

# Adaptive image interpolation

Ashok Kumar Murmu



Department of Computer Science and Engineering  
National Institute of Technology Rourkela  
Rourkela – 769 008, India

# Adaptive image interpolation

Dissertation submitted in

*May 2014*

*to the department of*

***Computer Science and Engineering***

*of*

***National Institute of Technology Rourkela***

*in partial fulfillment of the requirements*

*for the degree of*

***Master of Technology***

*by*

***Ashok Kumar Murmu***

*(Roll 212CS1094)*

*under the supervision of*

***Dr. Pankaj Kumar Sa***



Department of Computer Science and Engineering

National Institute of Technology Rourkela

Rourkela – 769 008, India

*dedicated to my Parents*



Computer Science and Engineering  
**National Institute of Technology Rourkela**

Rourkela-769 008, India. [www.nitrkl.ac.in](http://www.nitrkl.ac.in)

**Prof. Pankaj Kumar Sa**

Asst. Professor

## Certificate

This is to certify that the work in the thesis entitled *Adpative Image Interpolation* by *Ashok Kumar Murmu*, bearing roll number 212CS1094, is a record of his research work carried out by him under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of *Master of Technology* in *Computer Science and Engineering*.

***Pankaj Kumar Sa***

# Declaration

I, **Ashok Kumar Murmu** (Roll **212cs1094**) understand that plagiarism is defined as any one or the combination of the following

1. Uncredited verbatim copying of individual sentences, paragraphs or illustrations (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.

2. Uncredited improper paraphrasing of pages or paragraphs (changing a few words or phrases, or rearranging the original sentence order).

3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did or wrote what. (Source:IEEE, the Institute, May 2014)

I have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of my work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.

I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such a complaint occurs. I understand fully well that the guide of the thesis may not be in a position to check for the possibility of such incidences of plagiarism in this body of work.

Date:

**Ashok Kumar Murmu**

Master of Technology

Computer Science and Engineering

NIT Rourkela

# Acknowledgement

This thesis work is a realization of support and guidance of people to whom i owe my sincere gratitude. To name a few and thank them will not be sufficient.

Certainly it would have been difficult without the involvement and encouragement of *Prof Pankaj Sa.* His profound research insight has guide me towards the completion of this thesis work.

Association with esteemed professors like *Prof. B. Majhi, Prof. S.K. Rath, Prof. S.K. Jena, Prof. D.P. Mohapatra, Prof. A.K. Turuk, Prof. S. Chinnara, Prof. B.D. Sahoo* has always been a constant source of inspiration for my thesis work.

I extend my thanks to my friends and research colleagues whose companion and support always proved beneficial. Special thanks to Suman, Biswojit, Rohit, Saroj, Rajesh for their suggestions and improvements in my thesis work.

Finally, it is my parents blessings and love that always provided motivation at difficult times.

*Ashok Kumar Murmu*

## **Abstract**

Simple interpolation techniques like nearest neighbor, bilinear, bicubic in the past had gained popularity due to their simplicity and low computational cost. But with the advent of high performing machines, demand for better interpolation methods at the expense of their computational complexity has arisen. In this endeavor, myriads of interpolation methods have been introduced. Some of which are based on edge intensity, curvature profile of image, fuzzy logic. While others are optimized for the particular needs like resistance to outliers, performance in real time basis etc. An extensive list of interpolation methods exists in literature. We have reviewed an adaptive interpolation technique based on Newton forward difference. This difference provides a measure of goodness for grouping of pixels around the target pixel for interpolation.

# Contents

Certificate	iii
Declaration	iv
Acknowledgment	v
Abstract	vi
List of Figures	ix
<b>1 Introduction</b>	<b>2</b>
1.1 Introduction . . . . .	2
1.1.1 Theory of Interpolation . . . . .	2
1.1.2 Resampling versus interpolation . . . . .	4
1.1.3 Sinc as ideal filter . . . . .	4
1.1.4 Basic Concepts . . . . .	4
1.2 Literature Review On Image Interpolation Techniques . . . . .	10
1.2.1 New edge-directed interpolation . . . . .	10
1.2.2 Improved new edge-directed interpolation . . . . .	10
1.2.3 Modified edge-directed interpolation . . . . .	11
1.2.4 Robust new edge-directed interpolation . . . . .	11
1.2.5 Edge-guided image interpolation via directional filtering and data fusion . . . . .	12
1.2.6 Fast algorithm for image interpolation . . . . .	12
1.2.7 Canny edge based image expansion . . . . .	12
1.2.8 Edge-directed inpterpolation . . . . .	13
1.2.9 Data dependent triangulation . . . . .	13
1.2.10 Iterative curvature-based interpolation . . . . .	14



1.2.11	Single image interpolation technique . . . . .	15
1.2.12	Real-time image resizing . . . . .	15
1.2.13	Curvature Interpolation Method for Image Zooming . . . . .	15
1.2.14	An edge-directed bicubic interpolation algorithm . . . . .	16
1.2.15	Soft adaptive gradient angle interpolation of gray scale image	16
1.2.16	Efficient super resolution using edge directed unsharp masking sharpening method . . . . .	16
1.2.17	A switching based adaptive image interpolation algorithm .	17
1.3	Motivation . . . . .	18
1.4	Thesis Layout . . . . .	18
<b>2</b>	<b>An Adaptive Image Interpolation Technique</b>	<b>19</b>
2.1	Mathematical Background . . . . .	19
2.2	Forward difference based adaptive interpolation . . . . .	21
2.3	Results . . . . .	23
2.4	Result analysis . . . . .	28
2.5	Summary . . . . .	28
<b>3</b>	<b>Conclusions and Future Work</b>	<b>29</b>
3.1	Conclusion . . . . .	29
3.2	Future work . . . . .	29

# List of Figures

1.1	Interpolation using a function . . . . .	3
1.2	Nearest neighbor interpolation (a) Kernel (b) Magnitude of Fourier transform (c) Logarithmic plot of magnitude [5] . . . . .	5
1.3	Linear interpolation. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude [5] . . . . .	6
1.4	Cubic spline interpolation with $a=-1$ .(a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude [2] . . . . .	7
1.5	Cubic spline interpolation with $a=-0.5$ (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude [2] . . . . .	8
1.6	Low order B-Splines [3] . . . . .	8
1.7	Cubic B-spline interpolation. (a) Kernel plotted for $ x  < 3$ . (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude [5] . . . . .	9
1.8	DDT Mesh [16] . . . . .	14
2.1	Relationship between source pixels and target pixel . . . . .	21
2.2	Grouping of pixels for second order interpolation . . . . .	22
2.3	Grouping of pixels for fourth order interpolation . . . . .	22
2.4	Lena $256 \times 256$ interpolated by a factor of 2 . . . . .	25
2.5	Barbara $512 \times 512$ Interpolated by factor of 2 . . . . .	26
2.6	Barbara $512 \times 512$ Interpolated by factor of 4 . . . . .	27

