

一种新型的图像恢复算法*

余国华, 田岩

(华中科技大学 电子与信息工程系, 湖北 武汉 430074)

摘要:针对巨型图片恢复处理过程中所面临的运算量庞大的技术瓶颈,从光学图像退化模型出发,发展了一种基于最小二乘方的恢复算法,在算法的实现过程中回避了变换到频域的常规做法,提出了一种新颖的级联模板的运算方法,在一定程度上减小了运算量,降低了复杂度,实验结果表明该方法是有效的。

关键词: 图像恢复; 最小二乘方; 级联模板

中图分类号: TP391 **文献标识码:** A **文章编号:** 1007-2276(2004)01-0034-04

Novel algorithm of image restoration*

YU Guo-hua, TIAN Yan

(Electronic & Information Engineering Department, Huazhong University of Science & Technology, Wuhan 430074, China)

Abstract: In order to reduce the complexity of calculation while processing large decreased image, an image restoration algorithm based on least square has been developed. When putting it into implementation, a new method named ally-template has been proposed, which parrys transform of frequency. Its algorithm architecture is easily implemented. The experimental results demonstrate this approach is effective.

Keywords: Image restoration; Least square algorithm; Ally-template

0 引言

图像恢复就是将降质了的图像恢复成原来的图像,是图像处理中的一个十分重要的研究领域。通过图像恢复技术,可有效地提高降质图像的质量,便于进行图像分析、识别、理解等。多年来人们对图像恢

复进行了大量的研究,提出了多种基于空间域和频域的图像恢复方法^[1,2]。首先针对光学图像退化模型,发展了一种基于最小二乘方的恢复算法,然后在程序实现的过程中回避了传统的傅里叶变换^[3],提出了级联模板的运算方法,在一定程度上减小了运算量,降低了复杂度,实验结果表明了该方法的有效性。

收稿日期:2003-07-20; 修订日期:2003-09-18

*基金项目:高等学校博士学科点专项基金资助项目(20010487030)

作者简介:余国华(1978-),男,湖北蕲春人,硕士生,主要从事图像处理及信号处理等方面的研究。

1 算法的提出

如果将图像的衍射降质过程模型简化为一个降质系统(或算子) \mathbf{H} ,并假设输入原始图像为 f ,经降质系统作用后输出的降质图像为 g ,在降质过程中引进的噪声为 n ,则模型可用方程表示为:

$$g = \mathbf{H}f + n \quad (1)$$

针对方程(1),本节研究在得到降质图像 g 的情况下如何恢复出原始图像 f 。

1.1 最小二乘方算法

为行文方便,考虑如下的方程:

$$y = \mathbf{A}x + b \quad (2)$$

对此方程,结合存在的问题,通常情况下算子 \mathbf{A} 是奇异的,即不可逆,因此不能直接求解,即便是可逆的,由于实际中的 \mathbf{A} 是阶数很大的矩阵,其逆运算是一个非常复杂的过程,因此也不可能采取直接求解的方式。为此,利用带约束的最小二乘方法^[4],可将方程(2)转化为下式:

$$\mathbf{A}^T \mathbf{A}x + \alpha \mathbf{M}^T \mathbf{M}x - \mathbf{A}^T y = 0 \quad (3)$$

即为:

$$(\mathbf{A}^T \mathbf{A} + \alpha \mathbf{M}^T \mathbf{M})x = \mathbf{A}^T y \quad (4)$$

式中 α 为大于0的常数; \mathbf{M} 为一正则化算子,通常取 Laplace 算子。

方程(4)进一步可简化为如下形式:

$$\tilde{\mathbf{A}}x = \tilde{b} \quad (5)$$

式中 $\tilde{\mathbf{A}} = \mathbf{A}^T \mathbf{A} + \alpha \mathbf{M}^T \mathbf{M}$; $\tilde{b} = \mathbf{A}^T y$ 。方程(5)对应的最小化泛函为 $\tilde{L}(x) = \frac{1}{2} x^T \tilde{\mathbf{A}}x - x^T \tilde{b}$ 。

关于方程(5)的解,由于矩阵 $\tilde{\mathbf{A}}$ 的尺寸非常大,直接求解是不可能的,通常采取基于梯度的迭代方法(Gradient-based Iterative Method),而此方法又有两种,即最速下降法(Steepest Descend, SD)和共轭梯度法(Conjugate Gradient, CG)^[5]。

