

基于边界程度的图像插值算法

孔繁庭¹,刘俊华²

(1. 甘肃联合大学 电子信息工程学院, 兰州 730010; 2. 北京前沿软件开发有限公司, 北京 100041)

(kfanting@163.com)

摘要:为了有效地消除传统图像插值中出现的边缘锯齿现象,提出了基于边界程度的图像插值算法。该算法充分考虑自然图像边缘缓慢变化的特点,在图像边缘平滑的假设下,采用边界程度信息来抑制锯齿现象。实验结果表明,该算法能够很好地解决图像插值中的边缘锯齿现象问题,且不会产生其他明显的人工痕迹。此外,该算法还保持了传统插值算法简单有效的特点,实现和应用起来都比较方便。

关键词:图像插值;边界程度;边缘锯齿;图像放大

中图分类号:TP391.41 **文献标志码:**A

Image interpolation algorithm based on edge degree

KONG Fan-ting¹, LIU Jun-hua²

(1. School of Electronic and Information Engineering, Gansu Union University, Lanzhou Gansu 730010, China;

2. Frontier Software Development Company Limited Beijing, Beijing 100041, China)

Abstract: In order to eliminate the phenomenon of contour jaggies in conventional image interpolation schemes, an edge-degree-based image interpolation method was proposed. Under the smooth contours assumption, the method considered the smoothly varying characteristics of nature image contours, and utilized the information of edge degree to suppress contour jaggies. Simulation results show that the proposed method produces fewer jaggies, and it doesn't introduce other obvious artifacts. Furthermore, the method still preserves the simplicity and efficiency of conventional image interpolation methods, and is easy for implementation and application.

Key words: image interpolation; edge degree; contour jaggies; image magnification

0 引言

图像插值是根据低分辨率的图像来估计高分辨率的图像,广泛应用于医学图像重建、图像几何变换、图像去马赛克效应、视频大小转换等领域。从视觉效果上来看,插值后的图像的边界清晰程度和是否有明显的人工痕迹(如锯齿现象、块状现象、环状伪像等)是衡量插值算法好坏的重要指标。此外,插值算法的复杂度也是很重要,尤其是在实时性要求比较高的场合。

目前,图像插值算法有很多,它们大致可以分为三类:直接插值法、基于正则化函数的插值法和基于偏微分方程(Partial Differential Equation, PDE)的定向扩散插值法^[1]。

直接插值法可分为线性插值和非线性插值。线性插值一般是利用周围邻近点的线性叠加来估计未知点的值,如最近邻插值、双线性插值、双三次插值和基于内核的插值等都属此类^[2]。该类方法经过几十年的发展,比较成熟,并且由于算法简单、速度较快等优点,在实际问题中应用十分广泛。该类方法的缺点在于对于任何一种算法,其插值函数一旦确定下来,它在图像空域内就是移不变的,这种不变性导致它很难适应图像边缘快速变化的特征,因此在这些区域内容易产生模糊和锯齿现象。

为解决线性插值中出现的问题,非线性插值被提了出来。该方法一般是根据图像的局部结构特征来进行插值,如文献[3-5]根据图像的边界方向进行插值,文献[6]是根据图

像的局部协方差自适应地调整加权系数,文献[7]在线性插值的基础上通过误差的微调进行插值等。此类方法大都基于一定的插值模型,若输入图像的结构与该模型相符,则插值效果就很好,反之效果就较差。如对于文献[6]中提到的方法,若图像局部区域内边界方向较乱时,插值后的图像在该区域就会呈现涡流或鬼影似的痕迹。

基于三角形的双线性图像插值算法^[4],在水平、垂直,以及对角线方向上能够在一定程度上抑制传统双线性插值中出现的锯齿现象和模糊现象。但在对于其他角度的边界而言,依旧会存在明显的锯齿。本文在保持三角形双线性插值优点的基础之上,针对其容易出现锯齿现象的区域,提出基于边界程度的插值算法。该算法首先假设图像的边缘形状是平滑的,然后利用边界程度信息来实现这一假设。实验仿真结果表明该算法能够很好地抑制边缘锯齿现象。

1 基于三角形的双线性图像插值算法

在文献[4]中,提出了基于三角形的双线性图像插值算法,该算法的示意图如图1所示。设待插值点 (x, y) 的4个最近邻点为 $(i+k, j+l)$,其中 $k, l=0, 1$,该算法首先将这4个点组成的正方形按对角线方向划分为两个三角形区域,然后对每个区域内的待插值点分别进行三点的双线性插值。三角形区域的划分标准是判断两组对角线的像素值差别,选择较小的那组对角线作为分割线,即若 (i, j) 与 $(i+1, j+1)$ 的差别较小,则选择沿 135° 对角线划分,反之沿 45° 对角线划分。不

