

# Interpreting Low Resolution CT Scan Images Using Interpolation Functions

Tarun Gulati

Department of Electronics & Communication  
Engineering  
M.M. University, Mullana, India

Maninder Pal

Department of Electronics & Communication  
Engineering  
M.M. University, Mullana, India

## ABSTRACT

This paper focuses on interpreting low resolution computed tomography (CT) scan medical images using interpolation functions. Image processing operations such as zooming and segmentation are very commonly performed on these images in medical sciences. However, it is very challenging to perform such operations because of poor resolution of these images. Over the last several years; significant improvements have been made in this area; however, it is still very challenging. In particular, zooming of such images is very complicated. For zooming, the process of re-sampling is normally employed. Therefore, this paper focuses on investigating the effect of interpolation functions on zooming low resolution images. For this purpose, ideally, an ideal low-pass filter is preferred; however, the same is difficult to realize in practice. Therefore, four interpolation functions (nearest neighbor, linear, cubic B-spline and high-resolution cubic spline with edge enhancement ( $-2 \leq a \leq 0$ )) are investigated in this paper for the low resolution medical CT scan images. From the results, it is found that cubic B-spline and high-resolution cubic spline have a better frequency response than nearest neighbor and linear interpolation functions. When these functions are applied for the purpose of zooming digital images, the best response was obtained with the high-resolution cubic spline functions; however, at the expense of increase in computation time.

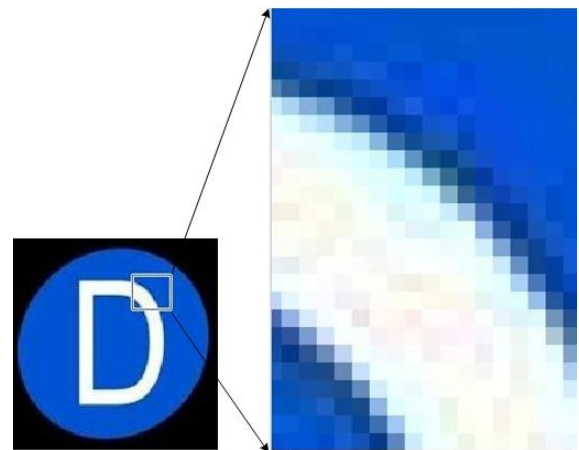
## Keywords

Pixel, Quantization, Sampling, Zooming and Interpolation.

## 1. INTRODUCTION

This paper focuses on zooming low resolution computed tomography (CT) scan digital images. The captured images are usually processed by digital image processors which render an image in a two-dimensional grid of pixels; characterized by a discrete horizontal and vertical quantization resolution. This finite resolution, especially for low resolution images, often results in visual artifacts, known as “aliasing” artifacts. These are very common in low resolution images and usually these aliasing artifacts either appear as zigzag edges called jaggies or produce blurring effects. Another type of aliasing artifacts is variation of color of pixels over a small number of pixels (termed pixel region). This type of aliasing artifacts produces noisy or flickering shading. A typical example of these artifacts is shown in Fig 1. These artifacts can be reduced by increasing the resolution of an image. This can be done using image interpolation, which is generally referred as a process of estimating a set of unknown pixels from a set of known pixels in an image. Image interpolation is in use for last many years, with the key interpolation techniques include linear, nearest neighbour & B-spline interpolation techniques. The typical applications of interpolation techniques include image resolution enhancement [1,2], image demosaicing [3-5] and unwrapping

Omni-images [4]. The quality of processed (zooming in this paper) images produced by interpolation algorithms depends upon the kinds of distortion and levels of degradation imposed on the interpolated image as well as the prior knowledge of the original image. The most commonly used interpolation techniques is the cubic B-spline interpolation. Analytically, it involves a cubic function, with the solution of the same depends upon the values of the co-efficient of the terms involved. This paper, therefore, is divided into two parts. The first part presents the analytical model of various interpolation functions. The second part investigates the effect of coefficients of these functions on the quality (PSNR) of the zoomed images using these functions.



**Fig 1: Typical example showing the effects of aliasing on the sharp corners, when zooming operation is performed.**

## 2. INTRODUCING INTERPOLATION

By definition, the term interpolation arises from the concept of re-sampling [6-7]. The term re-sampling refers to the process of transforming a discrete image that is defined at one set of coordinate locations to a new set of coordinate points. In particular, the process of interpolation refers to finding the information for undefined pixels or missed pixels in an image based on the information provided by given pixels, so that the interpolated image is as close to the actual one as possible. The given information usually includes information related to coordinates, color, gray level or density; with the image having any dimensions. The key interpolation methods include *sinc*, nearest neighbor, linear, cubic B-spline and high resolution cubic spline. A review of these techniques is presented below in Table 1.

