Blind Deconvolution of Bi-level Images with Successive Filtering

Edmund Y. Lam

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

ABSTRACT

Many imaging systems are involved in the capture of bi-level objects such as documents and bar codes. However, due to optical aberrations and other degradations such as from motion, the resulting images are no longer bi-level. Restoration is therefore necessary to produce a nearly bi-level output, which is much more convenient for further processing including recognition and identification. In this work, we tackle the problem using successive filtering, where each step is formulated as a quadratic programming optimization problem. This has the properties of fast convergence and good numerical stability. Simulation results show that by integrating the computation in the imaging system, this method is able to restore weak signals.

Keywords: Image restoration, blind deconvolution, bi-level images, iteration, resolution enhancement

1. INTRODUCTION

Bi-level, or two-tone, images are ubiquitous in many imaging systems. Usually, the two levels correspond to black and white, one of which is the image of interest and the other is the background. Computer vision algorithms are frequently employed to assist in the automatic recognition and identification of these images. Methods to treat bi-level images, such as morphological image processing, are well-developed and are often very efficient for real-time processing of such image data.[1]

In practice, however, the cameras often cause blurring to the system, and the resulting images are no longer bi-level. The degradation can be due to aberrations in the optics and motions or vibrations between the object and the camera. Furthermore, the exact format of these degradations is often only partially known or totally unknown to the system. In any case, the images captured are no longer bi-level, and the computer vision algorithms cannot take advantage of the simplicity in the processing of bi-level versus grayscale images.

The imaging process can be represented by

$$i(x,y) = g(x,y) * h(x,y) + n(x,y), \tag{1}$$

where $g(x,y)$ is the binary object, $h(x,y)$ is the point spread function, $n(x,y)$ is the noise, and $i(x,y)$ is the captured image. For a diffraction-limited system under incoherent light, $h(x,y)$ is seen to be a lowpass filter.[2] The observed image is therefore a noisy, blurred version of the object, with a lower resolution due to the filter. Digital post-processing is possible to restore the resolution content. In a classical problem formulation for image restoration, $h(x,y)$ is assumed to be known, and analytical solutions exist that are optimal in certain senses.[3] However, this assumption may not be realistic in many practical situations. The restoration problem without knowledge of the point spread function is called blind deconvolution,[4] and is the goal of the present paper.

Further author information: (Send correspondence to Edmund Lam.)
Edmund Lam: E-mail: elam@eee.hku.hk

ICO20: Optical Information Processing, edited by Yunlong Sheng, Songlin Zhuang, Yimo Zhang,
Proc. of SPIE Vol. 6027, 60274S, (2006) · 0277-786X/06/$15 · doi: 10.1117/12.668623

2. NEW ALGORITHM

Blind deconvolution is in general a difficult problem because of the amount of data to be estimated. In our setting, we want to make use of the prior information that $g(x, y) \in (0, 1)$ at every location. We employ the technique of quadratic programming, which is a very attractive technique because of its many properties, such as global solution and worst-case computation time. In contrast with other methods such as genetic algorithms or neural network, it allows for more insight in the nature of the solution.

2.1. Algorithm

As with many image restoration method, we restore the observed image by filtering it with a finite impulse response filter $w(x, y)$ to obtain $\hat{g}(x, y) = w(x, y) * i(x, y)$, where $\hat{g}(x, y)$ should ideally be a binary signal. We scale $i(x, y)$ so that after the restoration, the two values that $\hat{g}(x, y)$ would take are ± 1. Therefore, $[\hat{g}(x, y)]^2 - 1 \approx 0$ for all (x, y).

We set up the optimization problem as follows:

$$\begin{array}{ll} \text{minimize} & \|t(x,y)\|^2 \\ \text{subject to} & -t(x,y) \preceq [w(x,y) * i(x,y)] \bullet [w(x,y) * i(x,y)] - 1 \preceq t(x,y), \end{array} \tag{2}$$

where the symbol $\preceq$ denotes element-by-element comparison and $\bullet$ denotes element-by-element multiplication. When $w(x, y)$ produces nearly binary outputs at ± 1, $t(x, y)$ will be small at that pixel and the objective function will be minimized.

