Available online at www.sciencedirect.com
SCIENCE DIRECT®
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
Journal of Computational and Applied Mathematics 195 (2006) 46–53
www.elsevier.com/locate/cam

Adaptive osculatory rational interpolation for image processing☆

Min Hu[a], Jieqing Tan[b,*]

[a]College of Computer and Information Science, Hefei University of Technology, Hefei 230009, PR China
[b]Institute of Applied Mathematics, College of Sciences, Hefei University of Technology, Hefei 230009, PR China

Received 15 August 2004; received in revised form 15 March 2005

Abstract

Image interpolation is a common problem in image applications. Although many interpolation algorithms have been proposed in the literature, these methods suffer from the effects of imperfect reconstruction to some degree, most often, these effects manifest themselves as jagged contours or blurred edges in the image. This paper presents a method for preserving the contours or edges based on adaptive osculatory rational interpolation kernel function, which is built up by approximating the ideal interpolating kernel function by continued fractions. It is a more accurate approximation for the ideal interpolation in space domain or frequency domain than by other linear polynomial interpolation kernel functions. Simulation results are also presented to demonstrate the superior performance of image magnification.

Keywords: Image processing; Adaptive; Osculatory rational interpolation; Continued fractions

1. Introduction

Image interpolation is used in a wide variety of digital image processing tasks including image scaling, image warping and image restoration. In these processes, it is important to construct an efficient interpolation function. It is known that if a signal is band-limited and uniform sampling is done at above the Nyquist frequency, the signal can be reconstructed perfectly, and this reconstructed kernel function (the ideal interpolation) is the sinc function [1,2], defined by

$$f(x) = \frac{\sin(\pi x)}{\pi x}. \tag{1}$$

However, the ideal interpolation kernel function does not produce good results in practice: the interpolation function in the space domain does not have compact support, and it must be clipped to a bounded interval before being used to interpolate the samples of the image to be reconstructed (see Fig. 1). Therefore, in order to achieve a balance between computational efficiency and the quality of the interpolation image, many piecewise local polynomial interpolation kernels have been proposed and analyzed [4,5], for examples: nearest-neighbor interpolation, bilinear interpolation,

☆ Supported by the National Natural Science Foundation of China under Grant no. 10171026, no. 60473114 and Anhui Provincial Natural Science Foundation, China under Grant no. 03046102.
* Corresponding author.
E-mail address: jqtan@mail.hf.ah.cn (J. Tan).

0377-0427/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2005.07.011

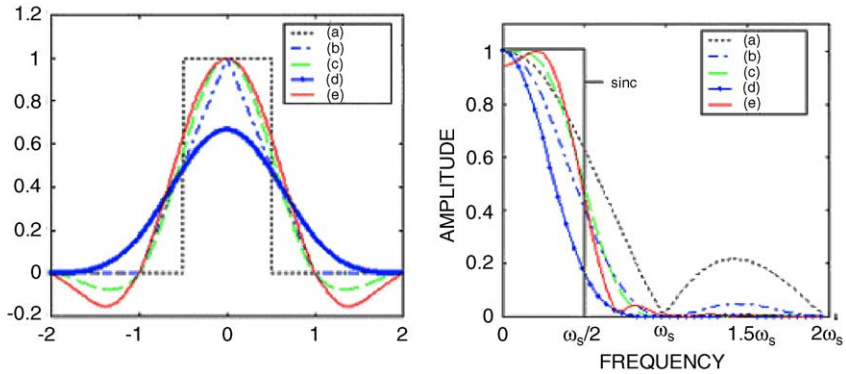

Fig. 1. (a) Nearest-neighbor, (b) linear, (c) Catmull-Rom cubic, (d) cubic B-spline. (e) osculatory rational interpolation kernel. (1-1): functional features, (1-2): Frequency responses on a linear scale for evaluation of pass band performance.

bicubic interpolation, etc. In general, images reconstructed by higher-order polynomial kernels have better vision result, no blocky artifacts, less blurring, but they may eliminate some high frequencies present in the original image, and make the contours or edges of interpolated images distorted. The direct reason is that their frequency properties are far from being the ideal kernel at pass band or near the stop band.

