

# A Robust Nonlinear Filtering Approach to Inverse Halftoning

Mei-Yin Shen and C.-C. Jay Kuo

*Integrated Multimedia Systems Center and the Department of Electrical Engineering-Systems,  
University of Southern California, Los Angeles, California 90089-2564*

E-mail: [meiyinsh@sipi.usc.edu](mailto:meiyinsh@sipi.usc.edu), [cckuo@sipi.usc.edu](mailto:cckuo@sipi.usc.edu)

Received August 4, 1999; accepted August 10, 2000

---

A new blind inverse halftoning algorithm based on a nonlinear filtering technique of low computational complexity and low memory requirement is proposed in this research. It is called blind since we do not require the knowledge of the halftone kernel. The proposed scheme performs nonlinear filtering in conjunction with edge enhancement to improve the quality of an inverse halftoned image. Distinct features of the proposed approach include efficiently smoothing halftone patterns in large homogeneous areas, additional edge enhancement capability to recover the edge quality, and an excellent PSNR performance with only local integer operations and a small memory buffer. © 2001 Academic Press

*Key Words:* digital halftoning; inverse halftoning; nonlinear filtering; error diffusion.

---

## 1. INTRODUCTION

Halftone images are binary images that provide a rendition of gray-level images. Inverse halftoning addresses the problem of recovering a continuous-tone image from a halftone image. Inverse halftoning has applications in halftone manipulation, conversion, and compression when a halftone image is the only available version of an image. In this research, a nonlinear filtering technique is proposed to solve the inverse halftoning problem.

Although the proposed framework can work in principle with any halftoning process, we focus on the most popular halftoning technique known as error diffusion in the current work. In error diffused halftones, the error between the continuous-tone image and the binary halftone at each pixel is diffused over a causal neighborhood. Most of the quantization noise power falls in the high spatial frequency range. A good inverse halftoning scheme should remove as much noise as possible while preserving the important image details. Several inverse halftoning methods have been studied in the literature, including linear adaptive filtering [7], nonlinear filtering [6, 11], wavelets [12], MAP estimation [8], and POCS [1, 3]. A typical inverse halftoning process consists of two key modules, halftone noise smoothing and image detail recovery, as detailed below.

Smoothing can be performed by using linear filtering techniques that include the halfband lowpass filtering [11], the Gaussian lowpass filtering [7], and the singular value decomposition (SVD) technique [3]. It can also be achieved with model-based nonlinear smoothing techniques such as MAP estimation [8] and the nonlinear filtering approach proposed in this work. Some work adopts edge information to improve smoothing results, for example, spatial varying FIR filtering [6] and wavelet denoising [12]. All the above methods can reduce halftone noise efficiently. However, the cutoff frequency of the lowpass filter or the threshold used in wavelet and model-based techniques plays a critical role in the overall performance of a given algorithm. Furthermore, halftone noise contributes its energy mostly to high frequency components, which can potentially corrupt or reduce the quality of image edges. This is the fundamental limitation of inverse halftoning algorithms relying on the smoothing process only. An edge recovery procedure is therefore required to obtain a high quality inverse halftoned image.

To recover edges, a bandpass filter with thresholding was used to extract and enhance edges in [7], and the application of a Gaussian filter to a highpass frequency band was studied in [12]. There is no explicit edge extraction step in the restoration-based approach [1, 8], where the edge information is recovered by projection to the original halftone image with the assumption that the halftone kernel is known.

In this work, we present a single-pass blind inverse halftoning algorithm with a low computational complexity and a low memory buffer. It is called blind since we do not assume that the halftone kernel is available. Our scheme performs nonlinear filtering in conjunction with edge enhancement to improve the quality of an inverse halftoned image. The proposed approach is universal in the sense that no prior knowledge about the forward halftoning process is assumed.

This paper is organized as follows. Two challenging problems for an inverse halftoning technique are discussed in Section 2. The proposed inverse halftoning algorithm is detailed in Section 3. Experimental results are given in Section 4. Finally, concluding remarks are stated in Section 5.

