边导向的双三次彩色图像插值

周登文 [1] 申晓留 [1]

摘 要 提出了一个新的边导向的双三次卷积 (Cubic convolution, CC) 彩色图像插值算法. 对于待插值的像素, 首先在其邻域检测两个正交方向边的强度. 如果该像素在一个强边上, 则沿着强边的方向执行 CC 插值估计该像素; 否则, 该像素在弱边或纹理区域, 通过加权平均两个正交方向的 CC 插值估计该像素. 本文方法也考虑了彩色平面之间的相关性. 实验结果显示, 本文方法显著优于经典的 CC 插值和其他一些插值方法.

关键词 彩色图像处理, 边导向, 双三次卷积, 图像插值, 图像放大

引用格式 周登文, 申晓留. 边导向的双三次彩色图像插值. 自动化学报, 2012, **38**(4): 525−530

DOI 10.3724/SP.J.1004.2012.00525

Edge-directed Bicubic Color Image Interpolation

ZHOU Deng-Wen[1] SHEN Xiao-Liu[1]

Abstract Image interpolation becomes very important because various low-cost digital cameras and camcorders have spread widely. Most of the existing image interpolation methods are geared to gray images. Interpolating a color image is usually implemented by the componentwise interpolation scheme using gray image interpolation methods, which is insufficient and results in additional artifacts. In this paper, a novel edge-directed bicubic color image interpolation method is proposed. Our method is an extension to the classical bicubic convolution interpolation. Our improvement includes: considering the correlation between color channels; giving a new method of color edge detection and a novel scheme estimating the missing pixels. The proposed method can well preserve the edges and textures, and noticeably suppress common artifacts such as blurring, blocking and aliasing in interpolated images. The experimental results demonstrate that it outperforms significantly the bicubic interpolation in terms of both objective and subjective measures.

Key words Color image processing, edge-directed, cubic convolution (CC), image interpolation, image zooming

Citation Zhou Deng-Wen, Shen Xiao-Liu. Edge-directed bicubic color image interpolation. *Acta Automatica Sinica*, 2012, **38**(4): 525−530

虽然图像采集设备的分辨率稳定提高, 但是新的应用常常需要有更高空间分辨率的图像. 图像插值的目标就是增加数字图像的空间分辨率, 它已广泛应用于医学成像、遥感及大幅广告等. 经典的图像插值方法包括: 双线性、双三次卷积[1] 和双三次样条插值[2] 等方法. 这些方法在图像光滑的区域具有很好的效果, 但是, 在图像边缘的区域会产生模糊 (Blurring)、分块 (Blocking) 和振铃 (Ringing) 等缺陷.

边导向的图像插值方法[3−11] 是当前图像插值研究的热点问题. 这些算法试图插值丢失的像素沿着而不是通过边的方向. 在一个高分辨率 (High resolution, HR) 的图像中, 对于丢失像素的插值, 其关键步骤是通过已知的低分辨率 (Low resolution, LR) 图像, 显式或隐式估计对应的 HR 图像中, 丢失像素位置边的方向. 边导向的插值方法比经典的插值方法有更高的计算复杂度, 但是用它们生成的 HR 图像有更清晰的边和更好的视觉质量.

到目前为止, 大多数的插值算法仅适用于灰度图像, 直接操作在彩色图像上的算法[12−14] 很少. 其原因可能是彩色图像插值能够通过灰度图像插值算法以逐平面插值的方式实现. 也就是说, 彩色图像包含红、绿和蓝三个颜色平面, 将灰度图像插值算法分别应用到这三个颜色平面, 可获得插值的彩色 HR 图像. 事实上, 这并不实用. 原因在于三个颜色平面中边的方向并不总是一致的. 另外, 三个颜色平面有很强的相关性[15], 逐平面的插值没有利用这种相关性, 由此会产生附加的缺陷. Lukac 等[12] 提出了一个彩色图像插值的向量滤波方法 (Vector filter, VF), VF 使用的滤波器包括向量中值滤波器 (Vector median filter, VMF)、基本的向量方向滤波器 (Basic vector direction filter, BVDF) 和数据

