

# Journal of Electronic Imaging

[SPIDigitalLibrary.org/jei](http://SPIDigitalLibrary.org/jei)

## **Pre- and postprocessing for multilayer compression of scanned documents**

Alexandre Zaghetto  
Ricardo L. de Queiroz



# Pre- and postprocessing for multilayer compression of scanned documents

Alexandre Zaghetto

Ricardo L. de Queiroz

Universidade de Brasilia

Department of Computer Science

ICC Centro, Caixa postal 4466

Brasilia, 70910-900 Brazil

E-mail: alexandre@cic.unb.br

---

**Abstract.** The mixed raster content (MRC) document-compression standard (ITU T.44) specifies a multilayer representation of a document image. The model is very efficient for representing sharp text and graphics over a background. However, its binary selection layer compromises the representation of scanned data and soft edges. Typical segmentation algorithms that split up the document into layers tend to lift letter colors to the foreground, so that soft edge transitions may not fully belong either to the foreground or background layers, causing “halos” around objects that impair compression performance. We present a method that sharpens the document before compression and softens its edges after MRC-based reconstruction. It builds an edge-sharpening map and estimates the original edge softness at the encoder. The generated map and softness parameters are then used to reconstruct the original soft edges at the decoder. An MRC encoding and decoding scheme based on H.264/AVC and JBIG2 has been used. Experimental results show that, for lower bit rates, the proposed pre-/postprocessing method can improve both subjective and objective compression performance over regular MRC. © 2011 SPIE and IS&T. [DOI: 10.1117/1.3644554]

---

## 1 Introduction

Digital documents are basically represented in two forms: vectorial or raster.<sup>1</sup> It is not much of a challenge to compress vectorized documents because each object can be compressed individually and the whole file can be compressed losslessly. The real challenge is to compress rasterized documents.

The mixed raster content (MRC) imaging model<sup>1–7</sup> has been proposed as a multilayer representation of a document. The basic three-layer MRC model represents a color image as two color image layers [foreground (FG) and background (BG)] and a binary image layer [mask (M)]. The mask layer describes how to reconstruct the final image from the FG/BG layers (i.e., to use the corresponding pixel from the FG or BG layers when the mask pixel is 0 or 1, respectively, in that position). An illustration of the imaging model is shown in Fig. 1. Because the original single-plane image is represented using multiple layers, each layer can be processed and compressed using different algorithms. Foreground and background processing operations may include resolution change

and data-filling procedures. The compression algorithm used for a given layer would be matched to the layer’s content,<sup>4,5</sup> allowing for improved compression while reducing distortion visibility. MRC has been proposed and/or accepted for several standards,<sup>6–9</sup> as well as used in several products.<sup>10–13</sup>

The MRC model is very efficient for representing sharp text and graphics over a background. However, because the mask layer is binary, it is difficult to deal with scanned data and soft edges. The main contribution of this paper is to propose pre- and postprocessing techniques that modify MRC to deal with soft scanned document edges. We also suggest the use of H.264/AVC (Advanced Video Coding),<sup>14–20</sup> operating in INTRA mode to encode foreground and background layers, and JBIG2 (Ref. 21) to encode the binary mask. JBIG2 is an international standard for lossy and lossless compression of bilevel images developed by the Joint Bi-level Image Experts Group (JBIG). It has been shown that JBIG2 may outperform JBIG1 by at least a factor of 2.<sup>22</sup> Regarding H.264/AVC, it is a video-compression standard and was not originally conceived to be applied as a still-image compression tool. Nevertheless, the many coding advances brought into H.264/AVC not only set a new benchmark for video compression, but they also make it a formidable compressor for still images.<sup>23–26</sup> If we set our H.264/AVC implementation to work on a sole INTRA frame, then it behaves as a still-image compressor. We refer to this coder as AVC-I (or AVC-INTRA). In many cases, AVC-I outperforms previous state-of-art coders, such as JPEG2000.<sup>27–29</sup> AVC outperformed JPEG-2000 in all tests in this paper, so that results for the latter are omitted. Because layer decomposition is not the main focus of this paper, we used a block-thresholding segmentation algorithm in our tests for convenience.<sup>3</sup>

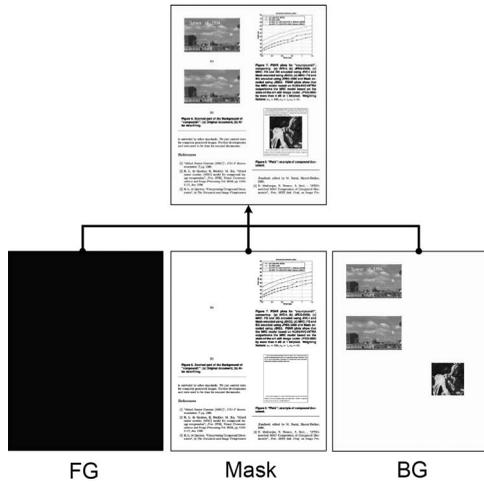
## 2 Dealing with Soft Edges in Mixed Raster Content

In a typical MRC segmentation, the imaged text has about the same shape and size as that in the mask plane. The reason for that is to increase compression in many scenarios and to provide prompt readability. When the image is scanned, the edges of text and graphics are not as sharp and we refer to them as “soft edges.” Because the selector plane is binary and the edge transitions are smooth, it is not possible to contain all the foreground material in the FG plane and to represent

---

Paper 11068R received Apr. 17, 2011; revised manuscript received Aug. 24, 2011; accepted for publication Sep. 8, 2011; published online Nov. 8, 2011.