基于梯度的迭代方法从解的一个初始猜测开始,

例如选取 $x_0 = 0$ 或者 $x_0 = \tilde{b}$ 。最速下降法(或最陡下降法)从取 $x_0 = \tilde{b}$ 开始。最速下降法就是沿性能曲面的最陡方向向下搜索曲面的最低点。曲面的最陡下降方向是曲面的负梯度方向,该迭代搜索过程首先从曲面上某个初始点出发,沿该点负梯度方向搜索至第1点,然后再从第1点开始,重复前面的过程,直至寻找到曲面的最低点。最速下降法的迭代方程为:

$$x_{k+1} = x_k + \alpha_k (\tilde{\mathbf{A}}x_k - \tilde{b}) \quad (6)$$

式中 α_k 的一个最佳选择是:

$$\alpha_k = \frac{-d^T \tilde{\mathbf{A}}^T \tilde{\mathbf{A}} d}{d^T \tilde{\mathbf{A}}^T \tilde{\mathbf{A}} \tilde{\mathbf{A}}^T \tilde{\mathbf{A}} d} \quad (7)$$

式中 $d = \tilde{\mathbf{A}}x_k - \tilde{b}$ 。当 $\alpha_k = \alpha$ 并为常数时,最优值为:

$$\alpha = \frac{2}{\lambda_{\min}(\tilde{\mathbf{A}}^T \tilde{\mathbf{A}}) + \lambda_{\max}(\tilde{\mathbf{A}}^T \tilde{\mathbf{A}})}$$

对于高度病态条件, $\tilde{\mathbf{A}}^T \tilde{\mathbf{A}}$ 的 $\lambda_{\max}/\lambda_{\min}$ 太大,则 α 应选择上限 $2/\lambda_{\max}$ 。

最速下降法只是在迭代开始的若干步能使目标泛函下降很快,以后的收敛变得越来越慢,用共轭梯度法可以使收敛得到改善。已有的结果表明,如果数值计算足够精确,利用共轭梯度法求解方程(6)能在一定的步数内达到收敛。如果数值计算不够精确或泛函不是二次的,就不能保证这样的收敛性质,但收敛速度仍比最速下降法快。

共轭梯度法从 $x_0 = 0$ 开始,并假定搜索方向为 $p_0 = \tilde{b}$,于是 $\Delta x_0 = \alpha p_0$ 。算法的每一步是沿着已经搜索的方向与 $\tilde{\mathbf{A}}$ 成共轭的新方向 p_k 寻找 $\tilde{L}(x)$ 的极小值,所谓 $\tilde{\mathbf{A}}$ 的共轭条件是指:

$$p_k^T \tilde{\mathbf{A}} p_i = 0 \quad i = 0, 1, 2, \dots, k-1 \quad (8)$$

对共轭梯度法的深入研究和计算实践证实,该方法的收敛速度与矩阵 $\tilde{\mathbf{A}}$ 的条件数紧密相关,即若 $\tilde{\mathbf{A}}$ 接近于单位矩阵,共轭梯度法的收敛速度会非常快。

基于以上分析,可以重述一下方程(1)的算法,由 $g = \mathbf{H}f + n$ 构造:

$$f_{k+1} = f_k + \alpha_k (\mathbf{H}^T \mathbf{H} f_k - \mathbf{H}^T g) \quad (9)$$

令 $f_0 = \mathbf{H}^T g$ 或 $f_0 = 0$, 其中:

$$\alpha_k = -\frac{d^T \mathbf{H}^T \mathbf{H} d}{d^T \mathbf{H}^T \mathbf{H} \mathbf{H}^T \mathbf{H} d} \quad (10)$$

$$d = \mathbf{H}^T \mathbf{H} f_k - \mathbf{H}^T g \quad (11)$$

该迭代算法具有很快的收敛性, 迭代若干步即可取得恢复图像。

1.2 基于级联模板的运算方法

降质模型是用矩阵的形式表达的, 见方程(1): 设降质系统的点扩展函数为 $h(x, y)$, 具有 $K \times K$ 个元素, 其中 $K = 2r + 1$, f, n, h, g 经过添零延伸和周期延拓^[6]分别得到 f_e, n_e, h_e, g_e , 于是有:

$$g_e(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_e(m, n) h_e(x-m+r, y-n+r) + n_e(x, y) \quad (12)$$

式中 $x = 0, 1, \dots, M-1; y = 0, 1, \dots, N-1; M$ 和 N 分别是 x 和 y 方向上的延拓周期。

如果用矩阵表示方程(12), 即为 $g = \mathbf{H}f + n$, 其中 g, f, n 为 $M \times N$ 维列向量, 这些列向量是由 $M \times N$ 维的函数矩阵 $g_e(x, y), f_e(x, y)$ 和 $n_e(x, y)$ 的各个行向量堆积而成的, 记向量为:

$$\Delta_j = [h_e(j, r) \ h_e(j, r+1) \ \dots \ h_e(j, M-1) \ h_e(j, 0) \ h_e(j, 1) \ \dots \ h_e(j, r-1)]^T \quad (0 \leq j \leq N-1) \quad (13)$$

