

COMPARISON OF DIFFERENT IMAGE INTERPOLATION ALGORITHMS

By

Divya Doma

**Problem Report submitted to the
College of Engineering and Mineral Resources at
West Virginia University
in partial fulfillment of the requirements
for the Degree of**

**Master of Science
in
Electrical Engineering**

Committee Members

**Dr. Xin Li, Ph.D. (Committee Chair)
Dr. Arun A. Ross, Ph.D.
Dr. Natalia A. Schmid, Ph.D.**

**Lane Department of Computer Science and Electrical Engineering
Morgantown, West Virginia
2008**

Keywords:

**Image Interpolation, Non-Adaptive Methods, Adaptive Methods, NEDI algorithm, DFDF
Algorithm, MSE, SSIM**

ABSTRACT

COMPARISON OF DIFFERENT IMAGE

INTERPOLATION ALGORITHMS

DIVYA DOMA

In this era of internet and multimedia communication, resizing of images to make them suitable for many advanced applications has become very important. In order to resize the images we use image interpolation. Image interpolation is a process of converting a low resolution to high resolution. It has to be done in such a way that the artifacts in the interpolated image are as small as possible. In this report we compare the various Non-Adaptive and Adaptive Interpolation techniques. The Non-Adaptive techniques which we have tested include Nearest Neighbor, Bilinear and Bicubic interpolation and also we have tested two Adaptive Interpolation techniques. The first method uses the local covariance coefficient estimates to obtain the high resolution image and the second method defines two observation sets in two orthogonal directions, for each pixel to be interpolated which produce two directional estimates for each pixel which are fused by using the linear minimum mean square error technique to get the high resolution image pixels. The Mean Square Error and Structural Similarity Index values are calculated for each of these techniques. The performance of these algorithms is compared by using the subjective quality of the interpolated images.

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Xin Li for his valuable guidance and excellent encouragement for the completion of this work. Also I would like to thank my committee members Dr. Arun A. Ross and Dr. Natalia Schmid for taking time from their busy schedules for my work.

I am very grateful to my family and friends whose constant love and support made everything possible.

TABLE OF CONTENTS

1.	INTRODUCTION.....	1
1.1	Image Interpolation Background	1
1.2	Need for Interpolation	2
1.3	Objective	2
1.4	Outline.....	3
2.	CLASSIFICATION OF INTERPOLATION ALGORITHMS.....	4
2.1	Non-Adaptive methods.....	4
2.1.1	Nearest Neighbor Interpolation.....	4
2.1.2	Bilinear Interpolation	5
2.1.3	Bicubic Interpolation.....	5
2.1.4	Higher Order Interpolation	6
2.2	Adaptive Methods.....	7
2.3	Interpolation Artifacts.....	7
2.4	Applications.....	8
3.	ADAPTIVE INTERPOLATION ALGORITHMS	11
3.1	A Survey of Some of the Interpolation Algorithms Developed In the Literature	11
3.2	Edge Preserving Interpolation System for a Digital Camcorder [3]	12
3.2.1	Proposed Algorithm [3].....	13
3.2.2	Hardware Implementation [3]	14
3.3	New Edge-Directed Interpolation [4].....	14
3.3.1	Proposed Algorithm [4].....	16
3.4	An Edge Guided Image Interpolation Algorithm via Directional Filtering and Data Fusion [5]	19
3.5	A High Quality 2 X Image Interpolator [6].....	21
3.5.1	Proposed Algorithm [6].....	21
4.	RESULTS AND DISCUSSIONS.....	24
4.1	Performance Metrics	24
4.1.1	Mean Square Error.....	24
4.1.2	Structural Similarity Index.....	24
4.2	Discussions	34

4.3 Computational Complexity	41
5. CONCLUSIONS AND FUTURE WORK	43
REFERENCES	45

LIST OF FIGURES

Figure 1. Bilinear Interpolation	5
Figure 2. Bicubic Interpolation	6
Figure 3. Interpolation Artifacts	8
Figure 4. Geometric Duality when Interpolating $Y_{2i+1,2j+1}$ from $Y_{2i,2j}$	18
Figure 5. Geometric Duality when interpolating $Y_{i,j}$ ($i+j = \text{odd}$) from $Y_{i,j}$ ($i+j = \text{even}$)	18
Figure 6. Pixels used in Interpolating the centre pixel x5 and the pixels in bold indicate the subset	21
Figure 7. Interpolation results of the image Cameraman. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF	27
Figure 8. Interpolation results of the image House. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF	28
Figure 9. Interpolation results of the image Peppers. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF	29
Figure 10. Interpolation results of the image Lena. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF	30
Figure 11. Interpolation results of the image Cameraman. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF	31
Figure 12. Interpolation results of the image Barbara. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF	32
Figure 13. Interpolation results of the image Fingerprint. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF	33

Figure 14. Formation of the low resolution image (LR) from the High Resolution (HR)	34
Figure 15. Interpolation results of the image Cameraman. (a) Original Image; interpolated image by (b)Bilinear; (c) DFDF; (d) Average of Bilinear and DFDF; (e) NEDI; (f) Average of DFDF and NEDI	39

LIST OF TABLES

Table 1. The MSE values of the recovered HR images.....	25
Table 2. The MSSIM values of the recovered HR images.....	26
Table 3. The MSE values computed by considering the interpolation as an average of two different algorithms.....	40
Table 4. The MSE values computed by considering the interpolation as an average of two different algorithms.....	41

1. INTRODUCTION

In recent years due to the development of many sophisticated digital viewing equipment like High Definition Television, there is a need for the images to be of greater image quality i.e., greater resolution of images is necessary for such applications. So in order to use such applications for the images which are of low resolution, they need to be converted to high resolution so we convert the already existing low resolution images to high resolution by using image interpolation. This technological development promises to produce greater movie quality and wide screen programs to our homes. However its importance goes well beyond the home entertainment and has many applications like interactive video, desktop publishing, medical imaging and computer graphics etc., in addition to the home entertainment.

1.1 Image Interpolation Background

Image interpolation is a process of resizing or scaling a digital image but to a large size i.e., generating a high resolution image from the available low resolution image. Interpolation is basically a method of increasing the number of pixels in a digital image. As the size of the image increases the pixels which comprise the image become more and more visible making the image look smoother so it might lose the edge details and some methods might result in jagged or blurry details in the image. Conversely when the image is resized to a lower size i.e., reducing an image size (down sampling) will tend to enhance the image smoothness and sharpness. Interpolation is basically approximating the values at unknown points using the known data. All we have is the low resolution image pixels and we need to generate the missing pixels in the high resolution counterpart using the image interpolation techniques. It is only an approximation and so the image quality degrades each time interpolation is performed.

Interpolation is basically converting an image from one resolution to other without losing the visual content in the image and also without inducing artifacts in the interpolated image. Most of the image interpolation techniques have been developed by interpolating the pixels based on the local characteristics like edge information, neighboring pixels.

1.2 Need for Interpolation

In order to use the already available images for the applications which require high resolution images, they have to be interpolated to get the high resolution images. Applications like HDTV require high resolution frames in a video, so the old videos which are of low resolution have to be converted to high resolution frames before they can be used in such applications. For converting the standard definition television frames to high resolution for playback on high definition television receivers and monitors image interpolation is needed. When an image is zoomed it is not possible to discover more information than it is already available so obviously the image quality suffers. So in order to know the missing pixel values we go for interpolation techniques. Also sometimes the images captured from a low resolution camera have to be enlarged for high quality prints in magazines, wall papers, catalogs and even for use in homes for displaying in the form of posters etc., Not only in these applications, image interpolation is also useful in computer graphics, editing, online image viewing, biomedical field, remote sensing and other fields.

1.3 Objective

The main objective of this work is comparing the various interpolation techniques that are existing in the literature to know the appropriateness of each of these methods to be applied for different applications.

1.4 Outline

Chapter 2 presents an overview of various adaptive and non-adaptive interpolation schemes that are available followed by a discussion on various kinds of artifacts caused due to the interpolation process. In chapter 3 we discuss various adaptive algorithms proposed for improving the output interpolated image. Chapter 4 gives the interpolation results for some non adaptive algorithms and adaptive algorithms like New Edge Directed Interpolation and Edge Guided Image Interpolation via Directional Filtering and Data Fusion and compares the results of these algorithms. Then we present the discussions on the visual quality of the interpolation results. Also we try to compare the different interpolation techniques using some objective metrics like mean square error and structural similarity index and we observe that they are not the ideal metrics for measuring the performance of the algorithm. In chapter 5 we discuss the concluding remarks and future work are presented.

2. CLASSIFICATION OF INTERPOLATION ALGORITHMS

The interpolation algorithms can be grouped into 2 categories.

2.1 Non-Adaptive methods

These methods treat all pixels equally i.e., the logic used to find the unknown pixels in the high resolution remains constant i.e., irrespective of the image features it is fixed [2]. In Non-Adaptive methods certain computations are performed indiscriminately to the whole image without considering the image contents. Non-Adaptive algorithms include nearest Neighbor, Bilinear, Bicubic, Spline, Cubic Polynomial, Sinc, Lanczos and others. Depending on the complexity of the algorithm they use somewhere between 0 to 256 adjacent pixels for interpolating [2]. The computational complexity increases with the increase in the number of pixels considered while interpolation and so does the processing time. But the accuracy also increases with the increase in the number of adjacent pixels considered while interpolating. And so there is a tradeoff between accuracy and computational complexity [2]. The smoothness of the enlarged image depends on how sophisticated the algorithm is.

