Bayesian Image Classification with Baddeley's Delta Loss

Arnoldo Frigessi
Dipartimento di Matematica
Terza Università di Roma, Italy

Håvard Rue*
Department of Mathematical Sciences
N.T.H., Norway

May 4, 1995

Abstract

In this paper we adopt Baddeley's delta metric (Baddeley 1992) as a loss function in Bayesian image restoration and classification. We develop a new algorithm that allows to estimate the corresponding optimal Bayesian estimator, for which good practical estimates can be obtained by approximately the same cost as traditional estimators like the Marginal Posterior Mode (MPM). A comparison of such classification with MPM shows significant advantages, especially with respect to fine structures.

Keywords: Bayesian inference; Unsymmetric loss functions; Image restoration; Markov chain Monte Carlo methods; Metropolis algorithm; Distance between binary images.

*Address for correspondence: Department of Mathematical Sciences, The Norwegian Institute of Technology, N–7034 Trondheim, Norway. E–mail: havard.rue@imf.unit.no.

1 Introduction

In image analysis one must often compare two different images. For instance, consider the problem of restoring a noisy image. For this we develop a new method and we would like to illustrate its properties by means of a simulation study. In this case the true image is known and a degredated version of it is generated and processed by the new algorithm, which outputs an approximation of the true image. Finally we *compare* the approximated image with the true one, and hope that differences are small, or better, that some objective quantitative distance between the two images is small. Similarly we compare the output images of two competitive restoration algorithms in order to understand different merits and weeknesses.

Computer scientists and engineers especially have proposed many different ways to measure the discrepancy between images. We refer to the first four sections of Baddeley (1992) for a recent summary of various metrics on the space of all possible images. For example, consider a finite lattice Λ with n sites and two images $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{x}' = (x'_1, x'_2, \ldots, x'_n)$. A very commonly used measure of discrepancy between $\mathbf{x}$ and $\mathbf{x}'$ is the mean squared error

$$d(\mathbf{x}, \mathbf{x}') = \frac{1}{n} \sum_{i \in \Lambda} (x_i - x'_i)^2.$$

Statisticians became also interested in image restoration, which can be rephrased in inferencial terms. The Bayesian approach has been particularly successful (see Besag, Green, Higdon, and Mengersen (1995) for a recent source of literature). A prior distribution is chosen to represent our informations on some characteristics of the true image, such as local homogeneity, textures etc. Furthermore a likelihood is assumed to model the noisy data acquisition. The posterior distribution is the main tool for inference, that proceeds with the choice of a loss function and the minimization of the posterior expected risk. This identifies the optimal Bayesian estimator (OBE) for the true unknown image.

Let $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ be the random vector representing the true image distributed according to a prior measure $P(\mathbf{x})$ and let $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_n)$ be the observed image. Denote by $P(\mathbf{y}|\mathbf{x})$ the likelihood and let $\pi(\mathbf{x}) = P(\mathbf{X} = \mathbf{x}|\mathbf{y}) \propto P(\mathbf{x})P(\mathbf{y}|\mathbf{x})$ be the posterior distribution. A *loss function* $l(\mathbf{x}, \hat{\mathbf{x}})$ is then chosen and the expected posterior loss is

$$\mathrm{E}_\pi \left[l(\mathbf{X}, \hat{\mathbf{x}}) \right].$$

This function has then to be minimized with respect to $\hat{\mathbf{x}}$ to obtain the OBE.

Markov random fields are particularly suited as prior measures. There is now a certain experience in the design of such prior models. Much less work has been done in terms of design of loss functions. In practice only a few are used, among which the mean sum of squared differences

$$l(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{n} \sum_{i \in \Lambda} (x_i - \hat{x}_i)^2. \tag{1}$$

One reason for this is that only for some loss functions one is able to determine the form of the OBE analytically. For example, in the above case, one easily gets as the optimal estimator the sitewise marginal posterior mean $\mathrm{E}_\pi[X_i]$. Computing $\mathrm{E}_\pi[X_i]$ is a hard computational task and an approximation of it can be obtained by running a Markov Chain Monte Carlo (MCMC) method with invariant measure π. Incidently, especially econometricians and engineers have considered the problem of basing decisions according to some formal utility function. As an example of such decision theoretic approaches we mention (Chicán 1991), where specific utility functions for technological and complex problems are discussed and applied.