# List of Tables

2.1	PSNR dB of $256 \times 256$ images for resizing factor 2 . . . . .	23
2.2	PSNR dB of $256 \times 256$ images for resizing factor 4 . . . . .	23
2.3	PSNR dB of $512 \times 512$ images for resizing factor 2 . . . . .	24
2.4	PSNR dB of $512 \times 512$ images for resizing factor 4 . . . . .	24

# Chapter 1

## Introduction

### 1.1 Introduction

Digital Image is a discrete representation of its continuous counterpart perceived through our eyes, a camera or any such devices. Its representation and processing in computer requires to stored it in digital format. Sampling the image for computer storage often degrades its visual representation in a variety of display units. So the image needs further processing to suit our demands. Image interpolation is one such image processing task to find the values of the pixels of the image which are not originally present in the image. It finds application in medical image processing like X-ray imaging, representation of multimedia content in web, satellite images processing for weather forecasting, industrial inspection for defective manufactured parts which requires image resizing, high resolution. In this technological endeavor several interpolation techniques have been developed ranging from very simple to highly complex techniques. Image interpolation become the preprocessing step for other image processing tasks like image registraions, image rotation. Image registration needs interpolation to accurately register the image at subpixel level.

#### 1.1.1 Theory of Interpolation

Image interpolation can be defined as fitting of a continuous function through discrete points in digital image [2]. Since the interpolation function is continuous we can find the new pixel value at our desired location. In other words

interpolation reconstructs the pixel values that are lost in sampling by smoothing with an interpolation function.

Mathematical representation of interpolation for equally spaced data is:

$$f(x) = \sum_{k=0}^{k-1} c_k h(x - x_k) \quad (1.1)$$

where  $h$  is the interpolation kernel, which is will be weighted by the value of the coefficient  $c_k$ . This weighted value will be finally assigned to  $k$  data samples,  $x_k$ . Above equation is a convolution operation where  $h$  is symmetrical. To compute the interpolated pixel value of  $x$  we need to find the sum of the values of the discrete input scaled by the corresponding values of the interpolation kernel i.e convolution of data sample with interpolation function.

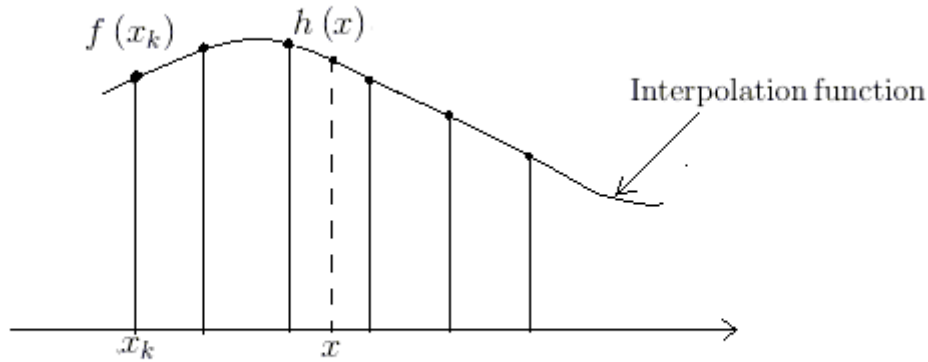


Figure 1.1: Interpolation using a function

Performance of Interpolation kernels measured by analyzing their frequency domain characteristics after applying the Fourier transformation. In frequency domain two characteristics are examined:-passband characteristics and stopband characteristics. Passband represent the range of frequencies that are allowed to pass through and stopband restrict the frequencies that are not permitted to pass. Magnitude of Fourier transform show the passband performance while logarithmic plot of magnitude represent the stopband performance. Criteria for passband is that its deviation from constant gain should be small. Large deviation due to constant gain will cause blurring due to attenuation. While criteria for stopband

restricts interpolation function to have small side lobes as prominent side lobes causes aliasing.

### 1.1.2 Resampling versus interpolation

Image resampling converts an image from one coordinate system to another. Image resampling actually consists of two steps. Image reconstruction followed by sampling. The image reconstruction fits the image samples by a continuous function. Reconstruction step is also called interpolation. Once the image is reconstructed, it can be sampled at any desired point. In reconstruction step, input signal is convolved with a continuous interpolation function.

### 1.1.3 Sinc as ideal filter

Frequency domain analysis of interpolation kernels implies that in passband it should have almost a constant value with very low gain and zero value within the stopband. Sinc function satisfies these two criteria but due its infinite nature in spartial domain it practically cannot be realized. Sinc function represented as:

$$h(x) = \frac{\sin(\pi x)}{\pi x} \quad (1.2)$$

It is difficult to convolve the signal with a infinite function. So sinc function is approximated for realization. This approximated sinc become non rectangular in frequency domain and also cause sidelobes in stopband.

### 1.1.4 Basic Concepts

Intepolation fits a continous function through sampled data point. Interpolation techniques are mainly divided in two categories:

- Non-adaptive techniques
- Adaptive techniques

#### 1.1.4.1 Nearest Neighbor

This is the simplest form of interpolation, where the interpolated pixel value determined by nearest neighbor in the proximity. Simplicity of calculation is the reason for its cheap computational cost. This interpolation also called pixel replication [1]. The interpolation kernel is given by [3]

$$h(x) = \begin{cases} 1 & 0 \leq |x| < 0.5 \\ 0 & 0.5 \leq |x| \end{cases} \quad (1.3)$$

In one dimension variation it takes two pixel into account however for two dimension image it takes four pixels into account for interpolation.

