Projection-based blind deconvolution

Yongyi Yang, Nikolas P. Galatsanos, and Henry Stark

Department of Electrical and Computer Engineering, Illinois Institute of Technology, 3301 South Dearborn Street, Chicago, Illinois 60616

Received December 9, 1993; revised manuscript received April 5, 1994; accepted April 12, 1994

We present a new projection-based algorithm for solving the classical blind-deconvolution problem. In our approach all known *a priori* information about both the unknown source and the blurring functions is expressed through constraint sets. In computer simulations the algorithm performed well even when the prior information was not accurate. To see how well our algorithm compares against others, we compared it with another recently published deconvolution method [J. Opt. Soc. Am. A **9**, 932 (1992)]. The advantages of each method are discussed.

1. INTRODUCTION

The blind deconvolution problem refers to a class of problems of the form

$$g(\mathbf{x}) = \int f(\mathbf{x} - \mathbf{y})h(\mathbf{y})d\mathbf{y} + n(\mathbf{x}), \qquad (1)$$

where $g(\mathbf{x})$ is the observed data, $f(\mathbf{x})$ is the unknown source signal, $h(\mathbf{x})$ is the unknown blurring function (impulse response) of the system, and $n(\mathbf{x})$ is noise. Recovering both $f(\mathbf{x})$ and $h(\mathbf{x})$ from $g(\mathbf{x})$ alone is a difficult problem.

With or without noise, Eq. (1) admits of an infinite number of solutions if no other constraints are available for $f(\mathbf{x})$ and $h(\mathbf{x})$. However, once constraints are imposed, the set of feasible solutions to Eq. (1) can be greatly reduced. A number of algorithms have been proposed for solving the blind-deconvolution problem. The most commonly used technique is described in Ref. 1. It is an iterative algorithm according to which the constraints (*a priori* knowledge) about both $f(\mathbf{x})$ and $h(\mathbf{x})$ are enforced. A difficulty with this algorithm is the lack of rigorous justification of its convergence, making it difficult to determine when to stop the iterations. Indeed Lane[2] argues that the general iterative deconvolution loop of Ayers and Dainty[1] can diverge in the sense that a new estimate can be, both visually and computationally, worse than the previous ones. In Ref. 3 an error function was minimized subject to the positivity constraint to produce a solution by simulated annealing. However, for a large-scale problem, this approach is not numerically efficient. In Ref. 4 a steepest-descent algorithm is used to minimize a cost function directly. However, owing to the nonconvexity of the cost function used, avoidance of suboptimal solutions is a problem. In a recent paper[2] Lane proposes an iterative algorithm based on conjugate gradient minimization in which he incorporates features that yield the stable convergence properties of Fienup's error-reduction algorithm.[5] In this algorithm the constraints are incorporated into an error function, and the problem is reformulated as an unconstrained minimization problem. Since the constraints are not enforced directly, it is not guaranteed that the convergent point will satisfy them. Furthermore, this algorithm is not scale invariant. In other words, the final solution of this iterative algorithm is sensitive to scaling by a constant factor in a nontrivial fashion.

In this paper we present a blind-deconvolution algorithm that is rooted in the theory of generalized projections.[6] In this approach the solution pair $f(\mathbf{x})$ and $h(\mathbf{x})$ is treated as an element in a Hilbert space; thus all *a priori* knowledge about both the source signal and the blurring function is expressed through constraint sets in this space. Since the solution must belong to the intersection set of all the constraint sets, it can be found through alternating projections. However, finding projections onto nonconvex sets of 2-tuples is not easy. Even if these projections can be found, numerical difficulties exist that must be overcome. We discuss these points further in Section 2. The point, however, is that implementation of pure-projection algorithms for the blind-deconvolution problem may not be a realistic possibility. For this reason we modify a pure-projection algorithm into one that still permits incorporation of prior knowledge yet is relatively simple to realize and has desirable convergence properties.

The rest of this paper is organized as follows. In Section 2 we present the generalized-projection formulation of the blind-deconvolution problem. In Section 3 we present the proposed new blind-deconvolution algorithm. In Section 4 we review the deconvolution algorithm presented in Ref. 2 and explain some of its shortcomings. In Section 5 we present numerical experiments in which we test and compare the proposed algorithm with the algorithm of Ref. 2 for both the noise-free and the photon-limited image data, using similarity[7] as the metric for objective comparison. Finally, in Section 6 we present our conclusions.

2. GENERALIZED-PROJECTION-BASED ALGORITHM

For the noiseless case Eq. (1) becomes

$$g(\mathbf{x}) = \int f(\mathbf{x} - \mathbf{y})h(\mathbf{y})d\mathbf{y}. \qquad (2)$$

0740-3232/94/092401-09$06.00

In this study we assume that both $f(\mathbf{x})$ and $h(\mathbf{x})$ in Eq. (2) are of finite energy; in other words, both are square integrable functions. Let L^2 be the space of square integrable functions. The space that we are working with is $\mathcal{H} = \{(u, v) : u \in L^2, v \in L^2\}$, which carries a norm

$$\|(u, v)\| \triangleq \left[\int |u(\mathbf{x})|^2 d\mathbf{x} + \int |v(\mathbf{x})|^2 d\mathbf{x} \right]^{1/2}.$$

Let us now define the set

$$C_g \triangleq \{(u, v) : u \odot v = g\} \tag{3}$$

as the set of all solution pairs, i.e., the set of all function pairs $u(\mathbf{x})$ and $v(\mathbf{x})$ such that

$$g(\mathbf{x}) = \int u(\mathbf{x} - \mathbf{y}) v(\mathbf{y}) d\mathbf{y}. \tag{4}$$

The symbol $\odot$ denotes convolution. We shall reserve the symbols f and h for the true source and the true blurring function, respectively. Clearly, the pair $(f, h) \in C_g$.

