

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/294738469>

A Comparative Analysis of Image Interpolation Algorithms

Article in IJARCCCE · January 2016

DOI: 10.17148/IJARCCCE.2016.5107

CITATIONS

26

READS

6,024

2 authors, including:



Pankaj Parsania

Anand Agricultural University

9 PUBLICATIONS 46 CITATIONS

SEE PROFILE

A Comparative Analysis of Image Interpolation Algorithms

Mr. Pankaj S. Parsania¹, Dr. Paresh V. Virparia²

Research Scholar, Department of Computer Science, Sardar Patel University, Vallabh Vidyanagar, Gujarat, India¹

Director, Department of Computer Science, Sardar Patel University, Vallabh Vidyanagar, Gujarat, India²

Abstract: Image interpolation is the most basic requirement for many image processing task such as computer graphics, gaming, medical image processing, virtualization, camera surveillance and quality control. Image interpolation is a technique used in resizing images. To resize an image, every pixel in the new image must be mapped back to a location in the old image in order to calculate a value of new pixel. There are many algorithms available for determining new value of the pixel, most of which involve some form of interpolation among the nearest pixels in the old image. In this paper, we used Nearest-neighbor, Bilinear, Bicubic, Bicubic B-spline, Catmull-Rom, Mitchell-Netravali and Lanczos of order three algorithms for image interpolation. Each algorithms generates varies artifact such as aliasing, blurring and moiré. This paper presents, quality and computational time consideration of images while using these interpolation algorithms.

Keywords: Nearest-neighbor, Bilinear, Bicubic, Bicubic B-spline, Catmull-Rom, Mitchell-Netravali, Lanczos.

I. INTRODUCTION

Image interpolation is a term used for image processing, but is often used with different terminologies in literature like image scaling, image resampling and image resize. There are many algorithms currently in use for the resizing of digital images. Most of them attempt to reproduce a visually attractive replica of the original. Now a days the technology for smaller display area for viewing on varieties of device, image size is generally down sampled (or sub sampled or decreased) in order to produce thumbnails. Up sampling an image (or enlarging or interpolating) is most common for viewing on large display size monitor or television [1]. While enlarging an image it is not possible to discover any more information in the image than already exists, and image quality inevitably suffers. However, there are several methods of increasing the number of pixels that an image contains, which can be created out of original pixels. These methods are often termed as image interpolation algorithms.

There are various algorithms currently in use for image interpolation of digital images. Some of the common interpolation algorithms are the nearest neighbor, bilinear, and bicubic [2]. Other interpolation algorithms such as Catmull-Rom and the Mitchell-Netravali generates better image quality. Lanczos algorithm utilizes 2-lobed, 3-lobed or 4-lobes sinc window function to implement interpolation [3]. These algorithms uses higher order interpolators which take more surrounding pixels into consideration, and thus also require more computation time, and retain the most of image details after an interpolation. They are extremely useful when the image requires multiple translation, rotations or distortions in separate steps. However, for single-step enlargements or image operations, these higher- order algorithms provide diminishing visual improvement and processing time increases significantly.

Interpolation using Shape-Preserving Approach [4], classification and stitching [5], grid based image interpolation [6], auto-regression based method [7], fuzzy area-based scaling [8] and interpolation used with Artificial Neural Networks [9] are some of the novel techniques used for interpolating images. Although these algorithms perform well, they require a lengthy processing time due to their complexity. This is intolerable for real-time image scaling. Hence, these algorithms have not been considered for the comparative analysis in this paper.

In this paper, various interpolation algorithms are used for image interpolation with translation parameters in x and y directions. The result obtained is presented and summarized with quality and time complexity parameters.

II. INTERPOLATION ALGORITHMS

Digital image interpolation is the process of generating a continuous intensity surface from discrete image data samples. There are many different types of interpolation methods, each resulting in a different look to final image. Thus, it is best if the quality, or visible distinction for each pixel, is retained throughout the interpolation function [10].