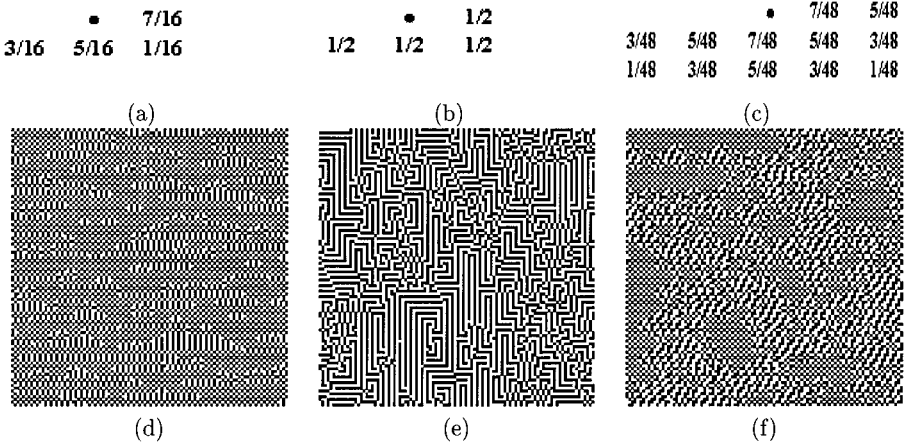
## 2. TWO PROBLEMS

Previous work on blind inverse halftoning was performed without knowing the diffusion kernel. It is nevertheless assumed that a “good” halftoning scheme has been applied so that the resulting halftone image will not have localized energy in smooth areas and edges are well represented. This is, however, not true in general. In this section, we address two major problems that have not yet been pointed out in previous research in blind inverse halftoning.

### 2.1. Halftone Artifacts

One of the most observable artifacts associated with error diffusion is the periodically repeating pattern appearing in areas where the image is a constant or slow varying. This is called the limit cycle behavior, which was studied in [2]. The limited cycle phenomenon results from a constant input, especially when the input value is a constant that slightly deviated from  $\frac{1}{2}$ ,  $\frac{1}{3}$ , and  $\frac{1}{4}$ . The stability of these periodic patterns is determined by the choice of error diffusion weights and the support of the error diffusion kernel.

An example of limited cycle patterns is shown in Fig. 1. In this example, the input is set to a constant  $\frac{130}{256}$ . It results in low frequency noise and makes a small perturbation to



**FIG. 1.** Examples of the limited cycle behavior with a constant input set to 130: (a)-(c) are error diffusion kernels used in (d)-(f), respectively, (d) is dominated by checkerboard patterns, (e) is dominated by worm patterns, and (f) has no obvious dominating patterns.

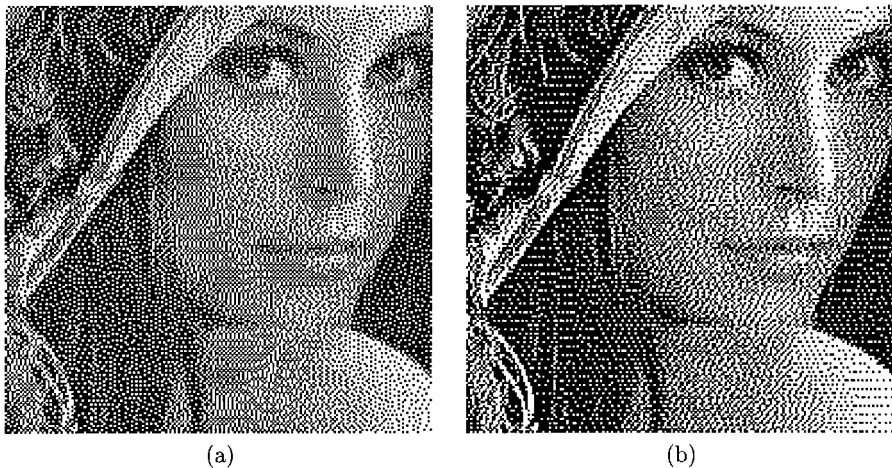
the checkerboard pattern. (Note that most error diffusion kernels generate a checkerboard pattern at mid-gray.) In Fig. 1d the Floyd–Steinberg kernel is used. The dominating pattern is the checkerboard. The worm pattern appearing in Fig. 1e is generated by equally weighted 4-coefficients in the diffusion kernel. Figure 1f is the result of the Jarvis kernel [4]. The Jarvis kernel has a larger support than the other two kernels. Increasing the support generally can reduce limited cycle patterns. The limited cycle behavior introduces low frequency noise or structure noise (e.g., worm patterns), which is difficult to remove by using a linear lowpass filter or even the wavelet denoising technique. A good blind inverse halftoning algorithm should be able to deal with the limited cycle behavior.