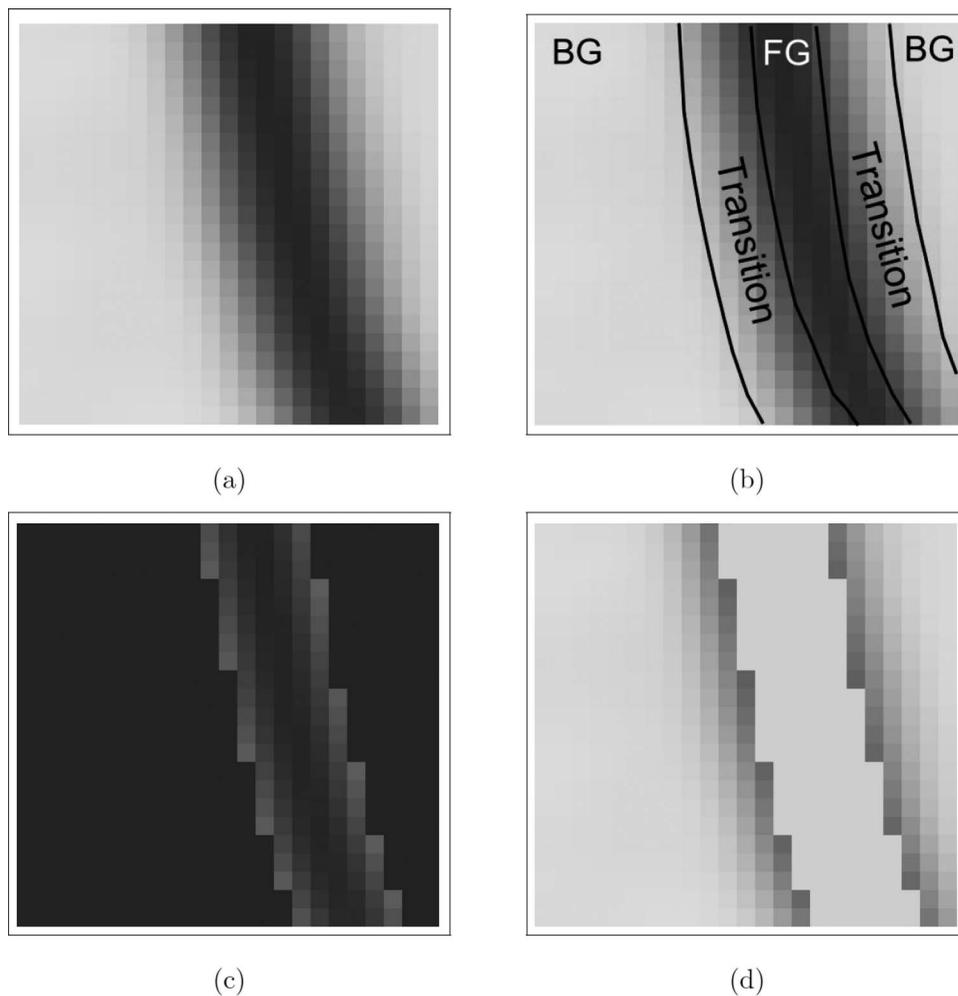
1017-9909/2011/20(4)/043005/10/\$25.00 © 2011 SPIE and IS&T



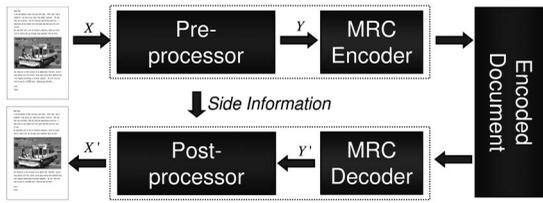
**Fig. 1** Illustration of the MRC imaging model. The basic three-layer MRC model represents a color image as two color image layers (FG and BG) and a binary image layer (M).

all the background in the BG plane. An illustration is shown in Fig. 2. In Fig. 2(a), there is a zoom of a small portion of text. One would typically classify the FG and BG regions as in Fig. 2(b), wherein, however, care is taken to identify transition regions where none of the FG/BG features are evident. If we segment by midgray threshold (light regions are BG and dark ones are FG), the borderline between FG and BG would lie somewhere in the transition region. Hence, each region would inevitably contain pixels that are not as light or dark as the BG/FG layer would require. A typical data-filling technique, to increase compression,<sup>1,30,31</sup> is to remove the BG data in the FG layer and to replenish it with the average of the FG. We do the opposite for the BG layer. Out of 256 gray values, if we use a 128 (midgray) threshold and replenishment values of 30 and 220, we obtain the FG/BG MRC layers shown in Figs. 2(c) and 2(d). Note the intrusive contours caused by spurious transition-region pixels, which would definitely harm compression performance.

The effect in images is that of a halo around the text in the FG/BG planes, as illustrated later in Figs. 8(a) and 8(c). This effect is very damaging to compression. Furthermore,



**Fig. 2** Example of segmenting soft edges. (a) Zoom of a portion of text and (b) one of its possible classifications into FG and BG, where it is noted the transition regions where no feature is evident. For a segmentation threshold of 128 (midgray) and replenishment values of 30 and 220, we obtain the (c) FG and (d) BG MRC layers. Note the intrusive contours caused by spurious transition-region pixels.



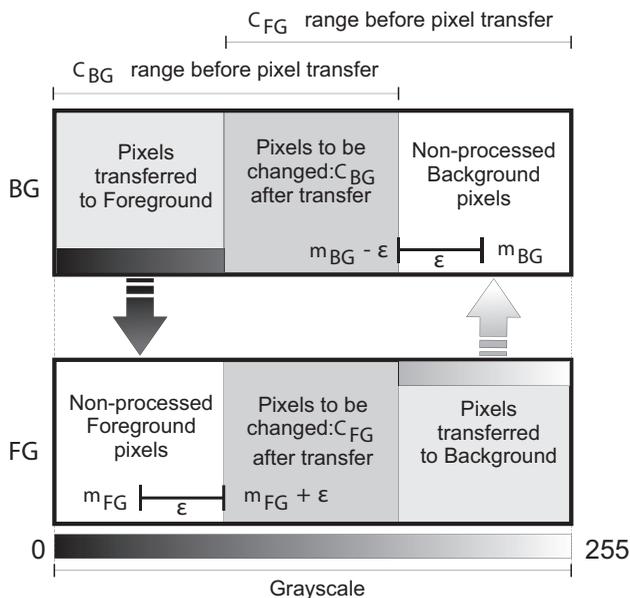
**Fig. 3** Scheme in which our solution is inserted, involving preprocessing, encoding, decoding, and postprocessing.

because it occurs inside the “useful” region, we cannot do much in this regard using only data-filling techniques.

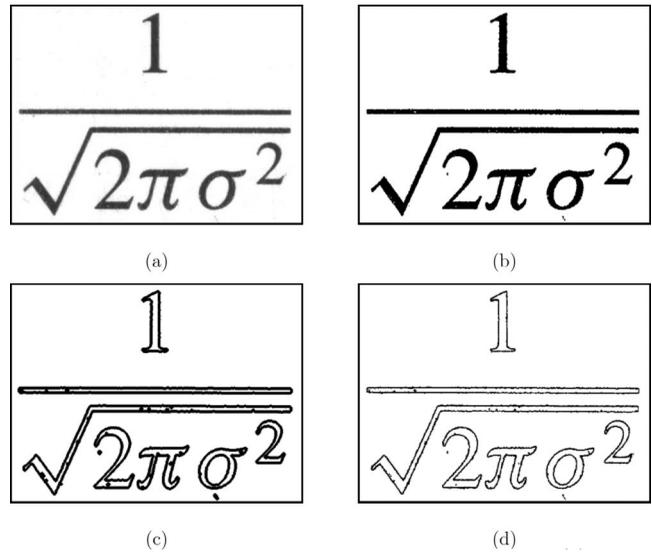
Figure 3 describes the complete scheme in which our solution is inserted. First, the original document  $X$  is input to a preprocessor that outputs a three-layer representation of the document  $Y$ . The preprocessor also constructs an edge sharpening map and estimates the original edge softness, both considered as side information. The layers are then MRC encoded, generating the encoded document. At the decoder, the encoded version of the document is decoded by the MRC decoder, resulting in  $Y'$ , a reconstructed version of MRC layers, which, together with the side information, is used by the postprocessor to assemble the reconstructed soft edge version  $X'$  of the document. This is an alternative to Ref. 32 in dealing with scanned data within MRC. There, they used dithering. Here, we rely on pre-/postprocessing.

