

基于腐蚀处理和多参数因子的 CDD 修复算法

杜闪闪, 韩超*

安徽工程大学电气工程学院, 安徽 芜湖 241000

摘要 针对曲率驱动扩散(CDD)修复算法迭代次数多、花费时间长及无法修复边界破损图像等问题,提出了一种基于腐蚀处理和多参数因子的 CDD 修复算法。对待修复图像的掩模进行腐蚀处理,从而避免因掩模过大造成的时间消耗。在曲率计算中引入自然对数指数因子,并在梯度模值计算中引入线性指数因子。通过调节不同指数因子中的参数值,避免了不合理的梯度模值、曲率值对扩散速度的影响。在曲率和梯度模值的计算中引入随迭代次数变化的自适应正提升参数,使受损区域能够被较好地修复。最后,根据边界破损的图像边界所处的位置加入相应的边界处理条件,解决了边界破损点无法得到修复的问题。仿真实验表明,改进后的 CDD 修复算法不但提高了图像修复速度,而且能够很好地修复边界有破损的图像。

关键词 图像处理; 曲率驱动扩散修复算法; 腐蚀运算; 指数因子; 自适应正提升参数; 边界处理条件

中图分类号 TP391

文献标识码 A

doi: 10.3788/LOP56.161011

CDD Inpainting Algorithm Based on Corrosion Treatment and Multi-Parameter Factors

Du Shanshan, Han Chao*

College of Electrical Engineering, Anhui Polytechnic University, Wuhu, Anhui 241000, China

Abstract The curvature driven diffusion (CDD) inpainting algorithm requires a significant number of iterations and long time for repairing images. In addition, this algorithm cannot repair the damaged points in the image boundary. Herein, an algorithm based on corrosion treatment and multi-parameter factors is proposed to address the limitations of the CDD inpainting algorithm. The main steps of the algorithm can be given as follows. First, the image mask to be repaired is corroded to eliminate the wastage of time because of an oversized mask. Second, a natural logarithmic exponential factor and a linear exponential factor are introduced in the curvature and gradient modulus calculations, respectively. Further, the effect of unreasonable gradient modulus and curvature values on the diffusion rate can be avoided by adjusting the parameters of different exponential factors. In addition, a self-adaptive positive lifting parameter that varies with the number of iterations is introduced to calculate the curvature and gradient modulus, improving the repair of the damaged area. Finally, the corresponding boundary treatment conditions are adopted according to the boundary locations of the images with broken boundaries, avoiding the problem that a broken boundary cannot be repaired. The simulation results demonstrate that the improved CDD inpainting algorithm improves the repairing speed and that images with broken boundaries can also be repaired.

Key words image processing; curvature driven diffusion inpainting algorithm; corrosion operation; exponential factor; self-adaptive positive lifting parameter; boundary treatment condition

OCIS codes 100.2000; 100.3008; 100.4994; 100.5010

1 引言

数字图像修复技术是一种根据图像中已有的像

素信息对受损区域进行修复的技术。近年来,数字图像修复技术得到了快速的发展^[1-3],在古代字画的保护、影视特效的制作上具有重要的价值。基于偏

收稿日期: 2019-01-21; 修回日期: 2019-03-18; 录用日期: 2019-03-27

基金项目: 安徽省教育厅高校自然科学基金项目(KJ2016A056)、安徽省省级重点实验室开放课题(1506c085002)、安徽省自然科学基金面上项目(1508085MF121)

* E-mail: hanchaozh@126.com

微分方程的数字图像修复算法因其数值实现简单而得到快速的发展。典型的基于偏微分方程的图像修复算法有:总变分(TV)修复算法、曲率驱动扩散(CDD)修复算法等。

基于 CDD 模型的图像修复算法是 Chan 等^[4]在 TV 模型的基础上提出来的。因为 Chan 等在 CDD 图像修复算法中引入了曲率的计算,从而避免了 TV 模型在修复大面积破损图像时视觉不连通的现象。CDD 模型扩散的强度由曲率和梯度模值信息共同确定,由于其为高阶偏微分方程,故在修复破损面积较大的图像时迭代次数较多,修复所需的时间较长。2008 年,赵颜伟等^[5]提出一种基于 CDD 模型的快速图像修复算法,他们通过增加每次计算的像素数,将已知像素直接加权合成待修复像素,并通过待修复目标像素周围已知像素的梯度方差确定修复的次序,从而在一定程度上减少了修复所需的时间,但是该算法对大面积破损图像的修复效果较差。屈磊等^[6]提出基于曲率驱动方向热流的图像修复算法,该算法可以较好地保持图像边缘信息,但是修复所需的时间较长。2009 年,廉晓丽等^[7]在 CDD 模型基础上对算法进行改进,去除梯度模值的计算,从而简化了计算公式,但是该算法对大面积破损图像的修复时间仍然较长,并且修复效果不理想。2010 年,王卫卫等^[8]提出非局部曲率驱动扩散的修复算法,将非局部导数算子引入曲率驱动扩散模型中,该算法对纹理图像具有较好的修复效果,但是该算法需要在全局搜索相似点,故增加了修复时间。谢爱敏等^[9]将 Split Bregman 方法引入到 CDD 模型中,该方法虽然提高了修复质量,但是对大面积破损图像的修复时间比较长。2012 年,印勇等^[10]提出一种自适应的图像修复算法,即利用图像的边缘细节信息自动选取不同的修复方式,但是该方法对大面积破损图像的修复效果不佳。之后,印勇等^[11]通过自适应系数确定不同的修复方法,在大曲率时使用 CDD 修复算法,在其余条件下使用 TV 修复算法。该方法在一定程度上提高了图像的修复质量,但是若使用该方法对大区域破损图像进行修复,其视觉效果不太理想。2014 年,Jiang 等^[12]利用 CDD 修复模型对西藏壁画进行修复,并将 CDD 修复效果与 TV 修复效果作比较。Jiang 等的工作给西藏壁画提供了一种新的视觉效果。2015 年,Jin 等^[13]提出基于滤波的低秩 Hankel 矩阵图像修复算法,该算法对于去除较大的目标物具有较好的效果,但是大量的逆矩阵计算增加了程序的复杂度。Li 等^[14]提

