Binary image restoration by positive semidefinite programming

Yijiang Shen, Edmund Y. Lam, and Ngai Wong

Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong

Received September 1, 2006; revised October 13, 2006; accepted October 13, 2006;
posted October 19, 2006 (Doc. ID 74508); published December 23, 2006

We report an optimization approach to restore degraded binary images by using positive semidefinite programming when the point spread function (PSF) is known. The approach takes advantage of the combinatorial nature of the problem, considering not only local similarity and spatial context but also the relationship between individual pixel values and the PSF. Numerical experiments confirm the superiority of the approach. © 2006 Optical Society of America

OCIS codes: 100.0100, 100.3020.

The goal of image restoration is to reconstruct the original scene from a degraded observation. A special case is where the true scenery is binary, applicable to various situations such as fingerprint recognition, automated document handling, and the like. Numerous attempts have been made to deal with binary image restoration. Hitchcock and Glasbey[1] tried to restore images of bloblike and filamentous objects, and Neifeld *et al.*[2] included prior knowledge concerning local correlations among pixel values into the Viterbi-based restoration process. Other methods, such as the weighted mean square error method of Meloche and Zamar,[3] the pulse coupled neural network of Gu *et al.*,[4] and the convergent method minimizing the total-variation functional of Chan *et al.*[5] are designed to deal with only noisy images, whereas the more common case with blurring is ignored. Generally speaking, image formation can be modeled as

$$g(x_1,x_2) = f(x_1,x_2) * h(x_1,x_2) + n(x_1,x_2), \tag{1}$$

where $f(x_1,x_2)$ and $g(x_1,x_2)$ represent the true and the degraded images, respectively, $h(x_1,x_2)$ is a linear shift-invariant blur known as the point spread function (PSF), $n(x_1,x_2)$ is the additive noise, and $*$ denotes the two-dimensional convolution operator. The inversion of blur is numerically unstable and can be made tractable only by including some assumptions about the true scene. For example, linear deconvolution methods such as the Wiener filter and one-step least squares[6] discard the high-frequency component of the images, and in maximum entropy restoration[7] the constraint is that pixels cannot take negative values. Further investigation can be found in Ref. 8. These generic restoration methods do not take advantage of the constraint that pixel values are binary when restoring degraded binary images.

The combinatorial nature of binary image denoising was noted in Ref. 9: for each pixel position i of an image, the pixel value g_i originates from either of two known prototype values which, without loss of generality, can be represented by -1 and $+1$, and g_i can be scaled accordingly. To restore the discrete-valued image function represented by $x \in \{-1,+1\}^n$, which contains the columns of the true binary image f stacked upon one another, we minimize

$$z(x) = \sum_i (x_i - g_i)^2 + \frac{\lambda}{2}\sum_{\langle i,j\rangle} (x_i - x_j)^2, \tag{2}$$

in which λ is the smoothness parameter and the second term sums over all pairwise adjacent variables in vertical and horizontal directions on the regular image grid. Equation (2) comprises a data fitting term and a smoothness term modeling spatial context, to be jointly minimized.

In this Letter we further explore the optimization process and report the application of positive semidefinite programming when the PSF is of arbitrary size. The approach takes into account local similarity, spatial context, and the relationship of individual pixel values and the PSF; consequently both image deblurring and denoising are simultaneously carried out in the restoration process. Some preliminary results of restoring small-size binary images blurred by PSFs limited to size 3×3 can be found in Ref. 10.

Suppose the degraded image g and the true image f is of size $u\times v$ and the PSF h is of size $m\times n$. For a specific pixel x_i in $x \in \{-1,+1\}^n$, the blurred pixel value x_i^b would be the linear combination of x_i and its neighboring pixel values within the range of the size of the PSF and the entry values of the PSF:

$$x_i^b = \sum_{j=1}^{m}\sum_{k=1}^{n} x_{i+(j-\tilde{m})+(k-\tilde{n})u} h_r(j,k), \tag{3}$$

where $h_r(i,j) = h(m+1-i, n+1-j)$ $\tilde{m} = \lfloor m+1/2 \rfloor$, and $\tilde{n} = \lfloor n+1/2 \rfloor$; its neighboring pixels beyond the image border will take the value 0. Thus, by replacing x_i in the first term of Eq. (2) with x_i^b in Eq. (3), we have

$$z(x) = \sum_i (x_i^b - g_i)^2 + \frac{\lambda}{2}\sum_{\langle i,j\rangle} (x_i - x_j)^2. \tag{4}$$

