by S. Mitra and J. Kaiser, New York, Wiley.

[26] C. Vogel, *A Multigrid Method for Total Variation-Based Image Denoising,* in Computation and Control IV, conference proceedings to be published by Birkhauser.

[27] C. Vogel and M. Oman, *Iterative Methods for Total Variation Denoising,* SIAM J. Sci. Statist. Comput., to appear.

[28] C. Vogel and M. Oman, *Fast Total Variation-Based Image Reconstruction,* to appear in ASME Proceeding on Inverse Problem, Boston, August 1995.

[29] C. Wong, *Block Toeplitz Type Preconditioners for Elliptic Problem,* M.Phil. thesis, The University of Hong Kong, 1994.

[30] P. Yip and K. Rao, *Fast Decimation-in-time Algorithms for a Family of Discrete Sine and Cosine Transforms*, Circuits, Syst., Signal Process, V3 (1984), pp. 387–408.

REFERENCES

[1] R. Acar and C. Vogel, *Analysis of Bounded Variation Penalty Methods*, Inverse Problems, Vol. 10 (1994), pp. 1217–1229.

[2] L. Alvarez and J. Morel *Formalization and Computational Aspects of Image Analysis*, Acta Numerica (1994), pp.1–59.

[3] E. Boman and I. Koltracht, *Fast Transform Based Preconditioners for Toeplitz Equations*, University of Connecticut, Preprint, August 1993.

[4] R. Chan, *The Spectrum of a Family of Circulant Preconditioned Toeplitz Systems*, SIAM J. Num. Anal., V26 (1989), pp. 503–506.

[5] R. Chan and T. Chan, *Circulant Preconditioners for Elliptic Problems* J. Numer. Lin. Alg. Appl., 1 (1992), pp. 77–101.

[6] R. Chan, W. Ching and C. Wong, *Optimal Trigonometric Preconditioners for Elliptic and Queueing Problems*, to appear in Proceedings for the Conference on Scientific Computating '94, SEA Bull., 1996.

[7] R. Chan and X. Jin, *A Family of Block Preconditioners for Block Systems*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 1218–1235.

[8] R. Chan, X. Jin and M. Yeung, *The Circulant Operator in the Banach Algebra of Matrices*, Lin. Alg. Appl. V149 (1991), pp. 41–53.

[9] R. Chan, J. Nagy and R. Plemmons, *FFT-Based Preconditioners for Toeplitz-Block Least Squares Problems*, SIAM J. Numer. Anal., 30:1740–1768, 1993.

[10] R. Chan and K. Ng, *Conjugate Gradient Methods for Toeplitz Systems*, Technical Report 94-8, Department of Mathematics, The Chinese University of Hong Kong, 1994, to appear in SIAM Review.

[11] R. Chan, K. Ng, and C. Wong, *Sine Transform Based Preconditioners for Symmetric Toeplitz Systems*, Technical Report 94-8, The Chinese University of Hong Kong, 1993, to appear in Lin. Alg. Appl..

[12] R. Chan and C. Wong, *Sine Transform Based Preconditioners for Elliptic Problems*, Technical Report 93-12-4, The Chinese University of Hong Kong, 1993.

[13] T. Chan, *An Optimal Circulant Preconditioner for Toeplitz Systems*, SIAM J. Sci. Statist. Comput., V9 (1988), pp. 766–771.

[14] T. Chan and J. Olkin, *Circulant Preconditioners for Toeplitz-block Matrices*, Numer. Algorithms, 6(1994) pp. 89–101.

[15] P. Concus, G. Golub, and G. Meurant, *Block Preconditioning for the Conjugate Gradient Method,* SIAM J. Sci. Statist. Comput., 6 (1985) pp. 220–252.

[16] T. Dupont, R. Kendall and H. Rachford, *An Approximate Factorization Procedure for Solving Self-Adjoint Elliptic Difference Equations,* SIAM J. Numer. Anal., 5 (1968), pp. 559–573.

[17] G. Golub and C. Van Loan, *Matrix Computations* , 2nd Ed., The Johns Hopkins University Press, Maryland, 1989.

[18] T. Huckle, *Circulant/skewcirculant Matrices for Solving Toeplitz Matrix Problems*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 767 – 777.

[19] I. Ito and K. Kunisch, *An Active Set Strategy for Image Restoration Based on the Augmented Lagrangian Formulatin*, Preprint, Center for Research in Scientific Computation, North Carolina State University.

[20] Y. Li and F. Santosa, *An Affine Scaling Algorithm for Minimizing Total Variation in Image Enhancement*, Technical Report CTC94TR201, Cornell Theory Center, Cornell University.

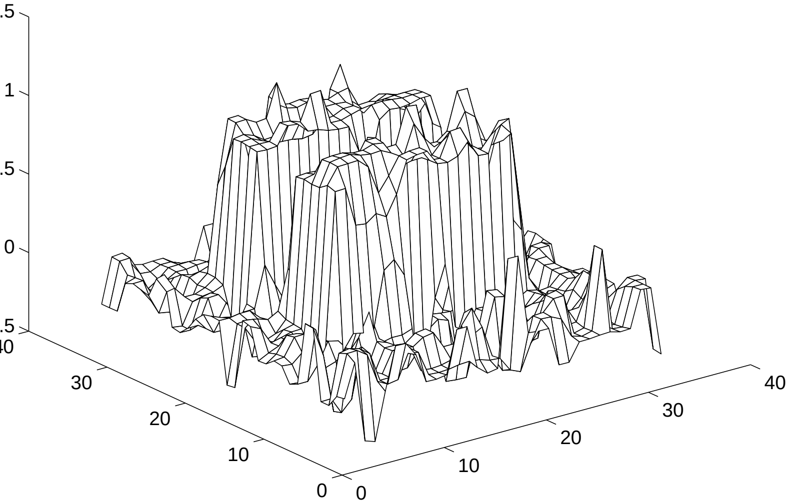
[21] T. Manteuffel and S. Parter, *Preconditioning and Boundary Conditions*, SIAM J. Numer. Anal., 27 (1990), pp. 656–694.

[22] M. Oman *Fast Multigrid Techniques in Total Variation-Based Image Reconstruction*, to appear in the Preliminary Proceedings of the 1995 Copper Mountain Conference on Multigrid Methods.

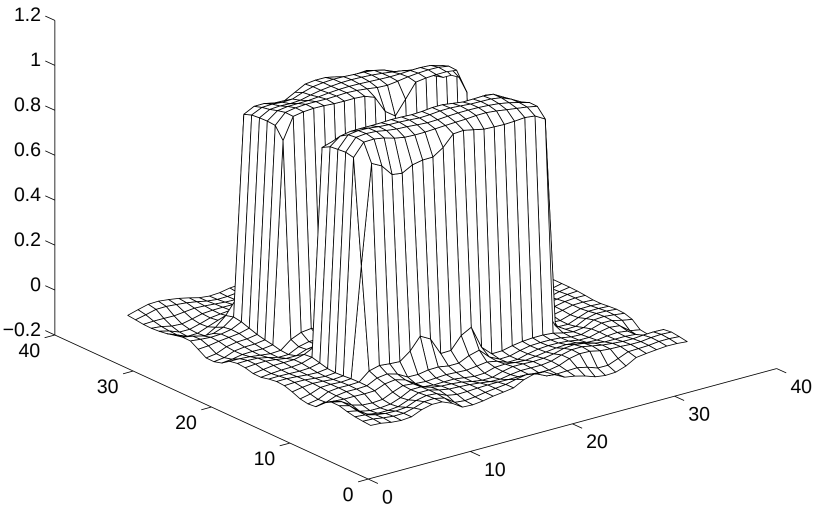
[23] L. Rudin, S. Osher, *Total Variation Based Image Restoration with Free Local Constraints*, IEEE International Conference on Image Processing, Austin, TX, 1994.