出基于多方向特征的彩色稀疏块图像修复方法,使用超小波变换估计受损图像的多方向特征。该方法使修复后的图像较好地保持结构与邻域的一致性、纹理的清晰度,但是对具有随机方向特征的不规则纹理破损图像的修复效果不理想。2015 年,Hu 等^[15]利用 CDD 模型修复横向多镜头视频中的测井图像,结果表明 CDD 模型优于 TV 模型,且 CDD 模型具有较好的修复效果。王军锋等^[16]通过改进 CDD 模型的形式,加快了修复的速度,但是其方法对大面积破损图像的修复上视觉效果不佳。吕慧显等^[17]将 CDD 模型扩展到小波域图像的修复上,该方法对长距离的破损图像的修复具有较好地视觉效果,但是其实现过程十分复杂。Kawai 等^[18]提出基于抽象现实几何背景的图像修复方法,采用似是而非的背景填充从图像中去除实物留下的空白区域,使用局部平面相似背景的几何组合,提高了搜索效率。然而,Kawai 等算法的修复质量易受掩模大小、图片分辨率等影响。2017 年,窦立云等^[19]提出一种基于双树小波的修复算法。首先,通过小波变换将受损图片分解为低频与高频两部分;之后,高频图像使用 TV 修复算法,低频图像使用 CDD 修复算法;最后,将修复好的两部分图像通过小波逆变化合成完整图像。该方法加快了修复的速度、提高了修复质量,但是其实现过程复杂。

本文在使用 CDD 算法修复受损图像前,先对受损图像的掩模进行腐蚀处理。之后,在不影响修复质量的基础上,对经典的 CDD 算法进行改进,在 CDD 模型扩散项中引入指数因子和自适应正提升参数。该算法可避免 CDD 模型在修复图像时扩散速度较慢的问题。通过 MATLAB(R2017a 版)软件进行仿真,结果表明,本文算法在图像修复上具有较好的修复效果,并极大地提高了修复效率。

2 CDD 模型分析

2001 年,Chan 等^[4]在 TV 模型的基础上引入曲率项,提出了 CDD 模型修复算法。该模型在修复大面积破损的图像时满足“视觉连通性”原则,CDD 修复模型表示为

$$\frac{\partial u}{\partial t} = \nabla \cdot \left[\frac{g(|k|)}{|\nabla u|_a} \nabla u \right] +$$

$$\lambda_e(x)(u_0 - u), x \in (D \cup E), \quad (1)$$

式中: t 为时间; D 为待修复区域, E 为 D 的邻域; ∇ 为梯度算子; $\lambda_e(x)$ 为 Lagrange 乘子, x 表示图像中的横坐标,当 $x \in E$ 时, $\lambda_e(x) = \lambda$; 第一项

$\nabla \cdot \left[\frac{g(|k|)}{|\nabla u|_a} \nabla u \right]$ 为扩散项, k 为曲率, $\frac{g(|k|)}{|\nabla u|_a}$ 为扩散系数, 记 $w_p = \frac{g(|k|)}{|\nabla u|_a}$, $|\nabla u|$ 为梯度模值,

$|\nabla u|_a = \sqrt{a^2 + |\nabla u|^2}$, a 是为了避免 $|\nabla u| = 0$ 而添加的正提升参数, $g(|k|)$ 为曲率; u_0 为受到白噪声污染的初始图像值; u 为修复后的图像值。该模型可以边修复边去噪。

通常曲率 $g(|k|) = |k|$, k 的计算公式为

$$k = \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) = \frac{\partial}{\partial x} \left(\frac{u_x}{|\nabla u|} \right) + \frac{\partial}{\partial y} \left(\frac{u_y}{|\nabla u|} \right), \quad (2)$$

式中: u_x 为待修复目标点在 x 方向的灰度值; u_y 为待修复目标点在 y 方向的灰度值。

从(2)式可以看出, 曲率的计算相对比较复杂, 需要计算较多像素点处的梯度信息, 因此迭代次数越多, 图像修复所需的时间越长。对于扩散系数 $w_p = \frac{g(|k|)}{|\nabla u|_a}$, 当 $g(|k|)$ 一定时, $|\nabla u|_a$ 与 w_p 成反比; 当 $|\nabla u|_a$ 一定时, $g(|k|)$ 与 w_p 成正比。当曲率 $g(|k|)$ 的数值较小, 梯度模值 $|\nabla u|_a$ 较大时, w_p 的数值较小, 扩散的速度较慢; 当曲率 $g(|k|)$ 数值较大, $|\nabla u|_a$ 数值也较大时, 或者当曲率 $g(|k|)$ 数值较小, $|\nabla u|_a$ 数值也较小时, 两者会相互抵消, 最后的扩散系数也会相对较小, 扩散的速度较慢; 当曲率 $g(|k|)$ 数值较大, 梯度模值 $|\nabla u|_a$ 较小时, w_p 的数值较大, 扩散的速度较快。从以上分析可见, w_p 的数值大小取决于 $g(|k|)$ 与 $|\nabla u|_a$ 的数值取向, 当二者确定的扩散系数 w_p 较小时, 修复速度较慢, 并且迭代次数较多。针对 CDD 模型修复算法的缺陷, 本文提出一种改进算法。

