

一种改进的快速步进图像修复算法

胡超, 吴骅, 屈磊, 韦穗

(安徽大学 计算智能与信号处理教育部重点实验室, 安徽合肥 230039)

摘要: 图像修复是数字图像处理的重要内容, 可用于恢复图像中小的破损区域、文字去除以及目标物体隐藏。基于水平集应用的快进修复算法可以简单快速且有效地修复数字图像中的破损区域, 但对边缘的保持能力不够。针对这一问题提出了改进方案。利用梯度排序来保持图像内部的边缘, 实验结果也表明改进后的修复结果要优于原算法。

关键词: 梯度排序; 快速步进; 图像修复

中图分类号: TP301.6

文献标识码: A

文章编号: 1673-629X(2007)05-0206-04

An Image Inpainting Algorithm Based on Improved Fast Marching Method

HU Chao, WU Hua, QU Lei, WEI Sui

(Ministry of Education Key Laboratory of IC & SP, Anhui University, Hefei 230039, China)

Abstract: Image inpainting is an important research field in digital image processing, which serves a wide range of applications, such as recovering the damaged regions, removing text and hiding some objects. The fast marching method based on the level set application can inpaint the damaged regions simply, quickly and effectively, but weak to preserve the edge information. Propose an improved algorithm, which uses the gradient-sorting estimation to preserve the edge, the results of experiments have shown the improved one is better.

Key words: gradient-sorting estimation; fast marching; image inpainting

0 前言

数字修复有广泛的应用, 例如从静止图像和视频中移去文字和标语, 通过移去划痕或污点重构变质的图像或创造艺术效果, 近年来得到了相当的关注。现有的大部分数字图像修复方法采用的主要思想为: 首先手工选定待修复区域; 然后利用该待修复区域周围已知色彩信息沿边界向内进行推进修复, 为了保证良好的图像修复效果和图像的整体性, 在进行推进修复的过程中应努力保持等照度线的平滑过渡。

许多修复方法建立在上述的思想, 如 Bertalmio 提出的基于偏微分方程(PDE)的方法^[1,2], Chan 提出的整体变分(TV)模型^[3]使用 Euler-Lagrange 方程和各向异性扩散来维持等值线方向。曲率驱动扩散(CDD)模型^[4]加强了 TV 模型沿等值线方向驱动扩散, 因此能修复更宽的区域。上述方法的本质都是在解一个描述信息扩散的偏微分方程(PDE), 然而在实

践中主要的问题是等值线估计和信息传输都受限于数值扩散。扩散可以保证 PDE 的稳定求解, 但不可避免地导致修复区域模糊, 而且 PDE 的求解需要大量的迭代运算以致修复速度很慢。Telea 提出了一种新的算法, 将破损区域看成水平集^[5]并用快速行进法(FMM)来传递图像信息^[6], 具有较快的修复速度。该算法首先利用快进算法选择修复路径, 并使用方向、距离和水平集三个权值对邻域像素进行加权平均, 然后沿着等照度线进行平滑估计。由于其邻域像素权值和梯度分散, 所以使用加权平均后对于边缘保持效果不理想。

针对快进修复方法提出了改进方案, 即利用梯度排序来估计等照度线方向, 然后沿着等照度线对待修复区域进行图像信息传递, 这样便可以保证等照度线尽可能平滑, 图像的边缘也能得到较好的保持。文中最后的实验也显示了改进方法有着较好的修复能力。

1 快进修复算法(FMM)原理

1.1 数学模型

记 Ω 为图像中的待修复区域, $\partial\Omega$ 为该区域的边界。假定 p 为 $\partial\Omega$ 上的一点。围绕 p 取一小的邻域 $B_\epsilon(p)$, p 点的修复应该由其邻域像素即 $B_\epsilon(p)$ 来决