Generally, almost every geometric transformation like translating, rotating, scaling, and warping requires interpolation to be performed on an image. Such transformations are essential to any commercial digital image processing software. The perceived quality of the interpolated images is affected by several issues such as sharpness of edges, freedom from artifacts and reconstruction of high frequency details etc. [11]

Interpolation function is performed by convolution operation which involves a large number of addition and multiplication operations. Hence, a trade-off is required

between computation complexity and quality of the interpolated image [12]. Various interpolation algorithms discussed in this paper are Nearest-neighbor, Bilinear, Bicubic, Cubic B-spline, Catmull-Rom, Mitchell-Netravali and Lanczos.

A. Nearest Neighbour

This is the simplest and requires the least processing time of all the interpolation algorithms. Nearest neighbor selects the value of the nearest pixel by rounding the coordinates of the desired interpolation point. Using this method one finds the closest corresponding pixel in the source (original) image for each pixel in the destination image [13]. New pixels are made the same as others close-by. The pixels or dots of color are duplicated to create new pixels as the image grows. It creates pixilation or edges that break up curves into steps or jagged edges. This form of interpolation suffers from normally unacceptable effects for both enlarging and reduction of images [10].

B. Bilinear Interpolation

Bilinear interpolation takes a weighted average of the four neighborhood pixels to calculate its final interpolated value. The result is much smoother image than the original image. When all known pixel distances are equal, then the interpolated value is simply their sum divided by four. This technique performs interpolation in both directions, horizontal and vertical. This technique gives better result than nearest neighbor interpolation and take less computation time compare to bicubic interpolation [14].

C. Bicubic Interpolation

Bicubic goes one step beyond bilinear by considering the closest 4x4 neighborhood of known pixels for a total of 16 pixels. Since these are at various distances from the unknown pixel, closer pixels are given a higher weighting in the calculation. Bicubic produces noticeably sharper images than the previous two methods, and is perhaps the ideal combination of processing time and output quality. For this reason it is a standard in many image editing programs including Adobe Photoshop, printer drivers and in-camera interpolation [15].

D. Cubic B-Spline

As bicubic interpolation, the cubic B-spline interpolation algorithm also interpolates from the nearest sixteen source pixels. However, this algorithm uses B-spline interpolating functions instead of cubic splines, which in general yield quite smooth results. It performs a convolution with a two-dimensional non separable filter, so its complexity is increased. In contrast, bicubic interpolation uses a convolution with a separable filter, and hence its complexity is less. Despite of this performance difference, bicubic B-spline has interesting characteristics of smoothness that make it a good option in some cases [3].

E. Catmull-Rom Interpolation

Catmull-Rom is a local interpolating spline developed for computer graphics purposes. Its initial use was in design of curves and surfaces, and has recently been used in several applications. Catmull-Rom splines are a family of cubic interpolating splines formulated such that the tangent at each point is calculated using the previous and

next point on the spline. The results are similar to ones produced by bicubic interpolation with regards to sharpness, But the Catmull-Rom reconstruction is clearly superior in smooth signal region [3].

F. Mitchell-Netravali approximation

This function is the weighted sum of Catmull-Rom spline and cubic B-spline. This method is compromise, producing little overshoot, high edge sharpness, and good signal continuity in smooth region. Since the resulting function does not pass through the original sample points, the Mitchell-Netravali method is an approximation and not an interpolation [3].

G. Lanczos interpolation

Lanczos interpolation function is a mathematical formula used to smoothly interpolate the value of a digital image between its samples. It maps each sample of the given image to a translated and scaled copy of the Lanczos kernel, which is a sinc function windowed by the central hump of a dilated sinc function. The sum of these translated and scaled kernels is then evaluated at the desired pixel [16]. Lanczos interpolation has the best properties in terms of detail preservation and minimal generation of aliasing artifacts for geometric transformations not involving strong down sampling. The number of neighboring pixels for consideration varies as the order of the kernel. If the order is chosen to be 2, 16 pixels are considered while if the order is 3, 36 neighboring pixels are utilized for interpolation. However the higher order Lanczos interpolation require high computational time, which make them not suitable for many commercial software [12].

