ignou
THE PEOPLE'S
UNIVERSITY

Indira Gandhi National Open University

MCS-230
Digital Image Processing and Computer Vision

Block

2

DIGITAL IMAGE PROCESSING-II

Unit 5 Transformation Techniques		111
Unit 6 Image enhancement and Filtering		137
Unit 7 Color image processing		188

PROGRAMME DESIGN COMMITTEE

Prof. (Retd.) S.K. Gupta , IIT, Delhi

Prof. Ela Kumar, IGDTUW, Delhi

Prof. T.V. Vijay Kumar JNU, New Delhi

Prof. Gayatri Dhingra, GVMITM, Sonipat

Mr. Milind Mahajan,
Impressico Business Solutions, New Delhi

Sh. Shashi Bhushan Sharma, Associate Professor, SOCIS, IGNOU

Sh. Akshay Kumar, Associate Professor, SOCIS, IGNOU

Dr. P. Venkata Suresh, Associate Professor, SOCIS, IGNOU

Dr. V.V. Subrahmanyam, Associate Professor, SOCIS, IGNOU

Sh. M.P. Mishra, Assistant Professor, SOCIS, IGNOU
Dr. Sudhansh Sharma, Assistant Professor, SOCIS, IGNOU

COURSE DESIGN COMMITTEE

Prof. T.V. Vijay Kumar, JNU, New Delhi

Prof. S.Balasundaram, JNU, New Delhi

Prof. D.P. Vidyarthi, JNU, New Delhi

Prof. Anjana Gosain, USICT, GGSIPU,
New Delhi

Dr. Ayesha Choudhary, JNU, New Delhi

Sh. Shashi Bhushan Sharma, Associate Professor, SOCIS, IGNOU

Sh. Akshay Kumar, Associate Professor, SOCIS, IGNOU

Dr. P. Venkata Suresh, Associate Professor, SOCIS, IGNOU

Dr. V.V. Subrahmanyam, Associate Professor, SOCIS, IGNOU
Sh. M.P. Mishra, Assistant Professor, SOCIS, IGNOU

Dr. Sudhansh Sharma, Assistant Professor, SOCIS, IGNOU

SOCIS FACULTY

Prof. P. Venkata Suresh, Director, SOCIS, IGNOU

Prof. Sandeep Singh Rawat, SOCIS, IGNOU

Dr. Akshay Kumar, Associate Professor, SOCIS, IGNOU

Dr. Sudhansh Sharma, Assistant Professor, SOCIS, IGNOU

Prof. V.V. Subrahmanyam, SOCIS, IGNOU

Prof. Divakar Yadav, SOCIS, IGNOU, New Delhi

Dr. M.P. Mishra, Associate Professor, SOCIS, IGNOU

COURSE PREPARATION TEAM

This Course is adapted from MMTE-003–Digital Image Processing and Pattern Recognition of School of Sciences, IGNOU

Content Editor

Prof K.K. Biswas (Retd.) - (*Content Editor*)
IIT Delhi.

Course Writer

Unit 5, 6 and 7

Dr. Ayesha Choudhary
School of Computers and Systems Sciences
JNU, New Delhi

COURSE COORDINATOR

Dr.Sudhansh Sharma,
Assistant Professor,
School of Computers and Information Sciences, IGNOU.

PRINT PRODUCTION

Sh Sanjay Aggarwal
Assistant Registrar, MPDD, IGNOU, New Delhi

April, 2023

Further information on the Indira Gandhi National Open University courses may be obtained from the University's office at Maidan Garhi, New Delhi-110068.
Printed and published on behalf of the Indira Gandhi National Open University, New Delhi by MPDD, IGNOU.
Laser Typesetter: Tessa Media & Computers, C-206, Shaheen Bagh, Jamia Nagar, New Delhi-110025

BLOCK 2 INTRODUCTION

In this Block we shall see how the quality of images can be enhanced in frequency domain. In earlier Block-1 we discussed various image enhancement and filtering techniques in the Spatial domain, very often the acquired images are not of very good quality. Sometimes overall image is dark or very bright and lot of interesting details are not very clear. Often the image is noisy because of problems in data acquisition.

In Unit 5, various image transformation techniques in frequency domain viz. DFT, DCT, DWT, Haar Transform are discussed

In Unit 6, relates to the discussion of the operations performed for the filtering of image in frequency domain, it also covers various concepts viz. Image smoothening, Image Sharpening, image degradation models,, various noise models are also discussed. Finally the unit concludes with the discussion over Inverse filtering and Wiener Filtering.

Humans use color information to distinguish objects, materials, food, places and time of the day. Color images surround us in print, television, computer displays, photographs and movies. Color images in general occupy more space when compared to grayscale and binary images.

In Unit 7, we shall discuss basic concepts related to color formation, human perception of color information, different types of color models RGB, HIS and CMY are discussed.

UNIT 5 IMAGE TRANSFORMATIONS-FREQUENCY DOMAIN

Structure **Page No.**

5.1	Introduction Objective	111
5.2	Discrete Fourier Transform	112
5.3	Discrete Cosine Tranform	119
5.4	Discrete Wavelet Transform	121
5.5	Haar Transform	127
5.6	Summary	132
5.7	Solutions/ Answers	133

5.1 INTRODUCTION

As we saw in Unit 2, an image can be transformed, to show or hide information in the image. Image transformations can be done in both the spatial and the frequency domain. In the spatial domain, image transformation is carried out by changing the value of the pixels based on certain constraints. These transformations can change the brightness and clarity of the images.

In this unit, we shall focus on the transformations in the frequency domain. We recall at this point that an image is also a 2D signal and that the transformations that can be applied on a signal can also be applied on any image. The transformations in the frequency domain provide us information on the frequency content of the image. These transformations can help represent the information in the image in a more compact form, thereby making it computationally easier to store and transmit the images. The transformations may also help in separating the noise and the salient information present in the image.

In Sec. 5.2, we shall focus on very important and useful image transformations, namely the Discrete Fourier transformation (DFT). We shall continue our discussion in Sec. 5.3 with the Discrete Cosine Transformation (DCT). Subsequently, Discrete Wavelet Transform will be discussed in Sec. 5.4 Thereafter, In Sec. 5.5, Haar transform will be discussed. As we go through this unit, we shall see the unique properties of each of these transforms.

Now we shall list the objectives of this unit. After going through the unit, please read this list again and make sure that you have achieved the objectives.

Objectives

After studying this unit you should be able to:

- find the Discrete Fourier Transform (DFT)
- compute the Discrete Cosine Transform (DCT)
- find the Discrete Wavelet Transform(DWT)
- find the Haar Transform
- apply the above mentioned transforms

We shall begin the unit with Discrete Fourier Transform(DFT).

5.2 DISCRETE FOURIER TRANSFORM

The Discrete Fourier Transform (DFT) transfers an image from the spatial domain to the frequency domain. It is one of the most important transforms in image processing, which enables us to decompose an image into its sine and cosine components. The output image after applying the Fourier transformation is represented as a linear combination of a collection of sine and cosine waves of different frequencies.

Consider a 1D function, $\{f(x), 0 \leq x \leq N-1\}$. The general form of a transformation is

$$g(u) = \sum_{x=0}^{N-1} T(u,x) f(x);\ 0 \leq u \leq N-1 \qquad (1)$$

where $T(u,x)$ is called the **forward kernel** of transformation and $g(u)$ is the transformed image.

If the transformation is the Discrete Fourier Transform.

Then,

$$g(u) = \sum_{x=0}^{N-1} \frac{1}{N} e^{-i2\pi\frac{x}{N}} f(x); u = 0,1,2,...N-1 \qquad (2)$$

The inverse 1-D DFT will then be,

$$f(x) = \sum_{x=0}^{N-1} e^{i2\pi\frac{ux}{N}} g(u) \qquad (3)$$

As can be seen the signal is written as a linear combination of an orthogonal set of basis functions. Similarly, an image can be transformed into a set of "basis images", which can be used for representing the image.

We can extend the transform to 2-D image.

Consider an image $f(x,y)$ of size $M \times N$. The 2-D DFT of $f(x,y)$ is defined as follows:

$$F(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y) e^{-i2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \quad (4)$$

And the inverse 2-D DFT is given by:

$$f(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v) e^{i2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \quad (5)$$

The 2-D DFT is separable, symmetric and unitary. In case of square images, $M = N$. Many a time in image processing we work with square images. Additionally if the imagesize is a power of 2, then DFT implementation becomes very easy. Computational complexity can be reduced by efficient algorithms such as FFT.

The representation of intensity as a function of frequency is called '**spectrum**'. In the Fourier domain image, each point to a particular frequency contained in the spatial domain image. The coordinates of the Fourier spectrum are the spatial frequencies. The spatial position information of an image is encoded as the difference between the coefficients of the real and imaginary parts. This difference is called the "**phase angle**". The phase information is very useful for recovering the original information. The phase information represents the edge information or boundary information of the objects present in an image. For applications such as medical image analysis, the phase information is very crucial in getting information from the image.

Note that while the image values $f(x,y)$ are going to be real, the corresponding frequency domain data is going to be complex. There will be one matrix containing real values $R(u,v)$ and the other matrix $I(u,v)$ will contain the imaginary component of the complex value. The amplitude spectrum or the magnitude for 2D DFT is given by

$$|F(u,v)| = [R^2(u,v) + I^2(u,v)]^{1/2} \quad (6)$$

where, R and I are real and imaginary parts of $F(u,v)$ and and all computations are carried out for the discrete variables $u = 0,1,2,\ldots,M-1$ and $v = 0,1,2,\ldots,N-1$. The spectrum tells us the relative magnitude at each frequency.

The power spectrum of the 2-D DFT is defined as

$$P(u,v) = |F(u,v)|^2 = R^2(u,v) + I^2(u,v) \quad (7)$$

and the phase spectrum of the 2-D DFT is given by

$$\text{ø}(u,v) = \tan^{-1}\frac{I(u,v)}{R(u,v)} \quad (8)$$

Note that the size of the image remains the same as the original image in spatial domain.Therefore, the magnitude, Fourier (phase) spectrum andthe power spectrum are all matrices of size $M \times N$.

Remark: We can find 2-D DFT of an image by simply computing a set of 1-D DFT is for all rows of $f(x, y)$. Thus, the 2-D DFT of an image $f(x, y)$ is

$$F(u, v) = \sum_{x=0}^{M-1} e^{\frac{-2\pi i u x}{M}} \sum_{y=0}^{N-1} f(x, y) e^{\frac{-2\pi i v y}{N}}.$$

$$= \sum_{x=0}^{M-1} F(x, v) e^{\frac{-2\pi i u x}{M}},$$

where $F(x, v) = \sum_{y=0}^{N-1} f(x, y) e^{\frac{-2\pi i v y}{N}}$.

Also, the 2-D DFT can also be found using the Eqn. (4) with the condition of separability as we used in Unit-2.

Let us discuss properties of 2-D DFT.

DFT has several useful properties which makes it an important transformation.

i) **Separability:** It is separable because a 2D transform is separable if $T(u, x, v, y) = T_1(u, x) \cdot T_2(v, y)$.

ii) **Symmetry:** It is symmetric because a 2D transform is symmetric if $T_1(u, x) = T_2(u, x)$.

iii) **Periodicity:** The 2-D DFT and the 2-D IDFT are both periodic, that is, $F(u, v) = F(u+M, v) = F(u, v+N) = F(u+M, v+N)$.

iv) **Conjugate symmetry:** $F(u, v) = F^*(-u+pM, -v+qN)$, where p and q being integers. The property of conjugate symmetry implies that $|F(u, v)| = |F(-u, -v)|$

v) If $f(x, y)$ is real and even then $F(u, v)$ is real and even.

vi) If $f(x, y)$ is real and odd then $F(u, v)$ is imaginary and odd.

vii) Let F be the DFT operator, then

$$F(f(x, y) + g(x, y) = F(f(x, y) + F(g(x, y)$$

However, $F(f(x, y) \cdot g(x, y) \neq F(f(x, y) \cdot F(g(x, y)$

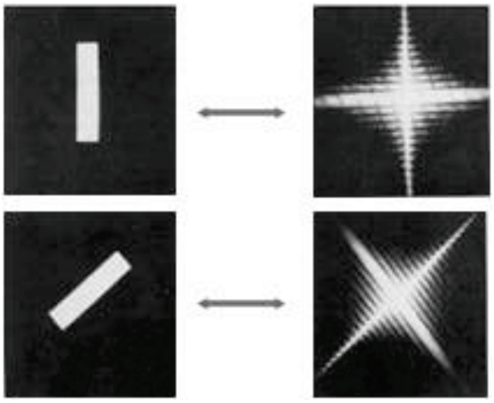
viii) Translation in the spatial domain by (x_0, y_0) implies

$$f(x - x_0, y - y_0) \leftrightarrow F(u, v) e^{-j2\pi\left(\frac{ux_0}{M} + \frac{vy_0}{N}\right)}$$

While translation in the frequency domain by (u_0, v_0) implies

$$F(u - u_0, v - v_0) \leftrightarrow f(x, y) e^{j2\pi\left(\frac{xu_0}{M} + \frac{yv_0}{N}\right)}$$

ix) The average value of the signal is given by



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

If we see the value of $F(0,0) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y) \Rightarrow F(0,0) = \bar{f}(x,y)$.

x) **Rotation:** Rotating $f(x,y)$ by θ rotates $F(u,v)$ by θ.

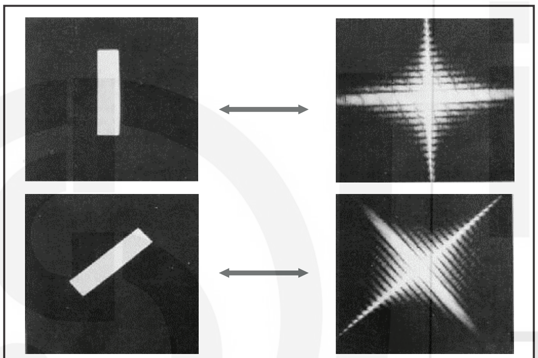


Fig. 1: Rotation in $f(x,y)$

Fig. 1 shows how the rotation in the spatial domain (on left) affects the rotation in the frequency domain (right).

After going through all the properties of DFT, let us see how do we visualize 2-D DFT. We need to translate the origin of the transformed image to the center of the image $(u,v) = (M/2, N/2)$ to be able to display the full period of the 2-D DFT. As we saw above, translating the Fourier image to the center, requires us to use the translation property of $F(u,v)$ with $u_0 = M/2$ and $v_0 = N/2$.

Then, $F\left\{f(x,y)e^{i2\pi\left(\frac{xu_0}{M}+\frac{yv_0}{N}\right)}\right\} = F(u-u_0, v-v_0)$ becomes

$$F\{f(x,y)e^{i\pi(x+y)}\} = F\{f(x,y)(-1)^{(x+y)}\} = F(u-M/2, v-N/2)$$

a) Original Image **b) Original DFT image** **c) Translated DFT image**

Fig. 2: DFT of an Original Image

We can see in Fig. 2 what changes do we see after DFT. Fig 2. (a) is the original image in the spatial domain, (b) is the 2-D DFT image, and (c) is the translated DFT image to show the full period of the 2D DFT of image in (a).

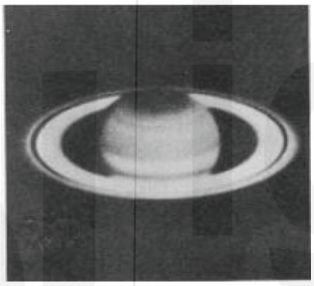
Let us see how do we visualize the range of 2-D DFT.

In general, the range of values the 2-D DFT $F(u,v)$ is very large. Therefore, when we attempt to display the values of $F(u,v)$, smaller values are not distinguishable because of quantization as can be seen in Fig. 3 (b). Therefore, to enhance the small values, weapply a logarithmic transformation given by

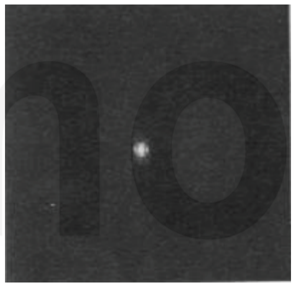
$$D(u,v) = c\log(1+|F(u,v)|)$$

Where, the parameter *c*is chosen so that the range of $D(u,v)$ is $[0,255]$.

$$c = \frac{255}{\log(1+\max\{|F(u,v)|\})}$$



a) The original image



b) The 2D DFT image

c) The 2D DFT image after log transform

Fig. 3: DFT image after log tranform

We can visualise the display of the amplitude of the 2-D DFT after logarithmic transformation in Fig. 3(b) and Fig. 3(c) respectively for the original image as shown in Fig. 3(a).

Example 1: Compute the DFT of the 1D sequence $f(x) = [1,0,-1,0]$.

Solution: Here $N = 4$. Using Eqn. (2) we get

$$g(u) = \frac{1}{4}\sum_{x=0}^{3} f(x)\cdot e^{\frac{-i,2\pi\cdot ux}{4}}; \; u = 0,1,2,3$$

$$= \frac{1}{4}\sum_{x=0}^{3} f(x)\cdot \left(e^{\frac{-2\pi i}{4}}\right)^{ux}; \; u = 0,1,2,3$$

$$= \frac{1}{4}\sum_{x=0}^{3} f(x)\ (-i)^{ux}; \quad u = 0,1,2,3$$

$$= \frac{1}{4}[f(0)(-i)^{0} + f(10(-i)^{u} + f(2)(-i)^{2u} + f(3)(i)^{3u}]; u = 0,.1,2,3$$

$$= \frac{1}{4}[1 + 0 + (-1)(-i)^{2u} + 0]; u = 0,1,2,3$$

$$= \frac{1}{4}[1 - (i)^{2u}]; u = 0,1,2,3$$

This gives $\mathbf{g} = \frac{1}{4}[0, 2, 0, 2]$, which is the DFT of $f(x)$.

***

Example 2: Construct a DFT matrix of order 2.

Solution: Here $N = 2$.

$$g(u) = \frac{1}{2}\sum_{x=0}^{1} f(x) e^{\frac{-i.2\pi.ux}{2}}; u = 0,1.$$

$$= \frac{1}{2}\sum_{x=0}^{1} f(x)(-1)^{ux}; u = 0,1$$

$$= \frac{1}{2}[f(0)(-1)^{0} + f(1)(-1)^{u}]; \quad u = 0,1$$

$$= \frac{1}{2}[f(0) + (-1)^{u} f(1); u = 0,1$$

Now, $g(0) = \frac{1}{2}[f(0) + f(1)]$

$g(1) = \frac{1}{2}[f(0) - f(1)]$

DFT matrix $= \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

***

Example 3: Compute the 2-D DFT of the 2×2 image

$$f(x, y) = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

Solution: Let the DFT of $f(x, y)$ be $F(u, v)$, which is given in Eqn. (4).

$$F(u, v) = \sum_{x=0}^{1}\sum_{y=0}^{1} f(x, y) e^{-2\pi i\left(\frac{ux}{2} + \frac{vy}{2}\right)}; u, v = 0, 1$$

$$= \sum_{x=0}^{1}\sum_{y=0}^{1} f(x, y)(-1)^{ux}(-1)^{vy}; u, v = 0, 1$$

$$= [f(0,0)(-1)^{0}(-1)^{0} + f(0,1)(-1)^{0}(-1)^{v} + f(1,0)(-1)^{u}(-1)^{0}$$

$$+ f(1,1)(-1)^{u}(-1)^{v}]; \quad u, v = 0, 1$$

$$= [f(0,0) + (-1)^{v} f(0,1) + (-1)^{u} f(1,0) + (-1)^{u}(-1)^{v} f(1,1)]; u, v = 0, 1,1.$$

$$F(0,0) = f(0,0) + f(0,1) + f(1,0) + f(1,1) = 4$$
$$F(1,0) = f(0,0) + f(0,1) - f(1,0) - f(1,1) = 0$$
$$F(0,1) = f(0,0) - f(0,1) + f(1,0) - f(1,1) = 0$$
$$F(1,1) = f(0,0) - f(0,1) - f(1,0) + f(1,1) = 0$$

Thus, the 2-D DFT of the given image is $\begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix}$.

F(0,0) is 4 which happens to be the average of all the intensity values in original image. The other values represent frequency values. But since there is no variation in values of original image, there is no frequency involved, and that is why the frequency values in DFT are zeroes.

Alternatively, the 2-D DFT can also be found using the DFT basis matrix formed by finding 1-D DFT of each row of $f(x, y)$ and then using that as kernel.

That is $F(u, v) = \text{kernel} \times f(x, y) \times (\text{kernel})^T$

We have already found the DFT matrix of order 2 in Example 2.

Therefore,

$$\text{kernel} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Hence, $F(u, v) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ [we are skipping $\frac{1}{2}$ as in 2-D we have taken $\frac{1}{MN}$ in inverse DFT]

$$= \begin{bmatrix} 2 & 2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix}$$

Both the results are same.

***

Try the following exercises.

E1) Does the implementation of a separable and symmetric transform, such as the DFT in an image requires the sequential implementation of the corresponding one-dimensional transform row-by-row and then column-by-column? Justify your answer.

E2) Find the DFT of the sequence $f(x) = [i, 0, i, 1]$.

E3) Construct a DFT matrix of order 4. Also, check whether DFT matrix is unitary matrix or not.

E4) Find the inverse 2-D DFT of $F(u, v)$ found in Example 3.

In the following section, we shall discuss discrete cosine transform.

5.3 DISCRETE COSINE TRANSFORM

The Discrete Cosine Transform DCT is a family of unitary transformations that transforms the real values of input image to another set of real values. Unlike the DFT that is complex, the DCT is a real transform because it projects the signal onto real cosinewaves.

The 1-D DCT is given as:

$$C(u) = a(u)\sum_{x=0}^{N-1} f(x)\cos\left[\frac{(2x+1)u\pi}{2N}\right]; 0 \le u \le N-1, \qquad (9)$$

$$\text{where, } a(u) = \begin{cases} \sqrt{\dfrac{1}{N}}; & u = 0 \\ \sqrt{\dfrac{2}{N}}; & u = 1, \ldots, N-1 \end{cases}$$

Then, the inverse transform is given by

$$f(x) = \sum_{u=0}^{N-1} a(u)C(u)\cos\left[\frac{(2x+1)u\pi}{2N}\right],$$

(10)

where $a(u)$ is the same function as used for DCT.

Let us visualize the effect of 1-D DCT through Fig. 4, where in the rows of the 8×8 transformation matrix of the DCT for a signal $f(x)$ with 8 samples are shown.

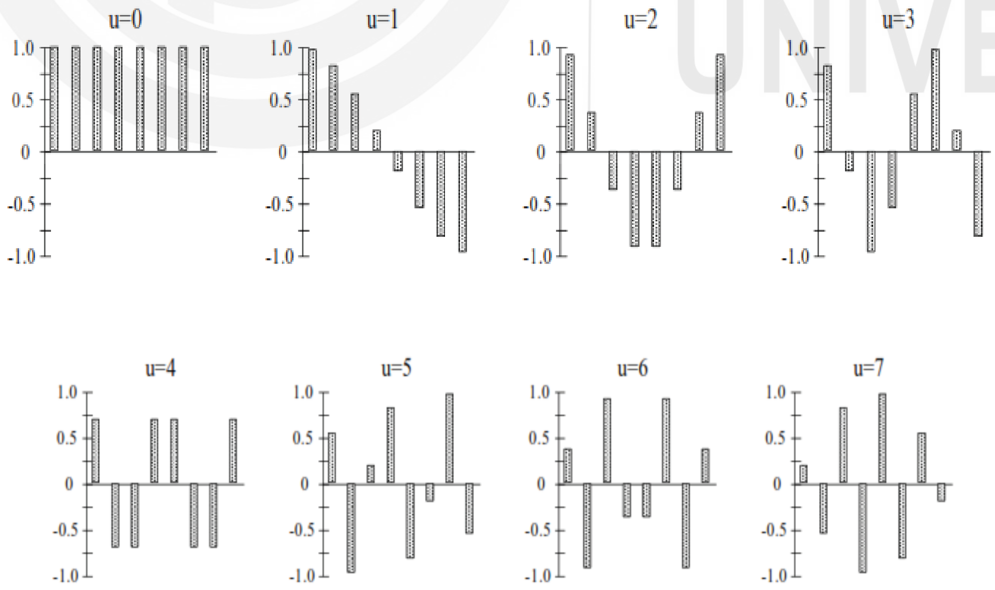


Fig. 4: 1-D DCT

The figures for values of u from 0 to 7 show the various rows of the 8×8 transformation matrix of the DCT for a 1D signal $f(x)$ with 8 samples.

Let us now expand 1-D DCT to 2-D DCT.

Consider an image $f(x, y)$ of size $M \times N$. Then, the 2-D DCT of the image is defined as:

$$C(u, v) = a(u)a(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y)\cos\left[\frac{(2x+1)u\pi}{2M}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right], \quad (11)$$

$$0 \le u \le M-1, 0 \le v \le N-1$$

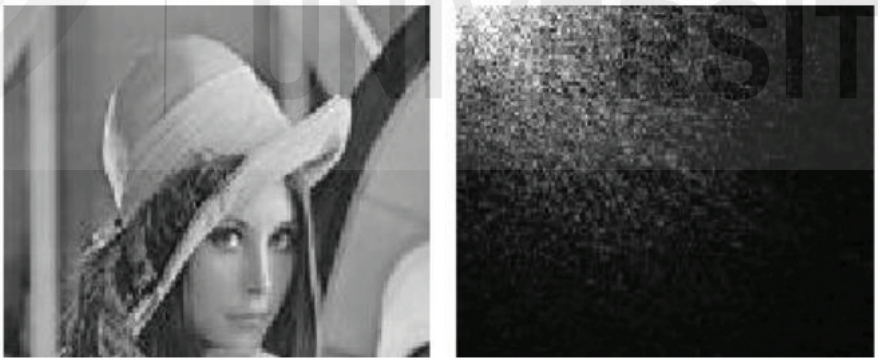
And the inverse 2-D DCT is given by

$$f(x, y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} a(u)a(v)C(u, v)\cos\left[\frac{(2x+1)u\pi}{2M}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (12)$$

$$0 \le x \le M-1, 0 \le y \le N-1$$

Where $a(u)$ is same as defined earlier for 1D DCT.