Fourier analysis of Nearest Neighbor

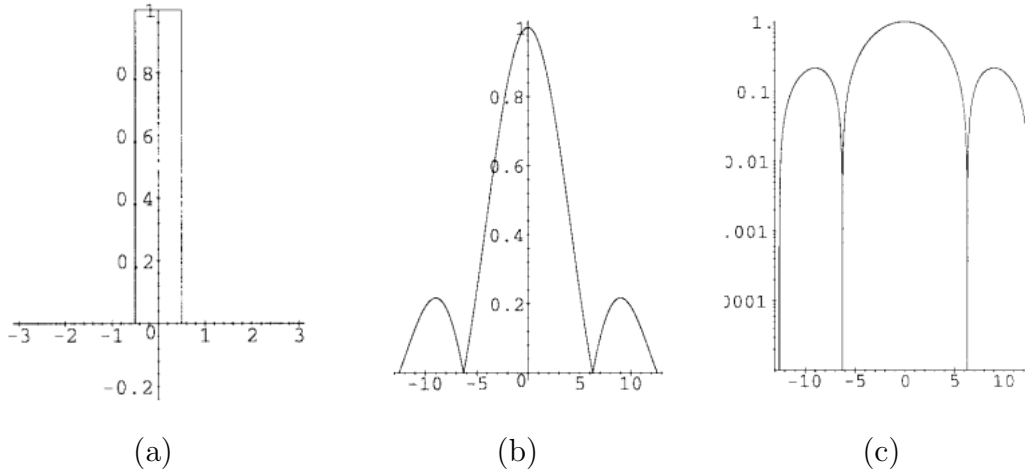


Figure 1.2: Nearest neighbor interpolation (a) Kernel (b) Magnitude of Fourier transform (c) Logarithmic plot of magnitude [5]

Since nearest neighbor interpolation kernel has prominent side lobes, it has poor frequency domain response. Nearest neighbor interpolation called point shift algorithm as the interpolated image shifted with respect to original image by difference between positions of coordinate locations. High magnification of image using nearest neighbor causes the image to look blocky.

#### 1.1.4.2 Bilinear

As the name suggests, it is linear interpolation in two directions, first in horizontal direction then by a vertical direction or vice-versa. Bilinear interpolation uses weighted average of the 4 neighborhood pixels to calculate its final interpolated

pixel. Bilinear interpolation performs better than NN as reduction of the stair-case effect makes the image looks smother. However, blurring effect is occurred by averaging the surround pixels. Since the passband is attenuated moderately, it causes smoothing of image. Interpolation kernel for linear interpolation samples the input with the following kernel [3]

$$h(x) = \begin{cases} 1 - |x| & 0 \leq |x| < 1 \\ 0 & 1 \leq |x| \end{cases} \quad (1.4)$$

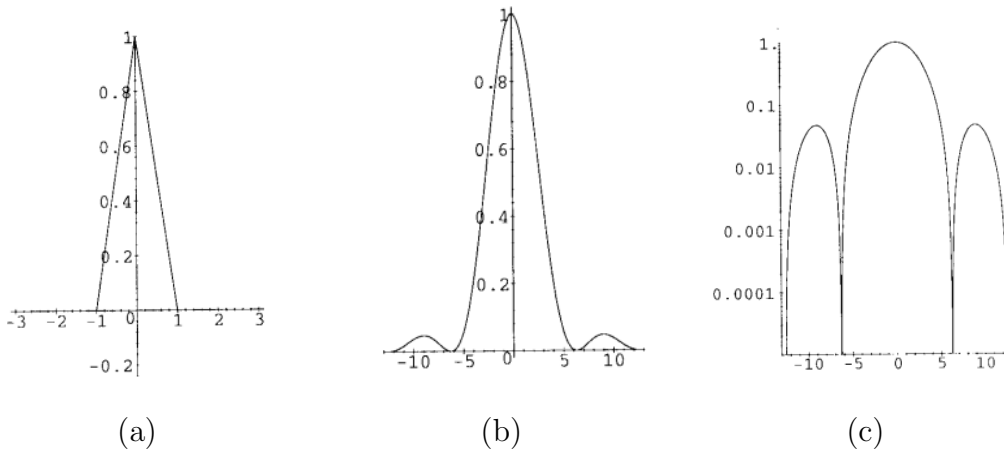


Figure 1.3: Linear interpolation. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude [5]

Linear interpolation kernel perform better than the nearest neighbor. In frequency domain it has less prominent side lobes, improving stopband performance. Linear interpolation is widely used as it provides reasonably good image quality at moderate cost. Frequency domain response is better than nearest neighbor. But aliasing of data due to attenuation of the high-frequency component at cutoff frequencies still occurs.

#### 1.1.4.3 Bicubic

Bicubic interpolation is the cubic interpolation in two dimension. Bicubic interpolation can be realized using Lagrange polynomial, cubic splines, etc. Third degree cubic spline interpolation formula serve as a good approximation to ideal sinc function. The interpolated image so obtained is smoother than NN and bilinear interpolation. Bicubic interpolation takes a weighted average of the 16 pixels to calculate gray value of pixel.



Cubic Interpolation kernel is derived from interpolation formula for general cubic spline by imposing constraints. The cubic interpolation kernel is [3]

$$h(x) = \begin{cases} (a+2)|x|^3 - (a+3)|x|^2 + 1 & 0 < |x| < 1 \\ a|x|^3 - 5a|x|^2 + 8a|x| - 4a & 1 < |x| < 2 \\ 0 & 2 < |x| \end{cases} \quad (1.5)$$

Where  $a$  is the free parameter. The value for this parameter determine the kernel performance. Choice for value of  $a$  between -3 and 0 approximate the sinc function kernel. Value of  $a = -1$  will amplify the frequencies at the upper-end of the passband. This amplification causes image sharpening. Robert Keys determined this constant by forcing the Taylor series expansion of the sampled sinc function to agree in as many terms as possible with the original signal, resulting in  $a=-0.5$ . This amplification cause image sharpening.

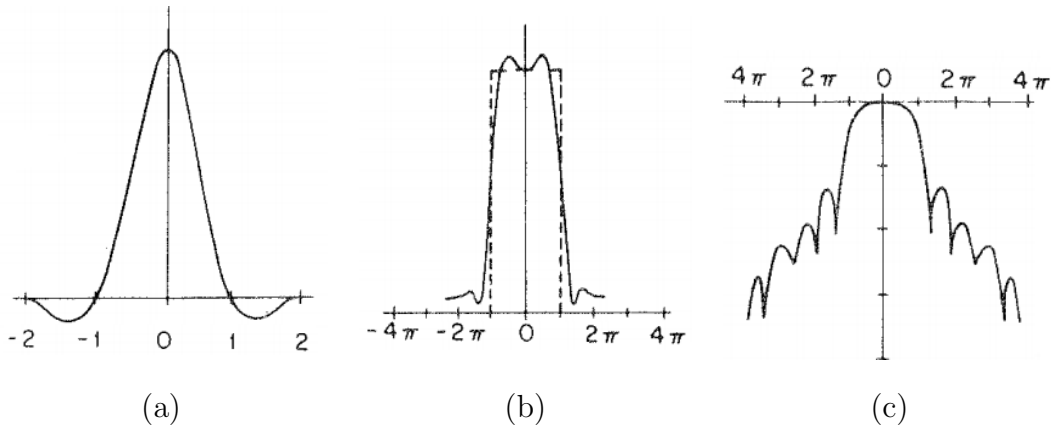


Figure 1.4: Cubic spline interpolation with  $a=-1$ . (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude [2]

