

一种基于递归反演的二维图像自适应算法

赖锦辉¹, 梁 松²

(1 广东石油化工学院 实验教学部计算机中心, 广东 茂名 525000;

2 广东石油化工学院 计算机与电子信息学院, 广东 茂名 525000)

摘 要: 针对图像除噪和图像解卷积处理问题, 在一维递归反演(RI)算法的基础上, 提出了一种用于二维自适应滤波的递归反演(RI)自适应算法. 通过采用可变步长, 从而避免使用系数更新方程中的反演相关矩阵, 达到了更稳定的性能. 在处理二维图像时, 它可以沿水平方向和垂直方向更新滤波器系数. 仿真结果表明, 在图像解卷积处理和自适应线谱增强器应用方面, 与二维 RLS 算法相比, 所提出的二维 RI 算法表现出更好的性能. 此外, 它的计算复杂性相对较低, 并且不需要自相关矩阵的反演, 从而拥有更稳定的性能.

关键词: 递归反演; 自适应滤波; 二维 RLS; 二维 RI; 图像解卷积处理

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2014)06-0077-04

A Two-dimensional Image Adaptive Algorithm Based on Recursive Inversion

LAI Jin-hui¹, LIANG Song²

(1 Computer Center, Guangdong University of Petrochemical Technology, Maoming 525000, China;

2 College of Computer and Electronic Information,

Guangdong University of Petrochemical Technology, Maoming 525000, China)

Abstract: To reduce noise of the image signals and image deconvolution problem, there is proposed recursive inverse (RI) adaptive algorithm for the two-dimensional adaptive filtering based on one-dimensional recursive inversion algorithm (RI). The proposed approach uses a variable step size and avoids the use of the inverse autocorrelation matrix in the coefficient update equation. The filter coefficients can be updated along both the horizontal and vertical directions on a 2-D plane. Simulation results show that the proposed 2-D RI algorithm leads to an improved performance compared to that of the 2-D RLS algorithm in an image deconvolution and an adaptive line enhancer problem setting.

Key words: recursive inverse; adaptive filtering; 2-D RLS; 2-D RI; image deconvolution

1 引言

自适应滤波技术在传统的信号处理问题上是非常有效的, 如可用于去除噪声、信道估计和系统识别. 由于数字成像和视频得到越来越广泛的应用, 二维自适应滤波技术对消费类电子产品和多媒体应用程序有着更重要的意义.

为了降低图像信号中的噪声, 开发和应用了不同类型的二维 LMS 自适应算法, 它们可以在二维平面上沿水平方向和垂直方向更新滤波器系数^[1].

其中, 最有效的算法是二维递推最小二乘算法(RLS)算法, 它在去除图像噪声和图像解卷积处理方面表现出了极高的性能. 靳庆贵等人^[2]提出利用限制更新的方法对分层交替非负最小二乘算法进行修改, 从而加速收敛. 简献忠^[3]等人在传统 NLMS 算法的基础上, 提出了重复调整归一化最小均方算法(DRNLMS)即在相邻两输入信号样本的间隔时间进行额外调整运算.

本文提出了一种新的二维递归反演(RI)自适应算法.

2 递归反演算法

2.1 递归反演算法的推导

正文内容. 单位符号无例外的采用正体字母. 注意区分单位符号的大小写: 一般单位符号为小写体, 来源于人名的单位符号首字母大写. 体积单位升的符号为大写 L.

下面给出的是维纳 - 霍普夫方程^[4], 通过该方程我们可以得到 FIR 滤波器系数的最优解:

$$\omega(k) = R^{-1}(k) p(k) \quad (1)$$

式中, $k = 1, 2, \dots$, $\omega(k)$ 表示滤波器权重向量的计算时间 k , $R(k)$ 表示滤波器输入的自相关矩阵的估计值, $p(k)$ 表示滤波器的输入和滤波器所需的响应之间的互相关向量的估计值. 为了更新滤波器系数, 式(1)中的解决方案在每次迭代中是必须应用的. 此外, 在每次迭代中, 只有自相关矩阵非奇异, 才能保证收敛到最优解^[5].