DCT and DFT are very similar, however, DCT has the advantage over DFT that DCT is real while DFT is complex. Moreover. DCT has better energy compaction in comparison to the energy compaction of DFT. Energy compaction is the ability to pack the energy of the spatial sequence into as few frequency coefficients as possible. This property is exploited for image compression and is a very important property.You can see in the Fig.5, that most of the DCT image is dark, which means DCT values are concentrated only in few pixels very near the origin. This indicates that DCT has high compaction.



a) Original image – Lena **b) 2D DCT of Lena**

Fig.5: The 2D DCT in (b) of the image Lena in (a) shows the high compaction capability of DCT.

Example 4: Compute the discrete cosine transform (DCT) matrix for order 2.

Solution: Using Eqn. (9), we substitute $N = 2,$ and we get

$$C(u) = a(u)\sum_{x=0}^{1} f(x)\cos\frac{(2x+1)u\pi}{2\times 2}; \; 0 \le u \le 1.$$

$$\text{where } a(u) = \begin{cases} \dfrac{1}{\sqrt{2}} & ; \quad u = 0 \\ \sqrt{\dfrac{2}{2}} = 1; & u = 1 \end{cases}$$

At $u = 0$, we get

$$C(0) = \frac{1}{\sqrt{2}} \sum_{x=0}^{1} f(x) \cos \frac{(2x+1)\pi \times 0}{4}$$

$$= \frac{1}{\sqrt{2}} \sum_{x=0}^{1} f(x) \cdot 1$$

$$= \frac{1}{2} \sum_{x=0}^{1} f(x)$$

$$= \frac{1}{2} [f(0) + f(1)]$$

At $u = 1$, we get

$$C(1) = 1 \cdot \sum_{x=0}^{1} f(x) \cdot \cos \frac{(2x+1) \cdot 1 \cdot \pi}{4}$$

$$= \sum_{x=0}^{1} f(x) \cos \frac{(2x+1)\pi}{4}$$

$$= \left[f(0) \cos \frac{\pi}{4} + f(1) \cos \frac{3\pi}{4} \right]$$

$$= \frac{1}{\sqrt{2}} f(0) - \frac{1}{\sqrt{2}} f(1)$$

By collecting the coefficients, we get the required DCT. Therefore, DCT is

$$C(u) = \begin{bmatrix} \dfrac{1}{2} & \dfrac{1}{2} \\ \dfrac{1}{\sqrt{2}} & \dfrac{-1}{\sqrt{2}} \end{bmatrix}$$

***

Now try the following exercises.

E5) Why is DCT important for image compression?

E6) Find the DCT of the matrix of order 4.

So far, we have discussed discrete fourier transform and discrete cosine transform.

5.4 DISCRETE WAVELET TRANSFORM

In Block-1 of this course we learned about the Spatial domain, it was learned that the Spatial domain is the normal image space where the term "the domain" refers to the normal image space that is represented as a matrix of pixels, . In Spatial domain, the transformation methods are executed by directly operating on the pixel values of an image. Adjustments in spatial domain are made to the values in order to obtain the desired level of improvement.

In earlier sections of this unit we learned about the second type of domain i.e. the frequency domain, where the pace at which the individual color components in an image shift is referred as the image's frequency and in this domain i.e. frequency domain the prime focus is on the rate at which the pixel values in the spatial domain vary. It is to be noted that, in any image the color changes very quickly, for the regions with high frequencies, whereas in regions that contain low frequencies, the color changes quite gradually.

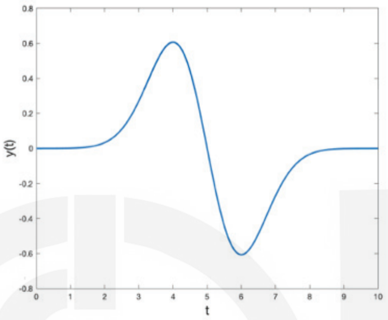
It is essential to keep in mind that, in contrast to the spatial domain, the frequency domain does not provide direct operations on the values. This restriction prevents you from performing some calculations in the frequency domain. In order to begin the processing of the image, it must first go through a transformation that restores it to its original frequency distribution. It is possible to separate the frequency components into two basic sub-components. Components with a high frequency that correlate to the edges of an image and components with a low frequency relates to the smooth regions of an image . This technique does not result in the production of an image as its end product; rather, it produces a transformation as its conclusion. It is necessary to carry out an inverse transformation on the data that was produced as a result of the processing that was done so that the image can be restored to its perfect, original form.

Also, we learned that Fourier transform is a powerful tool that has been available to signal analysts for many years. It gives information regarding the frequency content of a signal. However, the problem with using Fourier transforms is that frequency analysis cannot offer both good frequency and time resolution at the same time. A Fourier transform does not give information about the time at which a particular frequency has occurred in the signal. Hence, a Fourier transform is not an effective tool to analyse a non-stationary signal. To overcome this problem, windowed Fourier transform, or short-time Fourier transform, was introduced. Even though a short-time Fourier transform has the ability to provide time information, multi-resolution is not possible with short-time Fourier transforms. Wavelet is the answer to the multi-resolution problem. A wavelet has the important property of not having a fixed-width sampling window.

The technique of wavelet transformation, which is typically utilised for the analysis of images and the compression of data, will be investigated in this section. In spite of the fact that the frequency domain encompasses a number of other mathematical transformations, including the Fourier transform, the Laplace transform, and the Z transform, the wavelet transformation method is going to be the one that is discussed in detail in this section.

Let's begin with acquiring an understanding of what wavelets are and why we need this transformation before we move on to trying to comprehend the Discrete Wavelet Transformation, often known as the DWT. This will help us get a better grasp on the DWT. According to Wikipedia, "a wavelet is a wave-like oscillation with an amplitude that begins at zero, rises, and then decreases back to zero." It is best to think of it as a "short oscillation" in the majority of situations, which is analogous to what could be captured by a seismograph or heart monitor.

Let’s try to understand this concept of wavelet in a better way, with the explanation given below:
A wavelet is a wave-like oscillation that is localised in time; an example of this type of oscillation is provided further down in this paragraph. Scale and location are the two fundamental features that wavelets possess. How "stretched" or "squished" a wavelet is can be defined by its scale, which can also be referred to as its dilation. This characteristic is connected to frequency in the sense that it is understood for waves. The wavelet's position in time can be determined based on its location (or space).



$$-(x-b)e^{\frac{-(x-b)^2/(2a^2)}{\sqrt{2\pi}a^3}}$$

First derivative of Gaussian Function

The magnitude of the wavelet can be calculated by examining the value of the parameter labelled "a" in the preceding expression. When the value is decreased, the wavelet will take on an appearance that is more compressed. As a direct result of this, it is now possible to acquire information at high frequencies. In contrast, increasing the value of "a" will cause the wavelet to stretch, which will lead to the collection of information at low frequencies. The value of the "b" parameter is what decides where the wavelet is positioned in the image. When the value of "b" is decreased, the wavelet will shift to the left. Increasing the value of "b" will cause it to relocate to the right. In contrast to waves, wavelets only exhibit non-zero behaviour for a small period of time during which their locations are meaningful. This difference in behavior between wavelets and waves is called the wavelet scale. When we perform an analysis of a signal, in addition to being interested in the oscillations that the signal displays, we are also interested in the locations of those oscillations.

The fundamental concept here is to determine the proportion of a wavelet that exists in a signal at a specific scale and location. For those of you who are familiar with convolutions, this is a perfect example. A signal is convolved with a set of wavelets operating at a range of different scales. We go with a wavelet that has a specified scale. After that, we multiply the wavelet and the signal at each time step, and then we slide this wavelet across the entire signal, which means we change where it is located. The result of performing this multiplication provides us a coefficient that corresponds to that wavelet scale at that time step. After that, the wavelet scale is increased, and the procedure is carried out again.

Based on previous explanation, we understood that wavelets are functions that are concentrated in time and frequency around a certain location.

Generally, got confused for waves and wavelets but they are different the fundamental difference between the two is that a wave is an oscillating function of time or space that is periodic. The wave is an infinite length continuous function in time or space. In contrast, wavelets are localised waves. A wavelet is a waveform of an effectively limited duration that has an average value of zero.

A function () can be called a wavelet if it posses the following properties:

1. The function integrates to zero, or equivalently its Fourier transform denoted as $\psi(\omega)$ is zero at the onigin:

$$\int_{-\infty}^{\infty} \psi(x)dx = 0 \tag{12a}$$

This implies $\psi(\omega)|_{\omega=0}$ =0 in the frequency domain.

2. It is square integrable, or equivalently, has finite energy:

$$\int_{-\infty}^{\infty} |\psi(x)|^2 dx < \infty \tag{12b}$$

3. The Fourier transform $\psi(x)$ must satisfy the admissibility condition given by

$$C_\psi \int_{-\infty}^{\infty} = \frac{|\psi(\omega)|^2}{|\omega|} d\omega < \infty \tag{12c}$$

Interpretation of Eqs. (12a), (12b) and (12c)

Equation (12a) suggests that the function is either oscillatory or has a wavey appearance.Equation (12b) implies that most of the energy in $\psi(x)$ is confined to a finite interval, or in other words. $\psi(x)$ has good space localisation. Ideally, the function is exactly zero outside the finite interval. This implies that the function is a compactly supported function. Equation (12c) is useful in formulating the inverse wavelet transform. From Eq. (12c), it is obvious that in $\psi(x)$ must have a sufficient decay in frequency. This means that the Fourier transform of a wavelet is localized, that is, a wavelet mostly contains frequencies from a certain frequency band. Since the Fourier transform is zero at the origin, and the spectrum decays at high frequencies, a wavelet has a bandpass characteristic. Thus a wavelet is a 'small wave' that exhibits good time-frequency localisation. A family of wavelets can be generated by dilating and translating the mother wavelet in $\psi(x)$ which is given by

$$\psi_{(a,\ b)}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right) \tag{12d}$$

Here, a is the scale parameter and b is the shift parameter.

After understanding the concept of wavelets now it's time to begin with our topic of Wavelet Transform.

Wavelet transforms can be either continuous or discrete, depending on how they are implemented. The Continuous Wavelet Transform (CWT) uses every wavelet that is feasible across a range of scales and places, meaning that it has an endless number of scales and locations to choose from. This is the primary distinction between these two types. While the Discrete Wavelet Transform (DWT) uses a limited number of wavelets, which are defined according to a specific range of sizes and locations, this set of wavelets is not limited in any way. Few ore comparisons between CWT and DWT are given below"

	CWT- Continuous Wavelet Transform	DWT- Discrete Wavelet Transform
1 Scale	At any scale	Dyadic scales
2. Translation	At any point	Integer point
3. Wavelet	Any wavelet that satisfies minimum criteria	Orthogonal, biorthogonal, . . .
4. Computation	Large	Small
5. Detection	Easily detects direction, orientation	Cannot detect minute object if not finely tuned
6. Application	Object Detection, Pattern recognition Feature extraction	Compression, De-noising, Transmission Characterisation

So, the wavelet transform can be broadly classified into (i) continuous wavelet transform, and (ii) discrete wavelet transform. For long signals, continuous wavelet transform can be time consuming since it needs to integrate over all times. To overcome the time complexity, discrete wavelet transform was introduced. Discrete wavelet transforms can be implemented through sub-band coding. The DWT is useful in image processing because it can simultaneously localise signals in time and scale, whereas the DFT or DCT can localise signals only in the frequency domain.

It is to be noted that apart from image processing, the DWT is quite promising tool for the Signal processing also. After the suggestion of Mallat's, that signals may be represented at several resolutions using wavelet decomposition, Discrete Wavelet Transform (DWT) emerged as an extremely flexible tool for signal processing. Because the energy of wavelets is concentrated in time while still retaining the wave-like (periodic) characteristics, we discovered that wavelets make it possible to perform time and frequency analysis of signals at the same time. This was one of the key takeaways from the investigation into wavelets. As a consequence of this, wavelet representation offers a flexible mathematical tool for the analysis of transient, time-variant (non-stationary), signals that are not statistically predictable, particularly in the region of discontinuities. This quality is characteristic of images that have discontinuities at the edges. In DWT, a digital signal splits up into its component sub-bands, so that the lower frequency sub-bands have finer frequency resolution and coarser time resolution compared to the higher frequency sub-bands. .

The wavelet transformation technique overcomes the limitations of the Fourier method. The Fourier transformation, despite the fact that it deals with frequencies, does not reveal any facts regarding the passage of time. In accordance with the Heisenberg's Uncertainty Principle , we can either have a high frequency resolution but a low temporal resolution, or vice versa. The introduction to the Heisenberg's Uncertainty Principle is given below:

The Heisenberg uncertainty principle was originally stated in physics, and claims that it impossible to know both the position and momentum of a particle simultaneously. However, it has an analog basis in signal processing. In terms of signals, the Heisenberg uncertainty principle is given by the rule that it is impossible to know both the frequency and time at which they occur. The time and frequency domains are complimentary. If one is local, the other is global. Formally, the uncertainty principle is expressed as

$$(\Delta t)^2(\Delta\omega)^2 \geq \frac{1}{4}$$

In the case of an impulse signal, which assumes a constant value for a brief period of time, the frequency spectrum is infinite; whereas in the case of a step signal which extends over infinite time, its frequency spectrum is a single vertical line. This fact shows that we can always localise a signal in time or in frequency but not both simultaneously. If a signal has a short duration, its band of frequency is wide and vice versa.

The Wavelet Transform offers a number of important benefits, including the following:

- The Wavelet transform has the ability to concurrently extract local spectral and temporal information.
- A selection of different wavelets from a variety to choose

The first significant benefit is one that we have gone over in some detail already. This is most likely the most important advantage of utilising the Wavelet Transform. This may be preferable to employing a method such as a Short-Time Fourier Transform, which needs slicing a signal into segments and then applying a Fourier Transform to each individual segment.

The second essential benefit appears to be more of a logistical consideration. In the end, the most important thing to take away from this is that there is a large variety of wavelets from which to choose in order to get the one that most closely matches the characteristic shape that you are seeking to extract from your signal.

In comparison to the Fourier Transform, the Wavelet Transform has the primary benefit of being able to extract local information that is both spectral and temporal in nature. Analyzing electrocardiogram (ECG) readings, which comprise periodic and transient signals of relevance, is an example of a real-world use of the Wavelet Transform.

As a result, we realised that non-stationary signals are the ideal candidates for the use of the wavelet transform. By applying this transformation, one can obtain a high temporal resolution for high-frequency components while maintaining a decent frequency resolution for low-frequency components. This technique begins with a mother wavelet, which could be a Haar, Morlet, or Daubechies, among other options. After that, the signal is essentially recast as scaled and shifted iterations of the mother wavelet. **We will discuss Haar transformation in the subsequent section 5.5 of this unit**

Important points:

- The wavelet transform is used to decompose a time series; this results in waves that are not only localised in frequency but also in time.
- One of the most significant drawbacks of the Fourier Transform is that it collects global frequency information, which refers to frequencies that are present throughout an entire signal. There are some applications, such as electrocardiography (ECG), in which the signals include brief intervals of distinctive oscillation, that this form of signal decomposition would not suit very well. The Wavelet Transform is an alternate method that may be used, and what it does is it decomposes a function into a group of wavelets.
- A simple comparison between Wavelet Transform and Fourier Transform: The Fourier transform can be thought of as a special case of the wavelet transform. A function is decomposed by the Fourier transform into sine and cosine waves, which serve as the base functions. Although the period lengths of the sine and cosine waves change, the base functions remain the same across the entire interval. The wavelet transform, on the other hand, applies scaling as well as shifting to the basic functions. Also, the sine and cosine waves are not required to be the base functions, despite the fact that there are some well-known base functions. Instead, we are free to choose whichever functions we want to use as base functions, so long as they fulfills the fundamental condition of a wavelet, which is that it has a finite amount of energy.

5.5 HAAR TRANSFORM

The Haar transform is a wavelet transform. Wavelet transforms are based on small waves called wavelets which are of varying frequencies and limited duration. These are different from the Fourier transform, where the basis functions are sinusoids. Haar transform is a transform whose basis functions are orthonormal wavelets. The Haar transform can be expressed as

$$T = HFH^T \tag{13}$$

where, F is an $N \times N$ image matrix, H is the $N \times N$ Haar transform matrix and T is the resulting $N \times N$ transform.

The Haar transform, H, contains the Haar basis functions, $h_k(t)$. They are defined on a continuous interval, $t \in [0,1]$ for
$k = 0,1,\ldots,N-1$, where $N = 2^n$. Then, H is generated by uniquely

decomposing the integer k as $k = 2^p + q - 1$, where, $0 \le p \le n-1$ and when $p = 0, q = 0,1; p \ne 0$ then, $1 \le q \le 2^n$.

For example, when $N = 4$, k will take the values $k = 0,1,2,3$. For these the corresponding values of p and q have to satisfy that $k = 2^p + q - 1$.

Therefore, we compute the values of k, p and q in Table 1.

Table 1

k	0	1	2	3
p	0	0	1	1
q	0	1	1	2

Let t take the values from the set $\left\{\frac{0}{N}, \frac{1}{N}, \ldots, \frac{N-1}{N}\right\}$.

Then, the Haar basis functions are recursively defined as:

- For $k = 0$, the Haar function is defined as a constant

$$h_0(t) = 1/\sqrt{N} \quad (14)$$

- When $k > 0$, the Haar function is defined by

$$h_k(t) = \frac{1}{\sqrt{N}} \begin{cases} 2^{p/2} & ; \quad \frac{(q-1)}{2^p} \le t < \frac{(q-0.5)}{2^p} \\ -2^{p/2} & ; \quad \frac{(q-0.5)}{2^p} \le t < \frac{q}{2^p} \\ 0 & ; \quad \text{othewise} \end{cases} \quad (15)$$

As can be seen from the definition of the Haar basis functions, for the non-zero part of the function, the amplitude and width is determined by p while its position is determined by q.

We now show how the Haar transform matrix can be computed at $n = m/N$, where $n = 0,1,\ldots,N-1$ to form the $N \times N$ discrete Haar transform matrix through the following examples.

Example 5: For, $N = 2$, compute the discrete Haar transform of a 2×2 matrix.

Solution: Here, $N = 2$, we know that $N = 2^n$.

Substituting the value of N, we get $2 = 2^n$, which gives $n = 1$.

Since, $0 \le p \le n-1$, we get $0 \le p \le 0$.

Therefore, $p = 0$, and hence $q = 0, 1, 2$.

We determine the value of k using the relation $k = 2^p + q - 1$, we obtain

p	0	0
q	0	1
k	0	1

for $k = 0, h_0(t) = \frac{1}{\sqrt{2}}$ [using Eqn. (14)]

for $k = 1, h_1(t) = \frac{1}{\sqrt{2}}\begin{cases} 1 & ; t = 0 \\ -1 & ; t = 1/2 \end{cases}$

Thus, Haar transform is $h_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

***

Example 6: For $N = 8$, the 8×8 discrete Haar transform matrix.

Solution: As you know we need to find various parameters of Haar transform. So, we find them as follows:

i) Here $N = 8$,

ii) $N = 2^n \Rightarrow n = 3$

iii) when $p = 0, q = 0, 1$
$p = 1, q = 1, 2$
$p = 2, q = 1, 2, 3, 4$

iv) All the possible values of k for each set of p and q are given below:

p	0	0	1	1	2	2	2	2
q	0	1	1	2	1	2	3	4
k	0	1	2	3	4	5	6	7

v) Accordingly, $t = \left\{0, \frac{1}{8}, \frac{2}{8}, \frac{3}{8}, \frac{4}{8}, \frac{5}{8}, \frac{6}{8}, \frac{7}{8}\right\}$.

Now, we compute $h_k(t)$ for each k and t.

For $k = 0$, $h_0(t) = 0$. for all t.

In general, $h_k(t) = \frac{1}{\sqrt{8}}\begin{cases} 2^{p/2} ; & \frac{q-1}{2^p} \le t < \frac{q-0.5}{2^p} \\ -2^{p/2} ; & \frac{q-0.5}{2^p} \le t < \frac{q}{2^p} \\ 0 ; & \text{otherwise} \end{cases}$.

(16)

Now, let us find each $h_k(t)$ for each of the interval of t for a particular k using Eqn. (16) in the following table:

For $k = 1$

Parameters k, q, p	$h_k(t)$	Haar Transform after simplification
$k=1$, $q=1$, $p=0$	$h_1(t)=\frac{1}{\sqrt{8}}\begin{cases}1; & \frac{0}{2^0}\le t<\frac{1-0.5}{2}\Rightarrow 1\le t<\frac{1}{2}\\ -1; & \frac{1-0.5}{2}\le t<\frac{1}{2^0}\Rightarrow\frac{1}{2}\le t<1\\ 0; & \text{otherwise}\end{cases}$	$h_1(t)=\frac{1}{2\sqrt{2}}$; for $t=0,\frac{1}{8},\frac{2}{8},\frac{3}{8}$ $h_1(t)=\frac{-1}{2\sqrt{2}}$; for $t=\frac{4}{8},\frac{5}{8},\frac{6}{8},\frac{7}{8}$
$k=2$, $q=1$, $p=1$	$h_2(t)=\frac{1}{\sqrt{8}}\begin{cases}\sqrt{2} & ; 0\le t<\frac{1}{4}\\ -\sqrt{2} & ;\frac{1}{4}\le t<\frac{1}{2}\\ 0 & ;\text{otherwise}\end{cases}$	$h_2(t)=\frac{1}{2}$; for $t=0,\frac{1}{8}$ $h_2(t)=\frac{-1}{2}$; for $t=\frac{2}{8},\frac{3}{8}$ $h_2(t)=0$; for $t=\frac{4}{8},\frac{5}{8},\frac{6}{8},\frac{7}{8}$.
$k=3$, $q=2$, $p=1$	$h_3(t)=\frac{1}{\sqrt{8}}\begin{cases}\sqrt{2} & ;\frac{1}{2}\le t<\frac{3}{4}\\ -\sqrt{2} & ;\frac{3}{4}\le t<1\\ 0 & ;\text{otherwise}\end{cases}$	$h_3(t)=\frac{1}{2}$; for $t=\frac{4}{8},\frac{5}{8}$ $h_3(t)=\frac{-1}{2}$; for $t=\frac{6}{8},\frac{7}{8}$ $h_3(t)=0$; for $t=0,\frac{1}{8},\frac{2}{8},\frac{3}{8}$.
$k=4$, $q=1$, $p=2$	$h_4(t)=\frac{1}{\sqrt{8}}\begin{cases}2 & ; 0\le t<\frac{1}{8}\\ -2 & ;\frac{1}{8}\le t<\frac{1}{4}\\ 0 & ;\text{otherwise}\end{cases}$	$h_4(t)=\frac{1}{\sqrt{2}}$; for $t=0$ $h_4(t)=\frac{-1}{\sqrt{2}}$; for $t=\frac{1}{8}$ $h_4(t)=0$; for $t=\frac{2}{8},\frac{3}{8},\frac{4}{8},\frac{5}{8},\frac{6}{8},\frac{7}{8}$

$k=-5,$ $q=2,$ $p=2$	$h_5(t)=\frac{1}{\sqrt{8}}\begin{cases} 2\,;\ \frac{1}{4}\le t<\frac{3}{8} \\ -2\,;\ \frac{3}{8}\le t<\frac{1}{2} \\ 0\,;\ \text{otherwise} \end{cases}$	$h_5(t)=\frac{1}{\sqrt{2}};\text{for } t=\frac{2}{8}.$ $h_5(t)=\frac{-1}{\sqrt{2}};\text{for } t=\frac{3}{8}$ $h_5(t)=0;\text{ for}$ $t=0,\frac{1}{8},\frac{4}{8},\frac{5}{8},\frac{6}{8},\frac{7}{8}$
$k=6,$ $q=3,$ $p=2$	$h_6(t)=\frac{1}{\sqrt{8}}\begin{cases} 2\,;\frac{1}{2}\le t<\frac{5}{8} \\ -2\,;\frac{5}{8}\le t<\frac{3}{4} \\ 0\,;\ \text{otherwise} \end{cases}$	$h_6(t)=\frac{1}{\sqrt{2}};\text{for } t=\frac{4}{8}$ $h_6(t)=\frac{-1}{\sqrt{2}};\text{for}$ $t=\frac{5}{8}\ h_6(t)=0;\text{ for}$ $t=0,\frac{1}{8},\frac{2}{8},\frac{3}{8},\frac{6}{8},\frac{7}{8}$
$k=7,$ $q=4,$ $p=2$	$h_7(t)=\frac{1}{\sqrt{8}}\begin{cases} 2\,;\frac{3}{4}\le t<\frac{7}{8} \\ -2\,;\frac{7}{8}\le t<1 \\ 0\,;\ \text{otherwise} \end{cases}$	$h_7(t)=\frac{1}{\sqrt{2}};\text{for } t=\frac{6}{8}$ $h_7(t)=\frac{-1}{\sqrt{2}};\text{for } t=\frac{7}{8}$ $h_7(t)=0;\text{ for}$ $t=0,\frac{1}{8},\frac{2}{8},\frac{3}{8},\frac{4}{8},\frac{5}{8}.$

The Haar transform is given in the following matrix.

$$h_k(t)=\begin{bmatrix} \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} \\ \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{-1}{2\sqrt{2}} & \frac{-1}{2\sqrt{2}} & \frac{-1}{2\sqrt{2}} & \frac{-1}{2\sqrt{2}} \\ \frac{1}{2} & \frac{1}{2} & \frac{-1}{2} & \frac{-1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & \frac{-1}{2} & \frac{-1}{2} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}$$

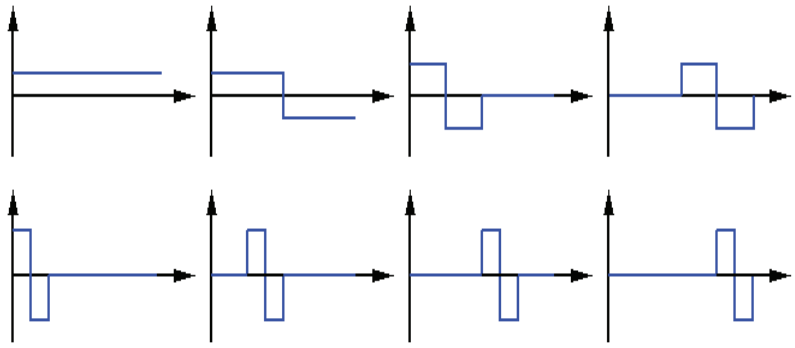


Fig. 6: Haar Basis Functions

The plot of these 8 basis functions are shown in Fig. 6.