### 3 Edge Sharpening and Softening

The halo cannot be removed with data filling, and we are forced to change the data. In effect, forcefully removing the halo is equivalent to changing the original image itself to make transitions sharper. We cannot assume that all image

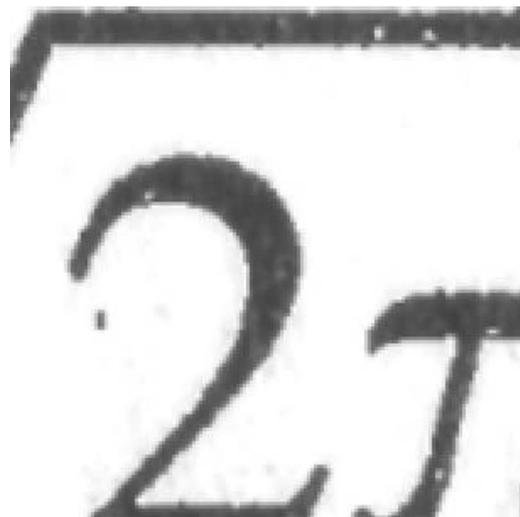


**Fig. 4** Transfer process: pixels marked by  $C_{BG}$  whose values are less than  $(m_{FG} + \epsilon)$  are transferred to FG layer. Pixels masked by  $C_{FG}$  whose values are greater than  $(m_{BG} - \epsilon)$  are transferred to the BG layer. The mask  $M$ ,  $C_{FG}$ , and  $C_{BG}$  are updated to accommodate this pixel transfer.

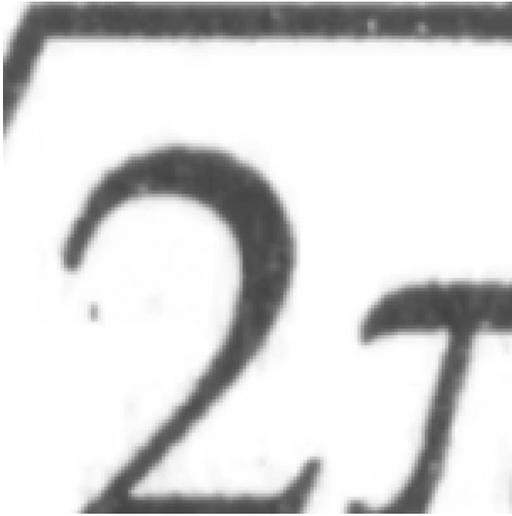


**Fig. 5** Finding the processing region: (a) Original scanned material, (b) mask  $M$  (c) candidate region for processing  $E$  and (d) region  $C$  marking pixels that could actually be changed.

edges can cause the problem nor that all mask transitions are subject to cause halo in scanned material. The effect occurs when transition in the mask coincides with the edges of the image. This means that a sharp mask transition is used to model a soft image transition, thus causing the unwanted spikes. Hence, the first step is to estimate where the halo will possibly occur. Our approach is to find transitions by applying the Sobel operator<sup>33</sup> to the mask. The resulting transitions are morphologically dilated by a  $(d \times d)$ -pixel structured element in order to mark a neighborhood. The image pixels that coincide with the dilated mask transitions are marked as possible processing targets. Let  $E$  be the set of pixel locations composing this region. The next step is to find pixels that are supposed to cause the halo effect. Let  $F$  and  $B$



**Fig. 6** Resulting image after edge “halo” removal processing. Note the sharper edges that coincide with the mask edges.



**Fig. 7** Resulting image after edge softening. Pixels in  $C$  were blurred by a  $15 \times 15$  Gaussian filter with standard deviation  $\sigma = 1.0$ .

represent the pixel positions where the mask indicates FG or BG, respectively. We compute averages as follows:

$$\begin{aligned} m_{FG} &= \text{mean}[x(i, j) | (i, j) \in F], \\ m_{BG} &= \text{mean}[x(i, j) | (i, j) \in B], \end{aligned} \tag{1}$$

where  $x(i, j)$  represents the original image.

We mark any pixel in the candidate region whose gray level is far apart from its layer average, i.e.,

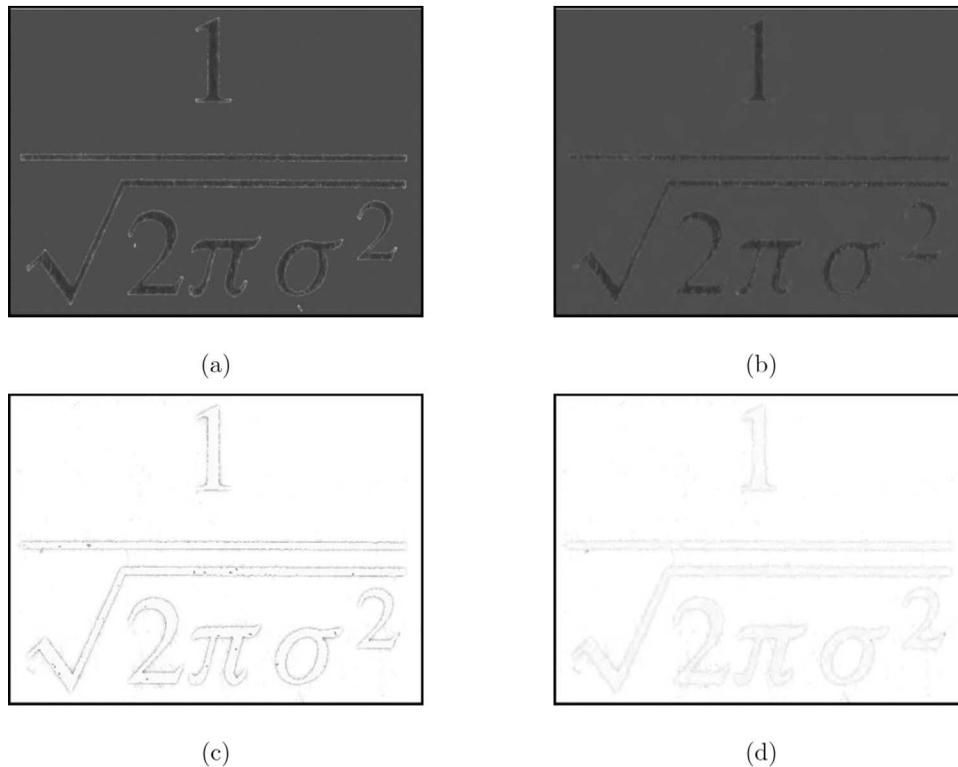
$$\begin{aligned} C_{FG} &= \begin{cases} 0 & : |x(i, j) - m_{FG}| > \epsilon | (i, j) \in (F \cap E) \\ 1 & : \text{otherwise} \end{cases}, \\ C_{BG} &= \begin{cases} 0 & : |x(i, j) - m_{BG}| > \epsilon | (i, j) \in (B \cap E) \\ 1 & : \text{otherwise} \end{cases}, \end{aligned} \tag{2}$$

where  $\epsilon$  is a tolerance value.