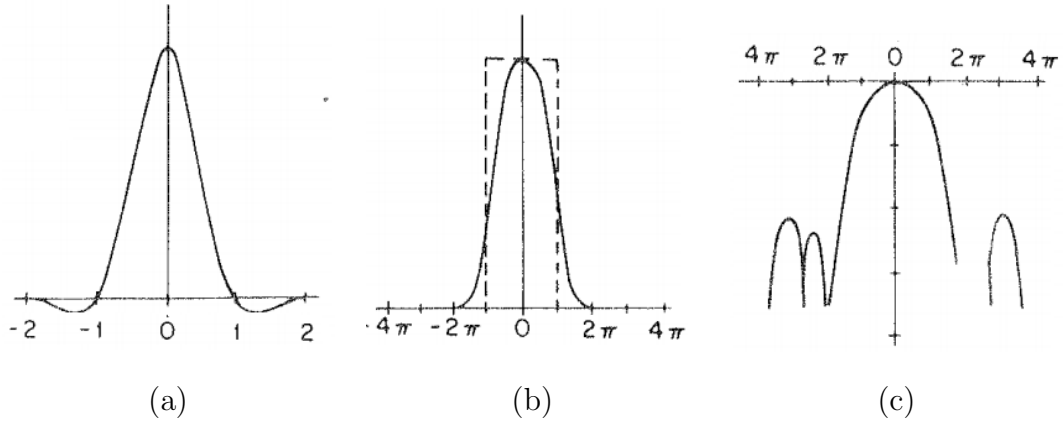


Figure 1.5: Cubic spline interpolation with  $a=-0.5$  (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude [2]

#### 1.1.4.4 B-Spline

A B-Spline or basis spline of degree  $n$  is convolution of box filter  $B^0$  [4]. Where  $B^0$  is given by

$$B^0 = \begin{cases} 1 & |x| < \frac{1}{2} \\ \frac{1}{2} & |x| = \frac{1}{2} \\ 0 & |x| > \frac{1}{2} \end{cases} \quad (1.6)$$

B-Spline of degree 1 can be found by  $B^1 = B^0 * B^0$  [3]. also second degree B-Spline can be found by  $B^2 = B^0 * B^1$ , where  $*$  represent the convolution operation.

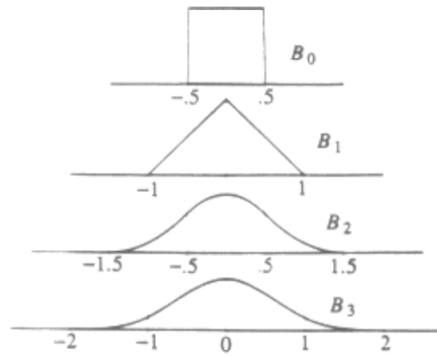


Figure 1.6: Low order B-Splines [3]

The third order B-Spline [3] can be represented as

$$h(x) = \frac{1}{6} \begin{cases} 3|x|^3 - 6|x|^2 + 4 & 0 \leq |x| < 1 \\ -|x|^3 + 6|x|^2 - 12|x| + 8 & 1 \leq |x| < 2 \\ 0 & 2 \leq |x| \end{cases} \quad (1.7)$$

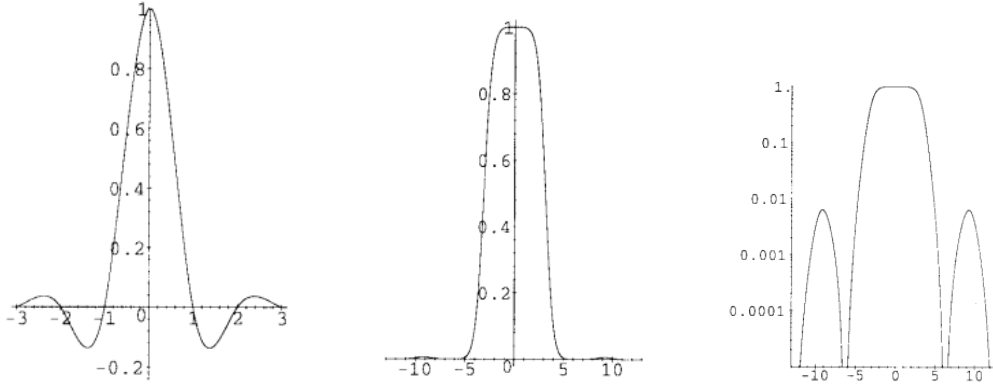


Figure 1.7: Cubic B-spline interpolation. (a) Kernel plotted for  $|x| < 3$ . (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude [5]

Kernel of B-Spline is strictly positive. This is an attractive attribute for its application in image processing. Kernels having negative lobes may generate negative values while interpolating positive data. In the context of image negative intensities are meaningless. Strictly positive interpolation kernels like B-Spline always guarantee the positive intensity values for image.

## 1.2 Literature Review On Image Interpolation Techniques

Non-adaptive interpolation techniques are based on direct manipulation on pixels instead of considering any statistical feature or content of an image. These are kernel based interpolation techniques where unknown pixel values are found by convolving with kernel. Hence they follow the same pattern of calculation for all pixels. Moreover most of them are easy to perform and have less calculation cost. Various non-adaptive techniques are nearest neighbor, bilinear, bicubic, etc.

Adaptive techniques consider image feature like intensity value, edge information, texture, etc. Non-adaptive interpolation techniques have problems of blurring edges or artifacts around edges and only store the low frequency components of original image. For better visual quality, image must have to preserve high frequency components and this task can be possible with adaptive interpolation techniques. Various adaptive techniques exists for image interpolation NEDI, DDT, ICBI, etc.

### 1.2.1 New edge-directed interpolation

Xin Li *et al* [7] proposed a new edge-directed interpolation (NEDI) that use the geometric duality between low-resolution covariance and high-resolution covariance of image which groups the pixels along the same direction. However this covariance-based interpolation have computational complexity at about two times higher than linear interpolation techniques. To reduce this complexity, trade off between visual quality and computation usually established by applying the edge directed interpolation for pixels around the edges while bilinear interpolation for non-edge pixels. As the number of pixels around the edge is less, computational burden is eased.

### 1.2.2 Improved new edge-directed interpolation

NEDI assumes the stationary local covariance, which introduces the artifacts in high-frequency regions. Also it has very high computational complexity. Nicola Asuni, Andrea Giachetti proposed an improved NEDI [8] by non-edge pixel handling, window shape, edge segmentation, value adjust and matrix conditioning,

error propagation and minimum norm solution.

