Statistica Sinica 3(1993), 117-139

IMAGE RESTORATION: FLEXIBLE NEIGHBORHOOD SYSTEMS AND ITERATED CONDITIONAL EXPECTATIONS

Heping Zhang

Yale University School of Medicine

Abstract: The iterated conditional expectations (ICE) procedure is studied under the Ising model. If the parameter in the Ising model is not well chosen, experiments show that the ICE suffers from problems of over- and under-smoothing. To detect and avoid these problems, a criterion is proposed in which the performance of the iterations is predicted. Based on this criterion, we have an algorithm that eliminates the over- and under-smoothing iterations, adjusts the parameter in the Ising model, and finally leads to the "right" parameter and the "right" number of iterations. The ICE procedure is successfully modified by the proposed algorithm according to the experiments.

Key words and phrases: Markov random field (MRF), iterated conditional expectations (ICE), Ising model, image restoration, stopping rule.

1. Introduction

Owen (1986, 1989) introduced the method of iterated conditional expectations (ICE) for segmenting images using the Ising model that is described in Section 2. As a simple modification of Besag's (1983) iterated conditional modes (ICM) algorithm, ICE replaces the discrete problem of ICM by a continuous one that retains more information from one iteration to the next. Such an idea is closely related to the theory of mean field approximation that has been modified for various applications; see, e.g., Bilbro et al. (1988). In very noisy images, ICE greatly improves the image segmentation, as shown in Owen (1986, 1989). A number of authors (e.g., Marroquin (1985), Kent and Maria (1988), Johnson et al. (1989) and Manbeck (1990)) applied ICE-like methods successfully in different contexts and showed the advantage of using the ICE scheme by comparing it with other competing methods.

However, convergence properties of the ICE algorithm are unknown. To study its performance, we design various experiments and make empirical observations. The numerical evidence suggests that the algorithm usually converges in

the sense that the restored image changes very little after a large number of iterations, though not necessarily resulting in a useful image. Without a well-chosen parameter in the Ising model, the restored image suffers from severe over- and under-smoothing problems. The present paper proposes a performance criterion to study the process of the image reconstruction. Based on this criterion, we implement a stopping rule which terminates the over- and under-smoothing iterations, and an adjusting procedure to search for a parameter which cures these problems. We succeeded using these protocols to modify the ICE scheme.

In the rest of the paper, Section 2 introduces the model and reviews some useful methods. In Section 3, two experiments are designed to illustrate how the choices of the parameter in the Ising model cause the over- and under-smoothing problems for the ICE method. More examples are studied in Section 4 to show the impact of introducing the performance criterion, the stopping rule, and the parameter adjusting procedure to the ICE algorithm.

2. Model and Methods

2.1. Model and notation

Suppose that the original image is an $M \times N$ array and that x_t denotes the gray level at pixel $t = (i, j)$, taking values from 0 to k. For binary images, $k = 1$. The configuration space is $\Omega = \{0, \ldots, k\}^n, n = MN$. A general model for noise degradation used in image processing is

$$Y_t = g(X_t)N_t^1 + N_t^2, \tag{1}$$

where the Y_t's are the intensities of the observed image, N_t^1 and N_t^2 are noise variables at pixel t, and $g(\cdot)$ is a blur function; see Jain (1989).

We consider a special case of the model (1):

$$Y_t = X_t + \epsilon_t, \tag{2}$$

namely, we assume the identity blur function and a degenerated speckle noise. So, each pixel t is associated with an undegraded gray level X_t, an observed intensity Y_t, and a noise level ϵ_t.

Many algorithms have been proposed to reconstruct images based on model (2). This paper focuses on the maximum *a posteriori* (MAP) approach by following the notation of Besag (1986).

Assumption 1

$$l(y|x) \equiv P[Y = y|X = x] = \prod_t f(y_t|x_t), \tag{3}$$

where $f(\cdot|\cdot)$ is a known conditional density. This assumption implies that the observed intensity at each pixel is conditionally independent given the true scene.

Assumption 2

$$p(x_t|X_{S\setminus t}) \equiv P[X_t = x_t|X_{S\setminus t}] = p_t(x_t|X_{\partial t}), \tag{4}$$

where p_t is specific to pixel t, $X_{\partial t}$ denotes the neighbor gray levels of pixel t, and $X_{S\setminus t}$ denotes gray levels elsewhere. That is, the undegraded image X is a realization of a locally dependent Markov random field with specified distribution $p(x)$. This assumption allows us to estimate the gray level at the current pixel through its neighborhood only. In this paper, the neighborhood of a pixel $t = (i, j)$ is defined by

$$\partial t = \{(k, l) : (i-k)^2 + (j-l)^2 \leq d\}.$$

When $d = 1$, 2, and 4, we call the resulting neighborhood systems the first, second, and third orders, respectively.

For a binary image, we consider the Ising model and take

$$p_t(1|X_{\partial t}) = \frac{\exp(\beta b_t)}{\exp(\beta b_t) + \exp(\beta w_t)}, \tag{5}$$

where b_t and w_t are the numbers of 1's and 0's in $X_{\partial t}$. Then, the conditional distribution of X_t, given the values of $X_{\partial t}$ and Y_t, is determined by

$$P[X_t = 1|X_{\partial t}, Y_t = y_t] = \frac{f_1(y_t)\exp(\beta b_t)}{f_1(y_t)\exp(\beta b_t) + f_0(y_t)\exp(\beta w_t)}, \tag{6}$$

where $f_c(y_t) \equiv f(y_t|X_t = c)$, $c = 0, 1$.

We estimate the true image X with $\hat{x}$ which maximizes the posterior probability

$$p(x|y) \propto l(y|x)p(x). \tag{7}$$

2.2. Methods

(i) Annealing method

Geman and Geman (1984) considered a system

$$p_T(x, y) \propto \{l(y|x)p(x)\}^{1/T},$$

where $T > 0$ is a parameter. In the limit as $T \to 0$, $p_T(x, y)$ is concentrated on $\hat{x}$.

They designed an algorithm using the Gibbs Sampler. At the kth iteration, choose "temperature" $T(k) = C/\log(1 + k)$. If C is large enough, the procedure

converges and finds $\hat{x}$ after a sufficient number of iterations. Since the rate that $T \to 0$ is very slow and the constant C is required to be large, the annealing process is computer-intensive. When the signal-to-noise ratio (SNR) is low, small C (e.g., $C = 1$) works well in some simulations. The magnitude of C depends on the SNR, but to our knowledge no explicit relation is available. To reduce the amount of computation, Gidas (1989) proposed a renormalization group (RG) approach. The RG method computes the MAP estimator of the undegraded image, by generating iteratively a multilevel *cascade* of restored images corresponding to different levels of resolution. His simulations show that the algorithm is much faster than the original annealing method and that the results are improved. However, it is usually impossible to justify the strong assumptions required in his theory and computation. A more severe problem of the simulated annealing, suggested by the work of Greig, Porteous and Seheult (1989), is that with *practicable* 'temperature' schedules it generally produces poor approximations to the MAP estimate to which it converges.

(ii) Iterated conditional modes

Besag (1986) proposed a simple, iterated method to avoid intensive computations based on the local characteristics of the image. At each iteration, x_t is assigned 1 if (6) is larger than 0.5 and 0 otherwise. $p(\hat{x}|y)$ never decreases at any stage during updating, and hence converges to a local maximizer of the posterior distribution. This procedure converges rapidly, but its results are not satisfactory in general. Jubb and Jennison (1988) suggested a modification which extends the range of ICM to high noise cases as well as reducing the computational cost.

(iii) Exact maximum a posteriori estimation

For binary images, Greig et al. (1989) showed how the images with MAP probability can be evaluated exactly using efficient variants of the Ford-Fulkerson algorithm for finding the maximum flow in a certain capacitated network. However, the parameter β involved in their model is considered to be fixed and known as with other existing methods. In Section 3, it will be seen that the choice of β is crucial to the quality of the restored images, as is also shown in their numerical results. There is no single choice of β to fit all images. For the simulations, we know the truth and thereby adjust the parameter. In practice, we should select β dynamically to best suit the image of interest. In fact, this motivates the current research.