As can be seen by Fig.6, all non-zero Haar functions $h_k(t), k > 0$ consists of a square wave and its negative version, and the parametersp defines the magnitude and width of the shape while q specifies the position (or shift) of the shape. This gives the unique property to the Haar transform that it not only represents the signal at different scales based on the different frequencies, but also represents their locations across time.

Moreover, an important property of the Haar transform matrix is that it is real and orthogonal, that is, $H = H^*$ and $H^{-1} = H^T$. The orthogonal property of the Haar transform allows the analysis of the frequency components of the input signal. The Haar transform can also be used for analyzing the localized feature of the signal.

a) Original Image **b) Output of Haar Transform**

Fig. 7: Haar Transform

Fig. 7 (b) shows the output of Haar Transform of the image in Fig. 7 (a).

Try the following exercises.

E7) Let $X = [x[0], x[1], x[2], x[3]]^T = [1, 2, 3, 4]^T$. Then, X is a 4-point signal. Find the Haar transform coefficients and show that the signal can be expressed as a linear combination of the basis functions by the inverse transform.

E8) For $N = 4$, compute h_4, which represents the 4×4 discrete Haar transform matrix.

Now let us, summarize what we have discussed in this unit.

5.6 SUMMARY

In this unit, we discussed transformations which convert the spatial domain image to the frequency domain. We saw that these transform provide a variety of information based on the frequency content of the image.We discussed in depth three very important image transforms, namely the Discrete Fourier transformation (DFT), the Discrete Cosine Transformation (DCT) and the Haar transform. We also discussed the properties of each of these transforms, which shall help us in using them for image filtering in the frequency domain.

5.7 SOLUTIONS AND ANSWERS

E1) Consider an image $f(x, y)$ of size $M \times N$ and the generic image transform T where, x indicates row and y indicates column.

Then,

$$g(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} T(u, x, v, y) f(x, y)$$

If, T is separable and symmetric, then we can write $g(u, v)$ as

$$g(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} T_1(u, x) T_2(v, y) f(x, y)$$

$$\Rightarrow g(u, v) = \sum_{x=0}^{M-1} T_1(u, x) \sum_{y=0}^{N-1} T_2(v, y) f(x, y)$$

Then, $\sum_{y=0}^{N-1} T_2(v, y) f(x, y)$ is the same as applying the one-dimensional transform long the x -th row of the image. By doing this for each of the M rows of the image, we obtain an intermediate image $F(x, v)$.

Then, $g(u, v) = \sum_{x=0}^{M-1} T_1(u, x) F(x, v)$

Therefore, we can see, the above sum corresponds to applying the one dimensional transform along the v -th column of the intermediate image (x, v).

Therefore, we have shown that the implementation of a separable and symmetric transform in an image requires the sequential implementation of the corresponding one-dimensional transform row-by-row and then column-by-column (or the inverse). We explain this in the Fig. 9:

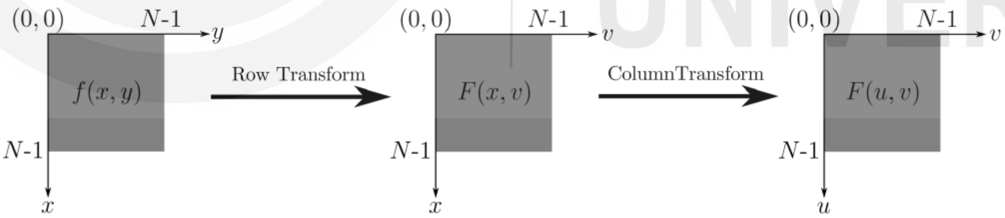


Fig. 9

E2) $$g'(u) = \frac{1}{4} \sum_{x=0}^{3} f(x) e^{\frac{-i \cdot 2\pi ux}{4}}; \quad u = 0, 1, 2, 3$$

$$= \frac{1}{4}[i + i(-i)^{2u} + (-i)^{3u}]; \; u = 0, 1, 2, 3$$

$$= \frac{1}{4}[i + i(-1)^{2u} + (i)^{u}]; \; u = 0, 1, 2, 3$$

$$\mathbf{g} = \frac{1}{4}[1 + 2i, \, i, \, -1 + 2i, \, -i].$$

E3) Here $N = 4$.

$$g(u) = \frac{1}{4}\sum_{x=0}^{3} f(x) e^{\frac{-i2\pi \cdot ux}{4}}; \; u = 0, 1, 2, 3$$

$$g(u) = \frac{1}{4}[f(0) + (-i)^{u} f(1) + (-i)^{2u} f(2) + (-i)^{3u} f(3)]; \; u = 0, 1, 2, 3.$$

$$g(0) = \frac{1}{4}[f(0) + f(1) + f(2) + f(3)]$$

$$g(1) = \frac{1}{4}[f(0) - i f(1) + f(2) + i f(3)]$$

$$g(2) = \frac{1}{4}[f(0) - f(1) + f(2) - f(3)]$$

$$g(3) = \frac{1}{4}[f(0) + i f(1) - f(2) + i f(3)]$$

Hence, the DFT matrix is $A = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix}$.

You may check if $A \times A^{*T} = I$.

E4) Using Eqn. (5), $f(x, y) = \dfrac{1}{2 \times 2}\sum_{x=0}^{1}\sum_{y=0}^{1} F(u, v).e^{2\pi i\left(\frac{ux}{2}+\frac{vy}{2}\right)}; u, v = 0,1$

$$= \frac{1}{4}\sum_{x=0}^{1}\sum_{y=0}^{1} F(u, v)(1)^{ux}(1)^{vy}; u, v = 0,1$$

$$= \frac{1}{4}\left[f(0,0) + (1)^{v} F(0,1) + (1)^{u} F(1,0) + (1)^{u}(1)^{v} F(1,1)\right]; u, v = 0,1$$

which gives $f(x, y) = \frac{1}{4}\begin{bmatrix} 4 & 4 \\ 4 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$.

E5) In general, in most of the images, large part of the signal energy lies at the low frequencies which appear in the upper left corner of the DCT image. Since the higher frequencies present in the lower right of the image are small enough to be neglected, the original image can be represented in less number of coefficients, thereby achieving compression. Therefore, as DCT has good compaction property, it can represent the original image in less number of coefficients and therefore, storage and transmission of the image is better and faster. Moreover, the original image can be recreated close to the original from the most important components of the DCT.

E6) $$C(u) = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}$$

E7) $X = [x[0], x[1], x[2], x[3]]^T = [1, 2, 3, 4]^T$ be the 4-point signal. Then, we shall use the basis matrix, H_4 to compute the Haar transform coefficients.

$$\frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}\begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 5 \\ -2 \\ -1\sqrt{2} \\ -1\sqrt{2} \end{bmatrix}$$

The inverse transform will be:

$$\frac{1}{2}\begin{bmatrix} 1 & 1 & \sqrt{2} & 0 \\ 1 & 1 & -\sqrt{2} & 0 \\ 1 & -1 & 0 & \sqrt{2} \\ 1 & -1 & 0 & -\sqrt{2} \end{bmatrix}\begin{bmatrix} 5 \\ -2 \\ -1\sqrt{2} \\ -1\sqrt{2} \end{bmatrix}$$

$$= \frac{1}{2}\left[5\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} - 2\begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} - \frac{1}{\sqrt{2}}\begin{bmatrix} \sqrt{2} \\ -\sqrt{2} \\ 0 \\ 0 \end{bmatrix} - \frac{1}{\sqrt{2}}\begin{bmatrix} 0 \\ 0 \\ \sqrt{2} \\ -\sqrt{2} \end{bmatrix}\right] = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

As can be seen, X is a linear combination of the basis vectors.

E8) Here $N = 4; n = 2; p = 0, 1; t = \left\{0, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}\right\}$. Let us write all the values of Haar transform in the following table:

k, q, p	$h_k(t)$	$h_k(t)$ after simplification
$p = 0, q = 0, k = 0$	$h_0(t) = \frac{1}{\sqrt{4}} = \frac{1}{2}$ for all t	$h_0(t) = \frac{1}{2}$ for $t = 0, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}$.
$p = 0, q = 1, k = 1$	$h_1(t) = \frac{1}{2}\begin{cases} 1; 0 \le t < \frac{1}{2} \\ -1; \frac{1}{2} \le t < 1 \\ 0; \text{otherwise} \end{cases}$	$h_1(t) = \frac{1}{2}; t = 0, \frac{1}{4}$ $h_1(t) = \frac{-1}{2}; t = \frac{2}{4}, \frac{3}{4}$

$p=1, q=1, k=2$	$h_2(t)=\frac{1}{2}\begin{cases}\sqrt{2}; 0 \le t < \frac{1}{4} \\ -\sqrt{2}; \frac{1}{4} \le t < \frac{1}{2} \\ 0; \text{otherwise}\end{cases}$	$h_2(t)=\frac{1}{\sqrt{2}}; t=0$ $h_2(t)=\frac{-1}{\sqrt{2}}; t=\frac{1}{4}$ $h_2(t)=0; t=\frac{2}{4},\frac{3}{4}.$
$p=1, q=2, k=3$	$h_3(t)=\frac{1}{2}\begin{cases}\sqrt{2}; \frac{1}{2} \le t < \frac{3}{4} \\ -\sqrt{2}; \frac{3}{4} \le t < 1 \\ 0; \text{otherwise}\end{cases}$	$h_3(t)=\frac{1}{\sqrt{2}}; t=\frac{2}{4}$ $h_3(t)=\frac{1}{\sqrt{2}}; t=\frac{3}{4}$ $h_3(t)=0; t=0,\frac{1}{4}.$

Hence,

$$h_k(t)=\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{-1}{2} & \frac{-1}{1} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}$$

UNIT 6 IMAGE ENHANCEMENT AND FILTERING IN FREQUENCY DOMAIN

Structure **Page No.**

6.1	Introduction Objectives	137
6.2	Shifting the Centre of the Spectrum	138
6.3	Image Smoothing in Frequency Domain	143
6.4	Image Sharpening in Frequency Domain	151
6.5	Image Degradation	155
6.6	Image Degradation/Restoration Model	157
6.7	Noise Models	158
6.8	Restoration in the Presence of Noise Only – Spatial Filtering	164
6.9	Periodic Noise Reduction	169
6.10	Estimation of Degradation Function	172
6.11	Inverse filtering	175
6.12	Wiener filtering	178
6.13	Summary	182
6.14	Solutions/Answers	182

6.1 INTRODUCTION

In the previous units of this course, we have considered an image in the spatial domain which is the form in which camera captures it. In this unit, we shall view the image as a signal and apply the well known filtering techniques used in signal processing. The only difference here will be that image would be considered as 2-D signal (along x and y axes). We shall see that this view of the image results in number of benefits over the spatial domain treatment.

In this unit, we will discuss various enhancement techniques in frequency (fourier) domain. We discuss the basic issues associated with frequency domain filtering. We also discuss various low pass and high pass filters in frequency domain with their applications and advantages in image enhancement.
By discussing the enhancement techniques in frequency (fourier) domain, we have looked at image improvement without bothering about source which caused a degradation in the quality of the image. If the source is known to us, it is possible to improve the quality of the image in a better way. Thus, it is required to discuss the concept of Image restoration/degradation.

Image restoration is a pre-processing method that suppresses a known **degradation**. Image acquisition devices introduce degradation because of defects in optical lenses, non- linearity of sensors, relative object camera motion, blur due to camera mis-focus, atmospheric turbulence etc. Restoration tries to reconstruct an image that was degraded by a known degradation function. Iterative restoration techniques attempt to restore an image by minimizing some parameter of degradation, whereas blind restoration techniques attempt improve the image without knowing the degradation function. Like image enhancement, image restoration also aims to improve image quality, but it is more objective process where as enhancement is a subjective process. Noise is visually unpleasant, it is bad for compression and bad for analysis. Restoration involves modeling of these degradations and applying inverse process to recover the original image.

So we learned that Image restoration is the process of retrieving an original image from degraded image. The idea is to obtain an image as close to the original image as possible. This is possible by removing or minimizing degradations. This is often difficult in case of extreme noise and blurs, and often called an **inverse problem**. An inverse problem aims to find the cause and extent of degradation.

In this unit we will also learn that the image Restoration involves modeling of the degradations and by applying inverse process we can recover the original image. For the restoration process, it is mandatory that we estimate the degradation process accurately. Else we will not be able to remove it.

Now, we shall list the objectives of this unit. After going through the unit, please read this list again and make sure that you have achieved the objectives.

Objectives

After studying this unit, you should be able to

- define images in frequency domain
- perform filtering in frequency domain
- apply different types of image smoothing filters
- apply different types of image sharpening filters
- describe images degradation models
- state difference between restoration and enhancement
- apply different noise models
- estimate degradation function
- apply Inverse filtering
- apply Wiener filtering

Let us begin with shifting the centre of the spectrum.

6.2 BASICS OF FILTERING IN FREQUENCY DOMAIN - SHIFTING THE CENTRE OF THE SPECTRUM

To start with we understood that any signal (periodic or non periodic) can be expressed as the summations of sines and/or cosines multiplied by a weighting function. This is carried out by applying Fourier transform on the image.

In 1D signal, the fourier transform takes the form

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi ux} dx$$

$F(u)$ can be expressed in polar coordinates:

$F(u) = |F(u)| e^{j\phi(u)}$

where $|F(u)| = \left[R^2(u) + I^2(u)\right]^{\frac{1}{2}}$ (magnitude or spectrum)

$$\phi(u) = \tan^{-1}\left[\frac{I(u)}{R(u)}\right] \quad \text{(phase angle or phase spectrum)}$$

$R(u)$: the real part of $F(u)$

$I(u)$: the imaginary part of $F(u)$

The various benefits of frequency domain analysis are the following:

1) It is convenient to design a filter in frequency domain. As filtering is more intuitive in frequency domain, designing an appropriate filter is easier.

2) Implementation is very efficient with fast DFT via FFT.

3) Convolution in spatial domain reduces to multiplication in frequency domain which is a much simpler process.

However, the image in spatial domain is not continuous but consists of discrete values. The discrete version of fourier transform is

$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x) e^{\frac{-2\pi ux}{N}}, u = 0, 1, ..., N-1$$

Also, since the image is two dimensional signal, we need 2D Fourier transform. For a $N \times N$ image it takes the form:

$$F(u, v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x, y) e^{-j2\pi\left(\frac{ux+vy}{N}\right)},$$

where u and v are the frequencies along x and y axes and take the values $0, 1, 2, ..., N-1$.

In the spatial domain we consider the origin to be located at top left corner of the image. For better display in the frequency domain, it is common to shift the origin to centre of the image.

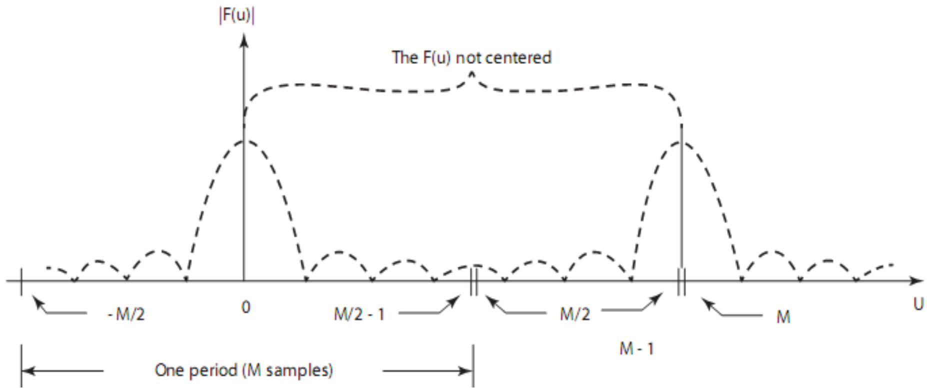
Periodicity of Fourier transform is given by

$$v(k, l) = v(k+M, l) = v(k, l+N) = v(k+M, l+N) \quad (1)$$

$$u(m, n) = u(m+M, n) = u(m, n+N) = u(m+M, n+N) \quad (2)$$

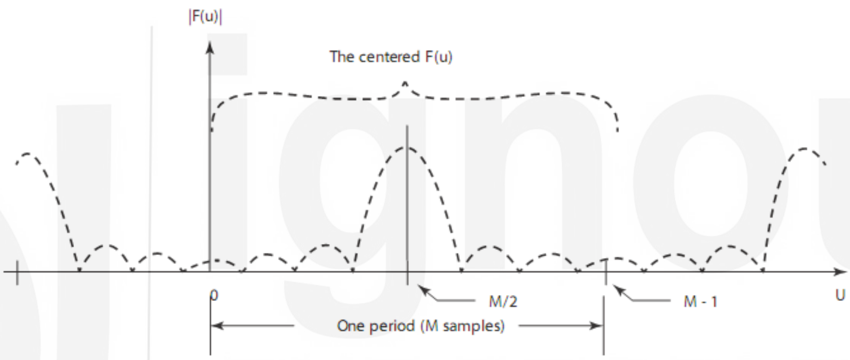
Fig 1(a) shows that the values from $N/2$ to $N-1$ are the same as the value from $N-1$ to 0. As DFT has been formulated for value of k in the interval $[0, N-1]$, the result of this formulation yield two back to back half periods in this interval. To display one full interval between 0 to $N-1$ as shown in Fig. 1(b), it is necessary to shift the origin of transform to the point $k = N/2$. To do so we have to take advantage of translation property of Fourier transform.

$$v(m,n)(-1)^{m-n} \xleftrightarrow{FT} v\left(k-\frac{M}{2}, l-\frac{N}{2}\right)$$



(3)

(a) Spectrum of f(x) without shifting centre.



(b) Spectrum of f(x) after shifting centre.

Fig. 1

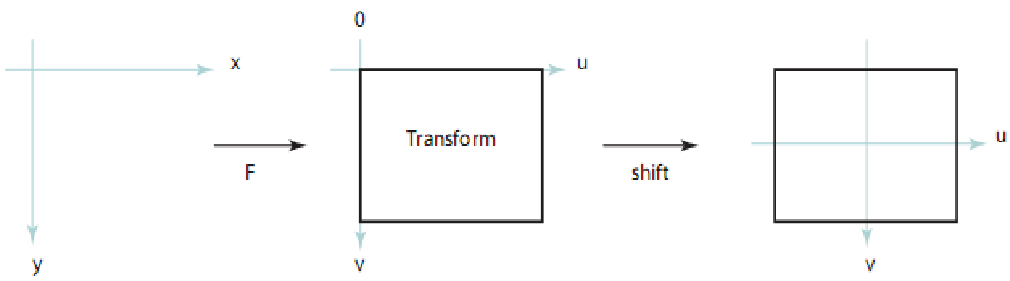
Fig. 2 (a) and (b) show how the origin shifts from left corner of the image to centre of the image.

Basic Property of images in Frequency Domain

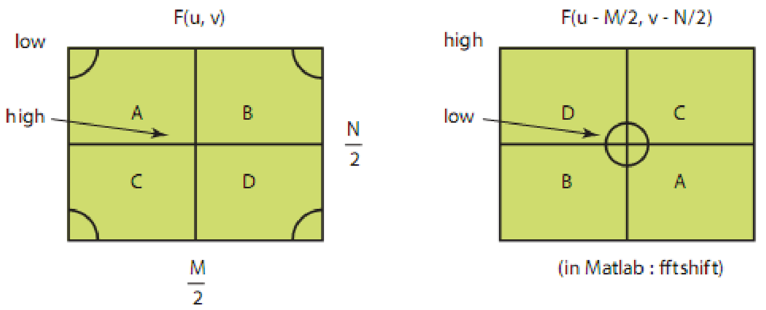
The forward transform of input image $u(m,n)$ is given by

$$v(k,l) = \frac{1}{N}\sum_{m=0}^{N-1}\sum_{m=0}^{N-1} u(m,n)\, W_N^{km}\, W_N^{ln} \qquad 0 \le k,l \le N-1$$

(4)



(a) Change of centre in the spectrum of an image.



(b) Change of centre in the spectrum of an image.

Fig. 2

Following properties of the Fourier transform are observed

i) Each term of $v(k,l)$ contains all the values of $u(m,n)$ modified by the values of exponential terms.
ii) Frequency is directly related to the rate of change of grey level values.
iii) DC value or the average grey level value in an image is the slowest varying components corresponding to $u = 0, v = 0$. It is also the largest component in the frequency domain.
iv) Smooth variation of grey levels corresponds to low frequency components. Slow varying components can be the background of an image, hair of a person, skin, or texture etc.
v) Faster grey level changes correspond to high frequency components. These can be the edges/boundary of the objects or noise present in the image.
vi) As we move away from origin, higher frequencies are encountered.

Fig. 3 shows the variation in frequency of a centred spectrum.

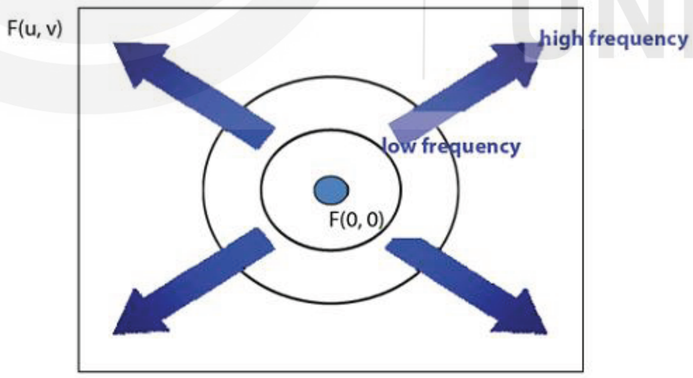


Fig 3: Frequency variation in an image.

Also, note that the rotation of an image in spatial domain causes exactly same rotation in frequency domain.

- Rotating $f(x, y)$ by θ rotates $F(u, v)$ by θ.

Once the image is transformed in frequency domain, it is easy to carry out image processing operations on it. We apply a low pass filter, if we are interested in only slowly varying components of the image (like object shapes), and we wish to suppress high frequency components (like noise). If we are interested in highlighting the edges or special textures, we can employ high pass filters, which will allow high frequency components to be displayed.

Filtering in frequency domain is multiplication of a suitable filter $H(u,v)$ by image in Fourier domain $F(u,v)$ to result in $G(u,v)$. By taking inverse Fourier Transform of $G(u,v)$ we get the image back in spatial domain.

Generally, the filters are centred and are symmetric about the centre. Input image should also be centred. Following steps are followed to carry out filtering in frequency domain (Fig. 4):

Step 1: Multiply input image $f(x,y)$ by $(-1)^{x+y}$ to move the origin in the transformed image to

$$u = \frac{M}{2} \text{ and } v = \frac{N}{2}$$

Step 2: Compute $F(u,v)$, Fourier transform of the output of step 1.
Step 3: Multiply filter function $H(u,v)$ to $F(u,v)$ to get $G(u,v)$.
Step 4: Take inverse Fourier transform of $G(u,v)$ to get $g(x,y)$.
Step 5: Take the real part of $g(x,y)$ to get $g_r(x,y)$
Step 6: Multiply the result of step 5 by $(-1)^{x+y}$ to shift the centre back to origin and enhanced image is generated.

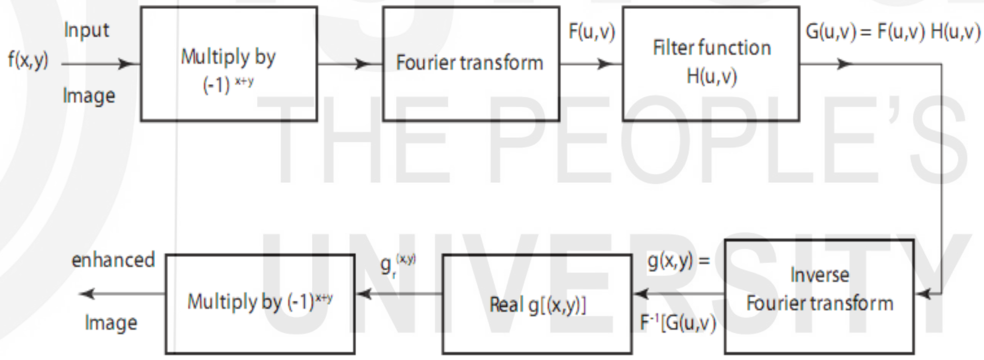


Fig. 4: Block Diagram of Filtering in Frequency Domain.

Types of Frequency Domain Filters

Frequency domain filters are categorized into three types.

1. Smoothing filters
2. Sharpening filters
3. Homomorphic filters

Smoothing filters are low pass filters and are used for noise reduction. It blurs objects. Sharpening filters are high pass filters and produce sharp images with dark background. Laplacian and high boost filters are used to produce sharp images. Homomorphic filters are based on illumination and reflectance model, and create a balance between smoothing and sharpening filtering effect. This classification is shown in Fig. 5.