收稿日期:2010-12-24;修回日期:2011-03-01。

作者简介:孔繁庭(1971-),男,甘肃天祝人,副教授,硕士,主要研究方向:信号分析、图像处理;刘俊华(1985-),男,安徽临泉人,助理工程师,硕士,主要研究方向:信号分析、图像处理。

失一般性,现假设沿着 45° 对角线划分, (x,y) 在右下方的三角形内,则根据线性差值理论, (x,y) 处的估计值为:

$$\hat{f}(x,y) = (1-p-q)f(i+1,j+1) + pf(i,j+1) + qf(i+1,j) \quad (1)$$

其中 $p = i+1-x, q = j+1-y$ 为待插值点到直角像素点的距离, $f(i,j)$ 为像素点 (i,j) 上的灰度值。

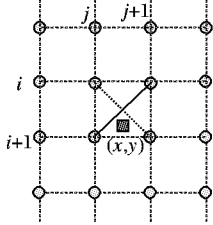


图1 基于三角形的双线性插值示意图

基于三角形的双线性插值本质上是尽量沿着边缘进行插值,从而减少插值中的模糊现象和锯齿现象。然而上述三角形区域的划分只考虑到4个点的情况,有时受噪声或其他因素的影响,并不能真正地反映图像边缘的方向。在进一步的改进中,考虑周围 3×3 窗口内每个正方形划分方向的统计信息。若窗口内沿某一方向划分的正方形个数大于或等于某个阈值 T_c ,则将中间正方形的划分方向修改为该方向。这在一定程度上更好地反映了图像局部的边缘方向。

在对角线方向和垂直水平的边界方向上,基于三角形的插值方法能够较好地抑制锯齿现象的产生。但对于其他方向的边界情况,锯齿现象依旧存在。本文即是针对三角线性插值的这一缺点,提出图像边缘形状平滑假设,并在该假设基础上对其进行改进。

2 基于边界程度的图像插值算法

我们认为如果能够假设图像的边缘形状是平滑的,则在进行图像插值时就可以利用该假设对前面所述各种插值方法表示的误差作出弥补,从而可以达到抑制边缘区域锯齿现象的目的。事实上对于自然场景中的边缘,除了少数的棱角外,其他绝大部分区域都是缓慢变化的,因此上述假设可以认为是近似成立的。

边界程度信息的引入主要是为了实现图像边界形状平滑的假设,而边界形状平滑假设如果能够实现就消除了边界锯齿现象。如图2所示,若灰色的边界是图中两个物体的分界线,线的两边是黑白两色的采样点,表征了两个物体。如果采用最近邻插值或线性插值对这些离散的点进行插值,得到的图像边界就很有可能是图像黑色的折线,这就出现了锯齿现象。如果能够在插值的时候,对 $(i,j+1)$, $(i+1,j)$ 等两边物体凸出的点进行抑制,使得插值后的边界曲线更加靠近这些点,那么就可以实现边界平滑的假设,从而达到消除锯齿现象的目的。

如何来表征或度量图中凸出的点,在本文中,采用边界程度信息。对于图2的情况,边界程度信息可以用每个点的水平方向、垂直方向和两个对角线方向上的边界程度的绝对值之和来表示,其中水平、垂直方向上的边界程度可以用式(2)中的 Sobel 算子来检测,而对角线 45° 和 135° 方向上的边界程度则用式(3)中类似的算子来检测^[8]。

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} \quad (3)$$

如果用 $E_{i,j}^h, E_{i,j}^v$ 分别表示像素点 (i,j) 水平和垂直方向上的边界程度,用 $E_{i,j}^{d1}, E_{i,j}^{d2}$ 分别表示 45° 和 135° 方向上的边界程度, $f(i,j)$ 为像素点 (i,j) 上的灰度值。则:

$$E_{i,j}^h = f_{i-1,j+1} + 2f_{i,j+1} + f_{i+1,j+1} - f_{i-1,j-1} - 2f_{i,j-1} - f_{i+1,j-1} \quad (4)$$

$$E_{i,j}^v = f_{i+1,j-1} + 2f_{i+1,j} + f_{i+1,j+1} - f_{i-1,j-1} - 2f_{i-1,j} - f_{i-1,j+1} \quad (5)$$

$$E_{i,j}^{d1} = f_{i,j+1} + f_{i+1,j} + 2f_{i+1,j+1} - 2f_{i-1,j-1} - f_{i-1,j} - f_{i,j-1} \quad (6)$$

$$E_{i,j}^{d2} = f_{i-1,j} + 2f_{i-1,j+1} + f_{i,j+1} - f_{i-1,j-1} - 2f_{i+1,j-1} - f_{i+1,j} \quad (7)$$

