SIAM J. SCI. COMPUT.
Vol. 21, No. 3, pp. 851–866

A FAST ALGORITHM FOR DEBLURRING MODELS WITH NEUMANN BOUNDARY CONDITIONS*

MICHAEL K. NG†, RAYMOND H. CHAN‡, AND WUN-CHEUNG TANG§

Abstract. Blur removal is an important problem in signal and image processing. The blurring matrices obtained by using the zero boundary condition (corresponding to assuming dark background outside the scene) are Toeplitz matrices for one-dimensional problems and block-Toeplitz–Toeplitz-block matrices for two-dimensional cases. They are computationally intensive to invert especially in the block case. If the periodic boundary condition is used, the matrices become (block) circulant and can be diagonalized by discrete Fourier transform matrices. In this paper, we consider the use of the Neumann boundary condition (corresponding to a reflection of the original scene at the boundary). The resulting matrices are (block) Toeplitz-plus-Hankel matrices. We show that for symmetric blurring functions, these blurring matrices can always be diagonalized by discrete cosine transform matrices. Thus the cost of inversion is significantly lower than that of using the zero or periodic boundary conditions. We also show that the use of the Neumann boundary condition provides an easy way of estimating the regularization parameter when the generalized cross-validation is used. When the blurring function is nonsymmetric, we show that the optimal cosine transform preconditioner of the blurring matrix is equal to the blurring matrix generated by the symmetric part of the blurring function. Numerical results are given to illustrate the efficiency of using the Neumann boundary condition.

Key words. deblurring, boundary conditions, Toeplitz matrix, circulant matrix, Hankel matrix, cosine transform

AMS subject classifications. 65F10, 65Y20

PII. S1064827598341384

1. Introduction.

A fundamental issue in signal and image processing is blur removal. The signal or image obtained from a point source under the blurring process is called the impulse response function or the point spread function. The observed signal or image g is just the convolution of this blurring function h with the "true" signal or image f. The deblurring problem is to recover f from the blurred function g given the blurring function h. This basic problem appears in many forms in signal and image processing [2, 5, 12, 14].

In practice, the observed signal or image g is of finite length (and width) and we use it to recover a finite section of f. Because of the convolution, g is not completely determined by f in the same domain where g is defined. More precisely, if a blurred signal g is defined on the interval $[a, b]$, say, then it is not completely determined by the values of the true signal f on $[a, b]$ only. It is also affected by the values of f close to the boundary of $[a, b]$ because of the convolution. The size of the interval that affects g depends on the support of the blurring function h. Thus in solving f from a finite length g, we need some assumptions on the values of f outside the domain where g is defined. These assumptions are called the boundary conditions.

[*] Received by the editors July 1, 1998; accepted for publication (in revised form) February 7, 1999; published electronically November 4, 1999.
http://www.siam.org/journals/sisc/21-3/34138.html

[†] Department of Mathematics, The University of Hong Kong, Pokfulam Road, Hong Kong (mng@maths.hku.hk). The research of this author was supported in part by HKU CRCG grant 10201939.

[‡] Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong (rchan@math.cuhk.edu.hk). The research of this author was supported in part by Hong Kong Research Grants Council grant CUHK 4207/97P and CUHK DAG grant 2060143.

[§] Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong (wctang@math.cuhk.edu.hk).

The natural and classical approach is to use the zero (Dirichlet) boundary condition [2, pp. 211–220]. It assumes that the values of f outside the domain of consideration are zero. This results in a blurring matrix which is a Toeplitz matrix in the one-dimensional case and a block-Toeplitz–Toeplitz-block matrix in the two-dimensional case; see [2, p. 71]. However, these matrices are known to be computationally intensive to invert, especially in the two-dimensional case; see [2, p. 126]. Also, ringing effects will appear at the boundary if the data are indeed not close to zero outside the domain.

One way to alleviate the computational cost is to assume the periodic boundary condition; i.e., data outside the domain of consideration are exact copies of data inside [12, p. 258]. The resulting blurring matrix is a circulant matrix in the one-dimensional case and a block-circulant–circulant-block matrix in the two-dimensional case. These matrices can be diagonalized by discrete Fourier matrices and hence their inverses can easily be found by using the fast Fourier transforms (FFTs); see [12, p. 258]. However, ringing effects will also appear at the boundary unless f is close to periodic, and that is not common in practice.

In the image processing literature, other methods have also been proposed to assign boundary values; see Lagendijk and Biemond [18, p. 22] and the references therein. For instance, the boundary values may be fixed at a local image mean, or they can be obtained by a model-based extrapolation. In this paper, we consider the use of the Neumann (reflective) boundary condition for image restoration. It sets the data outside the domain of consideration as reflection of the data inside. The Neumann boundary condition has been studied in image restoration [21, 3, 18] and in image compression [25, 20]. In image restoration, the boundary condition restores a balance that is lost by ignoring the energy that spreads outside of the area of interest [21] and also minimizes the distortion at the borders caused by deconvolution algorithms [3]. This approach can also eliminate the artificial boundary discontinuities contributed to the energy compaction property that is exploited in transform image coding [20].

The use of the Neumann boundary condition results in a blurring matrix that is a Toeplitz-plus-Hankel matrix in the one-dimensional case and a block Toeplitz-plus-Hankel matrix with Toeplitz-plus-Hankel blocks in the two-dimensional case. Although these matrices have more complicated structures, we show that they can always be diagonalized by the discrete cosine transform matrix provided that the blurring function h is symmetric. Thus their inverses can be obtained by using fast cosine transforms (FCTs). Because an FCT requires only real multiplications and can be done at half of the cost of an FFT (see [23, pp. 59–60]), inversion of these matrices is faster than that of those matrices obtained from either the zero or the periodic boundary conditions.

We also show that the use of the Neumann boundary condition provides an easy way to estimate the regularization parameter when using the generalized cross-validation. We remark that blurring functions are usually symmetric; see [14, p. 269]. However, in the case where the blurring function is nonsymmetric, we show that the optimal cosine transform preconditioner [6] of the blurring matrix is generated by the symmetric part of the blurring function. Thus if the blurring function is close to symmetric, the optimal cosine transform preconditioner should be a good preconditioner.

The outline of the paper is as follows. In section 2, we introduce the three different boundary conditions. In section 3, we show that symmetric blurring matrices obtained from the Neumann boundary condition can always be diagonalized by the discrete cosine transform matrix. In section 4, we show that using the Neumann boundary condition, the generalized cross-validation estimate of the regularization parameter can be done in a straightforward way. In section 5, we give the construction of the optimal cosine transform preconditioners for the matrices generated by nonsymmetric blurring functions. In section 6, we illustrate by numerical examples from image restorations that our algorithm is efficient. Concluding remarks are given in section 7.