## 2.2. Edge Sharpness

The sharpness of an edge is determined by halftoning schemes and halftone kernels. In general, an error-diffused image has more edge information than one using clustered-dot or disperse-dot methods. The error diffusion kernels also effect the quality of edges. Larger diffusion kernels tend to have sharper edges. An example to illustrate the influence of the error diffusion kernel is shown in Fig. 2. Clearly, the 12-coefficient Jarvis kernel [4] produces sharper edges than the 4-coefficient Floyd–Steinberg kernel at the cost of more noise in the slow varying regions. If significant edge information is lost during the forward halftoning process, the reconstructed inverse halftone will have rather smooth edges. In blind inverse halftoning, we cannot assume the edge is always perfectly rendered. The loss of edge resolution cannot be solved by edge preserving smoothing algorithms. Additional techniques, such as edge enhancement, must be performed to recover sharp edges.

## 3. PROPOSED INVERSE HALFTONING ALGORITHM

The proposed inverse halftoning algorithm consists of two components, i.e., noise smoothing and edge enhancement, to overcome the problems addressed in the previous section. A block diagram of the proposed algorithm is shown in Fig. 3. The lowpass filter serves as a preprocessor to provide a rough estimate of the continuous-tone image, and it is further



**FIG. 2.** Examples of the edge quality influenced by the diffusion kernel: (a) the Floyd–Steinberg kernel and (b) the Jarvis kernel.

refined by a nonlinear filter that can efficiently suppress low frequency noise resulting from limited cycle patterns in smooth regions. The loss of edge resolution is compensated by an edge enhancement technique using a bandpass filter and a morphological filter.

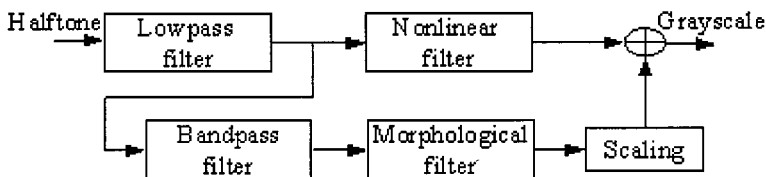
### 3.1. Design of Smoothing Filters

In [9, 10], a robust nonlinear filtering technique was adopted to reduce various types of compression artifacts. The nonlinear filter can suppress noise in continuous-tone images efficiently while preserving the edge information. The same technique can also be used in inverse halftoning to reduce halftone noise. However, the robust nonlinear filter in [9, 10] is a *location filter*. That is, the output is selected from the neighboring pixel values. For inverse halftoning, we have to preprocess the halftone image with a lowpass filter and then feed the output into the robust nonlinear filter for further refinement.

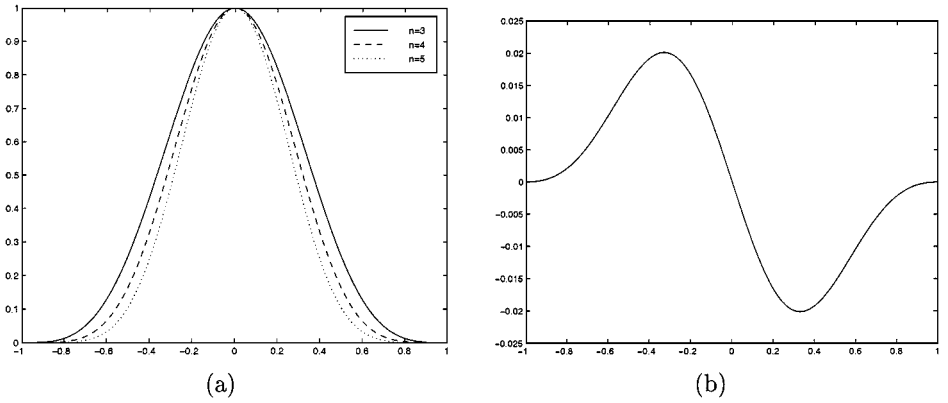
The lowpass filter plays a critical role for the final performance. To lower the computational cost, a 2D separable FIR lowpass filter is selected. The 1D version of the proposed lowpass filter is designed based on a spline function  $\theta$ ,

$$\theta(x, n) = \left( \frac{\sin \pi x}{\pi x} \right)^n, \quad n = 1, 2, \dots, N. \quad (1)$$