(iv) Iterated conditional expectation

Owen (1989) noted the following situation: If a pixel t has eight neighbors, all with the most recent posterior probability 0.51 of being a 1 (black), then ICM handles it the same way as one in which all neighbors had posterior probability 0.99 of being a 1, but in quite a different way from a pixel with all neighbors having probability 0.49 of being a 1. There is some loss of information in this

process of ICM. Owen (1986) proposes to replace the quantity b_t in (6) with $\sum_{s\in\partial t} p_s$; here p_s is the current value of $P[X_s = 1|X_{\partial s}, Y_s = y_s]$. The updates are via

$$p_t = \frac{f_1(y_t)\exp\left(\beta\sum_{s\in\partial t} p_s\right)}{f_1(y_t)\exp\left(\beta\sum_{s\in\partial t} p_s\right) + f_0(y_t)\exp\left[\beta\sum_{s\in\partial t}(1-p_s)\right]} \tag{8}$$

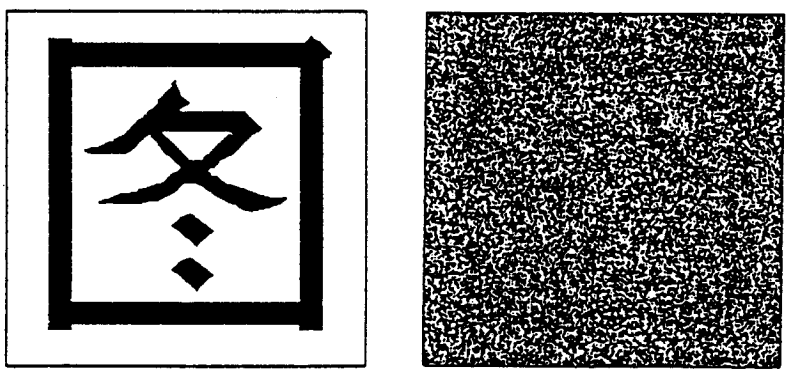
and ICE delays the assignment of black or white pixel until the end. Such an assignment may also be extended to the multicolor images. However, we shall not discuss this extension in this paper.

3. The Effect of β and the Neighborhood System

We now use ICE and show how the choices of the parameter β and the neighborhood system (defined in the previous section) in the Ising model affect the quality of the restored images. In all examples, the pictures are plotted with 256 by 256 pixels, where white and black pixels have bits 0 and 1, respectively.

Example 1. *Chinese Character*

Figure 1(a) is a Chinese character meaning "picture". We simulated our data Y according to model (2) and i.i.d. normal noise $N(0, 16)$, with the true image displayed in Figure 1(a). Figure 1(b) is a pointwise degraded image obtained by assigning a pixel to be black if its observed intensity is above 0.5, or white otherwise.

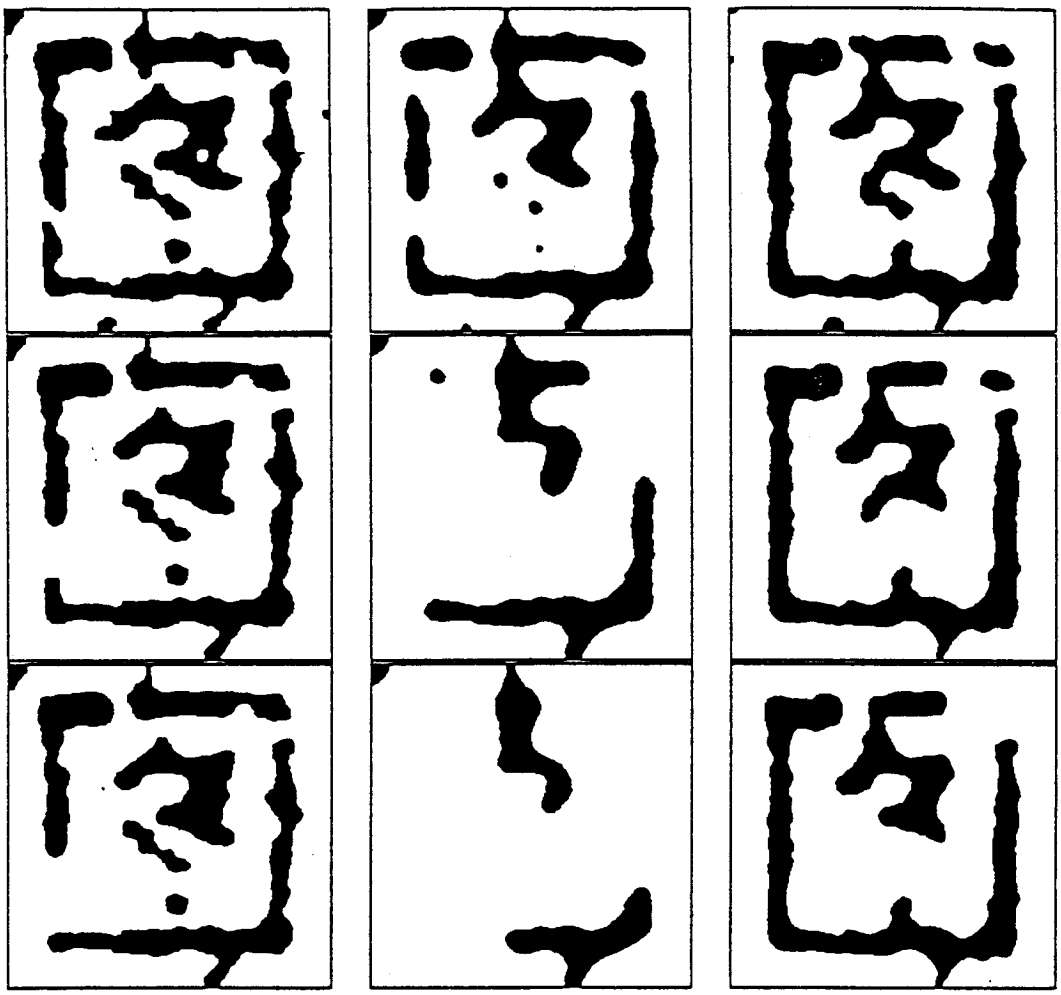


(a) Original image (b) Degraded image

Figure 1. Chinese character

Experiment 3.1. *This experiment is designed to show how the restored images change over the iterations. The program is stopped at the 100th, 500th and 1000th iterations. To illustrate how the neighborhood systems and the associated*

parameter β influence the resulting images, three kinds of neighborhood systems are used with typical choices of β.



(First, 100)	(Second, 100)	(Flexible, 100)
(First, 500)	(Second, 500)	(Flexible, 500)
(First, 1000)	(Second, 1000)	(Flexible, 1000)

Figure 2. Comparison of restored images from different orders of neighborhoods and iterations

We started with the first order neighborhood and $\beta = 1.0$. The three images in the first column of Figure 2 are the restored images from the 100th, 500th and 1000th iterations. Next, we tried the second order neighborhood with $\beta = 0.7$. The restored images are similarly displayed in the middle column of Figure 2. Now, Algorithm 1 below allows flexible neighborhood systems during updating, and the resulting images are shown in the last column of Figure 2.

Algorithm 1. *Choose Flexible Neighborhoods*

At the current pixel t, let p_t be defined in (8).

If $(0.49 < p_t < 0.51)$ then use the third order neighborhood and $\beta = 0.5$;
else if $(p_t > 0.35$ or $p_t < 0.65)$ then use the second order and $\beta = 0.7$;
else use the first order neighborhood and $\beta = 1.0$.

This algorithm chooses a larger neighborhood for a pixel having posterior probability close to 0.5. The cutoff points 0.35, 0.49, 0.51, and 0.65 are selected after a number of experiments. Owen (1989), among others, discussed the rationale of choosing β. Here, our rationale is to resolve the over- and under-smoothing problems for the ICE algorithm.

Remarks on Figure 2: (i) The most risky choice is the second order neighborhood with $\beta = 0.7$, which results in a very serious over-smoothing problem though it happens quite late. (ii) The process is very stable when the first order neighborhood is used with $\beta = 0.5$, however, the images are not smooth enough. (iii) When the order of the neighborhoods varies according to Algorithm 1, the process is stable and the images are smooth. Unfortunately, it is hard to establish an optimal way to vary the neighborhood systems (cf. Derin and Won (1987)). We do not pursue this topic here.

