

Regularized image restoration based on preconditioned conjugate gradient method

Ling Meng¹ and Bing Zhang²

Abstract Image restoration can be attributed to solving a linear systems and conjugate gradient method is an effective iteration algorithm for solving various linear systems. However the convergence rate of CG method is determined by condition number of coefficient matrix. The level1 and level2 preconditioner were used to reduce the condition number of coefficient matrix and to accelerate the convergence rate. Simulation experiment show that with same iteration times the PCG method reached better results than other method.

Keywords Image restoration, Preconditioned matrix, Conjugate gradient method, Toeplitz matrices, Preconditioned conjugate gradient method.

1 Introduction

The discrete image degradation model can be described as the linear systems as follows(see [2,9])

$$vec(d) = T \square vec(f) + vec(\eta) \quad (1)$$

where T is a block Toeplitz matrix with Toeplitz blocks, $vec(d)$, $vec(g)$ and $vec(\eta)$ are vectors which the discrete blurred image d , discrete original image f and noise η are reordered with column lexicographical order respectively. To

1..L.Meng

School of science , Communication University of China, Beijing.

e-mail: mengling@cuc.edu.cn

2.B. Zhang(✉)

School of science , Communication University of China, Beijing.

e-mail:zhangbin628@cuc.edu.cn

solve this ill-posed linear systems, we use the Tikhonov regularization method and obtain the following systems

$$(T^*T + \alpha L)f = T^*d \quad (2)$$

where L is symmetric positive definite block matrix. $\alpha > 0$ is the regularization parameter. To solve linear systems(2) is equivalent to solve the minimum value of the quadratic function :

$$\min_f J(f) = \frac{1}{2} f^T A f + b^T f + c, \quad A = T^*T + \alpha L, b = T^*d \quad (3)$$

The minimization of quadratic function was solved by conjugate gradient(CG) method [1] in optimization theory [2].The convergence rate of this iteration method is determined by condition number of A . The distribution of singular value of A is moer concentrated, the condition number of A is smaller and the convergence rate is fast. Preconditioned conjugate gradient (PCG) method [1,3] is an improved algothm for the CG method. We introduced the preconditioner M first to enable $M^{-1}A$ has more concentrated singular value distribution, then substituted A with $M^{-1}A$ in iteration algothm to accelerate the convergence rate.

2 Block Circulant Preconditioning

2.1 Level 1 Block Circulant Preconditioning

Assume T is block Toeplitz with Toeplitz blocks(bttb) with representation

$$T = \begin{bmatrix} T_0 & T_{-1} & \cdots & T_{2-n_y} & T_{1-n_y} \\ T_1 & T_0 & & & T_{2-n_y} \\ \vdots & & \ddots & & \vdots \\ T_{n_y-2} & & & T_0 & T_{-1} \\ T_{n_y-1} & T_{n_y-2} & \cdots & T_1 & T_0 \end{bmatrix}$$

$$T_j = \begin{bmatrix} t_{0,j} & t_{-1,j} & \cdots & t_{2-n,j} & t_{1-n,j} \\ t_{1,j} & t_{0,j} & & & t_{2-n,j} \\ \vdots & & \ddots & & \vdots \\ t_{n-2,j} & & & t_{0,j} & t_{-1,j} \\ t_{n-1,j} & t_{n-2,j} & \cdots & t_{1,j} & t_{0,j} \end{bmatrix}$$

$$L = \begin{bmatrix} L_0 & -I & O & \cdots & O \\ -I & L_0 & -I & \cdots & O \\ \vdots & -I & \ddots & \ddots & \vdots \\ O & \ddots & \ddots & L_0 & -I \\ O & O & \cdots & -I & L_0 \end{bmatrix}, \quad L_0 = \begin{bmatrix} 4 & -1 & 0 & \cdots & 0 \\ -1 & 4 & -1 & \cdots & 0 \\ \vdots & -1 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & 4 & -1 \\ 0 & 0 & \cdots & -1 & 4 \end{bmatrix}$$

where I and O represent $n_x \times n_y$ identity matrix and zero matrix respectively. We denote $C_1(T)$ the level 1 block circulant approximation [1] of T . $C_1(T)$ is obtained by replacing each of the blocks T_j by its best circulant approximation [1] $C_1(T)$ and $C_1(T)$ is block Toeplitz with circulant blocks. Similarly, one can compute $C_1(T)$, the Level 1 approximation to L .