Even to readers familiar with the theory of alternating projections, the object defined in Eq. (3) may seem strange. It is different from the well-known constraint sets in that it is a set whose elements are 2-tuples i.e., function pairs; thus the constraint is simultaneously imposed on two functions. As is to be expected, the projection onto C_g is considerably more difficult to compute than in the vast majority of ordinary constraint sets, i.e., those involving only a single function. In general, the set C_g contains infinitely many elements each of which is a feasible solution to the blind-deconvolution problem if no further information is provided about either the signal $f(\mathbf{x})$ or the system $h(\mathbf{x})$ or both. Thus we must use as much prior knowledge about f and h as possible in order to exclude spurious solutions in the set C_g and obtain a meaningful solution.

Let C_f and C_h denote the constraint sets based on the prior knowledge about f and h, respectively; i.e.,

$C_f \triangleq \{(u, v) : u$ satisfying prescribed prior knowledge about the source $f\}$,

$C_h \triangleq \{(u, v) : v$ satisfying prescribed prior knowledge about the blurring function $h\}$.

Then an element in the set $C_0 \triangleq C_g \cap C_f \cap C_h$ will be a solution to Eq. (2), which also satisfies all the available prior knowledge. In general, the set C_0 contains fewer elements than the set C_g, making the solution physically more meaningful.

The definitions of the sets C_f and C_h are based on available prior knowledge, and the sets are problem dependent. Usually they are determined by our understanding of the problem. Examples of such sets are

1. Sets based on the support region of the signal:

$$C_f \triangleq \{(u, v) : u(\mathbf{x}) = 0, \forall x \notin \Omega_f\},$$
$$C_h \triangleq \{(u, v) : u(\mathbf{x}) = 0, \forall x \notin \Omega_h\},$$

where Ω_f and Ω_h denote the support of f and h, respectively.

2. Sets based on the intensity range, for example,

$$C_f \triangleq \{(u, v) : u(\mathbf{x}) \in (f_{\min}, f_{\max})\},$$

where $f_{\min}$ and $f_{\max}$ are determined by the specific problem.

3. Sets based on the physical properties of the system; for example, if the system is band limited, then a constraint set can be defined as

$$C_h \triangleq \{(u, v) : v \leftrightarrow V(\omega) = 0, \forall \omega \notin \Omega_H\},$$

where Ω_H is the support of the Fourier transform of h.

4. Sets based on experimental measurement; for example, if we have some gross measurement $\tilde{h}(\mathbf{x})$ about the system blurring function $h(\mathbf{x})$, then we have

$$C_h \triangleq \{(u, v) : \|v - \tilde{h}\| \leq E\},$$

where E is some constant that can be determined by the measurement accuracy.

A brief remark is in order: the sets described in examples 1–4 have elements that are 2-tuples, but the constraint is applied to only one member of the function pair. In this sense these sets are much more like the constraint sets used in alternating projections than like the set in relation (3).

Let P_g, P_f, and P_h denote the projection operators onto the set C_g, C_f, and C_h, respectively; then the generalized-projection theory guarantees that the recursion

$$(f, h)_{(k+1)} = P_h P_f P_g (f, h)_k, \qquad k = 0, 1, 2, \ldots, \tag{5}$$

where $(f, h)_k$ is the estimated recovery of (f, h) at the kth iteration and $(f, h)_0$ is our initial estimate, i.e., the starting point, will converge to a point in C_0 provided that the algorithm does not stagnate at erroneous solutions called traps.[6] Note: in Ref. 6 Levi and Stark showed that an algorithm like the one in Eq. (5), i.e., involving nonconvex sets, has the summed distance error-convergence property if the number of constraint sets involved is at most two. However, as the reader can see, Eq. (5) involves three sets C_h, C_f, and C_g. The statement surrounding Eq. (5) is still true, because it is readily shown that if $C_h \cap C_f \triangleq C_{hf}$, then $P_{hf} = P_h P_f$, and thus, despite its appearance, Eq. (5) can be thought of as involving only two sets.

The projectors P_f and P_h corresponding to the sets defined above are well known. The projector P_g onto set C_g can be derived as follows: consider any element (f', h') in $\mathcal{H}$ but not in C_g; its projection onto set C_g, denoted $(\tilde{f}, \tilde{h})$, is solved by minimization of the quantity $\|(u, v) - (f', h')\|$ under the constraint that $(u, v) \in C_g$. The minimization can be done by use of the Lagrange multiplier technique. The solution can be given by the two-step procedure:

a. Solve for $\tilde{F}(\omega) = \mathcal{F}[\tilde{f}(\mathbf{x})]$ from

$$|\tilde{F}|^4 - |\tilde{F}|^2 \tilde{F} F' + H' G^* \tilde{F} = |G|^2, \tag{6}$$

where $\mathcal{F}$ is the Fourier transform operator, $F' = \mathcal{F}(f')$, $H' = \mathcal{F}(h')$, $G = \mathcal{F}(g)$, and G^* denotes the complex conjugate of G.

b. Solve for $\tilde{H}(\omega) = \mathcal{F}[h(\mathbf{x})]$ from

$$\tilde{H}(\omega) = G(\omega)/\tilde{F}(\omega). \tag{7}$$

One finds the projection by solving implicit nonlinear equation (6). This can be done numerically. We must point out that there might be more than one solution to Eq. (6), because the set C_g is nonconvex. Another difficulty in finding the projection is that in the region where $\tilde{F}(\omega)$ is small, numerical inaccuracies pose a problem in the computation of Eq. (7).