2. The deblurring problem.

For simplicity, we begin with the one-dimensional deblurring problem. Consider the original signal

$$\tilde{\mathbf{f}} = (\ldots, f_{-m+1}, \ldots, f_0, f_1, \ldots, f_n, f_{n+1}, \ldots, f_{n+m}, \ldots)^t$$

and the blurring function given by

$$\mathbf{h} = (\ldots, 0, 0, h_{-m}, h_{-m+1}, \ldots, h_0, \ldots, h_{m-1}, h_m, 0, 0, \ldots)^t. \tag{2.1}$$

The blurred signal is the convolution of $\mathbf{h}$ and $\tilde{\mathbf{f}}$, i.e., the ith entry g_i of the blurred signal is given by

$$g_i = \sum_{j=-\infty}^{\infty} h_{i-j} f_j. \tag{2.2}$$

The deblurring problem is to recover the vector $\mathbf{f} = (f_1, \ldots, f_n)^t$ given the blurring function $\mathbf{h}$ and a blurred signal $\mathbf{g} = (g_1, \ldots, g_n)^t$ of finite length. From (2.2), we have

$$\begin{pmatrix} h_m & \cdots & h_0 & \cdots & h_{-m} & & & & & & \\ & h_m & & h_0 & & h_{-m} & & & 0 & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & & & & \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots & & & \\ & 0 & & & h_m & & h_0 & & h_{-m} & & \\ & & & & & h_m & \cdots & h_0 & \cdots & h_{-m} & \end{pmatrix} \begin{pmatrix} f_{-m+1} \\ f_{-m+2} \\ \vdots \\ f_0 \\ f_1 \\ \vdots \\ f_n \\ f_{n+1} \\ \vdots \\ f_{n+m-1} \\ f_{n+m} \end{pmatrix} = \begin{pmatrix} g_1 \\ \vdots \\ g_n \end{pmatrix}. \tag{2.3}$$

Thus the blurred signal $\mathbf{g}$ is determined not by $\mathbf{f}$ only, but also by

$$(f_{-m+1}, \ldots, f_0, f_1, \ldots, f_n, f_{n+1}, \ldots, f_{n+m})^t.$$

The linear system (2.3) is underdetermined. To overcome this, we make certain assumptions (called boundary conditions) on the unknown data $f_{-m+1}, \ldots, f_0$ and $f_{n+1}, \ldots, f_{n+m}$ so as to reduce the number of unknowns.

Before we discuss the boundary conditions, let us first rewrite (2.3) as

$$T_l \mathbf{f}_l + T\mathbf{f} + T_r \mathbf{f}_r = \mathbf{g}, \tag{2.4}$$

where

$$T_l = \begin{pmatrix} h_m & \cdots & h_1 \\ & \ddots & \ddots \\ & & h_m \\ & & \\ 0 & & \end{pmatrix}, \quad \mathbf{f}_l = \begin{pmatrix} f_{-m+1} \\ f_{-m+2} \\ \vdots \\ f_{-1} \\ f_0 \end{pmatrix}, \tag{2.5}$$

$$T = \begin{pmatrix} h_0 & \cdots & h_{-m} & & 0 \\ \vdots & \ddots & \ddots & \ddots & \\ h_m & \ddots & \ddots & \ddots & h_{-m} \\ & \ddots & \ddots & \ddots & \vdots \\ 0 & & h_m & \cdots & h_0 \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n \end{pmatrix}, \tag{2.6}$$

$$T_r = \begin{pmatrix} & & 0 \\ & & \\ h_{-m} & & \\ \vdots & \ddots & \\ h_{-1} & \cdots & h_{-m} \end{pmatrix}, \quad \text{and} \quad \mathbf{f}_r = \begin{pmatrix} f_{n+1} \\ f_{n+2} \\ \vdots \\ f_{n+m-1} \\ f_{n+m} \end{pmatrix}. \tag{2.7}$$

2.1. The zero (Dirichlet) boundary condition. The zero (or Dirichlet) boundary condition assumes that the signal outside the domain of the observed vector $\mathbf{g}$ is zero [2, pp. 211–220], i.e.,

$$\mathbf{f}_l = \mathbf{f}_r = \mathbf{0},$$

the zero vector. The matrix system in (2.4) becomes

$$T\mathbf{f} = \mathbf{g}. \tag{2.8}$$

We see from (2.6) that the coefficient matrix T is a Toeplitz matrix.

There are many iterative or direct Toeplitz solvers that can solve the Toeplitz system (2.8) with costs ranging from $O(n \log n)$ to $O(n^2)$ operations; see, for instance, [19, 16, 1, 7]. In the two-dimensional case, the resulting matrices will be block-Toeplitz–Toeplitz-block matrices. Inversion of these matrices is known to be very expensive, e.g., the fastest direct Toeplitz solver is of $O(n^4)$ operations for an n^2-by-n^2 block-Toeplitz–Toeplitz-block matrix; see [17].

2.2. The periodic boundary condition. For practical applications, especially in the two-dimensional case, where we need to solve the system efficiently, one usually resorts to the periodic boundary condition. This amounts to setting

$$f_j = f_{n-j} \quad \text{for all } j$$

in (2.3); see [12, p. 258]. The matrix system in (2.4) becomes

$$B\mathbf{f} = [(0 \mid T_l) + T + (T_r \mid 0)]\,\mathbf{f} = \mathbf{g}, \tag{2.9}$$

where $(0 \mid T_l)$ and $(T_r \mid 0)$ are n-by-n Toeplitz matrices obtained by augmenting $(n-m)$ zero columns to T_l and T_r, respectively.

The most important advantage of using the periodic boundary condition is that B so obtained is a circulant matrix. Hence B can be diagonalized by the discrete Fourier matrix and (2.9) can be solved by using three FFTs (one for finding the eigenvalues of the matrix B and two for solving the system; cf. (3.5) below). Thus the total cost is of $O(n \log n)$ operations.

In the two-dimensional case, the blurring matrix is a block-circulant–circulant-block matrix and can be diagonalized by the two-dimensional FFTs (which are tensor-products of one-dimensional FFTs) in $O(n^2 \log n)$ operations.