It is evident that a loss function has a similar role as a discrepancy measure used to compare the quality of different restorations. In particular, if it seems appropriate to asses the performance of an algorithm by mean of a certain measure $d(\cdot,\cdot)$, then the same measure should be considered also as the loss function; hence the OBE will be deduced with respect to it. It seems to us also a question of coherence to maintain the same metric both in the planning of the restoration method and in assessing its virtues.

Recently Rue (1994, 1995) has considered the problem of obtaining good approximations for the OBE corresponding to more general loss functions. Essentially, one has to solve a highly dimensional nonlinear minimization problem by means of numerical methods, like Simulated Annealing, additionally to the MCMC algorithm needed to approximate posterior moments. Although computationally quite demanding, the procedure is feasible. Rue has shown how the use of certain loss functions, that penalize the aggregation of errors, gives rise to better image restorations.

In this paper we develop the OBE for a loss function recently proposed in Baddeley (1992) and termed *delta metric*, which is suitable for few–color images and especially binary images. The metric has many important properties and seems to override many other popular discrepancy measures in image analysis.

Here is the plan of the paper. In the next section we recall from Baddeley (1992) the definition of delta metric and describe some properties. Section 3 contains the description of the OBE when the delta metric is taken as the loss function. We give an algorithm, using simulated annealing, by means of which the estimator can be approximated. In Section 4 we propose, somehow similarly to Ferrari, Frigessi, and Gonzaga de Sá (1995), a fast algorithm that is based on the fact that certain local conditions are sufficient for fixing the pixel estimate $\hat{x_i}$ correctly, without the need of any annealing or such. The final session is devoted to simulation experiments that confirm that the delta-metric-based OBE often performs better than common estimators based on other loss functions.

2 Baddeley's Delta Metric

Consider a finite lattice Λ equipped with a metric ρ. For instance $\rho(i, j)$ is the length of the shortest path between sites i and j on the lattice, or the Euclidean distance between sites i and j on the Euclidean space embedding the lattice Λ. We will assume without loss of generality that the smallest distance between two distinct sites is normalized to 1. Consider first the set of all possible *binary* images on Λ,

$$\Omega = \{\mathbf{x} : x_i \in \{-1, +1\}, i = 1, 2, \dots, n\},$$

and define for every image $\mathbf{x} \in \Omega$ the corresponding set of *black* sites

$$B(\mathbf{x}) = \{i \in \Lambda : x_i = +1\}.$$

Binary images can be uniquely identified by their black sites set. The distance of a site j from a set $A \subseteq \Lambda$ is

$$d(j, A) = \min_{a \in A} \rho(j, a).$$

Consider a function w defined on $[0, \infty]$, strictly increasing, $w(t) = 0$ if $t = 0$, and that is concave, i.e. $w(s + t) \leq w(s) + w(t)$. Given two binary images $\mathbf{x}$ and $\mathbf{y}$, Baddeley (1992) define their *delta* distance

$$\Delta_w^p(\mathbf{x}, \mathbf{y}) = \left(\frac{1}{n} \sum_{i \in \Lambda} |w\{d(i, B(\mathbf{x}))\} - w\{d(i, B(\mathbf{y}))\}|^p \right)^{1/p}, \tag{2}$$

for any $1 \leq p < \infty$. We assume that w is eventually constant, i.e. there exists a value t_c such that $w(t) = c$ for any $t \geq t_c$. The contributions to (2) is then zero for sites further than c units away from $B(\mathbf{x})$ and $B(\mathbf{y})$. This has also the attractive property that the value of $w\{d(i, B(\mathbf{x}))\}$ does not change if we enlarge the lattice size. We assume for simplicity

$$w(t) = \min\{t, c\} \tag{3}$$

and we will study the variability of our estimates also in terms of c. We use the notation Δ_c^p in this case.

Baddeley (1992) shows that Δ_w^p is a metric on the set of all binary images that generates the so called hit-or-miss or *myopic topology*. This is an important property in the context of image analysis described in Serra (1982).