3. PROPOSED ALGORITHM

In addition to the numerical difficulties in solving Eq. (6) as discussed above, traps must also be avoided when one is implementing the algorithm of Eq. (5). Therefore the generalized-projection algorithm presented above is not practical to implement. Instead, consider the functional

$$J_{(u,v)} \triangleq \|g - u \odot v\|^2 = \int [g(\mathbf{x}) - (u \odot v)(\mathbf{x})]^2 d\mathbf{x} \geq 0.$$

Any element (u, v) in the set C_g is the point at which $J_{(u,v)}$ assumes its global minimum. The algorithm in Eq. (5) will converge to a feasible solution or stagnate at a trap (a morphological local minimum). At a feasible solution the known constraints imposed by the sets C_f and C_h are satisfied and $J_{(u,v)} = 0$, since the solution has membership in C_g. However, to avoid the serious mathematical problems associated with solving Eqs. (6) and (7), we propose an alternative algorithm that, while still subject to erroneous solutions associated with local minima of J, is significantly easier to implement. The algorithm involves the following steps:

1. Take an educated initial guess of (f, h) and call this (f_0, h_0). In the absence of any further knowledge, set $f_0 = P_f g$ and $h_0 = 0$.
2. Solve for

$$h_k = \arg\{\min_{h \in C_h} J_{(f_{k-1},h)}\}.$$

The expression is shorthand for "find the variable h that minimizes $J_{(f_{k-1},h)}$ subject to h lying in the set C_h and set this value of h equal to h_k."
3. Solve for

$$f_k = \arg\{\min_{f \in C_f} J_{(f,h_k)}\}.$$

4. Set $k = k + 1$; repeat step 2 and step 3 until convergence is achieved.

Theorem. The iteration defined above will converge to a local or global minimum of the functional $J_{(u,v)}$.

Proof. Note from steps 2 and 3 that

$$J_{(f_k,h_k)} \leq J_{(f_{k-1},h_k)} \leq J_{(f_{k-1},h_{k-1})}$$

and that $J(u, v)$ is nonnegative for all $(u, v) \in \mathcal{H}$. Hence the nonincreasing sequence converges either to zero or to a local minimum. ■

Remarks

1. If $J_{(f_k,h_k)}$ converges to 0, then a global minimum is found, and the resulting (f, h) will also be a stationary point of the earlier generalized-projection-based algorithm in Eq. (5).
2. If more-accurate information is available about h than about f, we can interchange the role of f with the role of h in the algorithm.
3. In earlier studies[2–4] a functional of similar nature is minimized to yield the solution. Because $J_{(u,v)}$ is a nonconvex function of (u, v), the proposed minimization algorithms are not efficient numerically.
4. Although stagnation at a local minimum is always a possibility, with the initialization given in step 1 our algorithm did not stagnate at local minima in our numerical work.

The functional $J_{(u,v)}$ is a quadratic function with respect to u if v is fixed, and vice versa. Therefore the minimization in steps 2 and 3 can be done by a standard gradient-projection method.[8,9] In step 2 the following iteration will yield the global minimum: for the kth cycle

$$h^{l+1} = P_h[h^l - \alpha_k \nabla J_{(f_{k-1},h^l)}], \tag{8}$$

where ∇ is the gradient operator, P_h is the projection operator onto C_h, $h^0 = h_{k-1}$, and α_k is a constant chosen to guarantee convergence.[9]

Equation (8) needs some discussion. First, observe that Eq. (8) is a contraction mapping, since P_h is nonexpansive and that for the proper value of α_k, the operation in brackets is a contraction. Also, the numerical realization of the functional $J_{(f,h)} = \|g - h \odot f\|^2$ when f is held fixed at, say, the vector value $\mathbf{f}_k$ becomes $J = \|\mathbf{g} - \mathbf{F}_k\mathbf{h}\|^2$, where $\mathbf{g}$ and $\mathbf{h}$ are vectors and $\mathbf{F}_k$ is a circulant/Toeplitz matrix of the shifted $\mathbf{f}_k$'s. The gradient of J is then easily computed to be

$$\nabla J = 2\mathbf{F}_k{}^t\mathbf{F}_k\mathbf{h} - 2\mathbf{F}_k{}^t\mathbf{g}, \tag{9}$$

where superscript t denotes the transpose. Let λ_k denote the largest eigenvalue of the matrix $\mathbf{F}_k{}^t\mathbf{F}_k$; then it is well known that Eq. (8) will lead to convergence, i.e., be a contraction mapping for $0 < \alpha_k < 1/\lambda_k$. For the numerical computation of Eq. (9) the two-dimensional discrete Fourier transform and the diagonalization properties of circulant matrices by the discrete Fourier transform are used.[10]

Another remark about Eq. (8) is relevant. We note that Eq. (8) interrupts the gradient search at each iteration by projecting onto the constraint set C_h. Why not let the gradient portion of the algorithm iterate until it finds an unconstrained global minimum and then apply the constraint? Logical as this procedure sounds, it could lead to the wrong answer, as the following simple but nontrivial example shows. Let the functional to be minimized be given by

$$J(x, y) = x^2 + \alpha y^2, \qquad \alpha > 0, \tag{10}$$

subject to the constraint described by the set

$$C = \{(x, y) : x \geq 0, y \geq 0, x + y = 1\}. \tag{11}$$

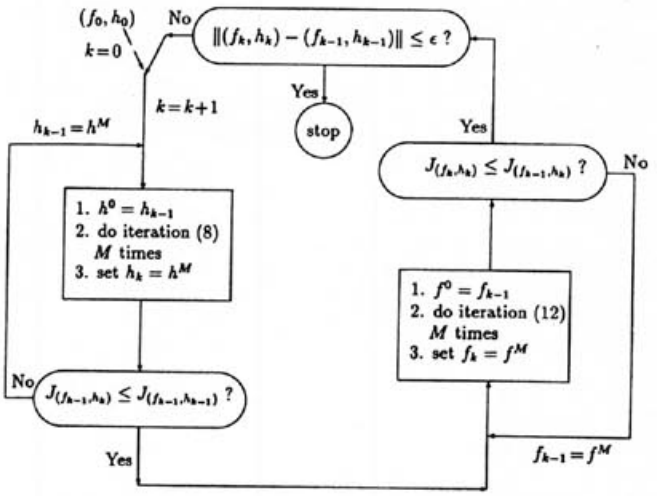


Fig. 1. Block diagram of the proposed algorithm.

If we run the gradient algorithm first, we obtain a minimum at $x = y = 0$. Subsequent projection onto C yields the point $x = y = 1/2$. However, if we apply the constraint before carrying out the gradient algorithm, we obtain as the constrained minimum the point $x = \alpha/(1 + \alpha)$, $y = 1/(1 + \alpha)$, which is the correct solution. Thus what seems to be logical at first glance turns out to be incorrect on reflection.