2.3. The Neumann boundary condition.

For the Neumann boundary condition, we assume that the data outside $\mathbf{f}$ are a reflection of the data inside $\mathbf{f}$. More precisely, we set

$$\left\{\begin{array}{ccc} f_0 & = & f_1 \\ \vdots & \vdots & \vdots \\ f_{-m+1} & = & f_m \end{array}\right. \quad \text{and} \quad \left\{\begin{array}{ccc} f_{n+1} & = & f_n \\ \vdots & \vdots & \vdots \\ f_{n+m} & = & f_{n-m+1} \end{array}\right.$$

in (2.3). Thus (2.4) becomes

$$A\mathbf{f} = [(0 \mid T_l)J + T + (T_r \mid 0)J]\,\mathbf{f} = \mathbf{g}, \tag{2.10}$$

where J is the n-by-n reversal matrix.

We remark that the coefficient matrix A in (2.10) is neither Toeplitz nor circulant. It is a Toeplitz-plus-Hankel matrix. Although these matrices have more complicated structures, we will show in section 3 that the matrix A can always be diagonalized by the discrete cosine transform matrix provided that the blurring function $\mathbf{h}$ is symmetric, i.e., $h_j = h_{-j}$ for all j in (2.1). It follows that (2.10) can be solved by using three FCTs in $O(n \log n)$ operations; see (3.5) below. This approach is computationally attractive as FCT requires only real operations and is about twice as fast as the FFT; see [23, pp. 59–60]. Thus solving a problem with the Neumann boundary condition is twice as fast as solving a problem with the periodic boundary condition.

We will establish similar results in the two-dimensional case, where the blurring matrices will be block Toeplitz-plus-Hankel matrices with Toeplitz-plus-Hankel blocks.

3. Diagonalization of the Neumann blurring matrices.

3.1. One-dimensional problems.

We first review some definitions and properties of the discrete cosine transform matrix. Let C be the n-by-n discrete cosine transform matrix with entries

$$[C]_{i,j} = \sqrt{\frac{2-\delta_{i1}}{n}} \cos\left(\frac{(i-1)(2j-1)\pi}{2n}\right), \qquad 1 \le i, j \le n, \tag{3.1}$$

where δ_{ij} is the Kronecker delta; see [14, p. 150]. We note that C is orthogonal, i.e., $C^tC = CC^t = I$. Also, for any n-vector $\mathbf{v}$, the matrix-vector multiplications $C\mathbf{v}$ and $C^t\mathbf{v}$ can be computed in $O(n \log n)$ real operations by FCTs; see [23, pp. 59–60].

Let $\mathcal{C}$ be the space containing all matrices that can be diagonalized by C, i.e.,

$$\mathcal{C} = \{C^t \Lambda C \mid \Lambda \text{ is an } n\text{-by-}n \text{ real diagonal matrix}\}. \tag{3.2}$$

Let $Q = C^t \Lambda C \in \mathcal{C}$. By multiplying $\mathbf{e}_1 = (1, 0, \ldots, 0)^t$ by both sides of $CQ = \Lambda C$, we see that the eigenvalues $[\Lambda]_{i,i}$ of Q are given by

$$[\Lambda]_{i,i} = \frac{[C\mathbf{e}_1]_i}{[C(Q\mathbf{e}_1)]_i}, \qquad i = 1, \ldots, n. \tag{3.3}$$

Hence, the eigenvalues of Q can be obtained by taking an FCT of the first column of Q. In particular, any matrix in $\mathcal{C}$ is uniquely determined by its first column.

Next we give a characterization of the class of matrices $\mathcal{C}$. Let us define the shift of any vector $\mathbf{v} = (v_0, \ldots, v_{n-1})^t$ as $\sigma(\mathbf{v}) \equiv (v_1, v_2, \ldots, v_{n-1}, 0)^t$. Define $T(\mathbf{v})$ to be the n-by-n symmetric Toeplitz matrix with $\mathbf{v}$ as the first column and $H(\mathbf{x}, \mathbf{y})$ to be the n-by-n Hankel matrix with $\mathbf{x}$ as the first column and $\mathbf{y}$ as the last column.

LEMMA 3.1 (see Chan, Chan, and Wong [6], Kailath and Olshevsky [15], Martucci [22], and Sanchez et al. [24]). *Let $\mathcal{C}$ be the class of matrices that can be diagonalized by the discrete cosine transform matrix C. Then*

$$\mathcal{C} = \{T(\mathbf{v}) + H(\sigma(\mathbf{v}), J\sigma(\mathbf{v})) \mid \mathbf{v} = (v_0, \ldots, v_{n-1})^t \in \mathbb{R}^n\}.$$

It follows from Lemma 3.1 that matrices that can be diagonalized by C are some special Toeplitz-plus-Hankel matrices.

THEOREM 3.2. *Let the blurring function h be symmetric, i.e., $h_i = h_{-i}$ for all i. Then the matrix A given in* (2.10) *can be written as*

$$A = T(\mathbf{u}) + H(\sigma(\mathbf{u}), J\sigma(\mathbf{u})), \tag{3.4}$$

where $\mathbf{u} = (h_0, h_1, \ldots, h_m, 0, \ldots, 0)^t$. In particular, A can be diagonalized by C.

Proof. By (2.10), $A = (0 \mid T_l)J + T + (T_r \mid 0)J$. By (2.6), it is clear that T is equal to $T(\mathbf{u})$. From the definitions of T_l and T_r in (2.5) and (2.7), it is also obvious that

$$(0 \mid T_l) + (T_r \mid 0) = T(J\sigma(\mathbf{u})).$$

Hence

$$(0 \mid T_l)J + (T_r \mid 0)J = H(\sigma(\mathbf{u}), J\sigma(\mathbf{u})). \qquad \square$$

By Theorem 3.2, the solution $\mathbf{f}$ of (2.10) is given by

$$\mathbf{f} = C\Lambda^{-1}C^t\mathbf{g}, \tag{3.5}$$

where Λ is the diagonal matrix holding the eigenvalues of A. By (3.3), Λ can be obtained in one FCT. Hence $\mathbf{f}$ can be obtained in three FCTs.

We remark that from (3.4), it is straightforward to construct the Neumann blurring matrix A from the Dirichlet blurring matrix $T = T(\mathbf{u})$ in (2.6). All we need is to reflect the first column of T to get the Hankel matrix $H(\sigma(\mathbf{u}), J\sigma(\mathbf{u}))$ and add it to T. Clearly the storage requirements of both matrices A and T are the same—we need to store only the first column.