Next, we find the pixels marked by  $C_{BG}$  whose values are less than  $(m_{FG} + \epsilon)$ . These pixels are transferred to FG layer. On the other hand, pixels marked by  $C_{FG}$  whose values are greater than  $(m_{BG} - \epsilon)$  are transferred to the BG layer. These transferred pixels will be left untouched, since they now belong to general BG/FG. The mask  $M$ ,  $C_{FG}$ , and  $C_{BG}$  are updated to accommodate this pixel transfer, illustrated in Fig. 4.

For the image in Fig. 5(a), and for  $\epsilon = 16$  (out of 256 gray levels), the mask  $M$ , the candidate region for processing  $E$ , and the map of the pixels to be changed (i.e.,  $C = C_{FG} \cup C_{BG}$ ) are shown in Figs. 5(b)–5(d), respectively. In order to clean up the edge spots, we replace the values of the pixels in  $C_{FG}$  by  $m_{FG}$  and the values of the pixels in  $C_{BG}$  by  $m_{BG}$ . The result is shown in Fig. 6, where we can note how sharp the transitions are. This process is not the same as sharpening the image. Only regions that might be affected by the halo are processed. General FG/BG is left untouched.

If one can send as side information the map  $C$ , then we can blur only the pixels that belong to this map. The result is an image such as the one shown in Fig. 7, which was reconstructed using an  $h \times h$  Gaussian filter with standard deviation  $\sigma$  ( $h = 15$  and  $\sigma = 1.0$ , in this example). Note



**Fig. 8** Original (a) FG and (c) BG; processed (b) FG and (d) BG. Note the halo around the unprocessed FG/BG text. Preprocessing improves the quality of FG/BG planes.

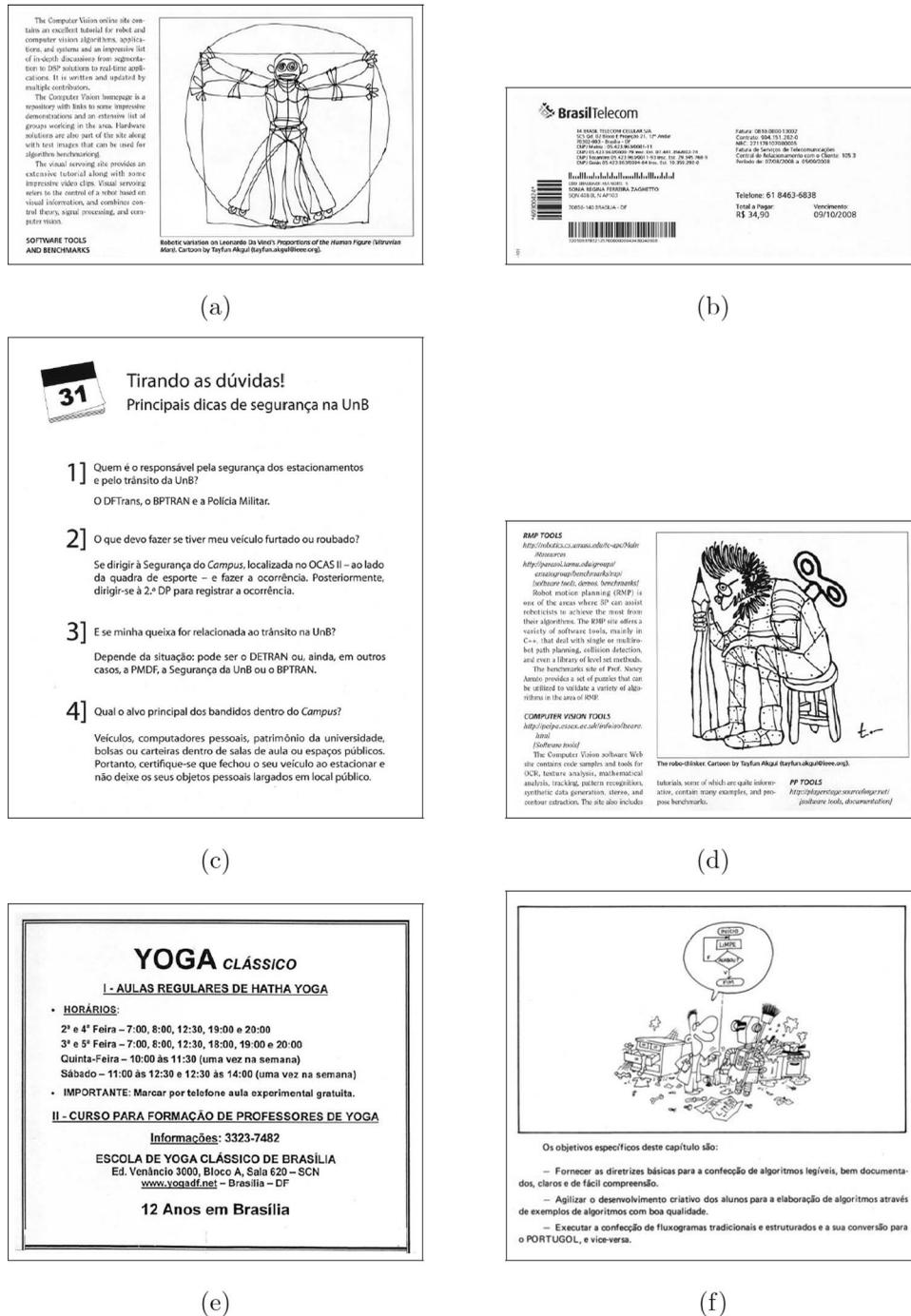


Fig. 9 Icons of the scanned text images used in our tests. Full resolutions are (a) 1284×2085 pixels, (b) 997×2456 pixels, (c) 1968×1685 pixels, (d) 1490×2100 pixels, (e) 1662×1943 pixels, and (f) 1464×1642 pixels. All images were captured at 300 dpi.