[24] L. Rudin, S. Osher and E. Fatemi, *Nonlinear Total Variation Based Noise Removal Algorithms*, Physica D., 60 (1992), pp. 259–268.

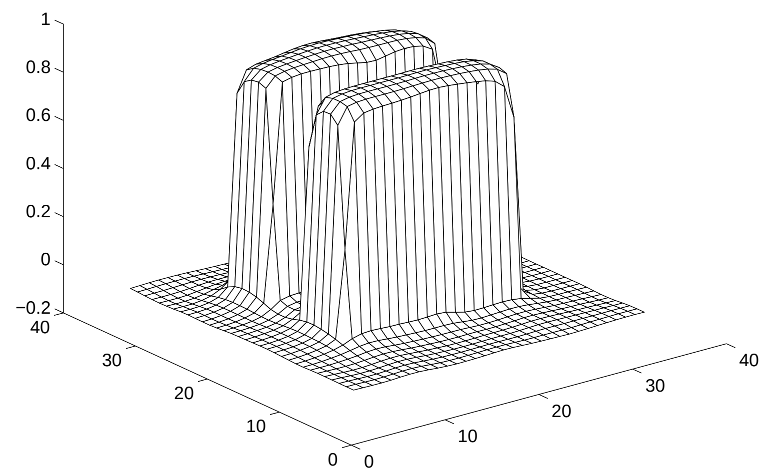
[25] H. Sorensen and C. Burrus, *Fast DFT and Convolution Algorithms*, Hankbook of Signal Processing edited