Equation (2), however, is not convex.[5] Generally speaking, non-convex optimization is difficult and often computationally intensive. Here, we opt to tackle the problem by stages:

1. Let $k = 0$.
2. Begin with an estimate of $w^{(k)}(x, y)$, where the superscript denotes the iteration number. Because $h(x, y)$ is a lowpass filter, $w^{(k)}(x, y)$ will be a highpass filter. We can then compute $\hat{g}^{(k)}(x, y) = w^{(k)}(x, y) * i(x, y)$.
3. We solve the optimization problem

$$\begin{array}{ll} \text{minimize} & \|t(x,y)\|^2 \\ \text{subject to} & -t(x,y) \preceq [\hat{g}^{(k)}(x,y)] \bullet [w(x,y) * i(x,y)] - 1 \preceq t(x,y). \end{array} \tag{3}$$

   If we use $\mathbf{w}$ as the raster-scan of $w(x, y)$ and $\mathbf{t}$ as the raster-scan of $t(x, y)$, the above problem is a quadratic programming formulation with respect to the variable $\phi = [\mathbf{w};\ \mathbf{t}]^T$. A similar construction has shown to be effective for blind equalization of constant modulus signals in communication systems.[6] Since quadratic programming is convex, it enjoys a lot of desirable properties, such as fast implementations with the interior-point algorithm.[5]

4. We extract the corresponding part of the solution ϕ above as $w_{\text{opt}}(x, y)$. We combine the result with the previous estimate of $w^{(k)}(x, y)$ to form a new estimate

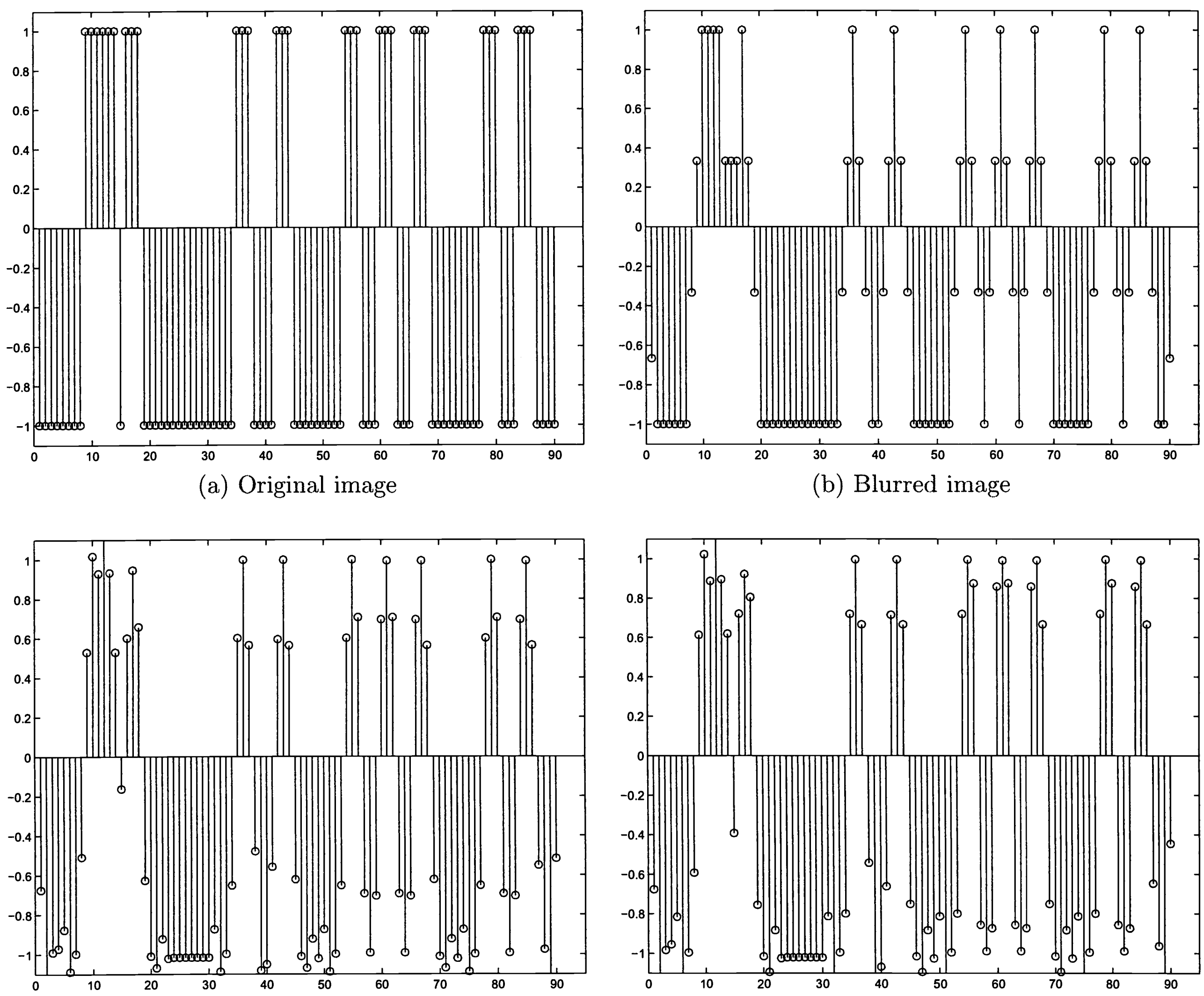
$$w^{(k+1)}(x,y) = \lambda w^{(k)}(x,y) + (1-\lambda) w_{\text{opt}}(x,y), \tag{4}$$