What happens for $c = 1$? Consider p integer and even. If $w(t) = \min\{t, 1\}$ then

$$(\Delta_c^p(\mathbf{x}, \mathbf{y}))^p = \frac{1}{n} \sum_{i \in \Lambda} |\mathbf{1}_{\{x_i \neq +1\}} - \mathbf{1}_{\{y_i \neq +1\}}|^p = \frac{1}{n} \sum_{i \in \Lambda} \mathbf{1}_{\{x_i \neq y_i\}}, \tag{4}$$

which is the misclassification error corresponding to (1) in the case of binary images. This fact allows us to find the usual Marginal Posterior Mode (MPM) optimal Bayes estimator as a particular case. For larger values of c the loss function deviates more and more from the misclassification error. For simplicity, let $\Lambda \in Z$, $p = 2$ and $c = 2$. One can expand

$$w\{d(i, B(\mathbf{x})\} = \mathbf{1}_{\{x_i \neq +1\}} + \mathbf{1}_{\{x_i \neq +1\}} \cdot \mathbf{1}_{\{x_{i+1} \neq +1\}} \cdot \mathbf{1}_{\{x_{i-1} \neq +1\}}, \tag{5}$$

so that

$$|w\{d(i, B(\mathbf{x}))\} - w\{d(i, B(\mathbf{y}))\}|^2 = \mathbf{1}_{\{x_i \neq y_i\}} + \text{rest}.$$

The *rest* consists of terms of the type

$$\mathbf{1}_{\{x_i \neq +1\}} \cdot \mathbf{1}_{\{x_{i+1} \neq +1\}} \cdot \mathbf{1}_{\{x_{i-1} \neq +1\}} \cdot \mathbf{1}_{\{y_i \neq +1\}}$$

and

$$\mathbf{1}_{\{x_i \neq +1\}} \cdot \mathbf{1}_{\{x_{i+1} \neq +1\}} \cdot \mathbf{1}_{\{x_{i-1} \neq +1\}} \cdot \mathbf{1}_{\{y_i \neq +1\}} \cdot \mathbf{1}_{\{y_{i+1} \neq +1\}} \cdot \mathbf{1}_{\{y_{i-1} \neq +1\}}$$

so that $\left(\Delta_2^2(\mathbf{x}, \mathbf{y})\right)^2$ can be split into a term which is the misclassification error and a part that takes into account the aggregation of discrepancies. This is a further desired property of a good loss function. Note that the interaction terms in the delta loss function are built in such a way that aggregation of errors in clusters up to range $2c - 1$ is penalized. Similar observations are valid for Λ in higher dimensions and other even values of p. (On passing, note that non-integer values of p and odd ones, although possible, do not seem to have a motivation in this setting.) An important aspect to be noted is the lack of *symmetry* in the way of considering $+1$ and -1. The *rest* above cannot be written in terms of

$$\mathbf{1}_{\{x_i \neq y_i\}} \cdot \mathbf{1}_{\{x_{i+1} \neq y_{i+1}\}}, \tag{6}$$

which would imply a symmetric treatment of the two colors, and that is typically the main tool in designing standard loss functions, see Rue (1995). Of course the lack of symmetry of the delta loss is evident from the definition itself, which measures the discrepancies of the set of black sites of two images. This attitude is particularly important in certain applications, for instance in classification, where the aim is to localize a particular class, say a tumor in medical imaging or corn fields in remote sensing, and one is essentially indifferent to the quality of the classification of other types. In any case one can easily symmetrize the delta loss function.

The delta loss function can be extended to images with more than two colors. First observe that one can keep the delta loss function based on the discrepancy of the black sites sets, if their location is the only scope of the image processing task. If more than one type is of interest, one can proceed as follows. Consider images $\mathbf{x}$ with χ colors $\{1, 2, \ldots, \chi\}$ and let

$$B_k(\mathbf{x}) = \{i \in \Lambda : x_i = k\}$$

be the set of pixels having color k in $\mathbf{x}$. Such images can be uniquely identified by the sets $B_1, B_2, \ldots, B_{\chi-1}$. If $\{1, 2, \ldots, \kappa\}$ are the colors of interest, then define