Let $A = T * T + \alpha L$, we take $M_1(A)$ as a preconditioner for A

$$M_1(A) = C_1(T^*)C_1(T) + \alpha C_1(L) \quad (4)$$

As $C_1(T_j)$ is a circulant matrix, it can be expressed as

$$C(T_j) = F_x^* \Lambda_j F_x, \quad j = 1 - n_y, \dots, 0, \dots, n_y - 1 \quad (5)$$

where F_x is the $n_x \times n_x$ Fourier matrix, and Λ_j is the diagonal matrix whose entries are the eigenvalues of $C(T_j)$. From this we obtain

$$C_1(T) = (I_y \otimes F_x)^* T(\Lambda) (I_y \otimes F_x) \quad (6)$$

Where I_y denotes the $n_y \times n_y$ identity matrix and $T(\Lambda)$ is the block Toeplitz matrix with the Λ_j as its diagonal blocks

$$T(\Lambda) = \begin{bmatrix} \Lambda_0 & \Lambda_{-1} & \cdots & \Lambda_{1-n_y} \\ \Lambda_1 & \Lambda_0 & \Lambda_{-1} & \vdots \\ \vdots & \ddots & \ddots & \Lambda_{-1} \\ \Lambda_{n_y-1} & \cdots & \Lambda_1 & \Lambda_0 \end{bmatrix}$$

There exists a permutation matrix P , corresponding to a reindexing of unknowns from column lexicographical order to row lexicographical order, for which $P^T T(\Lambda) P$ is a block diagonal. So

$$C_1(T) = (I_y \otimes F_x)^* P^T \text{diag}(D_1, \dots, D_{n_x}) P (I_y \otimes F_x) \quad (7)$$

Where $[D_k]_{i,j} = [\Lambda_{i-j}]_{k,k}$, $1 \leq i, j \leq n_y$, $k = 1, 2, \dots, n_x$. L can also be expressed with a similar form

$$C_1(L) = (I_y \otimes F_x)^* P^T \text{diag}(E_1, \dots, E_{n_x}) P (I_y \otimes F_x) \quad (8)$$

From above representation, we obtained the following expression for the $M_1(A)$

$$M_1(T) = (I_y \otimes F_x)^* P^T \text{diag}(D_k^2 + \alpha E_k) P (I_y \otimes F_x) \quad (9)$$

Now consider the computation of $w = M_1(A)^{-1}$,

$$w = M_1(A)^{-1} v = (I_y \otimes F_x)^* P^T \text{diag}(D_k^2 + \alpha E_k)^{-1} P (I_y \otimes F_x) v \quad (10)$$

where $v = \text{vec}(v)$ and v is an $n_x \times n_y$ array.

The matrix-vector product $\hat{v} = (I_y \otimes F_x) v$ corresponding to applying one-dimensional DFTs to the columns of v . The computation

$\hat{w} = P^T \text{diag}(D_k^2 + \alpha E_k)^{-1} P \hat{v}$ can be carried out by solving linear systems

$$(D_k^2 + \alpha E_k) \hat{w}_{k..} = \hat{v}_{k..}, k = 1, 2, \dots, n. \quad (11)$$

where $\hat{v}_{k..}$ denotes the k th row of \hat{v} , and storing $\hat{w}_{k..}$ as the k th row of

$\hat{w} = \text{array}(\hat{w})$, array is an inverse operator for vec . Finally, the

computation $w = (I_y \otimes F_x)^* \hat{w}$ corresponds to applying inverse DFTs to the columns of $\hat{\omega}$. This yields $\omega = \text{array}(w)$.