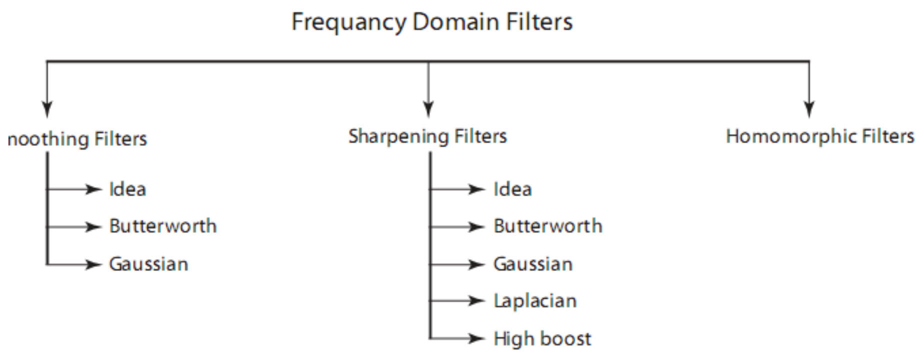


Fig. 5: Types of Frequency Domain Filters.

Try the following exercises.

E1) Write the steps involved in frequency domain filtering with the help of block diagram.

E2) Explain how image enhancement is better in the frequency domain as compared to spatial domain.

In the following section, we will discuss smoothing filters in the frequency domain.

6.3 SMOOTHING FILTERS

Smoothing filters are low pass filters (LPF). Edges, sharp transitions and noise in the grey levels contribute to high frequency contents in an image. A low pass filter only passes low frequency and blocks the high ones. It removes noise but in the process introduces blurring as a side effect in the image.
The basic model of filtering is

$$G(u, v) = H(u, v)\, F(u, v) \qquad (5)$$

where $F(u, v) =$ Fourier transform of the image to the filtered, $H(u, v) =$ Transfer function of the filter, and $G(u, v) =$ Enhanced image where high frequency components have been alternated.

The transfer function $H(u, v)$ is of three types

a) Ideal LPF
b) Butterworth LPF
c) Gaussian LPF

Ideal filters has sharp slope in transition band whereas Gaussian filter has smooth slope in transition. Butterworth filter has a parameter called filter order which controls the slope of transition band. Higher value of filter order leads to ideal filter.

6.3.1 Ideal Low Pass Filters (ILPF)

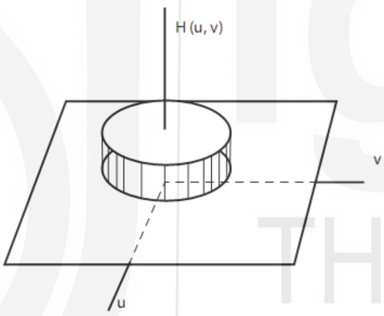
Low pass filter removes all frequencies above a certain frequency components D_0. Ideal low pass filter is defined by the transfer function

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

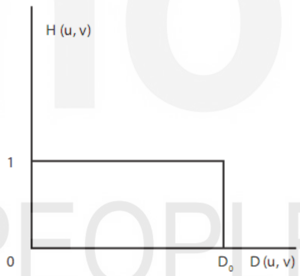
Where $D(u,v) = \left[\left(u - \frac{M}{2}\right)^{1/2} + \left(v - \frac{N}{2}\right)^{1/2}\right]$

$D(u,v)$ is the distance from point (u,v) to the centre $\left(\frac{M}{2}, \frac{N}{2}\right)$. If size of an image is $M \times N$, then the centre is at $\left(\frac{M}{2}, \frac{N}{2}\right)$. Filter transfer function is symmetric about the midpoint.

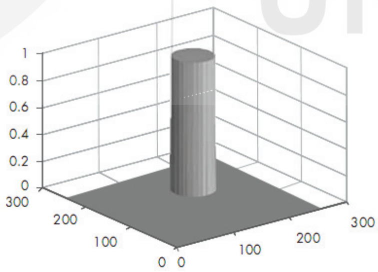
D_0 is non negative quantity specifying the frequency content to be retained. It is also called cut off frequency. Fig. 6(a), Fig. 6(b) is the plot of ILPF and Fig. 6 (c) is perspective plot and Fig. 6(d) is filter displayed as an image.



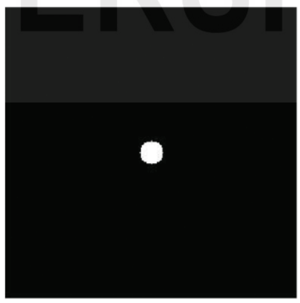
(a) Transfer Function



(b) Plot of Ideal LPF



c) Prospective plot of ILPF



(d) ILPF displayed as an image with $D_0 = 10$

Fig. 6

Choice of cut off frequency in an ideal LPF

1. The cutoff frequency D_0 decides the amount of frequency components passed by the filter.

2. Smaller the value of D_0, more are the number of frequency components eliminated by the filter.

3. In general, D_0 is chosen such that most of the frequency components of interest are passed while unnecessary components are eliminated.

4. A useful way to establish a set of standard cut off frequencies is to compute circles having a certain percentage of the total image power.

$$P_T = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} P(u, v)$$

Here $P(u, v) = |F(u, v)|^2$ is the total image power.

5. Consider a circle of radius $D_0(\alpha)$ as the cut off frequency with respect to a threshold a such that

$$\sum_u \sum_v P(u, v) = \alpha P_T .$$

6. Thus, we can fix a threshold α which tells how much of the total energy is retained and obtain an appropriate cut off frequency $D_0(\alpha)$.

Properties of ILPF

As it is an ideal filter, it is non-real, non-actual and non-physical. But it can be stimulated in computers. Fig 7 and 8(b) to (c) show low pass filtered images with different cut off frequencies. As the filter radius increases, less and less power and information is removed which resulted in less blurring. A very noticeable effect that can be seen in the output image is ringing.

Now, the question arises that what is ringing?

Ringing is undesirable and unpleasant lines around the objects present in the image Fig. 7 (b). As the cut of frequency D_0 increases, effect of ringing reduces. Ringing is a side effect of ideal lpf .

$\uparrow D_0$ ——— $\downarrow$ Ringing + blur

Why is there Ringing in Ideal LPF?

Ideal LPF function is a rectangular function as shown in Fig. 6-X. The inverse Fourier transform of a rectangular function, is a sinc function. We can observe two distinctive characteristic of sin c function:

1. A dominant component at the origin which is responsible for blurring.
2. Concentric circular components are responsible for ringing which is the characteristic of an ideal LPF.

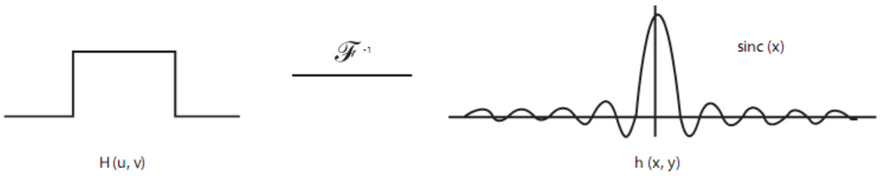
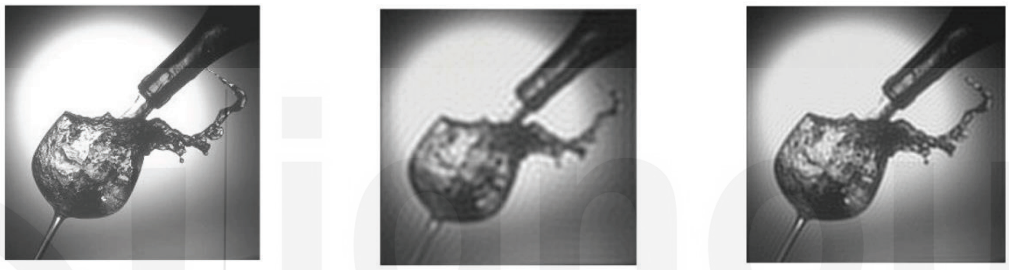


Fig. 6-X: Fourier Inverse of Rectangular Transfer Function

$$\text{Radius of the centre component} \propto \frac{1}{\text{cut off frequency}}$$

$$\text{Number of circles per unit distance from origin} \propto \frac{1}{\text{cut off frequency}}$$

Thus, as the cut off frequency (D_0) is increased, blurring as well as ringing reduces. The examples are given in Fig. 7 and Fig. 8.



(a) Original image **(b) Output of ILPF with D_0=30** **(c) Output of ILPF with D_0= 50**

Fig. 7

(a) Original image **(b) Output of ILPF with D_0=50** **(c) Output of ILPF with D_0=80**

Fig. 8

6.3.2 Butterworth Low Pass Filters (BLPF)

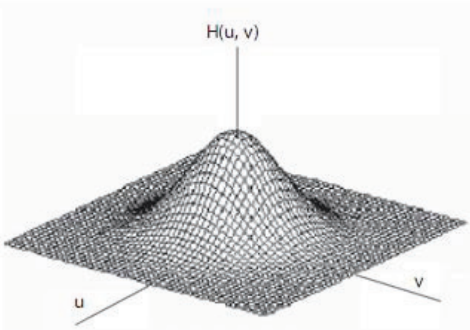
The Butterworth filter replaces the sharp cutoff of Ideal LPF by a smooth cutoff. Frequency response of BLPF does not have a sharp transition between pass band and stop band. It is more appropriate for image smoothing and does not introduce ringing effect for lower order filters. Transfer function of BLPF is given by

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

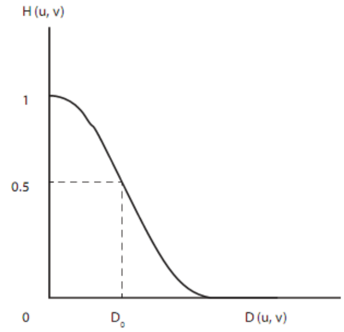
where the cut off frequency or distance from the centre $D_0 = \left(\frac{M}{2}, \frac{N}{2}\right)$, and

the filter order is n, and $D(u, v) = \left[\left(u - \frac{M}{2}\right)^2 + \left(u - \frac{N}{2}\right)^2\right]^{1/2}$

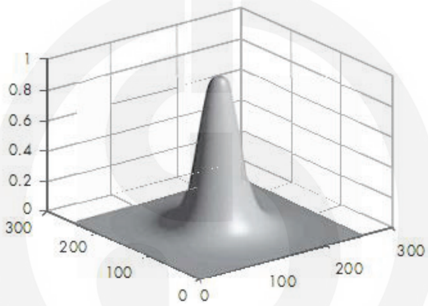
Fig. 9 (a) and Fig. 9 (b) show the transfer function of BLPF. Fig. 9 (c) is the plot of BLPF and Fig. 9 (d) is BLPF displayed as an image.



(a) Plot of BLPF



(b) Plot of BLPF transfer function



(c) Plot of BLPF

(d) BLPF displayed as an image

Fig. 9

Transfer function of BLPF does not have sharp transition near the cut off. For $n = 1$, the transition is very smooth. As the filter order increases, the transfer function approaches towards ideal LPF. No ringing is visible on the image filtered by BLPF for $n = 1$. Noise is reduced and blurring is observed in all the images. For $n = 2$, ringing is un-noticeable, but it can becomes more significant for higher values of n. Fig. 10 shows the increasing effect of ringing as n increases from 1 to 20.

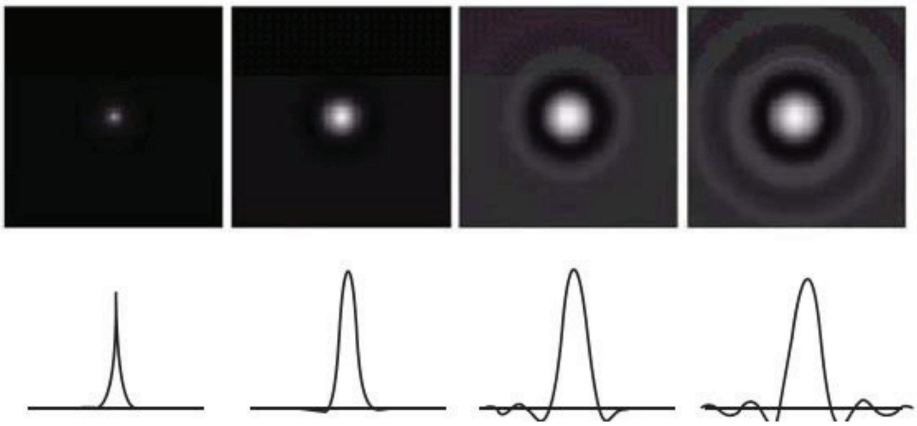


Fig. 10: Spatial Representation of BLPF of order 1, 2, 5 and 20 and Corresponding Intensity Profile

The output corresponding to the change in the values of D_o and n are shown in Fig. 11.

(a) Output of BLPF for D_0= 30

(b) Output of BLPF for D0= 40

(c) Output of BLPF for n = 4, D_0 = 30

(d) Output of BLPF for n = 20 D_0 = 30

Fig. 11

6.3.3 Gaussian Low Pass filters (GLPF)

Still a better variant of the low pass filter is the Gaussian Low Pass filter which have smooth transition between pass band and stop band. It does not introduce any ringing in the output image. The transfer function of GLPF is given by

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

where $D(u,v)$ is the distance from the origin of Fourier transform, and σ is the measure of spread/dispersion of the Gaussian curve.

Larger the value of σ, larger is the cut off frequency and the filter is milder. Let $\sigma = D_0$ then transfer function is given by

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

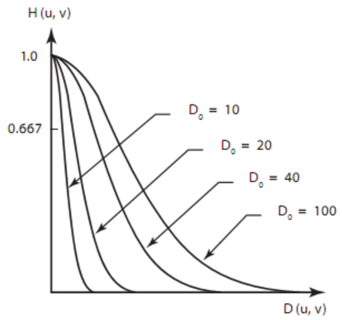
where D_0 is the cut off frequency.

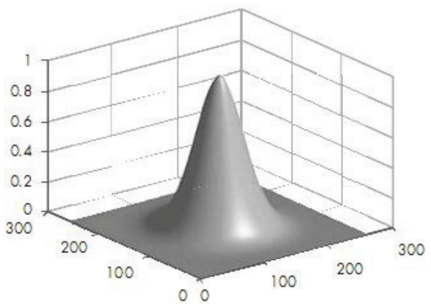
When $D(u,v) = D_0$, the amplitude of transfer function is down to 0.607 of its maximum value of 1.

$D_0 \uparrow$	= Milder filter, more components are passed
$D_0 \uparrow$	= $\downarrow$ blurring

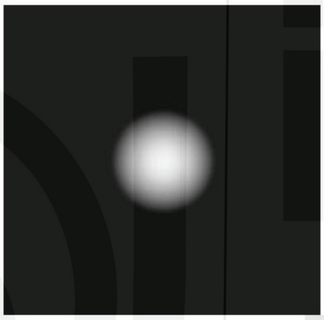
Fig. 12 (a) is GLPF transfer function, Fig. 12 (b) is plot of GLPF and Fig. 12

(c) is GLPF displayed as an image. Fig. 13 (a) to Fig. 13 (c) are GLP filtered images. No ringing is observed in the output, but only blurring is visible. As the cut off frequency increase, blurring reduces. No ringing in the output is a very big advantage of GLPF. These filters can be used in situations where no artifacts are desirable (eg. medical imaging). In medical imaging, GLPF is preferred over ILPF/ BLPF.





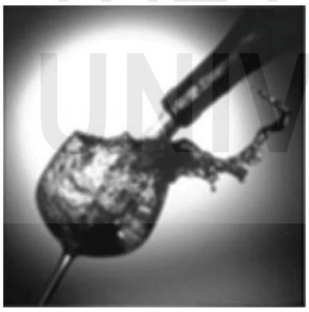
(a) GLPF Transfer Function for Various Values of D **(b) Plot of GLPF**



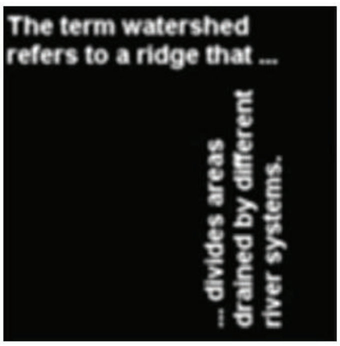
(c) GLPF Displayed as an Image

Fig. 12

(a) Output of GLPF for D_0 = 10



(b) Output of GLPF for D_0 = 300



(c) Output of GLPF with D_0 = 50

Fig. 13

	Ideal	Butter worth	Gaussian
Transfer Function	$H(u,v) = \begin{cases} 1, D(u,v) \le D_0 \\ 0, D(u,v) > D_0 \end{cases}$	$H(u,v) = \dfrac{1}{1+[D(u,v)/D_0]^{24}}$	$H(u,v) = e^{-D^2(u,v)/2D^2}$
Applications	Reduce noise	Reduce noise	Reduce noise
Problems	Blurring Ringing	Blurring, Ringing for higher order filters $(n > 2)$	Blurring, No ringing

Let us discuss some of the applications of Low pass filters in frequency domain.

6.3.4 Applications of Low Pass Filters

LPF are generally used as a preprocessing step before an automatic recognition algorithm. It is also used to reduce noise in images. Few examples are listed below.

1. **Character Recognition:** Input to an automatic character recognition system is generally of poor quality. Input may contain noise due to improper acquisition system or there may be gaps in the alphabets. (broken alphabets). Because of these problems, character recognition system fails to give expected results consistently. Hence, LPF is used as a preprocessing step to blur the image. Blurring is used to bridge small gaps in the alphabets. This is done using GLPF with D_0 = 80. This increases chances of getting correct result from automatic character recognition system.

2. **Object Counting:** Object counting is to count the number of objects in an image. The output of an object counting algorithm may give wrong output because of poor quantity of input image. If there are small gaps in the boundary of objects, automatic algorithm will not give expected results. Blurring is used to fill in small gaps in the boundary of objects which helps in producing correct results.

3. **Printing and Publishing Industry:** Unsharp masking is used in publishing industry to sharpen image where blurred version of an image is subtracted from the image itself to get sharpen image.

4. **"Cosmetic"** processing is another use of low pass filter prior to printing. Blurring is used to reduce the sharpness of fine skin lines and small blemishes on human face. Smoothened images look very soft and pleasing and face looks younger.

Try the following exercises.

E3) Give the formula for transform function of a Butterworth low pass filter.

E4) Explain and compare ideal low pass filter and Butterworth filter for image smoothing.

E5) Explain smoothing frequency domain filters. What is ringing effect?

E6) Discuss the applications of image smoothing filters.

In the following section we will discuss sharpening filters.

6.4 IMAGE SHARPENING IN FREQUENCY DOMAIN

In the Fourier transform of an image, high frequency contents correspond to edges, sharp transition in grey levels and noise. Low frequency contents correspond to uniform or slowly varying grey level values.

High pass filtering is achieved by attenuating low frequency components without disturbing high frequency components. High pass filter (HPF) can also be viewed as reverse operation of low pass filter. Transfer function of HPF is given by

$$H_{hp}(u,v) = 1 - H_{lp}(u,v)$$

Here, $H(u,v)$ is the transfer function of a LPF. Thus

Smoothing $\to$ LPF $\to$ attenuates high frequency components
Sharpening $\to$ HPF $\to$ attenuates low frequency components

Here, we discuss only real and symmetric filters. Following sharpening filters are discussed in this section:

1. Ideal high pass filter
2. Butterworth high pass filter
3. Gaussian high pass filter

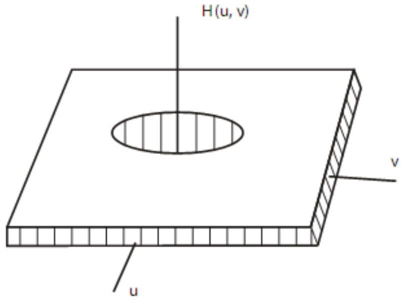
High pass filters are used for enhancing edges. These filters are used to extract edges and noise is enhanced, as a side effect.

6.4.1 Ideal High Pass Filter (IHPF)

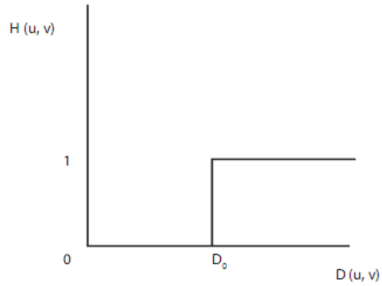
Transfer function of a 2D IHPF is given by

$$H(u,v) = \begin{cases} 1, & \text{if } D(u,v) \geq D_0 \\ 0, & \text{if } D(u,v) < D_0 \end{cases}$$

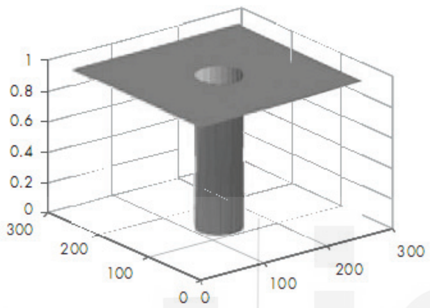
Here, D_0 is the cut off frequency and $D(u,v)$ is the distance from the origin of the Fourier transform. Fig. 14 (a) and Fig. 14 (b) is the IHPF and its transfer function respectively. Fig. 14 (c) is plot of IHPF and Fig. 14 (d) is IHPF as an image. Note that the origin (0,0) is at the centre and not in the corner of the image. The abrupt transition from 1 to 0 of the transfer function $H(u,v)$ cannot be realized in practice. However, the filter can be simulated on a computer. This filter sets to all frequencies inside the circle of radius D and passes all frequencies above D_0 without any attenuation. Ringing is clearly visible in the output (Fig. 15 (b), and Fig. 16(c)) other than sharp edges and boundaries. Output image looks very dark and dull as the high value DC component $G(0,0)$ is eliminated.



(a) Plot of IHPF



(b) Transfer function of IHPF



(c) Plot of IHPF

(d) IHPF displayed as an image

Fig. 14

(a) Output of IHPF for $D_0 = 50$

(b) Output of IHPF for $D_0 = 60$

Fig. 15

6.4.2 Butterworth High Pass Filter (BHPF)

Butterworth filter does not have sharp transition between passband and stop band. The slope depends on the order of the filter. Transfer function of BHPF is

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

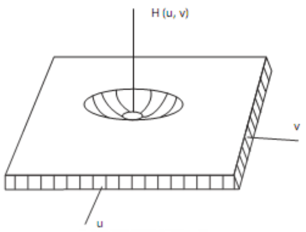
where n is the order of the filter, D_0 is the cut off frequency and $D(u,v)$ is the distance from the origin of Fourier transform.

Fig. 16 (a) and Fig. 16 (b) are BHPF transfer function and Fig. 16 (c) and Fig. 16 (d) are plot and image display of BHPF.

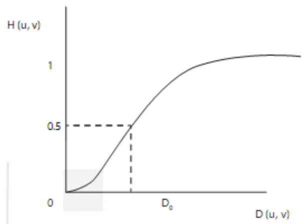
Frequency response does not have a sharp transition as in the ideal HPF.

Thus, less distortion is seen in the output with no ringing effect even for smaller values of cut off frequencies. This filter is more appropriate for image sharpening than ideal HPF as there is no ringing in output.

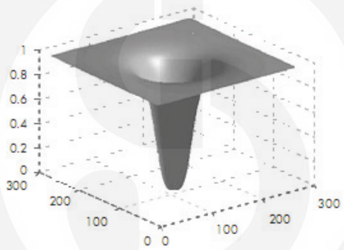
Fig. 16(b) is the plot of GHPF for $D_0 = 30, n = 2,$ and Fig. 16 (c) GHPF displayed as an image. Fig. 17(a) and Fig.17 (b) are the output of GHPF for D_0 = 30 and 130 respectively for $n = 2$. It is clear from the output, as D_0 increases, more and more power is removed from the output image. Thus, output looks sharper for higher value of D_0. Fig. 17(d) is the output for $D_0 = 30\ n = 20,$ ringing is clearly visible in the output. As n increases, ringing in butterworth filter increases.



(a) Transfer function of BHPF



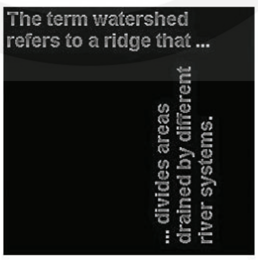
(b) Transfer function of BHPF



(c) Plot of BHPF

(d) BHPF displayed as an image

Fig. 16



(a) Output of BHPF with D_0 = 130, n = 2 (b) Output of BHPF with D_0= 30, n = 2

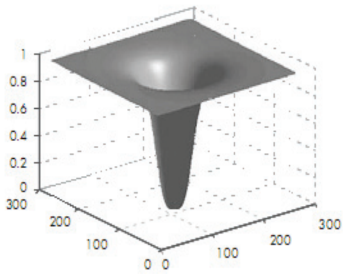
Fig. 17

6.4.3 Gaussian High Pass Filter (GHPF)

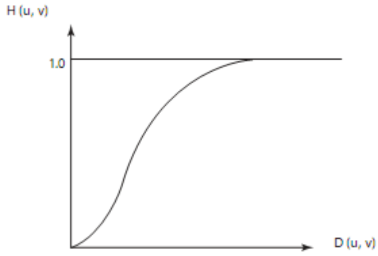
Gaussian high pass filters have smooth transition between passband and stopband near cutoff frequency. The parameter D is a measure of spread of the Gaussian curve. Larger the value D_0, larger is the cut off frequency. Transfer function of GHPF is

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

where D_0 is the cut off frequency and $D(u,v)$ is the distance from origin of Fourier transform.



(a) GHPF transfer function



(b) Plot of GHPF

(c) GHPF displayed as an image

Fig. 18

Fig. 18(a) is GHPF Transfer function filter. Plot and image are displayed in Fig. 18 (b) and Fig. 18 (c). Output in Fig. 19 is much smoother than previous two filters with no ringing effect.