3 改进 CDD 算法模型

基于 CDD 模型的修复算法对大面积破损图像的修复具有较好的视觉连通性。然而, 在修复图像前, CDD 算法需要先用掩模标注目标物。人为标注的掩模大小与实际的目标物大小往往会存在一定的误差。在标注掩模时, 为了将目标物完全遮掩, 人们往往会选择面积过大的掩模。可是, 掩模面积过大必然会导致修复过程的难度增加, 从而使图像修复所需的时间随之增加。为了减少掩模面积过大对 CDD 算法修复速度的影响, 在使用 CDD 算法修复前先对掩模进行腐蚀处理。

3.1 腐蚀处理

腐蚀是图形形态学运算之一, 腐蚀操作是将边

界由外向内收缩的过程, 从而消除需要处理的目标边缘点。腐蚀会缩减图像中的白色区域, 腐蚀后, 图像中白色区域的面积小于原图像的白色区域的面积。腐蚀运算类似于卷积操作, 将结构元素在目标图像中从左到右、从上到下移动, 如果结构元素的所有点都在目标图像内, 则目标图像内该结构元素被保留, 否则将被腐蚀掉。

如图 1 所示, 图 1(a) 为目标图像, 其被腐蚀后的图像为图 1(b)。通过对比图 1(a) 和 (b) 可以看出, 经过腐蚀后的图像白色部分小于原来图像中的白色部分。因此, 在修复图像前, 可利用腐蚀的特性对掩模进行适当地预处理, 经过腐蚀后再将缩小的掩模与所需要处理的图像目标物进行组合, 之后再对原来的图像进行修复, 最终可将需要修复图像中的目标物去除掉。通过腐蚀操作可以减小修复中多余掩模的面积。因为修复的过程实际上是去除图像中的掩模过程, 所以减少多余掩模的面积可以缩短图像修复所需的时间, 简化修复的过程。



图 1 腐蚀运算。(a) 目标图像; (b) 腐蚀后图像
Fig. 1 Corrosion operation. (a) Target image; (b) corroded image

3.2 快速 CDD 修复模型

如果直接使用 CDD 修复模型对经过腐蚀处理后的图像进行修复, 由于扩散项受曲率与梯度模值的共同影响, 当曲率值小于梯度的模值、或者曲率值为零时, 扩散系数都比较小, 图像修复的速度仍然提升不大。为了进一步加快修复的速度, 在 CDD 模型扩散项的曲率计算中引入自然对数指数因子, 并在梯度模值的计算中引入线性指数因子。通过调节不同指数因子中的参数大小即可以调节扩散系数的大小, 从而加快扩散速度。

$$\frac{\partial u}{\partial t} = \nabla \cdot \left\{ \frac{[g(|k|)]^{\ln(2-\mu)}}{|\nabla u|_{(2-\tau)}} \nabla u \right\} + \lambda_c(x)(u_0 - u), x \in (D \cup E), \quad (3)$$

式中: μ 是调节曲率大小的参数; τ 是调节梯度模值大小的参数。通过合理地调节 μ 和 τ 就可以加快图

像修复的速度。

此外,为了使修复后的图像更加自然,在曲率以及梯度模值的计算中均引入随着迭代次数变化的自适应正提升参数,即

$$\frac{\partial u}{\partial t} = \nabla \cdot \left\{ \frac{[g(|k|)]_{\eta}^{\ln(2-\mu)}}{|\nabla u|_{\eta}^{(2-\tau)}} \nabla u \right\} + \lambda_e(x)(u_0 - u), x \in (D \cup E), \quad (4)$$

式中:自适应正提升参数 $\eta = 0.01 \times m$, m 为迭代次数。引入自适应正提升参数可使扩散系数在迭代初期的取值较大,在迭代后期的取值较小,从而使受损区域被修复后的像素值能够平坦的过渡,修复后的图像看起来比较自然。

当破损区域处于图像边界处时,算法在计算边界受损点的梯度值时会缺少一个或者两个像素值。一般来讲,计算梯度值时需要受损点处上下左右 4 个点的像素值,因此传统的 CDD 算法无法修复边界破损点,视觉效果较差。本文针对此类破损,在上述算法中引入了边界处理条件,具体添加的公式根据所需修复的图像进行确定。当图像的 4 个边界均受损时,假设所读取的图像为 I_{img} , 图片的大小为 $M \text{ pixel} \times N \text{ pixel}$, 则引入的边界处理条件为

$$\begin{cases} I_{\text{img}}(:, 1) = I_{\text{img}}(:, 2), \\ I_{\text{img}}(:, N) = I_{\text{img}}(:, N - 1) \\ I_{\text{img}}(1, :) = I_{\text{img}}(2, :), \\ I_{\text{img}}(M, :) = I_{\text{img}}(M - 1, :) \\ I_{\text{img}}(:, 2) = I_{\text{img}}(:, 3), \\ I_{\text{img}}(:, N - 1) = I_{\text{img}}(:, N - 2) \\ I_{\text{img}}(2, :) = I_{\text{img}}(3, :), \\ I_{\text{img}}(M - 1, :) = I_{\text{img}}(M - 2, :) \end{cases}, \quad (5)$$