收稿日期 2011-07-05 录用日期 2011-09-22
Manuscript received July 5, 2011; accepted September 22, 2011
北京市支持中央在京高校共建项目 ("首都高校重点科研平台大型仪器设备共享管理信息系统"), 北京市自然科学基金 (4112061) 资助
Supported by Beijing-funded Joint Development Program of the Central Universities in Beijing ("the Sharing Management Information System of the Large Instruments of the Key Scientific Research Platforms of the Capital Universities") and Beijing Natural Science Foundation (4112061)
本文责任编委 刘一军
Recommended by Associate Editor LIU Yi-Jun
1. 华北电力大学控制与计算机工程学院 北京 102206
1. School of Control and Computer Engineering, North China Electric Power University, Beijing 102206

自适应的向量滤波器 (Vector data-adaptive filter, VDAF). Lukac 等的实验结果也显示 VDAF 优于 VMF 和 BVDF. VF 方法涉及到在一个邻域内计算聚集的 Minkowski 距离或角度, Jin 等[13] 改进了 VF 方法, 他们把 Minkowski 距离或角度结合在一起同时计算. Cha 等[14] 则是为标准的插值方法建议了一个后处理的模式, 以减轻逐平面插值而引起的缺陷.

本文提出了一个新的边导向的图像插值算法, 且直接应用于彩色图像. 我们也改进了现存的边检测模式. 边导向插值算法的性能很大程度上依赖于边检测的精度, 其主要问题是: 对边的检测不够准确, 尤其是在具有复杂边 (如纹理) 区域更是如此. 本文算法能够可靠地检测强边, 然后在检测的强边上执行双三次卷积 (Cubic convolution, CC) 插值, 而在弱边或纹理区域, 则加权平均两个正交方向的 CC 插值. 实验显示了本算法的优越: 与其他代表性方法相比, 能够很好地适应彩色图像的边结构, 具有更好的主观和客观测度.

1 插值算法

1.1 彩色图像边检测

在灰度图像中, 每个像素有一个数量灰度, 边是像素灰度显著变化的位置. 在彩色图像中, 每一个像素是一个向量, 包含红、绿和蓝三个颜色分量, 边是像素色度、亮度或两者同时显著变化的位置. 两个相邻像素向量 $\boldsymbol{I}_1(k)$ 和 $\boldsymbol{I}_2(k)$ 的变化可通过 Minkowski 距离来度量[12]:

$$d = \|\boldsymbol{I}_1 - \boldsymbol{I}_2\|_L = \left(\sum_{i=1}^{3} \|\boldsymbol{I}_1(k) - \boldsymbol{I}_2(k)\|^L\right)^{\frac{1}{L}} \quad (1)$$

$k = 1, 2, 3$, 分别表示红、绿和蓝三个颜色分量; L 是范数参数: $L = 1$ 是城市块 (City-block) 距离, $L = 2$ 是欧几里德 (Euclidean) 距离. 相邻像素向量的变化也可通过方向来度量[12, 16]. 两个彩色像素向量 $\boldsymbol{I}_1(k)$ 和 $\boldsymbol{I}_2(k)$ 之间的角度为

$$\alpha = \cos^{-1}\left(\frac{\boldsymbol{I}_1 \cdot \boldsymbol{I}_2}{|\boldsymbol{I}_1| \cdot |\boldsymbol{I}_2|}\right) \quad (2)$$

显然, 计算欧几里德距离和角度比计算城市块距离复杂得多. 本文使用城市块距离, 即式 (1) 中 $L = 1$, 测度两个相邻像素向量的变化. 也即:

$$d = \|\boldsymbol{I}_1 - \boldsymbol{I}_2\|_1 = \sum_{i=1}^{3} |\boldsymbol{I}_1(k) - \boldsymbol{I}_2(k)| \quad (3)$$