**Table 1. Description of various interpolation functions**

INTERPOLATING FUNCTIONS	DESCRIPTION
<b>Ideal</b>	
$h_{sinc}(x) = \frac{\sin(\pi x)}{\pi x} = sinc(x)$	<p>The ideal interpolating function has constant one value in the pass band and zero value in stop band in frequency domain. A closely ideal interpolating function is <i>sinc</i> function, which has infinite length in space domain and is sinusoidal in behavior. This feature makes the interpolation usually not practical; as it is difficult to convolve signal with such infinite function. Intuitively, one solution is to truncate the <i>sinc</i> function to a shorter length. However, truncating the <i>sinc</i> function in space domain will make the frequency domain representation no longer a perfect rectangle. The response in pass band will not be flat like before and there will be some irregular pattern over the stop band, as can be seen in Fig 2.</p>
<b>Nearest Neighbor</b>	
$h(x) = \begin{cases} 1 &  x  < 0.5 \\ 0 & elsewhere \end{cases}$	<p>The nearest-neighbor interpolation function has a rectangular shape in space domain as shown in Fig 2. The nearest-neighbor method is usually reported as the most efficient from the computation point of view; but, at the cost of poor quality as can be observed from its frequency domain. It is because the Fourier Transform of a square pulse is equivalent to a <i>sinc</i> function; with its gain in pass band falls off quickly. In addition, it has prominent side lobes as illustrated in the logarithmical scale. These side-lobes usually results in blurring and aliasing effects in the interpolated image. When applying nearest-neighbor algorithm for image interpolation, the value of the new pixel is made the same as that of the closest existing pixel.</p>
<b>Linear</b>	
$h(x) = \begin{cases} 1 -  x  &  x  < 1 \\ 0 & elsewhere \end{cases}$	<p>The linear interpolation function has a triangle shape in space domain as shown in Fig 2. Comparing to the nearest-neighbor function, this function is closer to the ideal square shape function so that more energy can be passed through. The side lobes in the stop band are also much smaller, though still considerable. Therefore, the performance of linear interpolation is reported better than the nearest-neighbor interpolation. However in frequency domain, this method is still attenuating the high frequency components and is aliasing data around the cutoff frequencies. This interpolation method has been reported to work better for image reduction, rather than image enlargement.</p>
<b>B-splines</b>	
$\beta_k^n(x) = \sum_{k=0}^{n+1} \frac{(-1)^k (n+1)!}{(n+1-k)!k!} \left(\frac{n+1}{2} + x - k\right)^n$ <p><math>\forall x \in R, \forall n \in N_+</math>, And, <math>(X)_+^n = (\max(0, x))^n \quad n &gt; 0</math></p>	<p>The image interpolated by nearest-neighbor method usually exhibits jaggedness while linear interpolator causes blurring Basis spline (B-spline) interpolations have, therefore, been introduced. The concept of splines and their mathematical representations were first described by Schoenberg in 1946 [6]. By definition, splines can be referred as piecewise polynomials with pieces that are smoothly connected together. B-splines are one of the most commonly used family of spline functions. These can be derived by several self-convolutions of a so called basis function and are shown in Fig 3.</p>
<b>High resolution cubic splines</b>	
$\beta^3(x) = \begin{cases} a_{30}x^3 + a_{20}x^2 + a_{10}x + a_{00} & x_1 \leq x < x_2 \\ a_{31}x^3 + a_{21}x^2 + a_{11}x + a_{01} & x_2 \leq x < x_3 \end{cases}$	<p>From Fig 2 and Fig 3, the cubic spline has better response in both pass band and stop band comparing with the nearest-neighbor and linear functions. However, the function is positive over the whole interval in the space domain which will smooth more than is necessary below the cut-off frequency. Therefore, the cubic B-spline function needs to be modified to have negative values in the space domain. This is called high resolution cubic spline function [8] which is symmetrical about zero. The constant <math>a</math> will be negative, in order to have the function positive in the interval <math>(x_1, x_2)</math> and negative in the interval <math>(x_2, x_3)</math>. These high resolution cubic functions for different values of constant <math>a</math> are shown in Fig 4. It can be observed from Fig 4 that the same provides a better high-frequency performance than the cubic B-spline described above. The value of <math>a</math> is taken over the interval <math>(-2, 0)</math>. When the value of <math>a</math> is increased from -2 to 0, then the frequency response matches more closely to the ideal rectangular function in the pass band and the transition between the pass band and stop band gets more sharper. In addition, the amplitude of the side band is also decreased.</p>

The B-spline function for  $n=0, 1, 2, 3$  are mentioned in Table 2 and their corresponding functions in time and frequency domain are shown in Fig 2 to 7.

**Table 2. B-Spline Functions**

n=0	n=1
$\beta^0(x) = \begin{cases} 1 &  x  < 0.5 \\ 0.5 &  x  = 0.5 \\ 0 &  x  > 0.5 \end{cases}$	$\beta^1(x) = \begin{cases} 1 -  x  & 0 \leq  x  < 1 \\ 0 & \text{elsewhere} \end{cases}$
n=2	n=3
$\beta^2(x) = \begin{cases} -2 x ^2 + 1 &  x  \leq 0.5 \\  x ^2 - \frac{5}{2} x  + \frac{3}{2} & 0.5 <  x  \leq 1.5 \\ 0 & \text{elsewhere} \end{cases}$	$\beta^3(x) = \begin{cases} \frac{2}{3} - \frac{1}{2} x ^2(2 -  x ) & 0 \leq  x  < 1 \\ \frac{1}{6}(2 -  x )^3 & 1 \leq  x  < 2 \\ 0 & 2 \leq  x  \end{cases}$