收稿日期: 2006-08-05

作者简介: 胡超(1982-), 男, 安徽庐江人, 硕士研究生, 主要研究方向为图像处理; 韦穗, 教授, 博导, 主要研究方向为计算机视觉、图像处理。

定。 $B_\epsilon(p)$ 区域中的已知点 q 可以按照下式来对点 p 进行一阶逼近:

$$I_q(p) = I(q) + \nabla I(q)(p - q) \quad (1)$$

然后把修复点 p 看成 $B_\epsilon(p)$ 中所有点 q 的函数,并用一个归一化的加权函数 $w(p, q)$ 进行加权累计,即:

$$\frac{\sum_{q \in I_k(p)} w(p, q) [I(q) + \nabla I(q)(p - q)]}{\sum_{q \in I_k(p)} w(p, q)} \quad (2)$$

这里的加权函数 $w(p, q)$ 既要能传输图像灰度值还要能保持 $B_\epsilon(p)$ 中的边缘细节。

1.2 利用快进方法(FMM)来修复

1.1节所叙述的过程可以完成 $\partial\Omega$ 上某个点的修复,要修复整个 Ω ,需要迭代地把方程(2)应用到 $\partial\Omega$ 上的所有离散点,并不断增加离初始边界 $\partial\Omega$ 的距离,向 Ω 内部推进其边界,直到所有的区域被修复。这里采用水平集函数演化中的快进算法^[7]。

简单地说,快进算法就是对待修复区域内部的点解如下 Eikonal 方程:

$$|\nabla T| = 1 \quad (3)$$

其中 T 是待修复 Ω 中的点到边缘 $\partial\Omega$ 的距离。在应用快进算法之前需要对 T 进行初始化并事先建立一个符号表:

- * ACTIVE:即将被处理的像素。其 T 值正在更新。
- * ALIVE: $\partial\Omega$ 外的像素(属于已知图像)。其 T 值和像素值已知。
- * INSIDE: $\partial\Omega$ 内的像素(属于未知图像)。其 T 值和像素值未知。

利用逆向差分法(Upwind),方程(3)的稳定解可由以下方程得到:

$$[\max(D_{i,j}^- T, 0)^2 + \min(D_{i,j}^+ T, 0)^2 + \max(D_{i,j}^- T, 0)^2 + \min(D_{i,j}^+ T, 0)^2]^{\frac{1}{2}} = 1 \quad (4)$$

式中: D^- 和 D^+ 分别是后向与前向差分,比如 $D_{i,j}^- T = T_{i,j} - T_{i-1,j}$, $D_{i,j}^+ T = T_{i,j+1} - T_{i,j}$ 。

解上述方程实际上就是求出 Ω 内部所有点到的 $\partial\Omega$ 距离 T ,然后按照 T 由小到大的顺序选择路径进行修复工作。FMM 保证了初始边界 $\partial\Omega$ 最近的点最先被填充,即总是最先修复离已知图像最近的像素点,直到待修复区域全部被处理。

1.3 修复单个像素点

由于将待修复的某个点看成其邻域 $B_\epsilon(p)$ 中所有像素点的函数,所以可以对 $B_\epsilon(p)$ 中已标记为 ALIVE 点按照式(2)进行迭代。加权函数 $w(p, q)$ 向待修复区域传递图像边缘细节以及其平滑区域,它在修复过程中起到关键作用,Telea 设计的加权函数

$w(p, q)$ 为:

$$w(p, q) = \text{dir}(p, q) \cdot \text{dst}(p, q) \cdot \text{lev}(p, q) \quad (5)$$

它是以下三个因子的乘积:

$$\text{dir}(p, q) = \frac{p - q}{\|p - q\|} \cdot N(p), \text{dst}(p, q) = 1 / \|p - q\|^2, \text{lev}(p, q) = 1 / (1 + |T(p) - T(q)|)$$

其中方向因子 $\text{dir}(p, q)$ 保证了 FMM 的信息传递方向,即接近法线方向 $N = \nabla T$ 的像素的贡献要高于远离 N 方向的像素。几何距离因子 $\text{dst}(p, q)$ 减少了离 p 较远的像素的贡献。水平集因子 $\text{lev}(p, q)$ 保证了接近通过 p 点的等照度线的像素比远离等照度线的像素贡献大。 p 点像素的梯度 ∇I 由中心差分来估计。

