Optimization of PSNR of Zooming Image using Cubic Spline Interpolation Method

Ankita Agrawal, Arpan Shah, Ramesh Bharti

Abstract—In this paper, we have presented the cubic spline interpolation method for zooming in an image. The cubic spline algorithm is programmed in MATLAB as m-file. Firstly, according to the zooming intensity, sufficient zeroes are inserted between each pixel and then those zeroes are replaced by the respective values generated by the m-file. Finally, a zoomed image is constructed.

Index Terms—Digital image processing, zoom in, cubic spline interpolation method, MATLAB.

I. INTRODUCTION

The field of digital image processing refers to processing digital images using digital computers. A digital image is comprised of finite number of values each of which has a location and a value. Generally speaking these values are referred to as pixels. Using MATLAB, this process becomes a bit easier as the principles involved in MATLAB for matrix manipulations can easily be used in image processing. One of the major applications of image processing is object or face recognition, in addition to which zooming in and zooming out an image is also a very important field of interest. In this paper, we are zooming in an image using cubic spline interpolation method. We programmed the method in MATLAB as m-file and tested it on “cameraman.tif”.

Cubic Spline Interpolation Method: This method relies on constructing a polynomial of low degree between each pair of known data points. If a first degree polynomial is used, it is called linear interpolation. For second and third degree polynomials, it is called quadratic and cubic splines respectively. The higher the degree of the spline, the smoother we get the curve. To obtain a smoother curve (or finer resolution), we get the curve. To obtain a smoother curve (or finer resolution), we construct a cubic polynomial of the form

\[
S_i(x) = \begin{cases} 
S_1(x) & \text{if } x_1 \leq x \leq x_2 \\
S_2(x) & \text{if } x_2 \leq x \leq x_3 \\
& \vdots \\
S_{n-1}(x) & \text{if } x_{n-1} \leq x \leq x_n 
\end{cases}
\]

where \(S_i(x)\) is a third degree polynomial defined by

\[
S_i(x) = a_i(x-x_i)^3 + b_i(x-x_i)^2 + c_i(x-x_i) + d_i, (1)
\]

\(x_i\) are the data points.

and \(i = 1,2,3,\ldots,n\).

The first and second derivatives of eq. (1) are given by

\[
S_i'(x) = 3a_i(x-x_i)^2 + 2b_i(x-x_i) + c_i, (2)
\]

\[
S_i''(x) = 6a_i(x-x_i) + 2b_i, (3)
\]

where \(i\) being same as \(i = 1,2,\ldots,n-1\).

Our spline will need to conform to the following stipulations.

1. The piecewise function \(S_i(x)\) will interpolate all data points.
2. \(S_i(x)\) will be continuous on the interval \([x_i,x_{i+1}]\).
3. \(S_i'(x)\) will be continuous on the interval \([x_i,x_{i+1}]\).
4. \(S_i''(x)\) will be continuous on the interval \([x_i,x_{i+1}]\).

Following the stipulations, we deduce the formulas for the coefficients as

\[
a_i = \frac{M_{i+1} - M_i}{h_i}, (4)
\]

\[
b_i = \frac{M_i}{6h_i}, (5)
\]

\[
c_i = \frac{y_i - y_{i+1}}{h_i} - \frac{(M_{i+1} + 2M_i)}{6}, (6)
\]

\[
d_i = y_i, (7)
\]

\(y_i\) is the pixel color intensity at data point \(x_i\).

After calculating the values of \(M_i\) using eq.(8), we calculate the values of \(a_i, b_i, c_i\) and \(d_i\) for each pair of adjacent pixel and calculate the respective values for the zeroes filled in between them. The formula for filling the values in their respective positions was generated by hit and trial method. Now after constructing the image matrix and using the ‘imshow’ command we get the zoomed image.

For cubic spline, we calculate the values of \(M_i\) in terms of \(y_i\) using eq.(8) and after calculating the values \(M_2, M_3, \ldots, M_{n-1}\), we find the values of \(M_1\) and \(M_n\) using eq.(9) and eq. (10).
Optimization of PSNR of Zooming Image using Cubic Spline Interpolation Method

Results: Taking ‘cameraman.tif’ as sample grey scale image and the zooming intensity as 2, we can compare the results as the psnr value was found to be 148.48 and the execution time was 384.1070 seconds.

Fig. 1 A part of cameraman.tif of size 50x50

Fig. 2. 4x Zoomed in image of size 200x200

So, we found that the zoomed image is much clear as compared to normally available zooming applications.

REFERENCES:
[2] Cubic Spline Interpolation by Sky McKinley and Megan Levine