2.2 Level 2 Block Circulant Preconditioning

We denote the level 2 block circulant approximation to T by $C_2(T)$. $C_2(T)$ can be obtained by replacing each of the diagonal block matrices D_k by its best circulant approximation $C(D_k)$. Let

$$(C_2(T))_{y,y} = \text{diag}(\hat{d}_1, \dots, \hat{d}_{n_x}) F_y \quad (12)$$

where $\text{diag}(\hat{d}_k)$ denotes the diagonal matrix whose diagonal entries comprise the components of the vector $\hat{d}_k \in \mathbb{C}^{n_y}$. This yields the representation

$$\begin{aligned} C_2(T) &= (I_y \otimes F_x)^* P^T (I_y \otimes F_x)^* \times \text{diag}(\hat{d}_1, \dots, \hat{d}_{n_x}) (I_y \otimes F_y) P (I_y \otimes F_x) \\ &= (F_y \otimes F_x)^* \text{diag}(\hat{d}_1, \dots, \hat{d}_{n_x}) (F_y \otimes F_x) \end{aligned} \quad (13)$$

Then, we obtain the representation for the $M_2(A)$,

$$\begin{aligned} M_2(A) &= C_2(T) * C_2(T) + \alpha C_2(L) \\ &= (F_y \otimes F_x)^* \text{diag}(|\hat{d}_k|^2 + \hat{e}_k) (F_y \otimes F_x) \end{aligned} \quad (14)$$

The computation $w = M_2(A)^{-1} v$ is easy. The matrix-vector

product $\hat{v} = (F_y \otimes F_x) v$ corresponds to applying the two-dimensional DFT

to $v = \text{array}(v)$. Next, to compute $\hat{w} = \text{diag}(|\hat{d}_k|^2 + \hat{e}_k)^{-1} \hat{v}$, take

$\hat{\omega} = \text{array}(\hat{w})$ to consist of columns

$$\hat{\omega}_{:,k} = \hat{v}_{:,k} \cdot / (|\hat{d}_k|^2 + \hat{e}_k), \quad k = 1, \dots, n_y$$

Finally, apply the inverse two-dimensional DFT to $\hat{\omega}$ to obtain $\omega = \text{array}(w)$.

3 Simulation and data analysis

Taking the atmospheric turbulence blur image as an example [7], and the value used for the regularization parameter is $\alpha=0.005$. The image lies on a 256×256 pixel grid. We apply MATLAB programming to conduct simulation experiment. And then compare the numerical performance of the various circulant preconditioning.

In order to simulate blurred image caused by atmospheric turbulence, we choose $\sigma=0.05$ for point spread function $K(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$, and variance noise is 0.001. Blurred image d can be obtained from (1) by using multiplication of block Toeplitz matrix with Toeplitz blocks with vector [1]. The original image(a) and blurred image (b) were showed in Figure.1. The restored images without preconditioner were showed in figure.2, (c) is the restored image with CG method iterating 10 times, (d)



Fig. 1 The original image (a) and blurred image(b)

is restored image iterating 50 times and (e) is the restored image iterating 100 times. Figure .3 are the restored images with level1 preconditioner, (f) is the restored image with PCG method iterating 10 times, (g) is restored image iterating 50 times and (h) is the restored image iterating 100 times. Figure .4 are the restored images with level2 preconditioner, (i) is the restored image with PCG method iterating 10 times, (j) is restored image iterating 50 times and (k) is the restored image iterating 100 times.