### 3. Two dimensional interpolation for processing images

For image processing, the one-dimensional interpolation function mentioned in Table 1 need to be transformed into two-dimensional function. The general approach is to define a separable interpolation function as the product of two one-dimensional functions [15]. To transform into multidimensional interpolation (say in two dimensions), firstly one variable is considered as fixed and then interpolation is performed along its axis (say y-axis, corresponding to y variable). This variable is then considered as fixed and then interpolation is calculated for other variable and axis, as shown in Fig 5. Finally, all the obtained functions are then combined to have:

$$f(x_i, y_i) = \frac{x_i - x_2}{x_1 - x_2} \frac{y_i - y_2}{y_1 - y_2} f(x_1, y_1) + \frac{x_i - x_1}{x_2 - x_1} \frac{y_i - y_2}{y_1 - y_2} f(x_2, y_1) + \frac{x_i - x_2}{x_1 - x_2} \frac{y_i - y_1}{y_2 - y_1} f(x_1, y_2) + \frac{x_i - x_1}{x_2 - x_1} \frac{y_i - y_1}{y_2 - y_1} f(x_2, y_2) \quad (1)$$

$$f(x, y) = \frac{1}{6T} \{ f'_{1-1}(x)[(3+y)^3 - 4(2+y)^3 + 6(1+y)^3 - 4y^3] + f'_1(x)[(2+y)^3 - 4(1+y)^3 + 6y^3] + f'_{1+1}(x)[(1+y)^3 - 4y^3] + f'_{1+2}(x)y^3 \} \quad (2)$$

Where,  $0 \leq x \leq 1$ ;  $0 \leq y \leq 1$ , T = sampling duration,

$$\tilde{f}_j(x) = \frac{1}{6T} \sum_l^3 b_{lj} * x^{3-l}, \quad j = l-1, l+1, l+2, \quad (3)$$

And

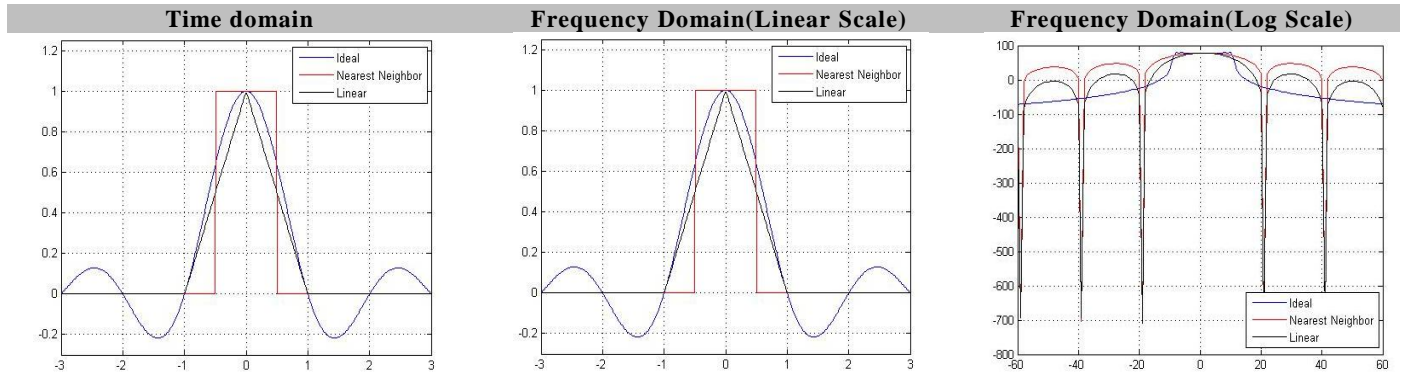
$$b_j = U * c_j; \text{ and } U = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \quad (4)$$

Where

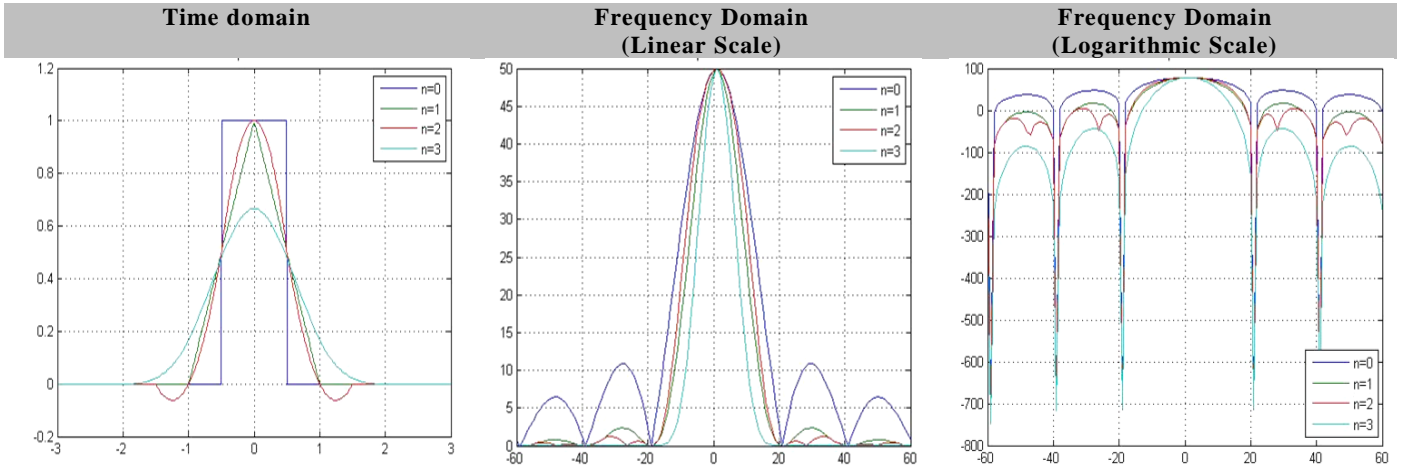
$$b_j = [b_{0j}, b_{1j}, b_{2j}, b_{3j}]^T \text{ and } c_j = [c_{k-1,j}, c_{k,j}, c_{k+1,j}, c_{k+2,j}]^T$$