3.2. Two-dimensional problems. The results of section 3.1 can be extended in a natural way to two-dimensional image restoration problems. In this case, one is concerned with solving a least squares problem similar to that in (2.3), except that the matrix is now a block matrix. For the zero boundary condition, the resulting blurring matrix is a block-Toeplitz–Toeplitz-block matrix of the form

$$T = \begin{pmatrix} T^{(0)} & \cdots & T^{(-m)} & & 0 \\ \vdots & \ddots & \ddots & \ddots & \\ T^{(m)} & \ddots & \ddots & \ddots & T^{(-m)} \\ & \ddots & \ddots & \ddots & \vdots \\ 0 & & T^{(m)} & \cdots & T^{(0)} \end{pmatrix}, \tag{3.6}$$

where each block $T^{(j)}$ is a Toeplitz matrix of the form given in (2.7). The first column and row of T in (3.6) are completely determined by the blurring function of the blurring process.

With the Neumann boundary condition, the resulting matrix A is a block Toeplitz-plus-Hankel matrix with Toeplitz-plus-Hankel blocks. More precisely,

$$\begin{aligned} A = {} & \begin{pmatrix} A^{(0)} & \cdots & A^{(-m)} & & 0 \\ \vdots & \ddots & \ddots & \ddots & \\ A^{(m)} & \ddots & \ddots & \ddots & A^{(-m)} \\ & \ddots & \ddots & \ddots & \vdots \\ 0 & & A^{(m)} & \cdots & A^{(0)} \end{pmatrix} \\ & + \begin{pmatrix} A^{(1)} & \cdots & A^{(m-1)} & & 0 \\ \vdots & \iddots & & & \\ A^{(m-1)} & & & & A^{(-m+1)} \\ & & & \iddots & \vdots \\ 0 & & A^{(-m+1)} & \cdots & A^{(1)} \end{pmatrix} \end{aligned} \tag{3.7}$$

with each block $A^{(j)}$ being an n-by-n matrix of the form given in (2.10). We note that the $A^{(j)}$ in (3.7) and the $T^{(j)}$ in (3.6) are related by (3.4). Thus again it is straightforward to construct the blurring matrix A from the matrix T or from the blurring function directly. Obviously, storage requirements of A and T are the same.

We next show that for a symmetric blurring function, the blurring matrix A in (3.7) can be diagonalized by the two-dimensional discrete cosine transform matrix. Hence inversion of A can be done by using only three two-dimensional FCTs.

THEOREM 3.3. *If the blurring function $h_{i,j}$ is symmetric, i.e.,*

$$h_{i,j} = h_{i,-j} = h_{-i,j} = h_{-i,-j}$$

for all i and j, then A can be diagonalized by the two-dimensional discrete cosine transform matrix $C \otimes C$, where $\otimes$ is the tensor product.

Proof. We note that $C \otimes C = (C \otimes I)(I \otimes C)$. Since each block $A^{(j)}$ in (3.7) is of the form given by (3.4), by Theorem 3.2, $A^{(j)}$ can be diagonalized by C, i.e.,

$$CA^{(j)}C^t = \Lambda^{(j)}, \qquad j = 1, \ldots, m.$$

It follows that

$$\begin{aligned} (C \otimes C)A(C^t \otimes C^t) &= (C \otimes I)(I \otimes C)A(I \otimes C^t)(C^t \otimes I) \\ &= (C \otimes I)\Lambda(C^t \otimes I), \end{aligned}$$

where

$$\Lambda = \begin{pmatrix} \Lambda^{(0)} & \cdots & \Lambda^{(m)} & & 0 \\ \vdots & \ddots & \ddots & \ddots & \\ \Lambda^{(m)} & \ddots & \ddots & \ddots & \Lambda^{(m)} \\ & \ddots & \ddots & \ddots & \vdots \\ 0 & & \Lambda^{(m)} & \cdots & \Lambda^{(0)} \end{pmatrix} + \begin{pmatrix} \Lambda^{(1)} & \cdots & \Lambda^{(m-1)} & & 0 \\ \vdots & \iddots & & & \\ \Lambda^{(m-1)} & & & & A^{(m-1)} \\ & & & \iddots & \vdots \\ 0 & & \Lambda^{(m-1)} & \cdots & \Lambda^{(0)} \end{pmatrix}. \tag{3.8}$$

Let P be the permutation matrix that satisfies

$$[P^t \Lambda P]_{i,j;k,\ell} = [\Lambda]_{k,\ell;i,j}, \qquad 1 \le i,j \le n,\ 1 \le k,\ell \le n;$$

i.e., the (i,j)th entry of the (k,ℓ)th block in Λ is permuted to the (k,ℓ)th entry of the (i,j)th block. Then we have $P^t(C \otimes I)P = (I \otimes C)$ and

$$P^t \Lambda P = \tilde{A} = \begin{pmatrix} \tilde{A}^{(1)} & & & 0 \\ & \tilde{A}^{(2)} & & \\ & & \ddots & \\ 0 & & & \tilde{A}^{(n)} \end{pmatrix}.$$

From (3.8), each matrix $\tilde{A}^{(j)}$ has the same form as A in (3.4). In particular, for all j, $C\tilde{A}^{(j)}C^t = \tilde{\Lambda}^{(j)}$, a diagonal matrix. Thus

$$\begin{aligned} (C \otimes C)A(C^t \otimes C^t) &= (C \otimes I)\Lambda(C^t \otimes I) \\ &= PP^t(C \otimes I)PP^t\Lambda PP^t(C^t \otimes I)PP^t \\ &= P(I \otimes C)\tilde{A}(I \otimes C^t)P^t = P \begin{pmatrix} \tilde{\Lambda}^{(1)} & & & 0 \\ & \tilde{\Lambda}^{(2)} & & \\ & & \ddots & \\ 0 & & & \tilde{\Lambda}^{(n)} \end{pmatrix} P^t, \end{aligned}$$

which is a permutation of a diagonal matrix and hence is still a diagonal matrix. □

4. Estimation of regularization parameters. Besides the issue of boundary conditions, it is well known that blurring matrices are in general ill-conditioned and deblurring algorithms will be extremely sensitive to noise [12, p. 282]. The ill-conditioning of the blurring matrices stems from the wide range of magnitudes of their eigenvalues [10, p. 31]. Therefore, excess amplification of the noise at small eigenvalues can occur. The method of regularization is used to achieve stability for deblurring problems. In the classical Tikhonov regularization [10, p. 117], stability is attained by introducing a regularization operator D which restricts the set of admissible solutions. More specifically, the regularized solution $\mathbf{f}(\mu)$ is computed as the solution to

$$\min_{\mathbf{f}(\mu)} \left\{ \mu \|D\mathbf{f}(\mu)\|_2^2 + \|\mathbf{g} - A\mathbf{f}(\mu)\|_2^2 \right\}. \tag{4.1}$$

The term $\|D\mathbf{f}(\mu)\|_2^2$ is added in order to regularize the solution. The regularization parameter μ controls the degree of regularity (i.e., the degree of bias) of the solution.