式中: $I_{\text{img}}(:, N)$ 表示 I_{img} 的第 N 列的所有元素; $I_{\text{img}}(M, :)$ 表示 I_{img} 的第 M 行的所有元素。

3.3 数值计算

记 O 为待修复的目标像素点, $\Lambda_O = \{E, N, W, S\}$ 为 O 点处的 4 邻域像素点, $z = \{e, n, w, s\}$ 为 O 点处的 4 邻域半像素点。记 $v = (v_1, v_2) = \frac{[g(|k|)]_{\eta}^{\ln(2-\mu)}}{|\nabla u|_{\eta}^{(2-\tau)}} \nabla u$, 通过中心差分法对(4)式进行离散化,则

$$\nabla \cdot v = \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} \approx \frac{v_{1,e} - v_{1,w}}{h} + \frac{v_{2,n} - v_{2,s}}{h}, \quad (6)$$

式中: h 为步长。

(6)式中 $h = 1$, 以 e 点为例,

$$v_{1,e} = \frac{[g(|k_e|)]_{\eta}^{\ln(2-\mu)}}{|\nabla u_e|_{\eta}^{(2-\tau)}} \left[\frac{\partial u}{\partial x} \right]_e \approx$$

$$\frac{[g(|k_e|)]_{\eta}^{\ln(2-\mu)}}{|\nabla u_e|_{\eta}^{(2-\tau)}} \frac{u_E - u_O}{h}, \quad (7)$$

$$|\nabla u_e|_{\eta}^{(2-\tau)} = \{(u_E - u_O)^2 + [(u_{NE} - u_{SE})/2]^2 + \eta^2\}^{\frac{2-\tau}{2}}, \quad (8)$$

$$[g(|k_e|)]_{\eta}^{\ln(2-\mu)} = \nabla \cdot \left[\frac{\nabla u_e}{|\nabla u_e|_{\eta}^{\ln(2-\mu)}} \right] = \frac{\partial}{\partial x} \left[\frac{\nabla u_{1,e}}{|\nabla u_e|_{\eta}^{\ln(2-\mu)}} \right] + \frac{\partial}{\partial y} \left[\frac{\nabla u_{2,e}}{|\nabla u_e|_{\eta}^{\ln(2-\mu)}} \right], \quad (9)$$

式中: u_{NE} 、 u_{SE} 均为待修复目标点在对角线方向的灰度值。同理可得 $v_{1,w}$ 、 $v_{2,n}$ 和 $v_{2,s}$, 将(6)~(9)式代入(4)式,化简后可得

$$u_O = \frac{\sum_{P \in \Lambda_O} \frac{[g(|k_z|)]_{\eta}^{\ln(2-\mu)}}{|\nabla u_z|_{\eta}^{(2-\tau)}} u_P + \lambda_e}{\sum_{P \in \Lambda_O} \frac{[g(|k_z|)]_{\eta}^{\ln(2-\mu)}}{|\nabla u_z|_{\eta}^{(2-\tau)}} + \lambda_e} u_{OO}, \quad (10)$$

式中: $|\nabla u_z| = \{|\nabla u_e|, |\nabla u_w|, |\nabla u_n|, |\nabla u_s|\}$ 为 4 个半像素点的梯度模值; $g(|k_z|) = \{|k_e|, |k_w|, |k_n|, |k_s|\}$ 为 4 个半像素点的曲率值; $u_P = \{u_E, u_W, u_N, u_S\}$ 为目标像素点的 4 邻域像素值, P 为待修复的目标像素点 O 点处的 4 邻域像素点。

扩散系数为

$$\omega_P = \frac{[g(|k_z|)]_{\eta}^{\ln(2-\mu)}}{|\nabla u_z|_{\eta}^{(2-\tau)}}, \quad (11)$$

式中: $\omega_P = \{\omega_E, \omega_N, \omega_W, \omega_S\}$ 为 $\{E, N, W, S\}$ 4 个点处对应的扩散系数。

权系数为

$$\begin{cases} h_{OP} = \frac{\omega_P}{\sum_{P \in \Lambda_O} \omega_P + \lambda_e} \\ h_{OO} = \frac{\lambda_e}{\sum_{P \in \Lambda_O} \omega_P + \lambda_e} \end{cases}, \quad (12)$$

从而得迭代公式为

$$u_O^{(m)} = \sum_{P \in \Lambda_O} h_{OP}^{(m-1)} u_P^{(m-1)} + h_{OO}^{(m-1)} u_O^{(m-1)}. \quad (13)$$

4 仿真结果与分析

本文利用 MATLAB(R2017a 版)软件进行仿真,中央处理器为 Intel(R) i7-4710MQ,主频为 2.50 GHz、内存为 8.00 GB、操作系统为 Windows 10(64 位)。实验主要针对图像中原有目标物的去

除过程进行仿真,图2是对达到边界的目标物进行去除,图3是对图像中不同色彩区域内小目标物进行去除,图4是对大面积目标物进行去除,其中图2和图4中的目标物均位于图像中单一色彩区域内。通过设置不同类型的目标物,可考察本文算法的适用范围。在图2~4中,本文算法的参数设置为: $\mu=1.4, \tau=1.1$ 。为了突出本文算法的优越性,将本文算法与CDD算法、文献[7]算法、文献[10]算法、文献[11]算法以及文献[16]算法进行比较,表1~3分别对应图2~4中三种情况下6种图像修复算法的迭代次数和时间。通过仿真图和仿真数据两个方面反映本文算法的优点。此外,为了说明参数 μ, τ 数值的选择对修复效果的影响,给出了本文算法的参数 μ, τ 变化时,对图2