$$\Delta_w^p(\mathbf{x}, \mathbf{y}) = \left(\frac{1}{\kappa n} \sum_{i \in \Lambda} \sum_{k=1}^{\kappa} |w\{d(i, B_k(\mathbf{x}))\} - w\{d(i, B_k(\mathbf{y}))\}|^p \right)^{1/p}, \tag{7}$$

for any $1 \leq p < \infty$. Note that one can symmetrize the loss function by putting $\kappa = \chi$. This means that a reordering of the colors would not affect the OBE. However, maybe surprisingly, also the symmetrized loss function (7) cannot be expanded in misclassification terms of type (6). One way to construct a loss function based on such terms by means of the delta loss is as follows. Given two images $\mathbf{x}, \mathbf{y}$ let $e_i = +1$ if $x_i \neq y_i$ and $e_i = -1$ otherwise. The binary image $\mathbf{e}$ identifies the misclassified pixels. We need also $\mathbf{b}$, the all black image, $b_i = +1$. Then

$$(\Delta_c^p(\mathbf{e}, \mathbf{b}))^p = \frac{1}{n} \sum_{i \in \Lambda} |w\{d(i, B(\mathbf{e}))\}|^p$$

can be expanded in terms of type (6) and penalizes the aggregation of misclassification errors. Using the all white image is also useful when objects need to be localized in images (Rue 1995). Although putting $\kappa = \chi$ in (7) can be useful also for binary images, when both colors are relevant, we proceed now with the unsymmetrized loss function $(\Delta_c^p(\mathbf{x}, \mathbf{y}))^p$, as given in (2) and (3) for binary images.

3 Optimal Bayes Estimator

In this section we describe an algorithm to compute the OBE with respect to the delta loss. We apply the ideas in Rue (1995). For some fixed p and weight function w, consider the expected posterior loss

$$\mathrm{E}_\pi \left[(\Delta_w^p(\mathbf{X}, \hat{\mathbf{x}}))^p \right]. \tag{8}$$

We construct an algorithm which minimizes (8) with respect to $\hat{\mathbf{x}}$ in the set of all possible binary images. We first take $p = 2$ and expand the square so that we get

$$\begin{aligned} \mathrm{E}_\pi \left[\{\Delta_c^2(\mathbf{X}, \hat{\mathbf{x}})\}^2 \right] = & \; \frac{1}{n} \sum_{i \in \Lambda} \mathrm{E}_\pi \Big[\{w(d(i, B(\mathbf{X})))\}^2 + \{w(d(i, B(\hat{\mathbf{x}})))\}^2 \\ & - \; 2w(d(i, B(\mathbf{X}))) \, w(d(i, B(\hat{\mathbf{x}}))) \Big]. \end{aligned}$$

Forgetting about the constant parts, we have to minimize

$$L(\hat{\mathbf{x}}) = \sum_{i \in \Lambda} \Big(\{w(d(i, B(\hat{\mathbf{x}})))\}^2 - 2w(d(i, B(\hat{\mathbf{x}}))) \, \mathrm{E}_\pi \left[w(d(i, B(\mathbf{X}))) \right] \Big). \tag{9}$$

The problem is hard for two reasons: (i) one has to compute or estimate the posterior expected values of $w(d(i, B(\mathbf{X})))$, i.e. the average posterior distance of pixel i to the closest black point; (ii) given such expected values, one has to solve a global minimalization problem over a very large (discrete) space. Let us first consider this second step. Assume for a while that

$$\mathrm{E}_\pi \left[w(d(i, B(\mathbf{X}))) \right] \tag{10}$$

are known exactly for every pixel i, and denote them μ_i. (Note that μ_i depends of c.) Then one can solve the minimization problem for $L(\hat{\mathbf{x}})$ by running a *Simulated Annealing* algorithm.

More precisely, start with some initial image $\hat{\mathbf{x}}_0$ and run a Metropolis Markov chain with a decreasing temperature schedule T_t. At each step choose a pixel and suggest for it a new color. Thinking in terms of the black sites set, if $B(\hat{\mathbf{x}}_t)$ is the set available at iteration t, choose a pixel uniformly, say i and propose the set $B\left(\hat{\mathbf{x}}^{(i)}\right)$ that coincides with $B\left(\hat{\mathbf{x}}_t\right)$ except for pixel i, that has left the set if it was included in it, or has entered it if it was excluded. Finally accept the move with probability

$$\min\left\{1, \exp\left(\frac{1}{T_t}(L(\hat{\mathbf{x}}_t) - L(\hat{\mathbf{x}}_t^{(i)}))\right)\right\}. \tag{11}$$

In order to assure convergence of $\hat{\mathbf{x}}_t$ to the global minimizer of (10), which we shall denote by $\mathbf{x}^*$, we have to take the slow cooling schedule $T_t = T_0/\log(t+1)$. We refer the reader to Gidas (1991) for background on Simulated Annealing.