但是在实验中发现,由于其邻域像素权值和梯度分散,所以使用加权平均后对于边缘保持效果不理想。针对这种情况,提出了改进方案。首先取以 p 点为中心、大小为 $\epsilon * \epsilon$ 的区域作为其邻域 $B_\epsilon(p)$,然后在该邻域中计算该点的梯度,并取前 ϵ 个最大梯度的平均作为该点梯度。其次,为了加强图像信息沿着等照度线方向传输,进行了梯度排序估计。

这里用 $\nabla I(p)$ 表示待修复 p 点的梯度方向,并令 $\nabla I^\perp(p)$ 为估计的 p 点的等照度线方向, $\overrightarrow{p - q}$ 表示的是归一化后的点 p 到 q 的方向。可以用下式来估计两个方向的接近程度:

$$E(p, q) = \nabla I^\perp(p) \cdot \overrightarrow{p - q} \quad (6)$$

可以看出,当点 q 越接近通过 p 点的等照度线, $E(p, q)$ 的值越大。然后对 $B_\epsilon(p)$ 中的 $E(p, q)$ 从大到小进行排序,并选取前 2ϵ 个为等照度线的权值非零点,记为 $q_i (1 \leq q_i \leq 2\epsilon)$,并为其分配权值。加权函数 $w(p, q_i)$ 采用 $\text{dst}(p, q_i) \cdot \text{lev}(p, q_i)$,其中 $\text{dst}(p, q_i) = 1 / \|p - q_i\|^2, \text{lev}(p, q_i) = 1 / (1 + |T(p) - T(q_i)|)$,归一化后的结果为:

$$W(p, q_i) = \text{dst}(p, q_i) \cdot \text{lev}(p, q_i) / \left(\sum_{i=1}^{2\epsilon} \text{dst}(p, q_i) \cdot \text{lev}(p, q_i) \right) \quad (7)$$

实验结果也将验证改进后的梯度和加权函数能够更好保持边缘的细节信息。

1.4 快进法修复图像步骤

根据 1.1~1.3 节内容,算法基本步骤如下:

(1)初始化。

①将初始边界 $\partial\Omega$ 所在的点标记为 ALIVE,并且 $T = 0$;

②考察 $\partial\Omega$ 上所有的 ALIVE 点的四邻点,如果有未被标记为 ALIVE 的,则以 ACTIVE 标记,并将其 T 赋为 1,然后将标记为 ACTIVE 的点置入一个按 T 从小到大排列的排序堆中。

③剩下的既不是 ACTIVE 也不是 ALIVE 标记的

其他点,标记为 INSIDE, $T = 10^6$ 。

(2) 快进修复。

① 输出 T 最小的 ACTIVE 点:从排序堆输出 T 最小的点 $p_{\min}(i_{\min}, j_{\min})$, 标记为 ALIVE, 然后从排序堆中删除。

② 执行修复:按照 1.3 节中的改进方法确定其邻域 $B_{\epsilon}(p)$, 并计算该点的梯度以及邻域像素的加权函数, 最后得到该点的修复结果。

③ 更新邻点:选择 p_{\min} 的四连通邻点 $(i_{\min} - 1, j_{\min}), (i_{\min}, j_{\min} - 1), (i_{\min} + 1, j_{\min}), (i_{\min}, j_{\min} + 1)$ 中标记不是 ALIVE 的点, 通过式(4)计算该点的 T 值。计算完毕, 如果邻点的当前标记为 INSIDE, 则将其标记修改为 ACTIVE, 插入排序堆中; 如果邻点的标记是 ACTIVE, 则利用新的 T 值更新该点的 T , 并在排序堆中重新进行排列。