2.1.1 Nearest Neighbor Interpolation

It is the simplest interpolation and requires least computation and takes least processing time. The nearest neighbor algorithm simply selects the pixel value of the nearest pixel and does not consider the values of other neighboring pixels at all i.e., the value of the missing pixel in the new image is the value of the nearest pixel in the original image. It simply makes each pixel bigger. This type of interpolation can only be used for closer examination of digital images because it does not change the pixel information of the image and does not introduce any anti-aliasing and so it is not the suitable method for enlarging images because it

introduces unwanted artifacts near the edges, where you find greater intensity changes and does not give smoother results. This method of interpolation results in blocky images so it is not useful in high quality imaging applications [10].

2.1.2 Bilinear Interpolation

It considers the closest 2 by 2 neighborhood of the known pixels surrounding the unknown pixel and then takes weighted average of these 4 pixels to get the missing pixel value.

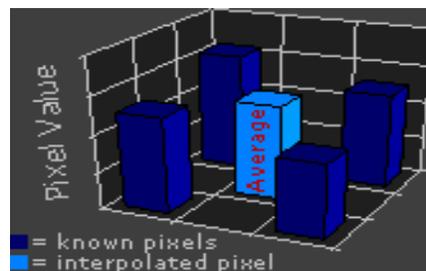


Figure 1. Bilinear Interpolation [2]

The averaging has an anti-aliasing effect and so results in relatively smooth edges. Even though it gives smoother results compared to the nearest neighbor algorithm, the results for sharp edge transitions are not ideal, but better comparatively. It gives blurring artifacts. Here all the 4 neighboring pixels are at equal distance so the sum is directly divided by 4 i.e., all the neighboring pixels are given equal weight.

2.1.3 Bicubic Interpolation

Bicubic is much more complicated and costly. Bicubic considers the closest 4 by 4 neighborhood of the known pixels surrounding the unknown pixel and takes the weighted average of these 16 pixels to get the interpolated value. Since all the 16 neighboring pixels are not at equal distance from the unknown pixel, the closer pixels are given more weight in the calculation. This method produces noticeable sharper images with smooth edges compared to

nearest neighbor and bilinear interpolation and is the ideal combination of output image quality and processing time. Although it gives better performance compared to bilinear interpolation the bicubic interpolation gives jaggging effect around the sharp edges, especially at the boundaries like dark colored texts against a light colored background.

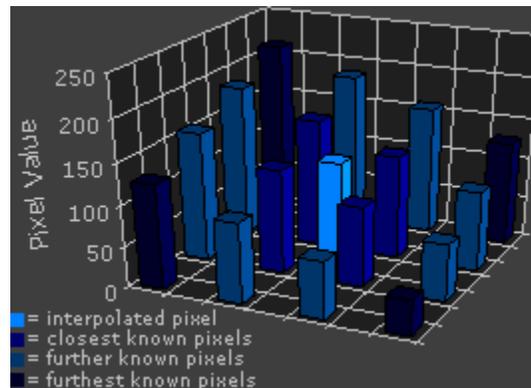


Figure 2. Bicubic Interpolation [2]

Most of the softwares which have the feature of image enlargement use this bicubic image interpolation for enlarging the images.

2.1.4 Higher Order Interpolation

There are many higher order interpolation techniques like spline and sinc interpolation which take into consideration more number of surrounding pixels. Since these algorithms consider more number of surrounding pixels it becomes computationally more complex and accurate. These algorithms are mainly used when the image requires multiple enlargements in different steps [2]. For applications which require single step enlargements, the higher order interpolation techniques do not result in much visible improvements. Therefore they are not used in such applications since the processing time is greatly increased and there is not much improvement in the visual quality. The spline based interpolation techniques are mainly

used in biomedical imaging. Most of the applications involving medical imaging use this spline interpolation [11]. The bilinear interpolation can be realized as equivalent to spline based interpolation but of degree 1. Spline based interpolation is more advantageous when we go for higher degrees. They provide a good tradeoff between the performance and the cost and so they are used in applications like medical imaging where quality of the medical images is the major concern [11]. The cubic spline interpolation method can be used at the expense of increased computational complexity, since it gives good image quality compared to the nearest neighbor or Bilinear interpolation [12].

2.2 Adaptive Methods

These methods change depending on what they are interpolating like if it is an edge pixel it is treated differently from a textured pixel region. The logic used for computing the unknown high resolution pixels is dependent upon the image features [2]. Adaptive algorithms include algorithms in licensed software such as Qimage, PhotoZoom Pro, Genuine Fractals, etc., Many of these apply a different algorithm depending on the type of the data, to minimize the interpolation artifacts in regions where they are most visible like at the edges. These algorithms are primarily designed to maximize artifact-free detail in enlarged photos, so not all can be used to distort or rotate an image [2].

2.3 Interpolation Artifacts

There are several artifacts that might arise while performing interpolation. The artifacts like ringing, aliasing, blocking and blurring are the most common ones. All the interpolation algorithms try to minimize the interpolation induced artifacts. Edge halos, Blurring and Aliasing are the three main artifacts induced during the interpolation process. Even with the

most sophisticated algorithm it is not possible to minimize all the three artifacts at the same time so one of the artifacts have to be increased or decreased at the expense of the other two [2]. With adaptive techniques the effect of these artifacts might be less since they apply different algorithms depending on the kind of the pixel i.e., depending on whether it is an edge, texture or smooth area but we cannot remove them completely, however they can induce non image textures. These degradations become worse when the magnification ratio increases. There exists a tradeoff between reducing the blocking artifact and excessive smoothness. Edge Halos occur only in the case of color images.

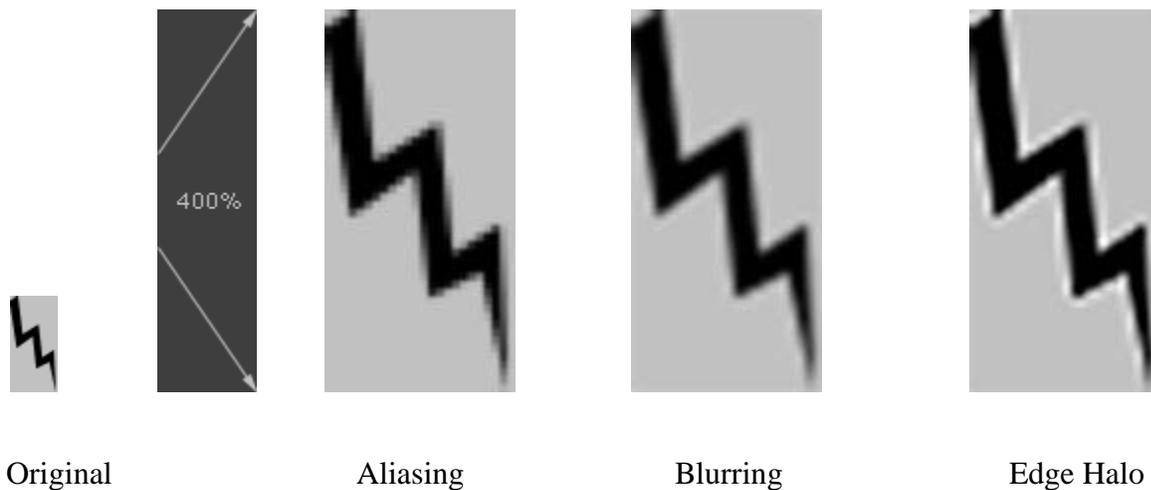


Figure 3. Interpolation Artifacts [2]

2.4 Applications

Image processing technologies have been playing a major role in many applications these days. They try to process the visual information so that it can be applied in many applications like multimedia and transmission of images through the internet and also to improve the visual effect of the images. Because of its applications image processing techniques are being widely used these days in many day to day applications. Image Interpolation is an

important image processing operation applied in a wide range of applications from computer graphics, rendering, editing, medical image reconstruction, satellite imaging and online image viewing [10].

For using the printing capabilities of a printer efficiently. For example when we consider an image taken from a 5MP camera and printing with a printer of 600dpi. This printer can print much larger sized images so the resolution of the image has to be increased in order to get a larger print so we use interpolation techniques for doing that.

Another important area where image interpolation is widely used is, in the medical image visualization which involves 2D operations like image zooming, panning, rotation and 3D operations like reslicing which are mostly used by the radiologists. The interpolation models are also needed for performing various types of image registrations which include intra-model and inter-model registrations for different kinds of data [35]. In many biomedical applications the main goal is to modify the sampling rate of pixels. This is called rescaling [8, 13]. It is mainly needed when the acquisition device for example a scanner has a non-homogeneous resolution, i.e., a fine within slice resolution and a coarse across-slice resolution. So reslicing has to be performed in order to perform operations on it [8, 13].

Image Interpolation is also applied in streaming videos in websites like youtube. These websites most of the times store videos at low resolutions and the videos have to be expanded as users wish to view the videos at full screen, so it requires that the images have to be interpolated to a higher resolution [14]. One more application where the same concept is used is in the modern displays like HDTV displays. In order to utilize the display capabilities of such devices, the images that are coming from the conventional low resolution source have to be first

converted to high resolutions through the process of interpolation. The images that can be viewed on a Standard Definition Television (SDTV) have to be converted to higher resolution i.e., they have to be magnified for them to be viewed on a High Definition Television (HDTV). SDTV images are much lower in quality in terms of resolution when compared to the HDTV images. For example, a 720 X 480 image looks very good when viewed on a standard definition television than when it is viewed on a high definition television. In converting the low resolution videos to high resolution videos, the temporal correlation property in video signals allows the interpolation being done using the multi frame super resolution techniques [14]. But in this report we study only the interpolation algorithms that can be applied on still frame images.