Three remarks on this algorithm. First, when computing the sign of the exponent in (11), one should notice that only very few terms in the sum have to be considered if we keep a record of the current values of $w(d(i, B(\cdot)))$. This is illustrated with the pseudocode for the simulated annealing algorithm in the Appendix. Second, Baddeley suggest to use the distance transform algorithm of Rosenfeld and Borgefors (see Borgefors (1986)) to speed up the computation of $d(j, B(\cdot))$. Third, such a simulated annealing algorithm is still slow in practice. One has to use a faster cooling schedule, for instance $T_t = T_0\, 0.99^t$, but this does not guarantee global convergence anymore, although it still allows to obtain reasonable results in our context.

We have to discuss the difficulty (i), i.e. the calculation of

$$\mu_i = \mathrm{E}_\pi\left[w(d(i, B(\mathbf{X})))\right], \quad \forall i \in \Lambda, \tag{12}$$

which are needed before we run the simulated annealing chain. Computing such moments exactly is almost always very hard, if the posterior measure is non trivial and shows some correlation among pixel variables. A practical way to obtain approximate values is to run an MCMC chain converging to π. Denote by $\mathbf{X}(t)$ a discrete time Markov chain on state space Ω, irreducible and weakly converging, as t goes to infinity, to the posterior measure π. An empirical estimator of μ_i is, for $i \in \Lambda$, $N_0 \geq 0$,

$$\hat{\mu}_i = \frac{1}{N}\sum_{t=N_0+1}^{N_0+N} w(d(i, B(\mathbf{X}(t)))). \tag{13}$$

(Refer to the literature list in Besag, Green, Higdon, and Mengersen (1995) on MCMC methods and its properties.) Of course, for $N \to \infty$, we have that $\hat{\mu}_i \to \mu_i$, and see Geyer (1992, Section 2) for a central limit theorem for (13). For N_0 and N large enough we are allowed to consider $\hat{\mu}_i$ as a good approximation of μ_i. There is progress on rates of convergence of MCMC's, refer to Frigessi, Martinelli, and Stander (1993), Mengersen and Tweedie (1995) and Rosenthal (1994) and reference therein on criteria for choosing N_0 and N properly. Good approximations of the posterior moments (12) are needed, otherwise the simulated annealing step can lead off from $\mathbf{x}^*$. In our implementation of the MCMC we run a Metropolis algorithm, but a Gibbs Sampler could also be used. In this latter case one may decide to use different approximations of μ_i, as suggested in the recent paper by Greenwood, McKeague, and Wefelmeyer (1995)

Summarizing, in order to minimize (9), we proceed as follows: (i) we run an MCMC long enough to get good estimates $\hat{\mu}_i$ of μ_i; (ii) we run a simulated annealing algorithm in order to minimize

$$\widehat{L(\hat{\mathbf{x}})} = \sum_{i \in \Lambda} \left(\{w(d(i, B(\hat{\mathbf{x}})))\}^2 - 2w(d(i, B(\hat{\mathbf{x}})))\hat{\mu}_i \right).$$

We thus obtain a binary image $\hat{\mathbf{x}}^*$ which can be considered as a good approximation of $\mathbf{x}^*$, if both steps (i) and (ii) have been run sufficiently long to assure convergence.

