Image Interpolation-the comparison

DHRUV KALARIA (EC-41) & MEHUL JANI (EC-38)

2010

IN SPATIAL & FREQUENCY DOMAIN

SUBMITTED TO,
MR. Biren P. Patel
Lab Project
DDIT

TABLE OF CONTENTS

Image Interpolation in Spatial & Frequency Domain -The Comparison.

Image Processing is a very vast and a diverse field which is very useful in any of the branch of the modern sciences from Electronics to medical sciences to Satellite image processing and much more. In this document we have tried to define and compare the image interpolation, one of the very basic and widely used theme in the branch concerned.

Image interpolation basically means 'zooming' an image i.e. increasing the pixel components equally/unequally in the row and the column concerned. So, the main problem here arises of when we try to increase the image resolution and not the optical resolution the image quality degrades.
We have taken into account this basic problem and tried to come to a better solution with the help of frequency domain properties.

So without a much ado we try to look into the inner depths of the image Processing.

CHAPTER 2
IMAGE FUNDAMENTALS

Before proceeding further into the sea of the image enhancement and certain terms which seem to be weird at the first look lets discuss some of the very basic terms used in the image processing field.

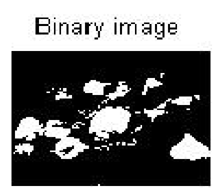
1. Pixel: In digital Imagining, a pixel is a point in raster image. Each Pixel in an image is addressable. Also, Pixel = pix(picture) + el(element)

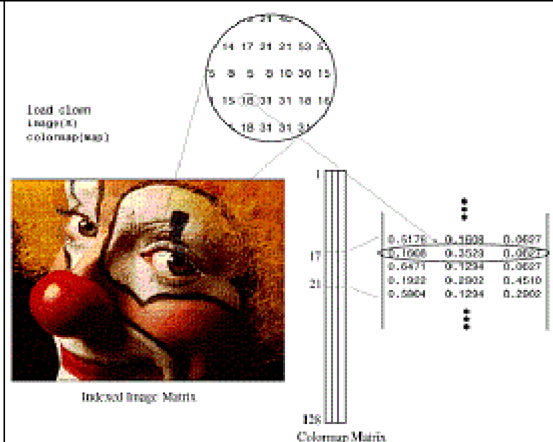
2. Bit Depth: No. of bits required to represent a pixel.
   For Gray Scale Bit Depth is 2 to 8
   For Color Image Bit Depth is 8 to 24 or higher

3. Resolution: No. of pixels in a given neighborhood defines the resolution of an image. Higher the resolution higher the image quality.

Sr. No.	Image Matrix	Resolution	Max. Picture Size
1.	960 x 1200	1.1 MegaPixel	5“ x 7“
2.	1300 x 1600	2 MegaPixel	8“ x 10“
3.	1600 x 2000	3 MegaPixel	9.5“ x 11.5“

4. Types of Images:

Sr. No.	Type of Image	Image
1.	Binary Image: Just two colors black and white to represent the pixel intensities.	
2.	Gray scale Image: It has more than 1 bit information per pixel. Normally it varies from 2to 8 i.e. a maximum of 255 gray shades.	

3.	<u>Indexed Image:</u> It uses two matrices to represent a color image namely: a colormap and an image matrix.	
4.	<u>RGB Image:</u> It uses three matrix to represent a color image by its three components namely: R,G & B.	

5. <u>Types of representation of Image Pixels</u>:

   1. Double type representation:
      Maximally used during the processing of the image at it has very large range of values. As, a drawback it utilizes large memory so it cannot be used for storing purpose.

   2. Unsigned Integer representation:
      Maximally used to store the image as it occupies less size, almost 1/8th size of the double class data type.

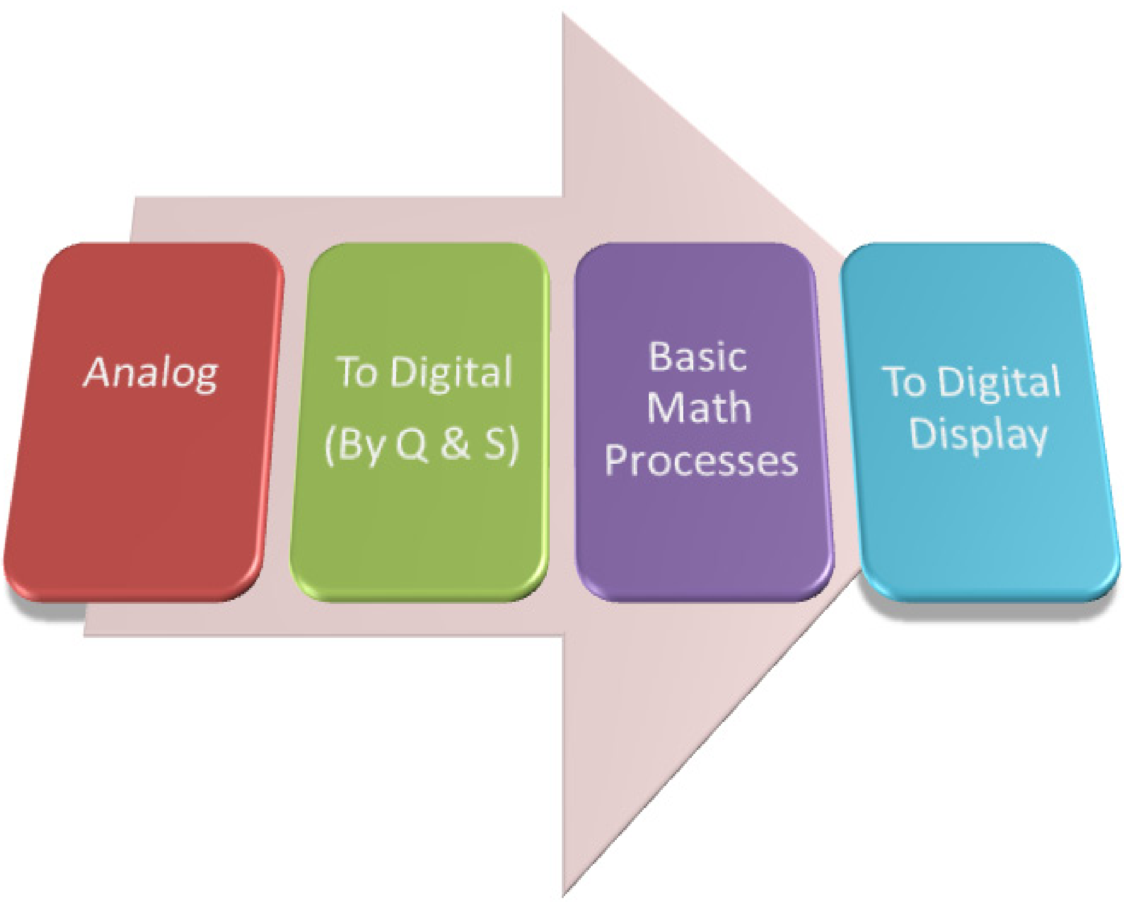
6. <u>Difference between Contrast and Gradient</u>.

   Contrast: It is the difference in pixel intensity of any two pixel in an Image.
   Gradient: It is a gradual transition from one value to another in Sequence.

CHAPTER 3:
IMAGE RESOLUTION

As, discussed in the previous chapter resolution is the number of pixels present in some defined neighborhood. So the resolution depends upon the sampling frequency of the imaging device if its higher than the number of pixels increases. This is the thing which we observe while buying a digital camera from ‘Amazon’ that it has how many Megapixels.

The basic algorithm goes like this:

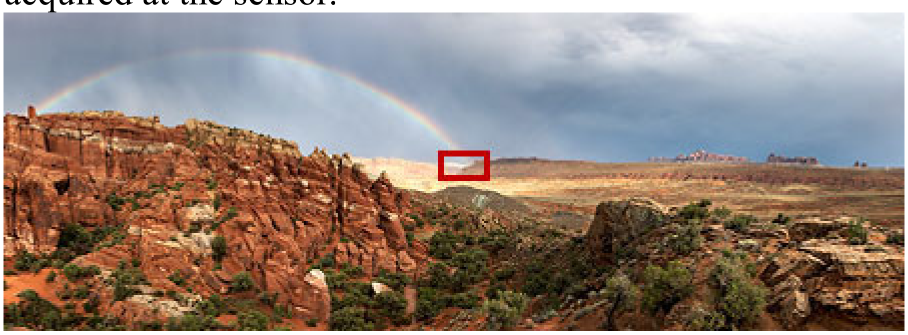


Metrics of Resolution:

1. LPI:
   Lines per inches.
2. DPI:
   Dots per inches.
   Printers use this format to provide the resolution.
   The Inkjet Printers print at 300 to 2400dpi
3. PPI:
   Pixels per inches.
   Monitors basically use this format to provide the resolution.
   Normally its value is 72ppi.

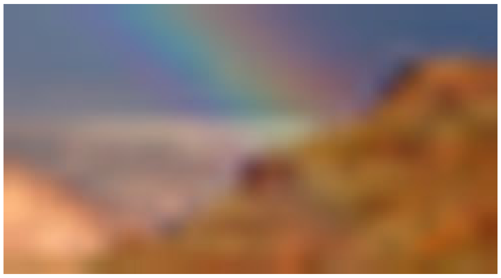
ON OPTICAL vs. DIGITAL ZOOM

Many compact digital cameras can perform both an optical and a digital zoom. A camera performs an optical zoom by moving the zoom lens so that it increases the magnification of light before it even reaches the digital sensor. In contrast, a digital zoom degrades quality by simply interpolating the image-- after it has been acquired at the sensor.



10X Optical Zoom

10X Digital Zoom



Even though the photo with digital zoom contains the same number of pixels, the detail is clearly far less than with optical zoom. Digital zoom should be almost entirely avoided, unless it helps to visualize a distant object on your camera's LCD preview screen. Alternatively, if you regularly shoot in JPEG and plan on cropping and enlarging the photo afterwards, digital zoom at least has the benefit of performing the interpolation before any compression artifacts set in. If you find you are needing digital zoom too frequently, purchase a teleconverter add-on, or better yet: a lens with a longer focal length.