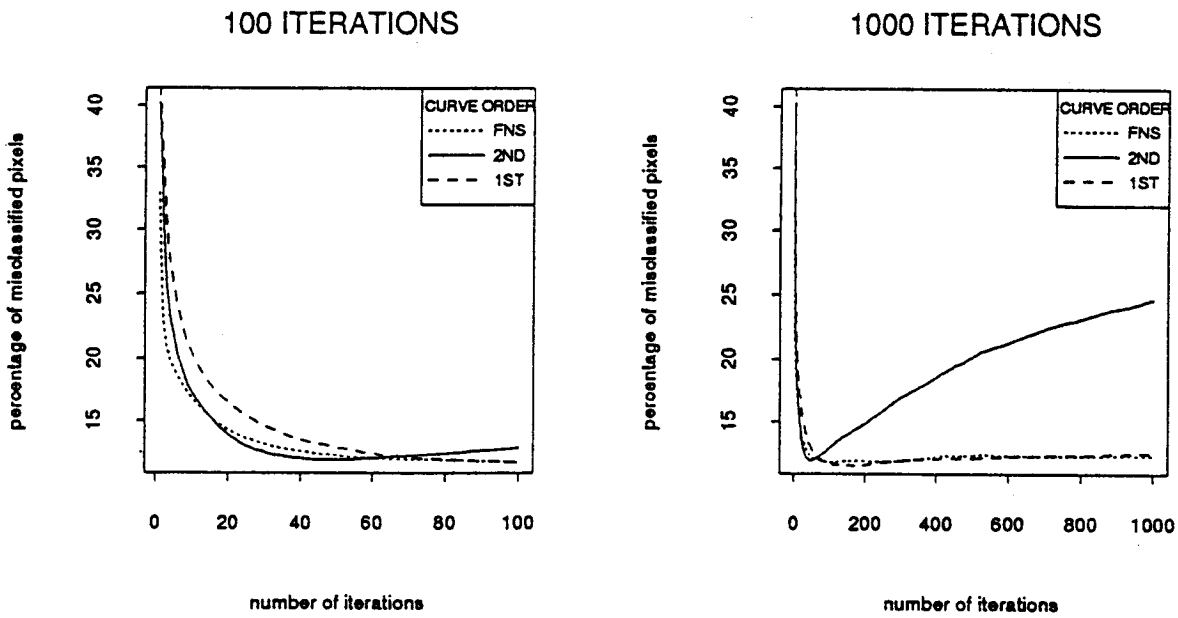


Figure 3. Misclassified ratio

To get more insight into the problems, we now introduce the misclassification rate, which is the ratio of the number of the misclassified pixels to n. Let us examine Figure 3, where the horizontal axis is the number of iterations and the vertical axis is the percentage of the misclassified pixels. The dotted, solid, and

dashed lines correspond to the misclassification rates obtained by the first, second, and flexible neighborhoods assigned by Algorithm 1, respectively. The left panel is an enlarged picture of the right one for only 100 iterations. It can be seen from this figure that the two curves resulting from the first and flexible orders merge at the stable states after about 80 iterations. In contrast, the curve from the second order increases after reaching its minimum at about the 40th iteration.

The misclassification rate is a natural criterion to judge the performance of image restoration as demonstrated in Figure 3, from which, however, the smoothness of the images has not been exploited enough. For example, the insignificant departure between the dotted and solid curves in Figure 3 does not reflect differences in smoothness of those corresponding images in Figure 2 (columns 1 and 3).

Experiment 3.2. *In the previous experiment, we fixed the parameter values with the specified order of neighborhood systems. In this experiment, we fix the neighborhood to be second order, but choose a series of* $\beta = 0.2, 0.3, 0.4, 0.5, 0.7, 1.0$.

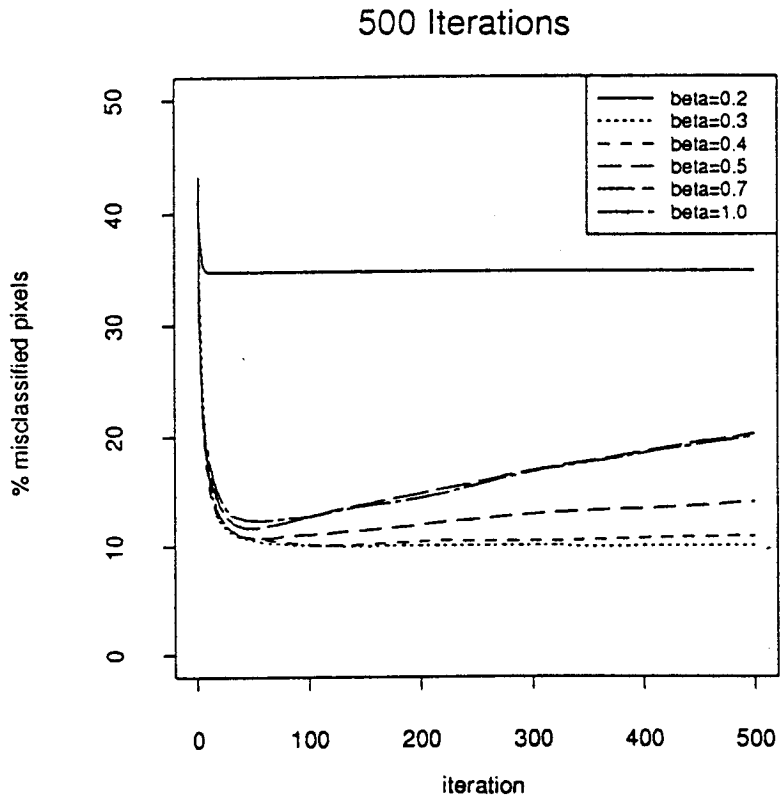
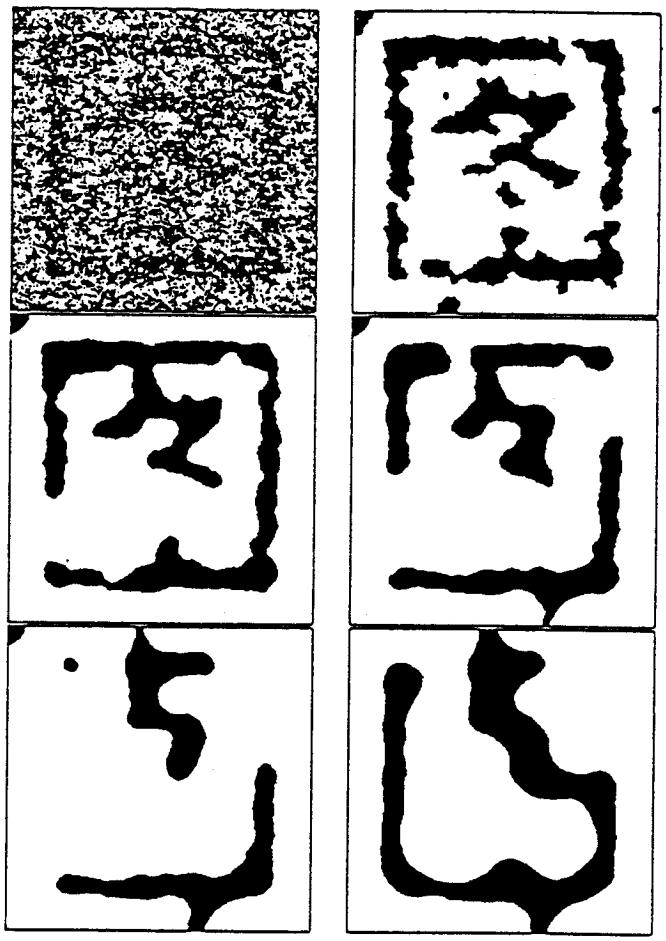


Figure 4. Misclassified rate for different parameters

Figure 4 uses the misclassification rate to show how the restored image changes for different β at each iteration, where each curve is labeled with the

corresponding β. we see that all curves decrease drastically at the beginning, but they behave differently later on. For small β like 0.2, the curve flattens very soon, and the misclassification rate cannot be reduced further. For large β like 1.0, the curve rises rapidly after reaching its minimum. For moderate β, say 0.3, the curve seems desirable. Figure 5 vividly displays all the images restored from the experiment after 500 iterations. From left to right and then top to bottom, the images come from $\beta = 0.2, 0.3, 0.4, 0.5, 0.7$, and 1.0.