In case of NEDI, if the local gray level variation is below a fixed threshold, it uses bilinear interpolation. But in case of iNEDI it uses bicubic interpolation in low frequency regions. This gives better edge direction preservation, accuracy using high value for threshold. NEDI uses square shape window which introduces the directional artifacts. iNEDI improves it by using circular window. To deal with various frequency regions iNEDI uses dynamic window size by increasing the radius of the circular window. iNEDI does the edge segmentation by excluding pixels from the circular window that are not belongs to local edge using region growing method. Value adjust of iNEDI removes the unwanted high frequencies.

### 1.2.3 Modified edge-directed interpolation

W.S. Tam, C.W. Kok and W.C Siu in, proposed Modified Edge-directed Interpolation [9] which adopted a training window structure to eliminate interpolation error propagation problem and extended the covariance matching into multiple directions to suppress the covariance mismatch problem. In the first step, it considers four training window to calculate the covariance energy and window with highest covariance energy to predict the edge direction. While in second step it consider the six training windows. So modifications can slightly improve image quality with high cost of computation.

### 1.2.4 Robust new edge-directed interpolation

Zhenhua Mai, Jeny Rajan, Marleen Verhoye, Jan Sijbers proposed Robust NEDI(R-NEDI) [10] which improves the least square nature of NEDI by non local mean(NLM). Non-local means(NLM) for denoising [11] which is able to differentiate between non-local neighborhood patterns from noiseless pixel values based on similarity of their corresponding neighborhoods. Because of least square nature of NEDI, it is not robust to the outlier. Performance of NEDI get degraded by heavy noise. In R-NEDI, first iterative reweighted least squares fitting is done which is then weighted by NLM weight function.

### 1.2.5 Edge-guided image interpolation via directional filtering and data fusion

Lei Zhang and Xiaolin Wu proposed edge-guided Image Interpolation algorithm via directional Filtering data fusion [12]. In this method each missing pixel neighborhood is divided into two sets based on two orthogonal directions. Since the missing pixel has a higher correlation with its neighbors in the edge direction, each set provides an estimate of missing pixel. These two directional estimates finally combined by linear minimum mean square error(LMMSE). The set perpendicular to edge direction give less LMMSE. The computational cost of LMMSE is eased by converting LMMSE to a optima weighting problem. This method preserve the edge direction and prevent ringing artifact.

### 1.2.6 Fast algorithm for image interpolation

Mei-Juan Chen,*et al.* [13] proposed a fast algorithm for image interpolation. In this method original image is divided into two groups:homogenous areas and edge areas. Each area is interpolated using separate algorithm, i.e. bilinear interpolation for homogenous areas or pixels, while edge pixels are interpolated using edge-oriented adaptive interpolation. In this method, pixels are categorized into the edge or homogenous part using a 3x3 window based on the difference of pixel values in four directions. If the calculated difference is less than the preset threshold value, the pixel is homogenous pixel otherwise it is a edge pixel. The non-interpolated pixel in the homogeneous area calculated using bilinear interpolation. Some of pixels left in edge pixels are interpolated using edge-adaptive interpolation.

### 1.2.7 Canny edge based image expansion

Hongjian Shi and Rabab Ward proposed Canny edge based image expansion [14] that uses Canny edge detection details to guide image expansion. In this method the original image is expanded by simple interpolation techniques like bilinear or bicubic. The interpolated image is applied to Canny edge detection to find the fine details of edges. Finally the pixel values of neighborhood of edge pixels are interpolated by linear interpolation. Pixel values around the edges looks crisper.

### 1.2.8 Edge-directed interpolation

Jan Allebach and Ping Wah Wong proposed edge directed interpolation [15] which creates high resolution edge map from low resolution image using sub-pixel edge estimation technique. Using high resolution edge map, high resolution image is created from low resolution image. This interpolation technique consist of two steps: rendering and data correction. In rendering step bilinear interpolation is modified to prevent interpolation across the edges. In correction phase high resolution image is passed through sensor model. The difference between estimated sensor data and true sensor data are iteratively reduced to modify the mesh, based on which bilinear interpolation done in rendering phase.

### 1.2.9 Data dependent triangulation

Data dependent triangulation(DDT) interpolation technique is developed to improve the visual quality of image and to reduce the computational complexity of image with respect to other linear interpolation techniques [16]. DDT is mainly used to overcome the disadvantages of bilinear interpolation technique. DDT gives better result than bilinear interpolation in term of visual appearance and has low computational complexity. DDT is better than other interpolation technique like NEDI, Edge Guided interpolation, for these following reasons:

- DDT is as simple as bilinear interpolation technique while the other techniques are complex.
- DDT can be used in arbitrary enhancement, arbitrary scaling while other techniques are defined for magnifying

In DDT first we have to find the triangles from four neighboring pixels. The diagonal pixels divide the four pixels into two triangles. Direction of diagonal is based on the edge present in the image [9]. After that we have to decide which triangle is best for our new pixels (consider one edge so that it consist two pixels). If the new pixels lie in the same triangle then apply interpolation based on only that three triangle vertices. If the new pixels are not lies in the same triangle then, we first store the edge direction of the four pixels in lookup table. Do inverse

mapping for each of new pixels and decide remaining pixels for those new pixels. Use that table in which edge direction information is stored to decide appropriate triangle and apply bilinear interpolation on that triangle to get interpolated image.

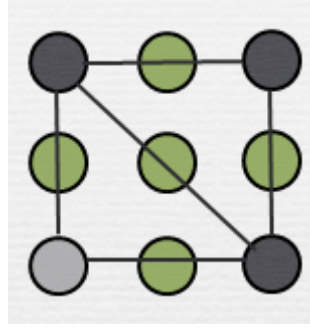


Figure 1.8: DDT Mesh [16]

### 1.2.10 Iterative curvature-based interpolation

Andrea Giachetti and Nicola Asuni has proposed real-time artifact-free image upscaling using iterative curvature-based interpolation(ICBI) [17]. ICBI technique minimizes the artifacts present in the image. ICBI proposed is based on the idea that second order derivatives can be used as the approximation of the intensity curvature profile when the gray level variation is small. ICBI technique actually consists of two techniques:Fast curvature based interpolation(FCBI) and Iterative method to minimize the energy function. FCBI calculates the local approximation of second order derivative in two diagonal direction and calculate the new pixel values along the direction of second order derivative with low value. Minimization of energy function-energy function for interpolated pixel obtained depending on its local continuity in second order derivatives. This energy function is iteratively improved till a threshold value is reached.