CHAPTER 4:

Image Processing Toolbox:

The File Formats:

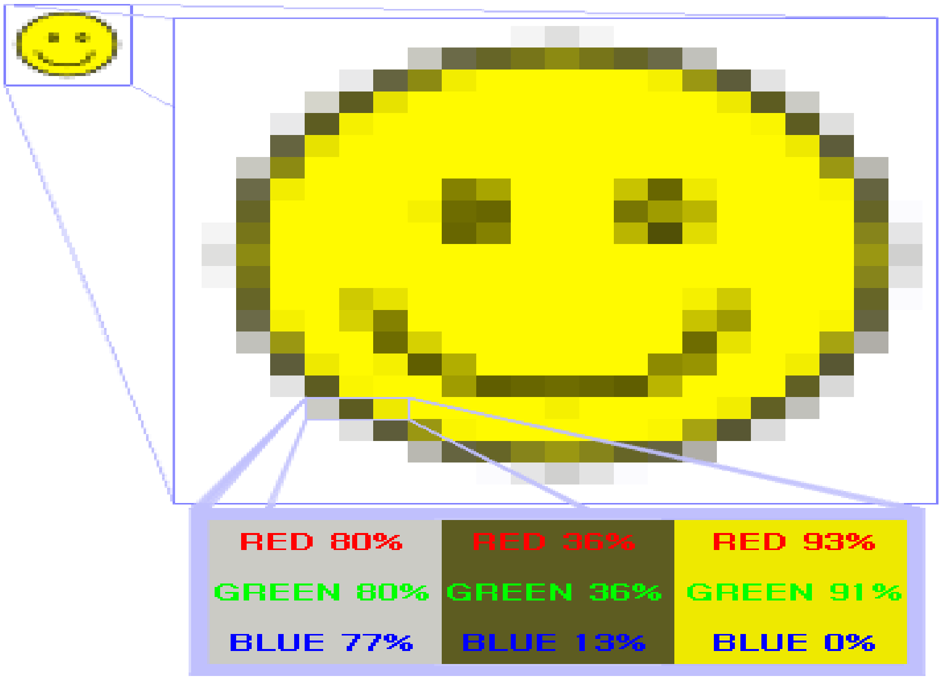
This word has been so general and is been used by even layman extensively. So but with no basics. So, just exposing the veil of what actually they stand for and that there are also file formats other than '.jpg', I would just like to discuss about them in short with their basic applications where they are used according to their property.

There are basically 2 types of images which we come across as far as 2D image graphics is concerned.

1. Raster Graphics
2. Vectored Graphics

Raster Graphics Basics:

Raster graphics are resolution dependent. They cannot scale up to an arbitrary resolution without loss of apparent quality. This property contrasts with the capabilities of vector graphics, which easily scale up to the quality of the device rendering them. Raster graphics deal more practically than vector graphics with photographs and photo-realistic images, while vector graphics often serve better for typesetting or for graphic design. Modern computer-monitors typically display about 72 to 130 pixels per inch (PPI), and some modern consumer printers can resolve 2400 dots per inch (DPI) or more; determining the most appropriate image resolution for a given printer-resolution can pose difficulties, since printed output may have a greater level of detail than a viewer can discern on a monitor. Typically, a resolution of 150 to 300 pixel per inch works well for 4-color process (CMYK) printing.
Some of them are listed below.



JPEG/JFIF

JPEG (Joint Photographic Experts Group) is a compression method; JPEG-compressed images are usually stored in the JFIF (JPEG File Interchange Format) file format. JPEG compression is (in most cases) lossy compression. The JPEG/JFIF filename extension in DOS is JPG (other operating systems may use JPEG). Nearly every digital camera can save images in the JPEG/JFIF format, which supports 8 bits per color (red, green, blue) for a 24-bit total, producing relatively small files. When not too great, the compression does not noticeably detract from the image's quality, but JPEG files suffer generational degradation when repeatedly edited and saved. The JPEG/JFIF format also is used as the image compression algorithm in many Adobe PDF files.

JPEG 2000

JPEG 2000 is a compression standard enabling both lossless and lossy storage. The compression methods used are different from the ones in standard JFIF/JPEG; they improve quality and compression ratios, but also require more computational power to process. JPEG 2000 also adds features that are missing in JPEG. It is not nearly as common as JPEG, but it is used currently in professional movie editing and distribution (e.g., some digital cinemas use JPEG 2000 for individual movie frames).

Exif

The Exif (Exchangeable image file format) format is a file standard similar to the JFIF format with TIFF extensions; it is incorporated in the JPEG-writing software used in most cameras. Its purpose is to record and to standardize the exchange of images with image metadata between digital cameras and editing and viewing software. The metadata are recorded for individual images and include such things as camera settings, time and date, shutter speed, exposure, image size, compression, name of camera, color information, etc. When images are viewed or edited by image editing software, all of this image information can be displayed.

TIFF

The TIFF (Tagged Image File Format) format is a flexible format that normally saves 8 bits or 16 bits per color (red, green, blue) for 24-bit and 48-bit totals, respectively, usually using either the TIFF or TIF filename extension. TIFF's flexibility can be both an advantage and disadvantage, since a reader that reads every type of TIFF file does not exist. TIFFs can be lossy and lossless; some offer relatively good lossless compression for bi-level (black&white) images. Some digital cameras can save in TIFF format, using the LZW compression algorithm for lossless storage. TIFF image format is not widely supported by web browsers. TIFF remains widely accepted as a photograph file standard in the printing business. TIFF can handle device-specific color spaces, such as the CMYK defined by a particular set of printing press inks. OCR (Optical Character Recognition) software packages commonly generate some (often monochromatic) form of TIFF image for scanned text pages.

RAW

RAW refers to a family of raw image formats that are options available on some digital cameras. These formats usually use a lossless or nearly-lossless compression, and produce file sizes much smaller than the TIFF formats of full-size processed images from the same cameras. Although there is a standard raw image format, (ISO 12234-2, TIFF/EP), the raw formats used by most cameras are not standardized or documented, and differ among camera manufacturers. Many graphic programs and image editors may not accept some or all of them, and some older ones have been effectively orphaned already. Adobe's Digital Negative (DNG) specification is an attempt at standardizing a raw image format to be used by cameras, or for archival storage of image data converted from undocumented raw image formats, and is used by several niche and minority camera manufacturers including Pentax, Leica, and Samsung. The raw image formats of more than 230 camera models, including those from manufacturers with the largest market shares such as Canon, Nikon, Sony, and Olympus, can be converted to DNG.[2] DNG was based on ISO 12234-2, TIFF/EP, and ISO's revision of TIFF/EP is reported to be adding Adobe's modifications and developments made for DNG into profile 2 of the new version of the standard.

As far as videocameras are concerned, ARRI's Arriflex D-20 and D-21 cameras provide raw 3K-resolution sensor data with Bayern pattern as still images (one per frame) in a proprietary format (.ari file extension). Red Digital Cinema Camera Company, with its Mysterium sensor family of still and video cameras, uses its proprietary raw format called REDCODE (.R3D extension), which stores still as well as audio+video information in one lossy-compressed file.

PNG

The PNG, (Portable Network Graphics) file format was created as the free, open-source successor to the GIF. The PNG file format supports truecolor (16 million colors) while the GIF supports only 256 colors. The PNG file excels when the image has large, uniformly colored areas. The lossless PNG format is best suited for editing pictures, and the lossy formats, like JPG, are best for the final distribution of photographic images, because in this case JPG files are usually smaller than PNG files. Many older browsers currently do not support the PNG file format; however, with Mozilla Firefox or Internet Explorer 7, all contemporary web browsers now support all common uses of the PNG format, including full 8-bit translucency (Internet Explorer 7 may display odd colors on translucent images ONLY when combined with IE's opacity filter). The Adam7-interlacing allows an early preview, even when only a small percentage of the image data has been transmitted.

PNG provides a patent-free replacement for GIF and can also replace many common uses of TIFF. Indexed-color, grayscale, and truecolor images are supported, plus an optional alpha channel.

PNG is designed to work well in online viewing applications, such as the World Wide Web, so it is fully streamable with a progressive display option. PNG is robust, providing both full file integrity checking and simple detection of common transmission errors. Also, PNG can store gamma and chromaticity data for improved color matching on heterogeneous platforms.

Some programs do not handle PNG gamma correctly, which can cause the images to be saved or displayed darker than they should be.[3]

Animated formats derived from PNG are MNG and APNG. The latter is supported by Firefox and Opera and is backwards compatible with PNG.

GIF

GIF (Graphics Interchange Format) is limited to an 8-bit palette, or 256 colors. This makes the GIF format suitable for storing graphics with relatively few colors such as simple diagrams, shapes, logos and cartoon style images. The GIF format supports animation and is still widely used to provide image animation effects. It also uses a lossless compression that is more effective when large areas have a single color, and ineffective for detailed images or dithered images.

BMP

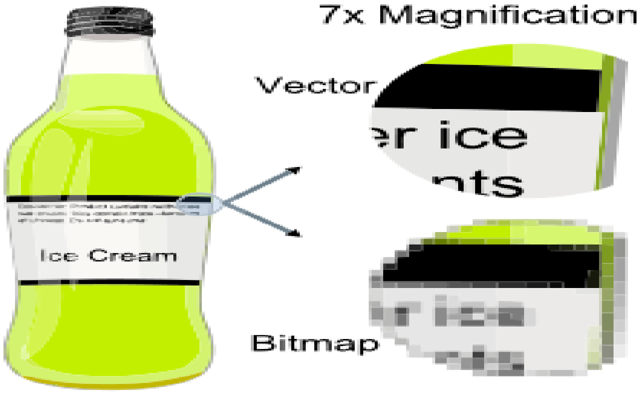
The BMP file format (Windows bitmap) handles graphics files within the Microsoft Windows OS. Typically, BMP files are uncompressed, hence they are large; the advantage is their simplicity and wide acceptance in Windows programs.

PPM, PGM, PBM, PNM

Netpbm format is a family including the portable pixmap file format (PPM), the portable graymap file format (PGM) and the portable bitmap file format (PBM). These are either pure ASCII files or raw binary files with an ASCII header that provide very basic functionality and serve as a lowest-common-denominator for converting pixmap, graymap, or bitmap files between different platforms. Several applications refer to them collectively as the PNM format (Portable Any Map).

Vectored Graphics Basics:
Vector graphics is the use of geometrical primitives such as points, lines, curves, and shapes or polygon(s), which are all based on mathematical equations, to represent images in computer graphics.

Vector graphics formats are complementary to raster graphics, which is the representation of images as an array of pixels, as it is typically used for the representation of photographic images.[1] There are instances when working with vector tools and formats is the best practice, and instances when working with raster tools and formats is the best practice. There are times when both formats come together. An understanding of the advantages and limitations of each technology and the relationship between them is most likely to result in efficient and effective use of tools.