$\beta = 0.2$	$\beta = 0.3$
0.4	0.5
0.7	1.0

Figure 5. Restored images from ICE with the second order neighborhood and different parameter β's after 500 iterations

We can learn from Figures 4 and 5 that there exists a critical value of β (e.g., 0.3 here), above which the restored image smoothes over and over such that much of its detail is lost; on the other hand, below which the image remains noisy. The former problem is called over-smoothing and the latter under-smoothing.

To resolve the over- and under-smoothing problems, one would presumably look for an "optimal" choice of β. We begin with a small β; if the process sticks early, increase β and restart. Repeat the process until an ideal value of β is found, and then use the final β to run for a while. Our experiments suggest that this is a safeguard procedure, since starting with large values of β tends to lose the fine, detailed features of the image. The question is how and when to increase β. A criterion will be proposed shortly.

4. Modification of ICE

4.1. Stopping rule

In this section, we try to predict the performance of the image restoration process. For simplicity of the algorithm, we merely focus on chasing the misclassification trend and providing information about the smoothness of the restored images.

4.1.1. Black and white intensities

Recalling Assumptions 1 and 2, we estimate the proportion of black pixels, denoted by R_{bw}, by the observed proportion,

$$\hat{R}_{bw} = \frac{1}{n}\sum Y_t. \tag{9}$$

When the restored image is either over-smoothed or under-smoothed, the fraction of its black pixels, say IR_{bw}, usually differs from R_{bw} as well as from $\hat{R}_{bw}$ (a trivial application of the strong law of large numbers). Moreover, the further IR_{bw} deviates from R_{bw}, the higher is the misclassification rate. We use the difference

$$d_{bw} = \left| IR_{bw} - \hat{R}_{bw} \right| \tag{10}$$

as one of the indicators for the quality of the restored image. Indeed, *a restored image cannot be good if it does not have the right amount of pixels of different colors.*

Example 1. (continued) *Chinese Character*

The Chinese character is now degraded by normal noise with two levels; $\sigma = 1.0$ and 4.0. The misclassification rate and d_{bw} are recorded at each iteration.

First, consider $\sigma = 4.0$ and use the second order neighborhood with $\beta = 0.5, 1.5$. Then, σ is re-set to 1.0. Figure 6 shows the results.

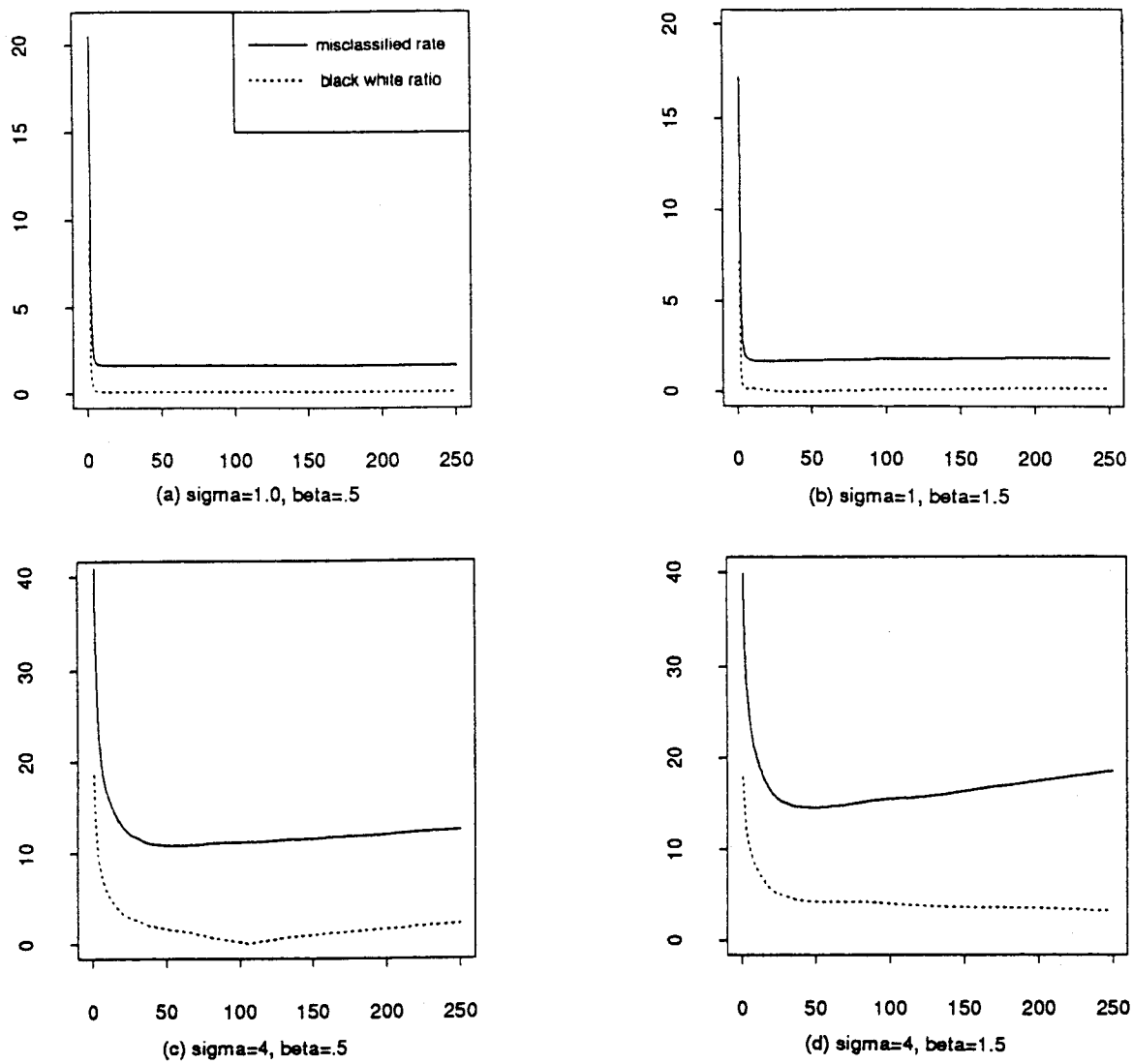


Figure 6. Similarity of misclassified rate and black white ratio. The horizontal and vertical axes are the number of iterations and the percentage of pixels, respectively.

Panel (c) is the plot for $\sigma = 4.0$ and $\beta = 0.5$; where the solid curve corresponds to the misclassification rate, and the dotted one to $100 \cdot d_{bw}$. The similarity of the two curves is persuasive. In Panels (a) and (b), the misclassification rate curves are above the other ones by approximately constant distances. In Panel (c), both curves behave similarly except for the period from the 50th to 100th iterations. In Panel (d), the misclassification rate curve goes up after about 30 iterations due to over-smoothing, but d_{bw} keeps decreasing slowly. The misclassification rate and d_{bw} share the feature that they first decrease drastically and then change (up- or downward) gradually after an intermediately decreasing period. So, the key lesson from Figure 6 is that *the iteration should be terminated when d_{bw} either changes much slower than it did at an early stage, or it begins to*

increase, because this signals that there is no more gain from further iterations. The following algorithm adds this simple idea to the ICE scheme.