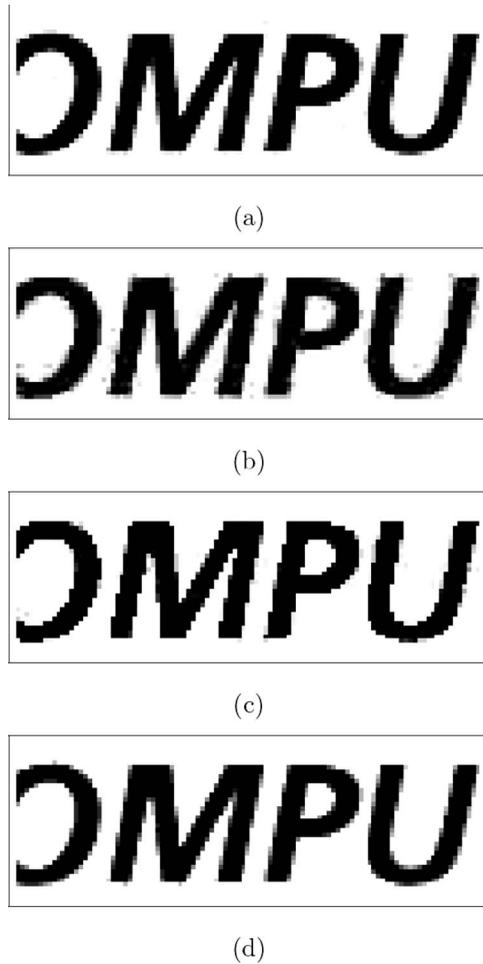
how only the edges that were preprocessed were softened again. In general, the image is not blurred. Because  $C$  is also a binary mask, it can be encoded using JBIG2. Skipping the postprocessing procedure may also be an alternative, because sharpened documents may lead to increased text readability and, thus, to better subjective quality.

Figure 8 shows FG/BG planes before and after halo processing and data-filling procedures. Note how the preprocessing improved the quality of the FG/BG planes. Datafilling is explained in the Appendix.

#### 4 Estimation of Pre- and Postprocessing Parameters

Because the edges are sharpened to accommodate the mask, in order to reconstruct soft edges, we have to somehow estimate the transition of the image edges.<sup>34</sup> The quest is to estimate the best values of parameters  $\epsilon$ ,  $h$ , and  $\sigma$  in a rate-distortion sense. For this, we determine the solution by minimizing the following cost function:

$$J(\epsilon, h, \sigma) = D + \lambda R, \quad (3)$$



**Fig. 10** Subjective comparison between methods: (a) original document (b) single-layer MRC (0.2056 bpp, 28.73 dB) (c) MRC without pre/postprocessing (0.2139 bpp, 31.06 dB) and (d) proposed MRC (0.1929 bpp, 31.96 dB).

where  $\lambda$  is a weighting factor,  $D$  is the distortion incurred by the preprocessing, MRC encoding/decoding, and postprocessing algorithms, and is defined as

$$D = \sum_{ij} |x(i, j) - \tilde{x}(i, j)|, \quad (4)$$

where  $x(i, j)$  and  $\tilde{x}(i, j)$  represent the original and reconstructed document, respectively.  $R$  is the bit rate for compressing the document and is defined as

$$R = R_{FG} + R_{BG} + R_M + R_C, \quad (5)$$

where  $R_{FG}$ ,  $R_{BG}$ ,  $R_M$ , and  $R_C$  are the rates for compressing FG, BG, M, and C, respectively.

The algorithm that determines the best values for preprocessing parameters,  $\epsilon$ ,  $h$ , and  $\sigma$ , is described as follows:

#### Algorithm 1

1.  $h \leftarrow h_0$ ;
2.  $\sigma \leftarrow \sigma_0$
3. **for**  $\epsilon \leftarrow \epsilon_0$  **to**  $\epsilon_k$
4.     **do** Generate map C using  $\epsilon$ ;
5.     Sharpen the edges using C;

6.     Run data-filling algorithm;
7.     MRC encode/decode FG, BG, and M;
8.     Encode C;
9.     Filter edges using a Gaussian filter with parameters  $(h_0, \sigma_0)$ ;
10.    Calculate and store cost  $J(\epsilon, h_0, \sigma_0)$ ;
11. Find  $\epsilon$  that results in the minimum cost  $J$  and make it  $\epsilon_{best}$ ;
12. Generate map  $C_{best}$  using  $\epsilon_{best}$ ;
13. Sharpen the edges using  $C_{best}$ ;
14. Run data-filling algorithm;
15. MRC encode/decode FG, BG, and M;
16. Encode  $C_{best}$ ;
17. **for**  $h \leftarrow h_0$  **to**  $h_i$
18.     **do for**  $\sigma \leftarrow \sigma_0$  **to**  $\sigma_j$
19.         **do** Filter edges using a Gaussian filter with parameters  $(h, \sigma)$ ;
20.         Calculate distortion  $D$ ;
21. Find  $(h, \sigma)$  pair that minimizes  $D$  and make it  $(h_{best}, \sigma_{best})$ .

Because FG and BG are encoded using AVC-I, a design quantizer parameter,  $QP_D$ , must be set for the MRC encoder in steps 7 and 15. The H.264/AVC quantizer parameter,  $QP$ , may vary from 0 to 51. Because we are interested in very low bit rates, a high  $QP_D$  ( $> 30$ ) is suggested.

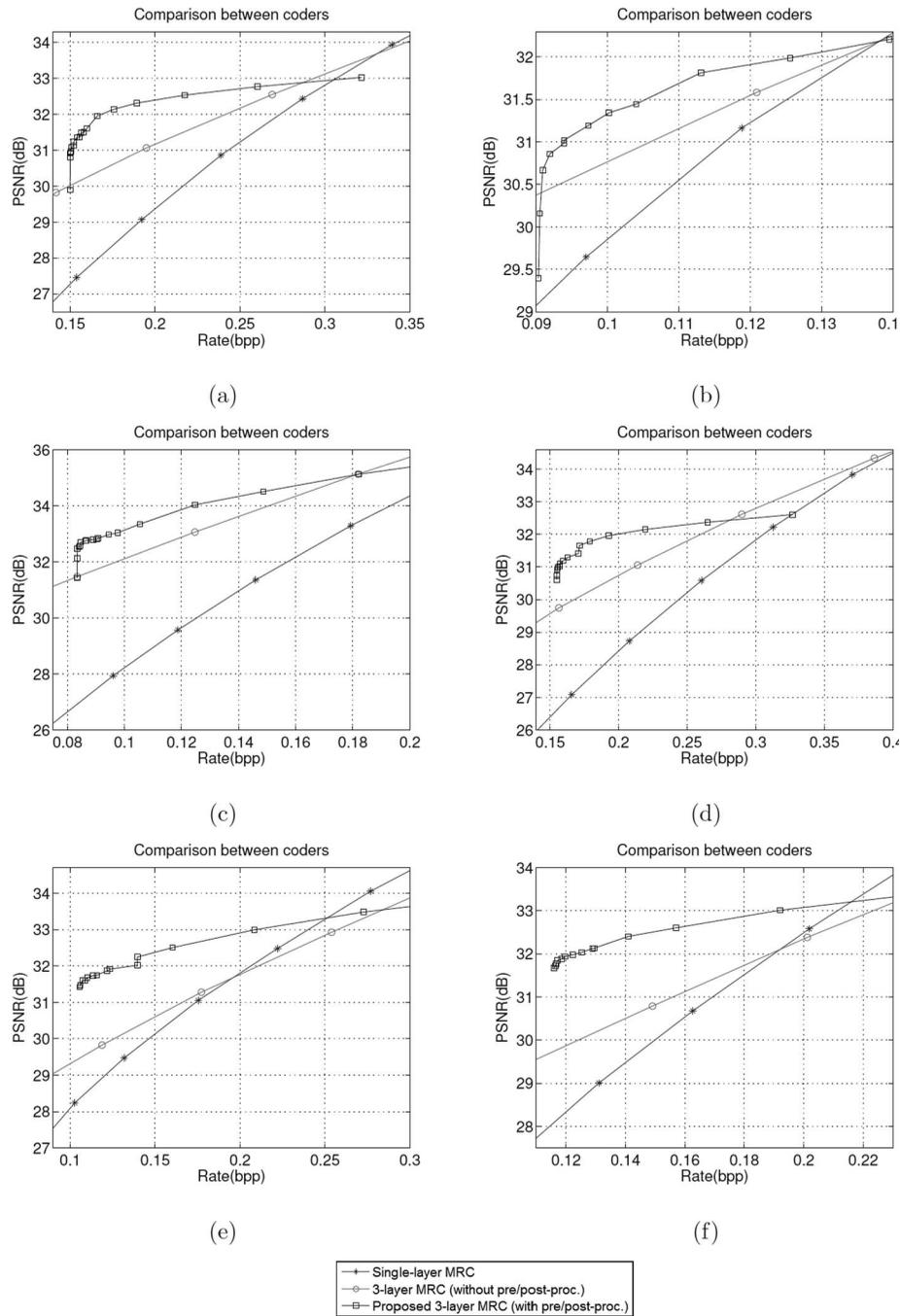
MRC imaging model also allows resolution change of FG/BG layers. Resize factors,  $S$ , of 1, 1/2, and 1/4 are used. The performance of the codec was evaluated for those values, as described by the following algorithm:

#### Algorithm 2

1. **for**  $S \leftarrow 1, 1/2, \text{ and } 1/4$
2.     **do for**  $QP \leftarrow QP_0$  **To**  $QP_k$
3.         **do** Generate rate-distortion points  $(R, D)$ ;
4.     Sort  $(R, D)$  points along  $R$ , in ascending order;
5.      $N \leftarrow$  number of  $(R, D)$  points;
6.     **for**  $i \leftarrow 1$  **To**  $N$
7.         **do if**  $D_i < D_{i-1}$
8.             **then** Select  $(R_i, D_i)$  point.

Because the main goal of the proposed method is to improve MRC rate-distortion performance, we have not addressed the problem of modeling the print/scan channel and, therefore, Algorithms 1 and 2 are used for each image separately.

In a simplified complexity analysis, we may consider that Algorithm 1 is dominated by the ‘‘MRC encode’’ operation (steps 7 and 15). H.264, used for foreground and background encoding, is far more complex than JBIG2 encoding, H.264 decoding, and image filtering operations. It is known that most of H.264 encoding complexity is due to the intra-/interprediction and rate-distortion optimization algorithms. A normal MRC scheme is implemented by one single MRC encode operation. As for Algorithm 1, step 7 suggests that if we have  $N_e$  values of  $\epsilon$ , the MRC encode operation



**Fig. 11** Objective performance comparison among coders for the text/graphics documents shown in Fig. 9. MRC with pre-/postprocessing outperforms MRC without pre-/postprocessing and single-layer MRC.

will execute  $N_e$  times. In step 15, this operation is executed once more. We conclude that our scheme is  $N_e + 1$  times more complex than a regular MRC encoder. In our tests, we used  $\epsilon = \{16, 28, 40, 52, 64\}$ , which leads to six times the complexity of normal approach. If we now consider Algorithm 2, with  $N_f$  different scale factors  $S$  and  $N_q$  different QP values, the complexity increases to  $N_e + N_f \times N_q + 1$  times the complexity of regular MRC, due to step 3. In our experiments, we used

$$QP_1 = \{29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51\},$$

for  $S = 1$ ,

$$QP_2 = \{18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51\},$$

for  $S = 1/2$ ,

$$QP_4 = \{07, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51\},$$

for  $S = 1/4$ ,

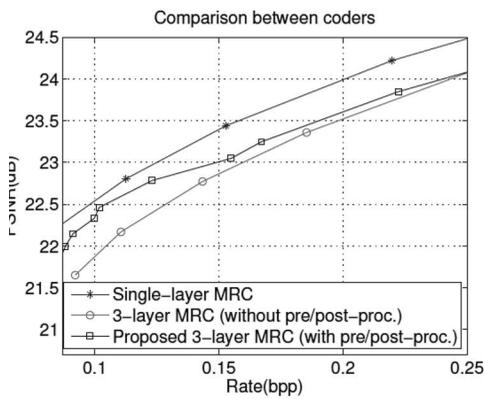
resulting in a 42-times-more-complex encoder.

## 5 Results

Images used in our tests are shown in Fig. 9. They were all captured with the same device. Parameters are picked from



(a)



(b)

**Fig. 12** Comparison among coders for compound documents. Even though, for more complex documents, improvement over single-layer MRC is not observed, three-layer MRC is improved through pre- and postprocessing.

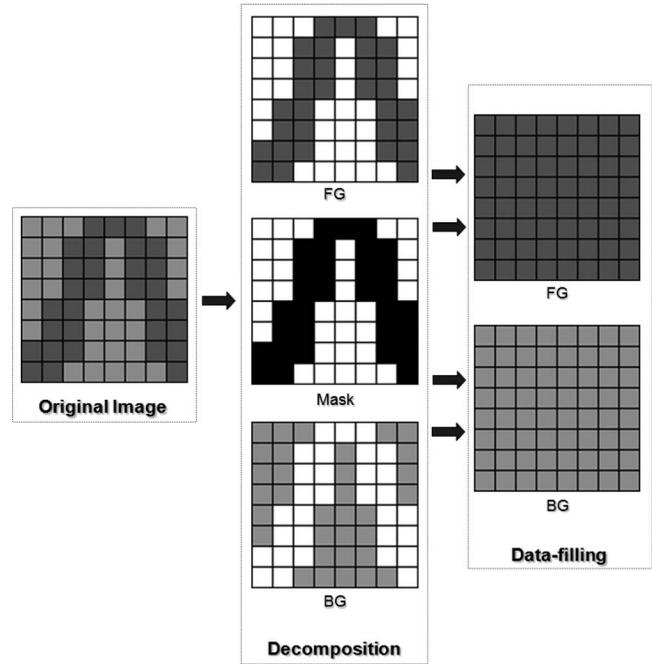
the following sets:

$$\epsilon \in \{16, 28, 40, 52, 64\},$$

$$h \in \{7, 9, 11, 13, 15, 17\},$$

$$\sigma \in \{0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.4\}.$$

The comparison was carried among three coders: (i) single-layer MRC, i.e. encoding the whole image with AVC-I; (ii) three-layer MRC, where the FG and BG layers are encoded with AVC-I while the mask is encoded with JBIG-2; and (iii) same as (ii) with pre- and postprocessing, (i.e., the proposed coder). Different rate-distortion points are obtained by varying the value of AVC-I's QP used to encode either the FG/BG or the single layer. The proposed MRC model typically outperforms MRC without pre-/postprocessing and single-layer MRC at lower bit rates, both subjectively and objectively. Figure 10 shows a zoomed part of a document in Fig. 9, comparing the compression using single and three-layer MRC. Results for the proposed MRC model indicate superior subjective quality.