III.RESULT ANALYSIS

All seven algorithms are implemented in java language. ImageJ 1.50b with java 1.6.0_24 (64bit) an open source java software developed by Wayne Rasband at National Institute for Health (NIH) is used for image read, write and display. The software installed on a standard Window 8 Enterprise machine having Intel® Core i3 CPU @ 3.07 Ghz and installed DDR3 RAM of 4GB, is used for compilation and execution of java programs. Two input test image of size 256x256 (Lena and Peppers) and corresponding interpolated output images of size 256x256 produced by seven implementations are shown in Fig. 1. During the interpolation of different images, translation parameters (in pixel) $dx = 10.5$ and $dy = -3.25$ is used in x and y direction respectively. As the images are translated, in this paper for display, we cropped and scaled images size to 256x256 pixel resolution.

While dealing with different algorithms it generates number of artifact such as aliasing, moiré and blurring. Nearest neighbour interpolation produces a high aliasing effect resulting in jagged edges. Bilinear interpolation reduces the aliasing effect but causes a moderate blurring of the edges. Bicubic and Cubic B-spline interpolation produces a moderate aliasing, blurring and an edge halo effect. Catmull-Rom interpolation is superior in smooth signal region and provides the sharpness as Bicubic and Cubic B-spline algorithms. Mitchell-Netravali provides

high edge sharpness and good signal continuity in smooth region. Lanczos interpolation delivers an image quality very similar to that of Bicubic, but also utilizes the sine function repeatedly which itself requires a large number of addition and multiplication operations according to Taylor series approximation.

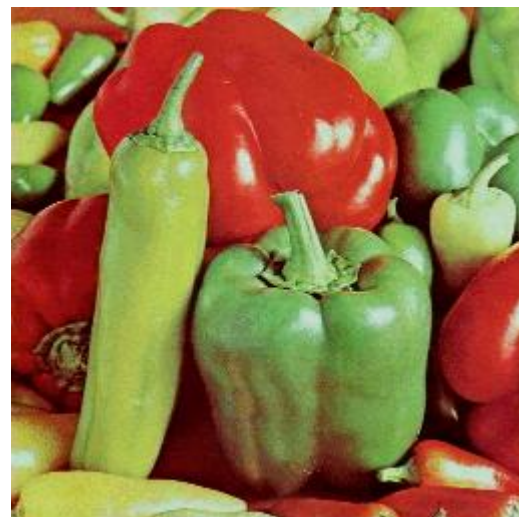
Comparison of time complexity of all seven image interpolation algorithms is shown in Table 1 for Lena image. Computationally, nearest neighbour interpolation is the least intensive as it considers only one pixel for interpolation and Lanczos order 3 uses the maximum time of all the seven algorithms due to more number of pixel consideration

Table1: Time complexity of interpolation algorithms

Interpolation Algorithm	Time in Second(s)
Nearest neighbour	0.081
Bilinear	0.140
Bicubic	0.169
Cubic B-Spline	0.174
Cutmull-Rom	0.156
Mitchell-Netravali	0.172
Lanczos order 3	1.094



(a)



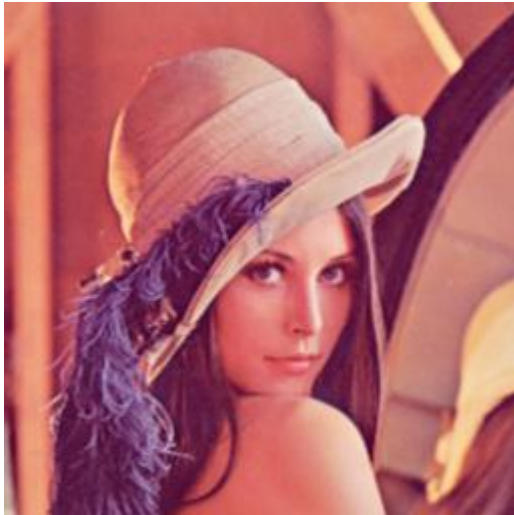
(a)



