A PRACTICAL ALGORITHM FOR DIGITAL IMAGE RESTORATION

N.N. Abdelmalek, T. Kasvand, J. Olmstead, and M.M. Tremblay

National Research Council of Canada, Ottawa

ABSTRACT

In this paper we report some experimental results of a practical algorithm for the solution of digital image restoration problems. The solution is obtained directly from the system of linear equations which result from the discretization of the Fredholm integral equation of the first kind. This algorithm uses a simple regularized least squares technique. Also the regularization parameter for the optimum solution is calculated by a direct (non-iterative) method. A computer simulated example using both space-invariant and space-variant, spatially separable point spread functions, is presented. We show that this method compares favorably with other known direct methods.

RÉSUMÉ

La présente communication décrit certains résultats obtenus grâce à un algorithme pratique de résolution des problèmes de restitution des images numériques. La solution découle directement du système d'équations linéaires qui représente sous forme discrète l'intégrale de la première espèce de Fredholm. Cet algorithme fait intervenir une technique des moindres carrés simple et régularisée. En outre, le paramètre de régularisation de la solution optimale est calculé par une méthode directe (non itérative). Un exemple de simulation par ordinateur, faisant appel à des fonctions d'étalement des points, séparables et, soit variantes, soit invariantes dans l'espace, est présenté. Nous démontrons que cette méthode se compare favorablement à d'autres méthodes directes connues.

I. Introduction

In the linear model, the image restoration problem is described by the Fredholm integral equation of the first kind. The discretization of this equation gives a system of linear equations of the form

$$(1) \qquad g = [H]f + e,$$

where g is a stacked real m-vector representing the known or given degraded image, f is a stacked real n-vector representing the unknown or undegraded image and e is a stacked real m-vector representing the noise term. [H] is an mxn real matrix resulting from the discretization of the point spread function in the integral equation. If the known image is represented by an IxJ matrix, m=I.J. Also if the unknown image is represented by KxL matrix, then n=K.L. Without loss of generality we assume in this paper that $m \geq n$.

A classical approach for solving Eq. (1) is to calculate its least squares solution. However, Eq. (1) in general is ill-posed in the sense that small changes in vector g may cause large changes in the solution vector f. That is, ill-posed problems are also ill-conditioned. See for example Phillips[1].

A successful technique for overcoming the ill-posedness of Eq. (1) is to dampen or regularize its least squares solution. The damped least squares solution to system (1) is obtained from the normal equation

$$(2) \qquad ([H]^T[H] + \varepsilon [I])f = [H]^T g$$

where $[H]^T$ is the transpose of matrix [H] and [I] is an n unit matrix. The parameter ε is a small positive quantity known as the regularization parameter. See for example Rutishauser[2], p. 481, where he called this the relaxed least squares solution. See also a recent paper by Varah[3], p. 102.

Hence, assuming that the matrix in the l.h.s. of Eq. (2) is nonsingular, an approximate solution to Eq. (1) is given by

$$(3) \qquad f = ([H]^T[H] + \varepsilon [I])^{-1}[H]^T g$$

The parameter ε in Eq. (2) is increased or decreased, and a new solution is calculated each time. This is usually done a few times until a physically acceptable solution is obtained. The cost of these repeated solutions, in terms of the arithmetic operations count, is prohibitively high, if the problem is solved from scratch each time a repeated solution is calculated.

In the present work, the solution is obtained directly from Eq. (2). Hence both space invariant and space variant point spread function cases may be solved by this method. Once more, the regularization parameter for the best or near best solution is obtained by an inverse interpolation method not iteratively as in Hunt[4] and Reddi[5]. This results in considerable saving of computer time. We conclude that the present method compares favorably with other known direct methods.