This 2D interpolation function is termed as bilinear Lagrange form equations[12-14]. Using this method, the cubic interpolation (Table I) in two dimensions [9-11] can be written as per Eq. 2, 3 & 4.

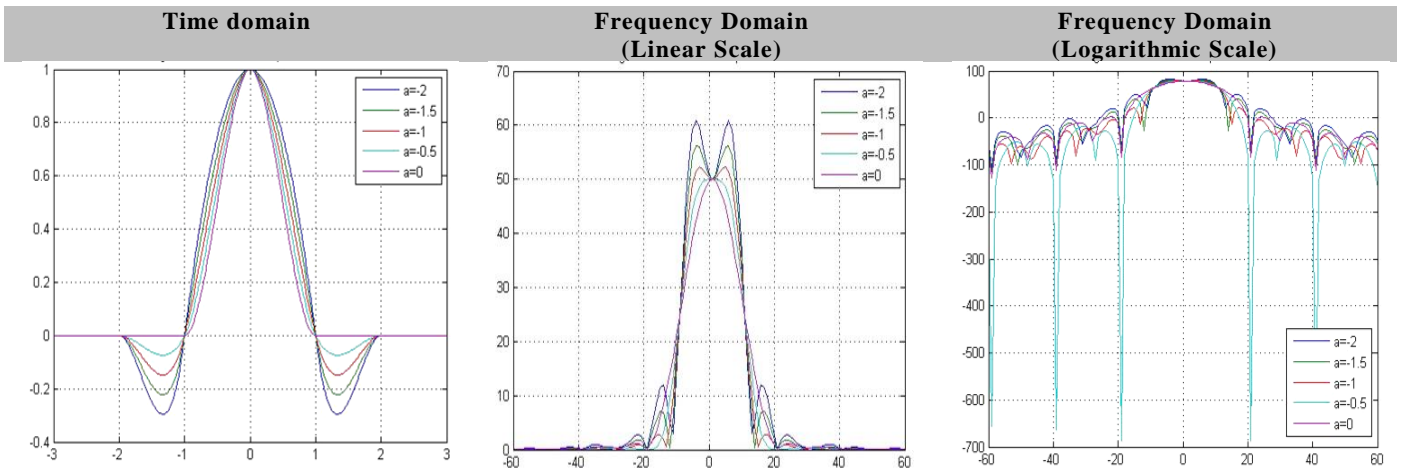
The values of b & c matrix [15-16] can be computed by substituting  $x=0$  and  $y=0$  in equation (2) above. The time and frequency domain (linear & logarithmic scale) representation of linear, nearest neighbor, bilinear, Cubic B-spline (n=0,1,2,3) interpolation function and high resolution cubic B-spline functions ( $a=-2, -1.5, -1, -0.5$  & 0) in two dimensions are shown in Fig 6 and Fig 7. These functions are then applied to CT scan image. The results obtained are shown in Fig 8 and Fig 9. A discussion on the results obtained is presented in next section.



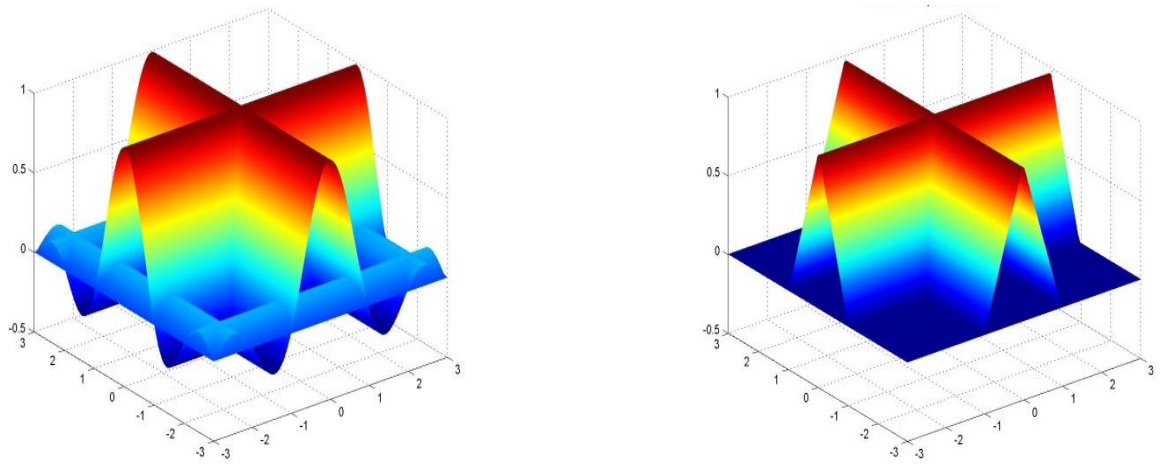
**Fig 2: The comparison of ideal, nearest neighbor and linear interpolation functions.**