Step of ICBI [18]

- (i) original pixels are put at locations  $2i, 2j$  of new enlarged grid
- (ii) pixels at locations  $2i+1, 2j+1$  are calculated with the FCBI method
- (iii) iterative correction are applied until given threshold for image variation
- (iv) the remaining pixel locations are calculated by the FCBI method
- (v) iterative correction applied to the newly added pixels
- (vi) repeat the whole procedure on the new image for further enlargements

### 1.2.11 Single image interpolation technique

A new single image interpolation technique for super resolution method [18] proposed by Aftab, Hassan, A. Bin Mansoor, and Muhammad Asim which uses a hybrid method of covariance method and iterative curvature based approach. First edges and smooth areas in the image are determined. Edges are interpolated using covariance based method and smooth areas interpolated by curvature based method.

### 1.2.12 Real-time image resizing

Xiao et al. proposed an adaptive interpolation scheme based on Newton polynomial [19]. This method based on the idea that relativity among adjacent pixel exploited by the second order difference. Based on the relative position of target pixel from source pixels, two groups are formed. By comparing the value of their second order differences, one group is selected. The target pixel interpolated by second order Newton interpolation function considering the pixels of the group selected.

### 1.2.13 Curvature Interpolation Method for Image Zooming

Kim, Hakran *et al.* proposed a new curvature interpolation method (CIM) [20] which uses gradient-weighted curvature calculated from the LR image. This curvature

acts as the basis for constructing HR image. Steps of this method are

1. calculation of the curvature from a given LR image
2. interpolation of the curvature
3. construction of the zoomed image

#### **1.2.14 An edge-directed bicubic interpolation algorithm**

Dengwen, Zhou proposed a edge-directed bicubic interpolation algorithm [21], which extends the bicubic interpolation. In bicubic interpolation pixels are interpolated horizontal and vertical direction causing ringing, blocking and blurring. This edge-directed bicubic interpolation algorithm determine the edge by ratio of two orthogonal directional gradients. Strong edge pixels will be interpolated by bicubic. To calculate the weak edge value, first bicubic values of orthogonal directionals are calculated. These bicubic values are combined in direct proportion to inverse gradient of orthogonal directionals.

#### **1.2.15 Soft adaptive gradient angle interpolation of gray scale image**

Zwart, Christine M., and David H. Frakes proposed a soft-adaptive gradient angle (SAGA) interpolation [22], where edge-directed interpolator is based on locally defined straight line approximations to image isophotes. In this method isophotes slope are represented by vector and this vector guide the interpolation. This method based on the assumption that interpolation along isophotes rather than across the isophotes reduce the artifacts in image.

#### **1.2.16 Efficient super resolution using edge directed unsharp masking sharpening method**

Peng, Kuo-Shiuan et al. proposed a real-time implementation in single image super resolution using edge directed unsharp masking sharpening (EDUMS) method [23]. This method combined the edge directed information and unsharp masking sharpening to reproduce the clear edge structure. The kernel of the proposed method is the edge structure detection and the fusion between the unsharp

masking results of the high resolution edge image (HR edge) and the detail image (HR detail).

The EDUMS method used edge directed sharpening to reproduce clear and smooth edge structures after upsampling the low resolution image and details were stretched by the simple unsharp masking sharpening from upscaled images.

### **1.2.17 A switching based adaptive image interpolation algorithm**

Agarwal, Nimisha et al. proposed switching based adaptive image interpolation [24] algorithm that uses different algorithms namely SAI, SPIA and Context-Based Image Interpolation Algorithm (CBIA) techniques, for both edgy and smooth type of images.

Steps of this method are:

- (i) Detect the edge concentration in a given LR image by determining its high-pass image.
- (ii) Based on edge concentration and the predetermined threshold, we classify the image as smooth or edgy type.
- (ii) In the LR image (smooth or edgy type image), the pixels among which variation is less (smooth pixels) are interpolated by CBIA method.
- (iv) The pixels where variation is large (detailed pixels) of smooth type LR image, are interpolated by SAI method, whereas for the edgy type image, such pixels are predicted by SPIA method

## 1.3 Motivation

Keeping the research directions a step forward, it has been realised that there exists enough scope for new research work in this field of image interpolation. In this thesis an adaptive interpolation scheme for better image quality at marginal computation cost is reviewed.

## 1.4 Thesis Layout

Rest of the thesis is organised as follows —

**Chapter 2: An adaptive image interpolation** In this chapter, we have studied an adaptive scheme based on Newton forward difference that exploits the relativity of adjacent pixels for interpolation.

**Chapter 3: Conclusion and Future Work** This chapter provides the quantitative comparisons of the outputs of reviewed technique with the other existing methods of interpolation using standard images.

## Chapter 2

# An Adaptive Image Interpolation Technique

In this chapter we will discuss an adaptive interpolaton technique based on Newton forward difference. Grouping of pixels around the target pixel based on forward difference, then interpolation using the difference is the basis of the reviewed method.

### 2.1 Mathematical Background

The nth order polynomial may be represented as

$$f_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \dots + a_n(x - x_0)(x - x_1) \dots (x - x_{n-1}) \quad (2.1)$$

Solving for the coefficients in first order polynomial when  $n = 1$  through two data points,

$$f_1(x_0) = a_0 + a_1(x_0 - x_0) \quad (2.2)$$

$$f_1(x_1) = a_0 + a_1(x_1 - x_0) \quad (2.3)$$

$$a_0 = f_1(x_0) \quad (2.4)$$

$$a_1 = \frac{f_1(x_1) - f_1(x_0)}{x_1 - x_0} \quad (2.5)$$

Since the  $f_1(x)$  and  $f(x)$  are equal at  $x = x_0$  and  $x = x_1$

$$a_0 = f(x_0) \quad (2.6)$$

$$a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \quad (2.7)$$

$$a_0 = f_0 \quad (2.8)$$

$$a_1 = \frac{f_1 - f_0}{x_1 - x_0} \quad (2.9)$$

if  $f(x_i)$  represented as  $f_i$ , using Newton Forward difference of 1st order  $\Delta f_0 = f_1 - f_0$  the above coefficient  $a_1$  can be written as

$$a_1 = \frac{\Delta f_0}{h} \quad (2.10)$$

where  $h = x_1 - x_0 = x_2 - x_1$  as the data points are equally spaced. Similarly we can find  $a_n$  coefficient as

$$a_n = \frac{\Delta^n f_0}{n! h^n} \quad (2.11)$$

Finally taking  $h=1$  and  $x - x_0 = t$  and Newton forward difference of  $k_{th}$  order

$$\Delta^k f_i = \Delta^{k-1} f_{i+1} - \Delta^{k-1} f_i$$

The final expression can be written as

$$N_n(x_0 + t) = \sum_{k=0}^n \frac{\Delta^k f_0}{k!} \prod_{j=0}^{k-1} (t - j)$$

where  $t = (x - x_0)$

Based on the above equation, 2nd and 4th order Newton polynomial can be written as

$$N_2 = f_0 + \Delta f_0 t + \Delta^2 f_0 t(t-1)/2!$$

$$N_4 = f_0 + \Delta f_0 t + \Delta^2 f_0 t(t-1)/2! + \Delta^3 f_0 t(t-1)(t-2)/3! + \Delta^4 f_0 t(t-1)(t-2)(t-3)/4!$$

## 2.2 Forward difference based adaptive interpolation

This reviewed method of adaptive interpolation is based on the relativity of adjacent pixels values . This method combine 2nd order and 4th order Newton forward difference to determine the unknown pixel values. Appropriateness for grouping of pixels for interpolation is determined by this forward difference.