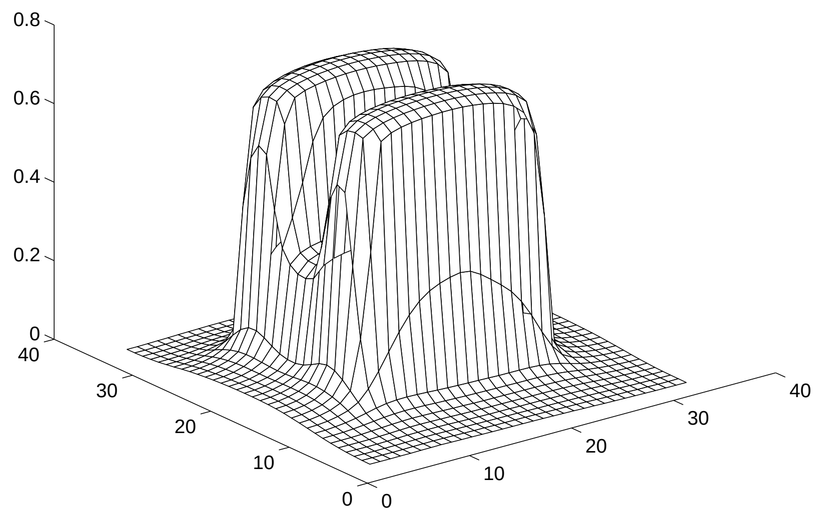
$\alpha = 0.00001$



$\alpha = 0.0001$



$\alpha = 0.001$



$\alpha = 0.01$

Fig 6. Deblurred images for $\beta = 0.1$, $n = 32$ and various α

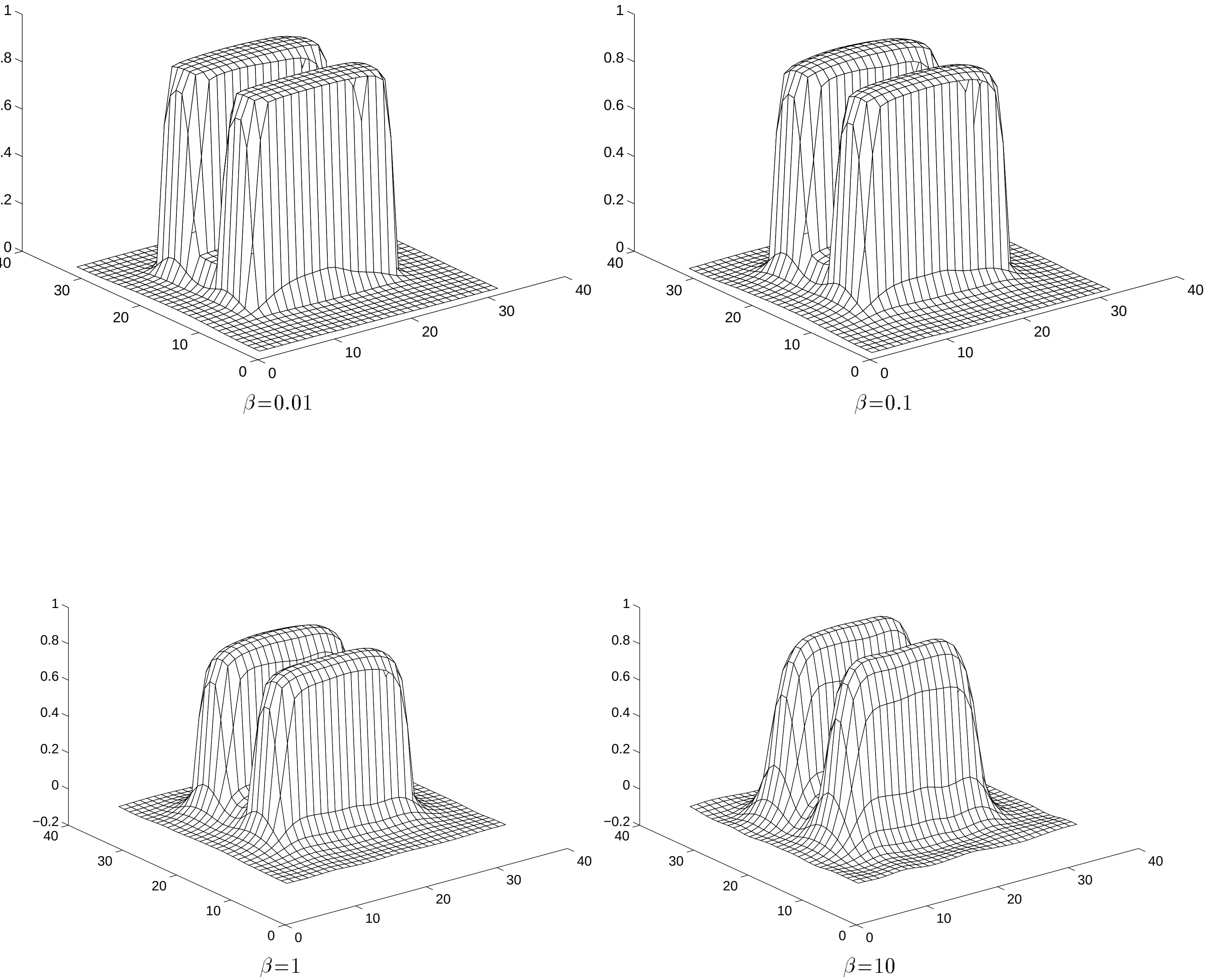


Fig 5. Deblurred images for $\alpha = 0.004$, $n = 32$ and various β

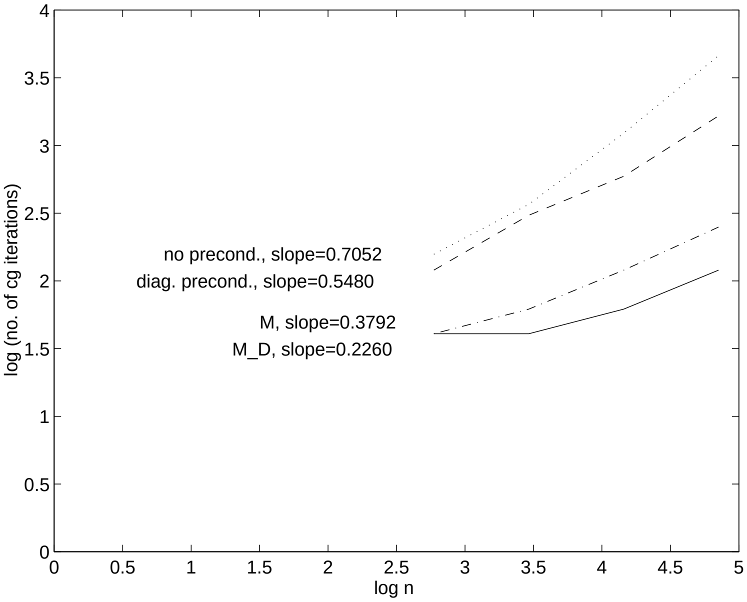


Fig. 2b $\alpha = 0.004, \beta = 10$

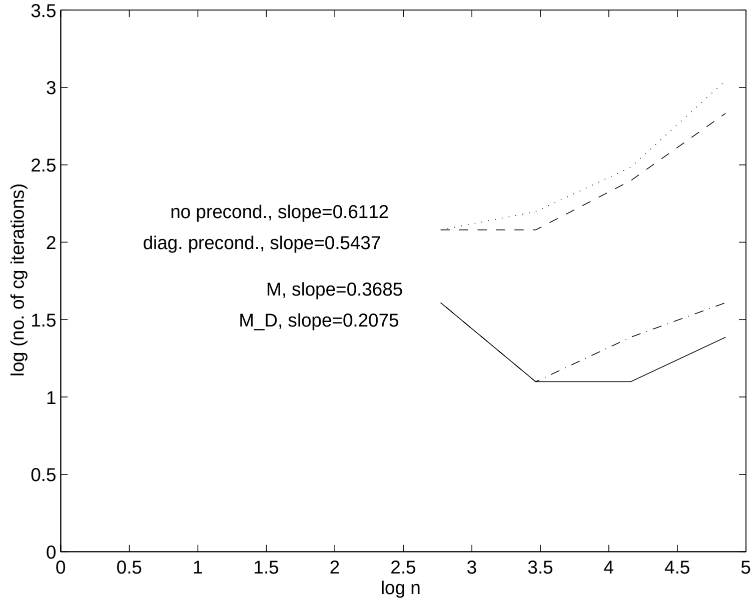


Fig. 2c $\alpha = 0.004, \beta = 100$

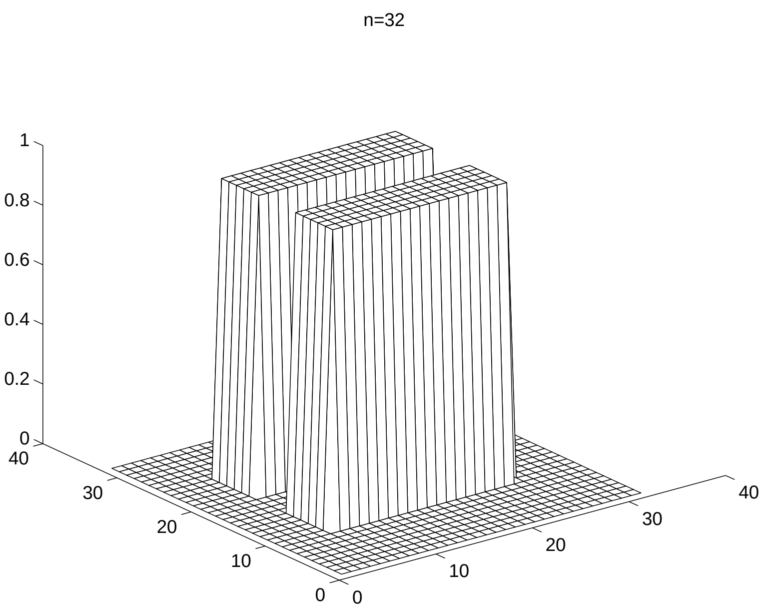


Fig. 3 True image

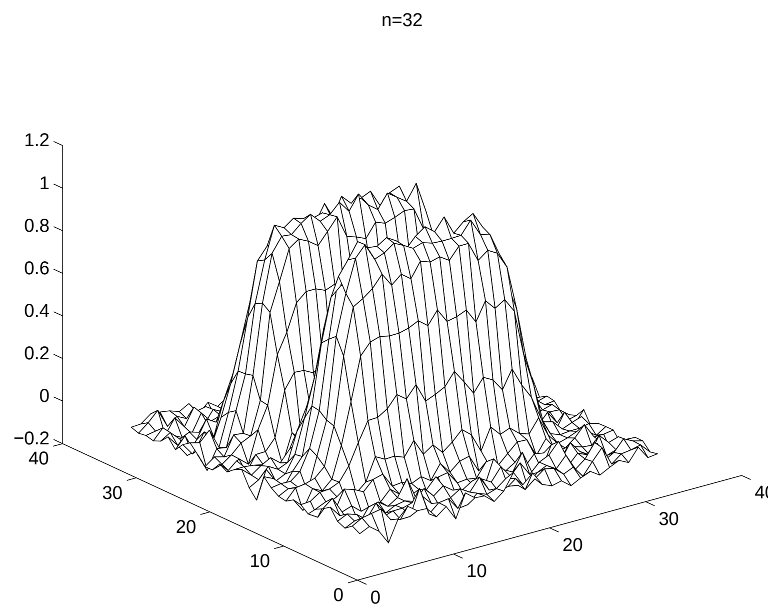


Fig. 4 blurred and noisy image with $\sigma = 0.3333$

	n=64					n=128				
β	#	I	Δ	M	M_D	#	I	Δ	M	M_D
0.01	150	122	59	61	24	255	> 200	71	82	32
0.1	129	70	37	33	16	209	117	52	43	22
1	84	40	24	16	11	139	69	35	22	14
10	52	22	16	8	6	85	39	25	11	8
100	15	12	11	4	3	33	21	17	5	4