II. Description of the present algorithm

Consider the matrix $([H]^T[H]+\varepsilon[I])$ in Eq. (2). Let $\varepsilon=(\varepsilon_1+\varepsilon_2)>0$, where $\varepsilon_1>0$ and $|\varepsilon_2|<\varepsilon_1$. Let also matrix $[C]$ be

(4a) $$[C] = ([H]^T[H]+\varepsilon_1[I]).$$

Then

(4b) $$([H]^T[H]+\varepsilon[I])=([C]+\varepsilon_2[I]).$$

We assume that ε is large enough such that matrix $([H]^T[H]+\varepsilon[I])$ is nonsingular and reasonably well conditioned, and also ε_1 is large enough such that matrix $[C]$ is nonsingular and reasonably well conditioned. Thus from (4b),

$$([H]^T[H]+\varepsilon[I])^{-1}=[C]^{-1}[[I]+\varepsilon_2[C]^{-1}]^{-1}$$

Hence provided that

(5) $$|\varepsilon_2|\cdot||[C]^{-1}|| <<1 ,$$

where $||\cdot||$ denotes any subordinate matrix norm, we may approximate the last equality and rewrite it as follows. See for example Stewart[6], p. 192.

$$([H]^T[H]+\varepsilon[I])^{-1}\simeq[C]^{-1}([I]-\varepsilon_2[C]^{-1})$$

Substituting in Eq. (3), the approximate solution f is given by

$$\hat{f} = [C]^{-1}[H]^Tg - \varepsilon_2[C]^{-2}[H]^Tg.$$

Or,

(6) $$\hat{f} = \hat{f}_0 - \varepsilon_2 v,$$

where $\hat{f}_0=[C]^{-1}[H]^Tg$ and $v=[C]^{-1}\hat{f}_0$.

Therefore, once the parameters ε_1 and ε_2 are correctly calculated, an approximate solution to the problem is given by (6). Also from (6), repeated solutions, i.e. for different values of ε_2 are easily calculated.

III. The regularization parameter

In this section we argue in favour of choosing a reasonable value for the parameter ε_1. Then we describe a method for calculating the parameter ε_2.

A. The choice of ε_1

It is known that the eigenvalues of matrix $[C]$ in (4a) are themselves the eigenvalues of matrix $([H]^T[H])$ with an ε_1 added to each one. Also $([H]^T[H])$ is a symmetric positive semi-definite matrix. That is its eigenvalues are real non-negative.

For low and moderate blur, matrix $([H]^T[H])$ is fairly well conditioned. That is its smallest eigenvalue is not very small. Thus adding a small positive parameter ε_1 to each of its diagonal elements would cause a small change to its smallest eigenvalues. Yet for severe blur, matrix $([H]^T[H])$ is nearly singular. That is its smallest eigenvalues are nearly zeros. Therefore by adding ε_1 to each of the diagonal elements of $([H]^T[H])$, the smallest eigenvalue of matrix $[C]$ would be approximately ε_1. Therefore, in general, we may state that the smallest eigenvalue of matrix $[C]=\alpha\varepsilon_1$, where $\alpha=1$ for severe blur and α is greater than 1 for low and moderate blur.

From this argument, for further use, we here state the following. Matrix $[C]$ is symmetric positive definite, its eigenvalues are themselves its singular values. Hence by considering the spectral norm, in (5),

(7) $$||[C]^{-1}||_2= 1/(\alpha\varepsilon_1).$$

Now, because of the round-off error accumulating in the calculation, a calculated parameter x say, is considered zero if $|x|<EPS$, where EPS is a specified tolerance. For the IBM 370 computer, the round-off level in single precision calculation is about 10^{-6}. For this computer, usually, we take $EPS=10^{-4}$. A reasonable choice of the parameter ε_1 would be of the order of $\sqrt{EPS}=0.01$. In this paper, we experimented with $\varepsilon_1=0.01$ and $\varepsilon_2=0.02$. It is found that this choice of ε_1 is adequate for problems solvable by other methods using the present form of point spread function matrix [H] of Gaussian distribution type. See part C in this section.

Matrix $[C]^{-1}$ in Eq. (4) is calculated by applying m Gauss-Jordan elimination steps with partial pivoting to matrix [C] and its updates. Since matrix [C] is symmetric positive definite, we pivot only over the diagonal elements of [C] and its updates. For the above choice of ε_1, for severe blur, the smallest pivot in the Gauss-Jordan steps is found to be, as expected[7,8], about 0.01 and 0.02 respectively for $\varepsilon_1=0.01$ and 0.02.