(a) Output of GHPF with D_0=30

(b) Output of GHPF with D_0=120

Fig. 19

Let us compare these three high pass filters in frequency domain filters in the following table.

Table 1

	Ideal	Butterworth	Gaussian
Transfer function	$H(u,v) = \begin{cases} 1, & D(u,v) \le D_0 \\ 0, & D(u,v) > D_0 \end{cases}$	$H(u,v) = \dfrac{1}{1+\left[\dfrac{D}{D(u,v)}\right]^{2n}}$	$H(u,v) = 1 - e\dfrac{D^2(u,v)}{2D_0^{\,2}}$
Application	**Edge enhancement**	**Edge enhancement**	**Edge enhancement**
Problems	**Ringing**	**No Ringing**	**No Ringing**

Try the following exercises.

E7) How many types of high pass filters are there in frequency domain? List them.

E8) Give the formula for transform function of a Gaussian high pass filter.

Now, Its time to discuss the concept of image degradation.

6.5 IMAGE DEGRADATION

Overall our objective is to improve image. For that it is important to understand image degradation if we want to remove it. Degradations are of three types

a) Noise

b) Blur

c) Artifacts

Let us define these one by one.

a) **Noise** is a disturbance that causes fluctuations in pixel values. Pixel values show random variations and can cause very disturbing effects on the image. Thus suitable strategies should be designed to model and remove/ reduce noise. Original image is shown in Fig. 20(a) and noisy image with added Gaussian noise is shown in Fig. 20(b).

(a) Original **(b) Noisy Image**

Fig. 20

b) **Blur** is a degradation that makes image less clear.This makes image analysis and interpretation very difficult. Motion blur is a very common cause of blurring where blur occurs due to the movement of object or camera. Fig. 21 (a) shows original image and Fig. 21 (b) shows blurred image.

(a) Original Image **(b) Blurred image**

Fig. 21

c) **Artifacts** or distortions are extreme intensity or color fluctuations that can make image meaningless. Distortions involve geometric transformations such as translation, rotation or change in scale.

Now, the question arises that what are the sources which contribute to image degradation. Image degradation (as shown in Fig. 22) can happen due to

a) **Sensor Distortions:** Involves quantization, sampling, sensor noise, spectral sensitivity, de-mosaicking, non linearity of sensor etc.

b) **Optical Distortions:** are geometric distortion, blurring due to camera mis-focus.

c) **Atmospheric Distortions:** are haze, turbulence etc.

d) **Other Distortions:** Low illumination, relative motion between object and camera etc.

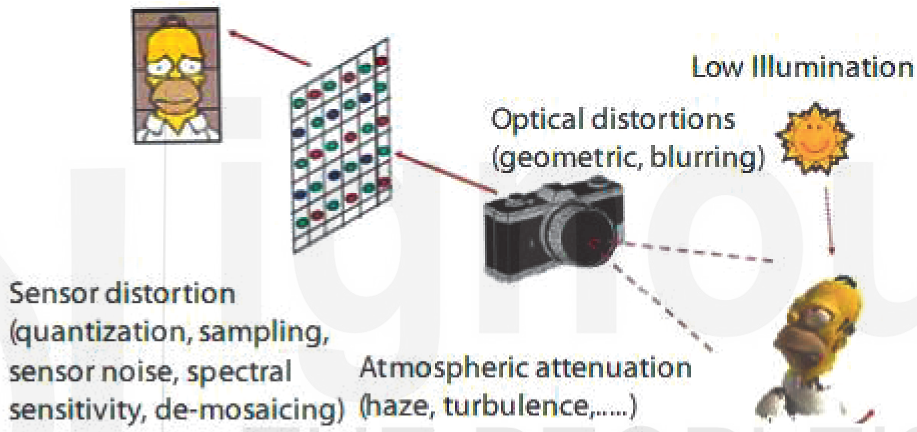


Fig. 22: Typical Degradation Sources

The processes that are used to remove degradation are mainly image enhancement and image restoration.

Restoration is the process of inverting a degradation using knowledge about its nature, whereas enhancement is a process that aims to improve 'bad' quality image so that it looks better. Restoration is distinguished from enhancement, as degradation can be considered as an external influence. Table 2 lists the differences between enhancement and restoration.

Table 2: Enhancement v/s Restoration

	Enhancement	Restoration
1.	It gives better visual representation	It remove effects of sensing environment
2.	No model required	Mathematical model of degradation
3.	It is a subjective process	It is an objective process
4.	Contrast stretching, histogram equalization etc are some enhancement techniques	Inverse filtering, wiener filtering, denoising are some restoration techniques.

Try the following exercises.

E9) What are the factors that can cause image degradation.

In this section we will discuss image degradation/restoration model.

6.6 IMAGE DEGRADATION/RESTORATION MODEL

Consider the block diagram given in Fig. 23 shows the block diagram of degradation/restoration model. Degradation function $h(x,y)$ and noise $n(x,y)$, operate on input image $f(x,y)$ to generate a degraded and noisy image $g(x,y)$.

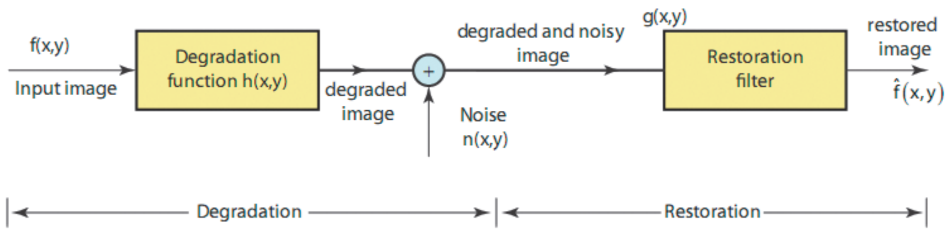


Fig. 23: Block diagram of degradation/restoration model

The notations used in the block diagram are

$f(x,y) =$ original image
$h(x,y) =$ degradation function
$n(x,y) =$ additive noise
$g(x,y) =$ degraded and noisy image
$\hat{f}(x,y) =$ restored image

The objective of restoration process is to estimate $\hat{f}(x,y)$ from the degraded version $g(x,y)$, when some knowledge of degradation function H and noise n is available. The degraded image $g(x,y)$ as shown in Fig. 24 can be expressed mathematically as

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

This equation is in spatial domain and * represents convolution operation. An equivalent frequency domain representation is graphically shown in Fig. 25 and expressed as

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

Here $G(u,v) = F[g(x,y)]$
$H(u,v) = F[h(x,y)]$
$F(u,v) = F[g(x,y)]$

$N(u,v) = F[n(x,y)]$

$F(u,v) = H^{4}(u,v)[G(u,v) - N(u,v)]$

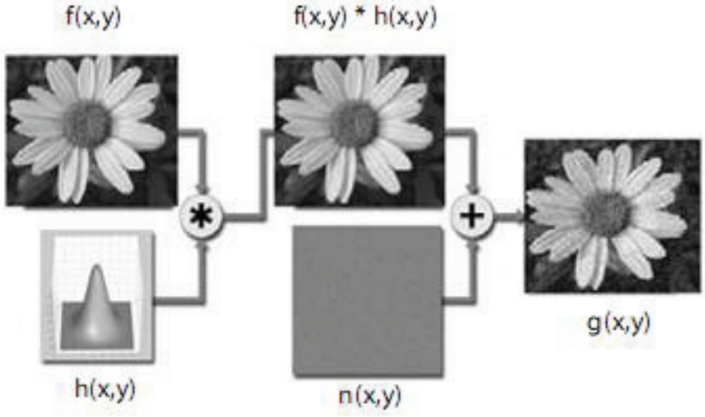
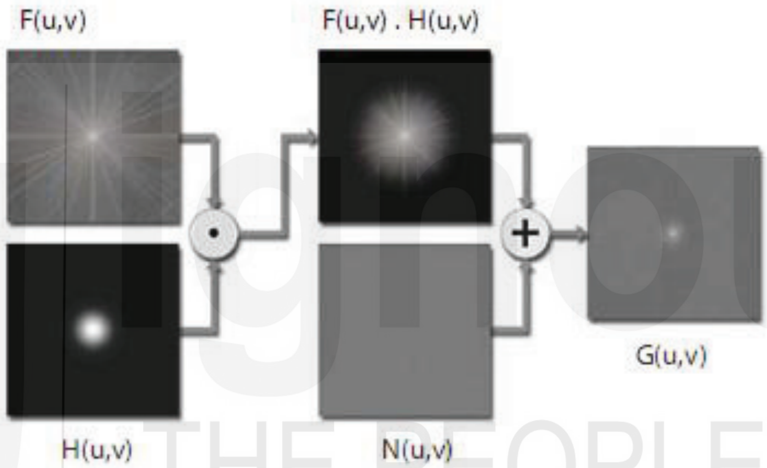


Fig. 24: Image Degradation Model (Spatial



Domain)

Fig. 25: Image Degradation Model (Frequency Domain)

Restored image can be obtained by the above equation. The problems in implementing this equation are

1) The noise N is unknown. Only the statistical properties of noise can be known.
2) The operation H is singular or ill posed. It is very difficult to estimate H.

Try an exercise.

E10) Explain in detail an image degradation model.

In the following section, we shall discuss noise models in detail.

6.7 NOISE MODELS

Major source of noise in digital images is during image acquisition. Non-ideal image sensors and poor quality of sensing elements contribute to majority of noise. Environmental factors such as light conditions, temperature of atmosphere, humidity, other atmospheric disturbances also account for noise in images. Transmission of image is also a source of noise. Images are corrupted with noise because of interference in the channel, lightning and other disturbances in wireless network. Human interference also plays a part in addition of noise in images.

Properties of Noise

Spatial and frequency characteristics of noise are as follows:

1) Noise is assumed to be 'white noise' (it could contain all possible frequency components), as such, fourier spectrum of noise is constant.

2) Noise is assumed to be independent in spatial domain. Noise is '**uncorrelated**' with the image, that is, there is no correlation between pixel value of image and value of noise components.

The spatial noise descriptor is the statistical behavior of the intensity values in the noise component. Noise intensity is considered as a random variable characterized by a certain probability density function (PDF).

Restoration techniques are oriented towards modeling the degradation (noise in this case) and restore an image to the original state. Most types of noise are modeled as known PDFs. Based on the estimated parameters from the noisy image, a particular noise PDF is chosen. Noise models are divided into two categories:

a) Noise which is independent of spatial location: Gaussian, Rayleight, Gamma, Exponential, Uniform noise are examples of this category.

b) Noise which is dependent on spatial location: Periodic noise is example of this type of noise.

Following are the most commonly occurring noise models:

Gaussian Noise

Gaussian noise model is most frequently used in practice. The PDF of a Gaussian random variable 'z' is given by

$$p(z) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(z-\mu)^2}{2\sigma^2}}, \qquad (2)$$

where $z =$ intensity/grey level value

$\mu =$ mean (average) value of z

$\sigma =$ standard deviation

Plot of $p(z)$ with respect to z is shown in Fig. 26. 70% of its values are in the range $[(\mu-\sigma),(\mu-\sigma)]$ while 95% of the values are in the range $[(\mu-2\sigma),(\mu+2\sigma)]$. DFT of **Gaussian (normal) noise is another Gaussian process**. This property of Gaussian noise makes it most often used noise model. Some examples where Gaussian model is the most appropriate model are electronic circuit noise, sensor noise due to low illumination or high temperature, poor illumination.

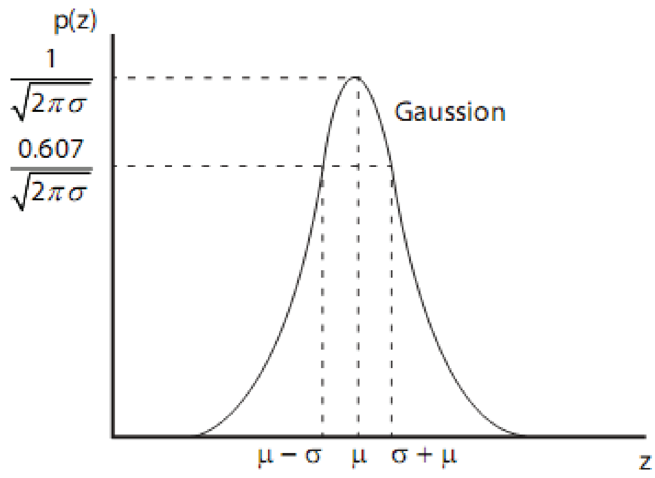


Fig. 26: PDF of Gaussian Noise Model

Gaussian noise is useful for modeling natural processes which introduce noise (e.g. noise caused by the discrete nature of radiation and the conversion of the optical signal into an electrical one – detector/shot noise, the electrical noise during acquisition – sensor electrical signal amplification, etc.).

Rayleigh Noise

Radar range and velocity images typically contain noise that can be modeled by the Rayleigh distribution. Rayleigh distribution is defined by

$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-\frac{(z-a)^2}{b}} & z \geq a \\ 0 & z < a \end{cases}, \quad (3)$$

Mean density is given as $\mu = a + \sqrt{\pi^{b/4}}$ and variance is given by $\sigma^2 = \frac{b(4-\pi)}{4}$.

Plot of PDF is shown in Fig 27. As it is clear that the curve doesn't start from origin and is not symmetrical with respect to the centre of the curve. Thus, Rayleigh density is useful for approximating skewed (non-uniform) histograms. This is mainly used in range imaging.

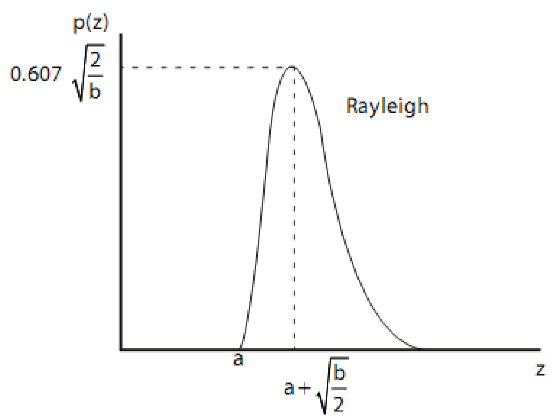


Fig. 27: PDF of Rayleigh Noise

Erlang (Gamma) Noise

Erlang noise is given by

$$p(z) = \begin{cases} \dfrac{a^b z^{b-1}}{(b-1)!} e^{-nz} & z \ge 0 \\ 0 & z < 0 \end{cases}, \qquad (4)$$

Where a and b are positive integers, mean density is given by $\mu = \dfrac{b}{a}$ and variance is $\sigma^2 = \dfrac{b}{a^2}$.

When the denominator is a gamma function, the PDF describes the gamma distribution. Plot is shown in Fig. 28.

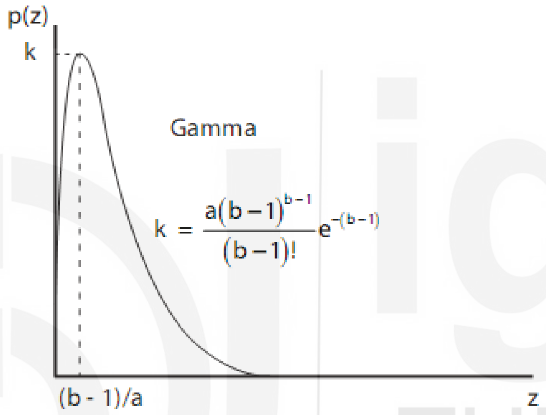


Fig. 28: PDF of Erlang Noise

Uniform Noise

Uniform noise is specified as

$$p(z) = \begin{cases} \dfrac{1}{b-a} & \text{if} \quad a \le z \le b \\ 0 & \text{otherwise} \end{cases}$$

Then mean and variance of uniform noise is given by

$$\mu = \frac{a+b}{2}, \quad \sigma^2 = \frac{(b-a)^2}{12}$$

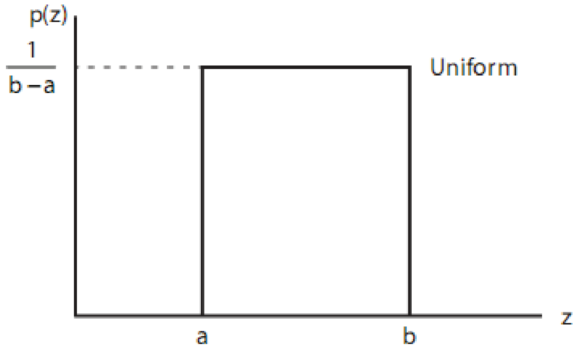


Fig. 29: PDF of Uniform Noise.

Fig. 29 shows the plot of PDF of uniform noise. Uniform noise is least used in practice.

Impulse (Salt and Pepper) Noise

Impulse (salt and pepper) noise is specified as

$$p(z) = \begin{cases} P_a & z = a \\ P_b & z = b \end{cases}, \qquad (6)$$

Fig. 30 shows the plot of PDF of impulse noise. If $b > a$, intensity (grey level) 'b' will appear as a light dot on the image and 'a' appears as a dark dot. This is a **'bipolar'** noise, If $P_a = 0$ or $P_b = 0 \Rightarrow$ **unipolar noise**
Generally, a and b values are saturated (very high or very low value), resulting in positive impulses being white (salt) and negative impulses being black (pepper). If $P_a = 0$ and P_b exists, this is called '**pepper noise**' as only black dots are visible as noise. If $P_b = 0$, only P_a exists, this is called '**salt noise**' as only white dots are visible on the image as noise.

Impulse noise occurs when quick transitions happen, such as faulty switching takes place. Noise parameters are generally estimated based on histogram of small flat area of noisy image.

The salt & pepper noise is generally caused by malfunctioning of camera's sensor cells, by memory cell failure or by synchronization errors in the image digitizing or transmission.

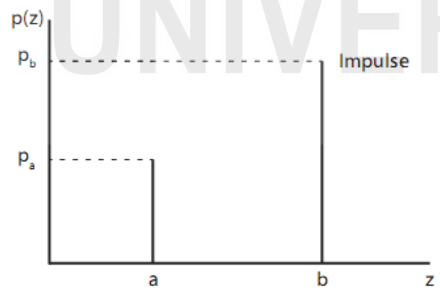


Fig. 30 PDF of Uniform noise.

Fig. 31 shown an example of impulse noise with $p = 0.1$ added to input image and togenerate a noisy image g. Noise level $p = 0.1$ means that approximately 10% of pixelsare contaminated by salt or pepper noise (highlighted by box)

128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128

f(x,y)

⇒

128	128	255	0	128	128	128	128	128	128
128	128	128	128	0	128	128	128	128	0
128	128	128	128	128	128	128	128	128	128
128	128	0	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
128	128	128	128	128	128	128	128	128	128
0	128	128	128	128	255	128	128	128	128
128	128	128	128	128	128	128	128	128	255
128	128	128	128	128	128	128	255	128	128

g(x,y)

Fig. 31: Numerical Example of Adding Impulse Noise with P = 0.1.

Fig. 32 shows the flower image with different types of noise. It is very easy to identify the effect of different types of noise on the images. Fig. 32 (a) shows original image, Fig. 32 (b) shows image with Gaussian noise. Fig. 32 (c) shows image with salt and pepper noise and Fig. 32 (d) shows image with uniform noise. The amount of noise added can also vary. If the amount of noise added is more, it becomes very difficult to remove it.

(a) Original image

(b) Image with Gaussian noise

(c) Image with salt & pepper noise

(d) Image with uniform noise

Fig. 32

Let us discuss an important type of noise.

Periodic noise is a spatially dependent noise. During Image acquisition, electrical or electromechanical interference may cause such type of periodic noise. A strong periodic noise can be seen in the frequency domain as equi-spaced dots at a particular radius around the centre (origin) of the spectrum. Fig. 33 shows image with periodic noise.

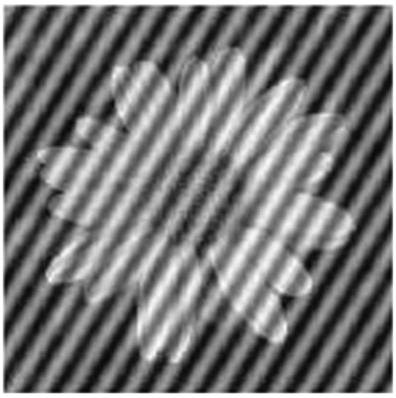


Fig. 33: Image with periodic noise

Now, you may like to try the following exercises.

E11) What is noise? How noise can be eradicated?

E12) Explain the types of noises based on its probability distribution.

In the following section, we will discuss restoration in the presence of noise only-spatial filtering.

6.8 RESTORATION IN THE PRESENCE OF NOISE ONLY – SPATIAL FILTERING

If H is an identity operator and degradation is only due to additive noise,

$$g(x, y) = f(x, y) + n(x, y)$$
$$G(u, v) = F(u, v) + N(u, v)$$

As noise is unknown, $f(x, y) = g(x, y) - n(x, y)$ is not realistic. Thus, spatial filtering is used when additive random noise is present. Mean and median filters are used for noise removal. Band reject and band pass filters are used for periodic noise removal.

6.8.1 Mean Filters

Spatial Smoothing concepts are explained in earlier unit 4 of this course. Now, Consider Fig. 34, $S_{x,y} = \text{Sub}$ image window of size $m \times n$ centred at (x, y). Fig. 35 shows 3×3 and 5×5 sub images. Mean filter computes average value of the corrupted image $g(x, y)$ in the area defined by $S_{x,y}$

$$\hat{f}(x,y) = \frac{1}{mn}\sum_{s,t \in S_{xy}} g(s,t)$$

Such filter smooths local variations in an image, thus reducing noise and introducing blurring. This filter is well suited for random noise like Gaussian, uniform noise.

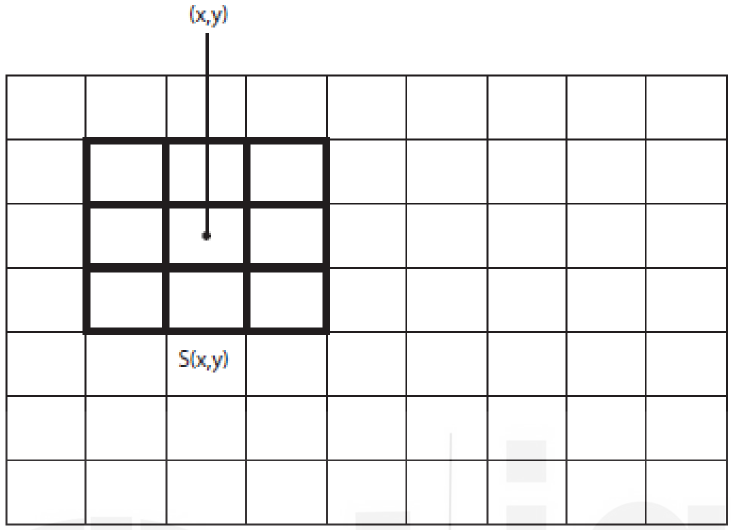
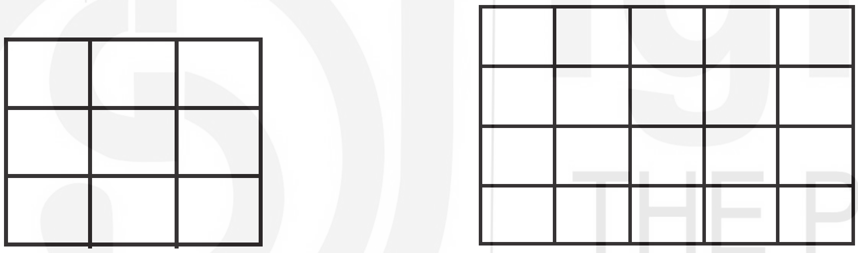


Fig. 34: Graphic Illustration of Sub-Image in An Image



(a) $S(x,y)$ of size 3×3 (b) $S(x,y)$ of size 5×5

Fig. 35: Sub-Image of Various Sizes

Thus, new value at (x,y) in image in Fig. 36 is
$\{g(s,t)\} = 19 = [30+10+20+10+250+25+20+25+3] = 46.7 \approx 47$

30	10	20
10	250	25
20	25	30

→

×	×	×
×	$46.7 \approx 47$	×
×	×	×

Fig. 36: Example of Mean Filtering.

Let us apply this in the following example.

Example 1: Show effect of 3×3 mean filter on a simple image in Fig. 37 (a) and Fig. 38 (a)

Solution: As explained in Unit 4, a 3 × 3 mean filter is overlapped with image and output for that particular pixel is derived and then filter centre is moved to the next pixel. We generate a lower size image because the filter mask doesn't overlap fully on first and last row and column.

0	0	0	0	0
0	0	0	1	1
0	0	1	1	1
0	0	1	20	1
0	0	1	1	1

(a)

Mean filter $\longrightarrow$

	$\frac{1}{9}$	$\frac{3}{9}$	$\frac{5}{9}$	
	$\frac{2}{9}$	$\frac{24}{9}$	$\frac{27}{9}$	
	$\frac{3}{9}$	$\frac{25}{9}$	$\frac{28}{9}$	

(b)

Fig.37: Input and Output Image.

Mean filter removes random noise by introducing blurring. Random noise value of 20 in Fig. 37 (a) is removed from the resultant image (b). But, importantly the edge is also diluted and blurred. In the second image Fig. 38 (a), which has fairly constant values, the pixel values remain more or less same in Fig. 38 (b).

0	1	2	3	4
5	6	7	8	9
5	5	5	9	9
5	5	5	9	9
5	5	5	9	9

(a)

Mean filter $\longrightarrow$

	4	5	6	
	4	6.5	8	
	5	6	8	

(b)

Fig.38 Input and Output Image.

Gaussian noise is added to the input image Fig. 39 (a). $3\times3, 5\times5$ and 7×7 mean filters are applied to the noisy image and the output images are displayed in Fig. 39 (b), (c), (d). As it is clear from the output, 3×3 filter (Fig. 39 (b)) does not remove the noise completely. Noise is still seen in the image but blurring is less. In 5×5 (Fig. 39 (c)) filtering more noise is removed but image gets blurred. In 7×7 (Fig. 39(d)), too much blurring is seen in the output.