Some of the Basic formats include:

CGM

CGM (Computer Graphics Metafile) is a file format for 2D vector graphics, raster graphics, and text, and is defined by ISO/IEC 8632. All graphical elements can be specified in a textual source file that can be compiled into a binary file or one of two text representations. CGM provides a means of graphics data interchange for computer representation of 2D graphical information independent from any particular application, system, platform, or device. It has been adopted to some extent in the areas of technical illustration and professional design, but has largely been superseded by formats such as SVG and DXF.

SVG

SVG (Scalable Vector Graphics) is an open standard created and developed by the World Wide Web Consortium to address the need (and attempts of several corporations) for a versatile, scriptable and all-purpose vector format for the web and otherwise. The SVG format does not have a compression scheme of its own, but due to the textual nature of XML, an SVG graphic can be compressed using a program such as gzip. Because of its scripting potential, SVG is a key component in web applications: interactive web pages that look and act like applications.

CHAPTER 5

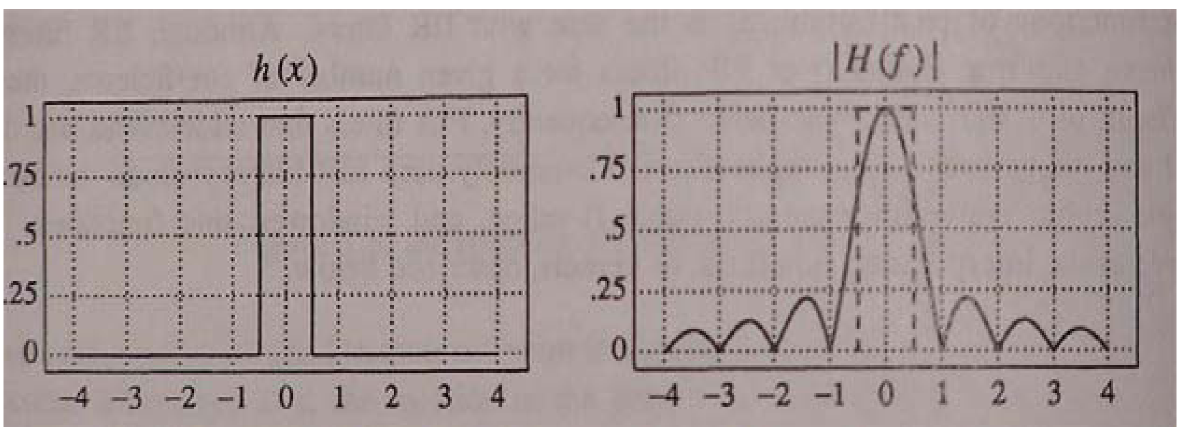
INTERPOLATION KERNELS

The numerical accuracy and computational cost of interpolation algorithms are directly tied to the interpolation kernel. As a result, interpolation kernels are the target of design and analysis. Here, analysis is applied to the 1-D case. Interpolation in 2-D is a simple extension of the 1-D case. In addition, data samples are assumed to be equally spaced along each dimension. This restriction causes no serious problem because images are usually defined on regular grids.

Nearest Neighbor

The simplest interpolation from a computational standpoint is the nearest neighbor, where each interpolated output pixel is assigned the value of the nearest sample point in the input image. This technique is also known as point shioft algorithm and pixel replication. The interpolation kernel for the nearest neighbor algorithm is defined as

$$h(x) = 1;\ 0 \le |x| \le 0.5$$
$$= 0;\ 0.5 \le |x|$$



Convolution in the spatial domain with the rectangle function h is equivalent in the frequency domain to multiplication with a sinc function. Due to the prominent side lobes and infinite extent, a sinc function makes a poor low-pass filter. This technique achieves magnification by pixel replication.
For large-scale changes, nearest neighbor interpolation produces images with blocky effects.

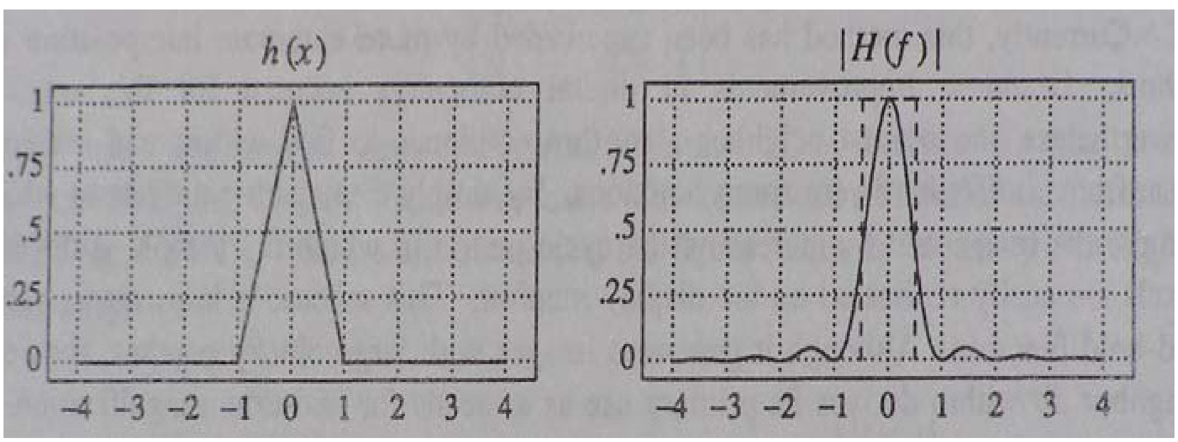
Linear Interpolation

Linear interpolation is a first degree method that passes a straight line through every two consecutive points of the input signal.

In the spatial domain, linear interpolation is equivalent to convolving the sampled input with the following kernel.

$$h(x) \quad = 1 - |x|; 0 \leq |x| \leq 1$$
$$\quad = 0; 1 \leq |x|$$

This kernel is also called triangle filter, roof function or Bartlett window. As shownin Figure 2 the frequency response of the linear interpolation kernel is superior tothat of the nearest neighbor interpolation function. The side lobes are less prominent, so the performance is improved in the stopband. A passband is moderately attenuated, resulting in image smoothing.



Linear interpolation produces reasonably good results at moderate cost. But for even better performance, more complicated algorithms are needed.

Cubic convolution

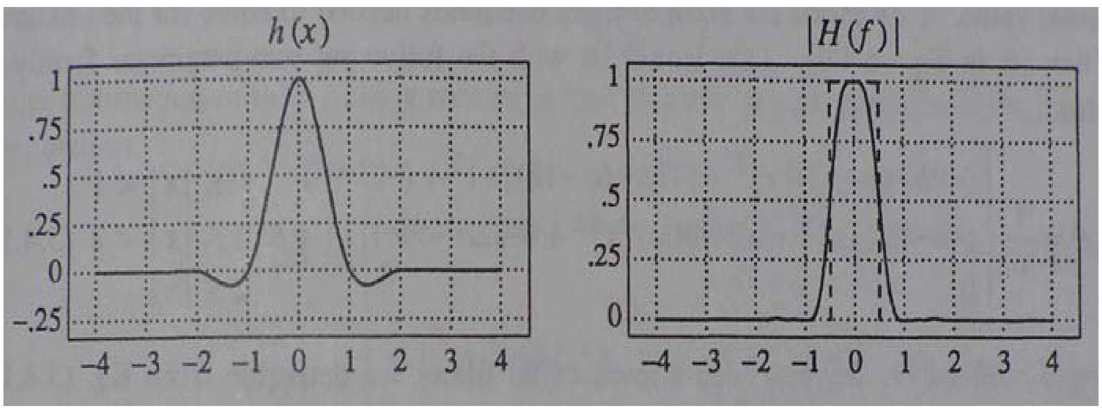
Cubic convolution is a third degree interpolation algorithm that fairly well approximates the theoretically optimum sinc interpolation function. The kernel is composed of piecewise cubic polynomials defined on subintervals (-2, -1), (-1, 0), (0, 1) and (1, 2). Outside this interval the interpolation kernel is zero. For deriving the cubic convolution kernel, we have to solve 8 linear equations with 7 unknown parameters, so the system has one “free” parameter that may be controlled by the user. The kernel is of form:

h(x)=

$$(a+2)|x|^3-(a+3)|x|^2+1 \qquad 0\le |x|<1$$

$$a|x|^3-5a|x|^2+8a|x|-4a \qquad 1\le |x|<2$$

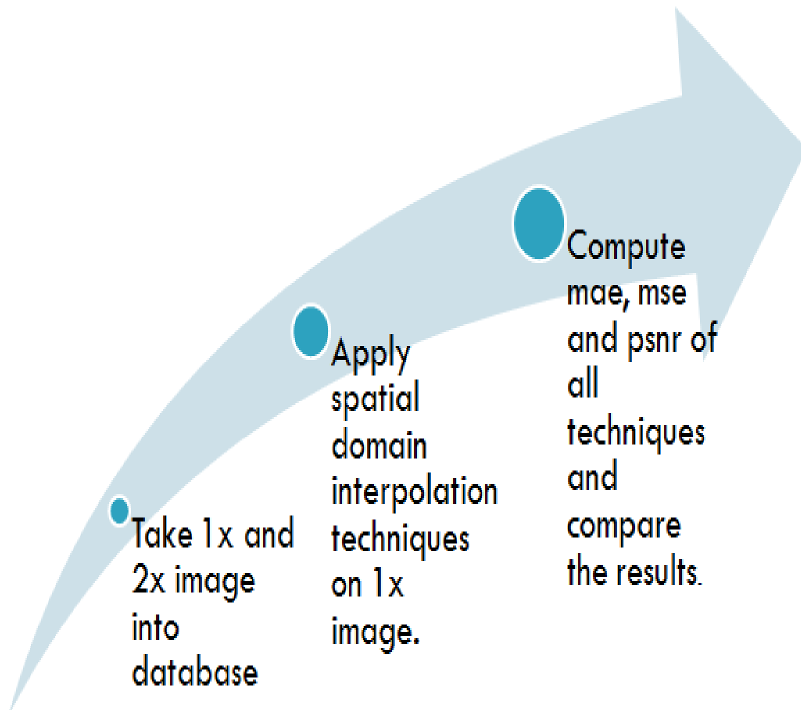
$$0 \qquad 2\le |x|$$



Choices for a are *a*=-1, *a*=-0.75 and *a*=-0.5. The performance of the interpolation kernel depends on *a*, and the frequency content of the image. For different images, different values of the parameter *a* gives the best performance

Chapter 6

Flowchart for Spatial Domain Interpolation



As mentioned in the chart,we have applied this on basic four image types

1. .bmp Image
2. .jpg Image
3. .png Image
4. .tif Image

The basic algorithm for it is shown in the next chapter.