1.2 色度 − 亮度空间

彩色监视器常常使用三个平面高度相关的 RGB 颜色空间[15]. 在不同的图像处理任务中, 常常需要利用或去除这种相关性. 例如, 几乎所有的图像去马赛克 (Demosaicking) 算法利用这种相关性, 从一个颜色平面估计其他颜色平面的高频信息[15]; 而图像去噪算法利用去相关的 YUV/YCbCr 颜色空间改进去噪性能[17]. 有些研究工作[18] 表明: 在 YUV/YCbCr 颜色空间, 边检测效果更好, 本文插值选择在去相关的 YCbCr 颜色空间进行, 然后, 将插值的 HR 图像变换至 RGB 颜色空间. 从 RGB 到 YCbCr 空间的变换公式为[19]

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.000 \\ 112.000 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (4)$$

1.3 方向双三次卷积插值

假定 I_h 是待插值的 HR 图像, 其尺寸为 $2M \times 2N$, 对应的 LR 图像 I_l 是对 I_h 以因子 2 向下取样的版本, 尺寸为 $M \times N$. I_h 与 I_l 的关系为 $I_l(i, j) = I_h(2i - 1, 2j - 1), 1 \le i \le M, 1 \le j \le N$. 如果 $M = N = 6$, 则 I_l 和 I_h 如图 1 所示, I_h 中 3/4 的像素需要估计, 其中, 实心圆点是已知的 LR 像素, 空心圆点是丢失的 HR 像素.

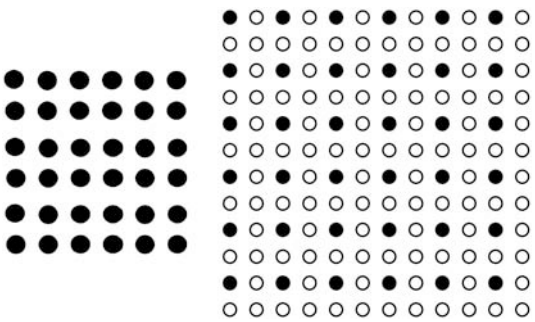


图 1 原 LR 图像 (左) 和 LR 图像向上取样的版本 (右) (其中实心圆点是已知的 LR 像素, 空心圆点是丢失的 HR 像素)

Fig. 1 The original LR image (left) and its upsampled version (right) (The solid dots are the known LR image pixels and the circles are the missing HR image pixels.)

边导向插值方法的一个关键步骤是要检测丢失像素所在边的方向, 然后, 插值运算沿着边的方向进行. 在一个连续的模拟图像中, 任意一点所在边的方向是任意的. 但对一个离散的数字图像, 任意一个像素所在的 3×3 的邻域, 只有 4 个方向, 即水平、垂直、45° 对角和 135° 对角方向. 所以, 本文算法只考虑这 4 个方向的边, 但在计算边的方向时, 使用

了 7×7 或 5×5 的更大的邻域. 另外, 丢失像素的插值在两个通道中进行. 第 1 个通道插值丢失的像素为 $I_h(2i,\,2j)$; 第 2 个通道插值其余丢失的像素为 $I_h(2i-1,\,2j)$ 和 $I_h(2i,\,2j-1)$.

1.3.1　第 1 个通道插值

如图 2, 对待插值的中心像素, 考虑其 7×7 的邻域, 分别通过计算在 45° 和 135° 方向聚集的城市块距离, 以确定这两个方向边的强度. 对于所在的计算方向, 聚集距离值越小, 则边的强度也越大, 反之边的强度越小. 45° 方向聚集距离为

$$d_1=\sum_{m=3,\pm1}\sum_{n=3,\pm1}|I_h(2i+m,2j-n)-I_h(2i+m-2,2j-n+2)| \tag{5}$$

135° 方向聚集距离为

$$d_2=\sum_{m=3,\pm1}\sum_{n=3,\pm1}|I_h(2i+m,2j+n)-I_h(2i+m-2,2j+n-2)| \tag{6}$$