而像素点 (i,j) 上总的边界程度就可以表示为:

$$E_{i,j} = |E_{i,j}^h| + |E_{i,j}^v| + |E_{i,j}^{d1}| + |E_{i,j}^{d2}| \quad (8)$$

如果 $E_{i,j}$ 相对于周围像素点的边界程度较大,则像素点 (i,j) 就是边界的凸出点,反之则是凹点。对于图1而言,若令白色的像素点灰度值为1,黑色的像素点灰度值为0,则通过式(8)检测出来的边界程度信息为:

$$\begin{matrix} 0 & 0 & 0 & 4 & 8 & 12 \\ 0 & 4 & 8 & 12 & 12 & 8 \\ 8 & 12 & 12 & 8 & 4 & 0 \\ 12 & 8 & 4 & 0 & 0 & 0 \end{matrix}$$

从图中可以看出图1中凸出点相对周围的边界程度都要大。因此在设计每个点上的加权系数时,应该使它是边界程度的递减函数。

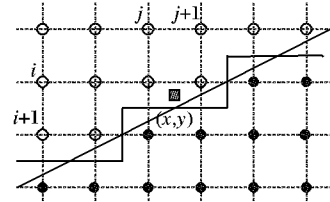


图2 抑制边缘锯齿现象示意图

这里首先需要设计一个边界程度信息的递减函数,然后将该函数的值与基于三角形的双线性插值结合起来。我们采用指数函数来作为边界程度信息的递减函数,如式(9)所示。

$$h(E_{i,j}) = \exp\{-\alpha E_{i,j}\} \quad (9)$$

其中 α 是可调参数。与基于三角形的双线性插值的结合是通过将该函数值与原来的权重相乘从而得到新的权重来完成。

对于图3待插值点 (x,y) 的情况,令:

$$W_{i+1,j+1} = (1-p-q)h(E_{i+1,j+1}) \quad (10)$$

$$W_{i,j+1} = ph(E_{i,j+1}) \quad (11)$$

$$W_{i+1,j} = qh(E_{i+1,j}) \quad (12)$$

则该点的估计值可表示为:

$$f(x,y) = \frac{W_{i+1,j+1}f(i+1,j+1) + W_{i,j+1}f(i,j+1) + W_{i+1,j}f(i+1,j)}{W_{i+1,j+1} + W_{i,j+1} + W_{i+1,j}} \quad (13)$$

其他待插值点的估计类似。

以图像插值在图像放大中的应用为例,设原始图像为 i ,放大后的图像为 j ,水平和垂直方向上的放大倍数分别为 w , h ,则整个算法流程可描述如下:

1) 利用式(2)、(3)中的算子检测原始图像的边界程度信息。

2) 对原始图像 i 中任意相邻的四个点组成的正方形,利用基于三角形的双线性插值中的方法确定其对角线的选取方向。

3) 对于图像 j 中的任意像素点 (m, n) ,根据式(14)确定其在原始图像中的映射点 (x, y) 。

$$\begin{cases} x = m/h \\ y = n/w \end{cases} \quad (14)$$

4) 根据该映射点所在的三角形区域利用式(13)计算其估计值。

3 实验结果与分析

在实验中,采用了广泛使用的 Cameraman 和 Lena 图像作为测试图像,原始图像的大小为 256×256 ,放大的倍数为 4×4 。算法中的参数 T_c 为 6, α 为 0.3。用 C 语言实现了双三次插值、EASE 插值、Triangulation 插值、NEDI 插值和本文基于边界程度的插值。实验结果如图 3~5 所示。