Step 3 can be done in a similar fashion:

$$f^{l+1} = P_h[f^l - \beta_k \nabla J_{(f^l,h_k)}], \tag{12}$$

where $f^0 = f_{k-1}$ and P_f is the projection operator onto C_f.

The iterations in Eqs. (8) and (12) have linear convergence rate.[11] The convergence becomes slow, especially when the solutions are close to the true minimum. In order to avoid this kind of behavior, we propose an algorithm, which is illustrated in Fig. 1.

In Fig. 1, M is a prescribed number; it can be 10, 20, etc. For example, in our numerical examples given below, it is chosen to be 40. Also in Fig. 1, ϵ is a prescribed constant serving as the convergence criterion.

From the proof of the theorem above, the convergence of the algorithm presented in Fig. 1 is guaranteed.

4. CONJUGATE-GRADIENT BLIND-DECONVOLUTION ALGORITHM

In a recent paper[2] Lane described a blind-deconvolution algorithm that, he argued, was superior to the general-deconvolution loop of Ayers and Dainty.[1] The algorithm by Lane performs deconvolution by minimizing

$$E_c = E_i + E_f, \tag{13}$$

where

$$E_f \triangleq \int [G(\omega) - F(\omega)H(\omega)] \times [G(\omega) - F(\omega)H(\omega)]^* \mathrm{d}\omega, \tag{14}$$

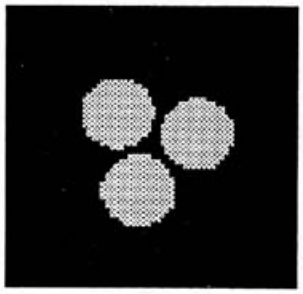
and F, H, and G are the Fourier transforms of f, h, and g, respectively. Also,

$$E_i \triangleq \int_{\overline{\Omega}_f} |f(\mathbf{x})|^2 \mathrm{d}\mathbf{x} + \int_{\overline{\Omega}_h} |h(\mathbf{x})|^2 \mathrm{d}x, \tag{15}$$

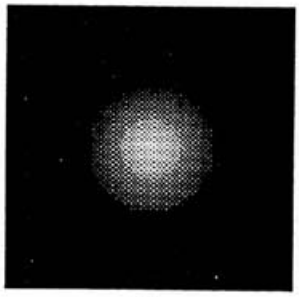
where $\overline{\Omega}_f$ and $\overline{\Omega}_h$ denote the sets of points where $f(\mathbf{x})$ and $h(\mathbf{x})$, respectively, violate their prior known constraints. Note that $E_f = J_{(f,h)}$.

Among the advantages of this approach is that the minimization problem is well defined, and conventional minimization routines can be used for its implementation. Another advantage of this approach is that it can be used in problems in which support constraints are inexact. In Ref. 2 a conjugate-gradient algorithm was used, and it was pointed out that the solution depends on the initial starting point. It is easy to see that the pairs of functions (f_k, h_k) and $[\alpha f_k, (1/\alpha)h_k]$, where α is any constant, yield the same data $g(\mathbf{x})$. However, the objective function E_c that is minimized yields different values for the equivalent pair of solutions (f_k, h_k) and $[\alpha f_k, (1/\alpha)h_k]$. Thus the deconvolution algorithm is sensitive to brightness levels.

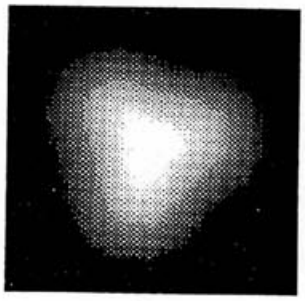
The implication of this observation is that the penalty term E_i cannot characterize the constraints (i.e., the prior knowledge) well. More specifically, when α is large, the values of h do not influence the value of E_i much. Thus the obtained solution is not sensitive to h even when h violates its constraints. On the other hand, when α is small the same problem occurs with f.



(a)



(b)



(c)

Fig. 2. (a) Original image, (b) blurring function, (c) blurred

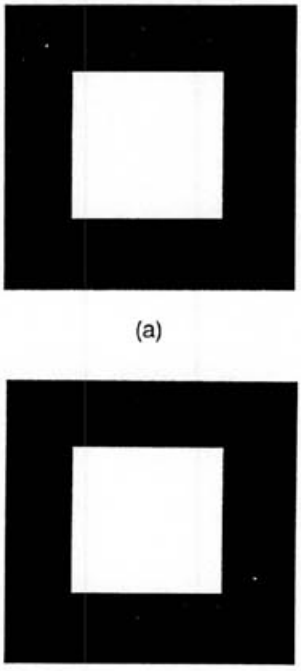


Fig. 3. (a) Support window for f: W_f, (b) support window for h: W_h.

This sensitivity to brightness levels is a serious disadvantage of this approach, because the metric that is optimized to produce the solution has different values for equivalent solutions. Furthermore, the choice of E_i in Eq. (15) cannot guarantee that the final solution will satisfy the constraints exactly. This difficulty limits the ability of this algorithm to move away from the local minima of E_c even when at these local minima the imposed constraints are violated.

5. NUMERICAL EXAMPLES

In this section we present two numerical examples to test the proposed algorithm. The proposed algorithm is also compared with the algorithm of Ref. 2. As a quantitative measure of the quality of the reconstructed images, we propose the similarity defined as

$$S_f(\hat{f}) = \max_{\mathbf{x}_0} \frac{\int f(\mathbf{x})\hat{f}(\mathbf{x} - \mathbf{x}_0)\mathrm{d}\mathbf{x}}{\|f\| \cdot \|\hat{f}\|}, \tag{16}$$

where $\hat{f}$ is the reconstructed image, f is the original image, and $\|\cdot\|$ denotes the norm.

The merit of this metric for evaluating image quality is its scale-invariant property; i.e., $S_f(\hat{f}) = S_f(\beta\hat{f})$ for any positive scalar β. Also, the value of $S_f(\hat{f})$ is always in the range of $[-1, +1]$. As can be seen, $S_f(\hat{f})$ is a measure of the similarity of the shapes of f and $\hat{f}$ rather than of their mean-square distance. Thus the similarity is better suited to evaluating the results of blind-deconvolution problems in which the resulting signal shape is more important than the distance of the resulting signals. This metric was used in Ref. 7 to define a constraint set for signal recovery and is more closely correlated with visual criteria than is the mean-square error. Thus in quantitative comparisons with other methods, the similarity metric is more likely to reflect the visual quality of the reconstruction. We use it in the numerical examples that follow.