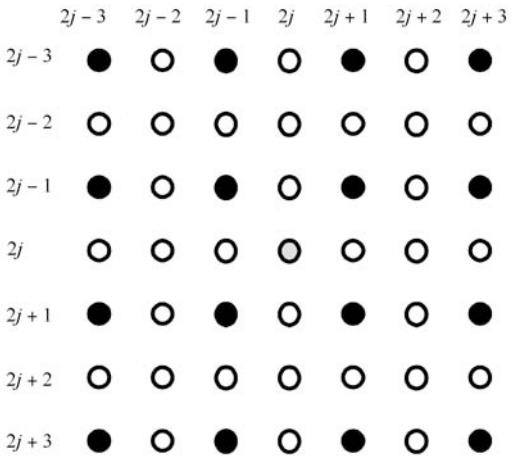


图 2　第一通道插值 (其中实心圆点是已知的 LR 像素, 空心圆点是丢失的 HR 像素)

Fig. 2　Interpolation in the first pass (The solid dots are the known LR image pixels and the circles are the missing HR image pixels.)

这两个正交方向聚集距离的比值用于确定边的方向, 即:

If $(1+d_1)/(1+d_2)>T$
　像素 $I_h(2i,2j)$ 是在 135° 强边上
Else if $(1+d_2)/(1+d_1)>T$
　像素 $I_h(2i,2j)$ 是在 45° 强边上
Else
　像素 $I_h(2i,2j)$ 是在弱边或纹理区域
End

T 是阈值, 根据经验确定; 距离加 1 是为了避免被零除. 这个决策的合理性在于: 在某个邻域, 如果存在某个方向强边, 在该方向上的距离会更小.

对于 2 倍插值, 双三次卷积插值滤波器为 $\boldsymbol{f}=[-1,9,9,-1]/16$[1]. 45° 和 135° 方向的 2 个样本向量分别为: $\boldsymbol{v}_1=[I_h(2i-3,2j+3),\ I_h(2i-1,2j+1),\ I_h(2i+1,2j-1),\ I_h(2i+3,2j-3)]$, $\boldsymbol{v}_2=[I_h(2i-3,2j-3),\ I_h(2i-1,2j-1),\ I_h(2i+1,2j+1),\ I_h(2i+3,2j+3)]$. 45° 方向的 CC 插值为

$$p_1=\boldsymbol{f}\cdot\boldsymbol{v}_1^{\mathrm{T}} \tag{7}$$

类似地, 135° 方向的 CC 插值为

$$p_2=\boldsymbol{f}\cdot\boldsymbol{v}_2^{\mathrm{T}} \tag{8}$$

$\boldsymbol{x}^{\mathrm{T}}$ 表示向量 $\boldsymbol{x}$ 的转置. 假定用 p 代表 $I_h(2i,2j)$, 若在 45° 强边上, 取 $p=p_1$; 若在 135° 强边上, 取 $p=p_2$; 否则, 插值 p 估计为

$$p=\frac{w_1p_1+w_2p_2}{w_1+w_2} \tag{9}$$

权 w_1 和 w_2 计算为

$$\begin{cases} w_1=\dfrac{1}{1+d_1^r} \\ w_2=\dfrac{1}{1+d_2^r} \end{cases} \tag{10}$$

参数 r 调整加权效果, 可以根据经验确定. 丢失像素 $I_h(2i,2j)$ 的估计可以总结为

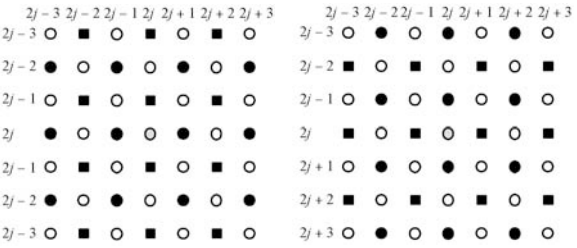
$$\begin{cases} \text{If } \dfrac{1+d_1}{1+d_2}>T \\ \quad p=p_2; \\ \text{Else if } \dfrac{1+d_2}{1+d_1}>T \\ \quad p=p_1; \\ \text{Else} \\ \quad \text{用式 (9) 计算 } p \text{ 及用式 (10) 计算权} \\ \text{End} \end{cases} \tag{11}$$

式 (11) 表明: 如果丢失的像素是在一个强边上, 则插值沿着检测的强边的方向; 如果丢失的像素是在一个弱边或纹理区域, 通过加权的方式插值, 利用权值隐含地补偿边的方向, 避免了其他边导向方法遇到弱边检测不准的问题.