(b)



(b)



(c)



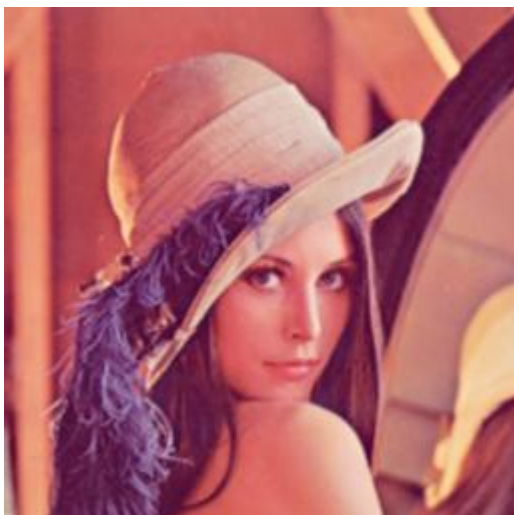
(c)



(d)



(d)



(e)



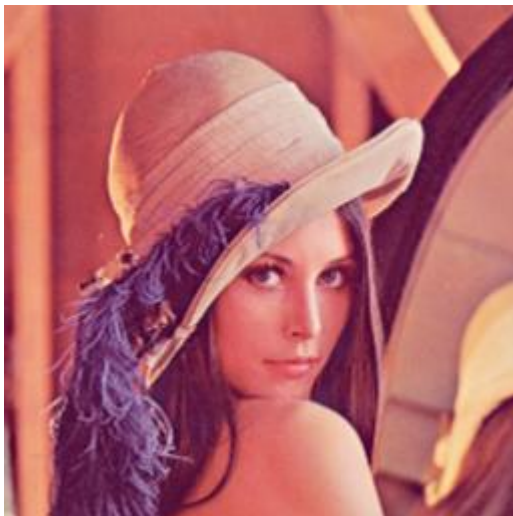
(e)



(f)



(f)



(g)



(g)



(h)



(h)

Fig. 1 (a) Input image (Lena and Peppers), (b) Nearest-neighbour, (c) Bilinear, (d) Bicubic, (e) Cubic B-spline, (f) Catmull-Rom, (g) Mitchell-Netravali, (h) Lanczos order 3

IV. CONCLUSION

Image interpolation has become a fundamental processing task and plays an important role in many computer graphics applications. Although there are myriad of interpolation algorithms available in literature, not all of them are suitable for implementing real-time image applications. In this paper, we discussed the main non-adaptive image interpolation algorithms suitable for real-time applications. A comparative analysis of seven image interpolation algorithms is presented based on the results of their software implementation. Of the seven image interpolation algorithms evaluated, the Catmull-Rom and Mitchell-Netravali approximation algorithms offer the best result between image quality and computational complexity for real-time image interpolation applications. Bicubic and Cubic B-spline generates interesting result, Bicubic uses less time complexity compared to Catmull-Rom and Mitchell-Netravali. Lanczos order 3 is using maximum time computational complexity, hence results evaluated are not at par with the above discussed algorithms.

ACKNOWLEDGEMENT

The authors express their deep sense of gratitude to **Wilhelm Burger** and his team for helping the implementation of java codes on ImageJ 1.50b for our research paper. We extend our sincere acknowledgements to all the faculty members of our institutes and other contributors for providing us the technical assistance for carrying out the study.