The numerical results of Eq. (4) will find its minimum when the restored $x \in \{-1,+1\}^n$ is used; thus the optimization process that minimizes Eq. (4) will simultaneously deconvolve and denoise g by assigning every individual x_i as -1 or $+1$, which justifies the omission of noise in Eq. (3). With g_i and $x_i^2 = 1$ being constants, the computation of Eq. (4) leads to

0146-9592/07/020121-3/$15.00

$$-2g_i\sum_i x_i^b+\sum_i (x_i^b)^2-\lambda\sum_{\langle i,j\rangle} x_i x_j. \tag{5}$$

We define $g_{(\hat{h},\hat{v})}$ as shifting g by $\hat{h}$ rows and $\hat{v}$ columns, in which $\hat{h}>0$ means shifting $\hat{h}$ rows down, otherwise $|\hat{h}|$ up; and $\hat{v}>0$ means shifting $\hat{v}$ columns right, otherwise $|\hat{v}|$ left. If we define b_i of size $uv\times 1$ as

$$b_i=-4\sum_{j=1}^{m}\sum_{k=1}^{n} g_{(j-\tilde{m},k-\tilde{n})i}h_r(j,k), \tag{6}$$

the first term in Eq. (5) becomes $\frac{1}{2}b^Tx$.

Let $s=i+(j-\tilde{m})+(k-\tilde{n})u$, $t=i+(\hat{j}-\tilde{m})+(\hat{k}-\tilde{n})u$, wherein $j=\hat{j}$ and $k=\hat{k}$ do not happen at the same time; an arbitrary summation term $\Sigma x_s h_r(j,k)x_t h_r(\hat{j},\hat{k})$, in $\Sigma(x_i^b)^2$ over all pixels, differs from $\Sigma x_i x_{i+(j-\hat{j})+(k-\hat{k})u}h_r(j,k)h_r(\hat{j},\hat{k})$ in the summation area. We construct Q of size $uv\times uv$ such that, after the summation area of an individual summation term is defined, matrix entries $Q(i,i+(j-\hat{j})+(k-\hat{k})u)$ will be added with $4h_r(j,k)h_r(\hat{j},\hat{k})$. Similarly, the third term of Eq. (5) can also be represented in Q, viz., matrix entries $Q(i,j)$ will be added with -2λ for adjacent pixels i,j. Thus the second term and the third term can be represented in the form of $\frac{1}{4}x^TQx$, and Eq. (4) leads to the optimization problem

$$\inf_x\left(\frac{1}{4}x^TQx+\frac{1}{2}b^Tx\right),\qquad x\in\{-1,+1\}^n. \tag{7}$$

The objective function of Eq. (7) can be further homogenized in the following way:

$$x^TQx+2b^Tx=\begin{pmatrix}x\\1\end{pmatrix}^T L\begin{pmatrix}x\\1\end{pmatrix},\qquad L=\begin{bmatrix}Q & b\\ b^T & 0\end{bmatrix}. \tag{8}$$

Denoting the Lagrangian multiplier variables with y_i, $i=1,\ldots,n$, Eq. (8) can be relaxed to

$$z_d=\sup_y e^Ty,\qquad L-D(y)\in S_+^n, \tag{9}$$

where $D(y)$ denotes the diagonal matrix with diagonal values y_i, e is a vector of all entries being 1, and S_+^n is the positive semidefinite cone, which represents a convex optimization problem. Choosing Lagrangian multiplier $X\in S_+^n$, the dual problem z_p to z_d is

$$z_p=\inf_{x\in S_+^n} L\cdot X,\qquad D(X)=I, \tag{10}$$

with $\cdot$ being the standard matrix scalar product, $D(X)$ being matrix X with off-diagonal elements set to zero, and I being the unit matrix, which again is convex. Details of relaxation in Eq. (8) can be found in Ref. 9. The convex optimization solver SeDuMi[11] is used to compute the optimal solution X^* of Eqs. (10). Then the combinatorial solution x of the original problem (8) can be found by using the randomized-hyperplane technique proposed by Goemans and Williamson.[12]