(b)、3(b)和4(b)中掩模图的修复效果图(如图5所示)以及仿真数据(如表4所示)。对比结果证明,通过合理设置参数 μ, τ 的数值可加快图像修复的速度。

图2为Jump图,图片大小为310 pixel \times 208 pixel。对于图2(a)中的目标物进行去除,首先需要目标物进行标记,标记结果如图2(b)所示。标记之后,对标记的掩模图采用6种方法进行修复,分别为本文算法、CDD算法、文献[7]算法、文献[10]算法、文献[11]算法和文献[16]算法,结果分别如图2(c)~(h)所示。另外由于受损区域达到图像的边界处,因此在本文算法中加入了边界处理条件。从图2(c)的仿真结果可以看出,本文算法具有较好的视觉效果。

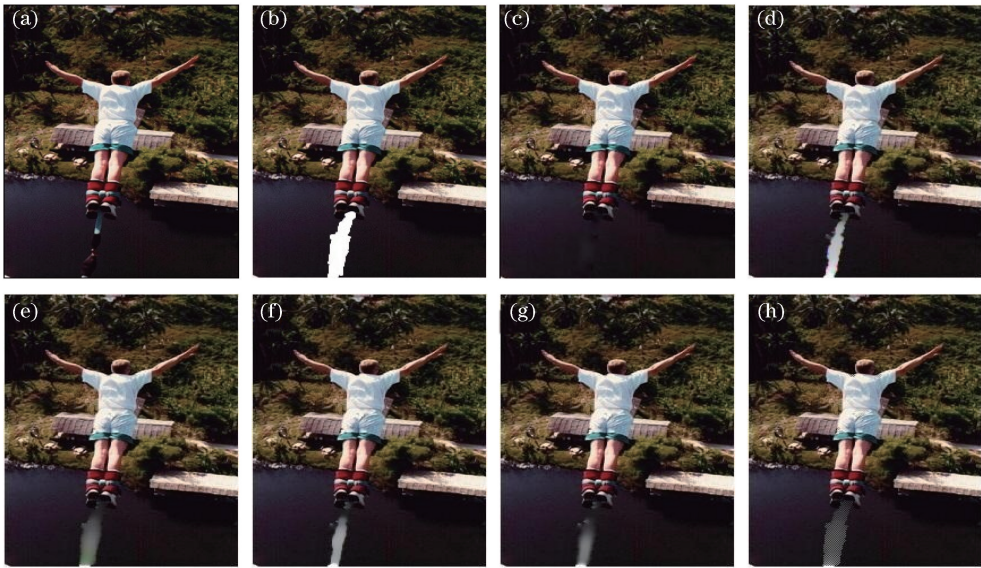


图2 仿真结果一。(a) Jump;(b)掩模图;(c)本文算法;(d)CDD算法;(e)文献[7]算法;
(f)文献[10]算法;(g)文献[11]算法;(h)文献[16]算法

Fig. 2 First simulation results. (a) Jump; (b) mask image; (c) proposed method; (d) CDD algorithm; (e) algorithm in Ref. [7]; (f) algorithm in Ref. [10]; (g) algorithm in Ref. [11]; (h) algorithm in Ref. [16]

表1是使用上述6种不同算法对图2(b)进行修复时的数据对比结果,本文算法只需要迭代200次($m=200$),并且只需花费较少的时间就可以将目标

物去除。其他算法的迭代次数均为500,需要花费比本文算法更长的时间进行图像修复,但是从效果图来看,其他算法的修复效果依然不如本文算法。

表1 6种图像修复算法的迭代次数和时间(仿真结果一)

Table 1 Iteration times and time for six kinds of inpainting algorithms (first simulation results)

Parameter	Proposed method	CDD algorithm	Algorithm in Ref. [7]	Algorithm in Ref. [10]	Algorithm in Ref. [11]	Algorithm in Ref. [16]
m	200	500	500	500	500	500
t/s	19.8906	33.5469	33.3438	33.9063	35.1875	33.8906

图3的大小为244 pixel \times 370 pixel,需要对图3(a)中远处的球以及人物进行去除,其掩模图如图3(b)所示。图3(c)~(h)是使用上述6种不