Table 2. $\alpha = 0.004$

	n=16					n=32				
α	#	I	Δ	M	M_D	#	I	Δ	M	M_D
0.1	8	51	47	6	8	10	97	86	7	12
0.01	54	27	22	13	10	89	51	33	23	15
0.001	43	20	15	16	11	62	32	19	25	12
0.0001	31	25	24	16	15	45	25	17	20	14
0.00001	58	78	77	23	23	50	53	53	21	20

Table 3. $\beta = 0.1$

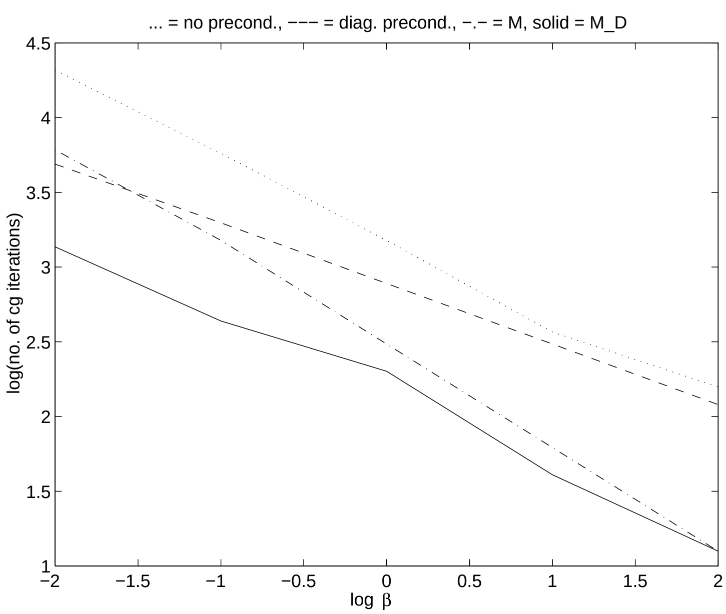


Fig. 1 $n = 32$, $\alpha = 0.004$

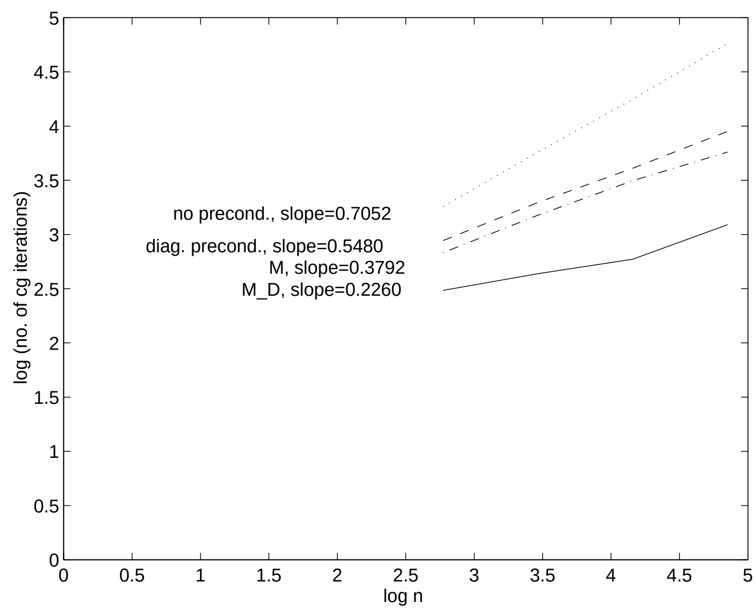


Fig. 2a $\alpha = 0.004$, $\beta = 0.1$

Tables 2 and 3 show the number of iterations required for convergence of the FP iteration and the CG iteration for different choices of preconditioners and parameters. Note that the CG iteration numbers shown in Tables 2 and 3 are the average number of CG iterations per FP step. The symbol "#" denotes the number of iterations for FP. The notations I, Δ, M, M_D denote respectively no preconditioner, diagonal scaling preconditioner, cosine transform preconditioner and cosine transform preconditioner with diagonal scaling. Some of the data are plotted in Fig. 1 and 2.

We observe from Fig. 1 that the M_D preconditioner requires significantly fewer iterations than other preconditioners for almost all values of α, β and n. Moreover, we can observe that the smaller the β is, the more ill-conditioned the system is. In fact, it can be easily shown that the coefficient $\kappa_\beta(u)$ in (6) of the elliptic operator $\mathcal{L}_\beta(u^k)$ is bounded above and below by $1/\sqrt{\beta}$ and $O(1/n)$ respectively. Therefore by Theorem 2, for fixed n, $\kappa(c_2(L_\beta)^{-1}L_\beta) \leq O(1/\beta)$. We note that the slope of the dotted line (i.e. no preconditioning) in Fig. 1 is -0.5301 which agrees with the bound $O(1/\sqrt{\beta})$ for the number of PCG iterations.

From Fig. 2a-c, we observe that the number of iterations corresponding to I grows like $O(n^{0.7})$. Therefore, $\kappa(A_\beta(u^k)) \approx O(n^{1.5})$. If the preconditioner M_D is used, the number of iterations grows like $O(n^{\frac{1}{4}})$, i.e., $\kappa(M_D^{-1}A_\beta(u^k)) \approx O(n^{\frac{1}{2}})$. However, preconditioners Δ and M reduce the growth of the number of iterations only to $O(n^{\frac{1}{2}})$ and $O(n^{0.35})$ respectively. We remark that for $\beta = 100$, the number of iterations corresponding to M_D seems to be independent of n, see Fig. 2c. This agrees with Theorem 2 and the remark after Theorem 3.

From Table 3, we observe that our preconditioner M_D is quite insensitive to α, at least in the range $[0.1, 0.0001]$, unlike the unpreconditioned system. When α becomes very small ($\alpha \leq 0.00001$), the iteration number does increase. However, this only happens when the regularization is too small.

In Fig 5, we show the recovered images for various β. The smaller β is, the closer the recovered image is to the true image. Fig. 6 shows how the recovered images depend on the value of α.

	n=16					n=32				
β	#	I	Δ	M	M_D	#	I	Δ	M	M_D
0.01	65	42	25	30	16	117	75	40	44	23
0.1	46	26	19	17	12	81	43	27	24	14
1	30	15	13	9	8	50	24	18	12	10
10	15	9	8	5	5	31	13	12	6	5
100	6	8	8	5	5	9	9	8	3	3

where $\tilde{A}(u^k) = \tilde{K}^*\tilde{K} + \alpha\tilde{L}_\beta(u^k)$, $\tilde{K} = K\Delta^{-1/2}$, $\tilde{L}_\beta(u^k) = \Delta^{-1/2}L_\beta(u^k)\Delta^{-1/2}$ and $\tilde{u}^k = \Delta^{1/2}u^k$. In summary, the Level-2 cosine preconditioner with diagonal scaling is given by

$$M_D = \hat{K}^*\hat{K} + \alpha c_2(\tilde{L}_\beta(u^k))$$

where $\hat{K} = c_2(K)c_2(\Delta^{-1/2})$. We note further that if Λ_1, Λ_2 and Λ_3 are respectively the eigenvalue matrices of $c_2(K)$, $c_2(\Delta^{-1/2})$ and $c_2(\tilde{L}_\beta(u^k))$, then the preconditioner can be expressed as

$$M_D = (C_n \otimes C_n)^t \, (\Lambda_1^*\Lambda_1\Lambda_2^*\Lambda_2 + \alpha\Lambda_3) \, (C_n \otimes C_n).$$

Hence, the preconditioner can be easily invertible.