B. Calculating ε_2

To start, the parameter ε_2 should satisfy the inequality (5). From (7), the inequality (5) is satisfied if

$$|\varepsilon_2| << \alpha\varepsilon_1 \quad , \tag{8a}$$

where α is defined in section 3A above.

However, since it is only known that $\alpha>1$ for low and moderate blur, then for low and moderate blur, we may safely replace (8a) by the following inequality which does not include α, namely

$$|\varepsilon_2| < \varepsilon_1 \; . \tag{8b}$$

The method of calculating ε_2 is analogous to the method of calculating the parameter γ for a relevant problem by Hunt[4]. This method is based on the knowledge of the unbiased estimate of the variance denoted by $S^2(e)$ and of the mean denoted by $\mu(e)$ of the noise vector e in Eq. (1). It is assumed that $S^2(e)$ and $\mu^2(e)$ are known and thus e^2 is estimated.

From Eq. (1), the residual vector for the calculated solution $\hat{f}$ is given by

$$\rho = [H]\hat{f}-g.$$

Or by using (6),

$$\rho = \rho_0 - \varepsilon_2 u \; , \tag{9a}$$

where $\rho_0=[H]\hat{f}_0-g$ and $u=[H]v$. Then if the calculated solution $\hat{f}$ of (3) equals the ideal solution f of (1), $\rho^2=e^2$. Hence we here attempt to calculate ε_2 which results in ρ^2 being as near as possible to e^2.

From (9a)

$$\rho^2 = \rho^T\rho = \rho_0^2 - 2\varepsilon_2\rho_0^T u + \varepsilon_2^2 u^2. \tag{9b}$$

That is for values of ε_2 satisfying (8), the relation between ρ^2 and ε_2 is a vertical parabola. The vertex of this parabola is obtained at a negative value of ε_2.

The solution $\hat{f}$ is calculated from (6) for 3 different values of ε_2; namely for $\varepsilon_2=0$, $\pm\varepsilon_1/2$. The values of ρ^2 are calculated from (9) for the three values of ε_2.

Then inverse interpolation is used to calculate the parameter ε_2 which results in $\rho^2=e^2$. We used the inverse interpolation method described by Ralston[9], pp.57-62.

C. Some practical considerations

We here account for some practical situations concerning the parameters ε_1 and ε_2. That is when these parameters do not satisfy (8).

For low and moderate blur, particularly for large noise term e, the calculated parameter ε_2 is positive and may be $> \varepsilon_1$. The calculated solution vector $\hat{f}$ in this case would be inaccurate. This situation occured in our calculation when we took $\varepsilon_1=0.01$ for moderate blur with added large noise. This resulted in the restored image of Fig. 2d below. If this situation occurs, we may recalculate the problem with a larger value of ε_1 which equals the sum of the old values of ε_1 and ε_2. Then the new calculated ε_2 would be very small; almost zero.

For severe blur, for the case of no added noise, the calculated ε_2 is negative and nearly equals $-\varepsilon_1$. Again in this case, the calculation would be inaccurate. It is found in this case, that a value of $\varepsilon_2=-\varepsilon_1/2$, would both nearly satisfy the inequality (8a) for $\alpha=1$, and produce a small value of ρ^2 to be tolerated.

Finally for severe blur with added noise, the calculated restored image for a particular ε_1 may appear very noisy. A larger value of ε_1 would eliminate this noise but reduce the sharpness of the image.

IV. Experimental results

A computer program for the present algorithm is written in FORTRAN IV and has been tested on the IBM 370/3032 computer.

The example considered here is the classical image of the GIRL. To start, a portion of the image of the GIRL is decimated by taking every second pixel every second line. This gives the 103x64 matrix [F] in figure 1. The image [G] is obtained by blurring the ideal image [F] and adding noise to the blurred image.

We experimented with $\varepsilon_1=0.01$ and $\varepsilon_1=0.02$. We found that for severe blur with no added noise $\varepsilon_1=0.01$ is an adequate choice, although there is no much difference between the restored images for $\varepsilon_1=0.01$ and $\varepsilon_1=0.02$. For severe blur with added noise $\varepsilon_1=0.02$ was an adequate choice. Also for low and moderate blur with added noise, $\varepsilon_1=0.02$ was an adequate choice. See Fig 2 below where restored images for moderate blur with added noise are shown for $\varepsilon_1=0.01$ and $\varepsilon_1=0.02$.