One can find the solution $\mathbf{f}(\mu)$ in (4.1) by solving the normal equations

$$(\mu D^t D + A^t A)\mathbf{f}(\mu) = A^t \mathbf{g}. \tag{4.2}$$

Usually, $\|D\mathbf{f}\|_2$ is chosen to be the L_2 norm $\|\mathbf{f}\|_2$ or the H_1 norm $\|L\mathbf{f}\|_2$ where L is the first order difference operator matrix; see [14, 8, 12]. Correspondingly, the matrix D^tD in (4.2) is the identity matrix or the discrete Laplacian matrix with some boundary conditions. In the latter case, if the zero boundary condition is imposed, D^tD is just the discrete Laplacian with the Dirichlet boundary condition. For the periodic boundary condition, D^tD is circulant and can be diagonalized by the FFTs; see, for instance, [12, p. 283]. For the Neumann boundary condition, D^tD is the discrete Laplacian with the Neumann boundary condition, which can be diagonalized by the discrete cosine transform matrix; see, for instance, [4]. Thus if we use the Neumann boundary condition for both the blurring matrix A and the regularization operator D^tD, then the matrix in (4.2) can be diagonalized by the discrete cosine transform matrix and hence its inversion can still be done in three FCTs for any fixed μ; cf. (3.5).

Another difficulty in regularization is the choice of μ. Generalized cross-validation [11] is a technique that estimates μ directly without requiring an estimate of the noise variance. It is based on the concept of prediction errors. For each $k = 1, \dots, n$, let $\mathbf{f}_\mu^k$ be the vector that minimizes the error measure:

$$\sum_{i=1, i\neq k}^{n} ([\mathbf{g}]_i - [A\mathbf{f}]_i)^2 + \mu\|D\mathbf{f}\|_2^2,$$

where $[A\mathbf{f}]_i$ is the ith element of $A\mathbf{f}$ and $[\mathbf{g}]_i$ is the ith element of $\mathbf{g}$. If μ is such that $\mathbf{f}_\mu^k$ is a good estimate of $\mathbf{f}$, then $[A\mathbf{f}_\mu^k]_k$ should be a good approximation of $[\mathbf{g}]_k$ on average. For a given μ, the average squared error between the predicted value $[A\mathbf{f}_\mu^k]_k$ and the actual value $[\mathbf{g}]_k$ is given by

$$\frac{1}{n}\sum_{k=1}^{n}([\mathbf{g}]_k - [A\mathbf{f}_\mu^k]_k)^2.$$

The generalized cross-validation (GCV) is a weighted version of the above error:

$$v(\mu) = \frac{1}{n}\sum_{k=1}^{n}([\mathbf{g}]_k - [A\mathbf{f}_\mu^k]_k)^2 \left[\frac{1 - m_{kk}(\mu)}{1 - \frac{1}{n}\sum_{j=1}^{n} m_{jj}(\mu)}\right]^2,$$

where $m_{jj}(\mu)$ is the (j,j)th entry of the so-called influence matrix

$$M(\mu) = A(A^tA + \mu D^tD)^{-1}A^t.$$

In [11], Golub, Heath, and Wahba have shown that $v(\mu)$ can be written as

$$v(\mu) = n\frac{\|[I - M(\mu)]\mathbf{g}\|_2^2}{\text{tr}[I - M(\mu)]^2}. \tag{4.3}$$

The optimal regularization parameter is chosen to be the μ that minimizes $v(\mu)$. Since $v(\mu)$ is a nonlinear function, the minimizer usually cannot be determined analytically. However, if the Neumann boundary condition is used for both A and D^tD, we can rewrite $v(\mu)$ as

$$v(\mu) = n\frac{\sum_{i=1}^{n}\left[\frac{\mu\beta_i}{\alpha_i^2+\mu\beta_i}\right]^2[C\mathbf{g}]_i}{\left[\sum_{i=1}^{n}\frac{\mu\beta_i}{\alpha_i^2+\mu\beta_i}\right]^2},$$

where α_i and β_i represent the eigenvalues of A and D^tD, respectively. We recall that α_i and β_i can be obtained by taking the FCT of the first column of A and D^tD, respectively, since the matrices can be diagonalized by the discrete cosine transform matrix C. Thus the GCV estimate of μ can be computed in a straightforward manner; see [13].

For the periodic boundary condition, the GCV estimate can also be computed by a similar procedure. However, if we use the zero boundary condition, determining the GCV estimate of μ will require the inversion of a large matrix, which is clearly an overwhelming task for any images of reasonable size.

5. Optimal cosine transform preconditioners. Because all matrices in $\mathcal{C}$ are symmetric (see (3.2)), discrete cosine transform matrices can only be diagonalized blurring matrices from symmetric blurring functions. For nonsymmetric blurring functions, matrices in $\mathcal{C}$ may be used as preconditioners to speed up the convergence of iterative methods such as the conjugate gradient method. Given a matrix A, we define the optimal cosine transform preconditioners $c(A)$ to be the minimizer of $\|Q - A\|_F$ over all Q in $\mathcal{C}$; see [6]. Here $\|\cdot\|_F$ is the Frobenius norm.

In [6, 16], $c(A)$ are obtained by solving linear systems. Here we give a simple approach for finding $c(A)$.

THEOREM 5.1. *Let h be an arbitrary blurring function and A be the blurring matrix of h with the Neumann boundary condition imposed. Then the optimal cosine transform preconditioner $c(A)$ of A can be found as follows:*

1. *In the one-dimensional case, $c(A)$ is the blurring matrix corresponding to the symmetric blurring function $s_i \equiv (h_i + h_{-i})/2$ with the Neumann boundary condition imposed.*
2. *In the two-dimensional case, $c(A)$ is the blurring matrix corresponding to the symmetric blurring function given by $s_{i,j} \equiv (h_{i,j} + h_{i,-j} + h_{-i,j} + h_{-i,-j})/4$ with the Neumann boundary condition imposed.*

Proof. We give the proof only for the one-dimensional case. The proof for the two-dimensional case is similar. We first note that if U and V are symmetric and skew-symmetric matrices, respectively, then $\|U + V\|_F^2 = \|U\|_F^2 + \|V\|_F^2$. Hence, for any $Q = C^t\Lambda C \in \mathcal{C}$, we have

$$\|Q - A\|_F^2 = \|\Lambda - CAC^t\|_F^2 = \|\Lambda - \frac{1}{2}C(A + A^t)C^t\|_F^2 + \|\frac{1}{2}C(A - A^t)C^t\|_F^2.$$

Since the second term in the right-hand side above does not affect the diagonal matrix Λ, the minimizer Λ is given by the diagonal of $\frac{1}{2}C(A + A^t)C^t$.