   where λ is set according to how dependable the previous iteration is deemed to be.

5. We use this new $w^{(k+1)}(x, y)$ and go back to step (2) with $k := k + 1$.

2.2. Simulation

We first show a one-dimension example. Figure 1(a) is an ideal binary image with values adjusted to ± 1. It is passed through a simple averaging filter with length 3 to arrive at (b). If we simply threshold the output with positive values set to $+1$ and negative values set to -1, we will lose the singular points such as in position 15. These singular points are important as they can represent, for example, thin lines or dots important for an optical character recognition (OCR) system. Instead, we subject the blurred image to our blind deconvolution scheme outlined in the previous section. The result after 5 iterations is shown in (c), while the result after 10 iterations is shown in (d). We can observe that in both cases, with the restoration in place a simple thresholding will set the singular point at 15 to the correct value, and further iterations increase the margin between the threshold value and the restored value at the singular point. This is very desirable to withstand the effect of noise. In addition, Figure 2 shows the normalized value of the optimization functional with respect to the number of iterations. We can see that the functional is decreasing, indicating that the restored values are closer to ± 1.



(a) Original image

(b) Blurred image

(c) After 5 iterations

(d) After 10 iterations

Figure 1. Blind deconvolution of binary (1D) images with the proposed scheme.

Next we also show the results of applying the algorithm on an image. The original text image is shown in