REFERENCES

- [1] D. Vaquero, M. Turk, K. Pulli, M. Tico, and N. Gelfand, "A survey of image retargeting techniques", SPIE Optical Engineering+ Applications, pp. 779814-779814, Aug. 2010.
- [2] R. C. Gonzalez, Digital image processing. Pearson Education India; 2009.
- [3] W. Burger, and M. J. Burge, Digital image processing: an algorithmic introduction using Java, 1st ed., Springer, 2009.
- [4] G. X. Zhang, M.M. Cheng, S.M. Hu, and R.R. Martin, "A Shape-Preserving Approach to Image Resize," Pacific Graphics 2009, vol. 28, no. 7, Sept. 2009.
- [5] N. Muelle and T. K. Nguyen, "Image interpolation using classification and stitching", in Proc. of the IEEE International Conference on Image Processing, pp. 901-904.
- [6] H.B. Kekre, T Sarod and S Thepade, "Grid based image scaling technique," International Journal of Computer Science And Applications., vol. 1, no. 2, Aug. 2008.
- [7] W. Zhe, Z. Jiefu and Z. Mengchu, "A Fast Autoregression Based Image Interpolation Method", in Proc. of the IEEE International Conference on Networking, Sensing and Control, pp.1400-1404, 2008.
- [8] A. Amanatiadis, L. Andreadis and K. Konstantinidis, "Fuzzy Area-Based Image Scaling", in Proc. of the IEEE Instrumentation and Measurement Technology Conference, pp. 1-6, 2007.
- [9] F. Liang and K. Xie, "An Image Interpolation Scheme combined with Artificial Neural Network", In Proc. of the Third International Conference on Natural Computation, pp. 99-102, 2007.
- [10] V. Patel and K. Mistree, "A Review on Different Image Interpolation Techniques for Image Enhancement", International Journal of Emerging Technology and Advanced Engineering, Vol. 3, pp. 129-13, Dec. 2013.
- [11] S.D. Ruikar and D.D. Doye, "Image Denoising using Tri Nonlinear and Nearest Neighbour Interpolation with Wavelet Transform", International Journal of Information Technology and Computer Science, Vol.4, pp. 36-44, Aug.2012.
- [12] C. Singh, S. Singh, R. Saini and A. K Saini, "A Comparative Analysis of Image Scaling Algorithms". International Journal of Image, Graphics and Signal Processing, pp.55-62, April. 2013
- [13] P. Miklo., "Image interpolation techniques", In 2nd Siberian-Hungarian Joint Symposium on Intelligent Systems. Oct. 2004
- [14] A. Prajapati, S. Naik and S. Mehta, "Evaluation of Different Image Interpolation Algorithms", International Journal of Computer Applications, Vol.58, Issue.1, pp. 6-12, Jan. 2012.
- [15] P. S. Parsania and P. V. Virparia, "A Review: Image Interpolation Techniques for Image Scaling", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2, Issue 12, pp. 7409-7413, Dec. 2014
- [16] (2016) Wikipedia website. [Online]. Available: http://en.wikipedia.org/wiki/Lanczos_resampling

BIOGRAPHIES



Pankaj S. Parsania, received MSc IT in year 2004 and in pursuit for Ph.D. from Department of Computer Science, Sardar Patel University, Vallabh Vidyanagar, Gujarat, India. Currently he is working as an Assistant Professor at College of FPTBE, AAU, Anand, Gujarat, India.

His area of interests are Digital Image Processing, Data Structure, Java, and Database Management System.



Paresh V. Virparia, joined the Department of Computer Science, Sardar Patel University, Vallabh Vidyanagar in 1989 and currently working as a Director and Professor. 7 research scholars have completed their Ph.D. (Computer Science) under his guidance. Currently, 7 more students are doing their Ph. D. under the guidance of him. His publications include 40 papers in International Journal, 16 papers in National Journals and 47 papers in national conferences/seminars. His research interests include the areas of Computer Simulation & Modeling, Digital Image Processing, Data Mining, Networking and IT enabled services. He is an editor and editorial review board member in several journals/magazines.