From the view that the interpolation kernels should approximate the ideal kernel, the geometry of the interpolation kernels should be controlled firstly. The standards of improving the kernels frequency spectrum are: (1) good stop band and pass band, (2) the width of main band should be narrow. Secondly, the method of approximating the sinc function by using nonlinear algorithm can give an efficient result, because the sinc function is a nonlinear function. For above two reasons, this paper presents a new method to establish an adaptive osculatory rational interpolation kernel by using continued fractions.

This paper is organized as follows. In Section 2, we introduce the concept of Thiele-type continued fractions as the basis of this proposed interpolation scheme. In Section 3, a method to obtain the adaptive osculatory rational interpolation kernel is described. In Section 4, the space-domain property and frequency-domain property of the established kernel are analyzed, and the comparison with other polynomial interpolation kernels is made. Experimental results of this new kernel with discussions and conclusions are presented in Sections 5 and 6, respectively.

2. Thiele-type osculatory rational interpolants

Suppose $f(x)$ is a function defined on a subset G of the complex plane and $X = \{x_i | i \in \mathbb{N}\}$ is the set of points belonging to G. The nth Thiele-type interpolating continued fraction of $f(x)$ is of the following form [6,7]:

$$R_n(x) = \varphi[x_0] + \frac{x - x_0|}{|\varphi[x_0, x_1]} + \cdots + \frac{x - x_{n-1}|}{|\varphi[x_0, x_1, \cdots, x_n]}, \tag{2}$$

where $\varphi[x_0, x_1, \ldots, x_i]$, $i = 0, 1, 2, \ldots, n$, are the inverse differences defined as follows:

$$\varphi[x_i] = f(x_i),$$
$$\varphi[x_i, x_j] = \frac{x_j - x_i}{f(x_j) - f(x_i)},$$
$$\varphi[x_0, x_1, \ldots, x_k] = \frac{x_k - x_{k-1}}{\varphi[x_0, x_1, \ldots, x_{k-2}, x_k] - \varphi[x_0, x_1, \ldots, x_{k-2}, x_{k-1}]}.$$

$R_n(x)$ is a rational function of type $([(n+1)/2]/[n/2])$ satisfying $R_n(x_i) = f(x_i)$, $i = 0, 1, 2, \ldots, n$, and $[x]$ denotes the greatest integer not exceeding x.

If some support abscissae coincide with each other, say, $x_{i0} = x_{i1} = \cdots = x_{ip}$, then the above defined inverse differences have to be computed by taking limits (we refer to [7] for details), the rational function $R_n(x)$ satisfies

$$R_n(x_i) = f(x_i), \quad i \in \{0, 1, \ldots, n\} - \{i0, i1, \ldots, ip\},$$
$$R_n^{(k)}(x_{i0}) = f^{(k)}(x_{i0}), \quad k = 0, 1, \ldots, p,$$

and $R_n(x)$ is said to be an osculatory rational interpolation function to $f(x)$.

3. Adaptive osculatory rational interpolation kernel function

Interpolation kernel function should be more accurately approximate to the ideal interpolation (the sinc function) not only in space domain but also in frequency domain, moreover the interpolation kernels should be a symmetric function. Here we follow the rules of cubic interpolation kernel [5], the osculatory rational interpolation kernel is composed of piecewise $(\frac{2}{2})$ type rational fractions defined on the subintervals $(-2, -1)$, $(-1, 0)$, $(0, 1)$, and $(1, 2)$. Outside the interval $(-2, 2)$, the interpolation kernel is zero. For unit-spaced samples, the interpolation function can be written in the form

$$g(x) = \sum_{k=0}^{3} g(x_k) I(x - x_k),$$

where the $g(x_k)$ are the sample values and $I(x)$ is the interpolation kernel. In this paper we denote the rational interpolation kernel by $\mathrm{RI}(x)$.

Since $\mathrm{RI}(x)$ is symmetric, we only need to construct the representation of $\mathrm{RI}(x)$ on the subinterval $(0, 2)$. $\mathrm{RI}(x)$ is approximate to the sinc function more accurately on subinterval $(0, 1)$ than on subinterval $(1, 2)$. Based on the preceding section, the construction of $\mathrm{RI}(x)$ can be summed up into the following two steps:

Step 1: The rational interpolation kernel on the subinterval $[0, 1]$.