(a) Original Image

(b) Filtered Image by Mean Filter 3×3

(c) Filtered Image by 5 × 5 **(d) Filtered Image by Mean Filter Mean Filter 7 × 7**

Fig. 39

Let us discuss median filter.

6.8.2 Median Filters

Median filter replaces the pixel value by the median of the pixel values in the neighbourhood of the centre pixel (x, y). The filtered image is given by

$$\hat{f}(x, y) = \underset{(s,t)\in S_{xy}}{\text{median}} \{g(s, t)\}$$

Fig. 40 shows the procedure of applying 3×3 median filter on an image. As impulse noise appears as black (minimum) or white (maximum) dots, taking median effectively suppresses the noise.

Thus, median filter provides excellent results for salt and pepper noise with considerably less blurring than linear smoothing filter of the same size. These filters are very effective for both bipolar and unipolar noise. But, for higher noise strength, it affects clean pixels as well and a noticeable edge blurring exists after median filtering.

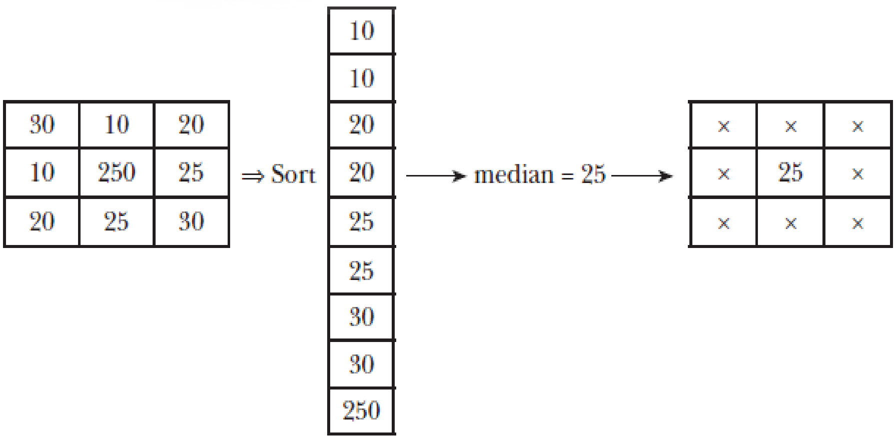


Fig. 40: Example of Median Filtering.

To understand this clearly, see the following example.

Example 2: Show the effect of 3×3 median filter on a simple image given in Fig. 41(a) and Fig. 41(b).

128	128	128	128	128
128	0	128	128	128
128	128	128	128	128
128	128	128	128	128
128	128	128	128	128

Median filter →

	128	128	128	
	128	128	128	
	128	128	128	

(a) Input image **(b) Output image**

Fig. 41

Solution: When a 3×3 median filter is implemented, all 9 pixels around 'hotspot' are arranged in ascending/ descending order. Center pixel is taken as output and center pixel is replaced by the output. This process is repeated for the entire image.

128	128	128	0	128
128	0	128	128	128
0	0	255	255	255
0	0	128	255	0
128	0	0	0	128

Median filter →

	128	128	255	
	0	128	255	
	0	0	128	

(a) Input image **(b) Output image**

Fig. 42

***

It is clear from Example 2, Fig. 41 (a) and Fig. 41 (b) that if noise strength is low in noisy image, output is completely clean. But if noise strength is more (more number of noisy pixels in the image), output is not completely noise free as can be seen in Fig. 42 (a) and Fig. 42 (b).

Let us see the effect of the median filter.

Salt and pepper noise is added to an image given in Fig. 43 (a), noisy image is shown in Fig. 43 (b). 3×3 mean filter and 5×5 median filter is applied on it. As it is clear from the result, (Fig. 43 (c) and Fig. 43 (d)) mean filter is not effective in removing salt and pepper noise. But median filter completely removes salt and pepper noise without introducing any blur.

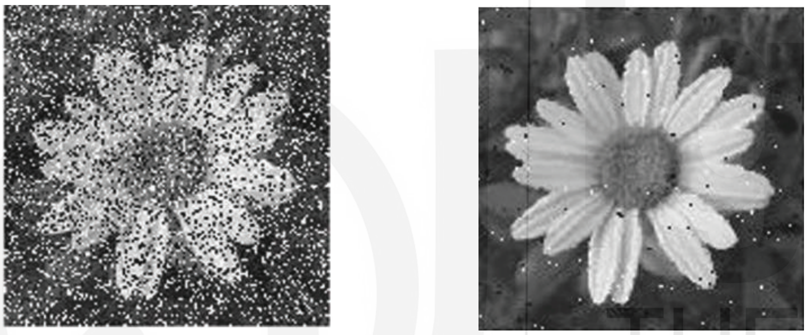
(a) Original Image

(b) Noisy Image

(c) Filtered Image with Mean Filter **(d) Filtered Image with Media Filter**

Fig. 43

Salt and pepper noise with density of 0.3 is added to an image. The noisy image (Fig. 44 (a)) is filtered using $3\times3, 5\times5$ and 7×7, median filter. The results in Fig. 44 (b), (c), (d) show that 3×3 median filter is unable to remove the noise completely as the noise density is high. But 5×5 and 7×7 median filters remove noise completely but some distortions are seen specially in Fig. 44 (d).



(a) Noisy Image **(b) Filtered Image with 3 × 3 Median Filter**

(c) Filtered Image with 5 × 5 Median Filter **(d) Filtered Image with 7 × 7 Median Filter**

Fig. 44

Now, in the following section, we shall discuss noise reduction.

6.9 PERIODIC NOISE REDUCTION

Periodic noise is spatially dependent noise and it occurs due to electrical or electromagnetic interference. It gives rise to a regular noise pattern in an image. Frequency domain (fourier domain) techniques are very effective in removing periodic noise. Basic steps in frequency domain filtering remain same as discussed above. Here, we are discussing two frequency domain filters; namely band reject filter and band pass filter.

6.9.1 Band Reject Filter

Removing periodic noise from an image involves removing a particular range of frequencies from the image. Transfer function of ideal band reject filter is

$$H(u,v) = \begin{cases} 1 & D(u,v) < D_0 - \frac{W}{2} \\ 0 & D_0 - \frac{W}{2} \le D(u,v) \le D_0 + \frac{W}{2}, \\ 1 & D(u,v) > D_o + \frac{W}{2} \end{cases} \tag{7}$$

where W is the width of the band (band width), D_0 is its radial centre and $D(u,v)$ is the distance from the origin and is given by

$$D(u,v) = \left[\left(u - \frac{M}{2}\right)^{1/2} + \left(v - \frac{N}{2}\right)^{1/2}\right]$$

$D(u,v)$ is the distance measured from the point (u,v) to the centre $\left(\frac{M}{2}, \frac{N}{2}\right)$.

If size of an image is $M \times N$, then the centre is at $\left(\frac{M}{2}, \frac{N}{2}\right)$.

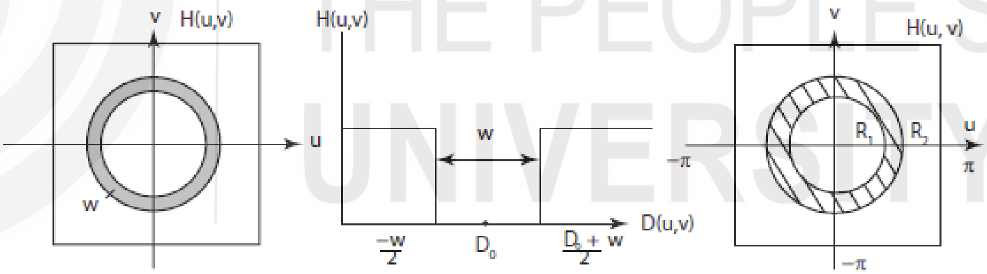


Fig 45: Frequency response of ideal band reject filter

Transfer function of butter worth band reject filter of order 'n' is given by

$$H(u,v) = \frac{1}{1 + \left[\frac{D(u,v)W}{D^2(u,v) - D_0^2}\right]^{2n}} \tag{8}$$

Gaussian band reject filter is given by

$$H(u,v) = 1 - e^{\frac{1}{2}\left[\frac{D^2(u,v) - D_0^2}{D(u,v)W}\right]} \tag{9}$$

Fig. 46 gives the plots of ideal, butterworth and gaussian band reject filters.

(a) Ideal **(b) Butterworth (order 1)** **(c) Gaussian**

Fig. 46: Plots of Band Reject Filters

6.9.2 Band Pass Filter

Band pass filter performs just opposite to band reject filter. The transfer function of band pass filter can be obtained from band pass filters.

$$H_{bp}(u,v) = 1 - H_{br}(u,v), \tag{10}$$

Where, H_{bp} is transfer function of band pass filter and H_{br} is transfer function of band reject filter.

Ideal band pass filter is given by

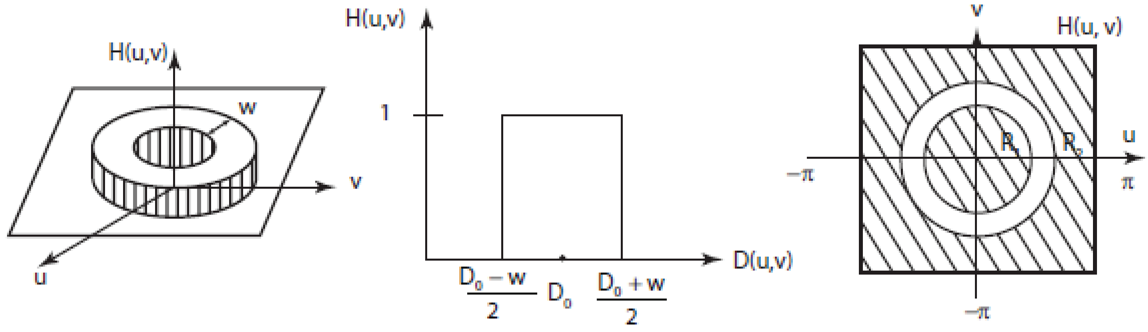
$$H(u,v) = \begin{cases} 0 & D(u,v) \le D_0 - \dfrac{W}{2} \\ 1 & D_0 - \dfrac{W}{2} < D(u,v) < D_0 + \dfrac{W}{2}, \\ 0 & D(u,v) \ge D_0 + \dfrac{W}{2} \end{cases} \tag{11}$$

Where $D(u,v)$ is the distance from origin, W is the band width D_0 is the radial centre or the cut off frequency.

Fig. 47 shows the transfer function of ideal band pass filter. Butterworth band pass filter of order 'n' is given by

$$H(u,v) = 1 - H(u,v)_{\text{butterworth bandreject}} \tag{12}$$

$$H(u,v) = 1 - \frac{1}{1 + \left[\dfrac{D(u,v)W}{D^2(u,v) - D_0^2}\right]^{2n}} = \frac{\left[\dfrac{D(u,v)W}{D^2(u,v) - D_0^2}\right]^{2n}}{1 + \left[\dfrac{D(u,v)W}{D^2(u,v) - D_0^2}\right]^{2n}} \tag{13}$$



(a) **(b)** **(c)**

Fig. 47: Frequency Response of Ideal Band Pass Filter

Similarly, Gaussian band pass filter is given by

$$H(u,v) = 1 - H(u,v)_{\text{gaussianbandpass}}$$

$$= 1 - \left[1 - e^{-1/2\left[\frac{D^2(u,v) - D_0^2}{D(u,v)W}\right]^2}\right]$$

$$= e^{-1/2\left[\frac{D^2(u,v) - D_0^2}{D(u,v)W}\right]^2}$$

6.10 ESTIMATION OF DEGRADATION FUNCTION

In order to restore the image, we need to estimate the degradation function. There are three principal ways to estimate the degradation function to be used in restoration:

1) Observation
2) Experimentation
3) Mathematical modelling

Once the degradation function has been estimated, then, restoration is a de convolution process also called **Blind Deconvolution.**

6.10.1 Observation

In restoration using observation, we assume that an image $g(x,y)$ is degraded with an unknown degradation function H. We try to estimate H from the information gathered from the image itself. For example, in case of burred image, a small rectangular section of image containing a part of object and background is taken (Fig. 48). To reduce the effect of noise, the chosen part should be such that it shows presence of a strong signal. We try to un-blur that sub-image manually as much as possible and generate $f_s(x,y)$ from $g_s(x,y)$.

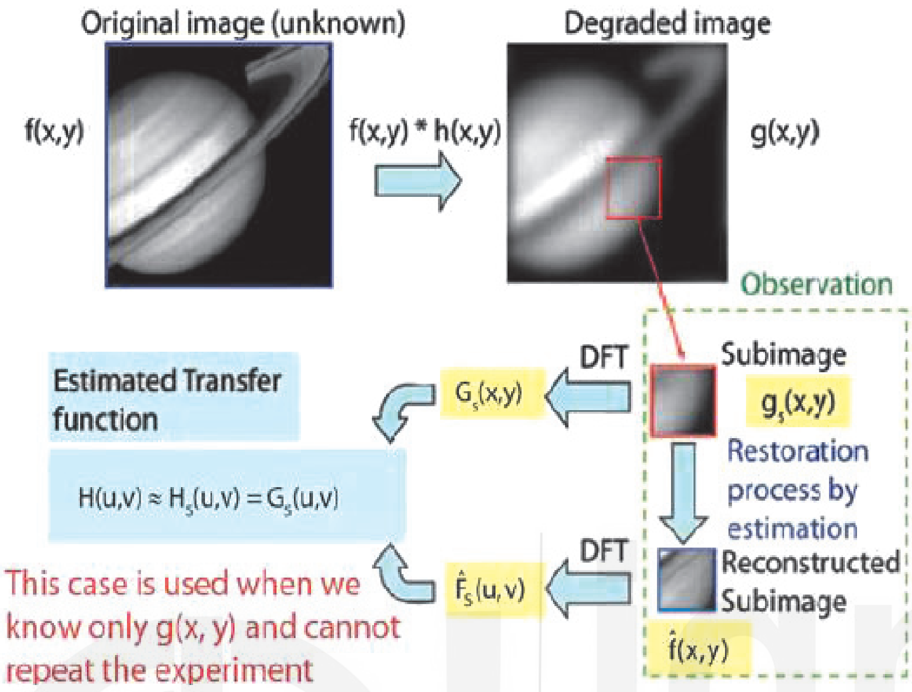


Fig. 48: Estimation by image observation

Here, $g(x, y)$ is the original sub image

$\hat{f}_s(x, y)$ is the restored version of $g_s(x, y)$

Thus, degradation can be estimated for the sub image by

$$H_s(u,v) = \frac{g_s(u,v)}{\hat{F}_s(u,v)}$$

From the characteristics of $H_s(u,v)$, we try to deduce the complete degradation function $H_s(u,v)$ based on the assumption of position invariance. For example, if $H_s(u,v)$ has a Gaussian shape, we can construct $H(u,v)$ on a larger scale with the same (Gaussian)shape. This is a very involved process and is used in very specific situations.

6.10.2 Experimentation

It is possible to estimate the degradation function accurately if the equipment used to acquire the degraded image is available. The processes is shown in Fig. 49.

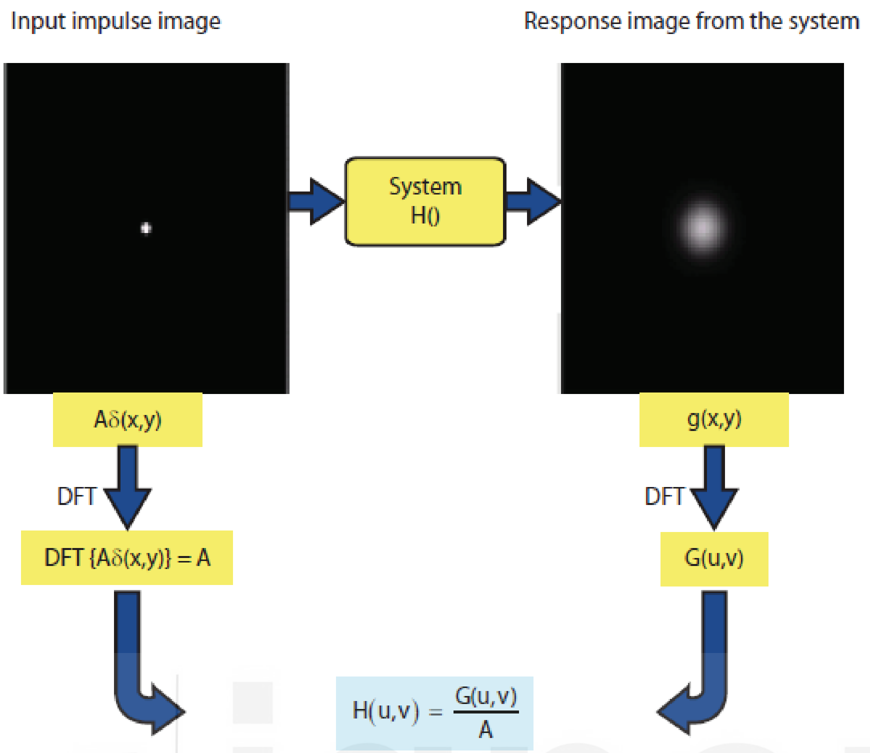


Fig. 49: Estimation by experimentation

We now list the steps performed for estimation.

Step 1: First step is to adjust the equipment by varying the system setting such that the image obtained is similar to the degraded image that needs to be restored.

Step 2: Second step is to obtain the impulse response of the degradation by imaging an impulse using the same system setting, since a linear space – invariant system is completely characterized by its impulse response. An impulse is simulated by a maximally bright dot of light. As shown in the Fig. 2, impulse response is given by

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

where $G(u,v) = DFT[g(x,y)] = DFT\,[\text{degraded impulse}]$, and A is the constant describing the strength of the impulse.

6.10.3 Modelling

Modelling is used to estimate the degradation function. Scientists have studied several environmental conditions and other processes which cause degradation, and have formulated several fundamental models for the degradation functions. Degradation model based on atmospheric turbulence blur is given as

$$H(u,v) = e^{-k(u^2+v^2)5/6}$$
$$h(x,y) = e^{-k(x^2+y^2)5/6},$$

where k is a constant that depend on the nature of blur. Various values used for the constant k along with their type of turbulence are given as

$k = 0.0025$ for server turbulence
$k = 0.001$ for wild turbulence
$k = 0.00025$ for low turbulence

This is commonly used in remote sensing and axial imaging applications. Degradation model for uniform out of focus blur (optical blur).

$$h(x,y) = \begin{cases} \frac{1}{L^2}; & \frac{-L}{2} \leq x^0, y^0 \leq \frac{L}{2} \\ 0 \ ; & \text{otherwise} \end{cases}$$

(1)

$$H(u,v) = \frac{T}{\pi(ua+vb)} \sin(ua+vb)\, e^{-j\pi(ua+vb)}$$

(2)

Blur can be due to motion (camera and object moving with respect to each other). With suitable values of T, a and b, blurred image can be generated using this transfer function. Fig. 50(a) shows original image and Fig. 50(b) shows blurred image.

(a) Original Image **(b) Blurred Image**

Fig. 50

Try an exercise.

E13) What are the different methods of estimation of image degradation function?

In the following section, we shall discuss inverse filtering.

6.11 INVERSE FILTERING

Inverse filter is also known as reconstruction filter. Deblurring is very important in restoration applications because blurring is visually annoying. It is bad for analysis, and de-blurred images have plenty of applications. Applications include astronomical imaging, law enforcement (identifying criminals), biometrics etc. In this unit we will discuss Inverse filtering, pseudo-inverse filtering and Wiener filtering for deblurring.

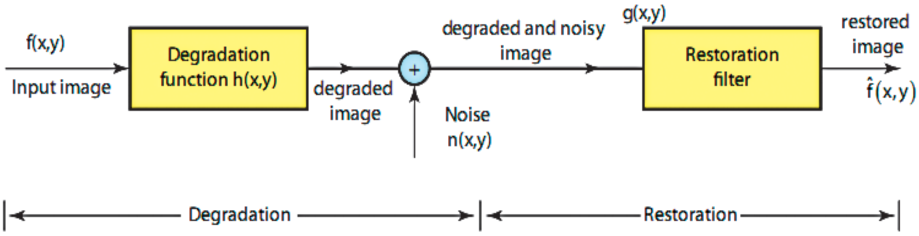


Fig. 51: Block diagram of degradation/restoration model

As in the absence of noise, degradation model becomes,

$$G(u,v) = F(u,v)H(u,v) \quad (3)$$

Simplest approach to restoration is direct inverse filtering, where we can compute an estimate $\hat{F}(u,v)$, of the transform of the original image simply by dividing transform of the degraded image by the degradation function.

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)} \quad (4)$$

In presence of noise, degradation model as shown in Fig. 51 becomes

$$G(u,v) = F(u,v)H(u,v) + N(u,v) \quad (5)$$

After applying inverse filtering

$$\hat{F}(u,v) = H_R(u,v)G(u,v) \quad (6)$$

Substituting the values of $H_R(u,v)$ and $G(u,v)$, we get

$$\begin{aligned}\hat{F}(u,v) &= \frac{1}{H(u,v)}[F(u,v)H(u,v) + N(u,v)] \\ &= \frac{F(u,v)H(u,v)}{H(u,v)} + \frac{N(u,v)}{H(u,v)} \\ &= F(u,v) + \frac{N(u,v)}{H(u,v)} \quad (7)\end{aligned}$$

Thus, in case of noisy degraded images, output is also noisy.

If $H(u,v) \approx 0 \frac{N(u,v)}{H(u,v)} \rightarrow \infty$

noise is amplified and it dominates the output.

Limitations of inverse filtering are:

1) It is an unstable filter
2) It is sensitive to noise. In practice, inverse filter is not popularly used.

To remove the limitations of inverse filter, pseudo inverse filters are used. Pseduo Inverse filter is defined as,

$$H_R(u,v) = \begin{cases} \dfrac{1}{H(u,v)} & |H(u,v)| \geq \in \\ 0 & |H(u,v)| < \in \end{cases}$$

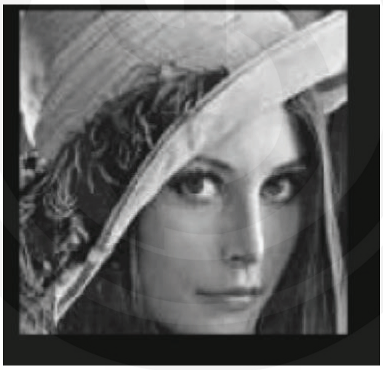
where, $\in$ is a small value.

Pseudo-inverse filters eliminates the first problem of inverse filters (un-stability).

As $H(u,v) \to 0, H_R(u,v) = 0$.

Hence, it does not allow $H_R(u,v) \to \infty$.

It is a stable filter. This filter also has a problem of noise amplification. Inverse filtering is applied to the image blurred with a Gaussian (Fig. 52(b)). The output image (Fig. 52(c)) is very close to the original image (a). Then same inverse filtering is used when the blurred image is subjected to additive noise with different strengths. Outputs are shown in Fig. 52(d), (e) and (f). As noise increases, the output of filter goes down.



(a) Original Image

(b) Image Blurred with a Gaussian

(c) Inverse Filter Applied to Noiseless Blurred

(d) Inverse Filter Applied to Blurred Image Plus

(e) Inverse Filter Applied to Blurred Image Plus Noise (0.1)

(f) Inverse Filter Applied to Blurred Plus Noise (0.5)

Fig. 52

Try an exercise.

E14) Explain in brief the inverse filtering approach and its limitations in image restoration.

In the following section, we shall discuss wiener filter.

6.12 WIENER FILTER

Wiener filter is also known as minimum mean square error. This approach includes both the degradation function and power spectrum of noise characteristics in developing the restoration filter. Wiener filter restores the image in the presence of blur as well as noise.

This method is founded by considering image and noise as random variables and objective is to find as estimate $\hat{f}$ of the uncorrupted image f such that the mean square error between them is minimized. This error is given by

$$e^2 = E\{(f - \hat{f})^2\}), \qquad (9)$$

where, $E\{.\}$ is the expected value of the argument. Noise and image are assumed to be uncorrelated. Filter transfer function is given by

$$H_R(u,v) = \frac{S_{fg}(u,v)}{S_{gg}(u,v)}, \qquad (10)$$

where, $S_{fg}(u,v)$ is the power spectral density of recovered image and noisy image and $S_{gg}(u,v)$ is the power spectral density of noisy image

$$H_R(u,v) = \frac{S_{fg}(u,v)}{S_{gg}(u,v)} = \frac{H^*(u,v)S_{ff}(u,v)}{|H(u,v)|^2 S_{ff}(u,v) + S_{nn}(u,v)}, \qquad (11)$$

where, $H(u,v) =$ degradation function.

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

$S_{nn}(u,v) =$ power spectral density of noise
$S_{ff}(u,v) =$ power spectral density of undergraded image.

If $S_{ff}(u,v), S_{nn}(u,v)$ and $H(u,v)$ is known $H_R(u,v)$ is completely known. Wiener filter works very well for specific applications and is not suitable for general images. For example, if a wiener filter $H_R(u,v)$ is working well for faces, same filter would not work for landscapes etc. Now we discuss several cases to test wiener filter.

Case 1: When, there is no noise $S_{nn}(u,v) = 0$

$$H_R(u,v) = \frac{H^*(u,v)S_{ff}(u,v)}{|H(u,v)|^2 S_{ff}(u,v) + 0}$$

$$H_R(u,v) = \frac{1}{H(u,v)} = \text{inverse filter}$$

Thus, if there is no noise, Wiener filter = Inverse filter.