1.3.2　第 2 个通道插值

估计出丢失的 HR 像素 $I_h(2i,2j)$ 后, 可以有助于估计其余丢失的像素 $I_h(2i-1,2j)$ 和 $I_h(2i,2j-$

1), 参见图 3 (a) 和图 3 (b), 黑色方块是第 1 通道中估计的像素. 需要注意的是: 为了方便, 图 3 中被估计的像素均被平移到邻域中心 $(2i, 2j)$.



(a) 丢失的像素 $I_h(2i-1,2j)$ 的插值 (b) 丢失的像素 $I_h(2i,2j-1)$ 的插值

(a) Interpolating the missing pixel $I_h(2i-1,2j)$ (b) Interpolating the missing pixel $I_h(2i,2j-1)$

图 3 第 2 通道插值 (其中实心圆点是已知的 LR 像素, 空心圆点是丢失的 HR 像素, 黑色方块是已经估计的像素)

Fig. 3 Interpolation in the second pass (The solid dots are the known LR image pixels. The black squares are the estimated pixels in the first pass.)

第 2 通道两种情形的插值过程是完全相同的, 现在考虑图 3 (a) 情形. 在被插值的中心像素 5×5 的邻域内, 计算水平和垂直方向的聚集距离. 水平方向聚集距离为

$$\begin{aligned} d_1 = & \sum_{m=\pm 1} \sum_{n=0,2} |I_h(2i+m, 2j-n) - \\ & I_h(2i+m, 2j-n+2)| + \\ & \sum_{m=0,\pm 2} |I_h(2i+m, 2j-1) - \\ & I_h(2i+m, 2j+1)| \end{aligned} \tag{12}$$

类似地, 垂直方向的聚集距离为

$$\begin{aligned} d_2 = & \sum_{m=0,2} \sum_{n=\pm 1} |I_h(2i-m, 2j+n) - \\ & I_h(2i-m+2, 2j+n)| + \\ & \sum_{n=0,\pm 2} |I_h(2i-1, 2j+n) - \\ & I_h(2i+1, 2j+n)|) \end{aligned} \tag{13}$$

水平方向和垂直方向插值样本点向量分别为 $\boldsymbol{v}_1 = [I_h(2i, 2j-3),\ I_h(2i, 2j-1),\ I_h(2i, 2j+1),\ I_h(2i, 2j+3)]$, $\boldsymbol{v}_2 = [I_h(2i-3, 2j),\ I_h(2i-1, 2j),\ I_h(2i+1, 2j),\ I_h(2i+3, 2j)]$, 应用式 (7) ∼ (11) 可以估计丢失的中心像素.

1.4 算法过程描述

2 倍插值的算法步骤总结如下.

步骤 1. 转换在 RGB 空间的 LR 图像到 YCbCr 空间;

步骤 2. 展开已知的 $M \times N$ LR 图像 I_l (如图 1 左) 在 $2M \times 2N$ HR 图像 I_h 对应的网格上 (如图 1 右);

步骤 3. 执行第 1 通道插值. 用式 (7) ∼ (11) 估计丢失的像素 $I_h(2x, 2y)$ $(x = 2, 3, \cdots, M-2, y = 2, 3, \cdots, N-2)$, 如图 2;

步骤 4. 执行第 2 通道情形 1 插值. 与步骤 3 相似, 丢失的像素 $I_h(2x-1, 2y)$ $(x = 4, 5, \cdots, M-4, y = 2, 3, \cdots, N-2)$ 被估计, 如图 3 (a);

步骤 5. 执行第 2 通道情形 2 插值. 与步骤 4 相似, 丢失的像素 $I_h(2x, 2y-1)(x = 2, 3, \cdots, M-2, y = 4, 5, \cdots, N-4)$ 被估计, 如图 3 (b);

步骤 6. 填充丢失的边界像素;

步骤 7. 将插值的 HR 图像 I_h 转换到 RGB 颜色空间.