由 Δ_j 生成的循环矩阵记作 $\mathbf{H}_j = \text{CycMatrix}(\Delta_j)$, 又记矩阵为:

$$\mathbf{L} = [\mathbf{H}_r \ \mathbf{H}_{r-1} \ \dots \ \mathbf{H}_{N-1} \ \mathbf{H}_0 \ \mathbf{H}_1 \ \dots \ \mathbf{H}_{r-1}]^T \quad (14)$$

则 \mathbf{H} 可表示为 $\mathbf{H} = \text{CycMatrix}(\mathbf{L})$ 。

当点扩展函数 $h(x, y)$ 中心对称时, 有 $\mathbf{H} = \mathbf{H}^T$ 。记向量为:

$$\tilde{\Delta}_j = [h_e(j, r) \ h_e(j, r-1) \ \dots \ h_e(j, 0) \ h_e(j, M-1) \ h_e(j, M-2) \ \dots \ h_e(j, r+1)]^T \quad (15)$$

式中 $0 \leq j \leq N-1$ 。记矩阵为:

$$\tilde{\mathbf{L}} = [\mathbf{H}_r^T \ \mathbf{H}_{r-1}^T \ \dots \ \mathbf{H}_0^T \ \mathbf{H}_{N-1}^T \ \mathbf{H}_{N-2}^T \ \dots \ \mathbf{H}_{r-1}^T]^T \quad (16)$$

其中 $\mathbf{H}_j^T = \text{CycMatrix}(\tilde{\Delta}_j)$, $0 \leq j \leq N-1$, 则有 $\mathbf{H}^T = \text{CycMatrix}(\tilde{\mathbf{L}})$ 。当点扩展函数 $h(x, y)$ 中心对称时, 有 $h_e(j, r+j) = h_e(j, r-j)$, 其中 $j = 0, 1, 2, \dots, r$; 当 $2r \leq i \leq N-1$ 时, 恒有 $h_e(j, i) = 0$ 。因此对每一个分块循环矩阵有 $\mathbf{H}_{r-j}^T = \mathbf{H}_{r-j}$, 其中 $j = 0, 1, 2, \dots, r$; 当 $2r < i \leq N-1$, 恒有 $\mathbf{H}_i^T = \mathbf{H}_i = 0$, 从而有 $\mathbf{H} = \mathbf{H}^T$ 。

由此, 对于降质系统点扩展函数中心对称的情形, 可得到以下重要结论:

(1) 对任一数字图像 f , $\mathbf{H}f$ 和 $\mathbf{H}^T f$ 表示相同的作用效果;

(2) 算子 \mathbf{H} 和 \mathbf{H}^T 所对应的点扩展函数同为 $h(x, y)$ 。

上述结论可运用于算法的实现过程中。首先找到点扩展函数 $h(x, y)$ 所对应的模板 \mathbf{P} , 然后把 \mathbf{P} 当作滑动窗口置于原始图像, 逐一做模板运算就可以很方便地得到降质图像 $\mathbf{H}f$ 。同样的方法可得到 $\mathbf{H}^T f$, 而 $\mathbf{H}^T \mathbf{H}f$ 则可由 f 做两次模板运算得到, 这就是所谓的级联模板运算 (Ally-Template, AT)。通过级联模板运算, 可以方便地实现方程(9)~(11)。

降质模型中降质算子 \mathbf{H} 一般都是阶数很大的循环矩阵, 其运算量相当惊人, 通常的做法是转移到频域里通过傅里叶变换进行的^[3], 这里回避了常规方法, 直接在空域里通过级联模板的思路实现了恢复算法, 不仅恢复效果良好, 算法的复杂度亦有所降低。下面以一次傅里叶变换和一次级联模板运算为例, 简单地分析一下各自的复杂度。