④ 收敛检查:如果 $T_{p_{\min}} < T_{\text{threshold}}$, 或者排序堆空, 则修复结束, 否则重复进行以上操作。

2 实验结果与分析

笔者在 Matlab 环境下实现了以上算法。图像中浅色环形为破损区域, 针对模拟图像和真实图像分别进行了修复实验, 并与原 FMM 算法的修复结果进行了比较。

从实验可以看出, 改进后的算法能够有效地改善修复结果。

2.1 模拟图像的实验结果

如图 1 所示, 对图 1a 中的环形破损区域进行了修复。邻域 $B_{\epsilon}(p)$ 中的 ϵ 取 3 个像素, 如果取的过大将会导致边界的模糊。图 1b 则是使用快进修复的结果。图 1c 是使用改进后算法的修复结果, 其边界信息得到了较好的保持。

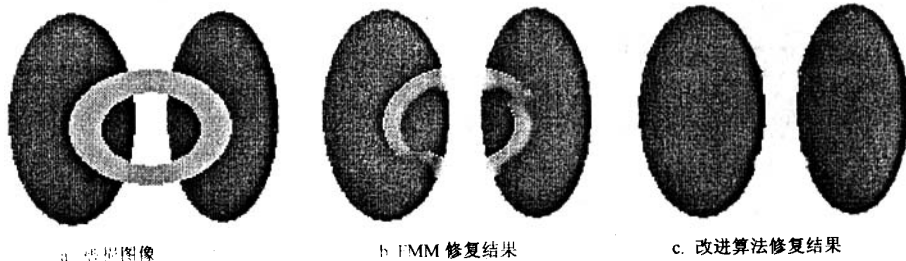


图 1 破损区域图以及修复结果

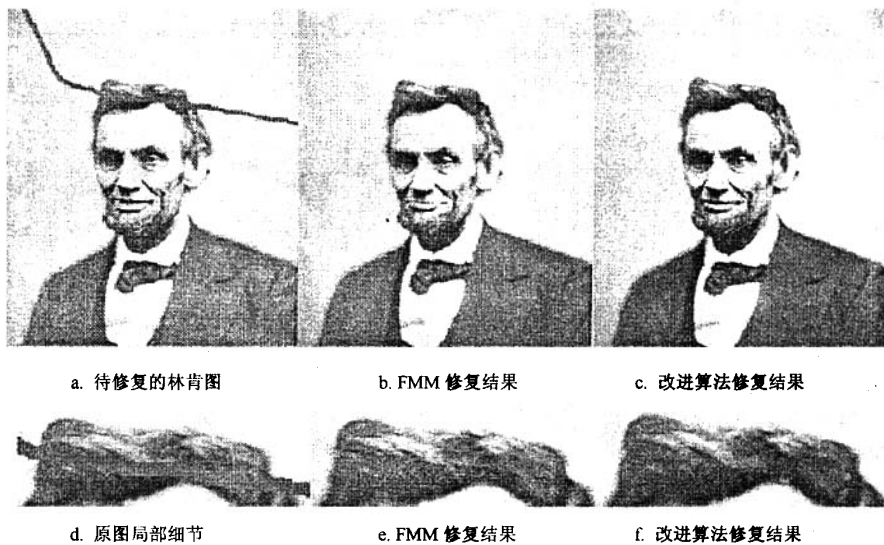


图 2 林肯图的修复结果

2.2 真实图像的实验结果

这里所测试的图像均来自于互联网。图 2 对林肯图上的折痕进行修复, 图 2b, c 分别是使用不同方法得到的修复结果, 图 2d~f 针对其局部细节进行了放大。图 3 对水果图上的涂鸦文字进行了修复, 结果如图 3b~f 所示。在获得原始未破坏的图像的情况下, 人为地在图像上添加了一些污渍, 然后进行修复, 如图 4 所示。最后, 采用均方误差(MSE)来衡量这两种方法的修复效果。均方误差(MSE)将从 R, G, B 三个通道进行误差分析, 如表 1 所示, 其结果显示了改进后的算法要优于原算法。

