,v) + N(u,v)$$

Where, H(u,v) – Fourier transform of h(x,y)

F(u,v) – Fourier transform of f(x,y)

N(u,v) – Fourier transform of n(x,y)

<u>3.3.2 Types of Image Blur:</u>

Blur can be introduced by an improperly focused lens, relative motion between camera and the scene or atmospheric turbulence. Blurring is a form of bandwidth reduction of an ideal image owing to the imperfect image formation process.

1) **Gauss Blur:**

Gauss blur is defined by the following point-spread function:

$$h(x,y) = \frac{1}{\sigma\sqrt{2\Pi}} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Here, σ is called the variance of the blur occurs due to long time atmosphere exposure.

2) **Out-of-focus Blur:**

This blurring is produced by a defocused optical system. It distributes a single point uniformly over a disk surrounding the point. The point spread function of the out-of-focus blur is given by

$$h(x,y) = c\begin{cases} 1, \sqrt{(x-c_x)^2 + \left(y-c_y\right)^2} \ \le \ r \\ 0, \qquad\qquad\qquad Otherwise \end{cases}$$

Where, r is the radius and (c_x, c_y) is the centre of the out-of-focus point spread function.

3) Motion Blur

Motion blur is due to relative motion between the recording device and the scene. When an object or the camera is moved during light exposure, a motion blurred image is produced

$$h(x,y,L,\phi) = \begin{cases} \frac{1}{L} \ if \ \sqrt{x^2+y^2} \ \le \frac{L}{2} and \frac{x}{y} = \ -tan\phi \\ 0 \qquad\qquad\qquad Otherwise \end{cases}$$

Where, L – Length of motion

Φ – Angle of radiance

4) Atmospheric Turbulence Blur:

Atmospheric turbulence blur is introduced by variety of factors like temperature, wind speed, exposure time, for long exposures, the point spread function is given by,

$$h(x,y,\sigma_G) = C \ exp\left(-\frac{x^2+y^2}{2\sigma_G^2}\right)$$

Where, σ_G – Amount of spread of the blur

3.3.3 Algebraic Approach:

3.3.4 Inverse Filtering:

- Inverse filtering is defined as the process of ***recovering the input*** of a system from its output.
- Direct inverse filtering is the ***simplest technique*** of restoration.

Concept:

Let an image f(x,y) is degraded by a degradation function H. the obtained degraded image is denoted as g(x,y).

$$g(x,y) = h(x,y) * f(x,y) + n(x,y) \qquad (1)$$

The inverse filtering divides the transform of the degraded image G(u,v) by the degradation function H(u,v) and determines an approximation of the transform of the original image.

It is expressed as,

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)} \qquad (2)$$

Where, F$(u,v) - Transform\ of\ the\ original\ image$

$\hat{F}(u,v) - Approximation\ of\ F(u,v)$

Drawbacks:

The inverse filtering divides the transform of the degraded image G(u,v) by the degradation function H(u,v).

$$G(u,v) = H(u,v)F(u,v) + N(u,v) \qquad (3)$$

Substituting this in equation (2) gives,

$$\hat{F}(u,v) = \frac{H(u,v).F(u,v) + N(u,v)}{H(u,v)}$$

$$\hat{F}(u,v) = F(u,v) + \frac{N(u,v)}{H(u,v)} \qquad (4)$$

An exact inverse filtering expression can be obtained by substituting eq (2) in eq (4) as

$$F(u,v) = \frac{G(u,v)}{H(u,v)} - \frac{N(u,v)}{H(u,v)} \qquad (5)$$

In addition to the transform of the original image, there is an extra term N(u,v)/H(u,v) in the right hand side of equation (4). This shows that the approximated function is not equal to the original function.

- The original or undegraded image can't be fully recovered by this method. Due to that the fourier transform of N(u,v) is unknown since it's a random function.
- Zero or small-value problem (If the degradation function value is ***small or zero***, the second term in eq(4) will have higher value than the first term,

F(u,v). This indicates the ***poor performance*** of the system which results in bad approximation of the original function. This is known as the ***'Zero or small-value problem'***

- If noise is present in the region of vertical high value strips, the inverse filtering corrupts the restored image.
- This method has no explicit provision to handle the noise.

Limitation:

- Inverse filtering is highly sensitive to noise.

Application:

- Removal of blur caused by uniform linear motion.

3.3.5 Wiener (or) Least Mean Square (LMS) Filtering:

- For restoration of an image, this method considers the degradation function as well as the statistical properties of noise.
- Therefore, it is used to improve direct inverse filtering because, it has a provision to handle the noise.

Objective:

The main objective of LMS filtering is to approximate the original image in such a way that the mean square error between the original and approximated images is minimized.

Mean Square Error:

The mean square error is found by taking the expected value of the difference between two images. It is expressed as,

$$\boldsymbol{e^2 = E\{(f - \hat{f})^2\}} \qquad (6)$$

Where, E{x} -Expected value of x

f – Uncorrupted (or) original image.

$\boldsymbol{\hat{f}}$- Approximation of 'f'

Assumptions Made:

The following assumptions are made to perform LMS error filtering.

- The image and noise are uncorrelated, (i.e) they have no relation.
- Either image or noise has zero mean.
- The approximated gray levels are a linear function of the degraded gray levels.