For higher values of p the expansion of $\mathrm{E}_\pi\left[\{\Delta_c^p(\mathbf{X}, \hat{\mathbf{x}})\}^p\right]$ is more complicated, but of course feasible. One has in any case to estimate posterior moments of $w(d(i, B(\mathbf{X})))$. For instance, for $p = 4$, one has to estimate

$$\mathrm{E}_\pi\left[w(d(i, B(\mathbf{X})))\right], \; \mathrm{E}_\pi\left[(w(d(i, B(\mathbf{X}))))^2\right], \; \mathrm{E}_\pi\left[(w(d(i, B(\mathbf{X}))))^3\right].$$

One MCMC chain is enough to give reasonable estimates of such moments if it is run long enough. The simulated annealing step is also affected by the choice of a higher value for p in the sense that the function to be minimized will look more complex than (9). But this difference is not very relevant in terms of computing time. We can say that our algorithm is able to treat every value of p, but it will need more time to converge for higher values of p. We should also notice the effect on the computation time of the choice of the cutoff value c in (3): increasing c slows down every step of the algorithm, since more terms (in an expression similar to (9)) have to be considered both in step (i) and (ii).

We compare the computational difficulty of the Bayesian estimation based on the delta loss, with the computation of other usual Bayesian estimators. Consider for instance the misclassification loss function (4); traditionally, in order to compute its corresponding OBE one proceeds as follows. First the posterior expected loss is minimized analytically, something that can be done because the loss acts on every pixel individually. This gives the OBE at pixel i as the color that maximize the marginal posterior probability, and a MCMC with target π is run to approximate these modes. Our algorithm also needs an MCMC run to estimate moments, so it is not different with this respect. It is however slower because a further minimization has to be done numerically by simulated annealing. In dealing with general loss functions, it turns out that one has to first estimate certain moments and then minimize the (expanded) posterior risk. With classical simple loss functions the procedure is reversed, since first the expected risk is minimized and then moments are estimated. Although we believe that in certain situations the classifications based on the delta loss will be better enough to motivate in any case a higher computational cost, we shall propose in the next section another algorithm able to classify many pixels correctly in very short time, so that only on the remaining ones one has to run numerical minimization routines.

A further vague observation. Look back to (9) and imagine to take derivatives with respect to $\hat{\mathbf{x}}_i$ (which is of course illegal!). We end up with a sort of method of moment equation where the OBE $\hat{\mathbf{x}}^*$ is such that $d(i, B(\hat{\mathbf{x}}^*))$ balances the exact moments μ_i. Really this does not need to happen term by term, but it indicates that the estimator is indeed quite reasonable.

Baddeley suggests also other forms for the function w, for instance $w(t) = t/(1+t)$. In this case the function w is not eventually constant over a finite range. Our algorithm can handle also this situation; however the full density of $d(i, B(\mathbf{X}))$ for all pixel i has to be estimated, not only moments. A serious storage problem arises in this case. Practically one needs to keep w constant after a certain point. The delta loss function with w as in (3) is particularly suited for Bayesian applications, because it requires the estimation of simple moments. Consider for example the Pratt's "figure of merit" distance (Rosenfeld and Kak 1982), defined as

$$\mathrm{FOM}(\mathbf{x}, \mathbf{y}) = \frac{1}{\max\{|B(\mathbf{x})|, |B(\mathbf{y})|\}} \sum_{i \in B(\mathbf{y})} \frac{1}{1 + \alpha d(i, B(\mathbf{x}))},$$

(where α is usually set to 1/9). In this case the OBE can be estimated by our algorithm, but full density estimation of $d(i, B(\mathbf{X}))$ is indeed needed. Similar comments apply for the Hausdorff metric used as a loss function. Really, for finite images there is always the natural truncation point for w given by the diameter of the image itself. Finally, the presented algorithm is able to produce (approximated) OBE also with respect to loss functions of type $\{\Delta_w^p(\mathbf{x}, \mathbf{y})\}^p$, for multicolor images. Again only one MCMC chain needs to be run to estimate all needed moments.

4 Sufficient Local Conditions Characterizing the Optimal Bayesian Estimator

Our aim is now to determine conditions on the posterior moments $\hat{\mu}_i, \forall i \in \Lambda$, given in (10), that: i) can be checked very rapidly, and ii) allow us to decide the value of pixel variables of the estimated OBE $\hat{\mathbf{x}}^*$ without running a simulated annealing algorithm. In this section we assume that the preliminary MCMC run (i) has been performed and has given us reliable estimates of the moments μ_i. We write μ_i instead of $\hat{\mu}_i$, for simplicity, but we come back to the fact that in reality we have only approximated moments $\hat{\mu}_i$ of (10).