3. ADAPTIVE INTERPOLATION ALGORITHMS

3.1 A Survey of Some of the Interpolation Algorithms Developed In the Literature

There are many algorithms that are designed for the purpose of image interpolation. The main aim of these algorithms is to faithfully reconstruct the high resolution image from the low resolution counterpart. The human visual system is very sensitive to the edge structures so the main aim of these algorithms is to preserve the edge structure in the high resolution image [5]. The conventional linear image interpolation algorithms do not preserve the edge structure properly and they suffer from blocking effects, blurred details and ringing artifacts around the edges so there is a demand for high performance i.e., high quality image interpolation algorithms and hence some non linear image interpolation algorithms have been proposed which retain the edge sharpness in the interpolated images.

In some interpolation algorithms an adaptive interpolation is performed based on the intensity variations that are not distinguishable by the human eye and based on the local structure of the image [16]. The interpolation scheme proposed by Jensen and Anastassiou detects edges and fits them by some fitting operator developed by the algorithm in order to improve the visual quality of the images [15]. In edge preserving image interpolation system, a directional interpolation is used for preserving the edge structure resulting in more naturally looking artifact free images [3]. This method is used mostly in applications like in a camcorder zoom and is explained in more detailed in the coming section. Li and Orchard proposed a multi resolution covariance based adaptation technique in which we estimate the high resolution covariance coefficients from the covariance of the low resolution image which represents the

edge direction information [4] is described in the coming section. The interpolation algorithm by Carrato and Tenze first uses pixel replication and then it corrects them by using the 3X 3 edge patterns and optimizing the operator parameters [6]. This method is also described in detail in the coming section. In the edge guided image interpolation via directional filtering and data fusion the edge structures are preserved by dividing each pixel into two orthogonal observation sets. And these two pixel measurements are considered to be the noisy measurements of the missing pixel which are then fused to get a more robust estimate by using the linear minimum mean square estimation technique [5].

3.2 Edge Preserving Interpolation System for a Digital Camcorder [3]

Many image interpolation techniques introduce various artifacts like the blocking effect in the interpolated high resolution image. Edge preserving interpolation system provides better magnified images by preserving the edge details and not degrading the smooth areas. The nearest neighbor and bilinear interpolations result in blocky artifacts and some other sophisticated algorithms create blurred image while they reduce the blocking artifacts, as they apply the same algorithm for all kinds of details in the image i.e., they do not consider the local characteristics of the image.

The edge preserving interpolation system identifies the different kind of edges like the horizontal, vertical and diagonal edges details and then applies interpolation methods depending on whether it is edge or smooth area. A directional interpolation is performed to preserve the edge detail so that the sharpness of the interpolated image is maintained. The discrete cosine transform coefficients are used to extract the edge information. Five different types of edges are identified using these coefficients and the zero order and bilinear interpolation are performed together and then five different low pass Gaussian filters is applied adaptively

depending on the type of the edge in order to reduce the discontinuities resulting from the interpolation process.

3.2.1 Proposed Algorithm [3]

This algorithm can be mainly divided into 6 blocks. In block 1, the image that is to be interpolated is taken and a block is truncated from that image and all the operations in the successive blocks are performed on this image block and it is enlarged. This process is repeated for all the blocks in the image until all the blocks in the image are enlarged. In the next step the edge information for every 2 by 2 block is determined. In block 2 DCT coefficients for every 4 by 4 block around the corresponding 2 by 2 block are computed and then in block 3 the type of edge is determined depending on the DCT coefficients. The DCT coefficients represent the intensity variations in different directions. Depending on the DCT coefficients 5 types of edges i.e., horizontal edge, vertical edge, no edge and diagonal edge which has two variations in it depending on the direction of the diagonal edge. In block 4 a Cubic B-spline transform is applied in order to nullify the low pass filtering effect due to the interpolation procedure performed next to this. It is a kind of high pass filter which increases the sharpness of the image before it is interpolated. In block 5 interpolation is performed depending on the information obtained from blocks 2 and 3. For the case of no edge bilinear interpolation is performed. In case if a block has horizontal or vertical edge a one dimensional directional interpolation is performed according to the direction of the edge. In case of a diagonal edge the four corners are filled with each pixels intensity value and the diamond shape at the centre is filled with the one dimensionally bilinear interpolated data by using the two pixel intensity on the diagonal line. Finally in block 6 ellipsoidal low pass Gaussian filters is applied to eliminate blocking artifacts and discontinuities.

The geometries of these Gaussian filters are determined depending on the edge information obtained from blocks 2 and 3.

3.2.2 Hardware Implementation [3]

Since it requires great amount of computational complexity for calculating the DCT coefficients and we only need 2 out of the sixteen coefficients calculated. So in order to reduce the computational complexity the coefficients can be interpreted as the result of two convolution operations by using 4 by 4 DCT basis functions. The convolution operation can be realized by multiplying the output of the FFT block with the DCT basis functions and then performing inverse FFT. This way we can avoid using the DCT module thereby decreasing the computational complexity of the system but at the cost of increased hardware requirement. It accelerates the calculation of DCT coefficients. And in the next step edge identification is implemented following which the Direct B-spline transform is performed. The next step is the interpolation step in which the zero order and first order interpolation is combined and performed. The Gaussian low pass filters cannot be implemented because of its varied geometry and so the system uses look up tables for the filter coefficients.

This scheme not only maintains the geometry and sharpness of the original image but can also give good results even when the images are magnified to as large as 5 times or more. The proposed interpolation system is expected to be applied to image zooming system for digital video systems like camcorder.

3.3 New Edge-Directed Interpolation [4]

The basic non-adaptive interpolation techniques like bilinear and bicubic interpolation methods do not give better results at the areas where you find edges and the

textured area and result in blurred edges and annoying artifacts. They are preferred not because they give good performance but because they are less complex [4]. Hence we go for Adaptive interpolation techniques which determine the interpolation coefficients depending on whether it is edged or a smooth area. The Edge Directed Interpolation method improves the visual quality of the interpolated image when compared to the non-adaptive interpolation techniques. In any image interpolation technique the objective is to estimate the high resolution image from the available low resolution image. The method employed in doing it differs from one technique to the other. And the quality of the interpolated images and the performance of the interpolation algorithm depend on the model that is employed to describe the relationship between the low resolution pixels and the high resolution pixels. This method statistically models interpolation to any arbitrary orientation unlike the edge preserving interpolation algorithm used in a digital camcorder which is restricted to specific edge orientations.

The main idea of the Edge Directed Interpolation method is based on the geometric regularity of the edges, i.e., the image intensity field evolves more slowly along the edge orientation than across the edge orientation [17]. It affects the visual quality of the image like sharpness of the edges and artifact free images [4]. So we need to implement an orientation adaptive interpolation method that takes into account the geometric regularity of the edges. There are some interpolation techniques [3, 15, 18, 19] in literature that estimate the edge orientation and vary the interpolation coefficients depending on the edge orientation. But these methods categorize the edges into a limited number of choices like horizontal, vertical or diagonal affecting the accuracy of the model. This method is basically a covariance based adaptation method i.e., the relation between the high resolution pixels and low resolution pixels is developed based on the covariance coefficients. The idea is to find the high resolution covariance

from the low- resolution image that is to be interpolated. To do this we need to find the geometric duality between the low resolution covariance and high resolution covariance which couple the pair of pixels along the same orientation [4].

By using these high resolution covariances the optimal minimum mean square error interpolation can be derived by modeling the image as a locally stationary Gaussian process [4]. This method improves the quality of the interpolated images over the bilinear interpolation but at the cost of high computational complexity but this proposed algorithm improves the visual quality of the image at the edge pixels. And the bilinear interpolation works fine for the non edge pixels so hybrid approach is proposed in which the edge pixels are identified and the covariance based method is applied to those pixels and bilinear interpolation is applied to non-edge pixels. This way we can get an output with good visual quality and with reduced computational complexity.

This method of new edge directed interpolation improves the visual quality of the interpolated image but the objective quality measures like the Mean Square Error (MSE) do not always provide an accurate measure of the visual quality.

3.3.1 Proposed Algorithm [4]

The low resolution image $X_{i,j}$ of size $H \times W$ is directly derived by down sampling the high resolution $Y_{2i,2j}$ of size $2H \times 2W$ so that the output quality can be easily compared with the original image. We need to interpolate the pixels $Y_{2i+1,2j+1}$ from the available pixels $X_{i,j}$. In order to do this, we consider the four nearest neighbors along the diagonal directions [4] and the interpolation coefficients are computed by using the weiner filtering theory [20].

$$\hat{Y}_{2i+1,2j+1} = \sum_{k=0}^1 \sum_{l=0}^1 \alpha_{2k+l} Y_{2(i+k),2(j+l)} \quad \dots (1)$$

$$\vec{\alpha} = R^{-1} \vec{r} \quad \dots (2)$$

where $R = [R_{kl}]$, ($0 \leq k, l \leq 3$) and $\vec{r} = [r_k]$, ($0 \leq k \leq 3$) are the high resolution covariances, r_0 is defined by $E[Y_{2i,2j} Y_{2i+1,2j+1}]$. But the pixel $Y_{2i+1,2j+1}$ is the missing pixel in the high resolution image. So we need to find the high resolution covariances using the orientation which can be found using the local covariance structure. Hence we find the high resolution covariances from the low resolution using the geometric duality property. The orientation property of the edges effects the visual quality of the edges. Geometric duality is the correspondence between the high resolution covariance and the low resolution covariance that couple the pair of pixels at different resolution but along the same orientation [4]. Using this geometric duality we need to find the pixels $Y_{2i+1,2j+1}$ from the pixels $Y_{2i,2j}$ and also the pixels $Y_{i,j}$ ($i+j=\text{odd}$) from the pixels $Y_{i,j}$ ($i+j=\text{even}$). The geometric duality for both of these cases can be observed in figs 3.1 and 3.2.