The pixel to be interpolated takes into consideration six pixel values namely  $(\lfloor x \rfloor - 3), (\lfloor x \rfloor - 2), (\lfloor x \rfloor - 1), (\lceil x \rceil + 1), (\lceil x \rceil + 2), (\lceil x \rceil + 3)$  i.e three on each side . These pixels are grouped together as follows for second order difference two groups formed:-  $\{(\lfloor x \rfloor - 2), (\lfloor x \rfloor - 1), (\lceil x \rceil + 1)\}$  and  $\{(\lfloor x \rfloor - 1), (\lfloor x \rfloor + 1), (\lceil x \rceil + 2)\}$  .Similarly for fourth order difference two groups formed as  $\{(\lfloor x \rfloor - 3), (\lfloor x \rfloor - 2), (\lfloor x \rfloor - 1), (\lceil x \rceil + 1), (\lceil x \rceil + 2)\}$  and  $\{(\lfloor x \rfloor - 2), (\lfloor x \rfloor - 1), (\lceil x \rceil + 1), (\lceil x \rceil + 2), (\lceil x \rceil + 3)\}$ .

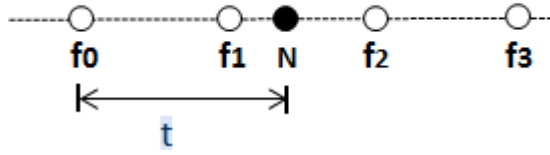


Figure 2.1: Relationship between source pixels and target pixel

Based on the min value  $\Delta^2 f_0$  ,  $\Delta^2 f_1$  ,  $\Delta^4 f_0$  ,  $\Delta^4 f_1$  , corresponding interpolation either second order or fourth order interpolation is done as:

$$N_2 = f_0 + \Delta f_0 t + \Delta^2 f_0 t(t-1)/2!$$

$$N_4 = f_0 + \Delta f_0 t + \Delta^2 f_0 t(t-1)/2! + \Delta^3 f_0 t(t-1)(t-2)/3! + \Delta^4 f_0 t(t-1)(t-2)(t-3)/4!$$

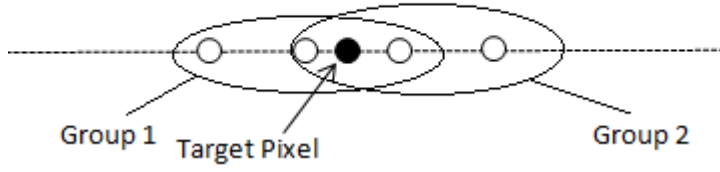


Figure 2.2: Grouping of pixels for second order interpolation

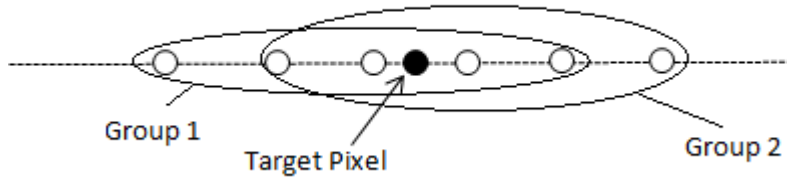


Figure 2.3: Grouping of pixels for fourth order interpolation

The steps for forward difference adaptive interpolation are :

- (i) Six pixels in the neighbourhood target pixel is chosen, three on each side
- (ii) Two groups of pixels formed based on their relative location to target pixel.
  - First group of pixels formed from the two adjacent pixels on each side of target pixel
  - Second group of pixels formed from the three adjacent pixels on each side of target pixel.
- (iii) Find the minimum of second order differences in the first group and minimum of fourth order differences for the second group.
- (iv) Calculate Newton second and fourth order interpolation value for the target pixel.
- (v) Target pixel is assigned with minimum of second order and fourth order interpolation value.



## 2.3 Results

Table 2.1: PSNR dB of  $256 \times 256$  images for resizing factor 2

Picture	PSNR (dB)					
	NN	Bilinear	Bicubic	Lanczos2	Lanczos3	Difference method
Lena	23.5478	24.9773	25.2188	25.2384	25.4325	28.0306
camera	22.1316	23.1531	23.3935	23.4128	23.6167	25.0576
bridge	21.9668	23.0021	23.2631	23.2811	23.5686	25.0004

Table 2.2: PSNR dB of  $256 \times 256$  images for resizing factor 4

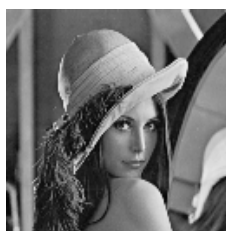
Picture	PSNR (dB)					
	NN	Bilinear	Bicubic	Lanczos2	Lanczos3	Difference method
Lena	18.2577	19.2482	19.4902	19.5031	19.9360	22.7912
camera	17.5652	18.4198	18.6664	18.6782	19.1251	20.8683
bridge	17.9290	18.7117	18.9419	18.9546	19.3755	20.9462

Table 2.3: PSNR dB of  $512 \times 512$  images for resizing factor 2

Picture	PSNR (dB)					
	NN	Bilinear	Bicubic	Lanczos2	Lanczos3	Difference method
Lena	27.8074	29.4514	29.6363	29.6471	29.7313	32.9367
Barbara	21.7021	22.1129	22.5725	22.6082	20.7464	21.5513
Mandrill	19.7576	20.4598	20.7737	20.8003	21.1649	22.0297
Boat	25.5740	26.8484	27.1053	27.1209	27.3140	29.8524
Peppers	27.3247	28.8075	28.9977	29.0182	29.1650	31.5894

Table 2.4: PSNR dB of  $512 \times 512$  images for resizing factor 4

Picture	PSNR (dB)					
	NN	Bilinear	Bicubic	Lanczos2	Lanczos3	Difference method
Lena	22.1035	23.1791	23.4343	23.4472	23.8765	27.2317
Barbara	19.3085	19.9181	20.1943	20.2171	20.7464	21.5513
Mandrill	16.8740	17.4293	17.7314	17.7493	17.4293	18.9896
boat	20.4407	21.4133	21.6487	21.6618	22.0829	24.4379
peppers	21.7165	22.8825	23.0945	23.1044	23.4542	27.1732