**Fig 3: The time and frequency domain cubic B-spline interpolation techniques for  $n=0, 1, 2$  and  $3$ .**

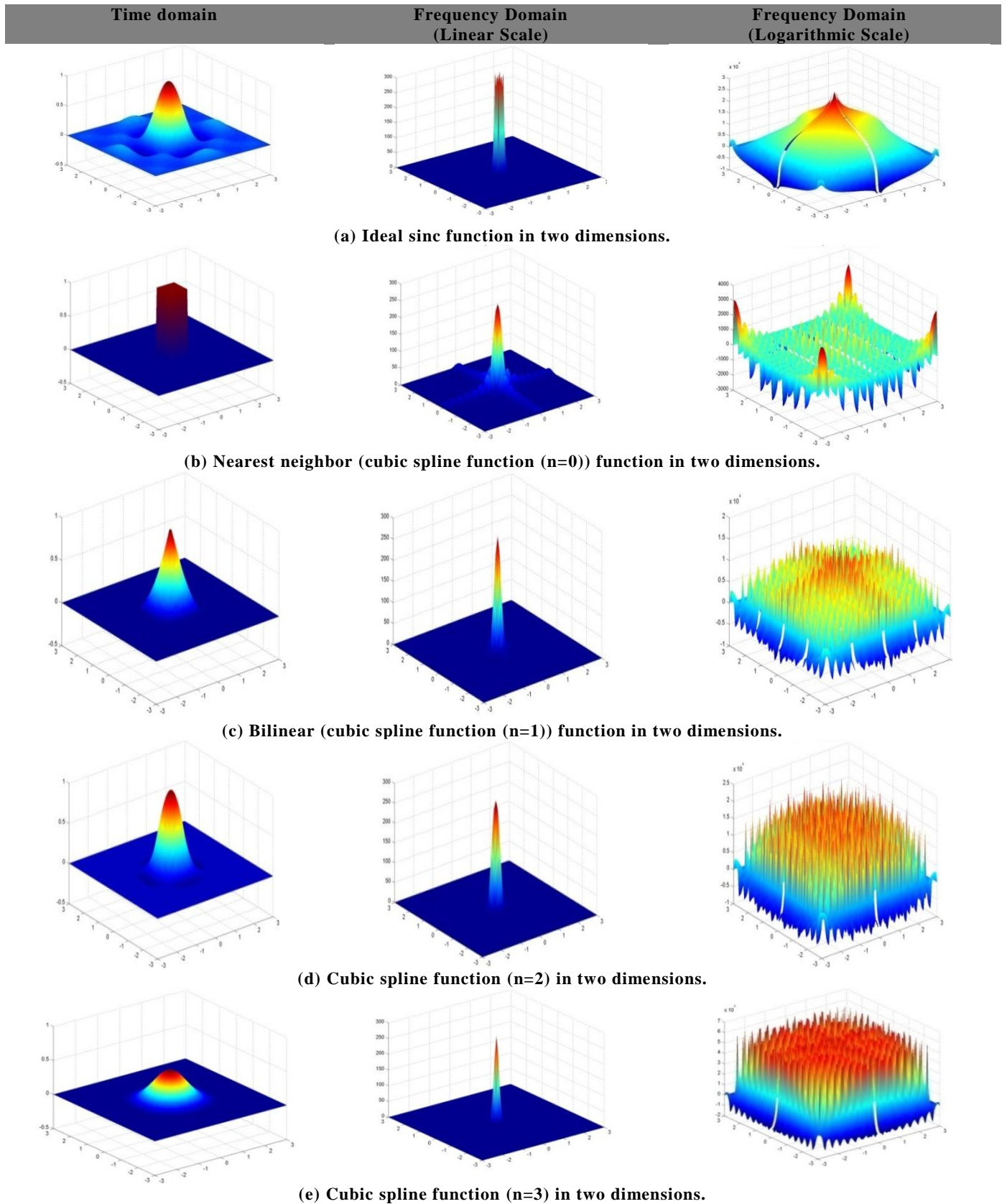


**Fig 4: The response of high resolution cubic B-spline interpolation function for  $a$  between  $-2$  &  $0$ .**

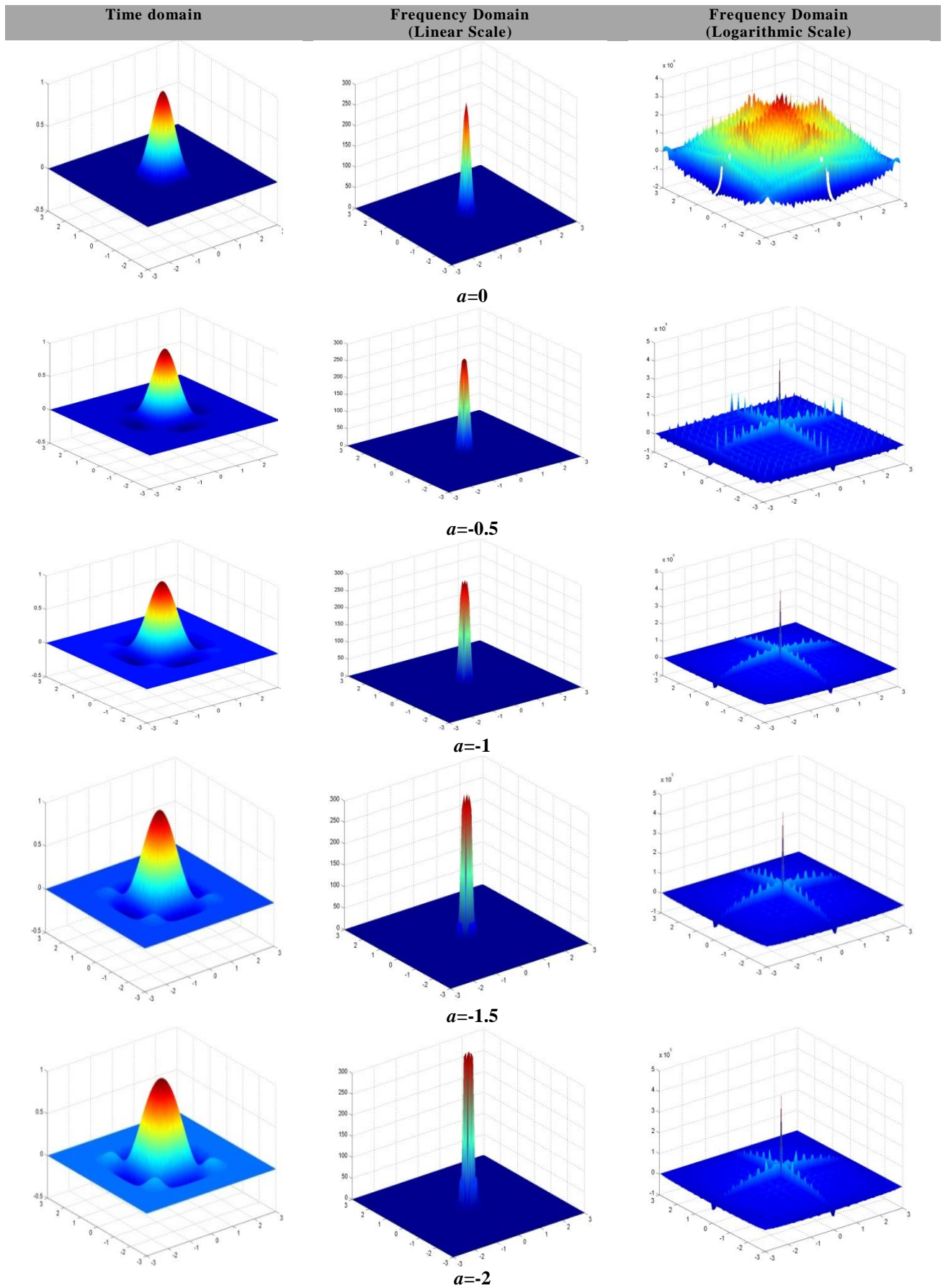


**Fig 5: An (a) ideal & (b) linear 1D dimensional function for two variables in  $x$ -axis and  $y$ -axis.**

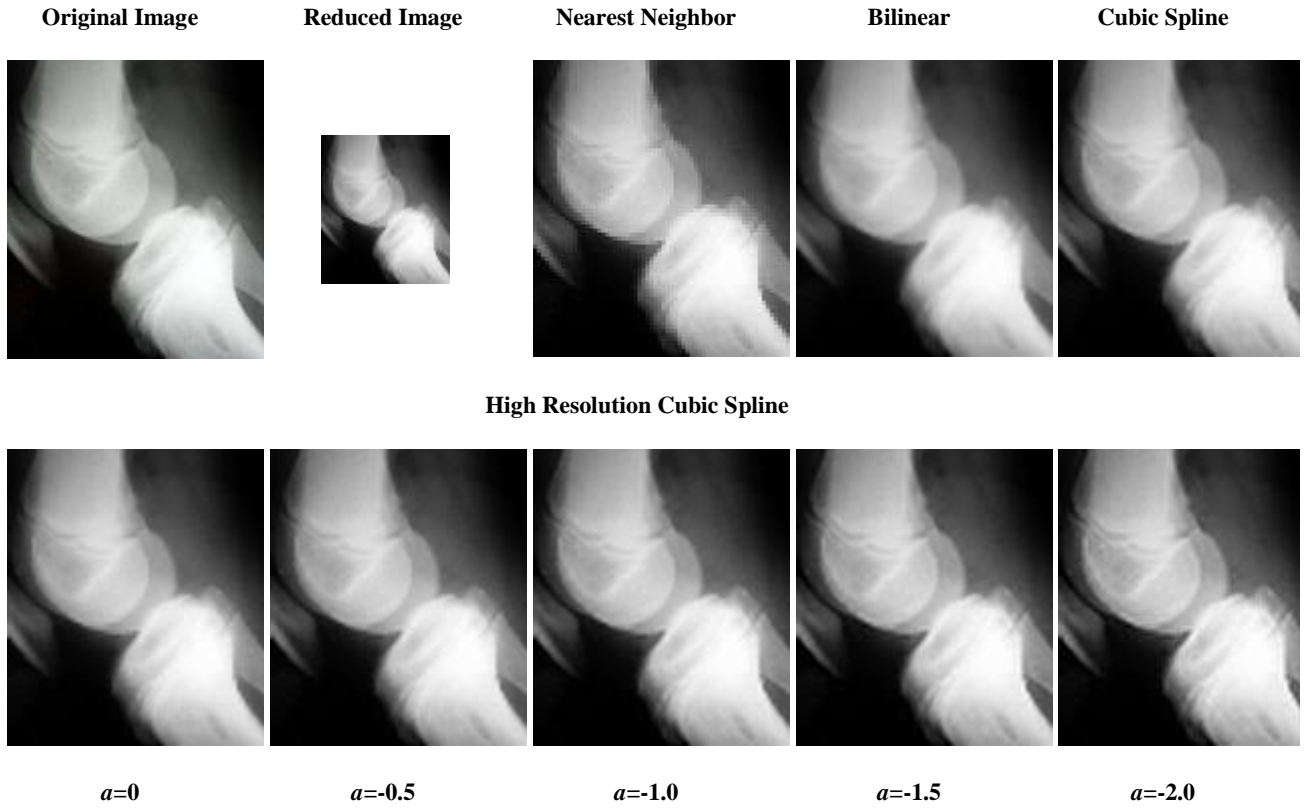




**Fig 6: The time domain and frequency domain (linear & logarithmic scale) response of (a) Ideal, (b) Nearest neighbor and Cubic spline function for n=0, (c) Bilinear and Cubic spline function for n=1, (d) Cubic spline function for n=2, and (e) Cubic spline function for n=3 in two dimensions.**



**Fig 7: The time domain and frequency domain (linear & logarithmic scale) response of high resolution cubic spline interpolation function for  $a=0,-0.5,-1,-1.5$  and  $-2$ .**



**Fig 8: A CT scan image of knee is reduced to 50% of its original size and then zoomed back to its original size using nearest neighbor, linear, cubic and high resolution cubic spline with  $a=0$ ,  $-0.5$ ,  $-1$ ,  $-1.5$  and  $-2$ .**

**Table 3. PSNR of images zoomed using Nearest Neighbor, Bilinear, Bicubic and High resolution cubic spline 2 dimensional interpolation function**

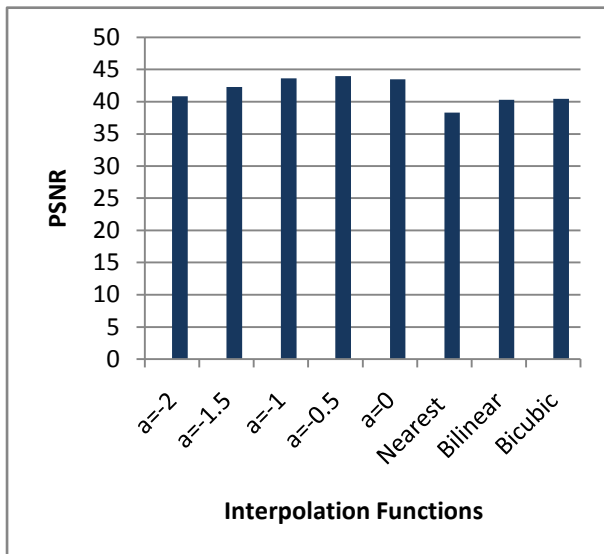
$a=-2$	$a=-1.5$	$a=-1$	$a=-0.5$	$a=0$
40.8679	42.3022	43.638	43.9643	43.471