上面的算法适用于图像 2 倍放大, 如果放大因子为 2 的整数次幂, 可以通过迭代使用上面的算法来实现. 提醒读者注意: 利用上述算法, 执行每个丢失像素的插值, 均包含三个颜色分量.

2 实验结果

本文方法与当前 4 个代表性算法进行了比较, 分别是: 标准的双线性 (Bilinear interpolation, BI) 和双三次卷积 (CC)[1] 插值、边导向的 ICBI (Iterative curvature based interpolation)[11] 插值和 CVOSI (Combining vector ordering and spatial information)[13] 彩色插值. 由于 Jin 等[13] 已证实 CVOSI 优于 Lukac 等[12] 建议的 VF (Vector filter) 方法, 因此无需同 VF 方法进行比较. 前三个方法是为灰度图像开发的, 放大彩色图像通过插值每一个颜色平面实现. CVOSI 直接操作在彩色图像上. BI 和 CC 通过 Matlab 的 INTERP2 函数实现[19], ICBI 的 Matlab 源码是由原作者 Giachetti 提供的, 我们自己实现了 CVOSI 算法. RGB 和 YCbCr 颜色空间的转换也是由 Matlab 的内部函数 RGB2YCBCR 实现的. 在本文算法中, 只有两个参数需要设置, 即式 (10) 中参数 r 和式 (11) 中阈值 T. 在所有的实验中, r 设置为 7, T 设置为 1.2. 为了公平地评估每一个算法的性能, 我们选择了 8 个测试图像, 如图 4, 它们有一个很宽的颜色域和自然纹理. LR 图像从原 HR 图像以因子 2 向下取样得到, 然后, 用不同的方法重构 HR 图像. 由于原 HR 图像是已知的, 重构的 HR 图像的彩色峰值信噪比 (Com-

posite peak signal-to-noise ratio, CPSNR) 能够被计算. 计算中排除了边界 12 个像素, 以去除边界的影响. CPSNR 计算公式如下:

$$CPSNR = 10\lg\left(\frac{255^2}{CMSE}\right) \tag{14}$$

其中

$$CMSE = \frac{1}{3HW}\sum_{k=1}^{3}\sum_{i=1}^{H}\sum_{j=1}^{W}(I_o(i,j,k) - I_r(i,j,k))^2 \tag{15}$$

I_o 和 I_r 分别表示原 HR 图像与重构的 HR 图像, 尺寸为 $H \times W$; k 与式 (1) 含义相同. 表 1 列出了 5 种方法 2 倍放大的结果.

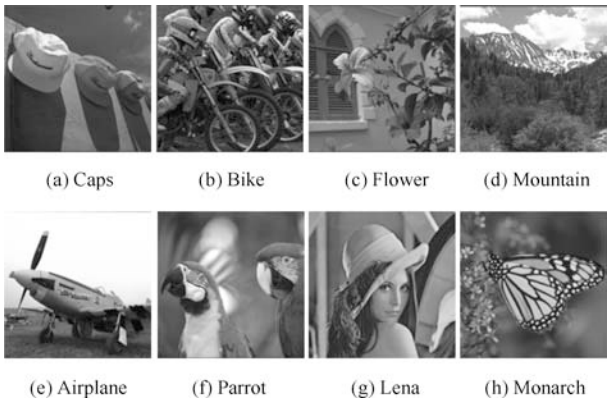


图 4　测试图像

Fig. 4　Set of testing images

表 1　不同算法 2 倍放大的 CPSNR (dB) 比较

Table 1　Comparison of CPSNR (dB) among different algorithms for 2× zoom

图像	BI	CC	ICBI	CVOSI	本文方法
Caps	33.08	33.12	32.99	32.99	**33.99**
Bike	25.26	25.50	25.17	25.03	**26.47**
Flower	31.20	31.66	31.71	30.96	**32.56**
Mountain	23.58	23.48	22.71	23.55	**23.71**
Airplane	29.62	29.67	29.59	29.54	**30.66**
Parrot	32.68	33.25	33.07	32.25	**34.20**
Lena	32.07	32.30	32.19	31.96	**32.89**
Monarch	29.43	30.04	30.61	29.17	**31.22**
Average	29.61	29.88	29.75	29.43	**30.71**