Approximated Image:

Based on the assumptions made, the approximated image in frequency domain which satisfies the minimum error function is given by

$$\hat{F}(u,v) = \left[\frac{H^*(u,v).S_f(u,v)}{S_f(u,v).|H(u,v)|^2 + S_n(u,v)}\right]G(u,v)$$

Simplifying the above equation gives,

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \frac{S_n(u,v)}{S_f(u,v)}}\right]G(u,v)$$

Multiplying and Dividing the above equation by H(u,v) gives,

$$\hat{F}(u,v) = \left[\left(\frac{1}{H(u,v)}\right)\frac{|H(u,v)|^2}{|H(u,v)|^2 + \frac{S_n(u,v)}{S_f(u,v)}}\right]G(u,v) \qquad (7)$$

[$H(u,v) . H^*(u,v) = |H(u,v)|^2$]

Where, H(u,v) – Transform of the degradation function.

$H^*(u,v)$ – Complex conjugate of H(u,v)

$S_n(u,v) = |N(u,v)|^2$ (Power spectrum of noise)

$S_f(u,v) = |F(u,v)|^2$ (Power spectrum of original image)

G(u,v) – Transform of degraded image

Equation (7) is called ***wiener filter*** and the terms inside the ***square brackets*** are referred to as the ***minimum mean square error filter*** or ***least mean square filter***.

If N(u,v) = 0;

When the noise N(u,v) = 0, then eq(7) becomes,

$$\hat{F}(u,v) = \left[\left(\frac{1}{H(u,v)}\right)\frac{|H(u,v)|^2}{|H(u,v)|^2 + 0}\right]G(u,v)$$

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)} \qquad (8)$$

The Eq(8) is similar to Eq(2) for inverse filtering. Thus, if noise is zero, Winer filter reduces to inverse filter.

With unknown quantities:

If power spectrum of the undegraded image $S_f(u,v)$ is unknown, the Eq(7) can be written as,

$$\hat{F}(u,v) = \left[\left(\frac{1}{H(u,v)}\right)\frac{|H(u,v)|^2}{|H(u,v)|^2 + K}\right]G(u,v) \qquad (9)$$

Where, K- Specified constant

Here, the noise is assumed to be ***White noise,*** Whose spectrum $|N(u,v)|^2$ is a constant.

Advantages:

- Wiener filter has no small or zero-value problem
- The results obtained are more closer to the original image than the inverse filtering

Disadvantages:

- It requires the power spectrum of the undegraded image and noise to be known, which makes the implementation more difficult.
- Wiener filter is based on minimizing a statistical criterion. Therefore, the results are optimal only in an average sense.

3.3.6 Constrained Least Squares Filtering:

- Constrained least squares filtering is a restoration technique which uses only the mean and variance of the noise.

Matrix Formulation:

The basic operation of the restoration process is given by an equation

$$g(x,y) = h(x,y) * f(x,y) + n(x,y)$$

The above equation can be written in ***vector-matrix*** form as

$$g = Hf + n \tag{10}$$

If g(x,y) has the size MxN

- g, f and n matrices will have dimensions MNx1.
- H matrix will have MNxMN dimensions.

Problems Encountered:

There are two main problems in the matrix calculation of constrained least squares filtering. They are:

(1) Even for medium size image, the matrix dimensions will be very large, which makes the computation difficulty.
(2) The matrix 'H' in Eq(10) is highly sensitive to noise.

Handling Noise Sensitivity:

The noise sensitivity problem can be reduced by performing the restoration based on some measures of smoothness, such as the Laplacian.

In such a case, the minimum value of a criterion function, C should be found. The function is defined as

$$C = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[\nabla^2 f(x,y)\right]^2 \tag{11}$$

The ***constraint*** here is:

$$\left|\left|g - H\hat{f}\right|\right|^2 = \left|\left|n\right|\right|^2 \tag{12}$$

Solution:

The solution to this problem in frequency domain is given by,

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \gamma |P(u,v)|^2}\right] G(u,v) \tag{13}$$

Where, γ – Parameter adjusted to satisfy the equation (12)

P(u,v) – Fourier Transform of the Laplacian operator given by,

$$p(x,y) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

If γ = 0, then equation (13) becomes,

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + 0.|P(u,v)|^2}\right] G(u,v)$$

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)}$$

Which is the inverse filtering operation. Thus, when γ = 0, constraint least square filter reduces to inverse filter.

Computation of γ:

'γ' is an important parameter which is adjusted until the desired results are obtained. Therefore, the computation of γ value should be done carefully.

Iteration Method:

1) Define a residual vector r given by

$$r = g - H\hat{f}$$

2) Now, it can be proved that ϕ(γ) is a monotonically increasing function of γ defined as,

$$\phi(\gamma) = r^T.r = \left||r|\right|^2$$

3) Next, adjust γ to satisfy the expression

$$\left||r|\right|^2 = \left||n|\right|^2 \pm a$$

Where, a – Accuracy Factor.

Newton – Raphson Algorithm:

This algorithm is used to improve the speed of restoration process in constrained least square filtering. Two quantities are needed to implement this algorithm.

(i) $||r||^2$
(ii) $||n||^2$

(i) To find $||r||^2$

The steps to find $||r||^2$ are:

i. The Residual vector equation can be rewritten as,

$$R(u,v) = G(u,v) - H(u,v)\hat{F}(u,v)$$

ii. Taking the inverse transform of R(u,v) gives r(x,y).

iii. Now, $||r||^2$ can be obtained by the equation,

$$||r||^2 = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} r^2(x,y)$$

(ii) To find $||n||^2$

The steps to find $||n||^2$ are:

i. Calculate the sample mean of the noise from the expression

$$m_n = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} n(x,y)$$

ii. Find the variance of the noise over the entire image using the sample average method. This is expressed as

$$\sigma_n^2 = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [n(x,y) - m_n]^2$$

iii. Now calculate $||n||^2$ by using

$$||n||^2 = MN[\sigma_n^2 - m_n]$$

Advantages:

- Only the mean and variance of the noise are required to be known to implement this method.
- For each input image, it produces an optimal result.