The low resolution covariance \hat{R}_{kl}, \hat{r}_k can be estimated from the local window of the low resolution using the classical covariance method [20]

$$\hat{R} = \frac{1}{M^2} C^T C, \hat{r} = \frac{1}{M^2} C^T \vec{y} \quad \dots (3)$$

Where $\vec{y} = [y_1 \dots y_k \dots y_{M^2}]^T$ is a vector whose elements pixels in the local window of size $M \times M$ and C is a $4 \times M^2$ matrix whose k th column is the four nearest neighbors of y_k along the diagonal direction. Combining [4] the above two equations we get

$$\vec{\alpha} = (C^T C)^{-1} (C^T \vec{y}) \quad \dots (4)$$

By substituting the value above value in the first equation we can find the interpolated value.

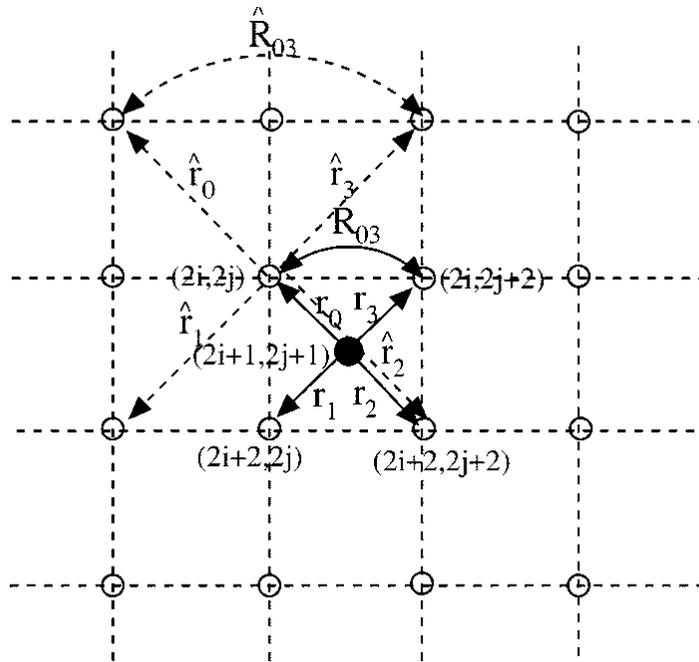


Figure 4. Geometric Duality when Interpolating $Y_{2i+1,2j+1}$ from $Y_{2i,2j}$ [4]

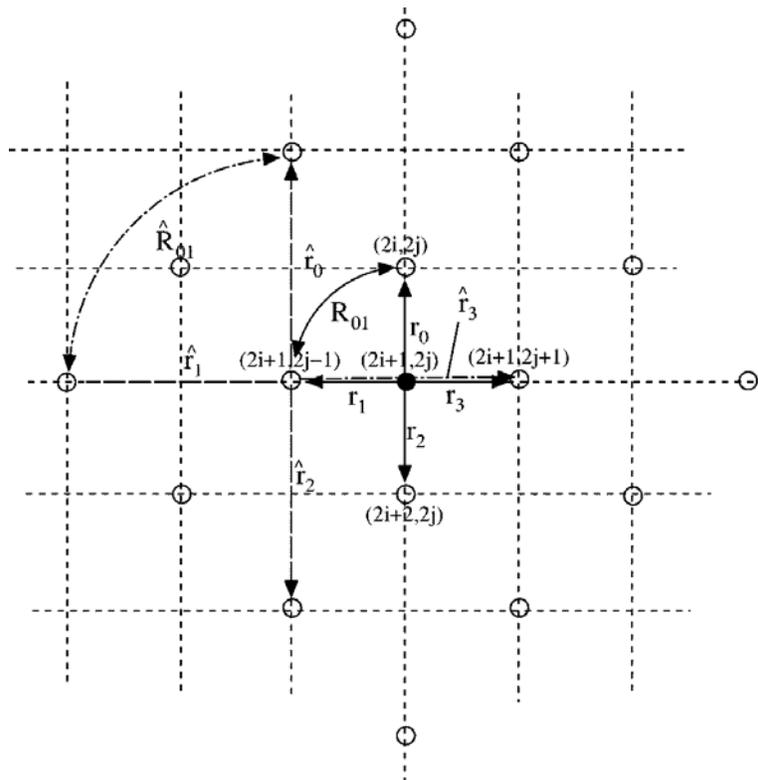


Figure 5. Geometric Duality when interpolating $Y_{i,j}$ ($i+j = \text{odd}$) from $Y_{i,j}$ ($i+j = \text{even}$) [4]

The main disadvantage with this technique is its computational complexity. In order to reduce the computational complexity we devise a hybrid method in which the covariance based method is only applied to the edge pixels since they are ones which get benefited greatly by using this method and for the rest of the pixels we use the simple non adaptive methods. The computational complexity reduces greatly by using this hybrid approach since the number of edge pixels will be less in many images. The edge pixels are identified by setting a threshold. A pixel is identified as an edge pixel if the local variance calculated from the 4 nearest neighbors is above a particular value.

3.4 An Edge Guided Image Interpolation Algorithm via Directional Filtering and Data Fusion [5]

There are some interpolation techniques in literature in which the edge direction is determined and depending on the edge direction the interpolation is performed. But estimating the edge direction with the only knowledge of the low resolution image is very difficult and might not always give correct results i.e., the resultant edge direction might be incorrect which leads to wrong results. In this method of the Edge Guided interpolation using the directional filtering and data fusion the missing pixel in the high resolution is estimated not in one direction but in two mutually orthogonal directions. These two results are the noisy estimates of the missing pixel. The pixel is finally interpolated using the principle of linear minimum mean square error method to get a better output which is visually better. This approach performs in a better way compared to some non adaptive interpolation techniques but the main drawback of this method is the huge computational complexity. As an extension to this algorithm a simplified approach is developed which reduces the computational complexity to a great extent without much degradation in the resultant output. The main aim is to generate a high resolution image

from the knowledge of the low resolution pixels without much interpolation artifacts i.e., without blurring or jaggy edges.

In this method first the edge direction is determined for the interpolation of the missing pixels. In order to extract this edge direction, the neighboring pixels of the missing pixel that is to be interpolated are divided into two directional subsets that are orthogonal to one another. The interpolation of the missing pixels is carried out in two steps, in the first step we interpolate the missing centre pixel which is surrounded by 4 known pixels in diagonal directions and in the next step the missing pixels are interpolated by using the knowledge of the pixels determined in the above step.

We interpolate the missing pixel along the two orthogonal directions and these are the noisy measurements of the missing high resolution pixel in two orthogonal directions and now the missing pixel has to be estimated from these measurements. We take into consideration the minimum mean square method to find the missing pixel using the above measurements. In order to do that we take some probability considerations into account and estimate the missing pixel using the covariance and mean values of the noisy measurements. This method requires great amount of computational complexity in the calculation if the mean, covariance and standard deviation values are computed for each and every pixel. So we use this method only for the edge pixels and for rest of the pixels we employ some basic interpolation methods. Since the number of edge pixels will be less using this method we reduce the computational complexity to a great extent.

3.5 A High Quality 2 X Image Interpolator [6]

This method is based on a non linear operator operating on a 3 X 3 pixel mask. All we do is normal pixel duplication and then a non linear correction of the samples which avoids the blocking artifacts in the interpolated images. In this method basically some predetermined edge patterns are used in order to determine the optimized parameters in the interpolation operator [31].

3.5.1 Proposed Algorithm [6]

We divide the pixel that is to be interpolated, into four parts as shown in the figure below and then the value of each of the pixel is corrected by using the pixel information in the subset mask S_k containing 6 pixels highlighted in the figure in order to correctly perform in the presence of sharp edges and smooth areas.

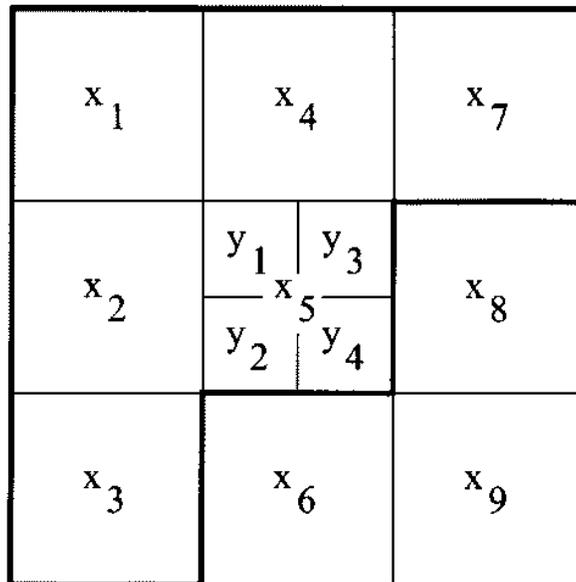


Figure 6. Pixels used in Interpolating the centre pixel x_5 and the pixels in bold indicate the subset [6]

The number of pixels in the subset mask are chosen such that it includes sufficient information in order for correct prediction of missing pixels and also avoiding complexity by suitably selecting the number of pixels because as the number of pixels in the subset mask increases, the complexity also increases. For y_1 the pixels that are considered in the subset mask are $x_1, x_2, x_3, x_4, x_5,$ and x_7 present in the upper left side of the mask. When the other pixels are being interpolated we rotate the subset mask and perform the operations described in the following steps.