同算法进行仿真的结果,从仿真图可以看出,本文算法以及文献[11]算法的仿真效果比其他算法效果更好。

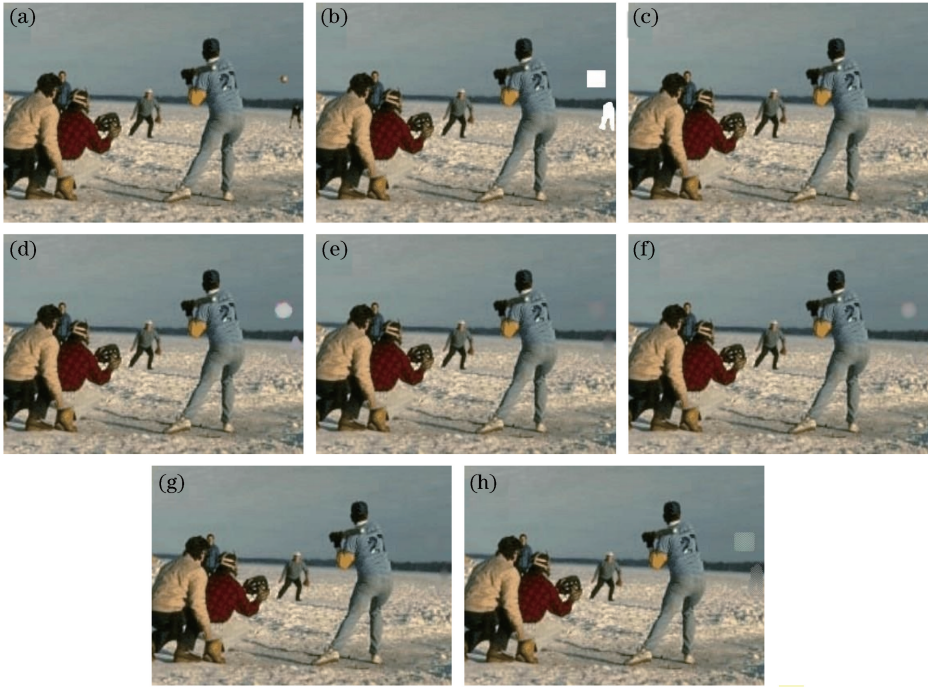


图3 仿真结果二。(a)原图;(b)掩模图;(c)本文算法;(d)CDD算法;(e)文献[7]算法;
(f)文献[10]算法;(g)文献[11]算法;(h)文献[16]算法

Fig. 3 Second simulation results. (a) Original image; (b) mask image; (c) proposed method; (d) CDD algorithm; (e) algorithm in Ref. [7]; (f) algorithm in Ref. [10]; (g) algorithm in Ref. [11]; (h) algorithm in Ref. [16]

表2是使用上述6种不同算法对图3(b)进行修复时,6种算法所需迭代次数与所需时间的对比结果。本文算法只需要迭代300次,用时

24.2500 s,而文献[11]算法达到相同的修复效果需要迭代500次,用时35.1094 s。综合图3和表2来看,本文算法更优越。

表2 6种图像修复算法的迭代次数和时间(仿真结果二)

Table 2 Iteration times and time for six kinds of inpainting algorithms(second simulation results)

Parameter	Proposed method	CDD algorithm	Algorithm in Ref. [7]	Algorithm in Ref. [10]	Algorithm in Ref. [11]	Algorithm in Ref. [16]
m	300	500	500	500	500	500
t/s	24.2500	34.9219	34.3594	34.6875	35.1094	34.8438

图4为文字图,图片大小为278 pixel × 413 pixel,图4(b)为其文字对应的掩模图。图4(c)~(h)为使用上述6种不同算法对图4(b)掩模去除的效果图。从效果图可以看出,本文算法、文献[7]算法和文献[11]算法比其他算法更优越。

表3是使用上述6种不同算法对图4(b)进行修复时,6种算法所需迭代次数与所需时间的对比结果。本文算法需要迭代300次,用时32.5625 s。文献[7]算法和文献[11]算法均需迭代800次,并且修复所需的时间是本文算法的两倍左右。表3反映出本文算法需要较少的迭代次数、花费较少的时间就可达到更好的修复效果。

图5为使用本文算法修复图2(b)、3(b)和4(b)的效果图,在迭代次数与图2~4保持相同情况下,

将参数 μ, τ 设置为其他3组数值。图5(a)~(c)是对图2(b)掩模图修复的效果图,图5(d)~(f)是对图3(b)掩模图修复的效果图,图5(g)~(i)是对图4(b)掩模图修复的效果图。从效果图中可以看出,当 μ, τ 被设置为其他数值时,修复效果不理想,说明合适的 μ, τ 数值可以改善修复效果。

表4是使用本文算法分别对图2(b)、3(b)和4(b)进行修复并得到对应效果图所需迭代次数与所需时间的结果。通过与表1~3中的本文算法数据对比可知,使用本文算法修复,在相同迭代次数下,不同的参数 μ, τ 数值会影响修复的速度,由表4可知,图2~4中本文算法中的参数 μ, τ 数值的选择比较合理,算法在进行图像修复时的速度是最快的。

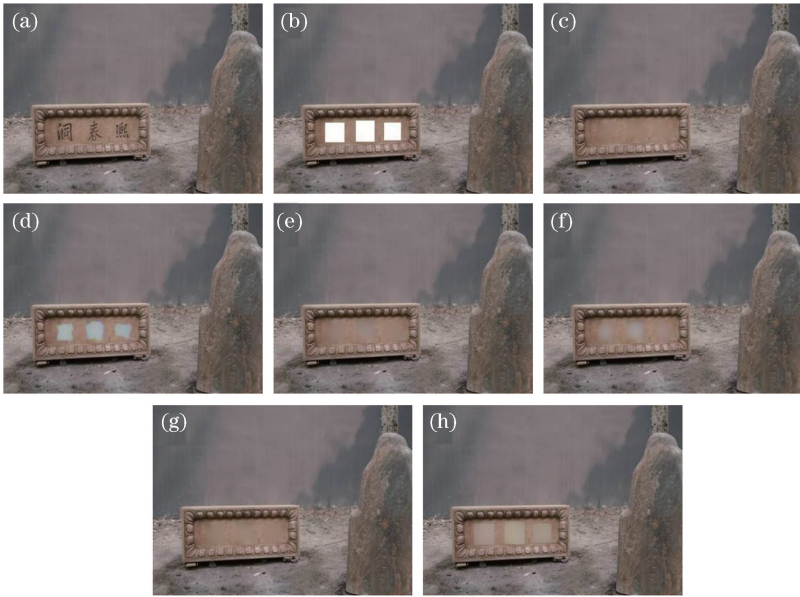


图 4 仿真结果三。(a)文字图;(b)掩模图;(c)本文算法;(d)CDD算法;(e)文献[7]算法;
(f)文献[10]算法;(g)文献[11]算法;(h)文献[16]算法