Before, We proceed the Metrics we have taken into consideration are

1. MAE(Mean Absolute Error)
2. MSE(Mean Square Error)
3. PSNR(Peak Signal to Noise Ratio)

1)MAE(Mean Absolute Error)

MAE can be given by the following formula:

```
for i=1:m
        for j=1:n
    r=(r+abs(original(i,j)-scaled(i,j));
    s=(s+original(i,j));
```

end
end
MAE=r/s;

2) MSE(Mean Square Error)

MSE can be represented by the following formula:

MSE=(norm(rescaled-original,2))/(norm(original,2));

3) PSNR(Peak Signal to Noise Ratio)

PSNR can be given by the following formula:

PSNR=$20 * \log (255/\sqrt[2]{MSE})$

Note:
Here, we have taken images in the form of:
y_1x = 64x64
y_2x = 128x128
y_4x = 256x256

Chapter 7

MAE Implemented in Spatial Domain.

```
%@BOUT:THIS PROGRAM DEALS WITH FINDING THE MAE(mean
absolute error) FOR THE
%INTERPOLATED IMAGE USING SPATIAL DOMAIN INTERPOLATION.
%DEVELOPER:DREAMZZZ TECHNOLOGY
%VERSION:1.1.0
%RUN COMPATIBILITY:MATLAB 7.5.0
%Image type Supported:*.bmp,*.jpg,*.tif,*.png

a = imread('tif_1x.tif');
a = double(a);
figure;imshow(a);

b = imread('tif_2x.tif');
b = double(b);
figure;imshow(b)

y = imresize(a,2,'nearest')
y = double(y);
[m n] = size(y)
double r;
r = 0;
double s;
s = 0;

for i = 1:m
    for j = 1:n
        r = (r+abs((b(i,j)-y(i,j))));
        s = (s+b(i,j));
    end
end
mae_nr = r/s;

y = imresize(a,2,'bilinear');
y = double(y);
[m n] = size(y)
double r;
```

```
r = 0;
double s;
s = 0;

for i = 1:m
      for j = 1:n
            r = abs(r+(b(i,j)-y(i,j)));
            s = (s+b(i,j));
      end
end
mae_bilinear = r/s;

y = imresize(a,2,'bicubic')
y = double(y);
[m n] = size(y)
double r;
r=0;
double s;
s=0;

for i = 1:m
      for j = 1:n
            r = (r+abs((b(i,j)-y(i,j))));
            s = (s+b(i,j));
      end
end
mae_bicub = r/s;

PLOT();


%@BOUT:THIS FILE DEALS WITH PRINTING THE IMAGE ONTO THE
SCREEN.
%DEVELOPER:DREAMZZZ TECHNOLOGY
%VERSION:1.1.0
%RUN COMPATIBILITY:MATLAB 7.5.0

function [] = PLOT()
a = imread('bmp_1x.bmp');
b = imread('bmp_2x.bmp');
```

```
subplot(2,3,1);
imshow(a);
title('Original Image');

subplot(2,3,2)
imshow(b);
title('Resized image Image');

c = imresize(a,2,'nearest')
subplot(2,3,4)
imshow(c);
title('Nearest neighbour image')

d = imresize(a,2,'Bilinear')
subplot(2,3,5)
imshow(d);
title('Bilinear interpolated image')

e = imresize(a,2,'bicubic')
subplot(2,3,6)
imshow(e);
title('Bicubic interpolated image')
```

Chapter 8

MSE And PSNR Implemented in Spatial Domian

```
%@BOUT:THIS PROGRAM DEALS WITH FINDING THE MSE and
PSNR(mean square error and peak signal to noise ratio) FOR THE
%INTERPOLATED IMAGE USING SPATIAL DOMAIN INTERPOLATION.
%DEVELOPER:DREAMZZZ TECHNOLOGY
%VERSION:1.1.0
%RUN COMPATIBILITY:MATLAB 7.5.0
%Image Type supported:*.bmp,*.jpg,*.png,*.tif

a = imread('tif_1x.tif');
imshow(a);
a = double(a);


b = imread('tif_2x.tif');
b = double(b);
%figure;imshow(b)

y = imresize(a,2,'nearest');
y = double(y);
mse_nr = (norm(y-b,2))/(norm(b,2));
psnr_nr = 20*log10(255/sqrt(mse_nr));


y = imresize(a,2,'bilinear');
y = double(y);
mse_bil = (norm(y-b,2))/(norm(b,2));
psnr_bil = 20*log10(255/sqrt(mse_bil));

y = imresize(a,2,'bicubic');
y = double(y);
mse_bic=(norm(y-b,2))/(norm(b,2));
psnr_bic = 20*log10(255/sqrt(mse_bic));

PLOT();
```

```matlab
%@BOUT:THIS FILE DEALS WITH PRINTING THE IMAGE ONTO THE
SCREEN.
%DEVELOPER:DREAMZZZ TECHNOLOGY
%VERSION:1.1.0
%RUN COMPATIBILITY:MATLAB 7.5.0

function []=PLOT()
a = imread('tif_1x.tif');
b = imread('tif_2x.tif');

subplot(2,3,1)
imshow(a);
title('Original Image');

subplot(2,3,2)
imshow(b);
title('Resized image Image');

c = imresize(a,2,'nearest')
subplot(2,3,4)
imshow(c);
title('Nearest neighbour image')

d = imresize(a,2,'Bilinear')
subplot(2,3,5)
imshow(d);
title('Bilinear interpolated image')

e = imresize(a,2,'bicubic')
subplot(2,3,6)
imshow(e);
title('Bicubic interpolated image')
```