相关矩阵的递归估计是

$$R(k) = \beta R(k-1) + x(k) x^T(k) \quad (2)$$

$$p(k) = \beta p(k-1) + d(k) x(k) \quad (3)$$

式中, β 是指非常接近与 1 的遗忘因子. 把式(2)和式(3)代入式(1), 使用矩阵反演引理^[6], 得到了牛顿 LMS 算法的更新方程. 牛顿 LMS 算法是相当于带自相关和互相关指数遗忘窗估计的维纳解. 牛顿 LMS 算法要求自相关矩阵的反演. 然而, 在 RI 算法中, 这是可以避免的.

考虑在每一步对式(1)中的 k 进行反复求解. 具体而言, 将下面的迭代收敛到维纳解:

$$\omega_{n+1}(k) = [I - \mu R(k)] \omega_n(k) + \mu p(k), n = 0, 1, 2, \dots \quad (4)$$

如果 μ 满足收敛判别法^[5]

$$\mu < \frac{2}{\lambda_{\max}(R(k))} \quad (5)$$

其中, $\omega_0(k) = \omega(k-1)$.

通过对自相关矩阵的更新方程进行求解, 把 $\rightarrow \infty$ 式(2)中 $R(k)$ 的期望值看作 $k \rightarrow \infty$, 把结果代入式(5), 我们可以得到

$$\begin{aligned} \mu(k) &< \frac{2}{\lambda_{\max}(R(k))} = \left[\frac{1-\beta}{1-\beta^k} \right] \left[\frac{2}{\lambda_{\max}(R_{xx})} \right] \\ &= \frac{\mu_{\max}}{1-\beta^k} \end{aligned} \quad (6)$$

式中, $R_{xx}(k) = E\{x(k)x^T(k)\}$. 反之:

$$\mu(k) = \frac{\mu_0}{1-\beta^k} \quad (7)$$

式中, $\mu_0(k) < \mu_{\max}$. 式(4)中的迭代具有较高的计算成本. 因此, 在每个时间步中只发生一次可变步长的迭代就足够了. 最后, 提出的 RI 算法的权值更新方程就变成了

$$\omega(k) = [I - \mu(k) R(k)] \omega(k-1) + \mu(k) p(k) \quad (8)$$

与 RLS 算法相比, RI 算法有一个很大的优点, 那就是它不需要更新反演自相关矩阵. 并且, 如表 1 所示, 它的计算复杂度也比 RLS 算法的小. 由于反演自相关矩阵的更新, RLS 型算法可能会遇到数值稳定性方面的问题, 这在 RI 算法中是不会产生的^[4]. 利用文献[6]中的快速实现技术, 可以进一步减少 RI 算法的计算复杂性. 在信道均衡设置方面对 RLS 算法的性能进行了测试, 并且在 AWGN, ACGN, AWIN 和 ACIN 环境中与 RLS 算法的性能进行了比较. 在减少脉冲组件(异常值)的影响方面, RI 算法的优势超过了 RLS 算法.

表 1 RI 算法、快速 RI 算法和 RIS 算法的计算复杂性

	乘 / 除	加 / 减
RI 算法	$\frac{5}{2} N^2 + \frac{7}{2} N$	$2 N^2 + N$
快速 RI 算法	$N^2 + (5 + \frac{3}{2} \log_2 N) N - 1$	$N^2 + (\frac{3}{2} + \frac{9}{2} \log_2 N) N + \frac{1}{2}$
RLS 算法	$3 N^2 + 11 N + 9$	$3 N^2 + 7 N + 4$

2.2 RI 算法的收敛性分析

通过对式(2)进行求解, 得到

$$R_k = \sum_{i=0}^k \beta^{k-i} x(i) x^T(i) \quad (9)$$

把式(9)代入式(8), 得到

$$W_k = [I - \mu_k \sum_{i=0}^k \beta^{k-i} x(i) x^T(i)] W_{k-1} + \mu P_k \quad (10)$$