For a given filter length, coefficients of the filter can be determined by uniformly sampling the above function between  $[-1, 1]$ , and the cutoff frequency can be adjusted by the value



**FIG. 3.** The block diagram of the proposed halftoning algorithm.



**FIG. 4.** (a) A family of spline functions used in the lowpass filter design, and (b) the derivative of a cubic spline.

of  $n$ . A larger value of  $n$  leads to a higher cutoff frequency and, consequently, a lower smoothing capability. Coefficients obtained by the above function should be normalized to achieve the unity gain at the DC component. The shape of this function with different  $n$  is depicted in Fig. 4a. Experiments show that the above lowpass filter is superior to the Gaussian filter with respect to error diffusion halftones. The output of this filter is a rough estimate of the original continuous-tone image and will be sent to a nonlinear filter and a bandpass filter for further processing.

The nonlinear filter adopted is the same as the deringing filter presented in [9, 10]. That is, given a set of samples  $x_1, x_2, \dots, x_N$  in the filtering window and the potential function  $\rho$ , the filter output is defined as

$$\hat{x} = \arg \min_{x_j} \sum_{i=1}^N \rho(x_i - x_j). \quad (2)$$

To ensure data fidelity, the following clipping process is applied,

$$d = \hat{x} - x, \quad (3)$$

$$x_c = x + c(d, Th1), \quad (4)$$

where  $x$  is the value of the pixel to be filtered,  $\hat{x}$  is the output of (2),  $x_c$  is the final estimate to replace  $x$  and  $c(d, Th1)$  is a clipping function to ensure data fidelity and is defined as

$$c(d, Th1) = \text{sign}(d) \cdot \max(0, |d| - \max(0, 2(|d| - Th1))).$$

The clipping function controls the amount of distortion introduced by the smoothing operation. The shape of  $c(d, Th1)$  is depicted in Fig. 5. It is clear that the result of nonlinear filtering is effective only when  $d$  is smaller than  $2Th1$ . The value  $Th1$  determines the amount of smoothing.

To reduce halftone noise, Huber and truncated quadratic potential functions are better choices for the potential function, since they are efficient in smoothing and can be implemented by using only integer operations. For images containing more homogeneous regions, the truncated quadratic function works the best. A set of commonly used nonlinear filters is given in Table I.

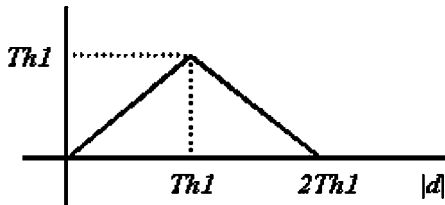


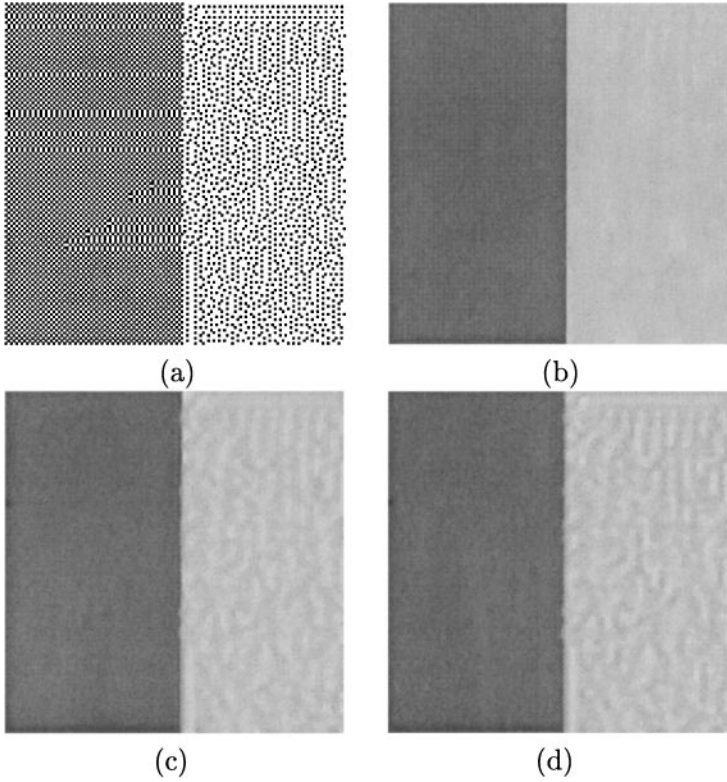
FIG. 5. The clipping function.