对于所有测试图像, 本文方法一致胜出了其他方法. 平均 CPSNR 值超出了排名第二的 CC 方法 0.83 dB. 对于有显著边和丰富纹理图像的 Bike, Airplane 和 Parrot, 本文方法超出其他最好结果近 1 dB. 限于篇幅, 仅给出 Monarch 图像的视觉结果, 如图 5. 这是一个 4 倍放大的 Monarch 图像, LR 图像是以因子 4 向下取样原 HR 图像得到的. 可以看到: 由于 BI 和 CC 不是沿着边插值, 产生了最严重的分块缺陷 (如图 5 (b) 和 5 (c)); CVOSI 用邻域像素加权平均的方法估计丢失的像素, 产生了相当模糊的图像 (如图 5 (e)). ICBI 保持边的能力仅次于本文的方法, 但是重构的 HR 图像中出现了许多虚假的颜色 (如图 5 (d)), 这是由于不同颜色平面边的方向不一致引起的. 只有本文方法清晰地保持了边和纹理结构及产生的走样最小 (如图 5 (f)).

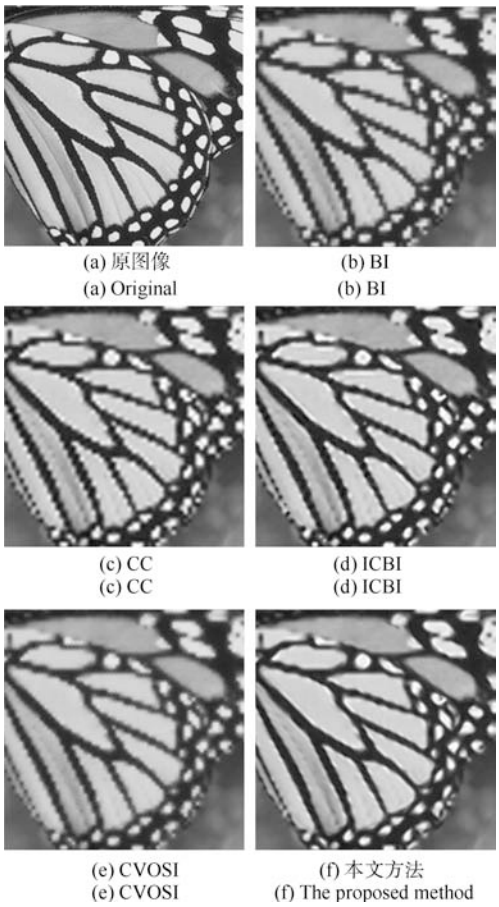


图 5　4 倍放大的 Monarch 图像

Fig. 5　Parts of the 4× zoomed image Monarch

3　结论

当前大多数图像插值算法是为灰度图像开发的, 直接应用于彩色图像的算法很少. 在本文中, 我们提出了一个新的适用于彩色图像的插值算法, 并给出了一个简单有效的彩色边检测算法. 如果丢失的像素在一个强边上, 我们沿着边的方向执行双三次卷积插值; 如果丢失的像素在弱边或纹理区域, 则通过加权平均两个正交方向上的双三次卷积插值, 估计丢失的像素. 本文方法还考虑了彩色平面之间的相关性, 插值在去相关的颜色空间中进行. 本文方法显著地超出了经典的双三次卷积插值, 也显著地胜出了其他一些代表性的方法.

