RESEARCH ARTICLE OPEN ACCESS

Image Interpolation Techniques in Digital Image Processing: An Overview

Shreyas Fadnavis*
*(Pune Institute of Computer Technology, Pune University, India)

ABSTRACT
In current digital era the image interpolation techniques based on multi-resolution technique are being discovered and developed. These techniques are gaining importance due to their application in variety if field (medical, geographical, space information) where fine and minor details are important. This paper presents an overview of different interpolation techniques, (nearest neighbor, Bilinear, Bicubic, B-spline, Lanczos, Discrete wavelet transform (DWT) and Kriging). Our results show bicubic interpolations gives better results than nearest neighbor and bilinear, whereas DWT and Kriging give finer details.

Keywords – Bicubic, Bilinear, DWT, Image Interpolation, Kriging

I. INTRODUCTION

Digital image processing has gained a lot of importance in the modern times due to the advancements in graphical interfaces. Digital image processing is a subfield of digital signal processing which has made tremendous progress in varied domains, due to its vast applications. Digital image processing can be understood as the method of processing an image using computer algorithms to improve the varied aspects of any particular image. Thus the most important aspect of image processing is the ways in which we can improve the quality (what in common terms is called clarity) of an image by using various techniques. Image interpolation is one such technique. Interpolation techniques determine the values of a function at positions lying between its samples. There are several interpolation techniques that have been documented in the past. The widely used techniques are nearest neighbor, bilinear, bicubic, B-splines, lanczos2, discrete wavelet transform, Kriging ([1]; [2]; [3]).

Image processing techniques gained lot of importance as it helps in improving low resolution images of CT scan, MRI, geographical images, images received on mobile phones and from satellites, etc. It can be used to resample the image either to decrease or increase the resolution ([4]). The quality of processed image depends on adopted interpolation technique. During last decade various techniques of image processing are developed for example image restoration, filtering, compression, segmentation etc. ([5]). However image interpolation is less explored. In this paper we take into account the performance of most commonly used interpolation techniques: nearest neighbor, bilinear, bicubic, B-splines, lanczos2, discrete wavelet transform and Kriging.

II. Results and discussions on interpolation techniques

2.1 Nearest neighbor

Nearest neighbor: It is a simplest interpolation. In this method each interpolated output pixel is assigned the value of the nearest sample point in the input image. The interpolation kernel for the nearest neighbor

$$h(x) = \begin{cases} 0 & |x| > 0 \\ 1 & |x| < 0 \end{cases} \qquad (1)$$

The frequency response of the nearest neighbor kernel is

$$H(\omega) = \text{sinc}\ (\omega/2) \qquad (2)$$

Although this method is very efficient, the quality of image is very poor. It is because the Fourier Transform of a rectangular function is equivalent to a sinc function; with its gain in pass band falls off quickly. Also, it has prominent side lobes are in the logarithmical scale.

2.2 Bilinear interpolation

Bilinear interpolation is used to know values at random position from the weighted average of the four closest pixels to the specified input coordinates, and assigns that value to the output coordinates. The two linear interpolations are performed in one direction and next linear interpolation is performed in the perpendicular direction. The interpolation kernel is given as

$$u(x) = \{\ 0 \qquad |x| > 1 \qquad (3)$$
$$\{\ 1 – |x| \qquad |x| < 1$$

X is distance between two points to be interpolated

2.3 Bicubic interpolation

The bicubic interpolation is advancement over the cubic interpolation in two dimensional regular grid. The interpolated surface is smoother than corresponding surfaces obtained by above mentioned methods bilinear interpolation and nearest-neighbour interpolation. It uses polynomials, cubic, or cubic convolution algorithm. The Cubic Convolution Interpolation determines the grey level value from the weighted average of the 16 closest pixels to the specified input coordinates, and assigns that value to the output coordinates, the first four one-dimension. For Bicubic Interpolation (cubic convolution interpolation in two dimensions), the number of grid points needed to evaluate the interpolation function is 16, two grid points on either side of the point under consideration for both horizontal and perpendicular direction. The bicubic convolution interpolation kernel is:

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

Where a is generally taken as -0.5 to -0.75

2.4 Basic-splines (B-spline)

The nearest neighbor and Bilinear interpolations compromises the quality of image over efficiency due to rectangular shape in the pass band and infinite side lobes. The B-spline interpolations smoothly connects polynomials with pieces ([6]). A B-spline of degree n is derived through n convolutions of the box filter, Bsp0. Thus B-spline of degree 1 can be represented as Bsp1=Bsp0*Bsp0. The second degree B-spline B2 is produced by convolving Bsp0*Bsp1 and the cubic B-spline Bsp3 is from convolving Bsp0*Bsp2. The interpolation kernel of cubic B-spline is :

$$h(x) =1/6\begin{cases} 3|x|^3-6|x|^2+4 & 0<|x|<1 \\ -|x|^3+6|x|^2-12|x|+8 & 1\le|x|<1 \\ 0 & 2\le|x| \end{cases} \quad (5)$$

2.5 Lanczos interpolation:

Lanczos filter is used to interpolate the value of a digital signal between its samples. It maps each sample of the given signal to a translated and scaled copy of the Lanczos kernel. Lanczos kernel is a sinc function windowed by the central hump of a dilated sinc function. Lanczos resampling used to increase the sampling rate of a digital signal ([7]). Lanczos kernel is the normalized sinc function sinc(x), windowed by the Lanczos window, or sinc window (sinc(*x*/*a*) for $-a \le x \le a$. *([7]).*

$$L(x) = \begin{cases} Sinc(x)\,sinc\left(\frac{x}{a}\right) & if -a<x<a \\ 0 & otherwise \end{cases} \quad (6)$$

Equivalently,

$$L(x) = \begin{cases} 1 & if\ x=0 \\ (a\sin(\pi x)\sin(\pi x/a))/\pi 2X2 & if\ 0<|x|<a \\ 0 & otherwise \end{cases} \quad (7)$$

Where a is a positive integer (2 or 3). It determines the size of the kernel. The Lanczos kernel has 2a − 1 lobes, a positive one at the centre and a − 1 alternating negative and positive lobes on each side. For signal with samples s_i, the value S(x) interpolated at an real argument x is obtained by the discrete convolution of those samples with the Lanczos kernel ([7]).

$$S(x) = \sum_{i=\lfloor x\rfloor-1+1}^{\lfloor x\rfloor+a} S_i L(x-i) \quad (8)$$

where *a* is the filter size parameter and [x] is the floor function. The kernel is zero outside of bounds.

The original image and image interpolated using nearest neighbor, bilinear, bicubic, b-spline and Lanczos interpolation are shown in figure 1and 2.

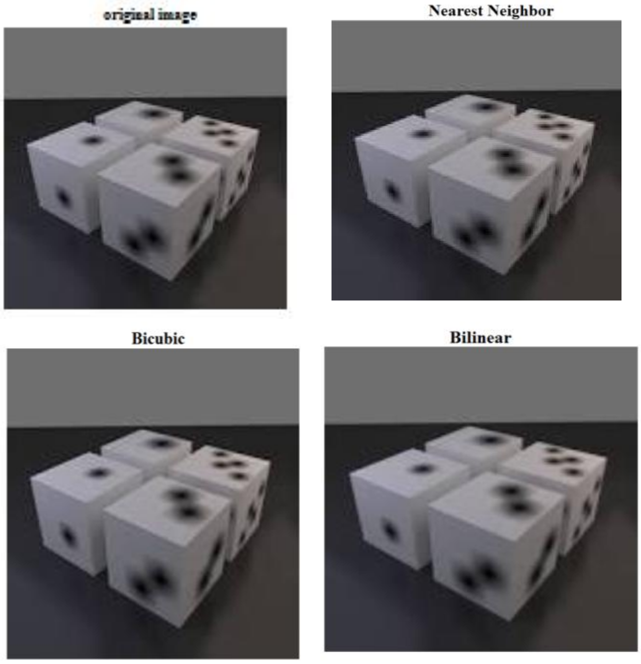

Figure1: Image after application of interpolation (a) original (b) Nearest neighbor(c) bicubic (d) bilinear