In order to explain how our sufficient conditions are build in general, we first consider the case $w(t) = \min(t, 1)$, that correspond, as mentioned in Section 2, to the misclassification error which leads to marginal posterior modes as optimal pixel restorations. For this, one has to run an MCMC algorithm in order to estimate posterior probabilities of type $\pi(X_i = -1)$ which is of course the same as estimating μ_i for binary images and $c = 1$. The MPM estimator is convenient expressed in the present case as

$$\text{if} \quad \pi(X_i = -1) < 1/2, \quad \text{then} \quad x_i^* = +1, \tag{14}$$

$$\text{if} \quad \pi(X_i = -1) > 1/2, \quad \text{then} \quad x_i^* = \text{-}1. \tag{15}$$

We now reformulate (15). For each pixel i consider the contribution to (9). If $i \in B(\mathbf{x}^*)$ then

$$\{w(d(i, B(\mathbf{x}^*)))\}^2 - 2w(d(i, B(\mathbf{x}^*)))\mu_i = 0$$

while if we put $i \notin B(\mathbf{x}^*)$ then we have

$$\{w(d(i, B(\mathbf{x}^*)))\}^2 - 2w(d(i, B(\mathbf{x}^*)))\mu_i = 1 - 2\mu_i$$

Since $c = 1$ and the smallest distance among two sites is 1, we need to compare just these two numbers in order to decide about the color of i in the optimal restoration:

$$\begin{aligned} &\text{if} \quad \mu_i > \frac{1}{2} \quad \text{then} \quad \mathbf{x}_i^* = -1, \quad \text{and} \\ &\text{if} \quad \mu_i < \frac{1}{2} \quad \text{then} \quad \mathbf{x}_i^* = +1. \end{aligned} \tag{16}$$

We interpret the check of the sign of $\mu_i - \frac{1}{2}$ as a sufficient condition that allows to decide about the color of pixel i in $\mathbf{x}^*$. (The condition is also necessary if $c = 1$.) In this example we see what we mean by "the condition can be checked locally": it involves only the moment μ_i relative to pixel i in order to decide about $\mathbf{x}_i^*$. For higher values of c we will be satisfied with conditions that, for each pixel, only use some μ_j for j in a local neighborhood of i for the decision concerning $\mathbf{x}_i^*$.

We define for every $i, j \in \Lambda$

$$r(i,j) = \min_{k \in \Lambda} \{\rho(i,k) : \rho(i,k) > \rho(i,j)\}.$$

If Λ is a finite subset of Z^2 and ρ is the shortest path metric, then $r(i,j) = \rho(i,j) + 1$. It is also useful to define the range of the loss function, for every $i \in \Lambda$ let

$$\partial(i) = \{j \in \Lambda : \rho(i,j) < c\}$$

be the range of $\{\Delta_c^p\}^p$.

We are ready for the first set of conditions allowing us to decide about $\mathbf{x}^*$ locally. We focus on $p = 2$ but extension to higher values of p and different w's are possible.

Theorem 1 *Let $\mathbf{x}^* \in \Omega$ be the optimal Bayesian estimator minimizing L as given in (9). For every site $i \in \Lambda$ the following conditions hold:*

1. *If for every $j \in \partial(i)$*

$$\mu_j \leq \frac{1}{2} (\rho(i,j) + \min\{c, r(i,j)\}) \tag{17}$$

then $\mathbf{x}_i^* = +1$.

2. *If for every* $j \in \partial(i)$

$$\mu_j \geq \frac{1}{2}(\rho(i,j)+c) \tag{18}$$

then $\mathbf{x}_i^* = -1$.