Downscaled Image



Nearest Neighbor



Bilinear



Bicubic



Lanczos2



Lanczos3



Difference method

Figure 2.4: Lena  $256 \times 256$  interpolated by a factor of 2



Downscaled Image



Nearest Neighbor



Bilinear



Bicubic



Lanczos2



Lanczos3



Difference method

Figure 2.5: Barbara  $512 \times 512$  Interpolated by factor of 2



Downscaled Image



Nearest Neighbor



Bilinear



Bicubic



Lanczos2



Lanczos3



Difference method

Figure 2.6: Barbara  $512 \times 512$  Interpolated by factor of 4

## **2.4 Result analysis**

From the computed PSNR values of the various interpolation algorithm, it is implied that our reviewed method of interpolation performs better in a quantitative way. However the some of the boundary pixels are sacrificed before measuring the PSNR values. Examining the PSNR values at high resizing factor greatly reduce the image quality. Of these non-adaptive interpolation method, Lanczos-3 interpolation perform the best at high computation cost. Though this reviewed algorithm performs much better, its computation cost is still high as we have to find the high order differences. Taking more numbers of pixels around the target pixel certainly increased PSNR values almost in all the variation of input images and resizing factors.

## **2.5 Summary**

In this chapter several adaptive interpolation techniques have been quantitatively compared with reviewed method using some of the standard test images. Some perform better at high computational cost while others may introduces image artifacts. Result shows the superiority of the reviewed technique.

# Chapter 3

## Conclusions and Future Work

### 3.1 Conclusion

Image interpolation is an important image processing task. Application that requires zooming, rotation, high resolution, rely heavily on interpolation techniques to perform their function. This is the obvious reason for importance of interpolation techniques in industries, research and academia. There exist a myriads of interpolation algorithms exists, from very basic to more specific interpolation techniques that customized to satisfy the particular need. With increase of device sophistication, better interpolation methods are preferred to utilize the device's built-in capability to full extent. In this thesis, we reviewed an adaptive image interpolation technique that perform better than existing classical methods with little more computational cost. However better quantitative measure of image quality compensates for the calculation of fourth order interpolation value.

### 3.2 Future work

The reviewed adaptive scheme is tested with only gray scale images. It may be extended for color images interpolation.

# Bibliography

- [1] Gonzalez, Rafael C., and Richard E. Woods. "Digital image processing." (2002).
- [2] Parker, J. Anthony, Robert V. Kenyon, and D. Troxel. "Comparison of interpolating methods for image resampling." *Medical Imaging, IEEE Transactions on* 2.1 (1983): 31-39.
- [3] Miklos, Poth. "Image interpolation techniques." 2nd Siberian-Hungarian Joint Symposium On Intelligent Systems. 2004.
- [4] Thvenaz, Philippe, Thierry Blu, and Michael Unser. "Image interpolation and resampling." (2000): 393-420.
- [5] Lehmann, Thomas Martin, Claudia Gonner, and Klaus Spitzer. "Survey: Interpolation methods in medical image processing." *Medical Imaging, IEEE Transactions on* 18.11 (1999): 1049-1075.
- [6] Keys, Robert. "Cubic convolution interpolation for digital image processing." *Acoustics, Speech and Signal Processing, IEEE Transactions on* 29.6 (1981): 1153-1160.
- [7] Li, Xin, and Michael T. Orchard. "New edge-directed interpolation." *Image Processing, IEEE Transactions on* 10.10 (2001): 1521-1527.
- [8] Asuni, Nicola, and Andrea Giachetti. "Accuracy Improvements and Artifacts Removal in Edge Based Image Interpolation." *VISAPP* (1). 2008.
- [9] Tam, Wing-Shan, Chi-Wah Kok, and Wan-Chi Siu. "Modified edge-directed interpolation for images." *Journal of Electronic imaging* 19.1 (2010): 013011-013011.
- [10] Mai, Zhenhua, et al. "Robust edge-directed interpolation of magnetic resonance images." *Physics in medicine and biology* 56.22 (2011): 7287.
- [11] Buades, Antoni, Bartomeu Coll, and J-M. Morel. "A non-local algorithm for image denoising." *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 2. IEEE, 2005.
- [12] Zhang, D., and Xiaolin Wu. "An edge-guided image interpolation algorithm via directional filtering and data fusion." *Image Processing, IEEE Transactions on* 15.8 (2006): 2226-2238.



- [13] Chen, Mei-Juan, Chin-Hui Huang, and Wen-Li Lee. "A fast edge-oriented algorithm for image interpolation." *Image and Vision Computing* 23.9 (2005): 791-798.
- [14] Shi, Hongjian, and Rabab Ward. "Canny edge based image expansion." *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*. Vol. 1. IEEE, 2002.
- [15] Allebach, Jan, and Ping Wah Wong. "Edge-directed interpolation." *Image Processing, 1996. Proceedings., International Conference on*. Vol. 3. IEEE, 1996.
- [16] Nickolaus Mueller, Yue Lu, and Minh N. Do. "Edge-Directed Image Interpolation"
- [17] Giachetti, Andrea, and Nicola Asuni. "Real-time artifact-free image upscaling." *Image Processing, IEEE Transactions on* 20.10 (2011): 2760-2768.
- [18] Aftab, Hassan, A. Bin Mansoor, and Muhammad Asim. "A new single image interpolation technique for super resolution." *Multitopic Conference, 2008. INMIC 2008. IEEE International. IEEE, 2008*.
- [19] Xiao, Jianping, et al. "Adaptive interpolation algorithm for real-time image resizing." *Innovative Computing, Information and Control, 2006. ICICIC'06. First International Conference on*. Vol. 2. IEEE, 2006.
- [20] Kim, Hakran, Youngjoon Cha, and Seongjai Kim. "Curvature interpolation method for image zooming." *Image Processing, IEEE Transactions on* 20.7 (2011): 1895-1903.
- [21] Dengwen, Zhou. "An edge-directed bicubic interpolation algorithm." *Image and Signal Processing (CISP), 2010 3rd International Congress on*. Vol. 3. IEEE, 2010.
- [22] Zwart, Christine M., and David H. Frakes. "Soft adaptive gradient angle interpolation of grayscale images." *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012.
- [23] Peng, Kuo-Shiuan, et al. "Efficient Super Resolution Using Edge Directed Unsharp Masking Sharpening Method." *Multimedia (ISM), 2013 IEEE International Symposium on*. IEEE, 2013.
- [24] Agarwal, Nimisha, et al. "A Switching Based Adaptive Image Interpolation Algorithm." 2012 IEEE