表 1 局部细节的 MSE 分析

算法	通道		
	MSE (R)	MSE (G)	MSE (B)
FMM 算法	5.0369	4.035	4.2387
改进算法	4.1308	3.2147	3.5895

3 结论

文中研究了快速步进算法修复图像的原理, 并提出了改进方案, 即利用梯度排序选择周围区域中权重较大的像素点来对目标像素点进行逼近。实验结果也



图 3 水果图修复结果

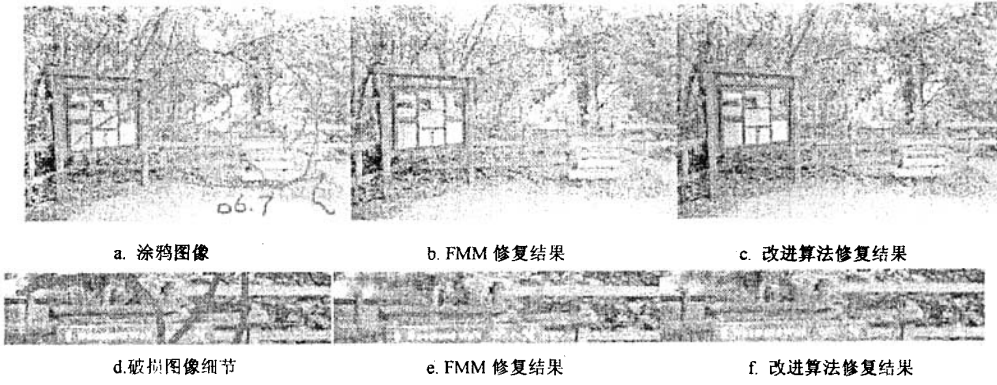


图 4 森林图修复结果

表明,改进后的算法要优于原先的 FMM 算法,且继承了原算法在运算速度上的优势。

参考文献:

[1] Bertalmio M, Sapiro G, Caselles V, et al. Image Inpainting [C]//In Proceedings SIGGRAPH 2000, Computer Graphics Proceedings, Annual Conference Series, edited by Kurt Akeley. Reading, MA:Addison - Wesley,2000:417 - 424.

[2] Oliveira M, Bowen B, McKenna R, et al. Fast Digital Image Inpainting[C]//In Proc. VIIP 2001. Marbella, Spain: [s. n.],2001:261 - 266.

[3] Chan T, Shen J. Mathematical Models for Local Non - Texture Inpainting[J]. SIAM J Appl Math,2001,62(3):1019 -

1043.

[4] Chan T, Shen J. Non - Texture Inpainting by Curvature - Driven Diffusions (CDD)[R]. CAM 00 - 35, Image Processing Research Group, UCLA,2000.

[5] Sethian J A. A Fast Marching Level Set Method for Monotonically Advancing Fronts[J]. Proc Nat Acad Sci,1996,93:1591 - 1595.

[6] Telea A. An image inpainting technique based on the fast marching method[J]. Journal of Graphics Tools,2004,9(1):23 - 34.

[7] 杨 新. 图像偏微分方程的原理和应用[M]. 上海:上海交通大学出版社,2003.

(上接第 205 页)

不必要的数据的传输,减少了客户等待时间和服务器的压力。随着 Gmail, Google - maps 的应用和各种浏览器的支持,Ajax 正逐渐被全世界的程序员所认识和使用。它的应用将越来越广泛。

参考文献:

[1] Asleson R. Ajax 基础教程[M]. 北京:人民邮电出版社,2006.

[2] 游丽贞. Ajax 引擎的原理和应用[J]. 微计算机信息,2006:22(2):205 - 206.

[3] Jaworski J. JavaScript 从入门到精通[M]. 北京:电子工业出版社,2002.

[4] 孙卫琴. 精通 struts: 基于 MVC 的 Java Web 设计与开发[M]. 北京:电子工业出版社,2005.

[5] 飞思科技产品研发中心. JSP 应用开发详解[M]. 北京:电子工业出版社,2002.