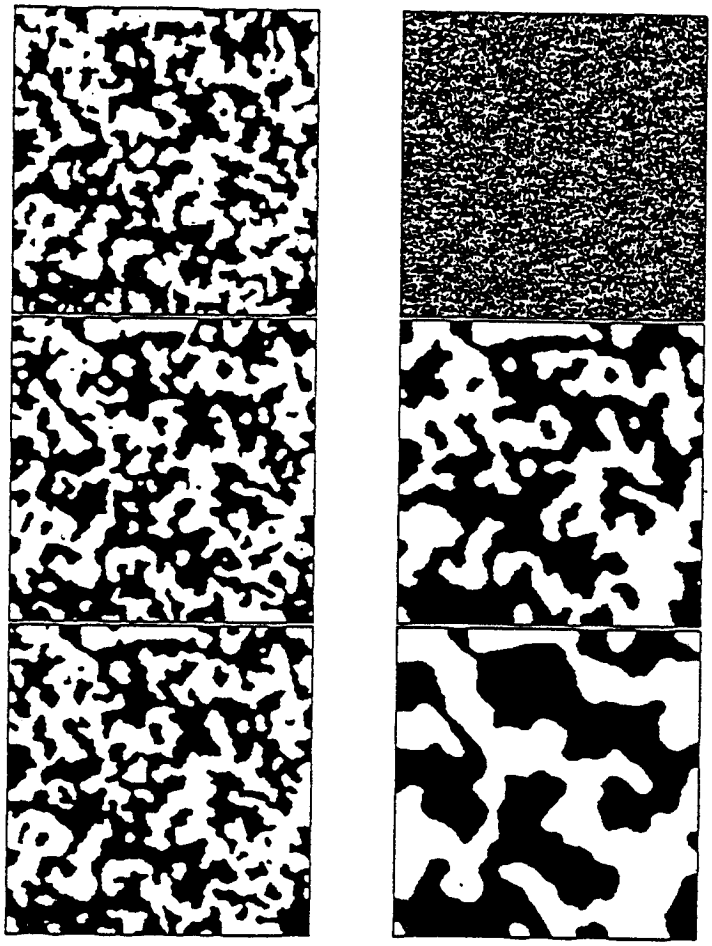
Algorithm 2. *Stopping Procedure Based on* d_{bw}
1. find $\hat{R}_{bw}$ from (9);
2. set j, the jump between iterations, proportional to the noise level (e.g., $[\sigma]+1$);
3. first compute d_{bw} in (10) from the pointwise degraded image and then from the restored image after $j+1$ iterations. Let df_0 be the difference between the two d_{bw}'s;
4. at iteration $jk+1$, compute df_k, which is the difference of d_{bw} between iterations $jk+1$ and $j(k-1)+1$;
5. continue until $df_k < \text{constant} \cdot df_0$.

Remarks on Algorithm 2: (i) Step 2 is optional. The reason for skipping a fixed number (j) of iterations is twofold. First, it takes $O(n)$ operations to calculate IR_{bw} and hence d_{bw}. One would avoid tracking d_{bw} at every iteration to improve the computational efficiency. Second, df_k from two consecutive iterations may be affected too much by the noise contained in the restored images. Jumping over some iterations can avoid the random fluctuation and catch the overall trend. Since σ controls the noise level in the restored images, j is set to be proportional to σ. On the other hand, a larger j implies a longer delay of our decision. To be conservative, we suggest $j = [\sigma]+1$. (ii) The constant in Step 5 is chosen so that the process stops near an ideal point; that is, we are not able to gain more from further iterations. The following example demonstrates this algorithm.

Example 2. *Markov Random Field* (MRF)

The image data is a "simulated" third order MRF obtained by the algorithm suggested in Cross and Jain (1983), with parameter 1.0 for the constant term, 3.0 for the first order neighbors, 2.0 for the second order neighbors, and 1.0 for the third order neighbors. It is not crucial to estimate the exact order of neighborhood systems and parameters of the MRF for the reconstruction problem. We select the second order neighborhood with two choices of β, 0.5 and 1.0. The noise $N(0,4)$ is added into the true image. The jump in Algorithm 2 is set at 3, i.e., $\sigma+1$.

The simulated MRF image is plotted in the upper-left panel of Figure 7, with the pointwise degraded image on the right. According to Algorithm 2 the process stops at the 7th and 10th iterations for $\beta = 0.5$ and 1.0, respectively. The middle and bottom images in the first column of Figure 7 are the respective restored images. The restored images with (on the left) or without (on the right) the stopping rule are compared in the bottom two rows of Figure 7.



β	Markov field	Degraded field
0.5	Restored image with stopping rule	Restored image without stopping rule
1.0	Restored image with stopping rule	Restored image without stopping rule

Figure 7. Simulated Markov random field

According to the pairs of images with or without the stopping rule in Figure 7, our procedure indeed helped us to stop at the right time. Figure 8 shows it more precisely; where the solid curve is the misclassification rate for $\beta = 0.5$, the dotted curve for $\beta = 1.0$, and the asterisks for the stopping spots. The stopping spots are well located at the minimum of the curves. Moreover, Figure 7 shows that, without the stopping rule, large β results in serious over-smoothing; see the right-bottom image. After implementing the stopping rule, we prevent the over-smoothing quite well; see the left-bottom image. This phenomenon is also apparent in other image data like the Chinese Character and the Scotland Map.

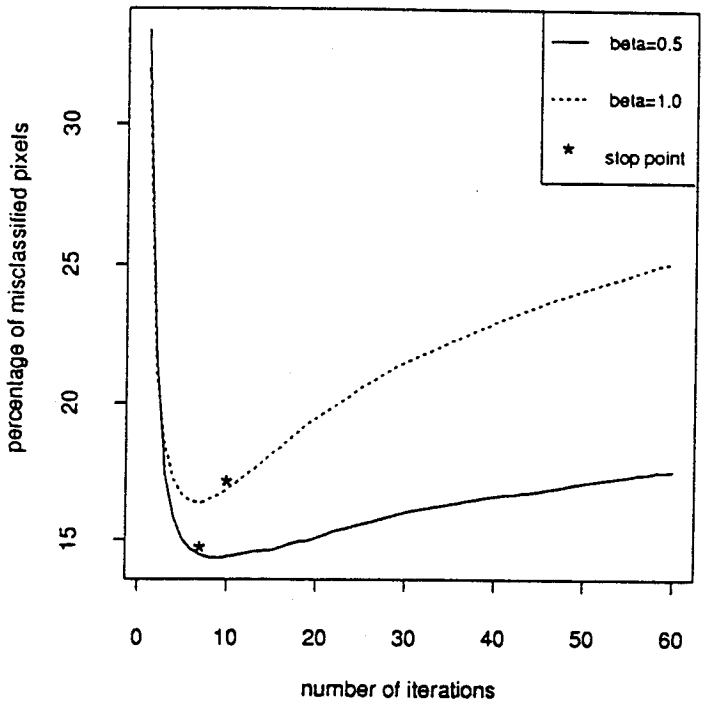


Figure 8. Misclassified rate and stopping points

Our final goal is to find an automatic procedure for choosing both the optimal parameter β and stopping time. To this end, it is necessary to establish a more sophisticated criterion than d_{bw} alone. For example, in the low noise cases, where the noises do not destroy the image too much, we could have

$$|NR_{bw} - R_{bw}| \leq |\hat{R}_{bw} - R_{bw}|, \tag{11}$$

where NR_{bw} is the fraction of the black pixels in the pointwise degraded image; that is, $\hat{R}_{bw}$ itself is not accurate enough. Therefore, d_{bw} is not very helpful when σ is very small. Further, we saw from Panel (d) in Figure 6 that d_{bw} alone is not enough in the high noise cases. As was mentioned earlier in Section 3, even the behavior of the misclassification rate can be predicted: it lacks information on the smoothness of the images. The problem is overcome by introducing the proportion of boundary pixels as a measure of smoothness, denoted by BR; here the boundary pixels are ones whose first order neighborhood has at least one pixel of different color. This is another important feature of the images. *BR is usually small when the image of interest is smooth.* From now on, $\hat{BR}$ denotes the proportion of the boundary pixels for the restored images. Our procedure should result in small $\hat{BR}$ and d_{bw} under the assumption that the image is smooth.

4.1.2. Smoothness and intensities

Let us start by studying Example 1 again.

Example 1. (continued) *Chinese Character*

First, we find $\hat{R}_{bw} = 0.2709$, as opposed to $R_{bw} = 0.2741$. They are quite close. In simulation, we can also obtain the proportion of the boundary pixels BR, which is 0.0675. However, we cannot estimate it from Y as we did for R_{bw}.

Next, use the first and second order neighborhood systems associated with different values of β. Table 1 summarizes the numerical results from ICE after 500 iterations. The first column shows the order of the neighborhood systems; NA is for the pointwise degraded image. The next column lists the β values. The last two columns are, respectively, the ratios of the boundary and black pixels of the restored images.