Nearest Neighbor		Bilinear		Bicubic
38.3138		40.2942		40.4279

#### 4. RESULTS AND DISCUSSIONS

In order to investigate the characteristics of the nearest neighbor, bilinear, cubic spline and high resolution cubic spline functions with  $a$  between  $-2$  to  $0$  (inclusive), the same are implemented in Matlab. Their response in time domain and frequency domain in one dimensions and two dimensions are computed and shown in Fig 2 to 7. In frequency domain, the response of these functions is evaluated on both linear and logarithmic scale. The linear scale is used to show the pass band performance; whereas, the logarithmic scale shows the stop band performance. From Fig 2 to 7, it is found that with respect to ideal interpolation function, the high resolution cubic spline functions have shown the best response in the pass band. It is observed that for the parameter  $a=-0.5$ , the response is flat at the intermediate frequencies. However, when the value of  $a$  is further decreased to  $-2$ , then the cut off frequency is traded for a more rapid transition between the pass band and stop band. In comparison to ideal and linear functions, the nearest neighbor function has a better response in the pass band; however, it offers more attenuation even at very low frequencies. Both the bilinear and cubic B-spline

interpolation function have comparatively poor response in the pass band. However, the high resolution cubic splines have good response in the stop band similar to the cubic B-spline. The linear and nearest neighbor interpolating functions have poor stop band performances; which means that resampling after interpolation with either of these two functions will result in a large amount of aliasing. Fig. 7 shows the results of resampling (zooming) an image with nearest neighbor, bilinear, cubic B-spline and high resolution cubic spline function with  $a=-2$ ,  $-1.5$ ,  $-1$ ,  $-0.5$  and  $0$ . To investigate the effects of interpolation functions, a CT scan low resolution images of knee (128x149 pixels) is taken. The image of knee is re-sized to 50%. This reduced size image is then zoomed equal to its original size using nearest neighbor, bilinear, cubic spline and high resolution cubic spline methods discussed above. The peak signal-to-noise of these images is computed with respect to the original image and is tabulated in Table 3. From the results, it is found that the high resolution cubic spline offers the best performance at  $a=-0.5$ , which confirms the theoretical results as well. The same results are obtained for PSNR in Table 3. So, it is concluded that for zooming low resolution CT scan images it is advisable to use cubic high resolution interpolation with  $a=-0.5$ , as it has wider response in the pass band and images interpolated using this function results better in visual appearance than others.