Case 2: Noise is present, but there is no degradation, $H(u,v) = 1$

$$\therefore H_R(u,v) = \frac{1.S_{ff}(u,v)}{1.S_{ff}(u,v) + S_{nn}(u,v)}$$

$$= \frac{S_{ff}(u,v)/S_{nn}(u,v)}{\dfrac{S_{ff}(u,v)}{S_{nn}(u,v)} + 1}$$

$$= \frac{SNR}{SNR + 1}$$

$$SNR = \text{Signal to noise ratio} = \frac{S_{ff}(u,v)}{S_{nn}(u,v)} = \frac{\text{PSD of signal}}{\text{PSD of noise}}$$

$$\Rightarrow SNR >> 1 \quad H_R(u,v) \approx \frac{SNR}{SNR} = 1$$

a) If signal to noise ratio is high

Thus, if SNR high, wiener filter acts like pass band and allows all the signal to pass through without any attenuation.

b) If $SNR << 1$, if Signal to noise ratio is low, then

$$H_R(u,v) = \frac{SNR}{1} = SNR$$

$$= \text{a very low value}$$

$$\approx 0$$

Thus, if SNR is low and noise level very high, $H_R(u,v) \approx 0$, acts as a stop band for signal and doesn't allow signal to pass, thus attenuating noise. If noise is high in the signal, wiener filter reduces it after filtering.

$$SNR = \frac{\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}|F(u,v)|^2}{\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}|N(u,v)|^2}$$

SNR gives a measure of the level of information bearing signal power (i.e. of the original, undegraded image) to the level of noise power. Images with low noise tend to have high SNR and conversely, the same image with higher noise level has a high SNR.

The mean square error is given by

$$MSE = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[f(x,y) - \hat{f}(x,y)]^2$$

Here, $f(x,y)$ is the original image and $\hat{f}(x,y)$ is the restored image.

Wiener filter is also called minimum mean square error (MMSE) or Least square (LS) filtering because it minimizes the error between the image and its estimate.

$$H_R(u,v) = \frac{H^*(u,v)S_{ff}(u,v)}{|H(u,v)|^2 S_{ff}(u,v) + S_{nn}(u,v)}$$

$$= \frac{H^*(u,v)}{|H(u,v)|^2 + \frac{S_{nn}(u,v)}{S_{ff}(u,v)}} \longrightarrow \text{difficult to estimate}$$

Wiener filter in this form is not very useful, as it is difficult of estimate noise power spectrum S_{nn} and undergraded image power spectrum S_{ff},

$S_{nn}(u,v) \rightarrow$ difficult to estimate

$S_{ff}(u,v) \rightarrow$ difficult to estimate

To solve this, we can do an approximation

$$\frac{S_{nn}(u,v)}{S_{ff}(u,v)} \approx k \text{ (approximated by constant } k)$$

$$H_R(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + k}$$

And estimated restored image

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

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

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

k is chosen experimentally and iteratively for best results. In Fig. 53, small noise is added to a blurred image, which is restored by wiener filter in Fig. 53(b).If the amount of added noise is increased Fig. 53(c), the restored image by wiener filter (Fig. 53(d)) is not good. Thus, it is apparent that the wiener filter only works well when the noise is small.

(a) Blurred Image with Small Additive noise

(b) Image Restored by by Wiener Filter

(c) Blurred Image with Increase Additive Noise

(d) Image Restored by Wiener Filter

Fig. 53

(a) Original Image

(b) Blurred Image

(c) Restored image

Fig.54: Applying Wiener Filter

Image is blurred using linear motion = 15, angle = 5 shown in Fig. 54(b). Wiener filter is used to deconvolve the blurred image. The output (Fig. 54(c)) is not clear as the wiener filter does not use any prediction about noise density.

Now, try an exercise.

E15) Discuss the minimum mean square error (Wiener) filtering.

Now, we summarise what we have studied in the unit.

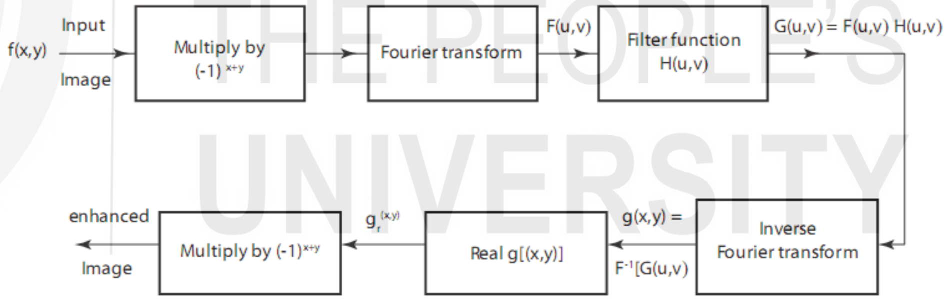
6.13 SUMMARY

In this unit, we have discussed the following points.

1. Image characteristics in frequency domain
2. Filtering in frequency domain
3. Basic steps of frequency domain filtering
4. Various low pass and high pass filters
5. Various image smoothing filters in frequency domain
6. Various image sharpening filters in frequency domain
7. Sources of degradation.
8. Difference between enhancement and restoration.
9. Image degradation/restoration model.
10. Various types of noises with their pdfs.
11. Mean and median filters for noise reduction
12. Band reject and band pass filters for periodic noise reduction.
13. Methods of estimation of degradation function.
14. Inverse filtering.
15. Wiener filtering.

6.14 SOLUTIONS AND ANSWERS

E1)



$$u = \frac{M}{2} \text{ and } v = \frac{N}{2}.$$

1. Multiply input image $f(x, y)$ by $(-1)^{x+y}$ to centre the transform to
2. Compute $F(u, v)$, Fourier transform of the output of step 1.
3. Multiply filter function $H(u, v)$ to $F(u, v)$ to get $G(u, v)$.
4. Take inverse Fourier transform of $G(u, v)$ to get $g(x, y)$.
5. Take the real part of $g(x, y)$ to get $g_r(x, y)$
6. Multiply the result of step 5 by $(-1)^{x+y}$ to shift the centre back to origin and enhanced image is generated.

E2) Image enhancement can be done very effectively in frequency domain. High frequency noise, undesirable breakages in the edges and other imperfections can be taken care by filtering in frequency domain. Low pass and high pass filters are implemented with ease and perfection in frequency domain.

E3) $$H(u,v) = \frac{1}{1+[D(u,v)/D_0]^{2n}},$$

where D_0 = Cut off frequency or distance from the centre n = filter order $\left(\frac{M}{2}, \frac{N}{2}\right)$

E4)

	Ideal	Butterworth	Gaussian
Transfer function	$H(u,v) = \begin{cases} 1, & D(u,v) \le D_0 \\ 0, & D(u,v) > D_0 \end{cases}$	$H(u,v) = \frac{1}{1+[D(u,v)/D_0]^{24}}$	$H(u,v) = e^{-D^2(u,v)/2D^2}$
Application	**Reduce noise**	**Reduce noise**	**Reduce noise**
Problems	**Blurring Ringing**	**Blurring, Ringing for higher order filters**	**Blurring no ringing**

E5) Smoothing filters are low pass filters (LPF). Edges, sharp transitions and noise in the grey levels contribute to high frequency contents in an image. A low pass filter only passes low frequency and blocks the high ones. It removes noise and introduces blurring as a side effect in the image.

Ringing is undesirable and unpleasant lines around the objects present in the image. As the cut of frequency D_0 increases, effect of ringing reduces. Ringing is a side effect of ideal *lpf*.

$\uparrow D_0$ ——— $\downarrow$ Ringing + blur

E6) LPF are generally used as a preprocessing step before an automatic recognition algorithm. It is also used to reduce noise in images. Few examples are listed below.

Character Recognition, Object counting, Printing and publishing industry, "Cosmetic" processing etc.

E7) The sharpening filters are listed as follows:

1. Ideal high pass filter
2. Butterworth high pass filter
3. Gaussian high pass filter

High pass filters are used for enhancing edges. These filters are used to extract edges and noise is enhanced, as a side effect.

E8) Gaussian high pass filters have smooth transition between passband and stop band near cut off frequency. The parameter D is a measure of spread of the Gaussian curve. Larger the value D_0, larger is the cut off frequency. Transfer function of GHPF is

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

where D_0 = cut off frequency and $D(u,v)$ is the distance from origin of Fourier transform.

E9) Image degradation can happen due to

a) **Sensor distortions:** Involves quantization, sampling, sensor noise, spectral sensitivity, de-mosaicking, non linearity of sensor etc.

b) **Optical distortions:** are geometric distortion, blurring due to camera mis-focus.

c) **Atmospheric distortions:** are haze, turbulence etc.

d) **Other distortions:** Low illumination, relative motion between object and camera etc.

E10) Fig shows the block diagram of degradation/restoration model. Degradation function $h(x,y)$ and noise $n(x,y)$, operate on input image $f(x,y)$ to generate a degraded and noisy image $g(x,y)$.

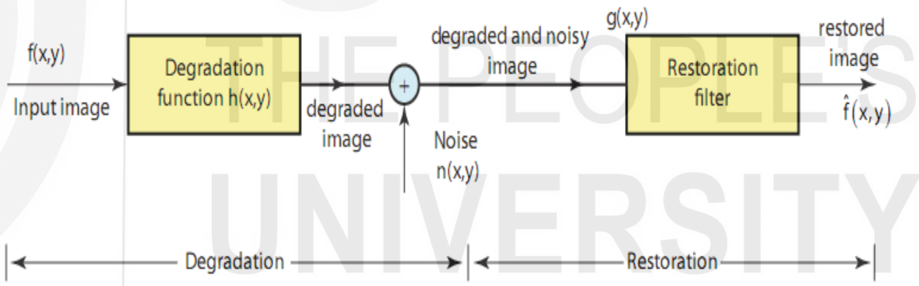


Fig.: Block diagram of degradation/restoration model

$f(x,y)$ = original image
$h(x,y)$ = degradation function
$n(x,y)$ = additive noise
$g(x,y)$ = degraded and noisy image
$\hat{f}(x,y)$ = restored image

E11) Noise is a disturbance that causes fluctuations in pixel values. Pixel values show random variations and can cause very disturbing effects on the image. Thus suitable strategies should be designed to model and remove/ reduce noise. Major source of noise in digital images is during image acquisition. Non-ideal image sensors and poor quality of sensing elements contribute to majority of noise. Environmental factors such as light conditions, temperature of atmosphere, humidity, other atmospheric disturbances also account for noise in images.

Transmission of image is also a source of noise. Images are corrupted with noise because of interference in the channel, lightning and other disturbances in wireless network. Human interference also plays a part in addition of noise in images.

Properties of Noise

Spatial and frequency characteristics of noise are as follows:

1) Noise is assumed to be 'white noise' (it could contain all possible frequency components), as such,Fourier spectrum of noise is constant.

2) Noise is assumed to be independent in spatial domain. Noise is '**uncorrelated**' with the image, that is, there is no correlation between pixel value of image and value of noise components.

Based on noise properties and types of noise, different filters are used to reduce/remove noise.

E12) **Gaussian Noise**

Gaussian noise model is most frequently used in practice. The PDF of a Gaussian random variable 'z' is given by

$$p(z) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(z-u)^2}{2\sigma^2}},$$

where $z =$ intensity/grey level value
$\mu =$ mean (average) value of z
$\sigma =$ standard deviation

Rayleigh Noise

Radar range and velocity images typically contain noise that can be modeled by the Rayleigh distribution. Rayleigh distribution is defined by

$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-\frac{(z-a)^2}{b}} & z \geq a \\ 0 & z < a \end{cases}$$

Mean density is given $\mu = a + \sqrt{\pi b/4}$ as $\sigma^2 = \frac{b(4-\pi)}{4}$

Erlang (Gamma) Noise

Erlang noise is given by

$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!} e^{-z} & z \geq 0 \\ 0 & z < 0 \end{cases}$$

a and b are positive integers. Mean density is given by

and variance $\sigma^2 = \frac{b}{a^2}$ is $\mu = \frac{b}{a}$

Uniform Noise

Uniform noise is specified as

$$p(z) = \begin{cases} \frac{1}{b-a} & \text{if } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$$

Then mean and variance of uniform noise is given by

$$\mu = \frac{a+b}{2}, \sigma^2 = \frac{(b-a)^2}{12}$$

Impulse (Salt and Pepper) noise

Impulse (salt and pepper) noise is specified as

$$p(z) = \begin{cases} P_a & z = a \\ P_b & z = b \end{cases}$$

E13) There are three methods of estimation of degradation function:
a) Observation
b) Experimentation
c) Modelling

E14) In presence of noise, degradation model as shown in figure 4 becomes

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

After applying inverse filtering

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

Substituting values of $H_R(u,v)$ and $G(u,v)$

$$\begin{aligned}\hat{F}(u,v) &= \frac{1}{H(u,v)}[F(u,v)H(u,v) + N(u,v)]] \\ &= \frac{F(u,v)H(u,v)}{H(u,v)} + \frac{N(n,v)}{H(u,v)} \\ &= F(u,v) + \frac{N(u,v)}{H(u,v)}\end{aligned}$$

Limitations of inverse filtering are:

1) It is an unstable filter

2) It is sensitive to noise. In practice, inverse filter is not popularly used.

E15) This approach includes both the degradation function and power spectrum of noise characteristics in developing the restoration filter. Wiener filter restores the image in the presence of blur as well as noise.

This method is founded by considering image and noise as random variables and objective is to find as estimate $\hat{f}$ of the uncorrupted image f such that the mean square error between them is minimized. This error is given by

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

Where $E\{.\}$ is the expected value of the argument. Noise and image are assumed to be uncorrelated.

$$\Rightarrow \boxed{H_R(u,v) = \frac{S_{fg}(u,v)}{S_{gg}(u,v)}}$$

$$\begin{aligned}\hat{F}(u,v) &= H_R(u,v)G(u,v)\\ &= \left[\frac{H^*(u,v)}{|H(u,v)|^* + k}\right]G(u,v)\\ &= \left[\frac{1}{H^*(u,v)}\frac{|H(u,v)|^2}{|H(u,v)|^2 + k}\right]G(u,v)\end{aligned}$$

k is chosen experimentally and iteratively for best results.

UNIT 7 COLOUR IMAGE PROCESSING

Structure **Page No.**

7.1	Introduction Objectives	188
7.2	Human Vision System	189
7.3	Colour Fundamentals	191
7.4	Colour Models RGB Model CMY and CMYK Model HSI Model	194
7.5	Pseudo-colour models	202
7.6	Summary	203
7.7	Solutions / Answers	204

7.1 INTRODUCTION

The purpose of this unit is to introduce the concepts related to colour image processing. As we have been working with grayscale images till now, we would like to have an in-depth discussion of how colour images are formed and the various colour models that exist.

We shall first discuss the human vision system in Sec. 7.2. A healthy vision system is capable of seeing the world in colour.

As you read further, we shall discuss the various colour models that exist and the advantages and disadvantages of each in Sec. 7.3.

Finally, we shall discuss pseudo-colour processing, also called false colour, which is the process of assigning colours to grey values based on specified conditions. We shall discuss colour processing in Sec. 7.5. We finally summarise the discussion in Sec. 7.6 and in Sec. 7.7, we give the solutions/answers/hints to the exercises.

Now we shall list the objectives of this unit. After going through the unit, please read this list again and make sure that you have achieved the objectives.

Objectives

After studying this unit, you should be able to:

- to differentiate between thousands of thousands of colours and their shades in the colour images.
- To define the different colour models and use them as per the requirements.
- To apply different pseudo colour models

A colour image is a powerful source of information. Human visual system has the ability to differentiate between hundreds of colours and their shades. Therefore, colour images contain a large amount of extra information compared to grey-scale images, that give a better understanding of the contents of the image, for example, in object detection and segmentation. If an image is captured by a full-colour sensor, then the resulting image is a full colour image.

A grayscale image can be converted into a colour image using the technique of pseudo-colour processing, where each intensity is assigned a colour.

Full colour image processing is primarily used in most applications such as visualisation and publishing. We start with discussion on human vision system in the following section.

7.2 HUMAN VISION SYSTEM

The human eye is nearly spherical in shape with a 20mm diameter on an average, as shown in Fig. 4.1. The three main parts of the eye are:

i) The cornea and sclera outer cover,

ii) the choroid and

iii) the retina.

Let us discuss them one by one briefly.

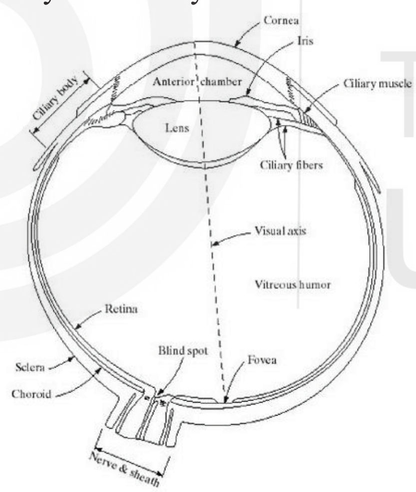


Fig.1: Structure of the human eye source

i) As you see in Fig. 1, the sclera is an opaque member that encloses theoptic globe all around, except at the anterior end, which is covered by the cornea. The cornea is a tough, transparent cover of the anterior chamber.

ii) Choroid is the layer under the sclera. The membrane choroid contains a network of blood vessels. These blood vessels form the major source of nutrition for the eyes. If the choroid is damaged and inflamed, it can restrict blood flow in the eye, resulting in serious damage of the eye cells. The role of the choroid is also to control the amount of light entering the eye as well as reduce the backscatter inside the eye.

The choroid is divided into two parts:

a. The ciliary muscles which relax and tighten to enable the lens to focus by changing its shape,

b. The iris diaphragm, that contracts and expands to control the amount of light that enters the eye.

The lens is a transparent, biconvex structure that helps to refract light into the eye such that the image is formed on the retina. The lens is flexible and ca change shape to change the focal length of the eye. THSI ensures that objects at various distances can be focussed upon and their images can be formed on the retina.

iii) The retina is the innermost membrane of the eye. It lines the wall of the complete posterior portion of the eye. The retina can be thought of as the image plane in the eye, since on properly focussing the eye on an object, light from that object is passed through the lens such that the image is formed on the retina.

The retina consists of two types of cells called rods and cones. The cones are highly sensitive to colour and are around 6-7 million in a human eye. The cones are located on the focea which is the central portion of the retina.

However, there are 75-150 million rod cells which are completely distributed all over the retina. The rod cells give the overall picture of the object in the scene, and reduce the amount of detail. Rods are also responsible for low light vision, also known as **SCOTOPIC vision**, while cones are responsible for bright light vision, also known as **PHOTOPIC vision**.

In the human eye, the distance between the retina and lens, that is the focal length, varies between 14 and 17 mm as the refractive power of the lens increases from min to max. For a nearby object, the lens is most strongly refractive. Moreover, the lens of the eye is very flexible and is flattened by controlling muscles to enable the eye to focus on distant objects.

To allow the eye to focus on objects close to the eye, the controlling muscles allow the lens to become thicker.

Here, you might be wondering how human eye adapts to different levels of brightness and how it discriminates various levels of brightness. The answer to your question is given below.

Brightness Adaptation and Discrimination: Human vision system is highly complex and can adapt to an enormous range of light intensity levels-of the order of $10^{\wedge}10$. The range starts from the scotopic threshold and goes upto the glare limit. The subjective brightness, the perceived intensity by the human eye, has been experimentally found to be a logarithmic function of the light intensity that falls on the eye.

Since the human eye cannot interpret this dynamic range simultaneously, brightness adaptation is carried out by the eye. The eye can discriminate only a small range of distinct intensity levels simultaneously. Brightness adaption level is the current sensitivity level of a human eye for a given set of conditions.

Now, try the following exercises.

E1) If an observer is looking at a tree that is 100m far and if h is the height of the tree in mm in the retinal image, what is h?

So, by now you know the fundamental concepts about human vision system. In the following section, we are going to highlight various colour models. You must have heard about some of them in your day-to-day life.

7.3 COLOUR FUNDAMENTALS

Every colour is defined using three quantities that are independent of each other, however, taken together they define a particular shade of a colour. These quantities are:

i) **Hue:** Hue component of a colour is defined by the dominant wavelength. The wavelength range on the electromagnetic spectrum that defines the visible colour spectrum lies between 400nm [nm is nanometre] that represents the violet colour and 700nm that represents the red colour as can be seen in Fig. 2.

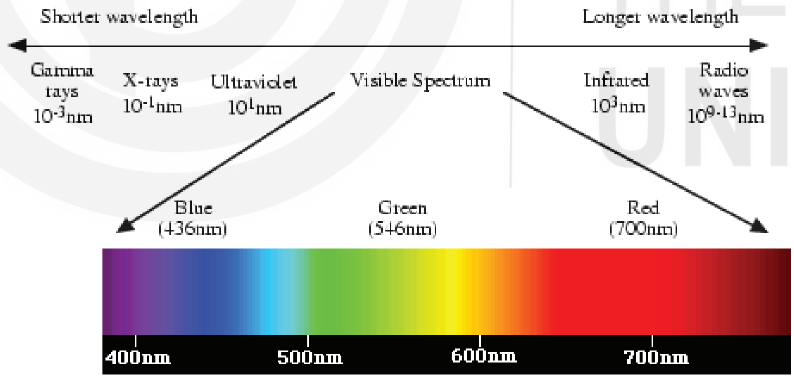


Fig. 2: Part of the electromagnetic spectrum that shows the visible spectrum Source:

ii) **Saturation**: The excitation purity of the colour is determined by the quantity known as **saturation**.It is dependent on the amount of white light that is mixed with hue of that colour. A fully saturated colour implies that no white light is mixed with that hue.

iii) **Chromaticity**: The sum of hue and saturation constitutes the *chromaticity* of the colour. Therefore, if there is no colour, it is called achromatic light.

iv) **Intensity**: The amount of light actually present defines the *intensity*. Therefore, intensity is a physical quantity. If more light is present, the colour is more intense. Achromatic light has only intensity but no colour. Grayscale images have only intensity

v) **Luminance or Brightness**: The perception of colour is the quantity known as luminance or brightness. For example, given two colours of the same intensity, such as blue and green, it is perceived that blue is much darker than green.

vi) **Reflectance**: The ability of an object to reflect light, is the reflectance property of the object. The reflectance property determines the colour of the object, since, we see those colours that are reflected back and not the ones that are absorbed. For example, an object that reflects green, absorbs all other colours in the white light spectrum except green.

There are about 6 to 7 million cones in the human eye and they are responsible for recognising colours. Nearly, 65% of the cones recognise red, 33% are sensitive to green and about 2% to blue. Red, green and blue are known as the primary colours and nearly all other colours are seen as a combination of these primary colours. However, there is a difference between the primary colours of light and the primary colours of pigments. The primary colours of light are red, blue and green and they can be added to produce the secondary colours of light that are yellow(red plus green), magenta (red+blue) and cyan (green + blue). Moreover, the primary colours of pigments are said to be those that absorbs a primary colour of light and reflects the other two. Therefore, in tHSI case, the primary colours are cyan, magenta and yellow while the secondary colours are red, blue and green.

For standardisation, in 1931, the *Commission Internationale de l'Éclairage (CIE)* defined specific wavelengths for the three primary colours: red = 700nm, blue: 435.8 nm and green = **546.1** nm. The amounts of red, blue and green required to form a colour are known as the Tristimulus values and are denoted as X, Y and Z.

Given X, Y and Z the tristimulus coefficients which define a colour. If we define x, y and z as the relative values of the primary colours, then these values can be found by

$$x = \frac{X}{X+Y+Z}, \; Y = \frac{Y}{X+Y+Z} \text{ and } z = \frac{Z}{X+Y+Z} \quad \text{... (1)}$$

It is obvious that

$$x + y + z = 1. \quad \text{... (2)}$$

Thus a 2-D diagram is adequate to show the coordinates x and y.

If we specify colours as a composition as x (red) and y (green). Then, given the values of x and y, the value of z (blue) can be computed as:

$$z = 1 - (x + y) \quad \text{... (3)}$$

Here, we can see only two variables are independent. Therefore, we can show these variables in 2-D coordinate system.

The point on the boundary of the chromaticity chart is fully saturated, while as a point moves farther from the boundary, more white light is added and is therefore, less saturated. The saturation is zero at the point of equal energy. A straightine joining any two points in the chromaticity diagram, determines all possible colours that can be obtained by combining the two colours at the endpoints of the segment. This can be extended to combining three colours. The three line segments joining the points pairwise form a triangle and various combinations of the colours at the vertices of this triangle give all colours inside the triangle or on the boundary of the triangle.

To understand this more clearly, we shall discuss few examples.

Example 1: Consider the coordinates of warm white $(0.45, 0.4)$ and the coordinates of deep blue $(0.15, 0.2)$. Find the percentage of the three colours red (X), green (Y) and blue (Z).

Solution: We first find the trichromatic coefficients x, y and z. At the point warm white, $x = 0.45$ and $y = 0.4$, therefore

$$\begin{aligned} z &= 1-(x+y) \\ &= 1-(0.45+0.4) \\ &= 0.15 \end{aligned}$$

Now, we shall find the tristimulus values X, Y and Z.

$$\frac{X}{X+Y+Z} = 0.45$$
$$\frac{Y}{X+Y+Z} = 0.4$$
$$\frac{Z}{X+Y+Z} = 0.15$$

Here, $X:Y:Z = 0.45:0.4:0.15$

Therefore the percentage of each colour would be as follows:

Percentage of red $(X) = 45\%$
Percentage of green $(Y) = 40\%$
Percentage of Blue $(Z) = 15\%$
At the point deep blue, $x = 0.15$, $y = 0.2$, therefore $z = 0.65$.

We can find the percentage of each colour as we found in case of warm white. We get percentage of red colour as 15%, percentage of green colour as 20% and the percentage of blue colour as 65%.

We can see the percentage is justified for each colour name.

***

Example 2: Find the relative percentage of colours warm white and deep blue which mixes the give the colour which lies on the line joining them. Use the coordinates of these points as given in Example 1.