在式(10)中, 令

$$(i) x^T(i) = R_{xx} + \delta R(i) \quad (11)$$

式中, $\delta R(i)$ 是自相关矩阵中随机的一部分. 把式(11)代入式(9), 经过一系列运算操作, 我们得出

$$A_K = \mu_0 R_{xx} + \tilde{A}_K \quad (12)$$

其中,

$$A_k = \mu_k R_k$$

$$\mu_k = \frac{\mu_0(1-\beta)}{1-\beta^{k+1}}$$

$$\delta \tilde{A}_k = \left[\frac{\mu_0(1-\beta)}{1-\beta^{k+1}} \right] \sum_{i=0}^k \beta^{k-i} \delta R(i)$$

按照互相关向量中的相同步骤, 得出

$$\mu_k p_k = \mu_0 \bar{p} + \delta p_k \quad (13)$$

其中 $\delta \tilde{p}_k = (\frac{\mu_0(1-\beta)}{1-\beta^{k+1}}) \sum_{i=0}^k \beta^{k-i} \delta p(i)$ 和 $\delta p(i)$ 是互相关向量中随机的一部分. 然后令

$$\bar{W}_k = \bar{W}_k + \delta \tilde{W}_k \quad (14)$$

其中 $\bar{w}_k = E\{w_k\}$ 和 $\delta \tilde{w}_k$ 是 w_k 中的随机一部分. 把式(12)、(13) 和(14) 代入式(10), 得

$$\bar{w}_k + \delta \tilde{w}_k = [I - \mu_0 R_{xx}] (\bar{w}_{k-1} + \delta \tilde{w}_{k-1}) - \delta \tilde{A}_K (\bar{W}_{K-1} + \delta \tilde{W}_{K-1}) + \mu_0 \bar{P} + \delta \tilde{P}_K \quad (15)$$

由于式(15) 由确定性和随机性部分组成, 它可以被分别写为

$$\bar{w}_k = [I - \mu_0 R_{xx}] (\bar{w}_{k-1} + \mu_0 \bar{P}) \quad (16)$$

和

$$\delta \tilde{w}_k = [I - \mu_0 R_{xx} + \delta \tilde{A}_k] \delta \tilde{w}_{k-1} + \delta \tilde{P}_k - \delta \tilde{A}_k \bar{w}_{k-1} \quad (17)$$

然后令

$$\bar{w}_k = + \Delta w_k \quad (18)$$

式中, w_0 是 w_k 的最优解. 请记住

$$\bar{P} = R_{xx} w_0 \quad (19)$$

把式(18) 和(19) 代入式(16) 中, 得到

$$\Delta w_k = [I - \mu_0 R_{xx}] \Delta w_{k-1} \quad (20)$$

式(20) 是一个线性时间不变的方程, 它的解为

$$\Delta w_k = [I - \mu_0 R_{xx}]^k \Delta w(0) \quad (21)$$

从式(21) 中, 如果 $|\lambda(1 - \mu_0 R_{xx})| < 1$, 当 $k \rightarrow \infty$ 时, $\Delta w_k \rightarrow 0$, 这就表明了平均意义上系数收敛到它们的最优解.

在式(9) 中, 我们可以看到, 式(17) 可近似简化为

$$\delta \tilde{w}_k = [I - \mu_0 R_{xx}] \delta \tilde{w}_{k-1} \quad (22)$$

那么权重向量的协方差可以被定义为 $Q_k =$

$E\{\delta \tilde{w}_k \delta \tilde{w}_k^T\}$, 计算公式如下:

$$Q_k = [I - \mu_0 R_{xx}]^k Q_0 [I - \mu_0 R_{xx}]^k \quad (23)$$

我们可以看到, 如果 $|\lambda(1 - \mu_0 R_{xx})| < 1$, 当 $k \rightarrow \infty$ 时, $Q_k \rightarrow 0$. 根据前面的结果, 这就保证了它在均方意义上的收敛. 在这里应当指出的是, 式(23) 中的结果意味着稳态协方差的权重向量误差为 0, 我们为了得到式(22) 对其进行了简化.