Table 1. Comparison of ratios for chinese character

order	β	$\hat{BR}$	IR_{bw}
NA	0.00	0.9313	0.4779
1	0.25	0.8463	0.4601
1	0.50	0.3927	0.3681
1	0.75*	0.0714	0.2752
1	1.00	0.0607	0.2708
1	1.25	0.0542	0.2503
1	1.50	0.0601	0.2696
2	0.20	0.6858	0.4234
2	0.30*	0.0746	0.2708
2	0.40	0.0538	0.2587
2	0.50	0.0452	0.2275
2	0.70	0.0303	0.1611
2	1.00	0.0378	0.2606

The β values marked with asterisks have the virtue that both $\hat{BR}$ and IR_{bw} are close to the underlying true values. The following combination, q of d_{bw} and $\hat{BR}$ is suggested for searching for such choices of β

$$q = \frac{\hat{BR}}{\sigma^2 + 1} + d_{bw}. \tag{12}$$

The quantity q contains the information of the misclassification rate through d_{bw} as well as the smoothness through both terms in (12).

Remarks on q: (i) In general, when $\hat{BR}, d_{bw}$, and hence q are "reasonably" small, the restored image, obtained from ICE, is rather good because it is smooth and has a reasonable amount of pixels of different colors. However, we need to pay

some attention to the situation; for instance, in the chess board (see the first row in Figure 11), if we simultaneously over-smooth a black square somewhere and a white square somewhere else, then d_{bw} remains unchanged and $\hat{BR}$ decreases, so q is smaller; but the image gets worse. It is worthwhile noting that the over-smoothing usually occurs after the period when q changes slowly. If q cannot be significantly reduced, the process will be terminated soon because further reduction of q is not meaningful. So, we usually quit before simultaneous black and white over-smoothing.

(ii) Formula (12) is a *simple* combination of $\hat{BR}$ and d_{bw} which trades off the two factors as follows. In the low noise cases, well restored images could have larger d_{bw} than the pointwise degraded images because $\hat{R}_{bw}$ is not accurate enough; the closeness to it means inaccuracy to some extent. q will be reduced as $\hat{BR}$ decreases though d_{bw} may increase. This is why under-smoothing is avoided. In the high noise cases, the magnitude of d_{bw} plays a more important rule than $\hat{BR}$. The weight $\frac{1}{\sigma^2+1}$ is introduced to prevent the dominance of $\hat{BR}$ over d_{bw} in the high noise cases; $\frac{1}{\sigma+1}$ is another reasonable choice.

(iii) At the initial step of ICE, one can artificially make a picture with $d_{bw} \approx 0$, in which half is white and the other half is black so that $\hat{BR}$ cannot be smaller. It appears that q has nothing to do in this situation. Fortunately, that is not the case. We can easily get around by asking the questions: (a) Why should we start from this initialization and why not? (b) If there is no reasons to answer why not, do we expect to reconstruct our image in one iteration? If not, we skip the beginning one or two iterations, and keep chasing q afterwards. Otherwise, at the initial step, we should compute d_{bw} from a randomly chosen part of the image.

4.1.3. More examples

More experiments will be conducted in order to convince that q provides useful messages for the image restoration, although evaluating a restored image is subjective. It is beyond the scope of this paper to discuss the evaluation of a restored image. The chinese character is used as a main example because it is less controversial, at least among Chinese, by visualizing it. In all our experiments we know the true image. As a simple way out, we sometimes plot the misclassified pixels accompanying the true and restored images. In the following, the second order neighborhood is chosen. q and the misclassification rate are recorded up to 500 iterations, and Figures 9 and 10 exhibit their relationship. Figure 9 is similar to Figure 6, but d_{bw} in Figure 6 is replaced with q in Figure 9. In Figure 10, the x-axis is the misclassification rate and y-axis is $100 \cdot q$.

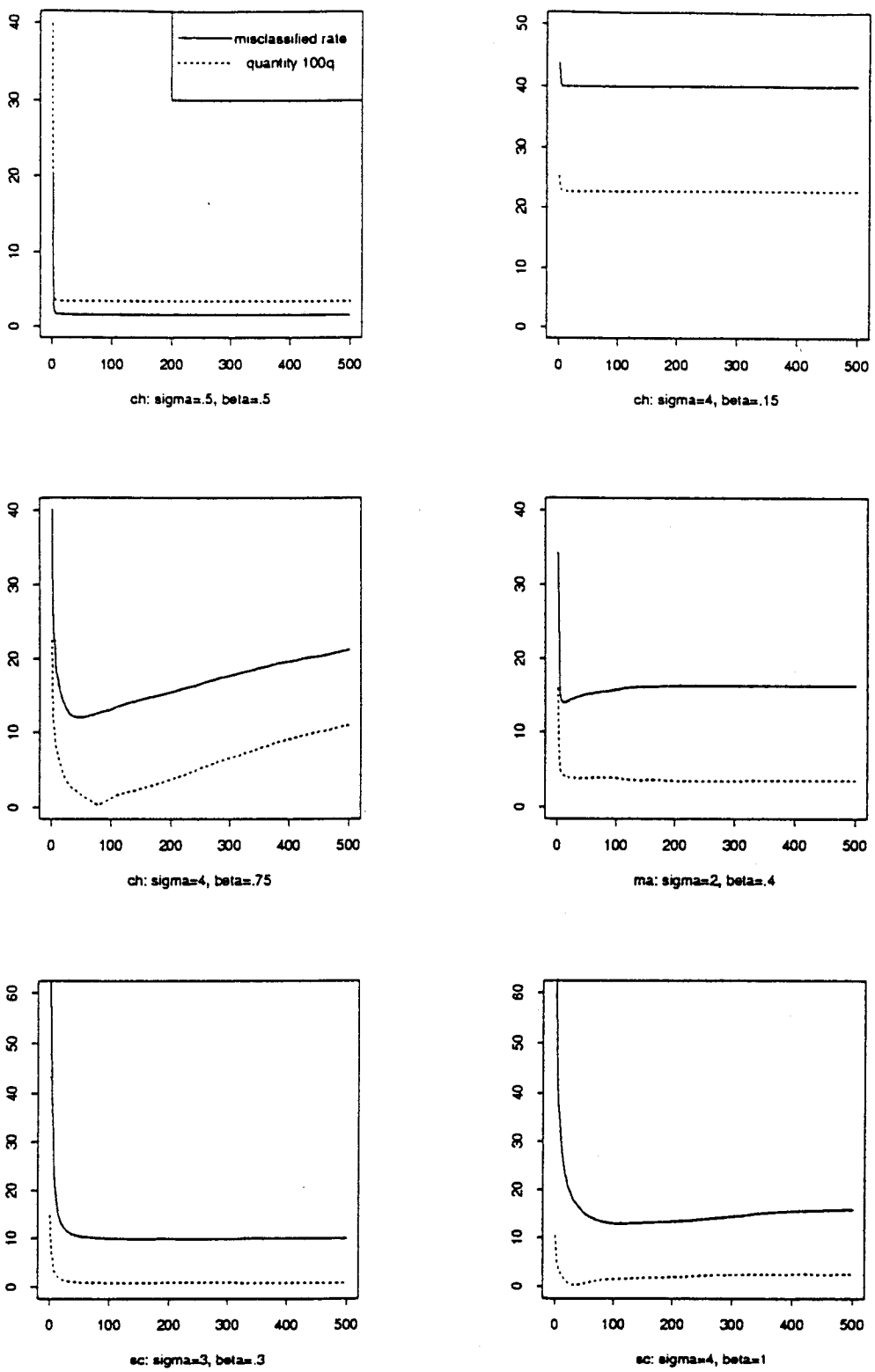


Figure 9. Similarity of misclassified rate and quantity $100q$. The horizontal and vertical axes are the number of iterations and the percentage of pixels, respectively.