Solution: Let the colour C lies on the line have the coordinate (x, y).

The distance of C from the warm white colour $= \sqrt{(x-0.45)^2 + (y-0.4)^2}$

Similarly, the distance of C from the deep blue colour
$= \sqrt{(0.15-x)^2 + (0.2-y)^2}$

The percentage of warm white in

$$C = \frac{\sqrt{(x-0.45)^2 + (y-0.4)^2} - \sqrt{(0.15-x)^2 + (0.2-y)^2}}{\sqrt{(0.45-0.15)^2 + (0.4-0.2)^2}} \times 100$$

This expression can be used to find the percentage of warm white colour at C by substituting the coordinates of the point C as per the situation. Also, the percentage of the deep blue colour would be (100 -percentage of warm white colour).

***

Now try the following exercises.

E2) Derive an expression to find the percentage of each colours $C_1, C_2,$ and C_3 at the point C which lies within the triangle having vertices as $C_1, C_2,$ and C_3.

In the following section, we shall discuss the most commonly used colour models such as the RGB (red, green, blue), CMY (Cyan, magenta, yellow) and HSI (hue, saturation, intensity).

7.4 COLOUR MODELS

Colour models or Colour spaces or Colour systems have been introduced so as to be able to specify each colour in a generally accepted manner. There are various colour models or colour spaces. Each colour space specifies a particular colour in a standard manner, by specifying a 3-D coordinate system and a subspace that contains all possible colours in that colour model. Then, each colour in that colour space is represented as a point in that subspace, given by three coordinates (x, y, z). These colour models are either oriented towards specific hardware or image processing applications. In this section, we shall discuss three important colour models and the conversion of one colour model into other.

Before we discuss each colour model, let us discuss the principles of absorption of colours of any model by human eye.

i) The human eye has absorption characteristics of colours and recognises them as variables. Thus, the colours red (R), green (G) and blue (B) are called **primary colours** of light.

ii) Secondary colours of light are produced by adding primary colours. For example red and blue produces magenta, red and green produces yellow, green and blue produces cyan, etc.

iii) Proportion of primary and secondary colours in appropriate amount produces white light.

Now, let us discuss each model separately.

7.4.1 The RGB Model

The RGB colour is based on a cartesian coordinate system, where the colour subspace is a cube with axes representing red, green and blue. A colour in the RGB model is therefore, specified as a 3-tuple (R, G, B) where, R, G and B represent the amount of red, green and blue, respectively, present in that colour. The geometry of the RGB colour model is a cube as shown in Fig. 3. It is assumed that all colour values are normalised, that is, it is a unit cube and all values of R, G and B lie between 0 and 1. It is clear from Fig. 3, that it is a model of a cube with the three axes for Red, Green and blue. The grayscale values lies on the diagonal of the cube, which joins the black $(0,0,0)$ and white $(1,1,1)$ vertices of the cube.

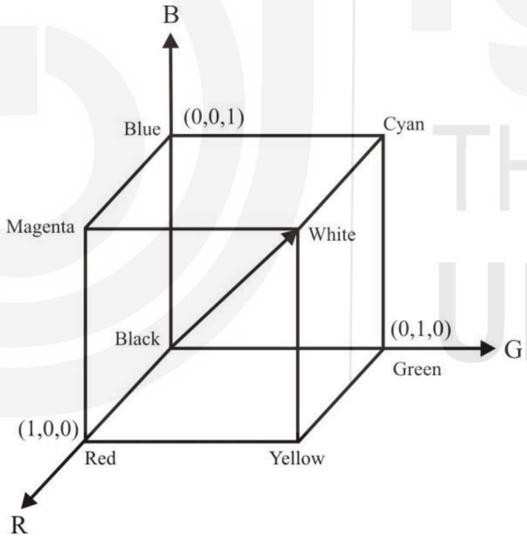


Fig. 3: The RGB colour (Image taken from [1])

A colour image in the RGB model consists of three images corresponding to each of the three colours: Red (R), Green (G) and Blue (B) colours. These three images combine to form one composite colour image on a monitor. To covert an RGB image to a grayscale image, the intensity of the gray-pixel is given by the average of R, G and B values. The RGB colour model is mainly used for colour monitors and screens.

Now the question arises how do we find the composite colour in RGB colour model at any point. For this we follow the following steps:

Step 1: Pixel depth is the number of bits used to represent each pixel. If an image in RGB model has 8-bit image in each of its three colours, then each

RGB pixel has a depth of 3 image planes $\times$ 8-bit per plane that is 24 bits. This gives rise to 2^{24} colour shades.

Step 2: We fix one of the three colours and let the other two colours to vary. Suppose we fix $R = 127$ and let G and B to vary. Then the colour at any point on the plane parallel to GB plane would be $(127, G, B)$, where, $G, B = 0, 1, ..., 255$.

Example 3: In a RGB image, the R and B components are at mid and the G component is at 1, then which colour would be seen by a person?

Solution: At the given point, we have

$$\frac{R}{2} + \frac{B}{2} + G = \frac{1}{2}(R + G + B) + \frac{G}{2}$$

$$= \text{midgrey} + \frac{1}{2}G.$$

$$= \text{Pure green with some grey component.}$$

***

Now, try an exercise.

E3) How many different shades of grey are there in a colour RGB system if each RGB image is an 8 bit image?

After discussing RGB model, we discuss CMY and CMYK colour models.

7.4.2 The CMY and CMYK Colour Model

Cyan (C), Magenta (M) and Yellow (Y) are the primary colours of pigments and the secondary colours of light. The CMY model is a subtractive model, implying that it subtracts a colour from the white light and reflects the rest. For example, when white light is reflected on cyan, it subtracts Red and reflects the rest. While RGB is an additive model, where something is added to black (0,0,0) to get the desired colour CMY is a **subtractive model**. The conversion between CMY and RGB model is given by the Equation below.

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \qquad \text{... (4)}$$

where, the RGB values have been normalised. THSI also gives a method to convert from RGB to CMY to enable printing hardcopy, since the CMY model is used by printers and plotters.

(a) (b)

Fig. 4: (a) RGB and (b) CMY

Fig. 4(a) shows the RGB model, in which the white colour is produced by adding the three primary colours Red, Green and Blue. Fig. 4 (b) shows the CMY model, where black is obtained as the sum of Cyan, Magenta and Yellow. The inverse relation between the RGB and CMY models are also shown by these two images.

In practice, black is obtained by combining cyan, yellow and magenta, however, this leads to a muddy looking black. For publishing, the black colour plays an important role, therefore, in the CMYK colour model, black is added as the fourth colour, where K stands for black.

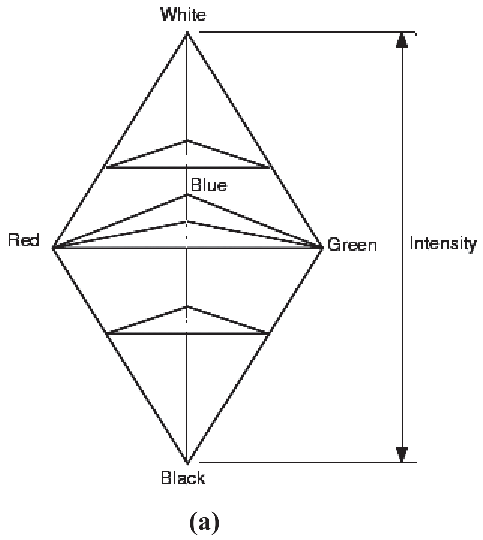
Now check by doing the following exercise, what have you understood.

E4) Why do we get green coloured paint on mixing blue and yellow coloured paints?

Now, let us discuss HSI model.

7.4.3 The HSI Model

This colour model is very close to human colour perception which uses the hue, saturation and intensity components of a colour, when we see a colour, we cannot describe it in terms of the amount of cyan, magenta and yellow that the colour contains. Therefore, the HSI colour model was introduced to enable describing a colour by its hue, saturation and intensity/ brightness. Hue describes the pure colour, saturation describes the degree of purity of that colour while intensity describes the brightness or colour, sensation. In a grayscale image, intensity defines the graylevel. Fig. 6 shows the HSI colour model and the way colours may be specified by this colour model.



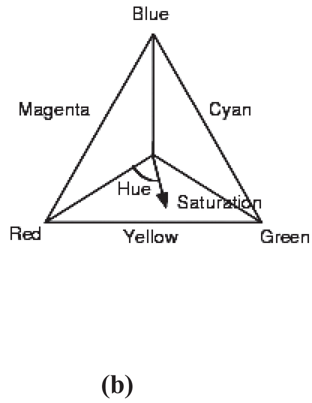


Fig. 5: (a) HSI and (b) RGB

In Fig.5, the HSI colour model is represented and its relation to RGB model is shown in the Fig. 5 (b). The HSI triangle in Fig. 5 (b) shows a slice from the HSI solid at a particular intensity as shown in Fig. 5 (a).

You may notice that in Fig. 5, the hue, saturation and intensity values required to form the HSI colour space can be computed using the RGB values.

Try the following exercises.

E6) Write the full form of HSI and define each of the components.

E7) What is colour space? Mention its classification.

Now we indicate how a RGB colour model is converted to a HSI colour model.

To convert an image in RGB format to HSI colour space, the RGB value of each pixel in the image, is converted to the corresponding HSI value in the following manner. Hue, H is given by

$$H = \begin{cases} \theta & \text{if } B \le G \\ 360 - \theta & \text{if } B > G \end{cases} \quad \ldots (5)$$

where,

$$\theta = \cos^{-1}\left\{ \frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2 + (R-B)(G-B)]^{1/2}} \right\}$$

Saturation, S is given by

$$S = 1 - \frac{3}{(R+G+B)}[\min(R,G,B)]$$

And, intensity, I is given by

$$I = \frac{1}{3}(R+G+B)$$

Where, the RGB values have been normalised in the range [0,1] and the angle θ is measured with respect to the red axis in the HSI space.

Now we would convert HSI colour model to RGB colour space.

Given pixel values in the HSI colour space in the interval [0,1], the RGB values can be computed in the same range. However, depending on the H value, the RGB values are computed in different sectors, based on the separation of the RGB colours by 120° intervals.

First, multiply the H value by 360°, to get the hue value in the interval [0°, 360°].

In the RG sector, H takes the value in the interval [0°, 120°[,

that is, $0^\circ \le H < 120^\circ$ and the RGB values are given by

$$B = I(1-S) \quad \text{... (8)}$$

$$R = I\left[1 + \frac{S\cos H}{\cos(60^\circ - H)}\right] \quad \text{... (9)}$$

$$G = 3I - (R+B) \quad \text{... (10)}$$

In the **GB Sector,** $120^\circ \le H < 240^\circ$ then in this case, we first convert the value of H as

Then, the RGB values are computed as

$$R = I(1-S) \quad \text{... (11)}$$

$$G = I\left[1 + \frac{S\cos H}{\cos(60^\circ - H)}\right] \quad \text{... (12)}$$

$$B = 3I - (R+B) \quad \text{... (13)}$$

In BR sector, when, $240^\circ \le H \le 360^\circ$, we first convert H as $H = H - 240^\circ$

Then, the RGB values are

$$G = I(1-S) \quad \text{... (14)}$$

$$B = I\left[1 + \frac{S\cos H}{\cos(60^{\circ} - H)}\right] \quad \text{... (15)}$$

$$R = 3I - (R + B) \quad \text{... (16)}$$

Example 4: Consider the image with different colours as given in Fig. 6 Write the RGB colours which would appear on monochrome display. You may assume that all colours are at maximum intensity and saturation. Also show each of the colour in black and white considering them as 0 and 255 respectively.

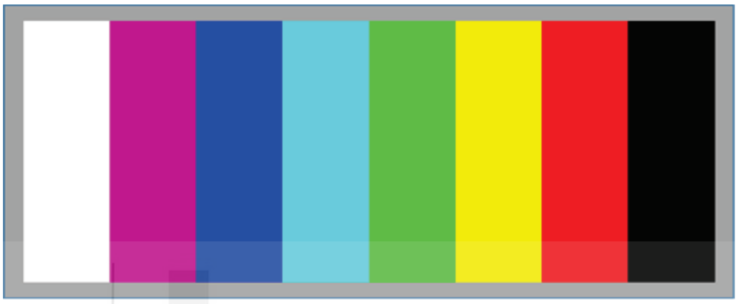


Fig. 6

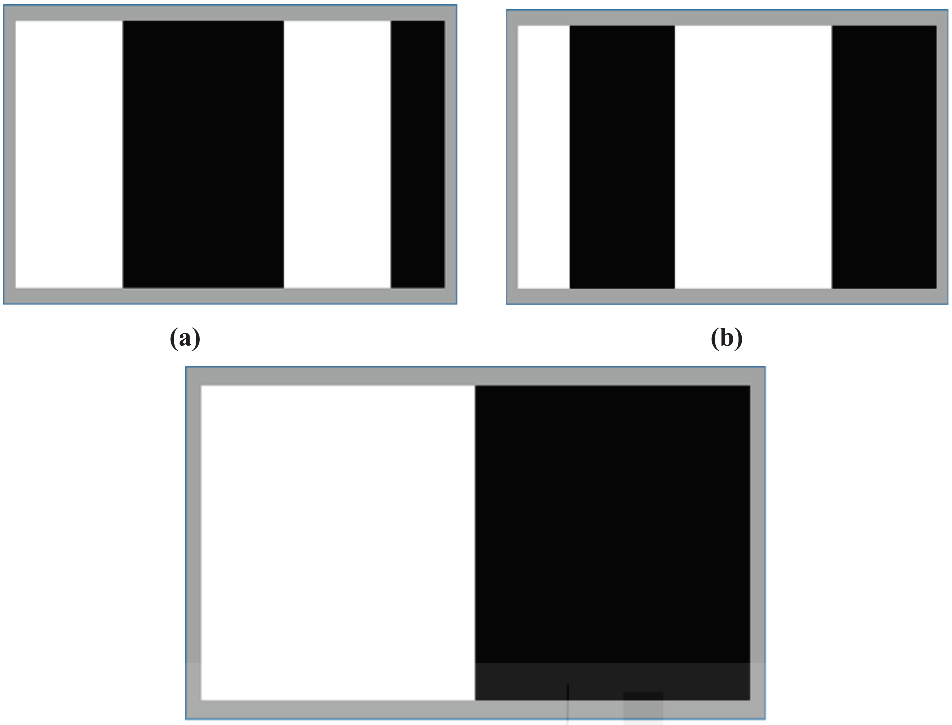
Solution: It is given here that the intensity and saturation are maximum, therefore the value of each of the components RGB would be 0 or 1.

Let us check each colour of the image one by one by starting from the left most colour.

Colour	RGB combination	Intensity/Saturation			Monochrome colours		
		R	G	B	R	G	B
White	R+G+B	1	1	1	255	255	255
Magenta	R+B	1	0	1	255	0	255
Blue	B	0	0	1	0	0	255
Cyan	G+B	0	1	1	0	255	255
Green	G	0	1	0	0	255	0
Yellow	R+G	1	1	0	255	255	0
Red	R	1	0	0	255	0	0
Black	NIL	0	0	0	0	0	0

Now hence forth we shall follow the conversion that 0 represents black and 255 represents white. Also, the grey is represented by 128. You see that the table has R colour series as 255, 255, 0, 0, 255, 255, 0. Thus, it would show W, W, B, B, B, W, W, B in monochrome display, which is shown in Fig. 7 (a).

Similarly monochrome display of green colour would be shown by the series W,B,B,W,W,W,B,B and blue would be shown as W, W, W, W, B, B, B, B, B as shown in Fig. 7 (b) and Fig. 7 (c).



(a) (b)

(c)

Fig. 7

***

Example 5: Let us sketch the HSI components of the image considered in Fig. 6 [Given in Example 4] on a monochrome display.

Solution: We transform HSI by computing values of H, S and I for each colour.

For white $R=1, G=1, B=1$

Using Eqn. (5), we can say that H does not exist as denominator is zero while computing θ.

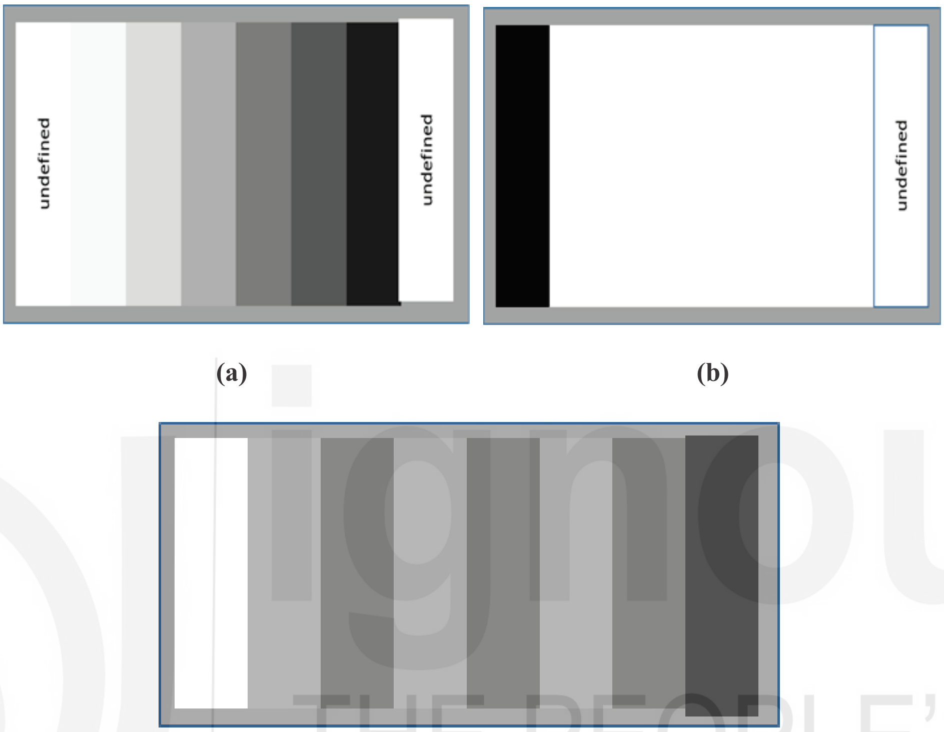
Using Eqn. (7), we get $I=\frac{1}{3}(1+1+1)=1$ and using Eqn. (6), we get

$$S=1-\frac{3}{1+1+1}[\min(1,1,1)]=0.$$

Similarly we can find H, S, I for each of the colours as shown in the following table.

Colour	R	G	B	H	S	I	Monochromatic		
							H	S	I
White	1	1	1	Cannot be computed	0	1	–	0	255
Magenta	1	0	1	$\frac{5}{6}$	1	$\frac{2}{3}$	213	255	170
Blue	0	0	1	$\frac{2}{3}$	1	$\frac{1}{3}$	170	255	85
Cyan	0	1	1	$\frac{1}{2}$	1	$\frac{2}{3}$	128	255	170

Green	0	1	0	$\frac{1}{3}$	1	$\frac{1}{3}$	85	255	85
Yellow	1	1	0	$\frac{1}{6}$	1	$\frac{2}{3}$	43	255	170
Red	1	0	0	0	1	$\frac{1}{3}$	0	255	85
Black	0	0	0	–	0	0	–	–	0



(a) (b)

(c)

Fig. 8

The output is given in Fig. 8 (a), Fig. 8 (b) and Fig. 8 (c) for each of the attribute H,S and I.

***

Now try the following exercises.

E8) Describe how the grey levels vary in RGB primary images that make up the font face of the colour cube.

E9) Transform the RGB cube by its CMY cube. Lable all the vertices. Also, interpret the colours at the edges with respect to saturation.

In the following section, we discuss pseudocolour image processing.

7.5 PSEUDOCOLOUR IMAGE PROCESSING

Pseudocolour image processing is the process of assigning colour to each pixel of a grayscale image based on specific conditions. As mentioned above, colour carries with it a large amount of information regarding the objects that we are viewing and therefore, for better visualisation, converting a grayscale image to a colour image helps in improved interpretation of the image.

Intensity slicing or density slicing is one of the simplest forms of pseudocolour image processing technique. In this technique, the image is interpreted as a 3D function and can be imagined as a set of 2D grid which are parallel to the coordinate planes and placed at each intensity value. Each plane can then be thought of as a slice of the image function in the area of intersection. For example, the plane at $f(x,y)=I_i$ slices the image function into two parts. Then, any pixel whose graylevel is on or above the plane can be coded in one colour and whose graylevel is below the plane can be coded in another colour, thereby converting the grayscale image into a two colour image.

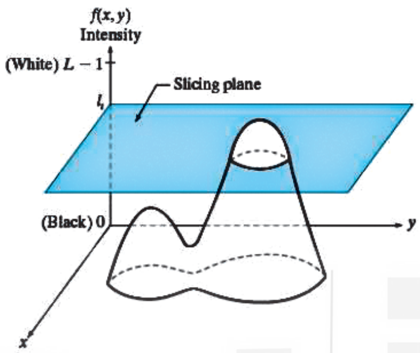


Fig. 9: Intensity slicing technique represented as slicing planes perpendicular to the intensity axis at various intensity levels.

In general, intensity slicing technique is given as follows:

Let $[0, L-1]$ be the existing grayscale values such that black is represented by l_0, that is, $f(x,y)=0$ and white is represented by l_{L-1}, that is, $f(x,y)=L-1$. Then, we define P planes such that they are perpendicular to the intensity axis at levels $l_1, l_2, ..., l_P$, where $0<P<L-1$. These P planes are parallel to the coordinate plane. The grayscale intensity levels are partitioned into $P+1$ intervals, namely, $V_1, V_2, ...V_{P+1}$. Then, assign a colour to each pixel in the following manner

$$f(x,y)=c_k \text{ if } f(x,y)\in V_k$$

where, V_k is the k^{th} intensity interval defined by planes at levels $lk-1$ and lk and c_k is the colour associated with V_k.

You may try an exercise:

E10) Define an application of intensity level slicing.

Now let us, summarise what we have discussed in this unit.

7.6 SUMMARY

In this unit, we discussed the following points:

1. The need for colour image processing. Since the human eye has the wonderful capability of seeing millions of colour, we realise that colour gives a large amount of information about the objects and scene in the images.

2. We first discussed the structure of the human eye and then the tristimulus theory that connects the perception of colour with the various colour models that exist.
3. We then discussed the main colour models or colour spaces that are mainly used in both TV and print.

7.7 SOLUTIONS/ANSWERS

E1) Since when object is far, the focal length is 17 mm for the human eye, therefore, $15/100 = h/17 \Rightarrow h(17*15)/100 = 2.55\,m$

E2) THSI problem is the extention of the problem solved in Example 2. Here, we consider two possibilities.

i) When the point C at which percentage of colours C_1, C_2 and C_3 to be found is on the sides of triangle. In this case the percentage is found by considering the point on the line joining the corresponding vertices as we solved in Example 2. There would be 0% from the vertex which does not lie on the line. For example, if the point lies on the line joining C_1 and C_2, then the percentage of C_1 and C_2 can be found as given below.

Let the coordinates of C_1 be $(x_1, y_1), C_2$ be $(x_2, y_2), C_3$ be (x_3, y_3) and C be (x, y).

Percentage of C_1 in

$$C = \frac{\sqrt{(x - x_1)^2 + (y - y_1)^2} - \sqrt{(x_2 - x)^2 + (y_2 - y)^2}}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \times 100$$

Percentage of C_2 in $C = (100 - \text{percentage of } C_1)\%$

Percentage of C_3 in $C = 0\%$.

ii) When the point C does not lie on the sides of the triangle with vertices C_1, C_2 and C_3.
In HSI case we join the point C with any of the vertices say C_3. We follow the following steps.

iii) Join the points C and C_3 and extend the line towards the side C_1C_2. Suppose it intersects C_1C_2 at C_4.

iv) Find the percentage of C_1 and C_2 at C_4.

v) Use the concept that the ratio of C_1 and C_2 will remain same at each of the points on the line C_3C_4.

vi) Now, we can easily find the coordinates of the point C_4 by writing equation of the lines C_1C_2 and C_3C. C_4 is the point of intersection of C_1C_2 and C_3C.

vii) Finally, we can find the percentage of C_4 and C_3 for the colour C.

E3) For an 8-bit image, there are $2^8 = 256$ possible values. A colour will be grey if each of the colour in RGB is same. Therefore, there can be 256 shades of grey.

E4) You can see in Fig. 5, yellow paint is made by combining green and red while imperfections in blue leads to reflection of some amount of green from blue paint also. Therefore, when both blue and yellow are mixed, both reflect the green colour, while all other colours are absorbed. Therefore, green coloured paint results from mixing of blue and yellow paints.

E5) H stands for Hue, which represents dominant colour as observed by an observer and the corresponding wavelength is also dominant. S stands for Saturation, which is the amount of white light mixed with a hue. I stands for intensity which reflects the brightness.

E7) A colour space allows one to represent all the colour perceived by human eye. The colour space can be broadly classified into (i) RGB, (ii) GMY and (iii) HSI colour space.

E8) Each of the components in RGB model would vary from 0 to 255. Here, we are discussing the front face. So, we fix all pixel values in the Red image as 255 and let the columns to vary from 0 to 255 in the green image and rows to vary from 255 to 0 in the blue image.

E9) The vertices would be as given below:

White $= (0,0,0)$

Cyan $= (1,0,0)$

Magenta $= (0,1,0)$

Blue $= (1,1,0)$

Green $= (1,0,1)$

Red $= (0,1,1)$

Black $= (1,1,1)$.

The edges which are free from black or white pixels are fully saturated. The saturation decreases towards the ends having black or white pixel.

E10) Detecting blockages in medical image processing is an application of intensity level slicing.

References

[1] R.C. Gonzoles and R.E. Woods, Digital Image Processing, Addision-wesley, 1992.

[2] A.K. Jain, Fundamentals of Digital Image Processing, PHI.