There are three parameters in the nonlinear filter. One is the filter size. Since the diffusion kernel has a smaller extent, a  $3 \times 3$  window size is usually sufficient for most images. If the underlying image contains a large amount of smooth regions, a larger window size can be used to reduce clustered noise. The second parameter is the threshold  $T$  used in potential functions. For the truncated quadratic potential function,  $T$  can be chosen to be between 5 and 20. A larger value of  $T$  tends to produce a smoother result. For the Huber function,  $T$  ranges from 1 to 10. For  $T = 1$ , this nonlinear filter is equivalent to the median filter. For better quality,  $T$  can vary adaptively based on edge information. The third parameter is the threshold for clipping, which is the value of  $Th1$  in (3). By empirically testing the average distortion of inverse halftoned results for a selected set of images, the value of  $Th1$  can be chosen to be one half or one quarter of the average value. The purpose of the clipping operation is to avoid additional distortion.

The cascade of lowpass and nonlinear filters can efficiently smooth the low-frequency clustered halftone noise in large smooth regions. Examples to demonstrate this phenomenon are shown in Figs. 6a–6d. The original image has two constant gray levels ( $\frac{127}{256}$  and  $\frac{200}{256}$ ). The resulting halftone is shown in Fig. 6a. The result of the proposed algorithm without edge enhancement is shown in Fig. 6b. In this case, a quadratic function with  $T = 10$  is used and the window size is  $9 \times 9$ . Results obtained by other blind inverse halftoning algorithms [6, 7] are shown in Figs. 6c and 6d for comparison. Clearly, the proposed algorithm is more efficient in reducing noise in large smooth regions. Note that the proposed technique can preserve the quality of sharp edges without further enhancement for this case. This is due to the fact that the original image has a crispy edge. Since there is no intermediate level, the resolution of the edge can be easily preserved. In addition, this image contains two constant

TABLE I  
Commonly Used Potential Functions

Type	$\rho(x)$
$L^2$	$x^2$
$L^1$	$ x $
“Fair”	$\gamma x  - \gamma^2 \log(1 + \frac{ x }{\gamma})$
Huber	$\rho(x) = \begin{cases} x^2, &  x  \leq \gamma \\ \gamma^2 + 2\gamma( x  - \gamma), &  x  > \gamma \end{cases}$
$L^\gamma$	$ x ^\gamma, 1 \leq \gamma \leq 2$
Truncated $L^2$	$\min\{\gamma x^2, 1\}$
Lorentzian	$\log(1 + \frac{1}{2}(\frac{x}{\gamma})^2)$ .
Tukey	$\rho(x) = \begin{cases} \frac{\gamma^2}{6}(1 - [1 - (\frac{x}{\gamma})^2]^3), &  x  \leq \gamma \\ \frac{\gamma^2}{6}, &  x  > \gamma \end{cases}$



**FIG. 6.** Examples of smoothing a halftone image: (a) the halftone of an image with two constant gray levels, 130/256 and 200/256; (b) the result by applying the proposed smoothing filters; (c) the result by applying the adaptive filtering technique [6]; (d) the result by applying the nonlinear smoothing technique [7].

gray levels separated by the sharp transition. It fits the truncated quadratic model the best so that the sharp edge can be well recovered by the proposed nonlinear filter with only a minor distortion.

### 3.2. Edge Enhancement

In the halftoning process, the gray-scale resolution (i.e., intensity) is represented by the density of randomly distributed binary patterns. In some edge regions, such as those around an unsharp edge, the intensity is changing gradually within small extent. For this case, intermediate intensities at the edge transition area cannot be well represented by halftone patterns. The resulting edge after inverse halftoning is smoother than the original one due to the loss of grayscale resolution. This loss of resolution cannot be recovered even with the proposed nonlinear filter. Note that our nonlinear filters use the first-order model for discontinuity that does not fit smoother edges. The use of a higher order model for edge approximation may solve this problem at the price of a higher complexity.