3 二维的递归反演算法(RI)

式(8)描述了 RI 算法的滤波器抽头向量更新方程. 其二维形式可以概括如下:

$$w_k(m_1, m_2) = [I - \mu_k R_k] w_{k-1}(m_1, m_2) + \mu_k P_k \quad (24)$$

式中, $w_k(m_1, m_2)$ 是尺寸为 $N \times N$ 的二维抽头权重向量 $m_1 = 0, 1, \dots, N-1$, $m_2 = 0, 1, \dots, N-1$, R_k

和 P_k 分别为瞬时的自相关和互相关矩阵. 可以对其进行以递归计算, 公式如下:

$$R_k = \beta R_{k-1} + x(n_1, n_2) x^T(n_1, n_2) \quad (25)$$

和

$$p_k = \beta R_{k-1} + d(n_1, n_2) x(n_1, n_2) \quad (26)$$

式中, 输出量的期望值为 $d(n_1, n_2)$, 滤波器的输入是 $x(n_1, n_2)$. 我们可以利用下面的有序向量对滤波器的输入 $x(n_1, n_2)$ 和抽头权重向量 $w_k(m_1, m_2)$ 进行定义:

$$x(n_1, n_2) = \begin{bmatrix} x(n_1, n_2) \\ \vdots \\ x(n_1, n_2 - N + 1) \\ \vdots \\ x(n_1 - N + 1, n_2) \\ \vdots \\ x(n_1 - N + 1, n_2 - N + 1) \end{bmatrix} \quad (27)$$

$$w_k(m_1, m_2) = \begin{bmatrix} w_k(0, 0) \\ \vdots \\ w_k(0, N-1) \\ \vdots \\ w_k(N-1, 0) \\ \vdots \\ w_k(N-1, N-1) \end{bmatrix} \quad (28)$$

二维递归反演算法的伪代码如下:

```

Step 1: Compute R
        for every pixel repeat steps from 2 to 7
Step 2: Initialize  $w = 3$ 
Step 3: If  $R(i, j) \neq$  Impulse pixel
        go to step 7
Step 4:  $\Delta i, j = \{ h(i_1, j_1) \mid i - (w-1)/2 \leq i_1 \leq i + (w-1)/2, \\ j - (w-1)/2 \leq j_1 \leq j + (w-1)/2 \}$ 
        /*  $b =$  no. of black pixels in the window
         $w =$  no. of white pixels in the window */
Step 5: If  $\Delta i, j \neq$  NULL
         $p(i, j) = \text{mean}(\Delta i, j)$ 
         $d(i, j) = |h(i, j) - p(i, j)|$ 
        else if ( $w < w_{\max}$ )
             $w = w + 2$ 
            goto step 4
        else
            if ( $b > w$ )
                 $h(i, j) = 0$ 
            else
                 $h(i, j) = 255$ 
Step 7: Goto next pixel
Step 8: Calculate threshold  $t$ , from detailed coefficient matrix  $d$ 
        for every pixel
Step 9: If ( $d(i, j) > t$ )
         $h(i, j) = p(i, j)$ 

```

在二维图形中的数据重用矩形配置见图 1.

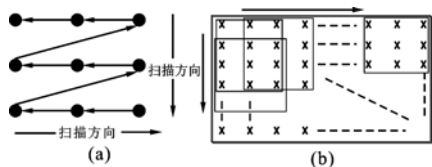


图1 在二维图形中的数据重用矩形配置

4 仿真结果

4.1 图像解卷积处理

盲图像解卷积处理是在不知真实图像或退化过程时,从退化观察方面重新构造原图的问题。

在数学上,可以将其表示为

$$g(x, y) = f(x, y) * h(x, y) \quad (29)$$

式中, $*$ 代表二维卷积运算, $g(x, y)$ 代表模糊的图像, $f(x, y)$ 代表真实的图像, $h(x, y)$ 代表给出的点扩散函数(PSF)。

$$h(x, y) = \begin{bmatrix} -0.035 & -0.65 & -0.35 \\ 0.45 & 0.09 & 0.45 \\ 0.13 & -0.65 & 0.13 \end{bmatrix} \quad (30)$$

图像解卷积实验仿真是在 MATLAB 7.0 平台下进行,观察比较该二维 RI 算法和二维 RLS 算法的性能。遗忘因子是非常接近 1 的,且参数 β 的最佳取值范围为 $0.9990 \leq \beta \leq 1.000$,因此在该实验仿真过程中,设二维 RI 算法的遗忘因子为 $\beta = 0.999$,二维 RLS 算法的 $\beta = 0.999$ 。