It is easy to check that

$$\frac{1}{2}(A + A^t) = T(\mathbf{u}) + H(\sigma(\mathbf{u}), J\sigma(\mathbf{u})) + H(\mathbf{v}, -J\mathbf{v}),$$

where

$$\mathbf{u} = \left(h_0, \frac{h_1 + h_{-1}}{2}, \frac{h_2 + h_{-2}}{2}, \ldots, \frac{h_{m-1} + h_{-m+1}}{2}, 0, \ldots, 0\right)^t$$

and

$$\mathbf{v} = \left(\frac{h_1 - h_{-1}}{2}, \ldots, \frac{h_{m-1} - h_{-m+1}}{2}, 0, \ldots, 0\right)^t.$$

We claim that the diagonal entries $[CH(\mathbf{v}, -J\mathbf{v})]C^t]_{i,i} = 0$ for all i. If this is true, then the minimizer Λ is given by

$$\Lambda = C[T(\mathbf{u}) + H(\sigma(\mathbf{u}), J\sigma(\mathbf{u}))]C^t,$$

and Theorem 5.1 then follows directly from Theorem 3.2.

To prove our claim, we first note that by (3.1),

$$\begin{aligned}[CJ]_{i,j} = [C]_{i,(n+1-j)} &= \sqrt{\frac{2-\delta_{i1}}{n}} \cos\left(\frac{(i-1)(2(n+1-j)-1)\pi}{2n}\right) \\ &= (-1)^{(i-1)}[C]_{i,j}, \qquad 1 \le i, j \le n.\end{aligned}$$

Also it is clear that $T \equiv JH(\mathbf{v}, -J\mathbf{v})$ is a skew-symmetric Toeplitz matrix. Therefore,

$$\begin{aligned}[CH(\mathbf{v}, -J\mathbf{v})]C^t]_{i,i} = [CJJH(\mathbf{v}, -J\mathbf{v})]C^t]_{i,i} &= [CJTC^t]_{i,i} \\ &= (-1)^i [CTC^t]_{i,i} = 0. \qquad \square\end{aligned}$$

In view of Theorem 5.1 and the results in section 3, it is easy to find $c(A)$ for blurring matrices generated by nonsymmetric blurring functions. We just take the symmetric part of the blurring functions and form the (block) Toeplitz-plus-Hankel matrices as given in (2.10) or (3.7). From Theorem 5.1, we also see that if the blurring function is close to symmetric, then $c(A)$ will be a good approximation (and hence a good preconditioner) to A; see the numerical results in section 6.

6. Numerical experiments.

In this section, we illustrate the efficiency of employing the Neumann boundary condition over the other two boundary conditions for image restoration problems. All our tests were done by Matlab. The data source is a photo from the 1964 Gatlinburg Conference on Numerical Algebra taken from Matlab. From (2.4), we see that to construct the right-hand side vector $\mathbf{g}$ correctly, we need the vectors $\mathbf{f}_l$ and $\mathbf{f}_r$; i.e., we need to know the image outside the given domain. Thus we start with the 480-by-640 image of the photo and cut out a 256-by-256 portion from the image. Figure 6.1 gives the 256-by-256 image of this picture.

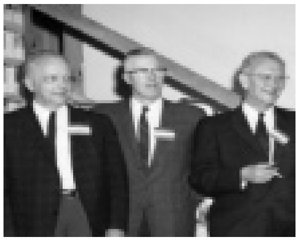
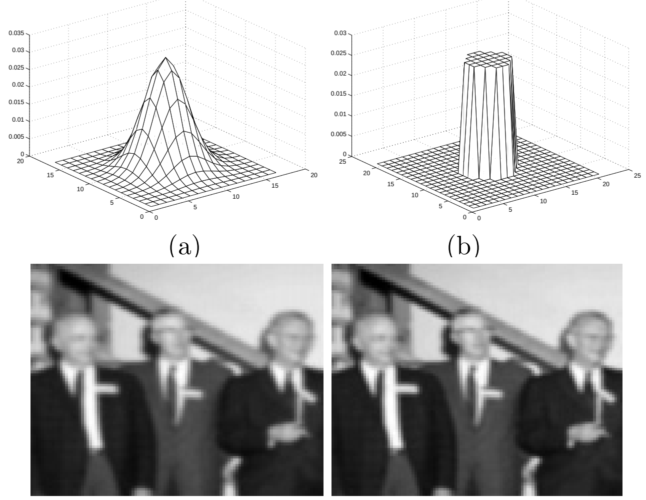


FIG. 6.1. *The Gatlinburg Conference image.*

We consider restoring the Gatlinburg Conference image blurred by the following two blurring functions (see [14, p. 269]):

(i) a truncated Gaussian blur,

$$h_{i,j} = \begin{cases} ce^{-0.1(i^2+j^2)} & \text{if } |i-j| \le 8, \\ 0 & \text{otherwise;} \end{cases}$$



(a) (b)

(c) (d)

FIG. 6.2. (a) *Gaussian blur;* (b) *out-of-focus blur; noisy and blurred image* (c) *by Gaussian blur and* (d) *by out-of-focus blur.*

(ii) an out-of-focus blur,

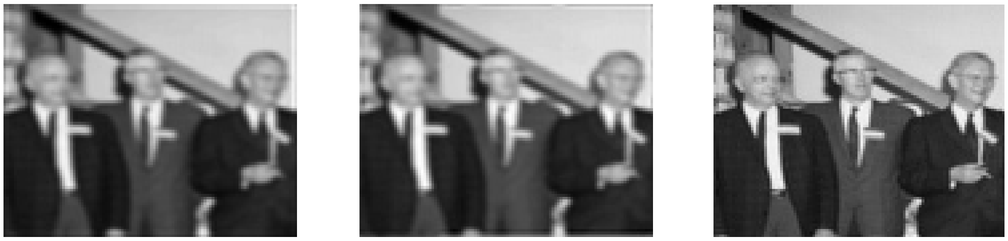
$$h_{i,j} = \begin{cases} c & \text{if } i^2 + j^2 \leq 4, \\ 0 & \text{otherwise,} \end{cases}$$

where $h_{i,j}$ is the jth entry of the first column of $T^{(i)}$ in (3.6) and c is the normalization constant such that $\sum_{i,j} h_{i,j} = 1$. We remark that the Gaussian blur is symmetric and separable, whereas the out-of-focus blur is symmetric but not separable; see Figures 6.2(a) and 6.2(b). A Gaussian white noise $\mathbf{n}$ with signal-to-noise ratio of 50dB is then added to the blurred images. The noisy blurred images are shown in Figures 6.2(c) and 6.2(d). We note that after the blurring, the cigarette held by Prof. Householder (the rightmost person) is not clearly shown; cf. Figure 6.1.