Here we choose four interpolation nodes: $(0, f(0))$, $(0.25, f(0.25))$, $(0.5, f(0.5))$, $(1, f(1))$, besides $\mathrm{RI}_{01}(x)$ satisfies $\mathrm{RI}'_{01}(1) = f'(1)$, then $\mathrm{RI}_{01}(x)$ can be expressed in the form of the following continued fraction:

$$RI_{01}(x) = \varphi[x_0] + \frac{x - x_0 |}{| \varphi[x_0, x_1]} + \frac{x - x_1 |}{| \varphi[x_0, x_1, x_2]} + \frac{x - x_2 |}{| \varphi[x_0, x_1, x_2, x_3]} + \frac{x - x_3 |}{| b_4},$$

where

$$\varphi[x_i] = f(x_i),$$

$$\varphi[x_0, x_1] = \frac{x_1 - x_0}{\varphi[x_1] - \varphi[x_0]},$$

$$\varphi[x_0, x_1, x_2] = \frac{x_2 - x_1}{\varphi[x_0, x_2] - \varphi[x_0, x_1]},$$

$$\varphi[x_0, x_1, x_2, x_3] = \frac{x_3 - x_2}{\varphi[x_0, x_1, x_3] - \varphi[x_0, x_1, x_2]}.$$

With the above interpolation data, one works out

$$RI_{01}(x) = 1 + \frac{x |}{| -2.5075} + \frac{x - 0.25 |}{| 0.2209} + \frac{x - 0.5 |}{| 1.8077} + \frac{x - 1 |}{| b_4}. \tag{3}$$

From the initial conditions $\mathrm{RI}'_{01}(1) = -1$ and Eq. (3) follows $b_4 = -1.3889$. After some computation and simplification, one gets

$$\mathrm{RI}_{01}(x) = \frac{-0.168x^2 - 0.9129x + 1.0808}{x^2 - 0.8319x + 1.0808} \quad (0 \leqslant x \leqslant 1).$$

Step 2: The rational interpolation kernel on the subinterval [1,2].

If the distance between the interpolated point and the nearest sample point is $d(0 < d \leqslant 0.5)$, then the interpolated point value is calculated from the four nearest sample points and the four interpolation kernel function values $\mathrm{RI}(1+d)$, $\mathrm{RI}(d)$, $\mathrm{RI}(1-d)$, $\mathrm{RI}(2-d)$, these kernel function values satisfy the condition

$$\mathrm{RI}(1+d) + \mathrm{RI}(d) + \mathrm{RI}(1-d) + \mathrm{RI}(2-d) = 1. \tag{4}$$

This equation can be written as

$$\mathrm{RI}_{12}(1+d) + \mathrm{RI}_{12}(2-d) = 1 - \mathrm{RI}_{01}(d) - \mathrm{RI}_{01}(1-d). \tag{5}$$

In order to maintain the shape of the sinc function, we set the interpolation nodes of $\mathrm{RI}_{12}(x)$ as $(1, f(1))$, $(1+d, f(1+d))$, $(1.5, f(1.5))$, $(2-d, f(2-d))$, $(2, f(2))$, and the Thiele-type continued fraction expression of the $\mathrm{RI}_{12}(x)$ is as follows:

$$RI_{12}(x) = \psi[x_0] + \frac{x - x_0}{\left|\psi[x_0, x_1]\right.} + \frac{x - x_1}{\left|\psi[x_0, x_1, x_2]\right.} + \frac{x - x_2}{\left|\psi[x_0, x_1, x_2, x_3]\right.} + \frac{x - x_3}{\left|\psi[x_0, x_1, x_2, x_3, x_4]\right.}. \tag{6}$$

In (6), $x_0 = 1$, $x_1 = 1 + d$, $x_2 = 1.5$, $x_3 = 2 - d$, $x_4 = 2$, $(d \neq 0.5)$, $\psi[x_0, x_1, \cdots, x_i]$, $i = 0, 1, 2, 3, 4$, are the inverse differences.