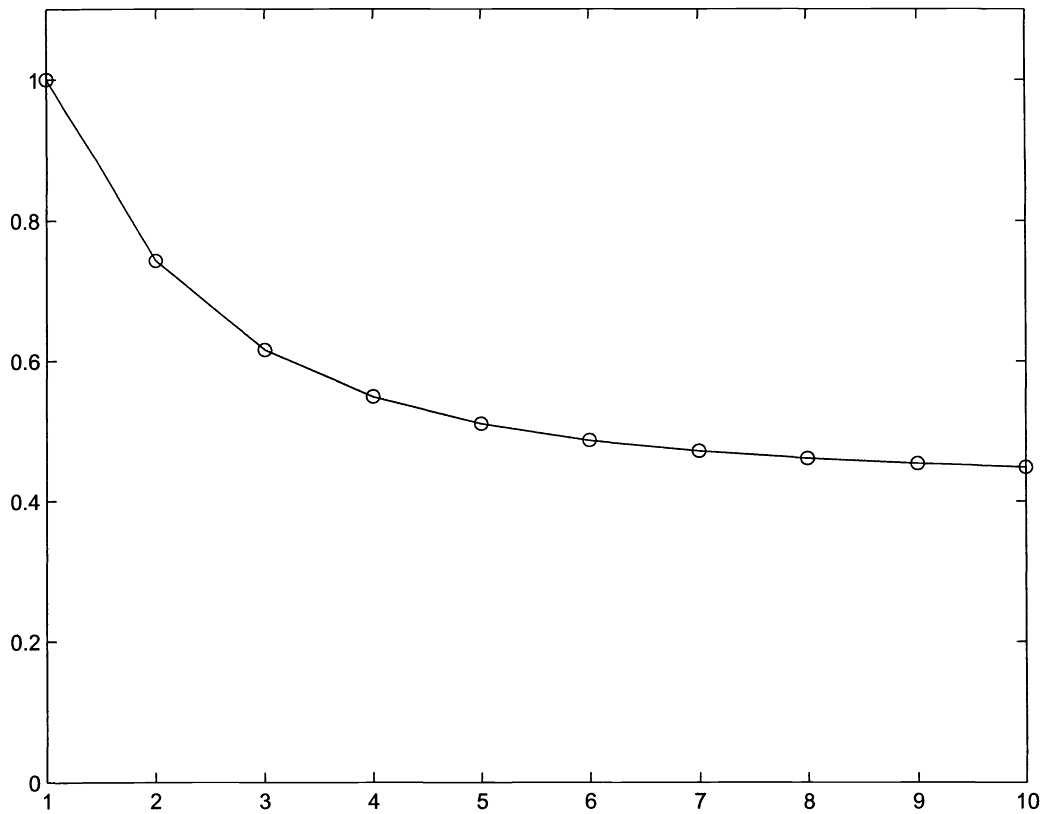


Figure 2. Value of the optimization functional with respect to the iterations.

Figure 3(a). It is then blurred and added with some Gaussian noise to give the image in (b). Even though we can still read the letters, if we subject the image to a simple thresholding operation we will obtain (c), which is very difficult for OCR. However, if we apply the blind deconvolution algorithm to the image in (b), after 10 iterations we obtain the restored image as shown in (d). Note that we used a very rough highpass filter as the initial estimate in accordance with step 2 in Section 2.1, with only a mild resemblance to the Wiener filter solution if $h(x, y)$ were known beforehand. The restored image evidently has increased high frequency information, which after subjecting to a thresholding would return a rather readable set of characters.

3. SUMMARY AND CONCLUSION

In this paper we have briefly described our formulation of an iterative quadratic programming to achieve blind deconvolution for binary images. The prior knowledge of binary source is embedded in the optimization formulation. Preliminary results show that this method is capable of restoring blurred images for machine vision tasks such as character recognition. Further research will focus on the initial estimation of the restoration function and speed of convergence.

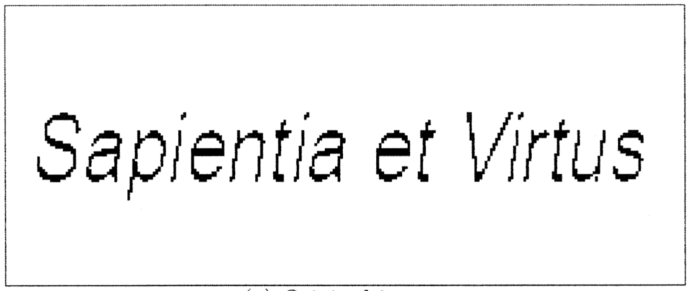
ACKNOWLEDGMENTS

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 Number HKU 7164/03E).