The execution time per run for this example on the IBM 370/3032 computer is about 50 seconds for all cases. This time includes creating intermediate data sets for printing intermediate results.

We have compared the arithmetic operations count with other direct methods such as those of Refs. 4, 7, 8 and 10. From this and from the numerical results, we conclude that the present method compares favorably with other known direct methods.

We wish to thank Catherine Merritt for her assistance and for reading the manuscript. Also we acknowledge the help of Michael Duggan.

References.

1. D.L. Phillips, A technique for the numerical solution of certain integral equations of first kind, ACM Journal 9(1962), 84-97.

2. H. Rutishauser, Once again: The least square problem, Linear Algebra and its applications, 1(1968), 479-488.

3. J.M. Varah, A practical examination of some numerical methods for linear discrete ill-posed problems, SIAM Rev. 21(1979), 100-111.

4. B.R. Hunt, The application of constrained least squares estimation to image restoration by digital computer, IEEE Trans. Comput., C-22(1973), 805-812.

5. S.S. Reddi, Constrained least-squares image restoration: an improved computational scheme, Applied Optics 17(1978), 2340-2341.

6. G. Stewart, Introduction to matrix computations, Academic Press, New York 1973.

7. N.N. Abdelmalek, T. Kasvand and J.P. Croteau, Image restoration for space invariant point spread functions, Applied Optics 19(1980), 1184-1189.

8. N.N. Abdelmalek and T. Kasvand, Digital image restoration using quadratic programming, Applied Optics, 19(1980), 3407-3415.

9. A. Ralston, A first course in numerical analysis, McGraw-Hill, New York 1965.

10. H.S. Hou and H.C. Andrews, Least squares image restoration using spline functions, IEEE Trans. Computers, C-26(1977), 856-873.

Figure 1. The 103x64 matrix representing a portion of the image of the GIRL decimated by taking every second pixel every second line.

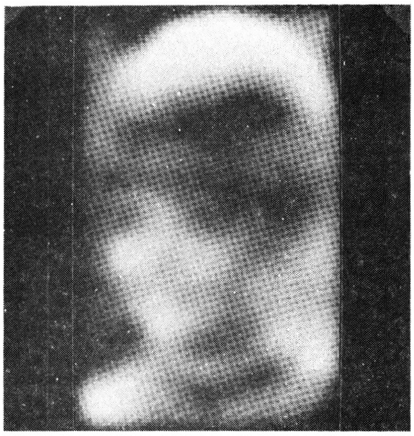
(a) (b)

(c) (d)

(e) (f)

Figure 2. Restoration from moderate SIPSF blur (a) Blurred noisy image, with additive Gaussian noise, mean =0 standard deviation =1, S/N =1750; (b) Blurred noisy image, with additive Gaussian noise, mean =0, standard deviation =3, S/N =195; (c) Restored for (a) for ϵ_1=0.01. (d) Restored for (b) for ϵ_1=0.01. (e) Restored for (a) for ϵ_1=0.02. (f) Restored for (b) for ϵ_1=0.02.

(a)



(b)

(c)

(d)

Figure 3. Restoration from severe SIPSF blur (a) Blurred image, with no additive noise; (b) Blurred noisy image, with additive Gaussian noise, mean =0, standard deviation =0.5, S/N =7000; (c) Restored for (a) for $\epsilon_1 = 0.02$. (d) Restored for (b) for $\epsilon_1 = 0.02$.

(a) (b)

(c) (d)

(e) (f)

Figure 4. Restoration from SVPSF blur. (a) Blurred image with no noise, low blur. (b) Blurred image with no noise, medium blur. (c) Blurred image with no noise, severe blur. (d) Restored for (a) for $\epsilon_1=0.01$. (e) Restored for (b) for $\epsilon_1=0.01$. (f) Restored for (c) for $\epsilon_1=0.01$.