Chapter 9

Results Obtained in Spatial Domain

1.bmp Image

KERNELS	MAE	MSE	PSNR
NEAREST	0.0975	0.0872	0.058725
BILINEAR	6.95E-05	0.0714	0.05963
BICUBIC	0.0886	0.0688	0.059756

2. JPEG Image:

KERNELS	MAE	MSE	PSNR
NEAREST	0.0687	0.033	62.944
BILINEAR	1.59E-05	0.0283	63.608
BICUBIC	0.0625	0.0278	69.37

Original Image

Resized image Image

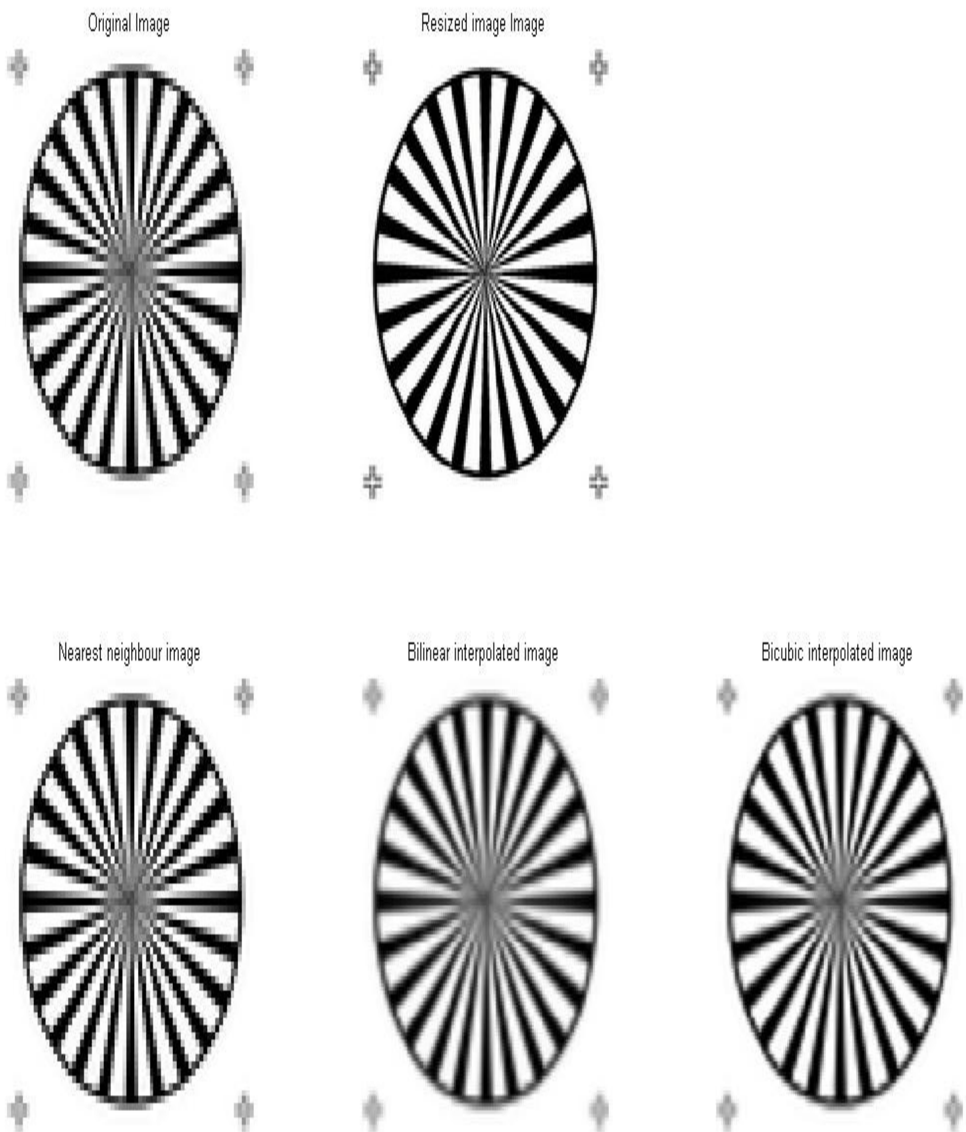
Nearest neighbour image

Bilinear interpolated image

Bicubic interpolated image

3. PNG IMAGE:

KERNELS	MAE	MSE	PSNR
NEAREST	0.1472	0.0816	59.015
BILINEAR	0.0034	0.0775	59.237
BICUBIC	0.1407	0.0669	59.877



4. TIFF IMAGE:

KERNELS	MAE	MSE	PSNR
NEAREST	0.0433	0.0816	58.2314
BILINEAR	5.23E-07	0.0775	59.2365
BICUBIC	0.0158	0.0669	59.8774

Original Image

Resized image Image

Nearest neighbour image

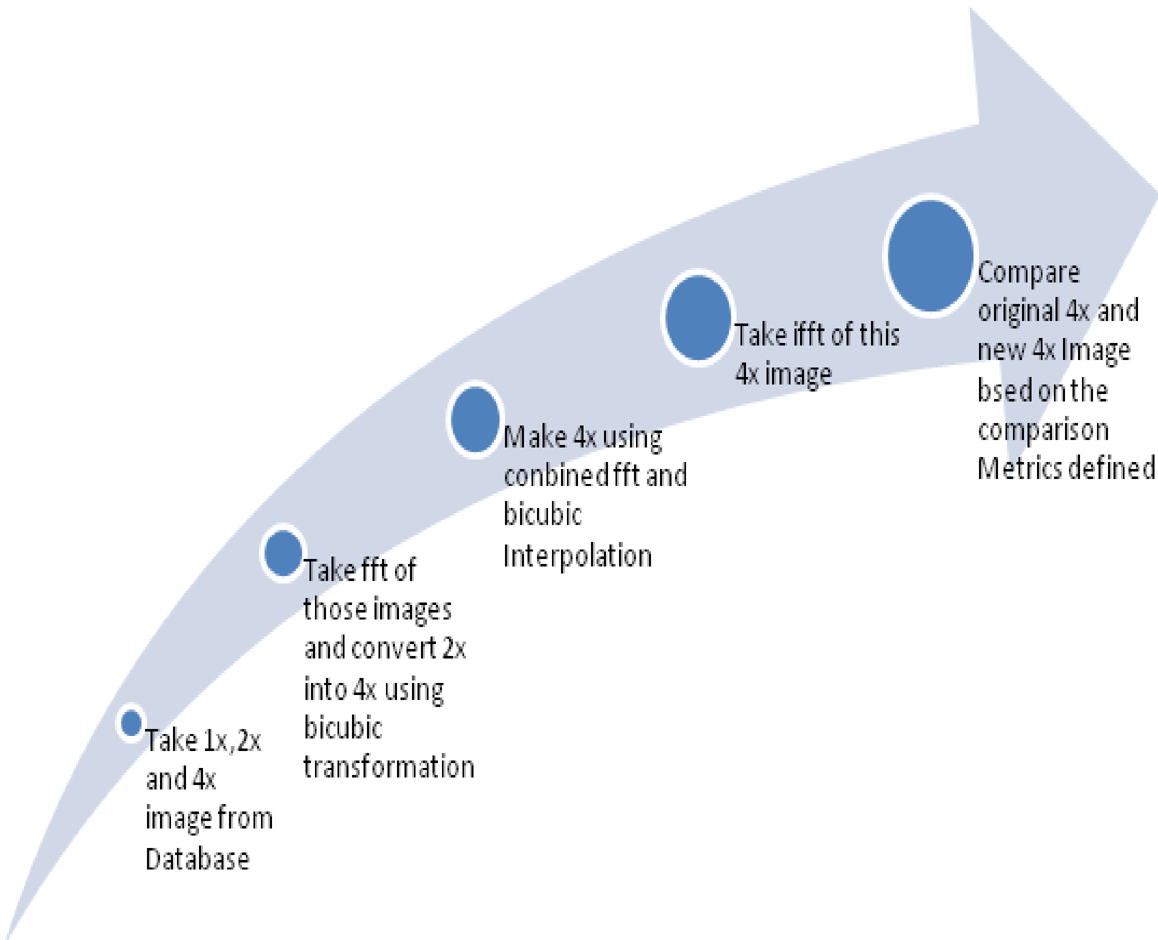
Bilinear interpolated image

Bicubic interpolated image

Chapter 10

Frequency domain Image Interpolation

Flow Chart for Frequency Domain Interpolation



We have applied the same algorithm on the various Image types and calculated the results on the Metrics we have taken into consideration.

CHAPTER11
MSE and PSNR implemented in Frequency Domain

```
%@BOUT:THIS PROGRAM DEALS WITH FINDING THE MSE and
PSNR(mean square error and peak signal to noise ratio) FOR
THE INTERPOLATED IMAGE USING FREQUENCY DOMAIN
INTERPOLATION.
%DEVELOPER:DREAMZZZ TECHNOLOGY
%VERSION:1.1.0
%RUN COMPATIBILITY:MATLAB 7.5.0
%IMAGE TYPE SUPPORTED:*.BMP,*.JPG,*.PNG,*.TIF


org1x = imread('png_1x.png');
org2x = imread('png_2x.png');
org4x = imread('png_4x.png');
org1xt = double(org1x);
org2xt = double(org2x);
org4xt = double(org4x);

forg1x = fft2(org1xt);
forg2x = fft2(org2xt);


double a;
double b;

a=imresize(org2x,2,'bicubic');
imshow(a);
at=double(a);
sc=fft2(at);
%imshow(sc);

for i=1:64
    for j=1:64
        m=2*(i-1)+1;
        n=2*(j-1)+1;

        fact1 = forg2x(m,n)/forg1x(i,j);
```

```
        fact2 = forg2x(m+1,n)/forg1x(i,j);
        fact3 = forg2x(m,n+1)/forg1x(i,j);
        fact4 = forg2x(m+1,n+1)/forg1x(i,j);

        x=4*(i-1)+1;
        y=4*(j-1)+1;
        b(x,y) = forg2x(m,n)*fact1;
        b(x,y+1) = forg2x(m,n)*fact2;
        b(x+1,y) = forg2x(m,n)*fact3;
        b(x+1,y+1) = forg2x(m,n)*fact4;

        b(x,y+2) = forg2x(m,n+1)*fact1;
        b(x,y+3) = forg2x(m,n+1)*fact2;
        b(x+1,y+2) = forg2x(m,n+1)*fact3;
        b(x+1,y+3) = forg2x(m,n+1)*fact4;

        b(x+2,y) = forg2x(m+1,n)*fact1;
        b(x+2,y+1) = forg2x(m+1,n)*fact2;
        b(x+3,y) = forg2x(m+1,n)*fact3;
        b(x+3,y+1) = forg2x(m+1,n)*fact4;

        b(x+2,y+2) = forg2x(m+1,n+1)*fact1;
        b(x+2,y+3) = forg2x(m+1,n+1)*fact2;
        b(x+3,y+2) = forg2x(m+1,n+1)*fact3;
        b(x+3,y+3) = forg2x(m+1,n+1)*fact4;
    end
end


for e=1:1:256
    for f=1:1:256
        if (e*f) < ((512)^2)
            b(e,f)=sc(e,f);
        end
    end
end

new4x=ifft2(b);
new4x=abs(new4x);
new4x1=uint8(new4x);
```

```
subplot(1,2,1);
imshow(new4x1);
title('using fft');
subplot(1,2,2);
imshow(org4x);
title('original')

[m n]=size(new4x)
double r;
r=0;
double s;
s=0;

for i=1:m
    for j=1:n
        r=(r+abs((a(i,j)-new4x(i,j))));
        s=(s+a(i,j));
    end
end
mae_bicub=r/s;


mse = (norm(new4x-org4xt,2))/(norm(org4xt,2));
psnr=20*log10(255/sqrt(mse));
```