In order to strengthen the relativity of the interpolation point and the nearest sample point, and ensure the correction of Eq. (5), we set

$$\begin{aligned}
&a = \mathrm{RI}_{01}(d) + \mathrm{RI}_{01}(1-d),\\
&\mathrm{RI}_{12}(1+d) = 2(1-a)/3,\\
&\mathrm{RI}_{12}(2-d) = (1-a)/3,\\
&\mathrm{RI}_{12}(1) = 0,\\
&\mathrm{RI}_{12}(1.5) = -0.1366,\\
&\mathrm{RI}_{12}(2) = 1e - 10.
\end{aligned}$$

Then Eq. (6) can be expressed as

$$\mathrm{RI}_{12}(x) = \frac{a_1x^2 + b_1x + c_1}{x^2 + b_2x + c_2}. \tag{7}$$

In (7), a_1, b_1, c_1, b_2, c_2 are the coefficients to be determined by the parameter d. The solution for the rational interpolation kernel function established by means of the Thiele-type continued fraction algorithm is

$$\mathrm{RI}(x) = \begin{cases} \dfrac{-0.168|x|^2 - 0.9129|x| + 1.0808}{|x|^2 - 0.8319|x| + 1.0808} & |x| \leqslant 1, \\ \dfrac{a_1|x|^2 + b_1|x| + c_1}{|x|^2 + b_2|x| + c_2} & 1 < |x| \leqslant 2, \\ 0 & 2 < |x|. \end{cases}$$

The form of $\mathrm{RI}(x)$ varies with d, in other words, $\mathrm{RI}(x)$ varies with the interpolating points, this is the reason why we call the $\mathrm{RI}(x)$ the adaptive osculatory rational interpolation kernel.

Example 1. Set $d = 0.25$, then we get

$$\begin{aligned}
&a = 1.1962,\\
&\mathrm{RI}_{12}(1+d) = -0.1308,\\
&\mathrm{RI}_{12}(2-d) = -0.0654,\\
&\psi[x_0] = 0,\\
&\psi[x_0, x_1] = -1.9113,\\
&\psi[x_0, x_1, x_2] = -0.1429,\\
&\psi[x_0, x_1, x_2, x_3] = 2.7588,\\
&\psi[x_0, x_1, x_2, x_3, x_4] = 0.3382.
\end{aligned}$$

Therefore

$$RI_{12}(x) = \frac{x}{-1.9113} + \frac{x-1.25|}{|-0.1429} + \frac{x-1.5|}{|2.7588} + \frac{x-1.75|}{|0.3382}.$$

After some computation and simplification, we get

$$\mathrm{RI}_{12}(x) = \frac{0.1953x^2 - 0.5858x + 0.3905}{x^2 - 2.4402x + 1.7676}.$$

Thus the rational interpolation kernel function has the following form:

$$\mathrm{RI}(x) = \begin{cases} \dfrac{-0.168|x|^2 - 0.9129|x| + 1.0808}{|x|^2 - 0.8319|x| + 1.0808} & |x| \leqslant 1, \\ \dfrac{0.1953|x|^2 - 0.5858|x| + 0.3905}{|x|^2 - 2.4402|x| + 1.7676} & 1 < |x| \leqslant 2, \\ 0 & 2 < |x|. \end{cases} \tag{8}$$

Example 2. Set $d = 0.4$,as we did above, then the rational interpolation kernel function is of the following form:

$$\mathrm{RI}(x) = \begin{cases} \dfrac{-0.168|x|^2 - 0.9129|x| + 1.0808}{|x|^2 - 0.8319|x| + 1.0808} & |x| \leqslant 1, \\ \dfrac{0.7588|x|^2 - 2.2764|x| + 0.5176}{|x|^2 - 3.0|x| + 3.6389} & 1 < |x| \leqslant 2, \\ 0 & 2 < |x|. \end{cases}$$

4. Analysis

In order to evaluate the adaptive rational interpolation algorithm, we compare it, respectively, with nearest-neighbor interpolation, linear interpolation, Catmull-Rom cubic interpolation and cubic B-spline interpolations which are used extensively in image processing.