Example 1. In this example, the original signal $f(\mathbf{x})$ is a 64×64 image of a triple star modeled by three circles, as shown in Fig. 2(a). The blurring function is modeled by a truncated Gaussian-shaped function:

$$h(x_1, x_2) = \begin{cases} 1/2\pi\sigma^2 \exp[-(x_1^2 + x_2^2)/2\sigma^2] & \text{for } x_1^2 + x_2^2 \le 4\sigma^2, \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

where σ is chosen to be 7 in this example. The blurring function $h(\mathbf{x})$ is shown in Fig. 2(b). The blurred image $g(\mathbf{x})$ is shown in Fig. 2(c).

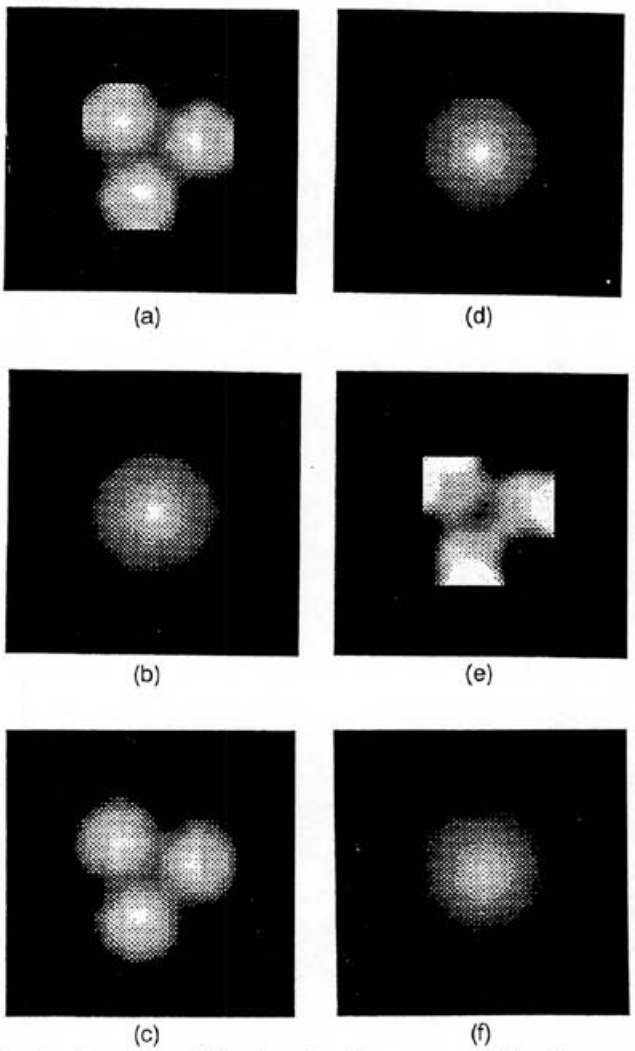


Fig. 4. Images and blurring functions recovered by the proposed algorithm with different support windows: (a) and (b) case 1, (c) and (d) case 2, (e) and (f) case 3.

Table 1. Similarity Metric of the Recovered Source and Blurring Functions of Fig. 4

Fig. 4	Case 1	Case 2	Case 3
S_f	0.9464	0.9328	0.8866
S_h	0.9963	0.9916	0.9957

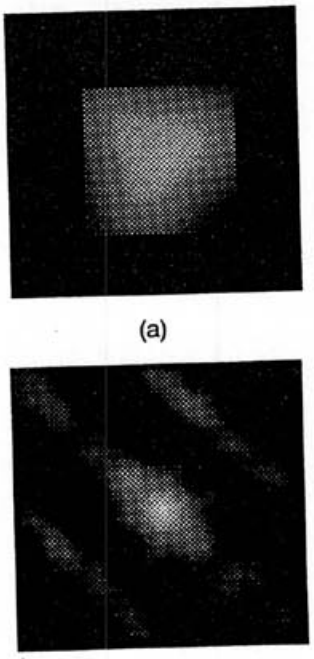


Fig. 5. (a) Image, (b) blurring function recovered by the algorithm in of Ref. 2 with the same support window as in case 1.

Table 2. Similarity Metric of the Recovered Source and Blurring Functions of Fig. 5

Fig. 5	(a) and (b)
S_f	0.8397
S_h	0.7372

To reconstruct $f(\mathbf{x})$ from data $g(\mathbf{x})$ we use the following constraints:

$$C_f = \{(u, v) : u(\mathbf{x}) \text{ is nonnegative for } \mathbf{x} \in W_f, \quad 0 \text{ otherwise}\},$$

where W_f is a square region centered at (0, 0) with dimension 33 × 33, as shown in Fig. 3(a). In other words, C_f implies that the original image is spatially limited.

Similarly, for the blurring function we have

$$C_h = \{(u, v) : v(\mathbf{x}) = 0, \forall \mathbf{x} \notin W_h\},$$

where W_h is a square region centered at (0, 0) with dimension 29 × 29, as shown in Fig. 3(b).

In this example an initial guess for f_0 is taken as

$$f_0(\mathbf{x}) = \begin{cases} g(\mathbf{x}) & \text{if } \mathbf{x} \in W_f, \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

and 0 is used for h_0.

In practice, the exact support information of the original image and the blurring function is rarely known, and therefore it would be unfair for us to assume perfect knowledge of this in evaluating the algorithm. Nevertheless, when both the signal and the blurring function are positive, the sum of the spatial supports of the signal and the blurring function is equal to the spatial support of the observed data. In this example the proposed algorithm was tested with three different spatial support constraints:

Case 1. W_f, 33 × 33 and W_h, 29 × 29; i.e., W_f and W_h are the smallest squares that contain f and h, respectively.