**Fig. 13** Diagram of an MRC decomposer based on a segmenter and plane-filling algorithms.

Objective results for text imagery are presented in Fig. 11, comparing the three coder approaches. The proposed MRC model outperforms MRC without pre-/postprocessing and single-layer MRC at lower bitrates. However, there is only a short interval wherein there are gains using the proposed scheme. This occurs because the method is bounded in peak signal-to-noise-ratio (PSNR) due to the edge sharpening/softening procedure. Also, the achieved bit rate has a lower bound because of the number of bits needed to encode  $C$  and  $M$  losslessly.

For more complex documents, such as the one shown in Fig. 12(a), with large scanned pictorial content, performance improvements over single-layer MRC are not observed, as demonstrated in Fig. 12(b). Nevertheless, pre- and postprocessing improved the performance of the three-layer MRC coder. Presumably, a better segmentor could more efficiently split up the image into layers and contribute to rate-distortion improvements. Segmentation, however, is not the focus of this paper and can be addressed as a future work. Furthermore, segmentation may take more than compression into account. For example, depending on the application, one might want to make the mask readable by itself.

It is also important to notice that although the value of  $\epsilon$  varied from 28 to 52 and  $\sigma$  varied from 0.8 to 1.3, depending on the image, the algorithm selected the filter size as 11 in all cases. This suggests that some optimized parameters may be derived from the scanning conditions. If this is proven to be true, then one could significantly speed up the proposed scheme.

## 6 Conclusions

We have developed methods to counter balance the effects of soft edges in MRC compression of scanned documents. A method to sharpen only the edges that might cause the halo

effect was presented. We have also developed an algorithm to estimate, at the encoder, the original edge softness of the image using Gaussian filters and generate a halo location map optimized in a rate-distortion sense. The filter parameters and the map are used to reconstruct soft edges at the decoder. A MRC encoding and decoding scheme based on H.264/AVC-INTRA and JBIG2 has been used, and we have shown performance improvements for lower bit rates.

In some situations, MRC is outperformed by single-layer MRC. One such a scenario is to encode scanned documents with large pictorial content. Nevertheless, the decision to use MRC may not be based on objective rate-distortion analysis, but rather on the functionality provided by MRC and on its subjective quality at very low bit rates. Features such as increased text readability, optical character recognition of text, and easier content selection might be decisive factors. In any case, pre-/postprocessing procedures described here will certainly improve the objective quality of the MRC coder operating on scanned documents.

Although the proposed method is meant to deliver a reconstructed image that should be as similar as possible to the original scanned one, in some particular applications the postprocessing procedure may be turned off. Subjectively, sharpened (preprocessed only) documents may present better quality than resoftened (postprocessed) ones. Hence, the decoder might chose between softening or not the text. Encoding may also be operated applying only preprocessing, ignoring postprocessing, and not transmitting  $C$ . Furthermore, regular MRC decoders would ignore the  $C$  map and decode the sharper version.

The proposed approach improves the reconstruction fidelity in the MRC compression of scanned documents. In effect, our results have shown that the method enables competitive MRC compression of soft-edge document images.

Finally, the proposed MRC modifications have three problems that must be addressed in future works. First, it is much more computationally expensive than other MRC implementations, mainly because of the multiple MRC encode steps. This might not be a problem only if the document is compressed once and stored in a repository for multiple decoding. However, the hypothesis that optimized parameters may be used for a set of data captured with the same device is very reasonable and may be investigated. Furthermore, defining a set of profiles for specific scanning conditions could speed up the proposed method. Second, we understand that more conclusive subjective tests must be carried. Our results point to a possible subjective improvement, which may be further studied through a more comprehensive evaluation protocol. And, third, we did not investigate the use of better image segmentors, which could yield better rate-distortion performance.

## Appendix: Data Filling

Once the image is segmented, there will be “don’t care” regions on BG and FG layers (i.e., pixels assigned to the BG are marked as “don’t care” on the FG and vice versa). These pixels can be replaced by anything to enhance compression, as illustrated in Fig. 13. The problem of data filling over the redundant data has been studied.<sup>1,30,31</sup> This paper uses an iterative wavelet-based plane filling,<sup>35</sup> which we describe next.

Let  $I_0$  be the starting FG plane with “don’t care” pixels replaced by  $m_{FG}$ , which is the FG average calculated using Eq. (1). Also, let  $\tilde{I}_n$  be the compressed and decompressed version of  $I_n$  using a given coder at a target bit rate. If we plan to use a wavelet coder, then  $\tilde{I}_n$  can be approximated as:

$$\tilde{I}_n = W^{-1}\{\text{round}[W(I_n)/Q] * Q\}, \quad (6)$$

where  $W$  denotes the wavelet transform,  $\text{round}(\cdot)$  is a rounding operator, and  $Q$  is a step size to quantize the wavelet coefficients. It is expected to use quite large  $Q$  numbers, such as those that would yield very high compression ratios. Then, for  $n = 0$  until  $n = \nu$ , where  $\nu$  limits the number of cycles to a maximum, we compute

$$I_{n+1}(i, j) = \begin{cases} I_n(i, j) & : (i, j) \in F \\ \tilde{I}_n(i, j) & : (i, j) \in B. \end{cases} \quad (7)$$

We stop the loop either after  $\nu$  cycles or when

$$\text{mean}[|I_n(i, j) - I_{n-1}(i, j)|] < \xi, (i, j) \in B, \quad (8)$$

where  $\xi$  is some tolerance number (i.e., it stops when the filling in the “don’t care” region converges). The same process applies to the BG plane, replacing foreground by background notation and vice versa.