The edge enhancement approach presented in this section is based on heuristic arguments. It, however, provides a good solution to the loss of grayscale resolution problem at a low computational cost. With this approach, useful edge information is obtained by applying a bandpass filter to a lowpass-filtered halftoning image. There are two ways to realize the bandpass filter. One is to apply two lowpass filters with different cutoff frequencies

and calculate the difference between the two filtered results. The other method is to take derivative of (1) and sample it with a desired filter length. The derivative of (1) is depicted in Fig. 4b.

In this work, we adopt the first approach, since a filter of shorter length can be used. After bandpass filtering, a morphological operation is applied to refine the bandpass filtered result. The morphological filter includes two steps. First, thresholding is applied to the output of the bandpass filter to generate a binary edge map  $E$ . Then a binary median filter is applied to  $E$  to remove noise in this map. This edge information is used to refine the original bandpass filter output by setting values of nonedge pixels to zero. Finally, the refined result  $B$  is added back to the nonlinear filter output  $S$  to obtain the inverse halftoning image  $I$ . In terms of mathematics, we have

$$I(m, n) = S(m, n) + \lambda B(m, n), \quad 1 \leq m, n \leq N, \quad (5)$$

where  $\lambda$  is a scaling factor to control the amount of edge enhancement. This enhancement technique has a very low complexity, since only windows of a small size and integer operations are used.

## 4. EXPERIMENTAL RESULTS

We performed experiments on the  $512 \times 512$  *Lena* and *Peppers* images, as well as the  $512 \times 640$  *Woman*, *Cafe*, and *Bike* images. These images were error diffused from their corresponding graylevel images using the Floyd–Steinberg kernel. We compare our results with several previous blind inverse halftoning work as follows.

### 4.1. Comparison of Lowpass Filters

The lowpass filter that performs an initial estimate of its corresponding continuous-tone image plays a critical role in the inverse halftoned result. Equation (1) provides a family of lowpass filters. The cutoff frequency of the lowpass filter should be high enough to preserve spatial details while providing a sufficient grayscale resolution for the robust nonlinear filter. To achieve a good trade-off between the performance and the complexity, a  $7 \times 7$  separable FIR filter with coefficients listed in Table II is used in the implementation. These coefficients are obtained by setting  $n = 3$  in (1) and normalized to have the unity gain. We compare the performance of this filter with other filters used in other inverse halftoning algorithms, including the  $9 \times 9$  Gaussian filter [3], the halfband lowpass filter [11], and the wavelet method [12]. The PSNR values for several test images after lowpass filtering are shown in Table III. We see that the proposed lowpass filter is superior to others.

### 4.2. Blind Inverse Halftoning

Our technique does not assume the availability of the error diffusion kernel. The PSNR results of our proposed technique and other blind inverse halftoning algorithms, including

**TABLE II**  
**Coefficients of the 7-tape Lowpass Filter Used in the Experiment**

$n$	-3	-2	-1	0	1	2	3
$H$	0.0089	0.0852	0.2409	0.3300	0.2409	0.0852	0.0089



**TABLE III**  
**Performance Comparison of Different Lowpass Filters**

Algorithm	Gaussian [3]	Half-band [11]	Wavelet [12]	Proposed
PSNR (dB)				
Lena	28.64	29.60	30.38	30.79
Peppers	27.59	27.76	28.56	28.65

statistical smoothing [11], wavelet [12], adaptive filtering [6], and nonlinear noise removal [7], are listed in Table IV. The PSNR figures are either taken directly from the publications [11, 12] or generated from the software provided by the authors' Web sites [6, 7]. Nonblind inverse halftoning algorithms assume that the halftone kernel is known and perform the projection to improve the quality. Results of these techniques are also provided in Table V for comparison. We can see clearly from Table IV that our technique achieves the best results among all blind inverse halftoning algorithms under comparison. Our method is also robust with respect to all test images. It is interesting to see that our blind inverse halftoning results are comparable to (or even outperform) those obtained from iterative projection. It is actually not surprising that the kernel information does not boost the inverse halftoning performance much via iterative projection since the halftone process is a many-to-one map; the original halftone image does not provide a strict constraint in the space domain. This is especially obvious when intensities are around mid-gray; projection to this set may move the estimate not toward the real solution, but toward a legal solution.