Case 2. W_f, 37 × 37 and W_h, 25 × 25; i.e., W_f is overestimated, and W_h is underestimated.

Case 3. W_f, 29 × 29 and W_h, 33 × 33; i.e., W_f is underestimated, and W_h is overestimated.

The reconstructed images of the source and the blurring functions are shown for cases 1, 2, and 3 in Figs. 4(a) and 4(b), 4(c) and 4(d), and 4(e) and 4(f), respectively. The

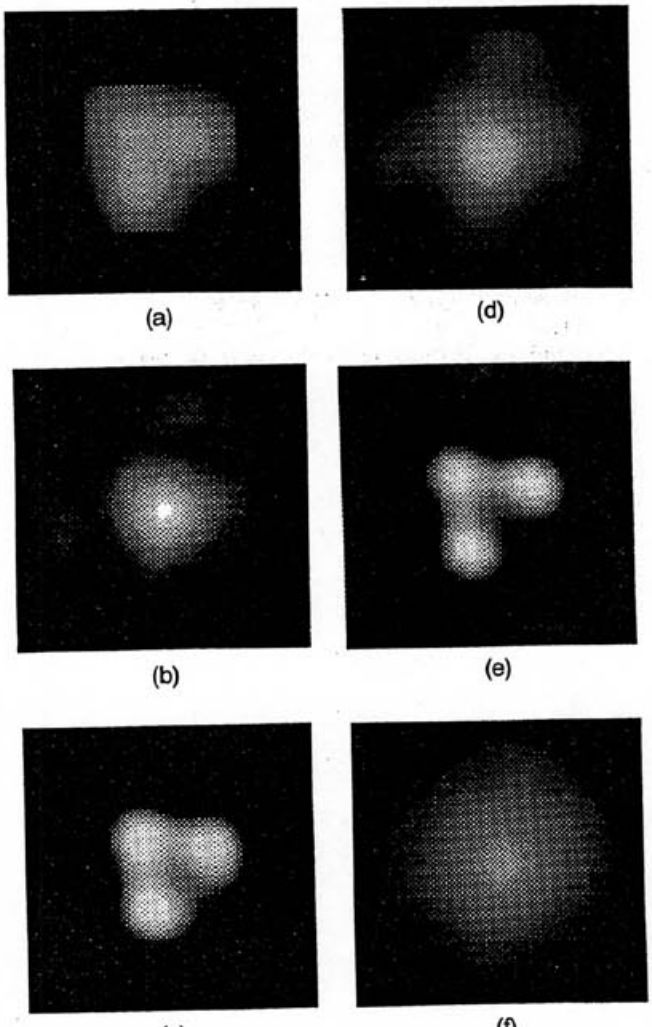


Fig. 6. Images and blurring functions recovered by the algorithm of Ref. 2 with the scaled initial conditions: (a) and (b) $\alpha = 0.1$, (c) and (d) $\alpha = 0.025$, (e) and (f) $\alpha = 0.01$.

Table 3. Similarity Metric of the Recovered Source and Blurring Functions of Fig. 6

Fig. 6	$\alpha = 0.1$	$\alpha = 0.025$	$\alpha = 0.01$
S_f	0.8543	0.8505	0.8101
S_h	0.9529	0.8216	0.7598

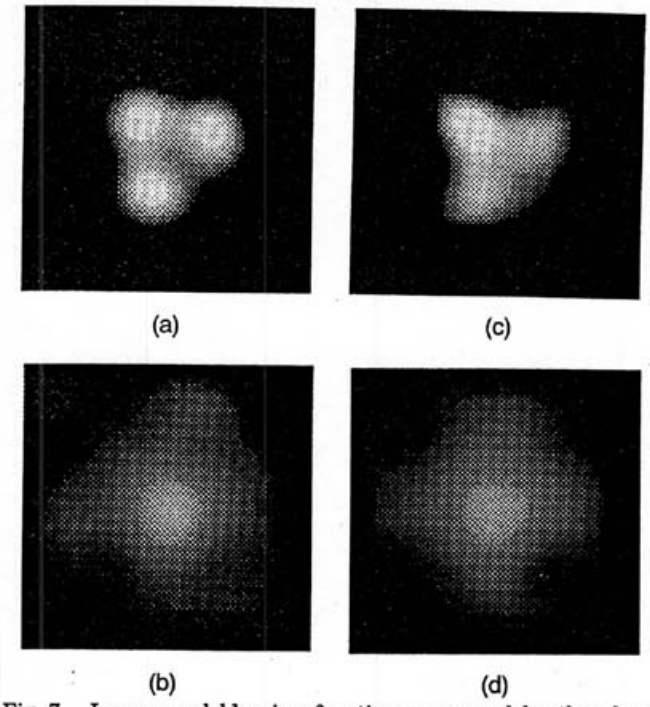


Fig. 7. Images and blurring functions recovered by the algorithm of Ref. 2 for (a) and (b) case 2, (c) and (d) case 3.

Table 4. Similarity Metric of the Recovered Source and Blurring Functions of Fig. 7

Fig. 7	Case 2	Case 3
S_f	0.8670	0.8067
S_h	0.8199	0.8028

value of the similarity metric for these images is computed and is shown in Table 1.

Using the same constraints as in cases 1, 2, and 3, we implemented the algorithm of Ref. 2. Strictly speaking, the algorithm of Ref. 2 is initialized with random numbers and, as shown[2] such initialization can yield good results for 32×32 pixel-square images. For the 64×64 pixel-square images that we used, the best results obtained from the algorithm of Ref. 2 were produced by the nonrandom initialization described below. First, the same constraints as in case 1 were used. With the same initial starting point as in Eq. (18), the resulting f and h are shown in Figs. 5(a) and 5(b). The similarity metric for these images was computed and is given in Table 2. As can be seen, the results do not accurately reflect the true source and blurring functions.