图2 不同算法处理图像恢复结果比较

从图 2 可以看出,和使用二维 RLS 算法恢复的图像相比,利用本文提出的算法恢复的图像具有较清晰的边缘。从 PSNR 方面来说,使用该算法得到的图像有 25.64 分贝,而使用二维 RLS 算法得出的图像只有 25.04 分贝。此外,和二维 RLS 算法不同的是,该算法不需要更新反演相关矩阵,这就使该算法具有更稳定的性能。

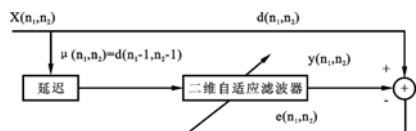


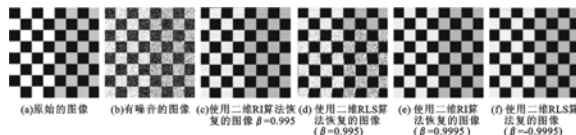
图3 二维自适应线谱增强器的框图

4.2 二维自适应线谱增强器

实验仿真是在 MATLAB 7.0 平台环境下搭建, RI 算法在式(1)中被应用到了二维的自适应线谱增强器。在图 3 中, $x(n_1, n_2)$ 是带有噪声的原始图像,

$\mu(n_1, n_2)$ 是所需响应的延迟信号。

在这里,使用 300×300 检测板图像代替“Lena”的图像。参数 β 的最佳取值范围为 $0.9990 \leq \beta \leq 1.000$,参考文献[7]中仿真实验,分别仿真了当 β 取值变化时两种算法的实现结果。从图 4 看出,随 β 变大,两种算法的恢复效果均随之变好,且通过检查恢复,和二维 RLS 算法相比,二维 RI 算法恢复的图像更具清晰的边缘。对于图 4(e)、(f)来说,在峰值信噪比方面,本文提出的算法有 27.1 分贝,而二维 RLS 算法有 25.4 分贝。

图4 随 β 值变化时不同算法处理图像恢复结果比较

5 结束语

本文提出一个新的二维递归反演自适应滤波算法。利用该算法,可在二维平面上沿水平和垂直方向更新滤波器系数。方法使用可变步长,避免使用系数更新方程中的反演自相关矩阵,从而使性能更稳定。模拟结果表明,该二维 RI 算法在图像解卷处理方面的性能比二维 RLS 算法有所改善(改善了 0.6 分贝)。在自适应谱线增强应用中,二维 RI 的优势更明显。对这两种算法进行收敛后,模拟结果表明,该二维 RI 算法比二维 RLS 算法拥有更好的性能(改善了 1.7 分贝)。此外,和二维 RLS 算法相比,该算法具有相对较低的计算复杂性。

参考文献:

- [1] 彭静玉. 一种新的自适应音频水印算法[J]. 计算机应用与软件, 2011(5): 285-287.
- [2] 靳庆贵, 梁国龙. 非负矩阵分解的分层最小二乘快速算法研究[J]. 计算机仿真, 2012(11): 174-179.
- [3] 简献忠, 刘树明, 李菲, 等. 基于改进 NLMS 算法的自适应滤波算法[J]. 微计算机信息, 2012(5): 27-28.
- [4] 倪元敏, 王军. 基于投影算法的二维自适应滤波器[J]. 重庆理工大学学报: 自然科学版, 2008(8): 112-118.
- [5] 李莲, 熊倩飞. 基于自适应阈值的图像去噪新算法[J]. 微电子学与计算机, 2013, 30(4): 83-86.
- [6] Sanubari J, Tokuda K. RLS-type two-dimensional adaptive filter with a t-distribution assumption [J]. Signal Process, 2008, 80(12): 2483-2495.
- [7] 王阳, 吴成茂, 梁睿. 基于混沌选择的自适应图像加密算法[J]. 微电子学与计算机, 2013, 30(5): 132-136.

作者简介:

赖锦辉 女, (1977-), 硕士, 讲师. 研究方向为网络、算法、人工智能等。