Finally, we comment on the cost of constructing M_D and of each PCG iteration. We note that L_β is a sparse matrix with only five nonzero bands. Also for K corresponding to a discretization of the convolution operator (2), K often will be a block Toeplitz matrix with Toeplitz blocks. By using Table 1, the construction cost of $c_2(K)$ and $c_2(L_\beta)$ can be shown to be $O(n^2 \log n)$ operations; see [7, 14]. The cost of one PCG iteration is bounded by the cost of the matrix vector multiplication $\tilde{A}(u^k)v = \Delta^{-1/2}(K^*K + \alpha L_\beta(u^k))\Delta^{-1/2}v$ and the cost of solving the system $M_D y = b$. The matrix vector multiplication $\Delta^{-1/2}v$ can be computed in $O(n^2)$ operations because $\Delta^{-1/2}$ is a diagonal matrix. Since $L_\beta(u^k)$ is banded, $L_\beta(u^k)v$ can also be done in $O(n^2)$ operations. For K being a block Toeplitz matrix with Toeplitz blocks, Kv can be calculated in $O(n^2 \log n)$ operations; see [7]. Therefore, the matrix vector multiplication can be done in $O(n^2 \log n)$ operations. The system $M_D y = b$ can be solved in $O(n^2 \log n)$ operations by exploiting the Fast Cosine transform. Therefore, the total cost of each PCG iteration is bounded by of $O(n^2 \log n)$.

4. Numerical Results. In this section, we present results of a numerical comparison of no preconditioning and with our preconditioner in solving the linear system in (8). Let us choose the following test image

$$u(x) = \chi_{[1/3,1/2]\times[1/4,5/6]} + \chi_{[2/3,5/6]\times[1/4,5/6]}$$

where $\chi_{[a,b]}$ denotes the indicator function for the interval $a \le x \le b$. Then we consider the spatially invariant discretized point spread function matrix K with first column given by

$$[K]_{(i-1)n+j,1} = \begin{cases} e^{-0.3(i-j)^2} & |i-j| < 9 \\ 0 & \text{otherwise} \end{cases}$$

which is consider in [9]. Next, we introduce the noise η with a Gaussian distribution with mean 0 and standard variation $\sigma = 0.3333$.

We will perform the FP iterations until the gradient g in (5) satisfies $||g(u^{k+1})||_2/||g(u^0)||_2 \le 10^{-3}$. We will apply the CG method to solve the linear system (8). The method will be stopped when the residual vector r_k of the linear system (8) at the k-th CG iteration satisfies $||r_k||_2/||r_0||_2 < 10^{-3}$. In our numerical experiment, we will concentrate on the performance of our preconditioner for various of parameters n, α and β.

Hence, $c_2(A_{nn})$ can be inverted easily.

For elliptic problem, it can be proved that the Level-2 optimal cosine transform preconditioner $c_2(A_{nn})$ is a "good" preconditioner.

THEOREM 2. *Let A_{nn} be the 5-point centered discretization matrix of*

$$-(a(x,y)u_x)_x - (b(x,y)u_y)_y + \gamma u = f(x,y) \quad on \quad [0,1]^2$$

with homogeneous Neumann boundary condition. Assume $\gamma > 0$ and the mesh is uniform with size $1/n$. Then we have

$$\kappa(c_2(A_{nn})^{-1}A_{nn}) \le (\frac{c_{\max}}{c_{\min}})^2$$

where $0 < c_{\min} \le a(x,y), b(x,y) \le c_{\max}$.

Proof. The proof is similar to that of Theorem 2 in [12]. □

Optimal cosine transform preconditioner can also be shown to be good for solving Toeplitz system; see [4, 7]. For the optimal cosine transform preconditioner, we can prove the following result.

THEOREM 3. *Let A_{nn} be a block Toeplitz matrix with Toeplitz blocks. If the generating sequence $a_k^{(j)} = (A_{nn})_{(j-1)n+k,1}$ of A_{nn} is absolutely summable, i.e.,*

$$\sum_{j=0}^{\infty}\sum_{k=0}^{\infty} |a_k^{(j)}| \le G < \infty.$$

Then the number of iterations for the PCG method converges with preconditioner $c_2(A_{nn})$ is bound by $O(n)$.

Proof. The proof is similar to that of Corollary 1 in [7]. □

We remark that the numerical results in [7] for the Level-2 optimal circulant preconditioner is better and the number of iterations seems to be independent of n.

3. Cosine Transform Preconditioner for TV denoising and deblurring. A straightforward preconditioner for $A_\beta(u^k)$ is $c_2(A_\beta(u^k)) = c_2(K^*K + \alpha L_\beta(u^k)) = c_2(K^*K) + \alpha c_2(L_\beta(u^k))$. However, computing $c_2(K^*K)$ according to the formula in Theorem 1 requires computing all the entries of K^*K and is costly. Another way is to approximate $c_2(K^*K)$ by $c_2(K)^*c_2(K)$. More precisely, a preconditioner for $A_\beta(u^k)$ in (8) can be defined as

$$M = c_2(K)^*c_2(K) + \alpha c_2(L_\beta(u^k)). \tag{14}$$

One problem with the preconditioner M in (14) is that it does not capture the possibly large variation in the coefficient of the elliptic operator in (8) caused by the vanishing of $|\nabla u|$ in (5). To cure this problem, we apply the technique of diagonal scaling. More precisely, if we denote $\rho(\cdot)$ to be the spectral radius of a matrix and we define $\Delta \equiv \rho(K^*K)I + \alpha \mathrm{diag}(L_\beta(u^k))$ then we consider solving the equivalent system

$$\tilde{A}(u^k)\tilde{u}^{k+1} = \tilde{K}^*z$$

summing $a_{i,j}$ for constant value of $|j-i|$ can be reduced to a scalar multiplication. Similarly, for banded matrix A_n with lower and upper band width b_l and b_u, the cost of forming $c(A_n)$ can be reduced to $O((b_l+b_u)n)$. We summarize the construction cost of $c(A_n)$ in Table 1.

A_n	cost of constructing $c(A_n)$
general	$O(n^2)$
Toeplitz	$O(n)$
banded	$O((b_l+b_u)n)$

Table 1 Cost of Constructing $c(A_n)$

2.2. Construction of Two-dimensional Preconditioner. For 2D nxn images, the matrices K^*K and L_β in (8) are block matrices of the following form:

$$A_{nn} = \begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,n} \\ A_{2,1} & A_{2,2} & \dots & A_{2,n} \\ \vdots & \ddots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \dots & A_{n,n} \end{pmatrix}.$$

Here $A_{i,j}$ are square matrices of order n.