## References

1. R. L. de Queiroz, “Compressing compound documents,” in *The Document and Image Compression Handbook*, M. Barni, Ed. Marcel-Dekker, New York (2005).
2. R. L. de Queiroz, R. Buckley, and M. Xu, “Mixed raster content (MRC) model for compound image compression,” *Proc. SPIE* **3653**, 1106–1117 (1999).
3. R. L. de Queiroz, Z. Fan, and T. D. Tran, “Optimizing block-thresholding segmentation for multilayer compression of compound images,” *IEEE Trans. Image Process.* **9**(9), 1461–1471 (2000).
4. D. Mukherjee, N. Memon, and A. Said, “JPEG-matched MRC compression of compound documents,” in *Proc. of IEEE Int. Conf. on Image Processing*, Vol. 3, Thessaloniki, Greece, pp. 434–437 (October 2001).
5. D. Mukherjee, C. Chrysafis, and A. Said, “JPEG2000-matched MRC compression of compound documents,” in *Proc. of IEEE Int. Conf. on Image Processing*, Vol. 3, pp. 73–76 (2002).
6. “Mixed raster content (MRC),” ITU-T Recommendation T.44 (1999).
7. R. Buckley, D. Venable, and L. McIntyre, “New developments in color facsimile and internet fax,” in *Proc. of IS&T’s 5th Color Imaging Conf.*, Scottsdale, pp. 296–300 (November 1997).
8. IETF RFC 2301, File Format for Internet Fax, <ftp://ftp.isi.edu/in-notes/rfc2301.txt> (March 1998).
9. “Information technology—JPEG2000 Image Coding System – Part 6: Compound Image File Format,” ISO/IEC 15444-6 (2003).
10. D. Huttenlocher and W. Rucklidge, “DigiPaper: a versatile color document image representation,” in *Proc. of IEEE Int. Conf. Image Processing*, 25PS1.3, Kobe, Japan (October 1999).
11. L. Bottou, P. Haffner, P. Howard, P. Simard, Y. Bengio, and Y. LeCun, “High quality document image compression using DjVu,” *J. Electron. Imaging* **7**(3), 410–425 (1998).
12. K. Barthel, S. McPartlin, and M. Thierschmann, “New technology for raster document image compression,” *Proc. SPIE* **3967**, 286–290 (1999).
13. J. Huang, Y. Wang, and E. Wong, “Check image compression using a layered coding method,” *J. Electron. Imaging* **7**(3), 426–442 (1998).
14. “Advanced Video Coding for Generic Audiovisual Services,” ITU-T Recommendation H.264, ISO/IEC 14496-10 Advanced Video Coding (March 2009).
15. I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*, Wiley, Hoboken, NJ (2003).
16. T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Trans. Circuits Syst. Video Technol.* **13**(7), 560–576 (2003).
17. T. Stockhammer, M. M. Hannuksela, and T. Wiegand, “H.264/AVC in wireless environments,” *IEEE Trans. Circuits Syst. Video Technol.* **13**(7), 657–673 (2003).

18. T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.* **13**(7), 688–703 (2003).
19. G. J. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions," *Proc. SPIE* **5558**, 454–474 (2004).
20. J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: tools, performance, and complexity," *IEEE Circuits Syst. Mag.* **4**(1), 7–28 (2004).
21. "Information technology—Coded representation of picture and audio information - lossy/lossless coding of bi-level images," ITU-T Recommendation T.88 (March 2000).
22. "JBIG Maui Meeting Press Release: New document compression standard quadruples compression of today's fax standards and runs at unprecedented speeds," ISO/IEC JTC1/SC29/WG1 (December 1999).
23. D. Marpe, V. George, and T. Wiegand, "Performance comparison of intra-only H.264/AVC and JPEG2000 for a set of monochrome ISO/IEC test images," Contribution JVT ISO/IEC MPEG and ITU-T VCEG, JVT M-014 (October 2004).
24. D. Marpe, V. George, H. L. Cycon and K. U. Barthel, "Performance evaluation of Motion-JPEG2000 in comparison with H.264/AVC operated in pure intra coding mode," *Proc. SPIE* **5266**, 129–137 (2004).
25. R. L. de Queiroz, R. S. Ortis, A. Zaghetto, and T. A. Fonseca, "Fringe benefits of the H.264/AVC," in *Proc. of Int. Telecom. Symp.*, Fortaleza, Brazil, pp. 208–212 (September 2006).
26. A. Zaghetto and R. de Queiroz, "Segmentation-driven compound document coding based on H.264/AVC-INTRA," *IEEE Trans. Image Process.*, **16**(7), 1755–1760 (2007).
27. "Information technology—JPEG2000 Image Coding System—Part 1: core coding system," ISO/IEC 15444-1 (2000).
28. C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: an overview," *IEEE Trans. Consumer Electron.* **46**(4), 1103–1127 (2000).
29. D. S. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards, and Practice*, Kluwer Academic, Dordrecht (2002).
30. R. L. de Queiroz, "On data-filling algorithms for MRC layers," in *Proc. IEEE Int. Conf. on Image Processing*, Vancouver, Vol. **II**, pp. 586–589 (September 2000).
31. G. Lakhani and R. Subedi, "Optimal filling of FG/BG layers of compound document images," in *Proc. of IEEE Int. Conf. on Image Processing*, Atlanta, pp. 2273–2276 (October 2006).
32. G. Feng and C. A. Bouman, "High-quality MRC document coding," *IEEE Trans. Image Process.* **15**(10), 3152–3169 (2006).
33. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ (2002).
34. A. Zaghetto and R. L. de Queiroz, "Iterative pre- and post-processing for MRC layers of scanned documents," in *Proc. of IEEE Intl. Conf. on Image Processing*, San Diego, pp. 1009–1012 (October 2008).

35. R. L. de Queiroz, "Pre-processing for MRC layers of scanned images," in *Proc. of IEEE Int. Conf. on Image Processing*, Atlanta, pp. 3093–3096 (October 2006).



**Alexandre Zaghetto** received his Engineer degree from Universidade Federal do Rio de Janeiro, Brazil, in 2002, his MSc from Universidade de Brasília, Brazil, in 2004, and his Dr. degree from Universidade de Brasília, in 2009, all in electrical engineering. He joined the Computer Science Department at Universidade de Brasilia in 2009.



**Ricardo L. de Queiroz** received his Engineer degree from Universidade de Brasilia, Brazil, in 1987, his MSc from Universidade Estadual de Campinas, Brazil, in 1990, and PhD from The University of Texas at Arlington, in 1994, all in electrical engineering. From 1990 to 1991, he was with the DSP research group at Universidade de Brasilia, as a research associate. He joined Xerox Corp. in 1994, where he was a member of the research staff until 2002. From 2000 to 2001, he was

also an adjunct faculty member at Rochester Institute of Technology. He joined the Electrical Engineering Department at Universidade de Brasilia in 2003, and in 2010, he became a full professor in the Computer Science Department. He has published over 140 articles in journals and conferences and has contributed chapters to books as well. He also holds 46 issued patents. He is an elected member of the IEEE Signal Processing Society's Image, Video and Multidimensional Signal Processing (IVMSP) and Multimedia Signal Processing (MMSP) Technical Committees. He is a past editor for the *EURASIP Journal on Image and Video Processing*, *IEEE Signal Processing Letters*, *IEEE Transactions on Image Processing*, and *IEEE Transactions on Circuits and Systems for Video Technology*. He has been appointed an IEEE Signal Processing Society Distinguished Lecturer for the 2011–2012 term. His research interests include image and video compression, multirate signal processing, and color imaging. He is a senior member of IEEE, and a member of the Brazilian Telecommunications Society and the Brazilian Society of Television Engineers.