If the purpose of inverse halftoning is to obtain the original continuous-tone image, then subjective quality of the inverse halftone is also important. In Fig. 7, the central portion of the *Lena* and *Peppers* images is enlarged and compared to that obtained from a wavelet-based blind inverse halftoning method [12], which tends to produce the best visual quality among previous work. It is clear that visual quality achieved by the proposed method is comparable to the best scheme. In general, a larger window size produces more pleasant visual quality at the cost of a higher computational complexity.

### 4.3. Complexity and Memory

The complexity of the proposed technique is slightly higher than that of the linear filtering approach [6, 7], but much less than that of the wavelet-based method [12].

To give an example, if we use a  $7 \times 7$  window for all three lowpass filters,  $5 \times 5$  for the binary median filter, and  $3 \times 3$  for the nonlinear filter, the number of operations required

**TABLE IV**  
**Performance Comparison of Different Blind Inverse Halftoning Algorithms**

Algorithm Ref.	PSNR				
	Lena	Peppers	Woman	Bike	Cafe
Statistical [11]	31.00	29.30	NA	NA	NA
Wavelet [12]	31.50	30.43	NA	NA	NA
Adaptive [6]	31.33	29.03	29.56	22.64	22.03
Nonlinear [7]	31.26	28.96	29.21	22.26	21.39
Proposed	31.62	30.82	29.60	22.97	22.41

*Note.* NA, Not available. The data are not provided by the publication.

**TABLE V**  
**Performance Comparison of Inverse**  
**Halfoning Algorithms That Require**  
**Knowledge of the Halftone Kernel**

Algorithm [Ref.]	PSNR	
	Lena	Peppers
Statistical [11]	32.00	30.30
Wavelet [12]	31.67	30.69

per pixel is less than 500. The execution of the proposed algorithm proceeds in the raster scan fashion. Several rows are required for the filtering process. In this example, at least 11 rows have to be buffered.

The complexity and memory requirements for several inverse halfoning schemes are compared in Table VI. The memory usage is the minimum number of data required for buffering to process one row of an image. The computational complexity is estimated



(a) Proposed



(b) Wavelet



(c) Proposed



(d) Wavelet

**FIG. 7.** Inverse halfoning results of the proposed technique (a and c) and results from the wavelet approach in [12] (b and d).

**TABLE VI**  
**Performance Comparison of Inverse Halftoning Schemes in Terms of Complexity and Memory Requirements**

Algorithm [Ref.]	Memory usage	Complexity
POCS [3]	$8N^2$	High
MAP [8]	$8N^2$	High
Wavelet [12]	$9N^2$	Med.
Adaptive [6]	$7N$	Low
Nonlinear [7]	$11N$	Low
Proposed	$11N$	Low

from algorithms given in the corresponding citation. *Low* means a number fewer than 500 operations per pixels, *median* denotes 500–2000 operations per pixel, and *high* means more than 2000 operations required. Clearly, scan-based techniques such as our algorithm have a memory requirement that is proportional to image width  $N$ , while frame-based methods have to store the entire frame and their complexity is proportional to  $N^2$  for an image of size  $N \times N$ . Furthermore, most frame-based techniques are iterative so that we have to buffer several copies of the image. The wavelet approach requires the largest amount of memory buffer due to the use of overcomplete wavelet decomposition, and all bands created in each decomposition have to be buffered.

#### 4.4. Other Diffusion Kernels

We also test our algorithm on halftone images generated by using the Jarvis kernel [4] and the multiscale halftoning algorithm [5]. Results from the proposed method and two other blind inverse halftoning algorithms are compared in Table VII. Throughout these tests, a  $3 \times 3$  nonlinear filter with the Huber potential function is used in our algorithm. Threshold  $T$  in the potential function is set to 2 and constraint  $Th1$  is set to 15. From Table VII, it is clear that the proposed algorithm outperforms the two other methods for all test images.