In [14], T. Chan and Olkin proposed the Level-1 and Level-2 circulant preconditioners for such block matrices. Following their approach, we will define the Level-1 and Level-2 cosine transform preconditioners for A_{nn}. The idea of the Level-1 and Level-2 preconditioners is to approximate the matrix A_{nn} in one direction and two directions respectively. The Level-1 preconditioner is constructed by taking approximation to each sub-block of A_{nn} and Level-2 preconditioner is constructed based on the Level-1 preconditioner. More precisely, the Level-1 cosine transform preconditioner is defined by

$$c_1(A_{nn}) = \begin{pmatrix} c(A_{1,1}) & c(A_{1,2}) & \dots & c(A_{1,n}) \\ c(A_{2,1}) & c(A_{2,2}) & \dots & c(A_{2,n}) \\ \vdots & \ddots & \ddots & \vdots \\ c(A_{n,1}) & c(A_{n,2}) & \dots & c(A_{n,n}) \end{pmatrix}.$$

To define the Level-2 cosine transform preconditioner, let us first give some notations. For any n^2-by-n^2 block matrix A_{nn}, we denote $(A_{nn})_{i,j;k,l}$ to be the (i,j)th entry of the (k,l)th block of A_{nn}. Let P be a permutation matrix which simply reorders A_{nn} in another coordinate direction. More precisely, P satisfies

$$(P^t A_{nn} P)_{i,j;k,l} = (A_{nn})_{k,l;i,j}, \quad 1 \le i,j \le n, 1 \le k,l \le n.$$

Then the Level-2 cosine transform preconditioner $c_2(A_{nn})$ for A_{nn} is defined by

$$c_2(A_{nn}) = P c_1(P^t c_1(A_{nn}) P) P^t. \tag{13}$$

It is easy to show that the approximation $c_2(A_{nn})$ can be diagonalized by $C_n \otimes C_n$:

$$c_2(A_{nn}) = (C_n \otimes C_n)^t \ \mathrm{diag}((C_n \otimes C_n) A_{nn} (C_n \otimes C_n)^t) \ (C_n \otimes C_n).$$

DEFINITION 1. *Let $w = (w_1, \ldots, w_n)^t$ be an n-vector. Define $\sigma(w) \equiv (w_2, w_3, \ldots, w_n, 0)^t$. Define $\mathcal{T}_n(w)$ to be the n-by-n symmetric Toeplitz matrix with w as the first column and $\mathcal{H}_n(w)$ to be the n-by-n Hankel matrix with w as the first column and $(w_n, \ldots, w_1)^t$ as the last column.*

LEMMA 2. *$\mathcal{B}_{n\times n} = \{\mathcal{T}_n(w) + \mathcal{H}_n(\sigma(w)) \mid w = (w_1, \ldots, w_n)^t \in R^n\}$.*

Proof. By noting that $Q_i = \mathcal{T}_n(e_i) + \mathcal{H}_n(\sigma(e_i))$ for $1 \le i \le n$, the proof is similar to that of Lemma 2 in [11]. $\square$

Now computing the optimal cosine transform approximation can be reformulated as solving the n-dimensional minimization problem,

$$\min_{w=(w_1,\ldots,w_n)\in R^n} \|\mathcal{T}_n(w) + \mathcal{H}_n(\sigma(w)) - A_n\|_F. \tag{11}$$

The minimum can be calculated by setting $\frac{\partial}{\partial w_i}\|\mathcal{T}_n(w) + \mathcal{H}_n(\sigma(w)) - A_n\|_F^2 = 0$, for $i = 1, \ldots, n$. The following theorem gives the solution with an explicit formula for the first column of $c(A_n)$. Before that, let us give a notation. For any matrix $A_n = [a_{ij}]$, let r_n be an n-vector with k-th component given by

$$(r_n)_k = \sum_{(Q_k)_{i,j}\neq 0} a_{i,j}, \tag{12}$$

which is just the sum of those a_{ij} for which the corresponding entries of Q_k are nonzero.

THEOREM 1. *Let A_n be an n-by-n matrix and $c(A_n)$ be the minimizer of $\|B_n - A_n\|_F$ over all $B_n \in \mathcal{B}_{n\times n}$. Denote by q the first column of $c(A_n)$. If s_o and s_e are defined respectively to be the sum of the odd and even index entries of r_n, then we have, for n even,*

$$\begin{aligned}
[q]_1 &= \frac{1}{2n^2}(2n[r_n]_1 + n[r_n]_2 - 2s_e) \\
[q]_i &= \frac{1}{2n^2}(n[r_n]_i + n[r_n]_{i+1} - 2s_e) \qquad i = 2, \ldots, n-1 \\
[q]_n &= \frac{1}{2n^2}(-2ns_o + (2n-2)s_e + n[r_n]_n)
\end{aligned}$$

and for n odd,

$$\begin{aligned}
[q]_1 &= \frac{1}{2n^2}(2n[r_n]_1 + n[r_n]_2 - 2s_o) \\
[q]_i &= \frac{1}{2n^2}(n[r_n]_i + n[r_n]_{i+1} - 2s_o) \qquad i = 2, \ldots, n-1 \\
[q]_n &= \frac{1}{2n^2}(-2ns_e + (2n-2)s_o + n[r_n]_n).
\end{aligned}$$

Proof. The proof is similar to that of Theorem 1 and Corollary 1 in [11]. $\square$

If A_n has no special structure, then clearly by (12), r_n can be computed in $O(n^2)$ operations because the $Q_i's$ has only $O(n)$ non-zero entries each. If $A_n = [a_{i,j}]$ is a Toeplitz matrix (correspond to K in (8)), then the sum in (12) can be computed without explicit addition because real operations; see [30].

Let $\mathcal{B}_{n\times n}$ be the vector space containing all matrices that can be diagonalized by C_n. More precisely,

$$\mathcal{B}_{n\times n} = \{C_n^t \Lambda_n C_n \mid \Lambda_n \textit{ is an } n\text{−}by\text{−}n \textit{ real diagonal matrix}\}.$$

For an n-by-n matrix A_n, we choose our preconditioner $c(A_n)$ to be the minimizer of $||B_n - A_n||_F$ in the Frobenius norm in the space $\mathcal{B}_{n\times n}$. According to the terminology used in T. Chan, the approximation is called the *optimal cosine transform preconditioner* for A_n and denoted by $c(A_n)$. It can be shown that $c(A_n)$ is linear and preserves positive definiteness; see [18].

We will show in the following that $c(A_n)$ can be obtained optimally in $O(n^2)$ operations for general matrices. The cost can be reduced to $O(n)$ operations when A_n is a banded matrix, or a Toeplitz matrix (in our case L_β and K respectively) which is the same as that for constructing the optimal preconditioners. In order to construct $c(A_n)$ efficiently, one way is to make use of a sparse and structured basis for $\mathcal{B}_{n\times n}$, [11].