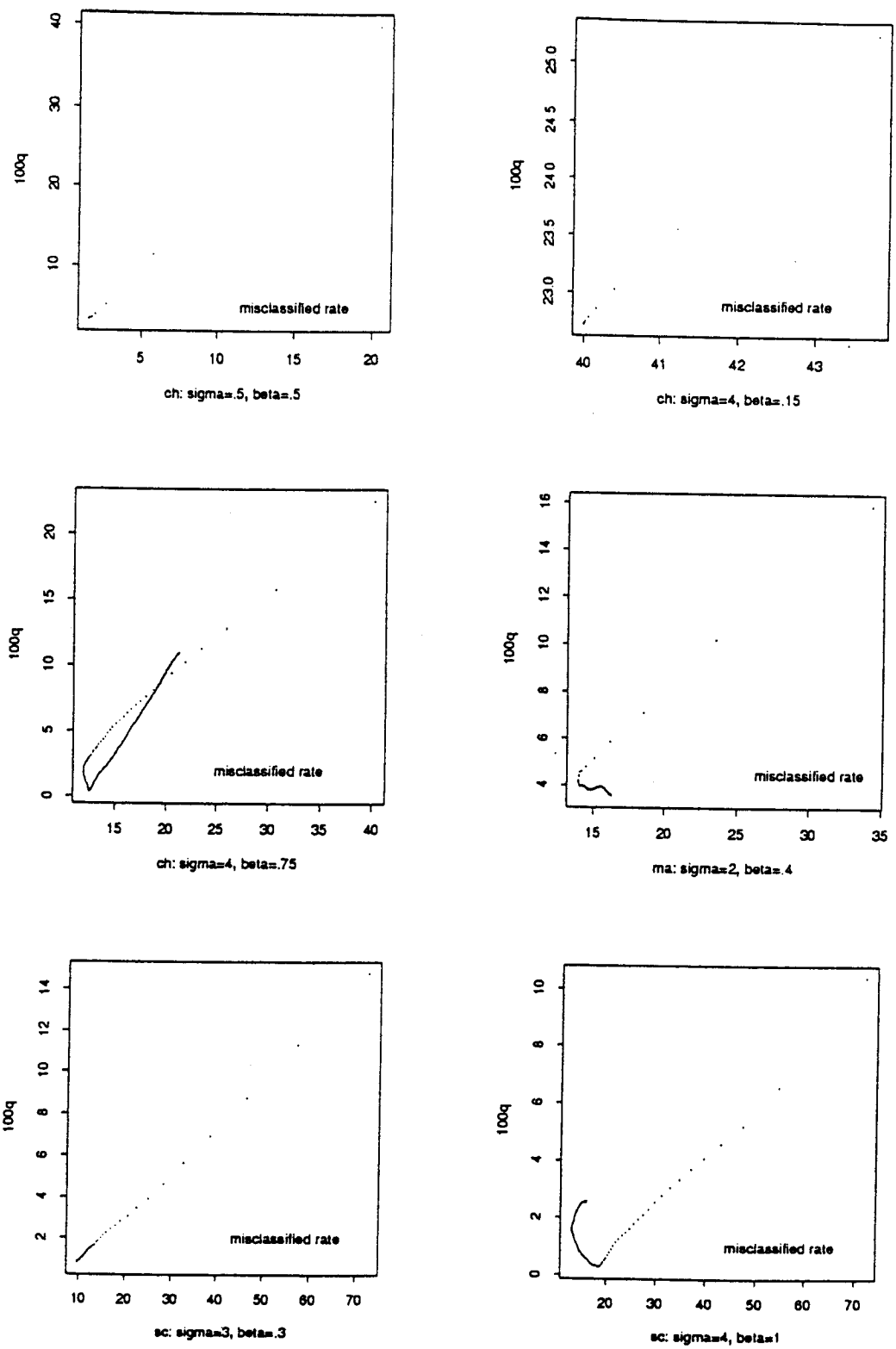


Figure 10. Misclassified rate (horizontal axis) vs quantity $100q$ (vertical axis).

Example 1. (continued) *Chinese Character*

The top-left panels in Figures 9 and 10 show that q and the misclassification rate are linearly correlated for the cases $\sigma = 0.5$, $\beta = 0.5$ and $\sigma = 4.0$, $\beta = 0.15$. Corresponding to $\sigma = 4.0$ and $\beta = 0.75$, the middle-left panels in Figures 9 and 10 are more interesting. At the beginning, the misclassification rate decreases significantly and q retains the linear relationship with it. When it starts to decrease and then increase slowly, q has the negative correlation with it. After a while, it increases more rapidly and q retains the strictly linear relationship with it.

Example 2. (continued) *Markov Random Field*

As before, $\sigma = 2.0$. Choose $\beta = 0.4$. The middle-right panels in Figures 9 and 10 show similar phenomena when q and the misclassification rate are not too small.

Example 3. *Scotland Map*

The original image data is the map of Scotland; see the third image in the first column of Figure 11. We tried two experiments. One has $\sigma = 3.0$, $\beta = 0.3$; the other $\sigma = 4.0$, $\beta = 1.0$. The corresponding graphs are at the bottom of Figures 9 and 10.

From these figures we can conclude that *when q is not too small, it has a clear linear relationship with the misclassification rate*; when it is too small, it implies d_{bw} is very small and the relationship may be destroyed by the inaccuracy of d_{bw} which results from $\hat{R}_{bw}$. Luckily, the accuracy of q is usually good enough to determine the optimal parameter and stopping time, as our examples suggest.

4.2. Main algorithm

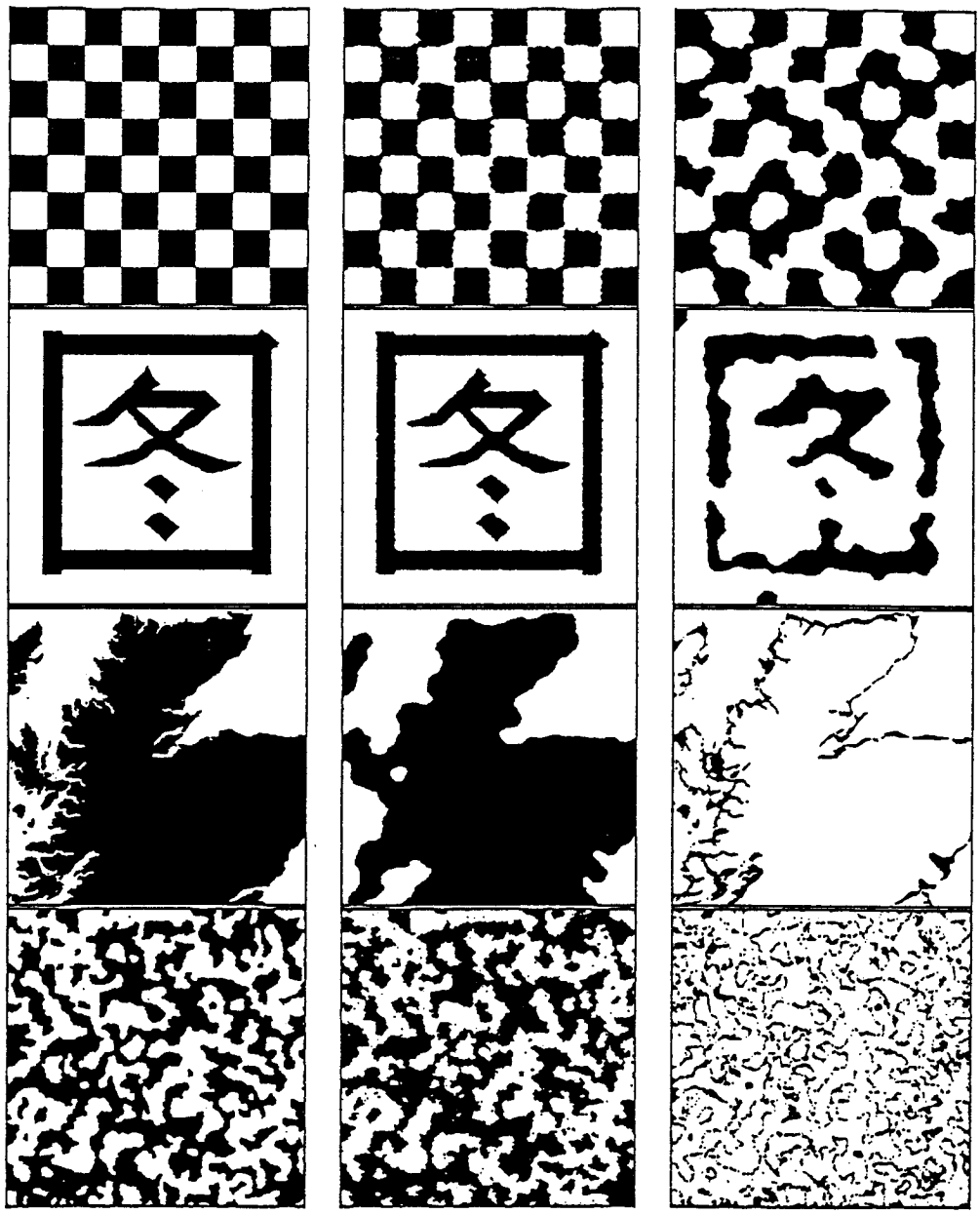
Now is the time to propose an algorithm which summarizes the previous discussion.