Fig. 4 Third simulation results. (a) Text image; (b) mask image; (c) proposed method; (d) CDD algorithm; (e) algorithm in Ref. [7]; (f) algorithm in Ref. [10]; (g) algorithm in Ref. [11]; (h) algorithm in Ref. [16]

表 3 6 种图像修复算法的迭代次数和时间(仿真结果三)

Table 3 Iteration times and time for six kinds of inpainting algorithms (third simulation results)

Parameter	Proposed method	CDD algorithm	Algorithm in Ref. [7]	Algorithm in Ref. [10]	Algorithm in Ref. [11]	Algorithm in Ref. [16]
m	300	800	800	800	800	800
t/s	32.5625	62.0156	61.9063	68.8438	66.1094	63.2344

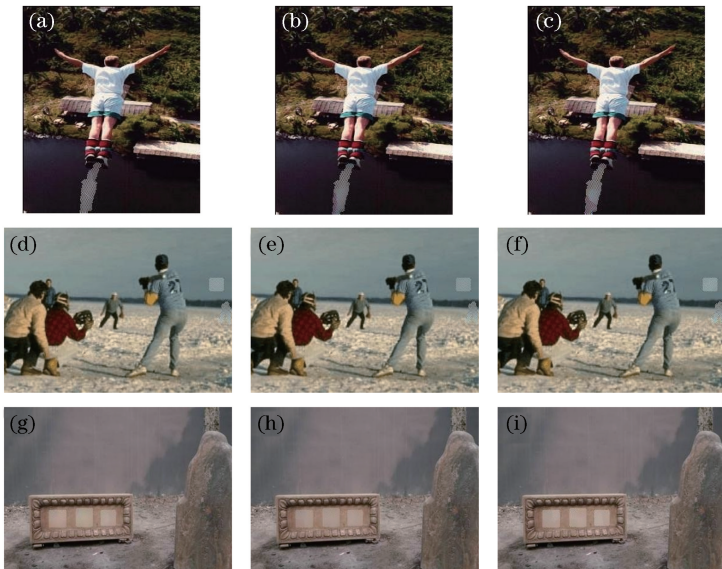


图 5 仿真结果四。(a) $\mu=1.7, \tau=2.1$; (b) $\mu=2.3, \tau=1.9$; (c) $\mu=2.2, \tau=2.2$; (d) $\mu=1.7, \tau=2.1$; (e) $\mu=2.3, \tau=1.9$;
(f) $\mu=2.2, \tau=2.2$; (g) $\mu=1.7, \tau=2.1$; (h) $\mu=2.3, \tau=1.9$; (i) $\mu=2.2, \tau=2.2$

Fig. 5 Fourth simulation results. (a) $\mu=1.7, \tau=2.1$; (b) $\mu=2.3, \tau=1.9$; (c) $\mu=2.2, \tau=2.2$; (d) $\mu=1.7, \tau=2.1$;
 $\tau=2.1$; (e) $\mu=2.3, \tau=1.9$; (f) $\mu=2.2, \tau=2.2$; (g) $\mu=1.7, \tau=2.1$; (h) $\mu=2.3, \tau=1.9$; (i) $\mu=2.2, \tau=2.2$

表4 本文算法在其他参数值下的迭代次数和时间

Table 4 Iteration times and time of the proposed algorithm under other parameters

Parameter	Fig. 2(b)			Fig. 3(b)			Fig. 4(b)		
μ	1.7	2.3	2.2	1.7	2.3	2.2	1.7	2.3	2.2
τ	2.1	1.9	2.2	2.1	1.9	2.2	2.1	1.9	2.2
m	200	200	200	300	300	300	300	300	300
t/s	20.1250	25.2031	25.1094	25.0625	29.9375	30.0313	33.0938	47.4688	53.0156

5 结 论

针对CDD修复算法在图像修复上需要大量的迭代和花费时间较长等缺陷,对CDD算法进行了改进。首先对掩模进行适当的腐蚀处理,从而避免了掩模过大造成的时间消耗。其次在原算法扩散项的曲率和梯度模值中引入不同类型的指数因子和随着迭代次数变化的自适应正提升参数,通过调节参数值的大小使扩散速度在迭代初期较大、迭代后期较小,从而达到快速修复的目的。最后,对有边界受损的图像在修复时加入边界处理条件,避免了边界受损点无法修复的问题。仿真效果图以及仿真数据对比表明,本文算法在图像修复上,减少了迭代次数,缩短了图像修复所需时间,并且修复效果较好。但是,本文算法对于纹理丰富的图像修复效果不够自然,其原因是该算法通过周围像素的信息由外向内不断扩散,最终完成修复。后续工作将针对本文算法的缺陷进行完善。

参 考 文 献

- [1] Cao W, Wang H B, Shi J, *et al.* Enhancement algorithm of finger vein image based on weighted guided filter with edge detection [J]. *Laser & Optoelectronics Progress*, 2017, 54(2): 021007.
曹伟, 王华彬, 石军, 等. 基于边缘检测加权引导滤波的指静脉图像增强算法[J]. *激光与光电子学进展*, 2017, 54(2): 021007.
- [2] Ni J Y, Li Q W, Zhou Y Q, *et al.* Underwater image restoration based on transmittance optimization and color temperature adjustment [J]. *Laser & Optoelectronics Progress*, 2017, 54(1): 011001.
倪锦艳, 李庆武, 周亚琴, 等. 基于透射率优化和色温调节的水下图像复原[J]. *激光与光电子学进展*, 2017, 54(1): 011001.
- [3] Yin S B, Wang Y B, Li D P, *et al.* Fast blind restoration for microscopic visual defocused images based on two guided filterings [J]. *Acta Optica Sinica*, 2017, 37(4): 0410002.
尹诗白, 王一斌, 李大鹏, 等. 两次引导滤波的显微

视觉散焦图像快速盲复原[J]. *光学学报*, 2017, 37(4): 0410002.