Chapter 11

Results in frequency Domain

1. BMP IMAGE

KERNELS	MAE	MSE	PSNR
BICUBIC + FFT	0	0.0131	66.9624

ORIGINAL IMAGE

FFT IMAGE

2. <u>JPEG IMAGE</u>

KERNELS	MAE	MSE	PSNR
BICUBIC + FFT	0	0.0141	66.6478

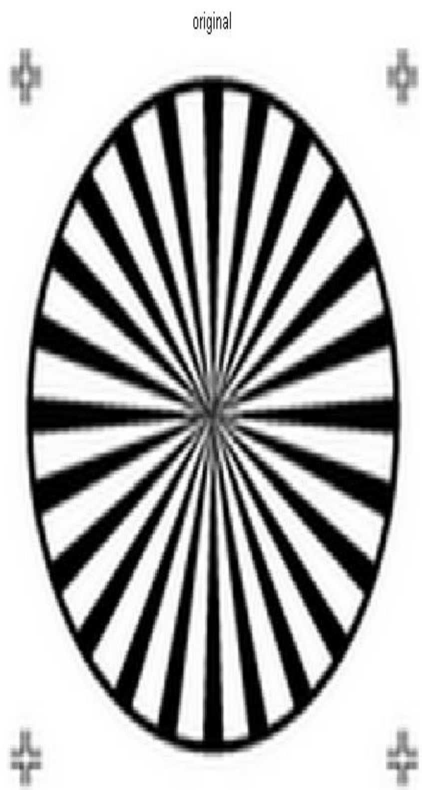
<u>ORIGINAL IMAGE</u>

<u>FFT IMAGE</u>

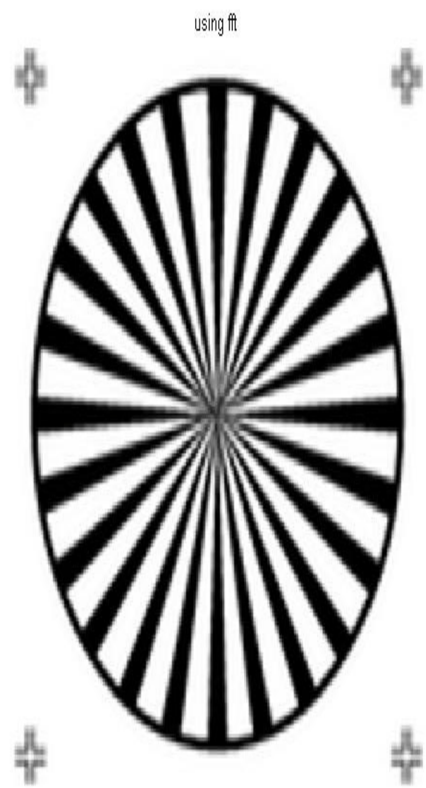
3. <u>PNG IMAGE</u>

KERNELS	MAE	MSE	PSNR
BICUBIC + FFT	0	0.0068	69.8032

ORIGINAL IMAGE



FFT IMAGE



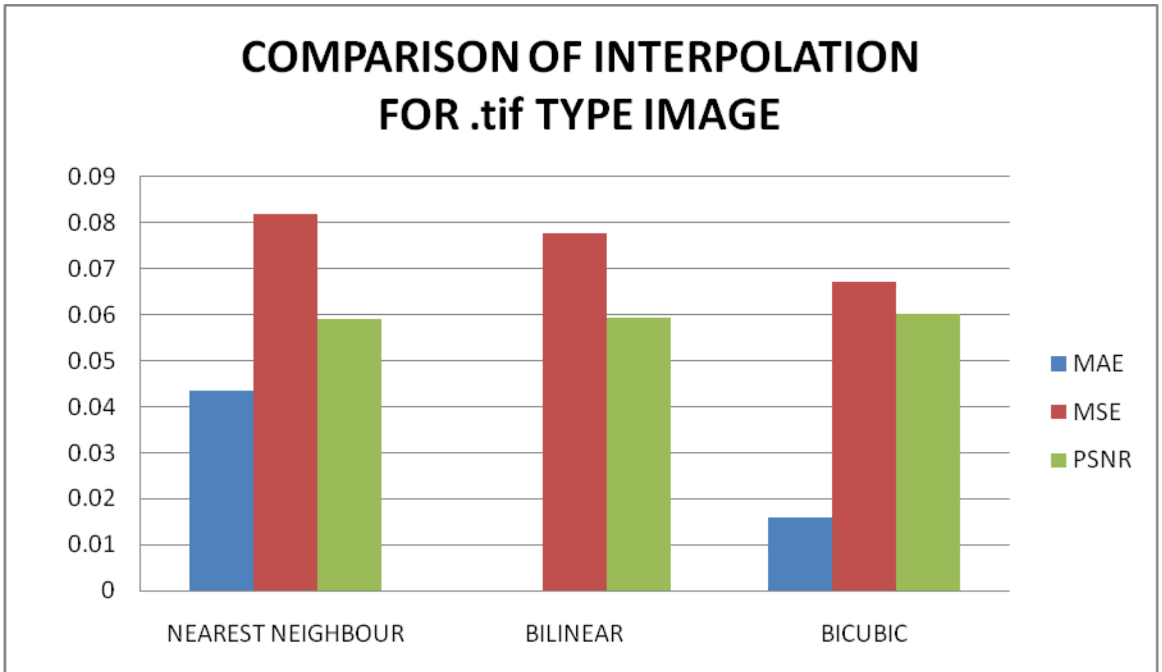
4. TIFF IMAGE

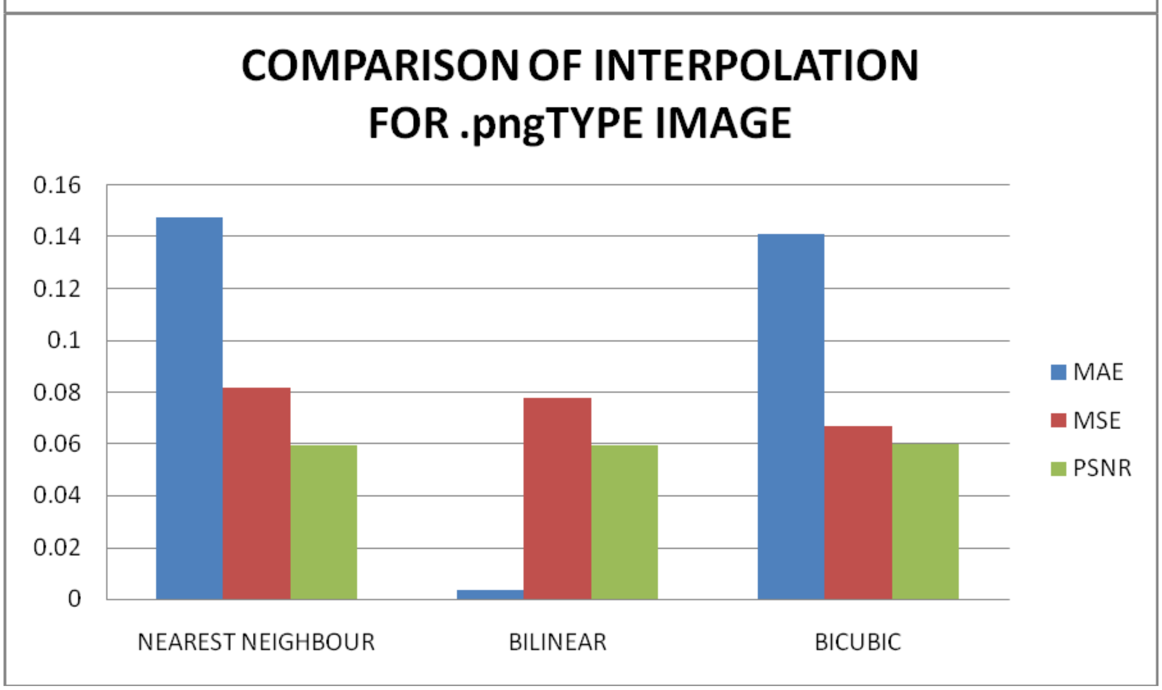
KERNELS	MAE	MSE	PSNR
BICUBIC + FFT	0	0.0246	64.2217