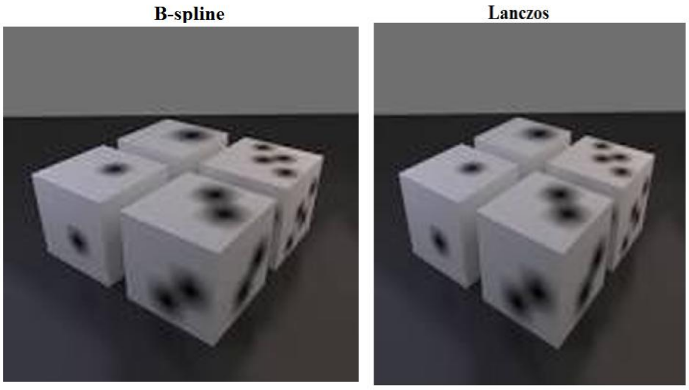

Figure2: Image after application of interpolation (a) b-spline (b) Lanczos

It can be seen that bicubic interpolations gives better results than nearest neighbor and bilinear interpolation.

2.6. DISCRETE WAVELET TRANSFORMS (DWT)

In the past few years wavelet transforms have gained more importance as compared to the traditionally used discrete Fourier transforms due its capabilities of helping determine both frequency and location information (termed as temporal resolution). One of the most important application of DWT is that it can be used for data compression be reducing the input, so that it can be used more practically. DWT uses the basic wavelet function φ (t) and scaling function ψ(t) for decomposition and reconstruction of sampling signals.

$$\psi(t) = \sum_n g(n)\sqrt{2}\,\pi(2t-n),$$
$$\varphi(t) = \sum_n h(n)\,\sqrt{2\pi}(2t-n) \qquad (9)$$

The basic wavelet function can be calculated from the scaling function g(n) and h(n) are digital filter coefficients; their relationship is expressed as

$$g(n) = (-1)^n\, h(l-n-1) \qquad (10)$$

Where g(n) and h(n) are high pass filet and low pass filter and l denotes filter length. DWT can analyze the signal layer by layer ([8]). In DWT and approximate coefficients retain low-frequency information of the original signal S(n) and less high-frequency noise ([9]). The wavelet transform decomposition process is expressed as

$$cA1 = \sum_n h(n-2k)S(n)$$
$$cD1 = \sum_n g(n-2k)S(n) \qquad (11)$$

where cD1 is a detailed coefficient. In the 2-D version of analysis case, the 1D analysis filter bank is first applied to the columns of the image and then applied to the rows. For the image size of N1 rows and N2 columns, applying the 1D analysis filter bank to each row of both of the two sub-band images, we have four sub-band images, each having N1/2 rows and N2/2 columns.

In this paper we present an example for two dimensional (2-D) discrete wavelet transform of a signal x is implemented by iterating the 2D analysis filter bank on the low pass sub-band image. In this case, at each scale there are three sub-bands instead of one. We create a random input signal x of size 128 by 64. We apply the DWT and its inverse, and show its reconstruction x from the wavelet coefficients. The three wavelets associated with the 2D wavelet transform are shown in figure 3 (details are available at http://eeweb.poly.edu/iselesni/WaveletSoftware/standard2D.html.

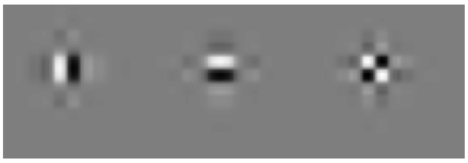

Figure 3 The three wavelets associated with the 2D wavelet transform

2.7 KRIGING METHOD OF IMAGE INTERPOLATION:

Generally during interpolation the value at the unknown location is found out by an algorithm which calculates the value of the given unknown (variable) as a weighted sum of the surrounding variables at their locations respectively. But this is not a very efficient way of predicting the unknowns the value may not be predicted properly in case the neighboring variables are placed in a few clusters which are placed far away from each other. However Kriging predicts these values in a rather more optimal and accurate way using the concept of weighted average from a data-driven weighting function in contrast to, the other image interpolation methods which use an arbitrary value for the weighting function. Kriging confer weights for each point according to its distance from the unknown value. Kriging interpolation can be carried out using the following set of equations:

$$Z(X) = m(X) + \varepsilon'(X) + \varepsilon'' \qquad (12\ )$$

In the above equation m(x) is a function which describes the structural or surface component of the image which is being investigated and ε'(x) be a function which describes the probability distribution if a random sequence obtained on the basis of autocorrelation of the unknowns in the image; ε'' is the term which indicates the random noise generated in the image, whose mean value is 0 and variance is σ^2.