We remark that the regularization parameter based on the GCV method (see section 4) is not suitable when the zero boundary condition is imposed. The method will require the inversion of many large matrices. As a comparison of the cost among different boundary conditions, we chose the optimal regularization parameter μ^* such that it minimizes the relative error of the reconstructed image $\mathbf{f}(\mu)$ which is defined as $\|\mathbf{f} - \mathbf{f}(\mu)\|_2 / \|\mathbf{f}\|_2$, where $\mathbf{f}$ is the original image. The optimal μ^*, accurate up to one significant digit, is obtained by trial and error.

In Figures 6.3 and 6.4, we present the restored images for the three different boundary conditions, the optimal μ^*, and the relative errors. We used the L_2 norm as the regularization functional here. We see from the figures that by imposing the Neumann boundary condition, the relative error and the ringing effect are the smallest. Also the cigarette is better reconstructed by using the Neumann boundary condition than by using the other two boundary conditions.

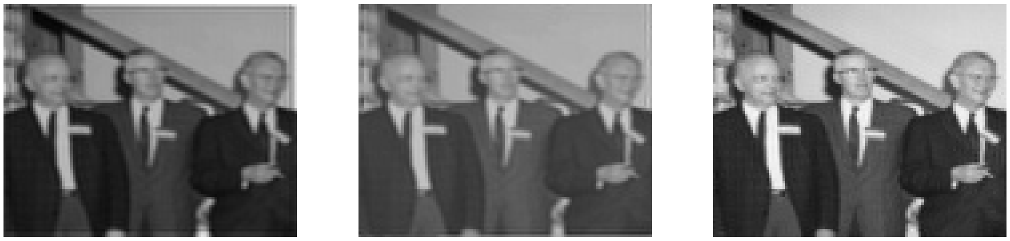
Next let us consider the cost. Recall that for each μ, we need only three two-dimensional FFTs and FCTs to compute the restored images for the periodic and the Neumann boundary conditions, respectively. Thus the costs for both approaches are about $O(n^2 \log n)$ operations, although the Neumann boundary condition is twice



$\mu^* = 0.3$ $\mu^* = 0.3$ $\mu^* = 0.0003$

rel. error $= 1.24 \times 10^{-1}$ rel. error $= 1.15 \times 10^{-1}$ rel. error $= 6.59 \times 10^{-2}$

FIG. 6.3. *Restoring Gaussian blur with zero (left), periodic (middle), and Neumann (right) boundary conditions.*



$\mu^* = 0.05$ $\mu^* = 0.05$ $\mu^* = 0.005$

rel. error $= 1.20 \times 10^{-1}$ rel. error $= 1.09 \times 10^{-1}$ rel. error $= 4.00 \times 10^{-2}$

FIG. 6.4. *Restoring out-of-focus blur with zero (left), periodic (middle), and Neumann (right) boundary conditions.*

as fast, because FCTs require only real multiplications [23, pp. 59–60]. For the zero boundary condition, we have to solve a block-Toeplitz–Toeplitz-block system for each μ. The fastest direct Toeplitz solver requires $O(n^4)$ operations; see [17]. In our tests, the systems are solved by the preconditioned conjugate gradient method with circulant preconditioners [7]. Table 6.1 shows the numbers of iterations required for the two blurring functions for different μ. The stopping tolerance is 10^{-6}. We note that the cost per iteration is about four two-dimensional FFTs. Thus the cost is extremely expensive, especially when μ is small. In conclusion, we see that the cost of using the Neumann boundary condition is lower than that of using the other two boundary conditions.

Finally, we illustrate the effectiveness of the optimal cosine transform preconditioners for blurring functions that are close to symmetric. More general tests on the preconditioners are given in [6]. We consider a two-dimensional deconvolution problem arising in the ground-based atmospheric imaging. Figure 6.5(a) gives the 256-by-256 blurred and noisy image of an ocean reconnaissance satellite observed by a ground-based imaging system, and Figure 6.5(b) is a 256-by-256 image of a guide star observed under similar circumstances; see [8]. The discrete blurring function $\mathbf{h}$ is given by the pixel values of the guide star image. The blurring matrix A is obtained as in (3.7) by imposing the Neumann boundary condition.

We note that the blurring function is not exactly symmetric in this case; see Figure 6.5(b). However, from the cross sections of the blurring function (see, for instance, Figure 6.5(c)), we know that it is close to symmetric. Therefore, we used

Table 6.1
The numbers of iterations required for using the zero boundary condition.

	μ					
Blurring function	3×10^{-5}	5×10^{-5}	3×10^{-4}	5×10^{-4}	3×10^{-3}	5×10^{-3}
Gaussian	67	65	41	40	25	25
Out-of-focus	81	73	48	43	30	26
Blurring function	3×10^{-2}	5×10^{-2}	3×10^{-1}	5×10^{-1}	3×10^{0}	5×10^{0}
Gaussian	17	17	13	12	7	7
Out-of-focus	22	21	15	14	9	9

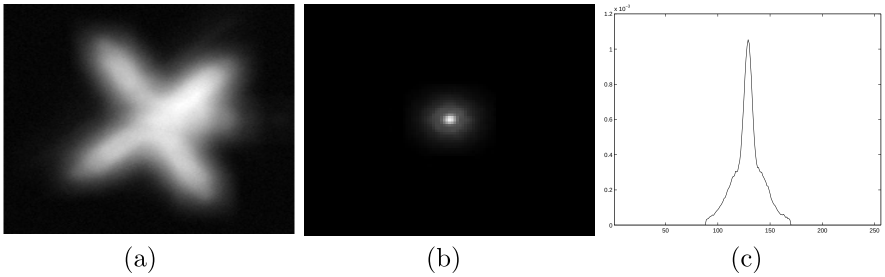


Fig. 6.5. (a) *Observed image,* (b) *the guide star image, and* (c) *a cross section of the blurring function.*