To test the sensitivity of the algorithm of Ref. 2 to brightness scaling, we scaled by a constant the same initial condition used above. More specifically, the algorithm of Ref. 2 was tested with initial conditions of the form

$$f_0(\mathbf{x}) = \begin{cases} \alpha g(\mathbf{x}) & \text{if } \mathbf{x} \in W_f \\ 0 & \text{otherwise} \end{cases}, \tag{19}$$

where α is some positive constant and $h_0 = 0$. Three values of $\alpha = 0.1$, 0.025, and 0.01 were used. The resulting f and h images are shown in Figs. 6(a) and 6(b), 6(c) and 6(d), and 6(e) and 6(f), respectively. The similarity metric of these images is given in Table 3. From the images in Figs. 5 and 6 and the similarity values in Tables 2 and 3, it is clear that the algorithm of Ref. 2 is sensitive to scaling of the initial conditions.

In contrast, the projection-based algorithm is insensitive to brightness scaling of the initial conditions because the functional $J_{(f_k,h_k)}$ is scale invariant, and we project onto constraint sets at every step of the optimization process. The insensitivity to scale was observed in our computer simulations.

It is observed that for the value $\alpha = 0.025$ the algorithm of Ref. 2 gives the best results. Therefore, in the rest of the numerical experiments presented in this paper, only this serendipitous value of α will be used. We also tested the algorithm with this value of α, using support constraints as in cases 2 and 3. The resulting images are shown in Figs. 7(a) and 7(b), and 7(c) and 7(d), respectively. The similarity metric for these images is given in Table 4.

From both the visual point of view and the viewpoint of the similarity criteria, it is clear that the conjugate-gradient method is inferior to the projection-based algorithm.

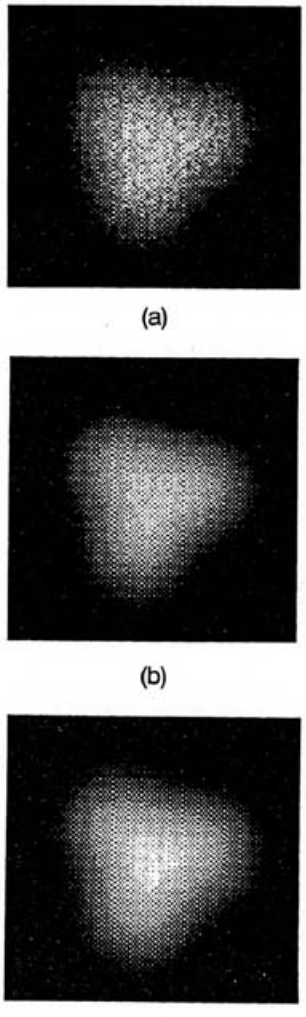


Fig. 8. Photon-limited data: (a) 10^5 counts, (b) 10^6 counts, (c) 5×10^6 counts.

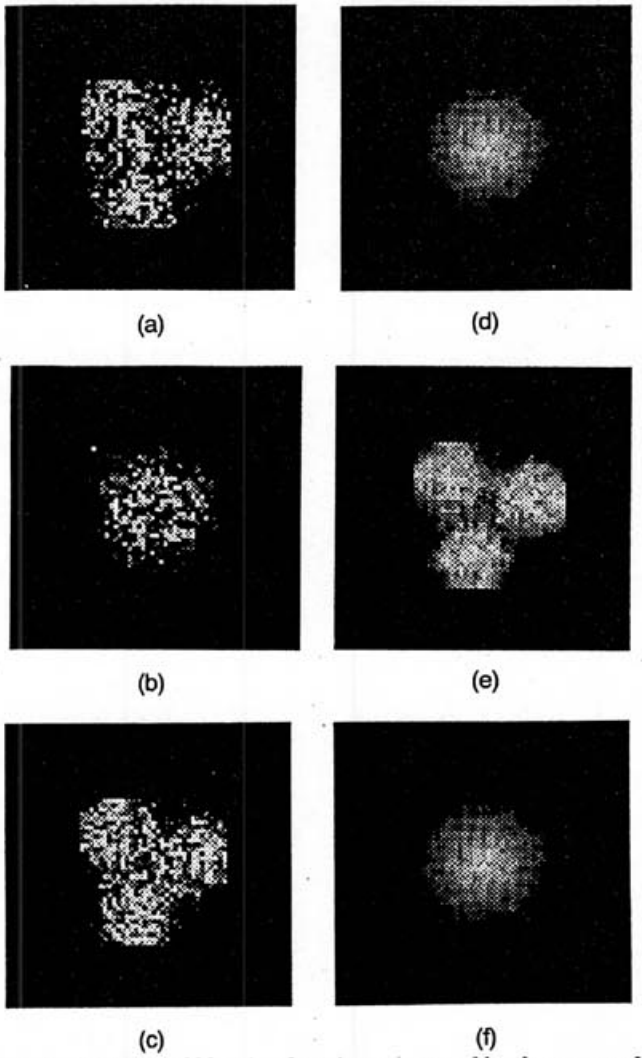


Fig. 9. Images and blurring functions recovered by the proposed algorithm from three levels of photon-limited data: (a) and (b) 10^5 counts, (c) and (d) 10^6 counts, (e) and (f) 5×10^6 counts.

Table 5. Similarity Metric of the Recovered Source and Blurring Functions of Fig. 9

Fig. 9	10^5	10^6	5×10^6
S_f	0.6970	0.7856	0.9149
S_h	0.6894	0.8995	0.9780

Example 2. We also tested the proposed algorithm with photon-limited simulated data to test the sensitivity of the proposed approach to noise. The same object and blurring function as in example 1 were used. However, in order to isolate the effect of the noise, we used the constraints only of case 1. Three levels of noise were examined. For the first, the total number of photons in the observed data was assumed to be 10^5, for the second 10^6, and for the third 5×10^6. The observed data for noise levels 1, 2, and 3 are shown in Figs. 8(a), 8(b), and 8(c), respectively. The images recovered with use of the projection-based algorithm from data with photon levels 1, 2, and 3 are shown in Figs. 9(a) and 9(b), 9(c) and 9(d), and 9(e) and 9(f), respectively. The similarity of these images is given in Table 5.