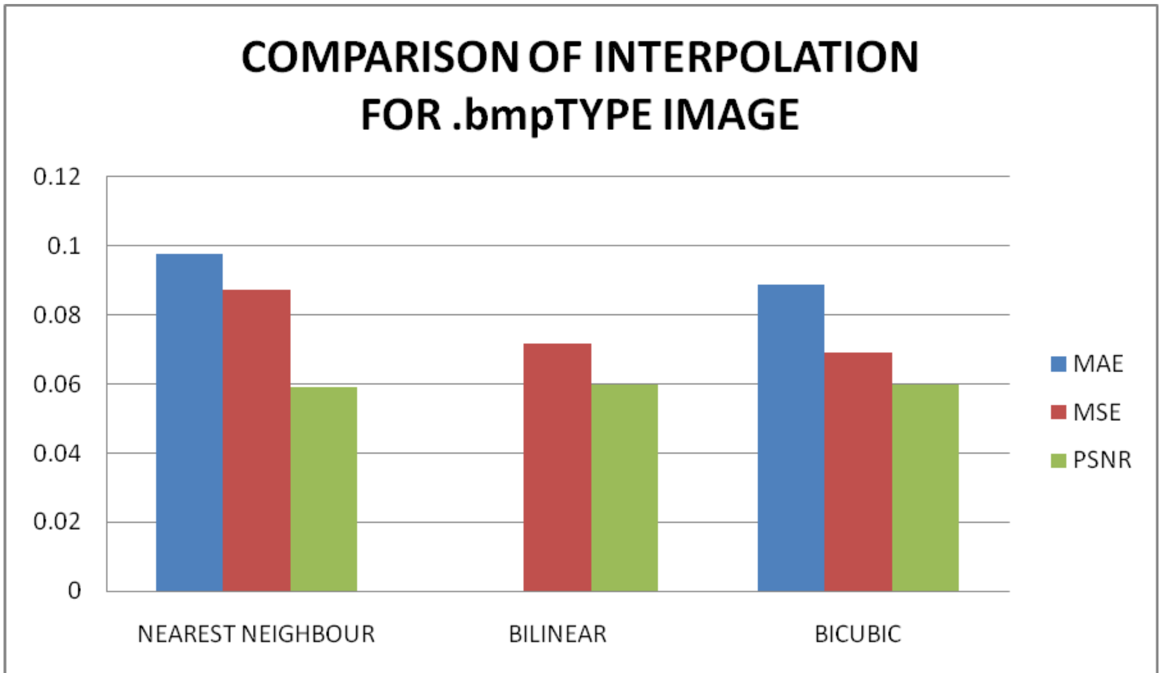
ORIGINAL IMAGE FFT IMAGE

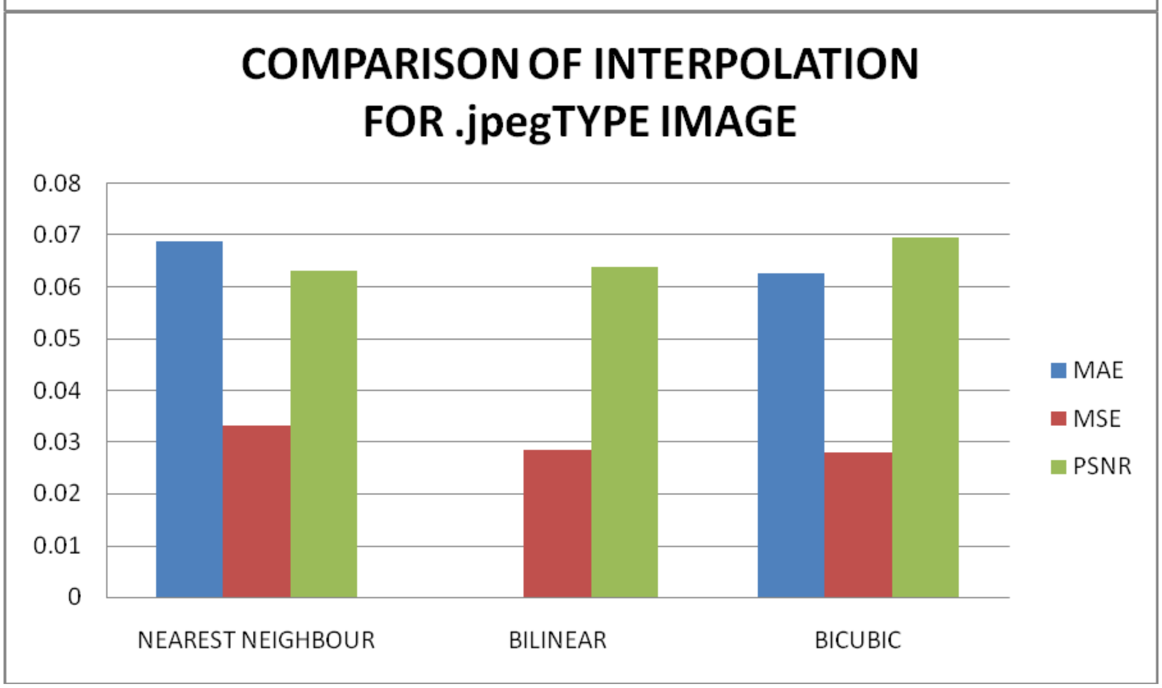
Chapter 12

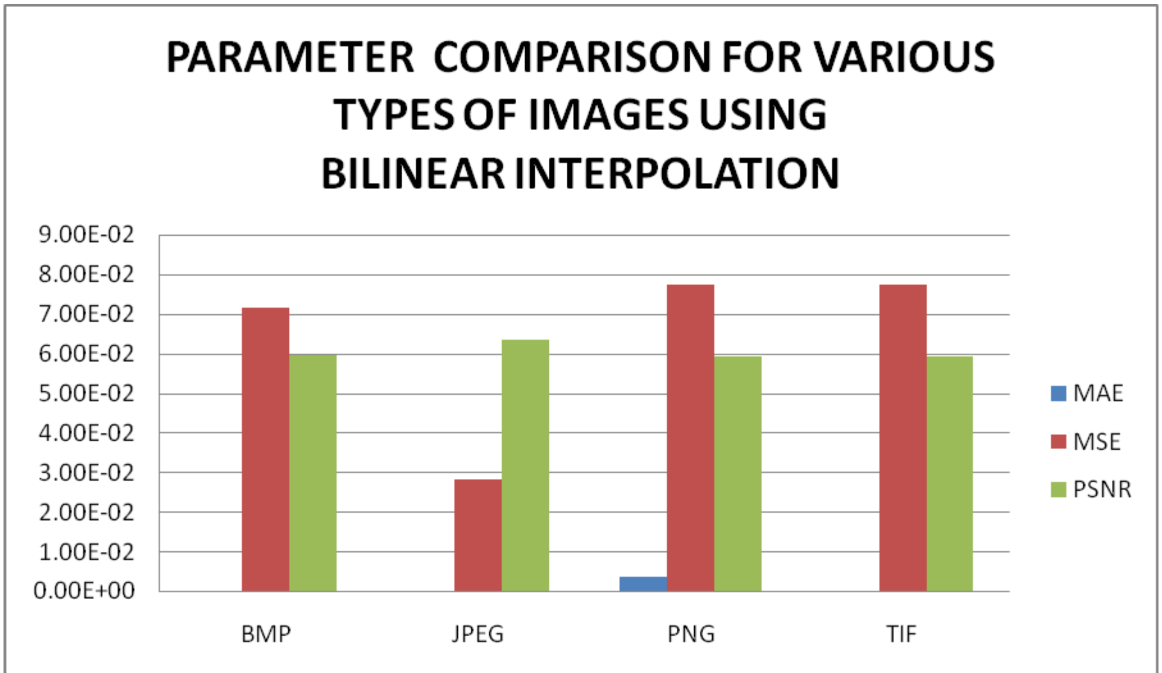
Observations and Comparison graphs

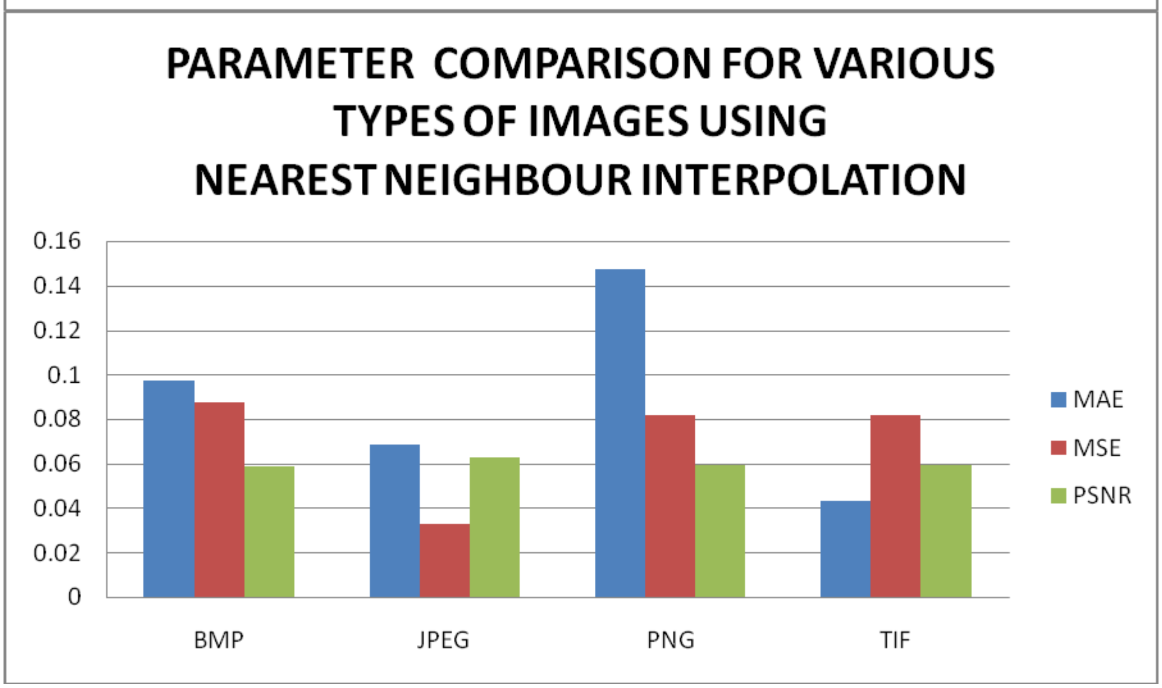


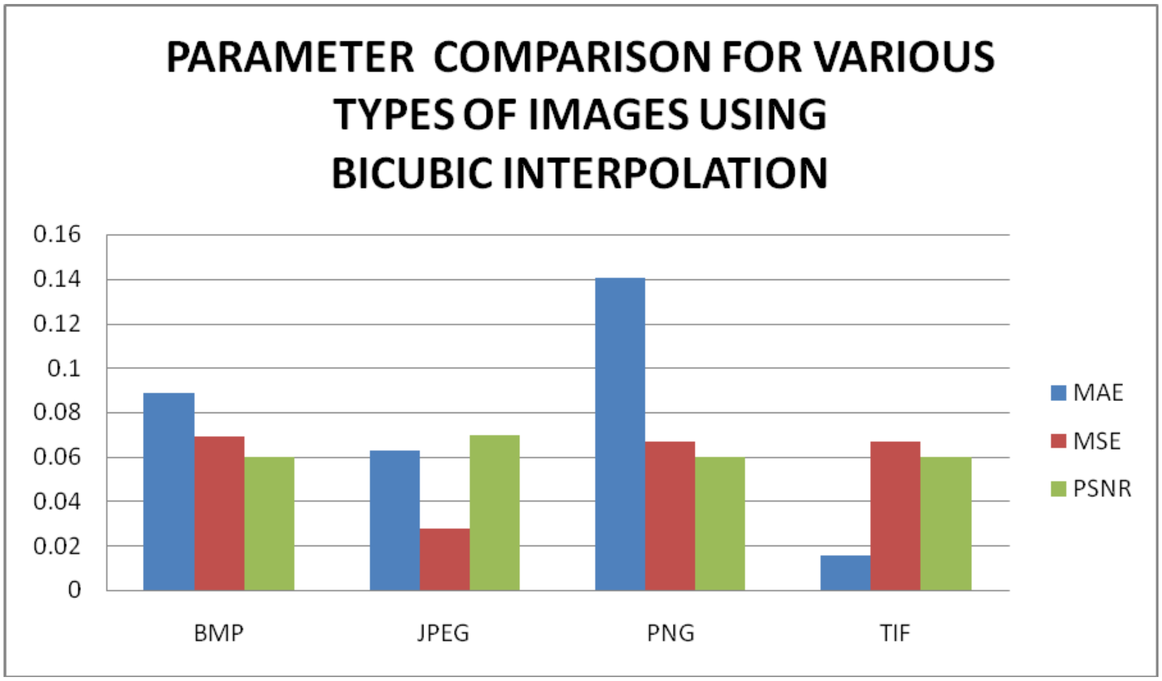


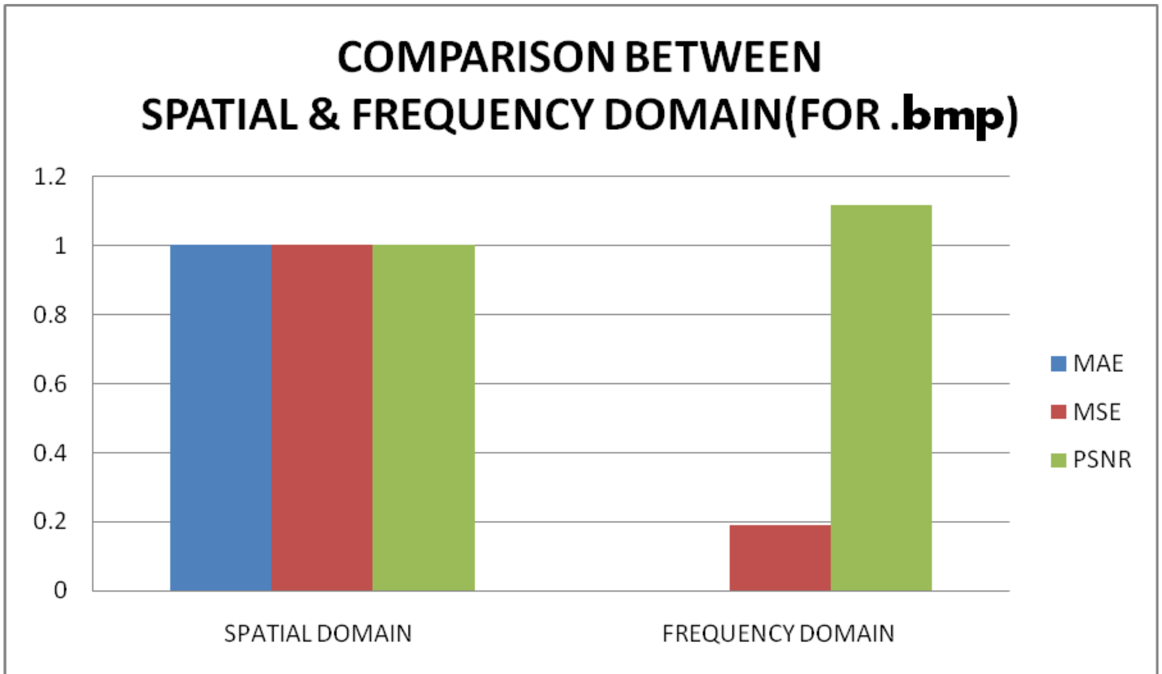

COMPARISON OF INTERPOLATION FOR .bmpTYPE IMAGE
0.12
0.1
0.08
0.06
0.04
0.02
0
NEAREST NEIGHBOUR
BILINEAR
BICUBIC
MAE
MSE
PSNR

