Analysis and comparison of image interpolation techniques

Digital Signal Analysis and Applications - Project

Mayank Gupta
201101004
mayank.g@students.iiit.ac.in

Ankush Jain
201101010
ankush.jain@students.iiit.ac.in

***Abstract*—Interpolation is an important problem in the field of Image Processing. It finds practical use in all kinds of image manipulation software in publishing, designing etc. Various methods for Image Interpolation have been proposed. We present an analysis of some of the common interpolation methods, and compare their results.**

Keywords—interpolation, zooming, bilinear, bicubic, spline, fft

I. INTRODUCTION

Image Interpolation refers to the process of enlarging the dimensions of an image, while having minimum effect on the contents, and quality of an image. It is a very common problem, and various algorithms, varying in complexity, processing power and output quality (including possible artifacts) have been proposed. Since there is an inherent information loss while *downsamping* an image, it is not possible to obtain exactly the same image while *upsampling* it. The algorithms differ in how well they hide the lack of information and the visual quality of the resultant output.

II. ALGORITHMS

A. Bilinear Interpolation

Bilinear interpolation considers the closest *2x2* neighborhood of known pixel values in order to extrapolate the unknown pixel. Following is the simplified formula for calculating the unknown value assuming the four points on which the value is known are *(0, 0), (0, 1), (1, 0) and (1, 1).*

$$f(x,y) = \begin{array}{c} f(0,0)(1-x)(1-y) \\ +f(1,0)x(1-y) \\ +f(0,1)(1-x)y \\ +f(1,1)xy \end{array}$$

B. Bicubic Interpolation

Compared with Bilinear Interpolation, Bicubic Interpolation takes into account the closest *4x4* neighborhood of known pixel values. The actual equation involved in the computation is quite complex, but is of the form

$$\sum \sum a_{ij} x^i y^j$$

C. Spline Interpolation

A mathematical curve called *spline* is the interpolant used to predict the unknown pixel value. A curve that best fits the known values is defined, and the unknown value is then extrapolated from that curve. It achieves better results compared to polynomial interpolation, as the error is quite small, even for low-degree spline curves. The mathematics of it is quite involved and we shall not present it in this report.

D. FFT-based Interpolation

This is based on a simple, but extremely useful property of *FFT* that zero-padding in one domain results in increased sampling rate in the other domain. The method is therefore simple, convert the given image to frequency domain, zero-pad the signal obtained, and then take the *inverse-fft* of the padded signal. The image obtained has more samples, and therefore a larger size.

III. PROCEDURE

1. We take a sample image, and convert it to *grayscale.* For the sake of simplicity, we perform the entire procedure on grayscale images.
2. For testing for a factor *f*, we first downsample that image by a factor of *f*, using standard inbuilt functions.
3. We then pass the smaller image so obtained by our interpolation algorithms, and scale it to the original size.
4. We then compute the RMS error of the original image with the images so obtained, and compare the results.
5. We also analyze the visual quality of the results.
6. The above procedure is repeated for different images, and different factors.

IV. RESULTS

Results for two images are displayed in this section. The first image is downsampled and then upsampled by a factor of *2*, while the second image is downsampled and upsampled by a factor


Fig. 1. Original figure – a cat


Fig. 2. Bilinear Interpolation, 2X


Fig. 3. Bicubic Interpolation, 2X


Fig. 4. Spline Interpolation, 2X


Fig. 5. FFT-based Transpolation, 2X

Fig. 6. Original Image

Fig. 7. Bilinear Interpolation – Factor 4x

Fig. 8. Bicubic Interpolation – Factor 4X

Fig. 9. Spline Interpolation – Factor 4X

Fig. 10. FFT-based Interpolation – Factor 4X

V. ROOT MEAN SQUARE ERROR

TABLE I. RMSE

Image	Interpolation Algorithm			
	Bilinear	***Bicubic***	***Spline***	***FFT***
Cat	3.4021	4.2031	3.4013	8.9536
Lena	4.1394	5.0797	4.4189	3.0195

a. *Computed using MATLAB*

VI. CONCLUSIONS

The RMS error values obtained are a bit random. The only conclusions that may be drawn are that the performance of *Spline Interpolation* seems to be consistently better than the average.

However, RMS error is not a very good metric for measuring extrapolation quality. It is only a measure of how much the pixels match, but the overall visual quality is ignored by the metric.

On visual comparison, we draw the following conclusions. Bicubic interpolation results in sharper images compared to Bilinear interpolation, although there is a significant amount of jaggedness in both.

Spline interpolation seems to result in *slightly* better output compared to Bicubic interpolation, at the cost of significantly extra computational intensiveness.

FFT-based interpolation produces the most interesting output. There is a very perceptible smoothness in the object boundaries, but it produces artifacts. The severity of the problem is dependent on the input image. The image from FFT-based interpolation is also a bit washed out, which results in a high RMS error shown previously. In spite of the high RMS error, the output is as acceptable, if not better, compared to other algorithms.

For regular purposes, Bicubic interpolation strikes the best balance between output quality and computational load. The quality of FFT-based interpolation is highly subjective, and may be considered excellent if artifacts are acceptable, or very poor otherwise.

REFERENCES

[1] Digital Image Zooming – Sample Codes in MATLAB, http://thilinasameera.wordpress.com/2010/12/24/digital-image-zooming-sample-codes-on-matlab/

[2] Image Zooming in MATLAB using Spline Interpolation, http://thilinasameera.wordpress.com/2011/05/05/digital-image-zooming-on-matlab-using-spline-interpolation/

[3] Image Interpolation with the Fast Fourier Transform, http://www.eliteraspberries.com/blog/2013/04/image-interpolation-with-the-fast-fourier-transform.html.

[4] Cubic Spline Data Interpolation – MathWorks India, http://www.mathworks.in/help/matlab/ref/spline.html