Nearest-neighbor interpolation kernel function:

$$I(x) = \begin{cases} 1 & |x| \leqslant \frac{1}{2}, \\ 0 & \frac{1}{2} < |x|. \end{cases}$$

The linear interpolation kernel function:

$$I(x) = \begin{cases} 1 - |x| & |x| \leqslant 1, \\ 0 & 1 < |x|. \end{cases}$$

Catmull-Rom cubic interpolation kernel function:

$$I(x) = \begin{cases} \frac{3}{2}|x|^3 - \frac{5}{2}|x|^2 + 1 & |x| \leqslant 1, \\ -\frac{1}{2}|x|^3 + \frac{5}{2}|x|^2 - 4|x| + 2 & 1 < |x| \leqslant 2, \\ 0 & 2 < |x|. \end{cases}$$

Cubic B-spline interpolation kernel function:

$$I(x) = \begin{cases} \frac{1}{2}|x|^3 - |x|^2 + \frac{2}{3} & |x| \leqslant 1, \\ -\frac{1}{6}|x|^3 + |x|^2 - 2|x| + \frac{4}{3} & 1 < |x| \leqslant 2, \\ 0 & 2 < |x|. \end{cases}$$

Here we set $d = 0.25$, the osculatory rational interpolation function is given in Eq. (8).

Various kernels are shown in Fig. 1(1-1). Most of the functional features are criteria applied in their derivation. Catmull-Rom cubic is generally accepted as the best cubic interpolant, but Fig. 1 shows that the approximation to the sinc function by the rational interpolation kernel is better in terms of the approximation to the sinc function.

Fig. 1(1-2) shows the frequency responses of various kernels approximating the sinc function for frequencies from 0 to $2w_s$ (w_s is sampling frequency). Only the positive frequency axis is shown as all of the functions are even. In the stop band a better performance is indicated by a function which stays closest to zero. Nearest-neighbor thus has appalling performance, with the linear, cubic, rational and cubic B-spline getting progressively better. The cubic B-spline has the best stop band performance of all the functions considered. The rational interpolating kernel's stop band responses are better. The best pass band performance of the five functions is shown by the rational interpolating kernel, followed by the Catmull-Rom. The interpolating quadratic has average stop band performance, thus the rational interpolating kernel has the best frequency response in high frequency region, which just meets the needs for processing images to preserve the high frequency information of image data.

According to the visual feature, the vision is sensitive to the image contours or edges, which are the high frequency information regions. Hence the rational interpolating kernel is a good choice for processing the image's edge regions and texture regions.

5. Computer simulations

The performances of the proposed kernel function can be deduced from some experiments. We believe that the improvements on visual quality brought by the osculatory rational interpolant can be easily observed when the images are viewed at a normal distance. In the first set of experiments with grayscale images (see Fig. 2), the osculatory

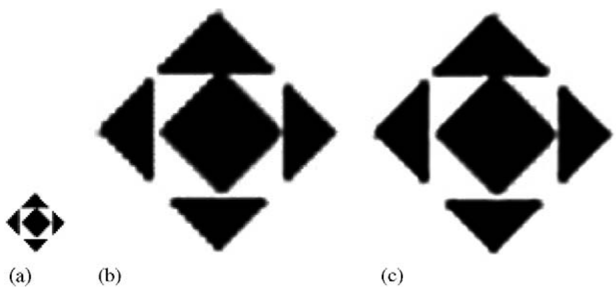

Fig. 2. "Rect" image with $3x$ magnification. Image (a) is original image. Image (b) is the result image magnified by the bicubic interpolation method. Image (c) is the result image magnified by the osculatory rational interpolation.