Algorithm 3. *Modification of ICE*

1. initialize β. A reasonable choice for many examples we studied is 0.1. Usually, the higher the noise level, the smaller β should be chosen to begin with;
2. set the amount of increment of β, usually at 0.1 or 0.05;
3. use the initial β to run the ICE procedure;
4. pause running if q decreases slowly. This is determined by Algorithm 2 in which d_{bw} is replaced with q;
5. if q is not below some small number, (0.09 works for all our examples) and β is not beyond some level, (the common choices are 0.9 - 1.3 for σ below 1, 0.5 - 0.7 for σ between 1 and 2, 0.4 - 0.5 for σ between 2 and 3, 0.3 - 0.4 for the rest) then increase β and go back to Step 3; otherwise terminate the process.

4.3. Demonstration

Four different kinds of images are used to demonstrate Algorithm 3.



Original image	Restored or misclassified images	
Chess board	Restored board $\sigma = 1.0$	$\sigma = 3.0$
Chinese character	Restored character $\sigma = 0.5$	$\sigma = 4.0$
Scotland map	Restored map $\sigma = 4.0$	Misclassified pixels
Markov field	Restored field $\sigma = 2.0$	Misclassified pixels

Figure 11. Comparison of images

Example 4. *Chess Board*

The original image is plotted on the top-left of Figure 11. The top-middle and top-right ones are restored from the degraded images with noise levels $\sigma = 1.0$ and 3.0, respectively.

Example 1. (continued) *Chinese Character*

The second row in Figure 11 is for the Chinese Character. The original image is on the left; next to it are the restored images from the noise levels $\sigma = 0.5$ and 4.0, respectively.

Example 3. (continued) *Scotland Map*

Now, $\sigma = 4.0$. Since it is not easy to asses the Scotland map visually, we display the true and restored images together with the misclassified pixels in the third row of Figure 11.

Example 2. (continued) *Markov Random Field*

Three images at the bottom of Figure 11 are plotted as in Example 3.

4.4. Conclusions

All restored images look remarkable. For the noise level σ below 2.0, we can almost completely reconstruct the true image; for σ above 2, the main structures are well recovered. Table 2 summarizes the goodness of fit for all examples, from which we can see that for restored images, the fraction of the black and boundary pixels is close to the true fraction. The misclassification rates are also listed.

Table 2. Goodness of restored image

		black ratio		bound ratio		mis pixels
image	σ	true %	restored %	true %	restored %	mean %
chinese	0.5	6.75	6.77	27.41	27.40	0.5325
chinese	4.0		6.18		26.39	9.6527
scotland	4.0	6.03	2.39	42.35	41.41	6.0392
markov	2.0	20.36	22.50	50.01	50.84	14.436
chess	1.0	10.64	10.27	50.00	50.07	2.6245
chess	3.0		9.31		48.27	10.6232

Throughout this paper, ICE serves as the main scheme. As is evident from our examples, ICE works very well after we set the automatic procedure for choosing the optimal parameter and stopping time. Our ideas are not restricted to ICE nor to the model (2), and should be applicable in implementing other procedures. For instance, for a noisy image resulting from transmitting a true binary image with certain known mis-transmission rate, one may replace σ^2 in (12) with the

mis-transmission rate and build q into this algorithm (not necessarily ICE) to guide the image processing.

Although we focus here on binary images, the ICE scheme can be extended to deal with multicolor images and to different contexts as mentioned in the introduction. For multicolor images, as a simple illustration, R_{bw} will simply be the average intensity of the true scene and the definition for boundary pixels remains. Hence, q is directly applicable.

In all examples, we added normal noises to true images, but the normality is not needed at all for the validity of our procedures.

The quantity q in (12) plays the key rule in the update step. It carries two important features of a restored image. However, we cannot offer a theoretical justification whether q can lead to the convergence of restored images to the true one. Fortunately, our experiments indicated that Algorithm 3 is not so sensitive to choices for which $\hat{BR}$ and d_{bw} are so combined that they will both be small if q is small and in which σ has a counter effect on $\hat{BR}$. (12) happens to be the simplest candidate to bear this information. Successfully, it leads ICE to produce better images, terminate nuisance iterations, and prevent the over- and under-smoothing problems.

Acknowledgment

I am very grateful to Professors Richard Olshen, Art Owen and Paul Switzer for their encouragement and helpful suggestions.

Reference

Besag, J. (1983). Discussion of paper by P. Switzer. *Bull. Internat. Statist. Inst.* **47**, 77–92.

Besag, J. (1986). On the statistical analysis of dirty pictures. *J. Roy. Statist. Soc. Ser.B* **48**, 259–302.

Bilbro, G., Mann, R., Miller, T. K., Synder, W. E., Van den Bout, D. E. and White, M. (1988). Simulated annealing using the mean field approximation. *IEEE Conference on Neural Information Processing Systems*, Denver, November.

Blake, A. (1989). Comparison of the efficiency of deterministic and stochastic algorithms for visual reconstruction. *IEEE Trans. Pattern Anal. Machine Intell.* **11**, 2–12.

Cross, G. C. and Jain, A. K. (1983). Markov random field texture models. *IEEE Trans. Pattern Anal. Machine Intell.* **5**, 25–39.

Derin, H. and Won, C. S. (1987). A parallel image segmentation algorithm using relaxation with varying neighborhoods and its mapping to array processors. *Comput. Vision. Graphics and Image Process.* **40**, 54–78.

Geman, D. and Geman, S. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.* **6**, 721–741.

Gidas, B. (1989). A renormalization group approach to image processing problems. *IEEE Trans. Pattern Anal. Machine Intell.* **11**, 164–180.

Greig, D. M., Porteous, B. T. and Seheult, A. H. (1989). Exact maximum a posteriori estimation for binary images. *J. Roy. Statist. Soc. Ser.B* **51**, 271–279.

Huang, T. S., Yang, G. J. and Tang, G. Y. (1979). A fast two-dimensional median filtering algorithm. *IEEE Trans. Acoust., Speech, Signal Process.* **27**, 460–464.

Jain, A. K. (1989). *Fundamentals of Digital Image Processing.* Prentice Hall, England Cliffs, New York.

Johnson, V. E., Wong, W. H., Hu, X. and Chen, C. (1989). Statistical aspects of image restoration. Technical Draft, Department of Statistics, University of Chicago.

Jubb, M. and Jennison, C. (1988). Aggregation and refinement in binary image restoration. School of Mathematical Sciences, Technical Report, University of Bath, Bath U.K.

Kent, J. T. and Maria, K. V. (1988). Spatial classification using fuzzy membership models. *IEEE Trans. Pattern Anal. Machine Intell.* **10**, 659–671.

Manbeck, K. (1990). Bayesian statistical methods applied to emission tomograph with physical phanton and patient data. Ph.D. Thesis, Brown University.

Marroquin, J. L. (1985). Probabilistic solution of inverse problems. Technical Report 860, Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

Owen, A. (1986). Discussion of Ripley (1986). *Canad. J. Statist.* **14**, 106–110.

Owen, A. (1989). Image segmentation via iterated conditional expectations. Technical Report, Department of Statistics, University of Chicago.

Ripley, B. D. (1986). Statistics, images, and pattern recongnition. *Canad. J. Statist.* **14**, 83–111.

Department of Epidemiology and Public Health, Yale University School of Medicine, New Haven, CT 06510, U.S.A.

(Received January 1991; accepted August 1992)