References

1 Keys R G. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1981, **29**(6): 1153−1160

2 Hou H S, Andrews H C. Cubic splines for image interpolation and digital filtering. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1978, **26**(6): 508−517

3 Allebach J, Wong P W. Edge-directed interpolation. In: Proceedings of the IEEE International Conference on Image Processing. Lausanne, Switzerland: IEEE, 1996. 707−710

4 Li X, Orchard M T. New edge-directed interpolation. *IEEE Transactions on Image Processing*, 2001, **10**(10): 1521−1527

5 Zhou D W. An edge-directed bicubic interpolation algorithm. In: Proceedings of the 3rd International Congress on Image and Signal Processing. Yantai, China: IEEE, 2010. 1186−1189

6 Mishiba K, Ikehara M. Image interpolation using edge-directed smoothness measure filter. *IEICE Transactions on Information and Systems*, 2010, **93**(6): 1618−1624

7 Tam W S, Kok C W, Siu W C. Modified edge-directed interpolation for images. *Journal of Electronic Imaging*, 2010, **19**(1): 013011-1−013011-20

8 Wang X Y, Chen Z F. An improved image interpolation algorithm. *Multidimensional Systems and Signal Processing*, 2009, **20**(4): 385−396

9 Xiong Z, Zhang Y, Sun X, Wu F. Fast directional image interpolation with difference projection. In: Proceedings of the IEEE International Conference on Multimedia and Expo. New York, USA: IEEE, 2009. 81−84

10 Zhang Xiong, Bi Du-Yan, Yang Bao-Qiang. An edge preserved image interpolation method. *Journal of Air Force Engineering University (Natural Science)*, 2007, **8**(3): 78−80 (张雄, 毕笃彦, 杨宝强. 一种保持图像边缘的插值方法. 空军工程大学学报 (自然科学版), 2007, **8**(3): 78−80)

11 Giachetti A, Asuni N. Fast artifacts-free image interpolation. In: Proceedings of the British Machine Vision Conference. Leeds, UK: BMVC, 2008. 123−132

12 Lukac R, Plataniotis N, Smolka B, Venetsanopoulos N. Vector operators for color image zooming. In: Proceedings of the IEEE International Symposium on Industrial Electronics. Dubrovnik, Croatia: IEEE, 2005. 1273−1277

13 Jin L, Li D, Song E. Combining vector ordering and spatial information for color image interpolation. *Image and Vision Computing*, 2009, **27**(4): 410−416

14 Cha Y, Kim S. Edge-forming methods for color image zooming. *IEEE Transactions on Image Processing*, 2006, **15**(8): 2315−2323

15 Gunturk B K, Altunbasak Y, Mersereau R M. Color plane interpolation using alternating projections. *IEEE Transactions on Image Processing*, 2002, **11**(9): 997−1013

16 Trahanias P E, Karakos D, Venetsanopoulos A N. Directional processing of color images: theory and experimental results. *IEEE Transactions on Image Processing*, 1996, **5**(6): 868−880

17 Foi A, Katkovnik V, Egiazarian K. Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images. *IEEE Transactions on Image Processing*, 2007, **16**(5): 1395−1411

18 Kumar P, Sengupta K, Lee A. A comparative study of different color spaces for foreground and shadow detection for traffic monitoring system. In: Proceedings of the 5th IEEE International Conference on Intelligent Transportation Systems. Singapore, Singapore: IEEE, 2002. 100−105

19 MathWorks. *Image Processing Toolbox User's Guide*. Natick: MathWorks, 2009

周登文 华北电力大学控制与计算机工程学院副教授. 主要研究方向为图像插值、降噪和分类. 本文通信作者.
E-mail: zdw@ncepu.edu.cn
(**ZHOU Deng-Wen** Associate professor at the School of Control and Computer Engineering, North China Electric Power University. His research interest covers image interpolation, denoising, and classification. Corresponding author of this paper.)

申晓留 华北电力大学控制与计算机工程学院教授. 主要研究方向为图像处理和地理信息系统.
E-mail: shenxiaoliu@163.com
(**SHEN Xiao-Liu** Professor at the School of Control and Computer Engineering, North China Electric Power University. His research interest covers image processing and geographic information system.)