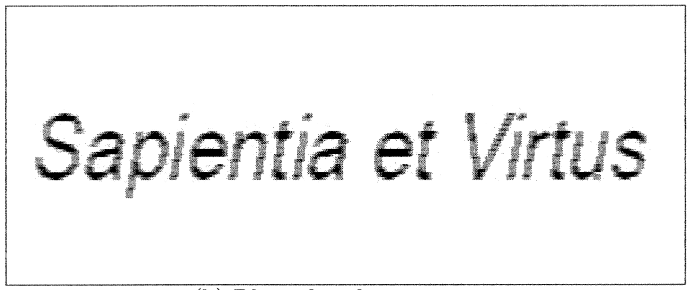
REFERENCES

1. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice Hall, Englewood Cliffs, New Jersey, second ed., 2002.

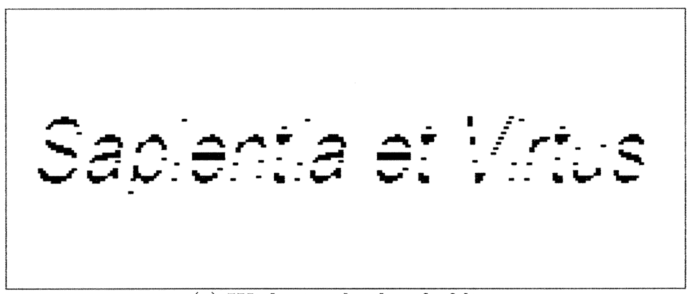
2. J. W. Goodman, *Introduction to Fourier Optics*, McGraw-Hill, New York, second ed., 1996.
3. H. Andrews and B. Hunt, *Digital Image Restoration*, Prentice Hall, Englewood Cliffs, New Jersey, 1977.
4. E. Y. Lam and J. W. Goodman, "An iterative statistical approach to blind image deconvolution," *Journal of the Optical Society of America A* **17**, pp. 1177–1184, July 2000.
5. S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, 2004.
6. I. Santamaría, C. Pantaleón, L. Vielva, and J. Ibáñez, "Blind equalization of constant modulus signals using support vector machines," *IEEE Transactions on Signal Processing* **52**, pp. 1773–1782, June 2004.



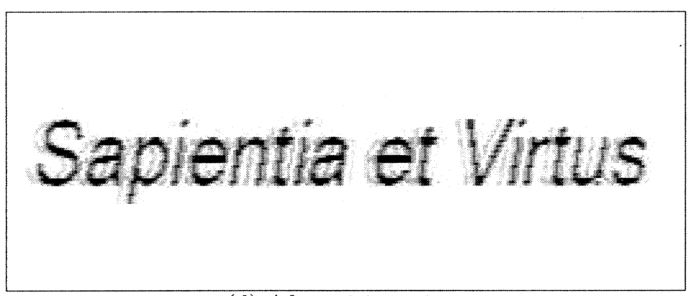
(a) Original image.



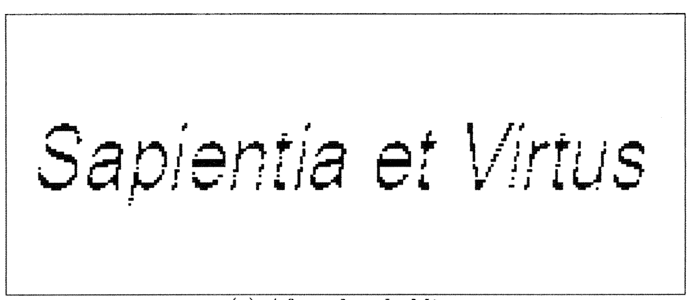
(b) Blurred and noisy image



(c) With simple thresholding



(d) After 10 iterations



(e) After thresholding

Figure 3. Blind deconvolution of binary (2D) images with the proposed scheme.