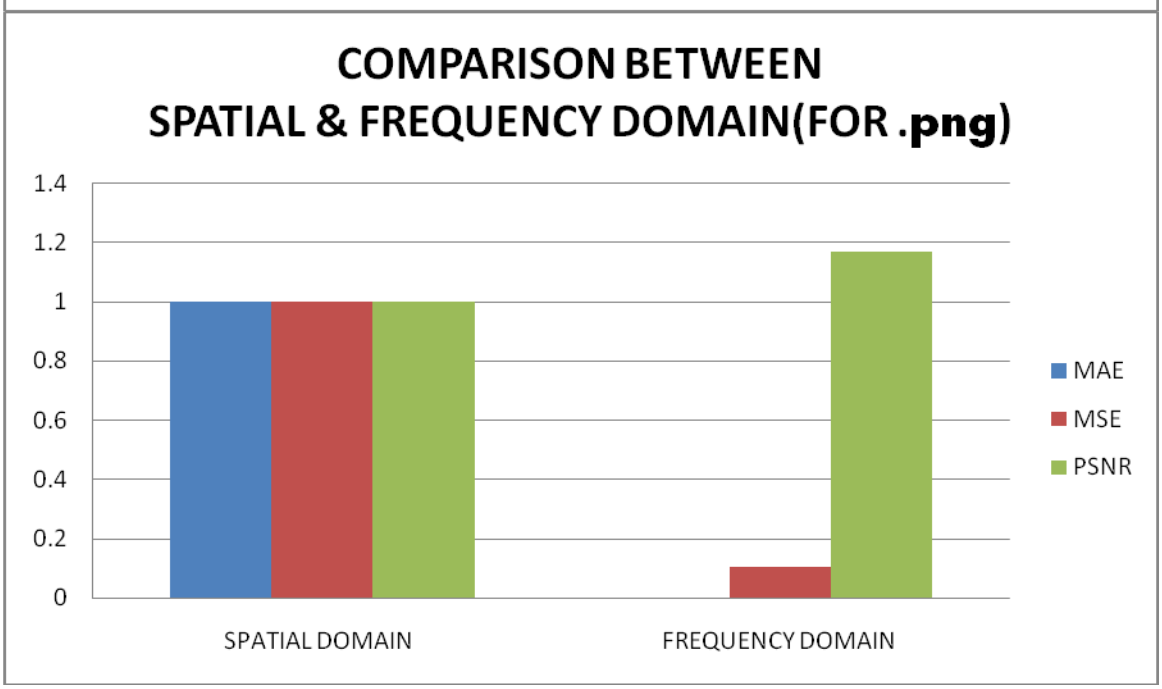

COMPARISON OF INTERPOLATION FOR .jpegTYPE IMAGE
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
NEAREST NEIGHBOUR
BILINEAR
BICUBIC
MAE
MSE
PSNR

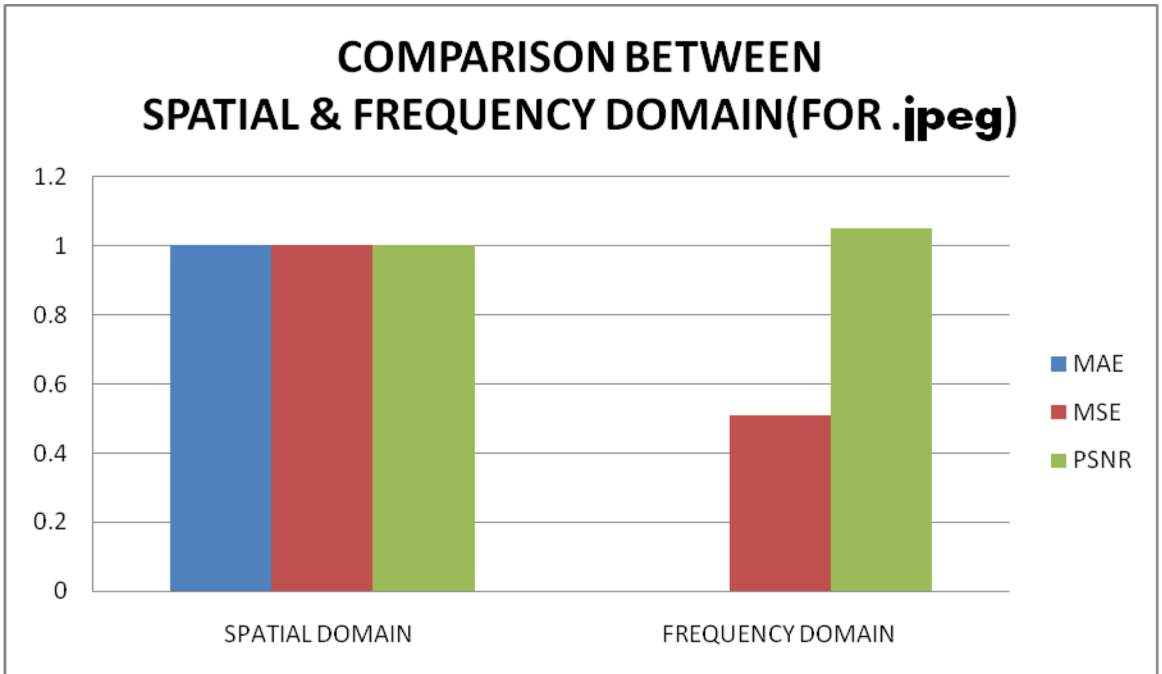

PARAMETER COMPARISON FOR VARIOUS TYPES OF IMAGES USING BILINEAR INTERPOLATION
9.00E-02
8.00E-02
7.00E-02
6.00E-02
5.00E-02
4.00E-02
3.00E-02
2.00E-02
1.00E-02
0.00E+00
BMP
JPEG
PNG
TIF
MAE
MSE
PSNR

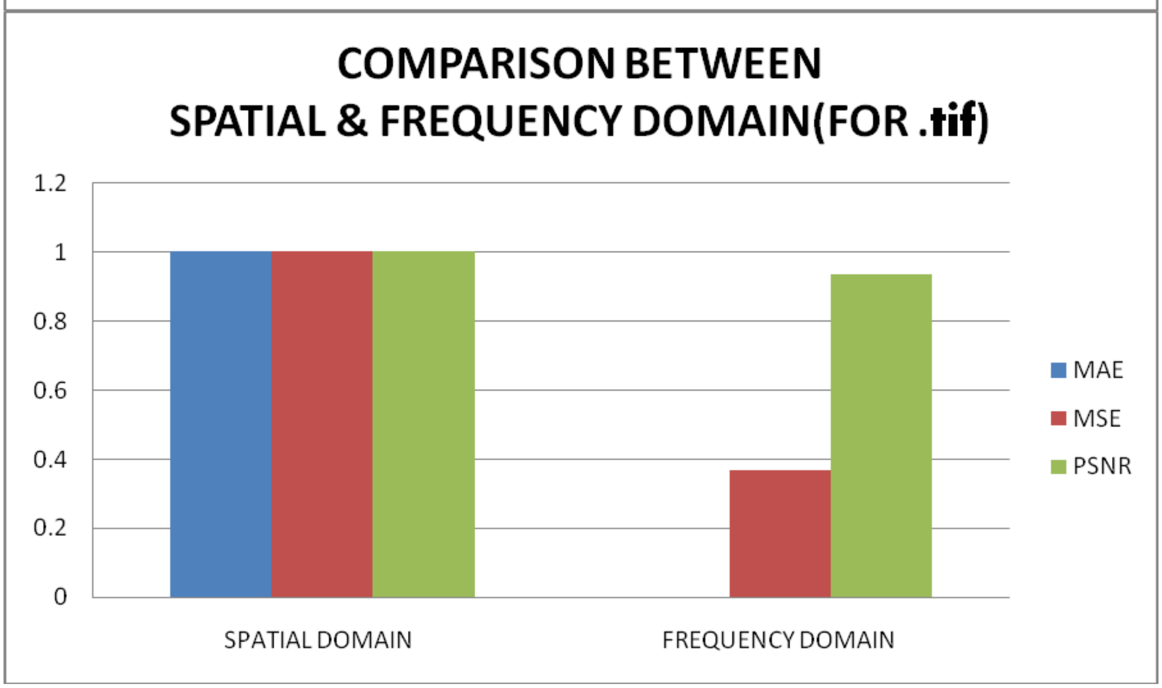

PARAMETER COMPARISON FOR VARIOUS TYPES OF IMAGES USING NEAREST NEIGHBOUR INTERPOLATION
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
BMP
JPEG
PNG
TIF
MAE
MSE
PSNR


PARAMETER COMPARISON FOR VARIOUS TYPES OF IMAGES USING BICUBIC INTERPOLATION
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
BMP
JPEG
PNG
TIF
MAE
MSE
PSNR


COMPARISON BETWEEN
SPATIAL & FREQUENCY DOMAIN(FOR .bmp)
1.2
1
0.8
0.6
0.4
0.2
0
SPATIAL DOMAIN
FREQUENCY DOMAIN
MAE
MSE
PSNR


COMPARISON BETWEEN
SPATIAL & FREQUENCY DOMAIN(FOR .png)
1.4
1.2
1
0.8
0.6
0.4
0.2
0
SPATIAL DOMAIN
FREQUENCY DOMAIN
MAE
MSE
PSNR


COMPARISON BETWEEN
SPATIAL & FREQUENCY DOMAIN(FOR .jpeg)
1.2
1
0.8
0.6
0.4
0.2
0
SPATIAL DOMAIN
FREQUENCY DOMAIN
MAE
MSE
PSNR


COMPARISON BETWEEN
SPATIAL & FREQUENCY DOMAIN(FOR .tif)
1.2
1
0.8
0.6
0.4
0.2
0
SPATIAL DOMAIN
FREQUENCY DOMAIN
MAE
MSE
PSNR

CHAPTER13
Merits and Demerits of the project.

Merits:

1.We have shown in this project that in case of spatial domain interpolation which image type is more suitable.

2.In the latest Image Viewer software like Picassa from Google also uses Bicubic Interpolation Technique while we have developed a little more good resolution enhanced technique.

3.Also we could see that the image quality obtained from the frequency domain filtering is much better than that obtained in the spatial domain.

Demerits:

1.For Spatial domain, with the best results obtained in the Bicubic Interpolation it is at the cost of large computational time as compared to the other.

2.In the frequency domain Interpolation that we used in that the 1x and the 2x images are mandatory to interpolate the Image.

3.We can only zoom in as 4x,8x,16x,32x only.We can’t have it zoom without the multiples of 2.

Chapter 13

Conclusion:

In the end after all this endeavor, We can say that in Spatial Domain Interpolation, bicubic Interpolation is better compared to Bilinear and Nearest neighbor. But bicubic Interpolation takes large time to compute as it deals with 16 components while nearest neighbor deals with 2 and bilinear deals with 4 components resp.

In frequency domain, we could see how the parameters which we considered ie MAE, MSE & PSNR (Low, Low &High) as compared to spatial domain results. But here too the computational costs increases. So, we can just say that selection of appropriate Interpolation technique is totally application based where in we can decide to have how much percentage of Quality vs. Time trade off we can allow.

Chapter 14

Bibliography:

[1] Randy Crane: *A Simplified Approach to Image Processing-classical and modern techniques in C, Prentice Hall PTR, Upper Saddle River, New* Jersey 07458

[2] T. Lehmann, C. Gonner, K. Spitzer: *Survey: Interpolation Methods in Medical Image Processing, IEEE Trans. Medical Imaging, vol. 18, pp.* 1049-1075, 1999

[3] Rafael C. Gonzalez, Richard E. Woods: *Digital Image Processing*, Prentice Hall, 2002

[4] George Wolberg: Digital Image Warping, IEEE Computer Society Press, Los Alamitos, California, 1990