The value of y_1 may be corrected by using some pixel differences [6] which are to be multiplied by their corresponding weights. The weights have to be suitably selected so that response is estimated correctly [6]

$$d1 = x_1 - x_5$$

$$d2 = (x_2 + x_4)/2 - x_5$$

$$d3 = (x_3 + x_7)/2 - x_5$$

so that

$$y_1 = x_5 + \sum_{k=1}^3 w_k d_k$$

The edges have to be correctly interpolated since sometimes non linear interpolation might result in steep edges due to abrupt large transitions in the signal [32] resulting in under or over shoots near the edges in the interpolated image. So in order to avoid this and to get a response that is proportional to the edge height, we consider the weights such that they are independent of the edge height.

The interpolator we select taking into account the complexity issues takes the form [6]

$$y_1 = x_5 + \sum_{k=1}^3 w_k d_k = x_5 + (q_1 d_1 + q_2 d_2 + q_3 d_3)/(q_4 + \epsilon)$$

We consider the constant ϵ in order to avoid a zero denominator and the weights q_j where $j = 1$ to 4 is calculated such that it takes into account all the possible differences of couples of pixels and of averages of couples of pixel differences considering the symmetry [6].

$$\begin{aligned} q_j = & p_{1j} (x_1 - x_5)^2 + p_{2j} (x_2 - x_4)^2 + p_{3j} (x_3 - x_7)^2 + p_{4j} ((x_1 - x_2)^2 + (x_1 - x_4)^2) \\ & + p_{5j} ((x_1 - x_3)^2 + (x_1 - x_7)^2) + p_{6j} ((x_2 - x_5)^2 + (x_4 - x_5)^2) \\ & + p_{7j} ((x_3 - x_5)^2 + (x_7 - x_5)^2) + p_{8j} ((x_3 - x_2)^2 + (x_7 - x_4)^2) \\ & + p_{9j} ((x_3 - x_4)^2 + (x_7 - x_2)^2) \end{aligned}$$

For $j = 1$ to 4 , $i = 1$ to 9 , p_{ij} are the parameters that are to be suitably selected. The values of these parameters are selected such that we get an artifact free image. This can be done by optimizing the values of the parameters by using a set of edges which contain both smooth and sharp transitions. By considering the different kinds of transitions and giving weights to them a suitable set of parameters are found out by taking the minimum mean square error also into account [6]

$$[p_{ij}] = \begin{bmatrix} 14.1 & 112.4 & 0.2 & -19.5 & 5.0 & 19.2 & -29.5 & -7.7 \\ 64.4 & -81.0 & 158.2 & 24.0 & 38.3 & 70.5 & 65.6 & 22.5 \\ 61.5 & 94.9 & -6.5 & 31.2 & 35.8 & 55.9 & -52.3 & -52.5 \\ 97.5 & 96.3 & -3.9 & 115.2 & 59.6 & 117.9 & 132.4 & 96.4 \end{bmatrix}^T$$

This technique gives good results for sharp synthetic and natural images with less blurring and undesirable artifacts [6].

4. RESULTS AND DISCUSSIONS

We tested the various adaptive and non adaptive algorithms on a set of images and calculated the two performance metrics which are described as follows.

4.1 Performance Metrics

4.1.1 Mean Square Error

Mean square error between the down sampled, interpolated image and the original image is the amount by which they differ. It can be defined as the sum of pixel wise quadratic differences of intensities in the two images. It can be computed as

$$MSE = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |I_{original}[m,n] - I_{interpolated}[m,n]|^2$$

Where *I_{original}* is the original image and *I_{interpolated}* is the interpolated version of the downsampled original image, M and N are the number of pixels in horizontal and vertical directions respectively.

4.1.2 Structural Similarity Index

SSIM Index is a metric that measures the similarity between two images. The original image is kept as a reference in order to calculate the SSIM value [8]. SSIM is a metric that is designed to calculate the minutest difference between the two images. In this report to calculate the SSIM we use the Zoug Wang's code [9] available to us. The SSIM index value lies between 0 and 1. A value of zero would mean that there is no correlation between the interpolated and original images and a value close to 1 means that the particular interpolation algorithm gives good results

and if it is equal to 1 indicates that the interpolated image and the original image are exactly the same.

The values of MSE and MSSIM for different algorithms tested on various images are given in tables 4.1 and 4.2 respectively. Where the low resolution image is obtained by directly downsampling the high resolution image along both the directions.

Table 1. The MSE values of the recovered HR images

Image	Nearest	Bilinear	Bicubic	NEDI	DFDF
Cameraman	412.627	187.968	188.8393	186.7472	176.3371
House	117.9029	41.1877	39.3918	49.4745	35.9041
Lena	97.6472	29.8418	26.3113	28.3279	26.4676
Barbara	390.2970	209.8498	238.8805	376.3261	225.036
Peppers	196.9503	66.8146	63.6234	85.3567	57.4024
Boat	187.4195	79.322	77.4626	80.6762	76.1021
Fingerprint	376.5494	72.3656	46.7549	88.9009	59.6874

Table 2. The MSSIM values of the recovered HR images

Image	Nearest	Bilinear	Bicubic	NEDI	DFDF
Cameraman	0.7849	0.8638	0.8634	0.8632	0.8675
House	0.8087	0.8837	0.8775	0.8702	0.878
Lena	0.8324	0.9129	0.9145	0.9126	0.9127
Barbara	0.7104	0.8032	0.7952	0.7654	0.7954
Peppers	0.8277	0.9112	0.9116	0.9113	0.9165
Boat	0.7434	0.8408	0.8413	0.8387	0.8412
Fingerprint	0.8176	0.9462	0.9615	0.9331	0.95

These interpolation algorithms were tested on various images and the output images are shown in figures 4.1 through 4.7



Figure 7. Interpolation results of the image Cameraman. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF

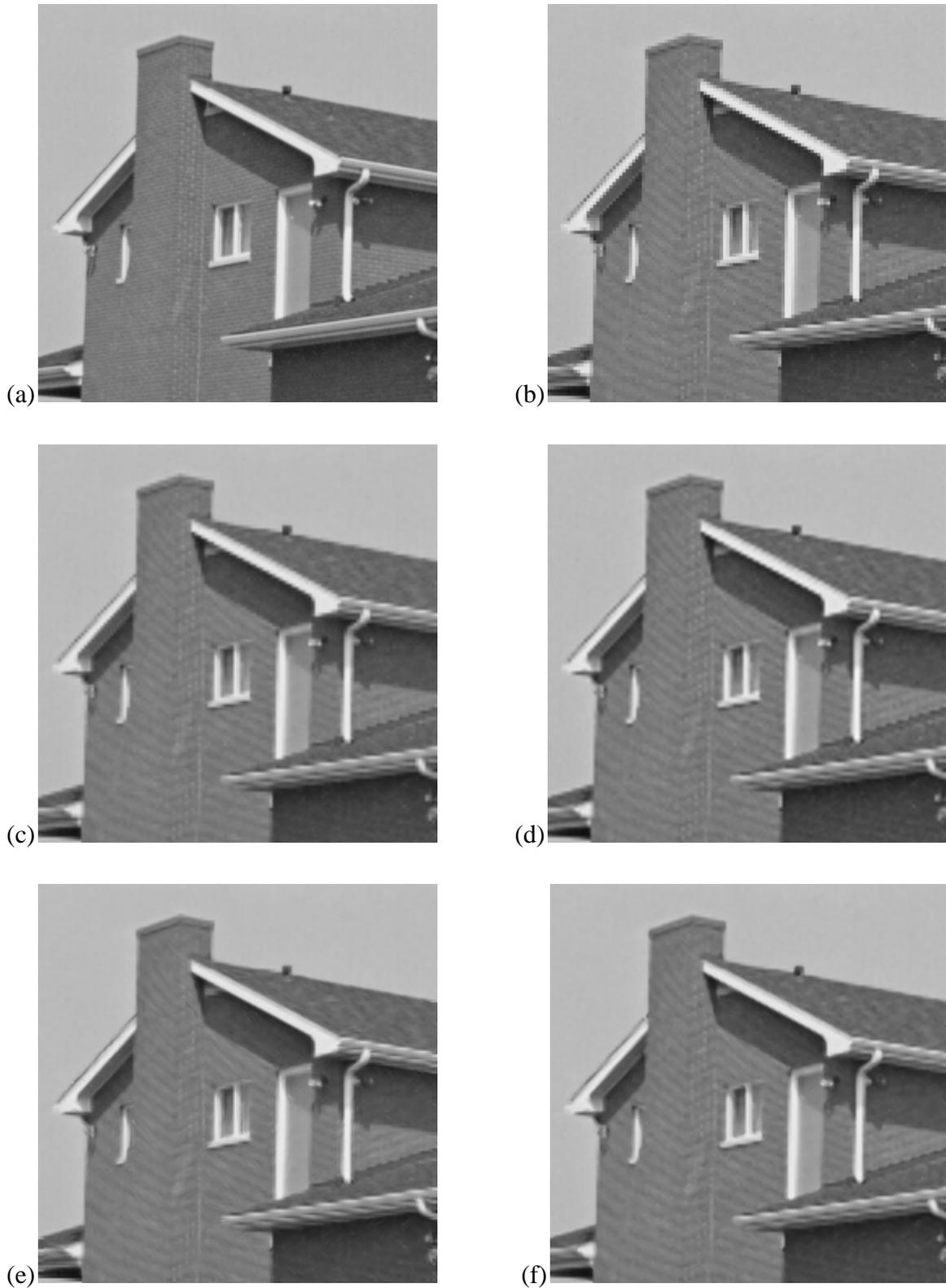


Figure 8. Interpolation results of the image House. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF

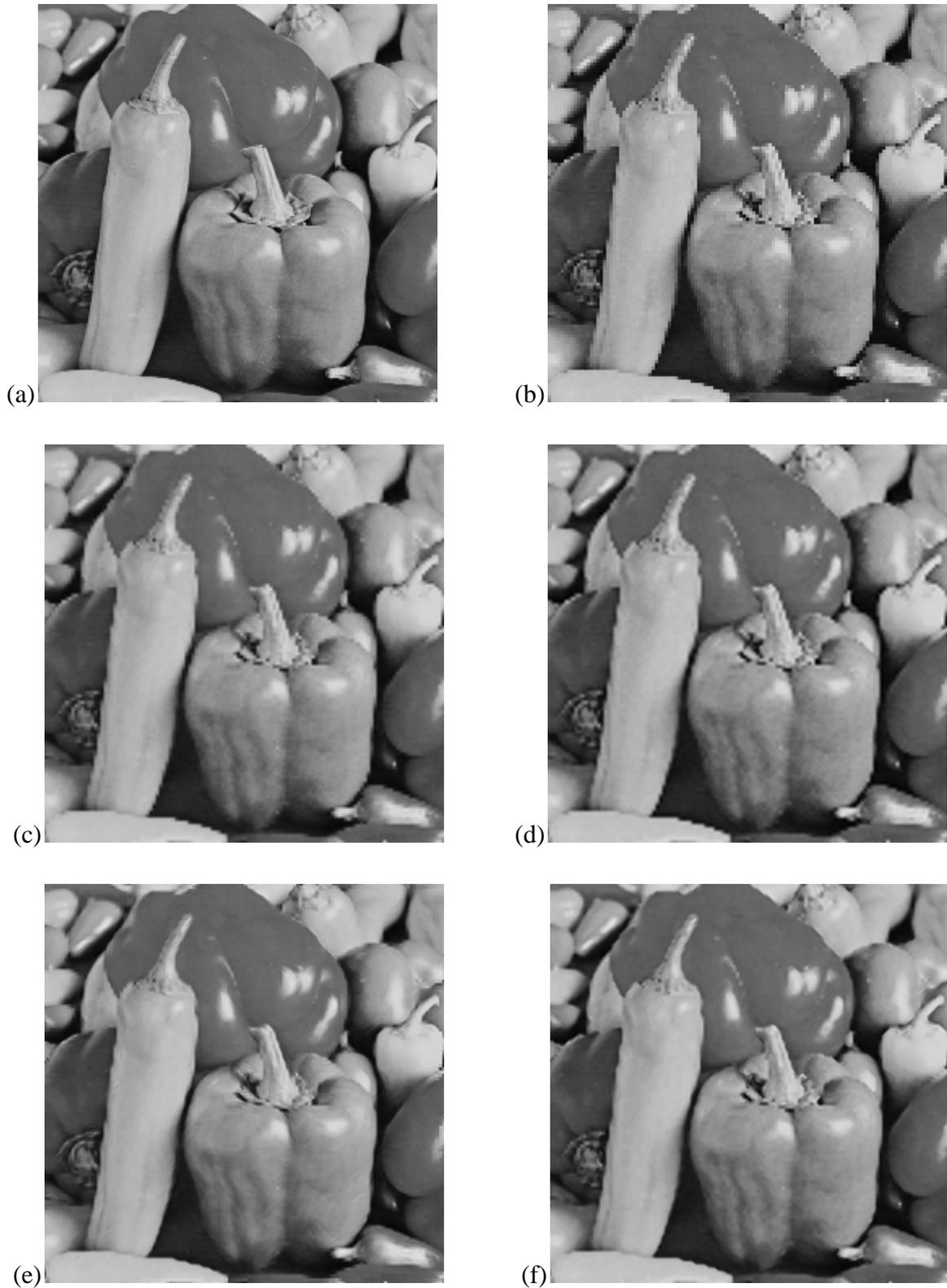


Figure 9. Interpolation results of the image Peppers. (a) Original Image; interpolated image by (b) Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF



Figure 10. Interpolation results of the image Lena. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF



(a)



(b)



(c)



(d)



(e)



(f)

Figure 11. Interpolation results of the image Cameraman. (a) Original Image; interpolated image by (b) Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF



Figure 12. Interpolation results of the image Barbara. (a) Original Image; interpolated image by (b) Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF

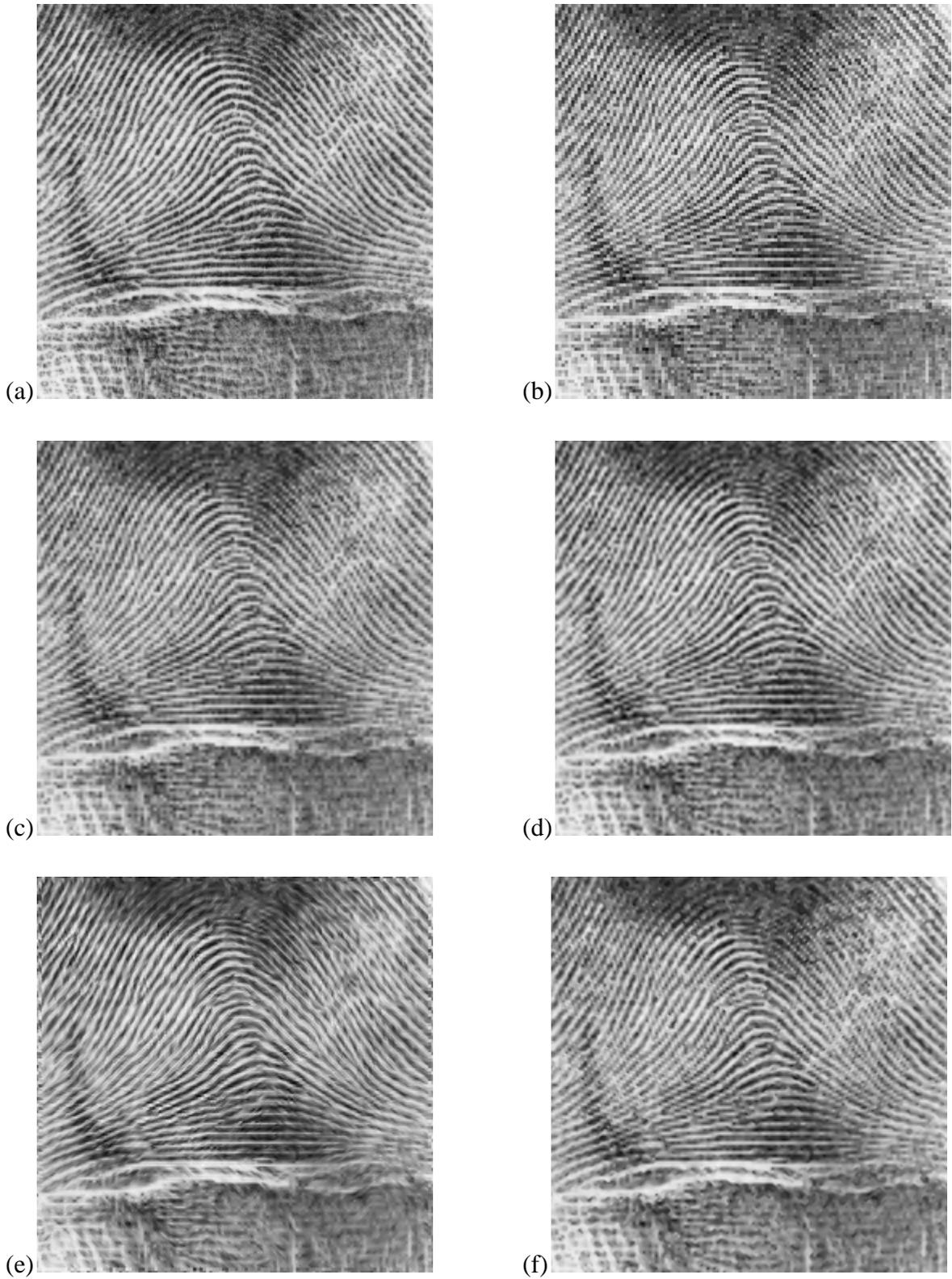


Figure 13. Interpolation results of the image Fingerprint. (a) Original Image; interpolated image by (b)Nearest Neighbor; (c) Bilinear; (d) Bicubic; (e) NEDI; (f) DFDF

4.2 Discussions

The interpolation algorithms New Edge Directed Interpolation and An Edge Guided Image Interpolation Algorithm via Directional Filtering and Data Fusion were tested and their performance was compared with the non-adaptive interpolation techniques like Nearest Neighbor interpolation, Bilinear interpolation and Bicubic Interpolation. The Original image is down sampled along each dimension to get the corresponding low resolution image as shown in the fig. 4.8 which shows the formation of low resolution image from the high resolution image by direct downsampling. The black dots represent the low resolution image $I_l(n,m)$ and the white dots represent the missing high resolution image pixels $I_h(2n,2m)$ due to downsampling.