- [4] Chan T F, Shen J H. Nontexture inpainting by curvature-driven diffusions [J]. *Journal of Visual Communication and Image Representation*, 2001, 12(4): 436-449.
- [5] Zhao Y W, Li X L. A rapid image inpainting algorithm based on CDD model [J]. *Computer Simulation*, 2008, 25(10): 223-227.
赵颜伟, 李象霖. 基于CDD模型的快速图像修复算法[J]. *计算机仿真*, 2008, 25(10): 223-227.
- [6] Qu L, Wei S, Liang D, *et al.* Inpainting model which based on curvature driven oriented heat-flows [J]. *Computer Engineering and Applications*, 2008, 44(14): 31-33, 38.
屈磊, 韦穗, 梁栋, 等. 基于曲率驱动方向性热流的图像修复模型[J]. *计算机工程与应用*, 2008, 44(14): 31-33, 38.
- [7] Lian X L, Xu Z Y, Feng L L, *et al.* Novel image inpainting based on partial differential equation [J]. *Computer Engineering*, 2009, 35(6): 234-236.
廉晓丽, 徐中宇, 冯丽丽, 等. 一种新的基于偏微分方程的图像修复[J]. *计算机工程*, 2009, 35(6): 234-236.
- [8] Wang W W, Li L, Han Y. Image inpainting based on nonlocal curvature-driven diffusion [J]. *Acta Optica Sinica*, 2010, 30(6): 1634-1638.
王卫卫, 李莉, 韩雨. 基于非局部曲率驱动扩散的图像修复[J]. *光学学报*, 2010, 30(6): 1634-1638.
- [9] Xie A M, Zhou G H, Feng X C. Image inpainting algorithm of curvature driven diffusions based on Split Bregman method [J]. *Laser & Optoelectronics Progress*, 2010, 47(8): 081002.
谢爱敏, 周光华, 冯象初. 一种基于Split Bregman方法的快速曲率驱动图像修补算法[J]. *激光与光电子学进展*, 2010, 47(8): 081002.
- [10] Yin Y, Hu L Y, Zeng Q. Adaptive image inpainting algorithm based on curvature driving [J]. *Journal of Optoelectronics & Laser*, 2012, 23(12): 2426-2431.
印勇, 胡琳昀, 曾强. 采用曲率驱动的自适应图像修复算法[J]. *光电子·激光*, 2012, 23(12): 2426-

- 2431.
- [11] Yin Y, Li D, Hu L Y. Adaptive image inpainting algorithm based on CDD model [J]. Journal of Chongqing University, 2013, 36(4): 80-86.
印勇, 李丁, 胡琳昀. 采用 CDD 模型的自适应图像修复算法[J]. 重庆大学学报, 2013, 36(4): 80-86.
- [12] Jiang J, Wang Z X. The research of Tibet mural digital images inpainting using CDD model[C]//2013 2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation (IMSNA), December 23-24, 2013, Toronto, ON, Canada. New York: IEEE, 2014: 805-807.
- [13] Jin K H, Ye J C. Annihilating filter-based low-rank Hankel matrix approach for image inpainting [J]. IEEE Transactions on Image Processing, 2015, 24(11): 3498-3511.
- [14] Li Z D, He H J, Tai H M, *et al.* Color-direction patch-sparsity-based image inpainting using multidirection features [J]. IEEE Transactions on Image Processing, 2015, 24(3): 1138-1152.
- [15] Hu H T, Tong X. Application of curvature driven diffusion model in lateral multi-lens video logging image inpainting[C]//2015 12th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), July 16-18, 2015, Qingdao, China. New York: IEEE, 2015: 1167-1171.
- [16] Wang J F, Pei Y X, Wang T. New algorithm of image inpainting curvature-driven diffusion model[J]. Computer Systems & Applications, 2016, 25(8): 135-138.
王军锋, 裴艳侠, 王涛. 图像修复的 CDD 模型新算法[J]. 计算机系统应用, 2016, 25(8): 135-138.
- [17] Lü H X, Zhao Z G, Guo Y J, *et al.* A new CDD model based on three channel multiwavelet tight frames[J]. Journal of Optoelectronics • Laser, 2016, 27(1): 77-86.
吕慧显, 赵志刚, 郭银景, 等. 基于三通道多小波紧标架的图像曲率修复模型[J]. 光电子 • 激光, 2016, 27(1): 77-86.
- [18] Kawai N, Sato T, Yokoya N. Diminished reality based on image inpainting considering background geometry [J]. IEEE Transactions on Visualization and Computer Graphics, 2016, 22(3): 1236-1247.
- [19] Dou L Y, Xu D, Li J, *et al.* Image inpainting based on dual-tree complex wavelet transform [J]. Computer Science, 2017, 44(S1): 179-182, 191.
窦立云, 徐丹, 李杰, 等. 基于双树复小波的图像修复[J]. 计算机科学, 2017, 44(S1): 179-182, 191.