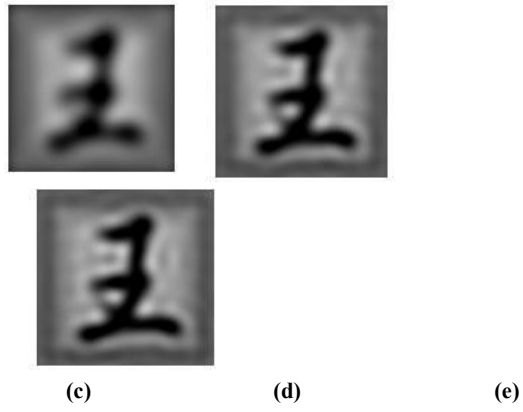


Fig. 2 Restored images without preconditioner

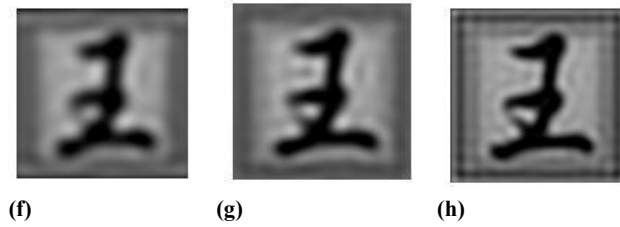


Fig. 3 Restored images with level1 preconditioner

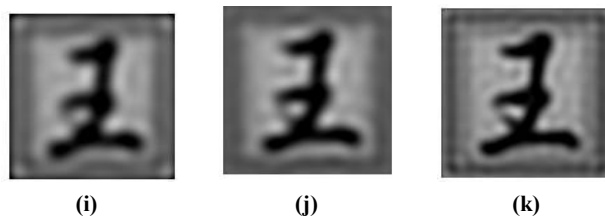


Fig. 4 Restored images with level2 preconditioner

Table 1 PSNR of restored images with different methods

Iteration times	PSNR of the restored images of different methods		
	<i>CG</i>	<i>Level1</i>	<i>Level2</i>
10	17.9596	18.1189	18.4710
50	18.3075	18.5688	18.4688
100	17.9077	18.6541	18.6548

From table.1 we can see that the restored images with PCG methods have higher PSNR than restored images with CG method, and PCG method with level 2 preconditioner is better than other methods.

4 Acknowledgment

This work was supported by a grants from the National Natural Science Foundation of China (No. 61071148)

Reference

- 1.Vogel .C.R. Computational methods for inverse problems, Montana State University, Bozeman, Montana, 2002.
- 2.Changfeng Ma. Optimization method and matlab program [M]. Beijing: Science Publisher, 2010.
- 3.Chen Pu, Tingsong Xiao, Shuming Sun, Mingwu Yuan. Realization of preconditioned conjugate gradient method and improvement [J].Beijing Peking University ("Engineering Mechanics "supplement "). 1998.
- 4.Strang .G.A proposal for Toeplitz matrix calculations, Studies in Applied Mathematics, 74(1986) ,pp,171-176.
- 5.Hanke .M and Nagy .J.G. Toeplitz approximate inverse preconditioner for banded Toeplitz matrices, Numerical Algorithms, 7(1994),pp.183-199.
- 6.Raymond chan .H, Michael K.N. Conjugate Gradient Methods for Toeplitz Systems , September 1995,TR-CS-95-07.
- 7.Qingfei Li, Zhichao Zhu, Shuai Fang. Restoration of atmospheric turbulence-degraded image analysis [J], Journal of HeFei University of Technology (NATURAL SCIENCE EDITION), 2011, 34 (1):80-82.
- 8.Min Yao. Digital image processing [M]. Beijing: Mechanical Industry Press, 2012.4
- 9.Yanfei Wang. Calculation method of inversion problem and its application [M] ,Beijing: Higher Education Press,2007
- 10.Chan.R. Circulant preconditioners for Hermitian Toeplitz systems, SIAM J. Matrix Anal. Appl,10(1989), 542-550.
11. Jain.A.K. Fundamentals of Digital Image Processing, Prentice-Hall, New York, 1989.