LEMMA 1. ***(Boman and Koltracht [3])*** *Let Q_i, $i = 1, \ldots, n$, be n-by-n matrices with the (h, k)th entry given by*

$$Q_i(h,k) = \begin{cases} 1 & \textit{if } |h-k| = i-1, \\ 1 & \textit{if } h+k = 2n-i+2, \\ 1 & \textit{if } h+k = i, \\ 0 & \textit{otherwise.} \end{cases} \qquad (10)$$

Then $\{Q_i\}_{i=1}^n$ is a basis for $\mathcal{B}_{n\times n}$.

We display the basis for the case $n = 6$.

$$Q_1 = \begin{pmatrix} 1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1 \end{pmatrix}, \quad Q_2 = \begin{pmatrix} 1&1&0&0&0&0\\1&0&1&0&0&0\\0&1&0&1&0&0\\0&0&1&0&1&0\\0&0&0&1&0&1\\0&0&0&0&1&1 \end{pmatrix}, Q_3 = \begin{pmatrix} 0&1&1&0&0&0\\1&0&0&1&0&0\\1&0&0&0&1&0\\0&1&0&0&0&1\\0&0&1&0&0&1\\0&0&0&1&1&0 \end{pmatrix},$$

$$Q_4 = \begin{pmatrix} 0&0&1&1&0&0\\0&1&0&0&1&0\\1&0&0&0&0&1\\1&0&0&0&0&1\\0&1&0&0&1&0\\0&0&1&1&0&0 \end{pmatrix}, Q_5 = \begin{pmatrix} 0&0&0&1&1&0\\0&0&1&0&0&1\\0&1&0&0&0&1\\1&0&0&0&1&0\\1&0&0&1&0&0\\0&1&1&0&0&0 \end{pmatrix}, \quad Q_6 = \begin{pmatrix} 0&0&0&0&1&1\\0&0&0&1&0&1\\0&0&1&0&1&0\\0&1&0&1&0&0\\1&0&1&0&0&0\\1&1&0&0&0&0 \end{pmatrix}.$$

In general, each Q_i has at most $2n$ non-zero entries.

In order to give a precise description of $\mathcal{B}_{n\times n}$, we introduce the following notations.

condition. If the boundary is rectangular, it was proved [12, 29] that the convergence rate of the PCG method with this preconditioner is independent of the grid size. In our present problem, Neumann boundary condition is imposed. Since the discrete Laplacian on a unit square with Neumann boundary conditions can be diagonalized by the discrete cosine transform matrix, this motivates us to use the optimal cosine transform approximation [6] to $L_\beta(u^k)$ as a preconditioner for the elliptic part in (8).

In addition, R. Chan, Ng and Wong [11] applied the sine transform approximation to construct preconditioners for Toeplitz systems. It gives rise to fast convergence of the PCG method. They mentioned that the PCG method with optimal cosine transform approximation can also produce the same convergence result. Therefore, it seems reasonable to use a cosine transform approximation to construct preconditioners for the system (8).

Our main idea in this paper is to propose a preconditioner by taking the sum of cosine transform approximation to the matrices K^*K and $L_\beta(u^k)$ separately, it can still be diagonalized by the discrete cosine transform matrix and therefore easily invertible.

In the next section, we will define and construct the optimal cosine transform approximation for a general matrix. In §3, we will use the approximation to construct a preconditioner for the system (8). In the final section, numerical performance of the preconditioner will be presented.

2. Optimal Discrete Cosine Transform Preconditioner. The concept of optimal transform approximation was first introduced by T. Chan [13]. Since preconditioners can be viewed as approximations to the given matrix A_n, it is reasonable to consider preconditioners which minimize $||B_n - A_n||$ over all B_n belonging to some class of matrices and for some matrix norm $||\cdot||$. T. Chan [13] proposed *optimal* circulant preconditioner that is the minimizer of the Frobenius norm $\|B_n - A_n\|_F$ over the class of all circulant matrices B_n. These preconditioners have been proved to be very effective preconditioners for solving Toeplitz systems with the PCG method; see [4]. Analogously, R. Chan, Ng and Wong [11] defined the optimal sine transform preconditioner to be the minimizer of $\|B_n - A_n\|_F$ over all matrices B_n which can be diagonalized by the discrete sine transform. They proved that for a large class of Toeplitz system, the PCG method with the sine transform preconditioner converges at the same rate as the optimal circulant one. Following the same approach, we are going to construct the optimal cosine transform preconditioner for general matrices and then apply it to precondition both K^*K and L_β in (8) separately. Recall that the reason for us to use "cosine" instead of "sine" is that the cosine transform approximation matches the Neumann boundary condition of the system (8). For a survey on fast transform type preconditioners, we refer the reader to [10].

2.1. Construction of One-dimensional Preconditioner. Let us denote C_n to be the n-by-n discrete cosine transform matrix. If δ_{ij} is the Kronecker delta, then the (i,j)th entry of C_n is given by

$$\sqrt{\frac{2-\delta_{i1}}{n}}\cos\left(\frac{(i-1)(2j-1)\pi}{2n}\right), \quad 1 \le i, j \le n, \tag{9}$$

see Sorensen and Burrus [25, p.557]. We note that the C_n's are orthogonal, i.e. $C_n C_n^t = I_n$. Also, for any n-vector v, the matrix-vector multiplication $C_n v$ can be computed in $O(n \log n)$

Many numerical schemes have been devised to obtain minimizer of the functional (4) by solving the gradient equation (7) directly. For example in [23, 24], an explicit time marching scheme for $u_t = -g(u)$ is used to solve (5). However, the time step is bounded above by a CFL condition. In [27], Vogel introduced the "lagged diffusivity fixed point iteration", which we denote by FP, to solve the system (7). If $A_\beta(u^k)$, K and L_β denote respectively the discretization matrices of $\mathcal{A}_\beta(u^k)$, $\mathcal{K}$ and $\mathcal{L}_\beta$, then the FP iteration will produce a sequence of approximations $\{u^k\}$ to the solution u and can be expressed as

$$(8) \qquad A_\beta(u^k)u^{k+1} \equiv (K^*K + \alpha L_\beta(u^k))u^{k+1} = K^*z, \qquad k = 0, 1, \ldots$$

In the denoising case ($K = I$), numerical experiment in [27] showed that the FP iteration gave a faster convergence rate than the time marching method. Note that in (8), obtaining u^{k+1} from u^k requires to solve a linear system with coefficient matrix $K^*K + \alpha L_\beta(u^k)$. For the denoising problem, the coefficient matrix is a sum of an identity matrix and a sparse symmetric positive definite matrix that arises in the discretization of the elliptic operator $\mathcal{L}_\beta(u^k)$. To solve such systems, one can use multigrid methods or ILU methods; see [27, 22, 26]. On the other hand, for the denoising and deblurring problem, K corresponds to a discretization of the convolution operator (2), and often K will be a Toeplitz matrix. Thus, the coefficient matrix in (8) will correspond to a sum of a convolution operator and an elliptic operator. We emphasize that it is not easy to devise fast iterative algorithms to solve this linear system. For example, the technique of applying multigrid method to solve such linear system is not yet well developed. Vogel and Oman [28] has recently proposed using a "product" preconditioner for (8) which allows the deblurring part K^*K and the PDE part L_β to be preconditioned separately. An alternative approach to solving the gradient equation (8) is to directly solve the minimization problem (4) by non-smooth optimization techniques; see for example [19, 20].