Lane's algorithm[2] was tested on these photon-limited data, with use of the same parameters and $\alpha = 0.025$. The deconvolved source and the blurring functions are shown in Figs. 10(a) and 10(b) for 10^5 counts, Figs. 10(c) and 10(d) for 10^6 counts, and Figs. 10(e) and 10(f) for 5×10^6 counts. The similarity results of these images are given in Table 6. Here for the first time we see a marginal improvement in similarity furnished by the conjugate-gradient method versus the projection-based approach at low counts. Visually, however, one could argue that the projection-based results are still superior, if only slightly. It is important to note that the similarity metric, under high-noise conditions, is not quite so useful as a predictor of visual quality, because the noise can overwhelm the shape information.

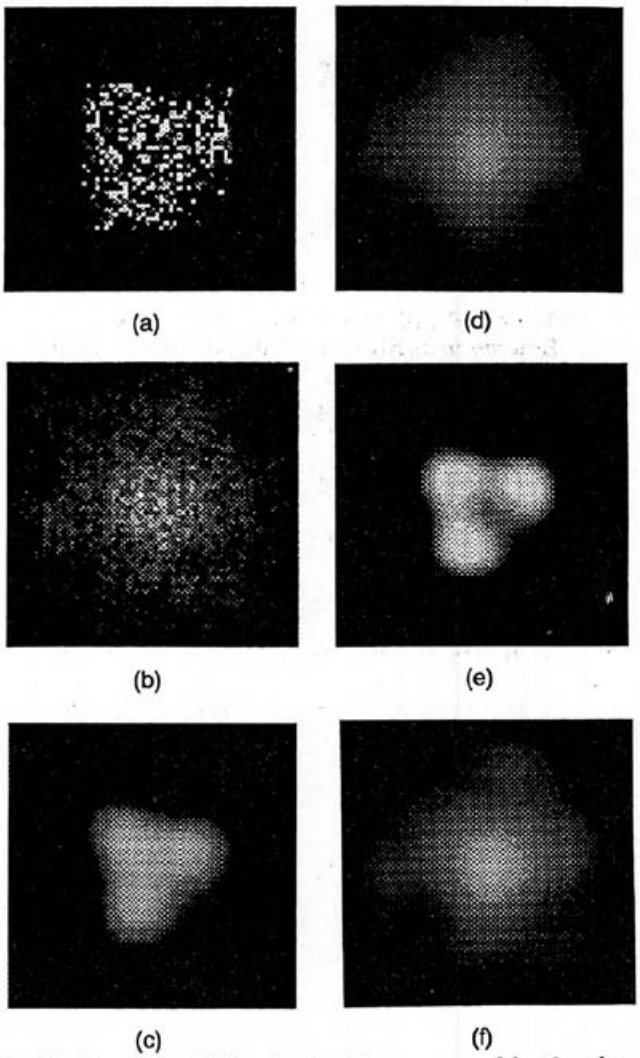


Fig. 10. Images and blurring functions recovered by the algorithm of Ref. 2 from three levels of photon-limited data: (a) and (b) 10^5 counts, (c) and (d) 10^6 counts, (e) and (f) 5×10^6 counts.

Table 6. Similarity Metric of the Recovered Source and Blurring Functions of Fig. 10

Fig. 10	10^5	10^6	5×10^6
S_f	0.5529	0.8433	0.8457
S_h	0.7561	0.7919	0.8244

6. CONCLUSIONS

In this paper we proposed a projection-based blind-deconvolution algorithm that avoids the numerical difficulties associated with implementation of a generalized-projection-type algorithm. Moreover, the proposed algorithm has good convergence properties and is flexible in permitting incorporation of different types of prior knowledge for both the source signal and the blurring function. Our numerical simulations demonstrated that the projection-based deconvolution algorithm gave good estimates of the source and the blurring functions without requiring perfect knowledge of the support information. Indeed, the algorithm demonstrated that, even when support parameters were overestimated or underestimated, the deconvolved results were superior to the unprocessed image. Also, the algorithm has the property that the final solution is invariant to the scaling of its components.

Finally, we compared our algorithm with a conjugate-gradient procedure published recently.[2] For the example considered, the projection-based procedure gave superior reconstructions.

ACKNOWLEDGMENT

Y. Yang acknowledges the generous support of the Peng Chun memorial fellowship. The authors appreciate the helpful suggestions of the two reviewers who carefully read this paper.

REFERENCES

1. G. Ayers and J. C. Dainty, "Iterative blind deconvolution method and its applications," Opt. Lett. **13,** 547–549 (1988).
2. R. G. Lane, "Blind deconvolution of speckle images," J. Opt. Soc. Am. A **9,** 1508–1514 (1992).
3. B. C. McCallum, "Blind deconvolution by simulated annealing," Opt. Commun. **75,** 101–105 (1990).
4. A. K. Katsaggelos, R. L. Lagendijk, and J. Biemond, "Constrained iterative identification and restoration of images," in *Proceedings of the European Signal Processing Conference, EURASIP 1988* (Elsevier, Amsterdam, 1988), pp. 1585–1588.
5. J. R. Fienup, "Phase retrieval algorithms: a comparison," Appl. Opt. **21,** 2758–2769 (1982).
6. A. Levi and H. Stark, "Image restoration by the method of generalized projections with applications to restoration from magnitude," J. Opt. Soc. Am. A **1,** 932–943 (1984).
7. H. Peng and H. Stark, "Signal recovery with similarity constraints," J. Opt. Soc. Am. A **6,** (1989).
8. S. L. S. Jacoby, J. S. Kowalik, and J. T. Pizzo, *Iterative Methods for Nonlinear Optimization Problems* (Prentice-Hall, Englewood Cliffs, N.J., 1972).
9. R. L. Lagendijk, J. Biemond, and D. E. Boekee, "Regularized iterative restoration with ringing reduction," IEEE Trans. Acoust. Speech Signal Process. **36,** 1874–1888 (1988).
10. B. R. Hunt, "The application of constrained least squares estimation to image restoration by digital computer," IEEE Trans. Comput. **C-22,** 805–812 (1973).
11. A. K. Katsaggelos and S. N. Efstratiadis, "A class of iterative signal restoration algorithms," IEEE Trans. Acoustics Speech Signal Process. **38,** 778–786 (1990).