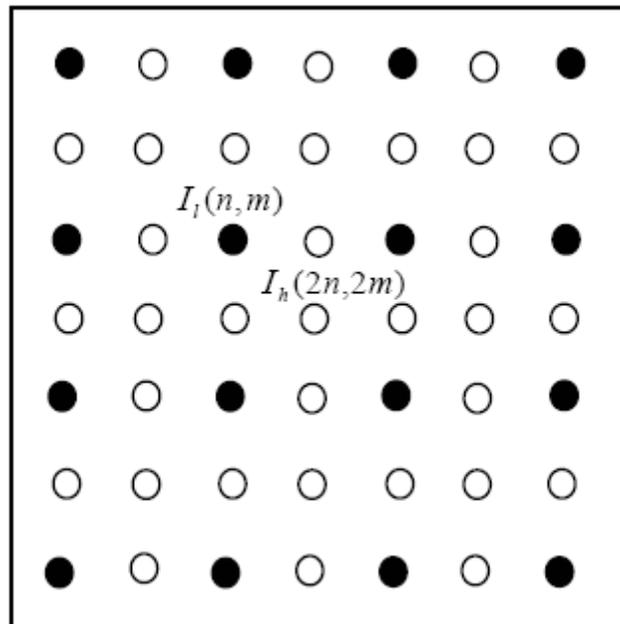


Figure 14. Formation of the low resolution image (LR) from the High Resolution (HR) [5]

Then the next step is to perform interpolation and the high resolution version of the image is recovered by using the interpolation algorithms which are specified above. This way we have the original high resolution image available with us so that we can compare both the

versions of the image. But during the process of downsampling aliasing might be introduced. The performance metrics we consider are the mean square error (MSE) and Structural Similarity Index (SSIM). These values are calculated by taking the original image before it is being downsampled as a reference and the values are calculated as discussed before in this chapter.

It has been observed that these metrics which measure the objective quality of the image are not the correct metrics to assess the visual quality of images. We can see that the quality of the image has improved when we compare the results of Bilinear and The New Edge Directed Interpolation but the Mean Square Error is greater which is supposed to be less. Hence it is not a good metric to assess the image quality.

In a similar way it is observed that the metric Structural Similarity index is greater for the Bilinear Interpolation compared to the new edge directed interpolation for some of the test images, but the subjective image quality of the output image is better for new edge directed interpolation. Hence this also is not a good metric for assessing the image quality. Therefore we rely on the subjective quality of the images i.e., we depend on the subjective evaluation to assess the visual quality of the output interpolated images for comparing the different algorithms present in literature since there is not always a correlation between the image quality assessment metrics and the visual appearance of the output interpolated images. The visual quality improvement obtained by the Adaptive algorithms, new edge directed interpolation and edge guided interpolation methods can be observed when we view the images.

We have chosen for our study, a variety of images like images which have great texture content and some images which are edge dominant and some smooth images and some images having content which is mixture of all the three said characteristics. We have taken a set

of 256 X 256 images and downsampled them to 128 x 128 and then applied the various adaptive and non-adaptive interpolation techniques to obtain the high resolution version of the downsampled image.

The Mean Square Error and the Structural Similarity Index values for different adaptive and non-adaptive image interpolation algorithms are presented in the tables and as discussed before these are not the reliable metrics to measure the visual quality of the interpolated image. So we base all our conclusions based on the subjective evaluations of the visual quality of the interpolated image.

It has been observed that the Non-Adaptive interpolation techniques like the Nearest Neighbor, Bilinear and Bicubic interpolation techniques produce output image with different problems like aliasing, blurring and zigzagging along the edges or the object boundaries. However when we compare these three non-adaptive interpolation techniques the edge sharpness is good in bicubic interpolation when compared to the nearest neighbor and bilinear interpolation algorithms, i.e., Bicubic interpolation gives better results in terms of edge smoothness along the object boundaries. The nearest neighbor interpolation gives output images which are highly corrupted in terms of the artifacts introduced as a result of interpolation also the mean square error is highest for this method of interpolation and the corresponding Structural Similarity Index being the lowest for the nearest neighbor interpolation.

The Adaptive Interpolation algorithms like the edge directed interpolation and the edge guided interpolation produce far better interpolated images when compared to the non-adaptive interpolation techniques. The result of the adaptive interpolation techniques is smoother edge transitions and reduced blurring near the object boundaries. When we view the interpolated

images, the images appear to have reduced artifacts when compared to the non-adaptive interpolation techniques.

When we compare the outputs of the adaptive and non-adaptive interpolation techniques we observe that there is a significant improvement in the visual quality in terms of sharpness of the edges and blurring effects, even in the objects which are in the background of the still image (considering the background in the cameraman image). The blurring effect also is reduced in case of adaptive interpolation techniques. We can see a huge improvement in the reduction of these annoying artifacts.

In edge dominant images like the house image or the peppers image we see a huge improvement in the visual quality of images since these are edge dominant images and the non-adaptive interpolation techniques produce blurry edges the image looks messy and so we see a significant improvement in the output image quality, when we go for Adaptive Interpolation techniques. The blurring effect at the edges has significantly reduced in case of the adaptive interpolation techniques. When you observe the Lena image the blurring at the edge the hat of lena has noticeable reduced. We can observe that the ringing artifacts have been greatly suppressed in the new edge directed interpolation, which is due to the orientation adaptation property of the algorithm.

In images which are texture based i.e., images with lots of horizontal and vertical edges like the Barbara image there is not much improvement in the visual quality of the output interpolated images. For the textured regions it has been observed that all algorithms give results in the same way except for some minor improvement. So in the Barbara image we see that all the algorithms perform in the same way except for the non textured regions where the edge directed

interpolation and edge guided interpolation outperform the non-adaptive interpolation techniques and give more smooth edges.

When we compare the Mean Square Error for the edge directed interpolation and the edge guided interpolation techniques, the mean square error values for all images is greater for the edge directed interpolation compared to the edge guided interpolation. And the Structural Similarity Index is greater for the edge guided interpolation when compared to the edge directed interpolation. Both of them indicate that the edge guided interpolation proposed by L. Zhang and X. Wu performs in a better way than the edge directed interpolation algorithm proposed by Li and Orchard. But when we view the images there is not much difference in the subjective quality of the output interpolated images. However it has been observed that the algorithm proposed by Li and Orchard preserves large edge structures such as those in lena and cameraman images.

We have observed that though the image quality is improved, the value of Mean Square Error has not been reduced so we have performed different tests by taking the average of two different algorithms and trying out all the different combinations and noted the mean square error and the structural similarity index. It has been observed that the mean square error values has decreased but for example consider the combination of average of bilinear interpolation and the edge guided interpolation for the cameraman image the value of mean square error and the structural similarity index have shown some improvement but the edges in the image have been slightly blurry. When we consider the combination of edge directed interpolation and edge guided interpolation the averaged output is shown in Fig. 4.9 (f). It has been observed that there is an improvement in the visual quality of the averaged output but it is not very significant and also the computational complexity will be huge since we have to implement both the algorithms so it is not preferred.



Figure 15. Interpolation results of the image Cameraman. (a) Original Image; interpolated image by (b)Bilinear; (c) DFDF; (d) Average of Bilinear and DFDF; (e) NEDI; (f) Average of DFDF and NEDI

The images are shown in fig 4.9 and the values of mean square error and the structural similarity index for different combinations of averaging of different algorithms are shown in tables 4.3 and 4.4.

Table 3. The MSE values computed by considering the interpolation as an average of two different algorithms

Combination	Cameraman	House	Lena	Barbara	Peppers	Boat	Fingerprint
Bilinear & Bicubic	186.5097	39.5411	27.4597	223.079	64.0746	77.3592	55.9615
Bilinear & NEDI	177.8643	39.6246	25.8962	267.008	60.1115	74.6445	69.1633
Bicubic & NEDI	176.4517	38.5254	24.2552	284.084	58.1622	72.9967	56.3643
Bilinear & DFDF	176.7926	37.2333	26.5059	213.622	59.0478	75.3253	59.2859
Bicubic & DFDF	177.2933	36.5147	25.0768	227.144	57.7265	74.5782	48.9076
NEDI & DFDF	170.9366	37.2408	24.5195	270.956	55.5104	72.5903	64.7269

Table 4. The MSE values computed by considering the interpolation as an average of two different algorithms

Combination	Cameraman	House	Lena	Barbara	Peppers	Boat	Fingerprint
Bilinear & Bicubic	0.8647	0.8815	0.9146	0.7999	0.9126	0.8423	0.9558
Bilinear & NEDI	0.8678	0.8827	0.9172	0.789	0.9173	0.8424	0.9462
Bicubic & NEDI	0.8686	0.8803	0.9184	0.7853	0.9182	0.8478	0.9547
Bilinear & DFDF	0.8684	0.8829	0.9155	0.8021	0.9168	0.8441	0.952
Bicubic & DFDF	0.8684	0.8798	0.9163	0.7986	0.9171	0.8443	0.9588
NEDI & DFDF	0.8702	0.88	0.9175	0.7869	0.9203	0.8466	0.948

4.3 Computational Complexity

The most common traditional image interpolation techniques such as nearest neighbor, bilinear and bicubic require only small amount of computation. However, since they are based on a simplified slow varying image model, they often produce interpolated images with various artifacts like aliasing, blurring and zigzagging of edges [14]. When we consider the edge directed interpolation proposed by Li and Orchard the main drawback of this algorithm is its huge computational complexity. For a local window size of 8 in order to compute the interpolation coefficients, it requires nearly 1300 multiplications per pixel [4]. So if we apply this covariance based adaptive technique to all the pixels then the overall complexity would be increased so we have used a hybrid approach where in the covariance based adaptation method is applied only to the edge pixels and for the non edge pixels we use the bilinear interpolation.