In this work, we apply the preconditioned conjugate gradient (PCG) method to solve (8) and we concentrate on finding a good preconditioner for (8). Given a matrix A, there are two criteria for choosing a preconditioner for A; see [17]. First, a preconditioner should be a "good" approximation to A. Secondly, it must be easily invertible. Recall that $A_\beta(u^k)$ corresponds to sum of a convolution operator and an elliptic operator. There are many "good" preconditioners for the individual parts. For example, for the elliptic part, we have the MINV preconditioner [15], the MILU preconditioner [16], the circulant and trigonometric type preconditioners [5, 6], multigrid and domain decomposition preconditioners [26]. For the convolution part, we have circulant type preconditioners [13] and sine transform preconditioners [11]. If A_1 and A_2 are "good" approximation for the elliptic part and the Toeplitz part in (8) respectively, then $A_1 + A_2$ should be a good approximation to $A_\beta(u^k)$. Therefore, $A_1 + A_2$ satisfies the first criteria of being a preconditioner. Unfortunately, the matrix-vector product $(A_1 + A_2)^{-1}v$ cannot be formed easily in general even though $A_1^{-1}v$ and $A_2^{-1}v$ can. Hence, this approach of constructing preconditioner for $A_\beta(u^k)$ cannot work in most situations. In this paper, we propose a preconditioner which is of the form $A_1 + A_2$ and which is easily invertible.

For matrices arising from elliptic boundary value problem, a "good" preconditioner must retain the boundary condition of the given operator [21]. Based on this idea, optimal sine transform preconditioners were constructed [12] for elliptic problems with Dirichlet boundary we need to simultaneously deconvolve and denoise the recorded image during the reconstruction process. We refer to this as the *denoising and deblurring* problem.

One of the successful approaches to estimate u from z is the Total Variation (TV) method of Rudin, Osher and Fatemi [24, 23]. They consider solving the following constrained minimization problem:

$$\min_u \int_\Omega |\nabla u| dx \qquad \text{subject to} \quad \|\mathcal{K}u - z\|^2 = \sigma^2 \tag{3}$$

where $\|\cdot\|$ denotes the norm on $L^2(\Omega)$ and σ is the noise level. The quantity $\int_\Omega |\nabla u| dx$ is called the *total variational* norm of u. This method is extremely effective for recovering blocky, discontinuous, function from noisy data.

Vogel considered the following closely-related regularization problem:

$$\min_u f(u) \equiv \min_u \frac{1}{2}\|\mathcal{K}u - z\|^2 + \alpha \int_\Omega |\nabla u| dx, \tag{4}$$

see [1, 27]. Here α is a positive parameter which measures the trade off between a good fit and an oscillatory solution. At a stationary point of (4), the gradient of f vanishes, giving:

$$\begin{aligned} g(u) \equiv \mathcal{K}^*(\mathcal{K}u - z) - \alpha \nabla \cdot \left(\frac{\nabla u}{|\nabla u|}\right) &= 0, \qquad x \in \Omega, \\ \frac{\partial u}{\partial n} &= 0, \qquad x \in \partial\Omega. \end{aligned} \tag{5}$$

The second term in g is obtained by taking the gradient of $\alpha \int_\Omega |\nabla u| dx$ and then applying integration by parts from which Neumann boundary condition results. We remark that the Euler Lagrange equation for (3) also has a form similar to (5).

Due to the term $1/|\nabla u|$, (5) is a degenerate nonlinear second order diffusion equation. The degeneracy can be removed by modifying the diffusion coefficient; see [27]. More precisely, if we let

$$\begin{aligned} \kappa_\beta(u) &= \frac{1}{\sqrt{|\nabla u|^2 + \beta}} \qquad \beta > 0, \\ \mathcal{L}_\beta(u)v &= -\nabla \cdot (\kappa_\beta(u) \nabla v) \end{aligned} \tag{6}$$

and

$$\mathcal{A}_\beta(u)v \equiv (\mathcal{K}^*\mathcal{K} + \alpha \mathcal{L}_\beta(u))v,$$

then (5) becomes the following non-degenerate system

$$\mathcal{A}_\beta(u)v = \mathcal{K}^* z, \qquad x \in \Omega, \qquad \text{with} \qquad \frac{\partial u}{\partial n} = 0, \qquad x \in \partial\Omega. \tag{7}$$

A recent survey of related PDE approach to image analysis can be found in [2].

COSINE TRANSFORM BASED PRECONDITIONERS FOR TOTAL VARIATION MINIMIZATION PROBLEMS IN IMAGE PROCESSING

RAYMOND H. CHAN*, TONY F. CHAN† AND CHIU-KWONG WONG‡

Abstract. Image reconstruction is a mathematically ill-posed problem and regularization methods must often be used in order to obtain a reasonable solution. Recently, the total variation (TV) regularization, as proposed by Rudin, Osher and Fatemi (1992), has become very popular for this purpose. In a typical iterative solution of the nonlinear regularization problem, such as the fixed point iteration of Vogel or Newton's method, one has to invert linear operators consisting of the sum of two distinct parts. One part corresponds to the blurring operator and is often a convolution; the other part corresponds to the TV regularization and resembles an elliptic operator with highly variable coefficients. In this paper, we present a preconditioner for operators of this kind which can be used in conjunction with the conjugate gradient method. It is derived from combining fast transform (e.g. circulant) preconditioners which the authors had earlier proposed for Toeplitz matrices and elliptic operators separately. Some numerical results will be presented.

Key words. total variation, image processing, denoising, deblurring, preconditioned conjugate gradient method.

1. Introduction. In this paper, we apply conjugate gradient preconditioners for the iterative solution of some large-scale image processing problems. The quality of the recorded image is usually degraded by blurring and noise. Given the recorded image, the blurring function and the noise distribution, the image restoration problem is to find an approximation to the true image. If we denote $\mathcal{K}$ to be the blurring operator and η the noise function, then the image restoration problem can be expressed as

$$z = \mathcal{K}u + \eta, \tag{1}$$

where z and u denote the functions containing the information of the recorded and original images. Note that when $\mathcal{K} = \mathcal{I}$, the identity operator, the image restoration problem means to extract image from a noisy image. This problem is usually referred to as the *denoising* problem. If $\mathcal{K}$ is a convolution operator,

$$\mathcal{K}u(x) = \int_\Omega k(x-y)u(y)dy, \tag{2}$$

* Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong. Research supported in part by HKRGC grant no. CUHK 178/93E and CUHK DAG grant no. 220600280.

† Department of Mathematics, University of California, Los Angeles, 405 Hilgard Avenue, Los Angeles, CA 90095-1555. Supported by grants NSF ASC-92-01266 and ONR-N00014-92-J-1890. Part of this work was performed during a visit to the Department of Mathematics at the CUHK.

‡ Department of Mathematics, University of California, Los Angeles, 405 Hilgard Avenue, Los Angeles, CA 90095-1555.