As discussed previously, the Jarvis kernel produces sharper edges at the price of more noise in homogeneous regions. We note that the Jarvis kernel produces higher contrast edges. Lowpass filtering of these edges generates an enhanced version of the original ones. In our approach, the Huber function is used as a cost function. It allows a smoother transition of large discontinuities, leading to an edge that is closer to the original one. It also produces better results in homogeneous areas since it is a quadratic cost function when the

**TABLE VII**  
**Performance of Blind Inverse Halftoning Algorithms Applied to Halftone Images Generated by Different Error Diffusion Algorithms**

Halftone algorithm	PSNR		
	Proposed	Adaptive [6]	Nonlinear [7]
Jarvis			
Lena	25.76	24.65	25.29
Peppers	25.42	23.91	24.35
Multiscale			
Lena	28.29	26.73	28.16
Peppers	29.07	27.99	28.71

residual is below threshold  $T$ . Multiscale error diffusion has a higher gain at edges than Floyd–Steinberg diffusion. As a result, edges in the inverse halftone images from Jarvis and multiscale error diffusion methods are overly sharp.

We can draw the following conclusions based on experimental results presented above.

- The proposed method is more robust w.r.t. different error diffusion kernels. Some inverse halftoning methods with predesigned passbands may fail when halftoning algorithms perform visual enhancement such as the use of the Jarvis kernel.
- Compared to most other methods that are difficult to optimize for different halftone images, our approach can simply replace the cost function in the nonlinear filter to obtain better results. This is one important advantage of the model-based filtering technique.

## 5. CONCLUSION

A blind inverse halftoning algorithm of low complexity was presented in this work. The cascade of lowpass and nonlinear filters in conjunction with the edge enhancement technique produce high quality inverse halftoned images at a low computational cost. Results of the proposed technique are comparable to those obtained by sophisticated restoration-based techniques, which often require information about the diffusion kernel. Besides, our method is more robust to different error diffusion kernels than other blind inverse halftoning algorithms, since our nonlinear filtering is a model-based approach where the smoothing operation is based on a generic image model for an ideal image. The advantage of using our technique is even more substantial when an image contains constant or slow varying gray levels in large areas separated by sharp edges.

## REFERENCES

1. G. Bozkurt and A. E. Çetin, Set theoretic inverse halftoning method for grayscale and color images, in *SPIE Proc. on Visual Communications and Image Processing, 1999*, Vol. 3653, pp. 151–159.
2. Z. Fan and R. Eschbach, Limited cycle behavior of error diffusion, in *Proc. IEEE International Conference on Image Processing, 1994*, Vol. 2, pp. 1041–1045.
3. S. Hein and A. Zakhor, Halftone to continuous-tone conversion of error-diffusion coded images, *IEEE Trans. Image Process.* **4**, 1995, 208–215.
4. J. Jarvis, C. Judice, and W. Ninke, A survey of techniques for displaying of continuous-tone pictures on bilevel displays, *Comput. Graphics Image Process.* **5**, 1976, 13–40.
5. I. Katsavounidis and C. C. J. Kuo, A multiscale error diffusion technique for digital halftoning, *IEEE Trans. Image Process.* **6**, 1997, 483–490.
6. T. D. Kite, N. D.-Venkata, B. L. Evans, and A. C. Bovik, A high quality, fast inverse halftoning algorithm for error diffusion halftones, in *Proc. IEEE International Conference on Image Processing, 1998*, Vol. 2, pp. 59–63.
7. N. D. Venkata, T. D. Kite, M. Venkataraman, and B. L. Evans, Fast blind inverse halftoning, in *Proc. IEEE International Conference on Image Processing, 1998*, Vol. 2, pp. 64–68.
8. R. L. Stevenson, Inverse halftoning via MAP estimation, *IEEE Trans. Image Process.* **6**, 1997, 574–583.
9. M. Y. Shen and C.-C. J. Kuo, Artifact Reduction in low bit rate wavelet coding with robust nonlinear filtering, in *Proc. of IEEE MMSP'98, Los Angeles, CA, Dec. 1998*, pp. 480–485.
10. Mei-Yin Shen and C.-C. J. Kuo, Fast compression artifact reduction technique based on nonlinear filtering, *Proc. of IEEE ISCAS'99, 1999*, Vol. 4, pp. 179–182.
11. P. W. Wong, Inverse halftoning and kernel estimation for error diffusion, *IEEE Trans. Image Process.* **4**, 1995, 486–498.
12. Z. Xiong, M. T. Orchard, and K. Ramchandran, Inverse halftoning using wavelets, in *Proc. IEEE International Conference on Image Processing, 1996*, pp. 569–572.