设原始图像大小为 $M \times N$, 降质系统点扩展函数含 $K \times K$ 个元素 (对应模板的大小亦为 $K \times K$), $K = 2r + 1$, 则:

$$\delta_{\text{DFT}} \approx o(M^2 \times N^2)$$

$$\delta_{\text{FFT}} \approx o\left(\frac{M \times N}{2} \log_2(M \times N)\right)$$

$$\delta_{\text{TPL}} \approx o(M \times N \times K^2)$$

一般情况下, K 较小 (如 $K = 3$ 或 5 等), 而图像尺寸 $M \times N$ 较大, 所以通过级联模板解决恢复算法具有明显的优越性。

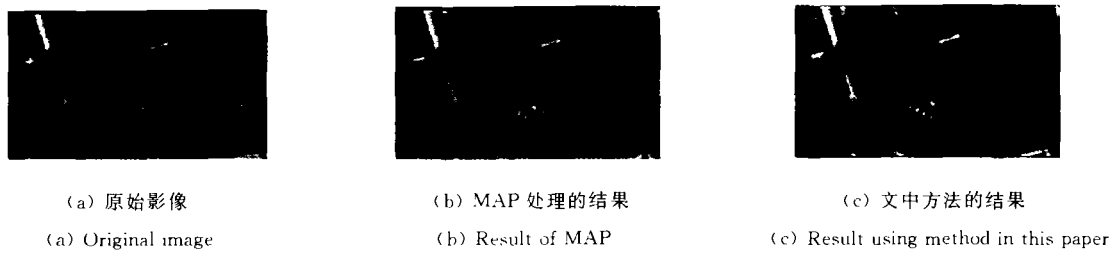


图 1 东京市某城区一角的退化图片、MAP 复原及文中方法的结果

Fig. 1 The degraded image of a certain zone in Tokyo, and the restored result of MAP algorithm and this ally-template algorithm

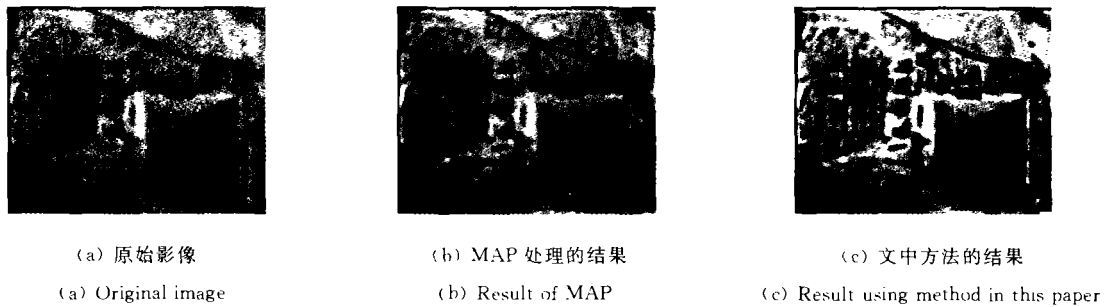


图 2 埃及金字塔图的退化图、MAP 复原及文中方法的结果

Fig. 2 The degraded image of the Egyptian pyramid, and the restored result of MAP algorithm and this ally-template algorithm

2 实验结果及分析

以东京某一局部城区和埃及金字塔图为原始影像(分别如图 1(a)和图 2(a)所示),图 1(b)和图 2(b)分别是采用最大后验概率处理后的结果^[1],图 1(c)和图 2(c)分别是文中方法处理的结果。

从视觉效果上看,无论是图 1 还是图 2,与原始图片相比均有明显的改进。下面对图 1 进行分析,如表 1 所示,运算时间采用主频 1.4 G 的机器上统计的结果。

表 1 文中方法和 MAP 方法的比较

Table 1 Contrast of this ally-template algorithm and MAP algorithm

| Item | MAP algorithm | AT algorithm |
|-------------|---------------|--------------|
| Run time s | 0.317 | 0.203 |
| SNR dB | 2.631 | 2.968 |
| Correlation | 0.752 | 0.927 |
| Entrope | 3.192 99 | 4.212 98 |

实验表明,文中的方法较之 MAP 方法所得到的恢复图像,无论是信噪比、相关性(与原始图像比)还是信息熵均有一定程度的提高。

参考文献:

- [1] Chan A, Meloche J. Image restoration[J]. Journal of Statistical Planning and Inference, 1997, 65: 233-254.
- [2] Banham M, Katsaggelos A. Digital image restoration[J]. IEEE Signal Processing Magazine, 1997, 11(2): 24-41.
- [3] 何斌,马天序,王运坚,等. Visual C++ 数字图像处理[M]. 北京:人民邮电出版社,2001.
- [4] Telalp Murat A. Digital Video Processing[M]. New York, Prentice Hall, 1996.
- [5] 邹谋炎. 反卷积和信号复原[M]. 北京:国防工业出版社, 2001.
- [6] 陈廷标,夏良正. 数字图像处理[M]. 北京:人民邮电出版社, 1990.