Since the number of edge pixels in an image are only a small fraction of the whole image, reduce the computational complexity is reduced to a great extent. And in order to compute whether a pixel is edge or a non edge pixel we use an activity measure, if it is above a particular threshold that pixel is considered to be edge pixel else a non edge pixel. And the computation of activity measure requires very less number of computations compared to the calculation of covariance coefficients [4]. In the Edge Guided interpolation proposed by L. Zhang and X. Wu the simplified algorithm requires 24 additions, 4 multiplications and 2 divisions [5]. Though these methods require comparatively high computational needs still they are used because of their good performance.

5. CONCLUSIONS AND FUTURE WORK

In the present work different image interpolation algorithms are compared and the results of various algorithms are shown in the previous chapter. And the Mean Square Error and the Structural Similarity Index values are also given in the tables in the previous chapter. In the Non-Adaptive interpolation techniques only the local pixel characteristics are considered to estimate the missing high resolution pixel so they produce artifacts like in the interpolated image but where as in the Adaptive Interpolation techniques pixel orientation is considered in order to compute the missing pixel so we get an interpolated image with much better sharper edges with reduced blurring. The Non-Adaptive Interpolation techniques fail to interpret the edges properly there by producing images with blurred edges so they are preferred only when there are limitations with the complexity. But with the evolution of fast running modern day processors it has become feasible to use more robust and complex algorithms which are computationally more complex and yielding better results without the artifacts.

The future work can be the New Edge Directed algorithm can be extended to reconstruct the color image from the samples produced from the charge coupled device (CCD). Reconstructing the color image from the CCD samples is demosaicking. The Edge Directed Interpolation can be used for this application also.

We can test the algorithms by first downsampling using the `imresize` function and then apply the different algorithms and compare their performance.

The two techniques DFDF and NEDI can be fused by taking two low resolution frames input frames of the same image and then bringing them to a common reference and

applying the two interpolation algorithms and taking an average of both the results to get the final interpolated image.

We can use the sharpness metric to quantify the performance of interpolation techniques since the mean square error and the structural similarity index metrics are not always correct in assessing the performance of the algorithm.

These interpolation techniques can be applied to compressed images. With the evolution of hand held devices having the capability of video playback and due to their limited display capabilities the video size is very small and the compression ratio is large so there is a need for the images to be interpolated for television play back.

REFERENCES

1. U.S. Congress, Office of Technology Assessment, *The Big Picture: HDTV and High-Resolution Systems*, **OTA-BP-CIT-64** (Washington, DC: U.S. Government Printing Office, June 1990).
2. Sean McHugh, Digital Image Interpolation, [Online]
<http://www.cambridgeincolour.com/tutorials/image-interpolation.htm>
3. Kwan Pyo Hong, Joon Ki Paik, Hyo Ju Kim, and Chul Ho Lee, "An Edge-Preserving Image Interpolation System for a Digital Camcorder", *IEEE Transactions on Consumer Electronics*, Vol. 42, No. 3, pp. 279-284, August 1996.
4. Xin Li and Michael T. Orchard, "New Edge-Directed Interpolation", *IEEE Transactions on Image Processing*, Vol. 10, No. 10, pp. 1521-1527, October 2001.
5. Lei Zhang and Xiaolin Wu, "An Edge Guided Image Interpolation Algorithm via Directional Filtering and Data Fusion", *IEEE Transactions on Image Processing*, Vol. 15, No. 8, pp. 2226-2238, August 2006.
6. Sergio Carrato and Livio Tenze, "A High Quality 2 X Image Interpolator", *IEEE Signal Processing Letters*, Vol. 7, No. 6, pp. 132-134, June 2000.
7. Xiangjun Zhang and Xiaolin Wu, "Image Interpolation by Adaptive 2-D Autoregressive Modeling and Soft-Decision estimation", *IEEE Transactions on Image Processing*, Vol. 17, No.6, pp. 887-896, June 2008.
8. B. Migeon and P. Marche, "In Vitro 3D Reconstruction of Long Bones Using B-scan Image Processing," *Medicine and Biological Engineering and Computers* , Vol. 35, pp 369-372, July, 1997.
9. Z. Wang, A. C. Bovik, H.R. Sheikh and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Transactions on Image Processing*, Vol. 13, no. 4, pp. 600-612, Apr. 2004.
10. Tinku Acharya, Ping-Sing Tsai, "Computational Foundation of Image Interpolation Algorithms," *ACM Ubiquity* Vol 8, 2007.
11. P. Thevenaz, T. Blu and M. Unser, "Image Interpolation and Resampling," *Handbook of Medical Image Processing*, Academic Press, New York, 2003.

12. J. Anthony Parker, Robert V. Keyon and Donald E. Troxel, "Comparison of Interpolating Methods for Image Resampling," IEEE Transactions on Medical Imaging, Vol. MI-2, No. 1, pp. 31-39, March 1983.
13. M. Haddad and G. Porenta, "Impact of orientation Algorithms on Quantitative Myocardial SPECT Perfusion Imaging," Journal of Nuclear Medicine, Vol. 39, pp. 1864-1869, 1998.
14. L. Zhang and XL. Wu, "Image Interpolation Using Multiscale Geometric Representations," IEEE Transactions on Image Processing, Vol.15, Issue 8, pp. 2226-2238, Aug. 2006.
15. K. Jensen and D. Anastassiou, "Subpixel Edge Localization and the Interpolation of Still Images," IEEE Transactions on Image Processing, Vol 4, No. 3, pp 285-295, March 1995.
16. Y. Wang and S. Mitra, "Edge Preserved Image Zooming," Proc. of European Signal Process, EURASIP-88, pp.1445-1448, Grenoble, France, 1988.
17. S. G. Mallat, A wavelet tour of signal processing. New York: Academic, 1998.
18. V. R. Algazi, G. E. Ford, and R. Potharlanka, "Directional Interpolation of images based on Visual Properties and Rank Order Filtering," in Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing, Vol. 4, 1991, pp. 3005-3008.
19. J. E. Adams Jr, "Interactions Between Color Plane Interpolation and other Image Processing Functions in Electronic Photograph," Proc. SPIE, Vol.2416, pp. 144-151, 1995.
20. N. Jayant and P. Noll, Digital Coding of Waveforms: Principles and Applications to Speech and Video. Englewood Cliffs, NJ: Prentice-Hall, 1984.
21. X. Li and M. Orchard, "Edge Directed Prediction for Lossless Compression of Natural Images," IEEE Trans. Image Processing, Vol. 10, pp. 813-817, June 2001
22. E. Meijering, "A Chronology of Interpolation: from Ancient Astronomy to Modern Signal and Image Processing," Proc, Of The IEEE, Vol. 90, pp. 319-342, Mar. 2002.
23. B. Vrcelj and P. P. Vaidyanathan, "Efficient Implementation of all-digital Interpolation," IEEE Trans. Image Processing, Vol. 4, pp. 285-295, Mar. 1995.

24. Y. Takahashi and A. Taguchi, "An Enlargement Method of Digital Images with the Prediction of High-Frequency Components," Proceedings of International Conference on Acoustics, Speech, and Signal Processing, Vol. 4, pp. 3700-3703, 2002.
25. Ying Zhu, S. C. Schwartz and M. T. Orchard, "Wavelet Domain Image Interpolation via Statistical Estimation," Proceedings of International Conference on Image Processing, Vol. 3, pp. 840-843, 2001.
26. D. D. Muresan, and T. W. Parks, "Prediction of Image Detail," Proceedings of International Conference on Image Processing, Vol. 2, pp. 323-326, 2000.
27. D. D. Muresan, "Fast Edge Directed Polynomial Interpolation," Proceedings of International Conference on Image Processing, Vol. 2, pp. 990-993, 2005.
28. William K. Pratt, Digital Image Processing, John Wiley & Sons, 2nd edition, April 1991.
29. I. Pitas, Digital Image Processing Algorithms and Applications, John Wiley & Sons, Inc. 2000.
30. R. G. Keys, "Cubic Convolution Interpolation for Digital Image Processing," IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP-29, pp. 1153-1160, June 1981.
31. G. Ramponi and S. Carrato, "Interpolation of the DC Component of Coded Images Using a Rational Filter," in Proc. ICIP'97, Santa Barbara, CA, Oct. 1997, pp. 1-389-1-392.
32. S. Carrato, G. Ramponi, and S. Marci, "A Simple Edge-Sensitive Image Interpolation Filter," in Proc. ICIP-96, Lausanne, Switzerland, Sept. 1996, pp. 111-711-111-714.
33. K. Jensen, and D. Anastassiou, "Spatial Resolution Enhancement of Images Using Nonlinear Interpolation," Proceedings of International Conference on Acoustics, Speech and Signal Processing, ICASSP-90, pp.2045-2048, Albuquerque, NM, 1990.
34. S. Thurnhofer, and S. Mitra, "Edge-Enhanced Image Zooming," Optical Eng., 35(7), pp. 1862-1869, 1996.
35. S. Screiner, C. B. Paschal and R. L. Galloway, "Comparison of Projection Algorithms used for the construction maximum intensity projection images," J. Computer Assisted Tomography, Vol. 20, No. 1, pp. 56-57, 1996.