The computation time quickly grows with the number of variables in the problem, such that the restoration of binary images with sizes around 100×100 is impractical. Alternatively, we segment the images to smaller blocks such as 10×10 before applying the approach to each block. An obvious problem is that when the pixels outside the boundary rows or columns of the block are neglected, error inevitably increases because the computation of x_i^b requires its adjacent pixels to be within the range of the PSF matrix as indicated in Eq. (4). Consequently, a trade-off has to be made between the computation time and the quality of the restored images. To this end, we adopt a scheme with overlapping rows and columns among image blocks. For example, an image block of size 16×16 is restored, but only the pixels in the center 10×10 block will be preserved.

Figure 1(a) presents the original binary image of size 123×91. Figures 1(b) and Fig. 1(c) are the images blurred by h_3 without and with noise, respectively, in which h_3 is the 3×3 averaging operator. Figure 1(d) and 1(e) are the images blurred by h_5 without and with noise respectively, where h_5 is a randomly generated 5×5 matrix and its sum is normalized to 1. In both cases, the added noise is Gaussian white noise of mean 0 and variance 0.01. Figures 2(a)–2(h) show the restoration of Figs. 1(b) and 1(c) without overlapping and with overlapping of 1–3 rows and 1–3 columns, respectively. Figures 3(a)–3(l) show the restoration results of Fig. 1(d) and Fig. 1(e) without overlapping and with overlapping of 1–5 rows and 1–5 columns, respectively. We also present in Figs. 4(a)–4(d) the restoration results of Figs. 1(b)–1(e) when a Wiener filter is used with the object

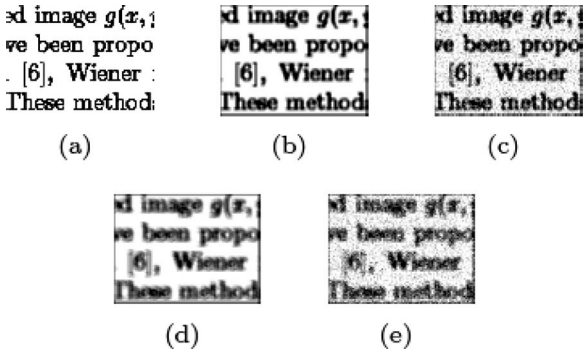

Fig. 1. (a) True image. Image (b) blurred by h_3, (c) blurred by h_3 and with noise, (d) blurred by h_5, (e) blurred by h_5 and with noise.

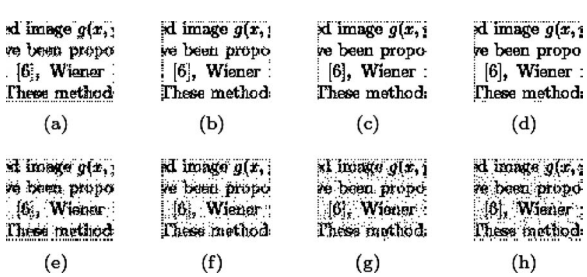

Fig. 2. Results of restoration (a)–(d) of Fig. 1(b), (e)–(h) of Fig. 1(c) without overlapping, with overlapping of 1–3 rows and 1–3 columns.

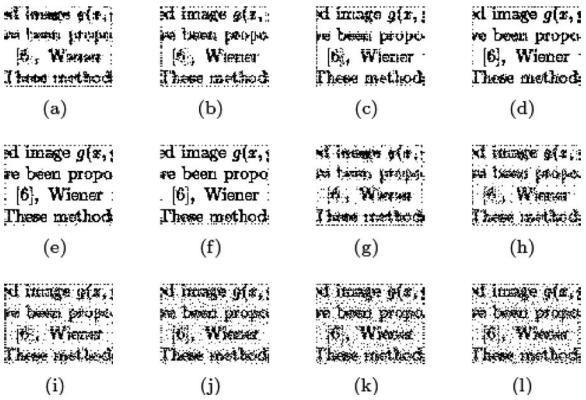

Fig. 3. Results of restoration (a)–(f) of Fig. 1(d), (g)–(l) of Fig. 1(e) without overlapping, with overlapping of 1–5 rows and 1–5 columns.