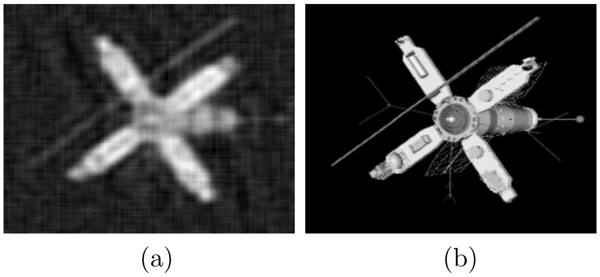


Fig. 6.6. (a) *Restored image* ($\mu^* = 1 \times 10^{-4}$) *and* (b) *the true image.*

Table 6.2
The number of iterations for convergence.

	μ					
Method	1×10^{-5}	5×10^{-5}	1×10^{-4}	5×10^{-4}	1×10^{-3}	5×10^{-3}
CG	345	231	134	113	98	87
PCG	5	4	4	3	3	2

the preconditioned conjugate gradient algorithm with the optimal cosine transform preconditioner to remove the blurring; see section 5. Again we use the L_2 norm as the regularization functional here. In Figure 6.6(a), we present the restored image with the optimal μ^*. The original image is given in Figure 6.6(b) for comparison.

Using the optimal cosine transform preconditioner, the image is restored in 4 iterations for a stopping tolerance of 10^{-6}. If no preconditioner is used, acceptable restoration is achieved after 134 iterations; see Table 6.2. We remark that the cost per iteration for using the optimal cosine transform preconditioner is almost the same as that with no preconditioner: they are about 1.178×10^8 and 1.150×10^8 floating point operations per iteration, respectively. Thus we see that the preconditioned conjugate gradient algorithm with the optimal cosine transform preconditioner is an efficient and effective method for this problem.

7. Concluding remarks. In this paper, we have shown that discrete cosine transform matrices can diagonalize dense (block) Toeplitz-plus-Hankel blurring matrices arising from using the Neumann (reflective) boundary condition. Numerical results suggest that the Neumann boundary condition provides an effective model for image restoration problems, in terms of both the computational cost and minimizing the ringing effects near the boundary. It is interesting to note that discrete sine transform matrices can diagonalize Toeplitz matrices with at most three bands (such as the discrete Laplacian with zero boundary conditions) but not dense Toeplitz matrices in general; see [9], for instance.

REFERENCES

[1] G. Ammar and W. Gragg, *Superfast solution of real positive definite Toeplitz systems*, SIAM J. Matrix Anal. Appl., 9 (1988), pp. 61–76.

[2] H. Andrew and B. Hunt, *Digital Image Restoration*, Prentice–Hall, Englewood Cliffs, NJ, 1977.

[3] M. Banham and A. Katsaggelos, *Digital image restoration*, IEEE Signal Processing Magazine, March 1997, pp. 24–41.

[4] M. Buckley, *Fast computation of a discretized thin-plate smoothing spline for image data*, Biometrika, 81 (1994), pp. 247–258.

[5] K. Castleman, *Digital Image Processing*, Prentice–Hall, Englewood Cliffs, NJ, 1996.

[6] R. Chan, T. Chan, and C. Wong, *Cosine transform based preconditioners for total variation deblurring*, IEEE Trans. Image Process., to appear.

[7] R. Chan and M. Ng, *Conjugate gradient methods for Toeplitz systems*, SIAM Rev., 38 (1996), pp. 427–482.

[8] R. Chan, M. Ng, and R. Plemmons, *Generalization of Strang's preconditioner with applications to Toeplitz least squares problems*, J. Numer. Linear Algebra Appl., 3 (1996), pp. 45–64.

[9] R. Chan, M. Ng, and C. Wong, *Sine transform based preconditioners for symmetric Toeplitz systems*, Linear Algebra Appl., 232 (1996), pp. 237–260.

[10] H. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.

[11] G. Golub, M. Heath, and G. Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics, 21 (1979), pp. 215–223.

[12] R. Gonzalez and R. Woods, *Digital Image Processing*, Addison–Wesley, Reading, MA, 1992.

[13] P. Hansen, *Regularization tools: A Matlab package for analysis and solution of discrete ill-posed problems*, Numer. Algorithms, 6 (1994), pp. 1–35.

[14] A. Jain, *Fundamentals of Digital Image Processing*, Prentice–Hall, Englewood Cliffs, NJ, 1989.

[15] T. Kailath and V. Olshevsky, *Displacement structure approach to discrete-trigonometric-transform based preconditioners of G. Strang type and T. Chan type*, Calcolo, 33 (1996), pp. 191–208.

[16] T. Kailath and A. Sayed, *Displacement structure: Theory and applications*, SIAM Rev., 37 (1995), pp. 297–386.

[17] N. Kalouptsidis, G. Carayannis, and D. Manolakis, *Fast algorithms for block Toeplitz matrices with Toeplitz entries*, Signal Process., 6 (1984), pp. 77–81.

[18] R. Lagendijk and J. Biemond, *Iterative Identification and Restoration of Images*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1991.

[19] N. Levinson, *The Wiener RMS (root mean square) error criterion in filter design and prediction*, J. Math. Phys., 25 (1946), pp. 261–278.

[20] J. Lim, *Two-dimensional Signal and Image Processing*, Prentice–Hall, Englewood Cliffs, NJ, 1990.

[21] F. Luk and D. Vandevoorde, *Reducing boundary distortion in image restoration*, in Proceedings SPIE 2296, Advanced Signal Processing Algorithms, Architectures and Implementa-

tions VI, San Diego, CA, 1994, pp. 554–565.

[22] S. Martucci, *Symmetric convolution and the discrete sine and cosine transforms*, IEEE Trans. Signal Process., 42 (1994), pp. 1038–1051.

[23] K. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*, Academic Press, New York, 1990.

[24] V. Sanchez, P. Garcia, A. Peinado, J. Segura, and A. Rudio, *Diagonalization properties of the discrete cosine transforms*, IEEE Trans. Signal Process., 43 (1995), pp. 2631–2641.

[25] G. Strang, *The discrete cosine transform*, SIAM Rev., 41 (1999), pp. 135–147.