**Fig 9: PSNR comparison of image of Knee zoomed using nearest neighbor, bilinear, cubic and high resolution cubic spline 2 dimensional functions.**

## 5. CONCLUSION

From the analytical functions and their graphical representations in time and frequency domain (linear and logarithmic scale), the nearest neighbor function does moderately well in the pass band; however, has comparatively higher side lobes in the stop band. The bilinear interpolation function performs comparatively better in the stop band; however, at the expense of considerable amount of smoothing in the pass band. The cubic B-spline has better stop band performance than nearest neighbor and linear; however, it has wider pass band. The high resolution cubic splines have the best combination of both pass band and stop band performance. However, this is observed for  $a=-0.5$ . In situations, when  $a$  is further decreased towards  $-2$ , then the function has a significant degradation in its response in both pass band and stop band. These functions are applied on a CT scan low resolution image with a zooming factor of 2. It is observed that the visual results of high-resolution cubic spline interpolation with  $a=-0.5$  are better as compared to nearest neighbor, bilinear and cubic spline functions. The PSNR corresponding to this function is also more than others as shown in Fig 9. However, it is computationally expensive. In addition, it is also observed that the nearest neighbor algorithm has the shortest extent in the image domain, one interpixel distance. The linear interpolating algorithm, extends over two interpixel distances; whereas, the high resolution cubic splines extends over four interpixel distances. In concern to zooming operation, the performance of the discussed interpolation functions can be arranged in a sequence as high resolution>cubic spline>bilinear>nearest neighbor.

## 6. REFERENCES

[1] M. Hanumantharaju, M. Ravishankar, D. R. Rameshbabu & S. Ramachandran, "Color Image Enhancement using Multiscale Retinex with Modified Color Restoration Technique," Second International Conference on Emerging Applications of Information Technology (ICEAIT), pp. 93-97, 2011.

[2] Z. Min, W. Jiechao, L. Zhiwei & L. Yonghua, "An Adaptive Image Zooming Method with Edge Enhancement," 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), pp. 608-611, 2010.

[3] Wing-Shan Tam, C. W. Kok & W. C. Siu, "A Modified Edge Directed Interpolation For Images," 17th European Signal Processing Conference (ESPC), 2009.

[4] W. Bender & C. Rosenberg, "Image Enhancement Using Non-uniform Sampling," Proc. SPIE-int. Soc. Opt. Eng., Volume 1460, pp. 59-70, 1991.

[5] P. Thvenaz, T. Blu & M. Unser, "Image Interpolation and Resampling," Handbook of Medical Imaging, Processing and Analysis, I.N. Bankman (eds.), Academic Press, San Diego CA, USA, pp. 393-420, 2000.

[6] M. Zamani, "An Applied Two-Dimensional B-spline Model for Interpolation of Data," International Journal of Advanced Research in Engineering and Technology (IJARET), Volume 3(2), July-December 2012.

[7] R. G. Keys, "Cubic convolution interpolation for digital image processing," IEEE Trans. Acoust., Speech, Sig. Proc., Volume 29(6), pp.1153-1160, 1981.

[8] R. W. Schafer & L. R. Rabiner, "A digital signal processing approach to interpolation," Proc. IEEE, Volume 61, pp. 692-702, 1973.

[9] T. Acharya & P. S. Tsai, "Computational Foundations of Image Interpolation Algorithms," ACM Ubiquity Volume 8, 2007.

[10] T. M. Lehmann, C. Gonner & K. Spitzer, "Interpolation Methods in Medical Image Processing," IEEE Transactions on Medical Imaging, Volume 18(11), pp. 1049-1075, 1999.

[11] J. Anthony Parker, R. V. Kenyon & D. E. Troxel, "Comparison of Interpolating Methods for Image Resampling," IEEE Transactions on Medical Imaging, Volume 2(1), pp. 31-39, 1983.

[12] E. Maelland, "On the comparison of the interpolation methods," IEEE Trans. Med. Imag., Volume 7(3), pp. 213-217, 1988.

[13] M. F. Fahmy, T. K. Abdel Hameed & G. F. Fahmy, "A Fast B-spline Based Algorithm for Image zooming and Compression," 24th National Radio Science Conference (NRSC), Egypt, 2007.

[14] H. S. Hou & H. C. Andrews, "Cubic Splines for Image Interpolation and Digital Filtering," IEEE Trans. Acoust., Speech, Signal Processing, Volume 26, pp. 508-517, 1978.

[15] S. E. Reichenbach & S. K. Park, "Two-parameter cubic convolution for image reconstruction," Proc. SPIE, Volume 1199, pp.833-840, 1989.

[16] M. Unser, "A Perfect Fit for Signal/Image Processing: Splines," Proceedings of the SPIE International Symposium on Medical Imaging: Image Processing (MI'02), 4684, Part I, pp. 225-236, San Diego CA, 2002.