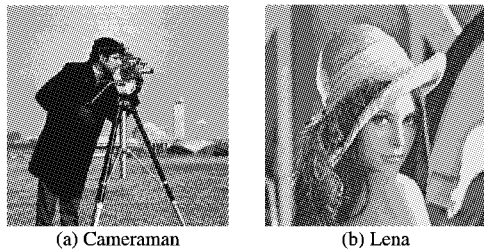


图3 原始图像



图4 Cameraman 放大 4×4 倍之后的局部效果

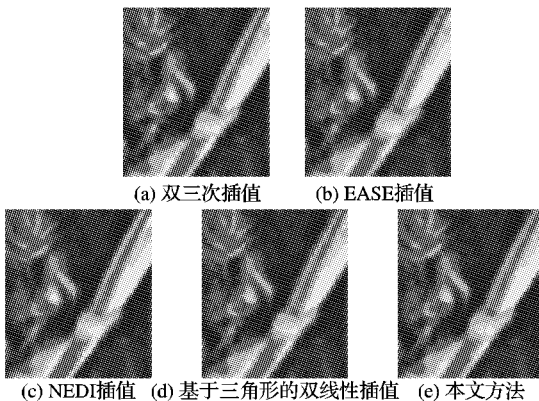


图5 Lena 放大 4×4 倍之后的局部效果

从图3~图5中可以看到,本文提出的方法在抑制图像放大中边缘出现的锯齿现象时优于双三次插值、EASE 插值、Triangulation 插值、NEDI 插值等插值方法,能够很好地消除边缘锯齿现象,同时也不会产生像 NEDI 插值算法中出现的鬼影或涡流似的人工痕迹。

表1对比了不同的图像插值算法执行时消耗的时间。从中可以看出,本文提出的方法慢于双三次插值和基于三角形的双线性插值。这主要是因为本文的算法插值时增加了加权系数归一化的操作和指数运算。总的来说,图像插值算法的复杂度和处理效果是一对矛盾,插值效果越好,算法自然也就越复杂,处理时间也就越长;反之,则插值效果较差。但本方法与 EASE 算法和 NEDI 算法相比,在速度上有明显的优势。此外,该算法从原理上来说比较简单,实现和推广起来也较为容易。

表1 图像插值算法消耗时间对比

插值算法	放大图像	
	Cameraman	Lena
双三次插值	0.30	0.29
EASE 插值	0.62	0.67
NEDI 插值	10.11	11.81
基于三角形的双线性插值	0.25	0.26
本文方法	0.53	0.54

4 结语

数字图像是通过采样的方式,在离散的坐标中表示连续的场景。当采样不满足奈奎斯特定理时,这种表示方式存在一定的失真。图像插值是采取一定的插值模型,来估计原始的或高分辨率下的真实场景。插值模型选择的恰当与否直接影响着最终的插值效果。本文针对图像插值中存在的边缘锯齿现象,考虑自然图像边缘形状平滑的特点,提出了基于边界程度的图像插值算法。该算法与其他基于边界的非线性图像插值算法的不同之处在于,它采用了边界程度信息而非边界方向信息。实验结果表明,该算法能够较好地解决锯齿现象问题,同时也不会产生其他明显的人工痕迹。此外,该算法原理也较为简单,便于实现和应用。

参考文献:

- [1] SHAO W Z, WEI Z H. Edge-and-corner preserving regularization for image interpolation and reconstruction[J]. Image and Vision Computing, 2008, 26(12): 1591-1606.
- [2] LEHMANN T M, GONNER C, SPITZER K. Survey: Interpolation methods in medical image processing[J]. IEEE Transactions on Medical Imaging, 1999, 18(11): 1049-1075.
- [3] LLEBACH J P, WONG P W. Edge-directed interpolation[C]// ICIP'1996: Proceedings of IEEE International Conference on Image Processing. Lausanne, Switzerland: IEEE, 1996: 707-710.
- [4] SU D, WILLIS P. Image interpolation by pixel level data-dependent triangulation[J]. Computer Graphics Forum, 2004, 23(2): 189-202.
- [5] GIACHETTI A, ASUNI N. Fast artifacts-free image interpolation[C]// Proceedings of British Machine Vision Conference 2008. Leeds, UK: BMVA, 2008: 123-133.
- [6] LI X, ORCHARD M T. New edge-directed interpolation[J]. IEEE Transactions on Image Processing, 2001, 10(10): 1521-1527.
- [7] CHA Y, SEONGJAI K. The error-amended sharp edge (EASE) scheme for image zooming[J]. IEEE Transactions on Image Processing, 2007, 16(6): 1496-1505.
- [8] BOVIK A. Handbook of image and video processing[M]. 2nd ed. Beijing: Publishing House of Electronics Industry, 2006.