Fig. 4. Results of restoration of Figs. 1(b)–1(e) by using the Wiener filter.

Table 1. Accuracy of Restored Images in Fig. 2

Fig. Part	Accuracy Rate (%)	Fig. Part	Accuracy Rate (%)
(a)	94.10	(e)	91.61
(b)	96.70	(f)	92.93
(c)	98.21	(g)	93.25
(d)	99.03	(h)	93.30

Table 2. Accuracy of Restored Images in Fig. 2

Fig. Part	Accuracy Rate (%)	Fig. Part	Accuracy Rate (%)
(a)	88.12	(g)	86.30
(b)	91.16	(h)	87.49
(c)	92.76	(i)	87.80
(d)	95.78	(j)	88.08
(e)	96.52	(k)	87.37
(f)	97.69	(l)	88.03

correlation matrix as the object prior, as an example of the generic restoration methods.

From Tables 1 and 2, where the accuracy rate indicates the percentage of correctly restored pixels out of the entire image, it can be seen that when the overlapping of the image block increases, the accuracy of the restored images increases, and the deconvolution has a better improvement than the denoising. This happens because in the restoration process, some pixels with small values in the blurred image will be deconvolved by the PSF to have much bigger values, and thus are recognized as +1 instead of the true value −1. Compared with the results in Table 3 for the Wiener filter, the optimization approach provides a much better accuracy. The results also show that, although the restorations have high accuracy, the quality of the restored images is not satisfactory if the accuracy is below 90%. In summary, the advantages of the proposed optimization method are threefold: it is accurate, noise resistive, and simple without introducing new parameters. It should also be noted that the computation time grows exponentially with the number of variables, which means that the price paid for the outstanding performance of the proposed approach is the computational complexity.

Table 3. Accuracy of Restored Images in Fig. 4

Fig. Part	Accuracy Rate (%)	Fig. Part	Accuracy Rate (%)
(a)	74.87	(c)	64.97
(b)	63.63	(d)	55.76

The work described in this paper was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (project HKU 7164/03E). Y. Shen's e-mail address is yjshen@eee.hku.hk.

References

1. D. Hitchcock and C. A. Glasbey, Biometrics **53**, 1010 (1997).
2. M. Neifeld, R. Xuan, and M. Marcellin, Appl. Opt. **39**, 269 (2000).
3. J. Meloche and R. H. Zamar, Can. J. Stat. **22**, 335 (1994).
4. X. D. Gu, H. M. Wang, and D. H. Yu, in *8th International Conference on Neural Information Processing* (Fudan University Press, 2001), pp. 922–927.
5. T. F. Chan, S. Esedoglu, and M. Nikolova, in IEEE International Conference on Image Processing (IEEE, 2005), pp. 121–124.
6. R. C. Gonzalez and R. E. Woods, *Digital Image Processing* (Prentice-Hall, 2002).
7. J. Myrheim and H. Rue, Comput. Vis. Graph. Image Process. **54**, 223 (1992).
8. M. R. Banham and A. K. Katsaggelos, IEEE Signal Process. Mag. **14**(2), 24 (1997).
9. J. Keuchel, C. Schellewald, D. Cremers, and C. Schnörr, IEEE Trans. Pattern Anal. Mach. Intell. **25**, 1364 (2003).
10. Y. Shen, E. Y. Lam, and N. Wong, *2006 IAENG International Workshop on Imaging Engineering (IWIE'06)* (International Association of Engineers, 2006), pp. 537–542.
11. J. F. Sturm, Optim. Methods Software **11**, 625 (1999).
12. M. X. Goemans and D. P. Williamson, J. ACM **42**, 1115 (1995).