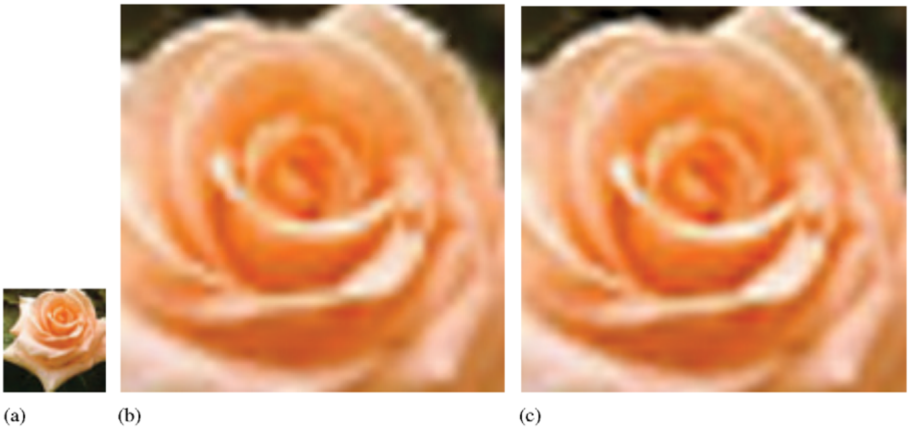

Fig. 3. "Rose" image with $6x$ magnification. Image (a) is original image. Image (b) is the result image magnified by the bicubic interpolation method. Image (c) is the result image magnified by the osculatory rational interpolation.

rational interpolation is compared with bicubic interpolation method. In Fig. 2, (a) is original image, (b) is the result image magnified by the bicubic interpolation method and (c) is the result image magnified by the rational interpolation method. We can observe that more jagged contours are dramatically produced in the magnified images by bicubic interpolation (Fig. 2(b)), while the osculatory rational method produces much smoother contours than bicubic does (Fig. 2(c)). In the second set of experiments with color images (see Fig. 3), Fig. 3(b) is based on bicubic interpolation techniques and Fig. 3(c) employs the osculatory rational interpolation. It can be observed that Fig. 3(c) has better visual quality, since the osculatory rational interpolation better preserves the geometric regularity around the color edges and thus generates interpolated images with higher visual quality.

From the performance analysis, we can conclude that osculatory rational interpolation based magnification gives comparatively better performance.

6. Summary

In this paper, we present a novel image magnification method based on the osculatory rational interpolation via continued fractions. On the one hand, this method has a good adaptability that the interpolation kernel function changes with the magnification. On the other hand, the kernel function has good stop band and pass band, which can significantly improve the processing results of the image contours and edges by existing methods for image magnification. The result images have better visual quality. Moreover, the executing time of this method is near that of the bicubic method, so it can be widely applied. Our future work include establishing the osculatory rational interpolation schemes for bivariate and vector valued functions, which may be developed on the basis of literature [3,8–11], to carry out the approximation to the multivariate kernel functions and explore the further applications of the methods in image processing.

References

[1] K.R. Castleman, Digital Image Processing, vol. 4, Tsinghua University Press, Beijing, 1998, pp. 97–117.
[2] J. Comes, L. Velho, Image Processing for Computer Graphics, Springer, New York, 1997, pp. 187–213.
[3] P.R. Graves-Morris, Vector valued rational interpolants I, Numer. Math. 42 (1983) 331–348.
[4] H.S. Hou, H.C. Andrews, Cubic splines for image interpolation and digital filtering, IEEE Trans. Acoust. Speech, Signal Process. 26 (6) (1978) 508–517.
[5] R.G. Keys, Cubic convolution interpolation for digital image processing, IEEE Trans. Acoust. Speech, Signal Process. ASSP-29 (6) (1981) 1153–1160.
[6] J. Stoer, R. Bulirsch, Introduction to Numerical Analysis, second ed., Springer, Berlin, 1992.

[7] J. Tan, The limiting case of Thiele's interpolating continued fraction expansion, J. Comput. Math. 19 (4) (2001) 433–444.
[8] J. Tan, Bivariate blending rational interpolants, Approx. Theory Appl. 15 (2) (1999) 74–83.
[9] J. Tan, Y. Fang, Newton–Thiele's rational interpolants, Numer. Algorithms 24 (2000) 141–157.
[10] J. Tan, P. Jiang, A Neville-like method via continued fractions, J. Comput. Appl. Math. 163 (2004) 219–232.
[11] J. Tan, G. Zhu, General framework for vector-valued interpolants, in: Z. Shi (Ed.), Proc. of the third China–Japan Seminar on Numerical Mathematics, Science Press, Beijing/New York, 1998, pp. 273–278.