$$E[Z(X)-z(x+h)]=0 \qquad (13)$$

M(x) is a function which helps in determining the trend over that particular region of the image. For example: if we take the case of a uniformly distributed image then, the difference between x and (x + h) would be 0 (h) being the distance between the two points. This also means that if the difference between the two points is less, then they will also have almost similar values.

$$E[\{ Z(x) - Z(X+h)\}^2] = E[\{\varepsilon'(x)- \varepsilon'(X+h)\}^2] = 2\,Y(h) \qquad (14)$$

Where Ƴ(h) is defined as the *semi variance*. Taking all this into account the above original equation 1 can also be expressed as:

$$Z(X) = m(x)+ □(h) + \varepsilon'' \qquad (15)$$

Here we present the Kriging interpolation on random filed with sampling locations. figure 6 (a) shows the random field with sampling locations. The kriging predictions, variogram and Kriging variance are shown in figure 4 (b)-(d) respectively.

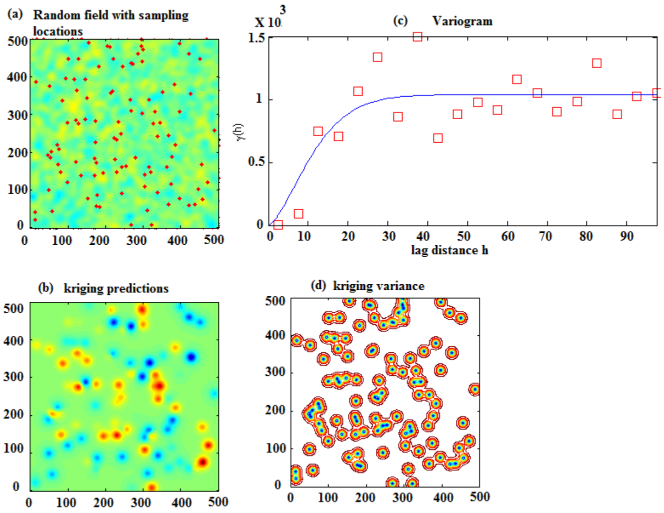

Figure 4 (a) Random field with sampling location (b) kriging predictions (c) variogram (d) kriging variance.

ACKNOWLEDGEMENTS

The author acknowledges the facilities provided by the college staff and library at Pune Institute of Computer Technology which were very helpful.

References:

[1] F. A. Jassim and F. H Altaany., Image Interpolation Using Kriging Technique for Spatial Data, Canadian Journal on Image Processing and Computer Vision Vol. 4 No. 2, 2013.

[2] R. S Asamwar. K. M. Bhurchandi and A. S Gandhi. , Interpolation of Images Using Discrete Wavelet Transform to Simulate Image Resizing as in Human Vision, International Journal of Automation and Computing, 7(1), 2010, 9-16, DOI: 10.1007/s11633-010-0009-7.

[3] R Roy., M Pal. and T Gulati., Zooming Digital Images using Interpolation Techniques, International Journal of Application or Innovation in Engineering & Management (IJAIEM), Volume 2, Issue 4, 34-45, 2013, ISSN 2319 – 4847.

[4] R. C. Gonzalez and R. E Woods.. Digital Image Processing, 2nd edition, Prentice Hall, 2002, pp. 272-274.

[5] C Khare. and K. K Nagwanshi., Image Restoration Technique with Non Linear Filter, International Journal of Advanced Science and Technology, Vol. 39, February, 2012.

[6] T. Acharya and P. S. Tsai, “Computational Foundations of Image Interpolation Algorithms,” ACM Ubiquity Volume 8, 2007.

[7] W. Burger and M. J. Burge, Digital Image Processing: An Algorithmic Introduction Using Java, (springer science, New York, NY, 1003,USA).

[8] Yu-Cheng Fan and Yi-Feng Chiang, Discrete Wavelet Transform on Color Picture Interpolation of Digital Still Camera, VLSI Design Volume 2013 (2013), Article ID 738057, 9 pages http://dx.doi.org/10.1155/2013/73805

[9] S. Peng and J. Liu, Medical image interpolation method based on similarity analysis of discrete wavelet transform , Proc. SPIE 6789, MIPPR 2007: Medical Imaging, Parallel Processing of Images, and Optimization Techniques, 678912 (November 14, 2007); doi:10.1117/12.750197