Proof: Let $\mathbf{x}^+$ and $\mathbf{x}^-$ be two configurations in Ω that differ only in pixel i, where they take value $+1$ and -1 respectively. By comparing L term by term we prove that $L(\mathbf{x}^+) \leq L(\mathbf{x}^-)$ when (17) is fulfilled. For simplicity abbreviate

$$d_j^+ = w(d(j, B(\mathbf{x}^+))) \quad \text{and} \quad d_j^- = w(d(j, B(\mathbf{x}^-))). \tag{19}$$

For every $j \in \Lambda$ we show that

$$d_j^+(d_j^+ - 2\mu_j) \leq d_j^-(d_j^- - 2\mu_j).$$

If $d_j^+ = d_j^-$, then this is trivially true. This is the case for every $j \notin \partial(i)$, because the color of site i cannot influence $d_j^\cdot$. Assume $d_j^+ \neq d_j^-$, which is the same as $d_j^+ \leq d_j^-$, because of the unsymmetric treatment of black and white colored sites. Of course $d_j^+ \leq \rho(i,j)$, but now it precisely holds $d_j^+ = \rho(i,j)$ since otherwise $d_j^+ = d_j^-$. From (19) we obtain

$$\mu_j \leq \frac{1}{2}(d_j^+ + d_j^-)$$

where the right hand side certainly is larger than

$$\frac{1}{2}(\rho(i,j) + r(i,j)),$$

for all possible colors of $\mathbf{x}$ outside site i. The proof under condition (18) runs in the same way, but in order to decide for -1 at pixel i an upper bound on d_j^- is needed, and we take simply c. $\square$

To illustrate this result assume $c = 2$, take Λ as the one dimensional lattice $\{1, \ldots, n\}$ (with free boundary conditions, say) and assume $\rho(i,j) = |i-j|$. The local conditions for a generic site i are

$$\text{if} \quad \mu_i \leq \frac{1}{2}, \quad \mu_{i+1} \leq \frac{3}{2}, \quad \mu_{i-1} \leq \frac{3}{2}, \quad \text{then} \quad \mathbf{x}_i^* = +1 \tag{20}$$

$$\text{if} \quad \mu_i \geq 1, \quad \mu_{i+1} \geq \frac{3}{2}, \quad \mu_{i-1} \geq \frac{3}{2}, \quad \text{then} \quad \mathbf{x}_i^* = -1. \tag{21}$$

In two dimensions, i.e. for $\Lambda \subset Z^2$, the conditions are the same, but have to be fulfilled by all four first neighbors of site i.

Conditions (17) and (18) are sufficient in order to be able to decide about the color of pixel i of the OBE. Once every pixel has been checked for such condition to hold, we remain with some sites for which these conditions fail to be true; for those sites i we cannot say anything about $\mathbf{x}_i^*$ at this point. Which means that we will need to run a simulated algorithm on such decimated lattice, where we have frozen to their correct color all sites for which indeed (17) or (18) hold. This is related to the method proposed in Ferrari, Frigessi, and Gonzaga de Sá (1995). Consider again the one dimensional lattice example and $c = 2$. By representing (20) and (21) in a cube in 3D with sides of length 2 it follows that there is quite a large volume (in Lebesgue measure) where a decision is not taken. The volume for which $\mathbf{x}_i^* = +1$ can be decided is 9/8, while in the case of $\mathbf{x}_i^* = -1$ it is 2/8 only. Especially this second part (18) of the condition should be improved.

It is important to notice that we need to check only a small set of local conditions regarding few moments μ_j for every i; specifically only for $j \in \partial(i)$. Hence a very rapid sweep of the lattice will do the job. If the undecided sites form separated clusters with a sufficient boundary region of sites with known values, one can proceed in parallel by treating each cluster individually, given correct frozen boundary conditions. The simulated annealing algorithm (ii), can thus be speeded up considerably, by decimative the image and also by subdividing it.

In reality, after the MCMC run of step (i) only estimates $\hat{\mu}_i$ are available, so that (17) and (18) are really checked using these approximations. Note that consistency is assured if the MCMC tends to stability. In any case a too short MCMC run, and hence bad moments estimates, will make the classification based on any loss function questionable.

There is a further advantage of the above outlined algorithm. This is concerned with the fact that one does not need to decide the cutoff point c in advance if we estimate μ_i for those values of c we may want to consider. Checking conditions (17) and (18) is so fast that it can be done for several values of c. In fact let S_i^c be the set of conditions as expressed by (17) above. Then $S_i^c \subseteq S_i^{c'}$ for every $c' \geq c$. Furthermore, the additional conditions present in $S_i^{c'}$ involve only μ_j for j such that $c \leq \rho(i,j) < c'$. The proof is straightforward, by looking back to the proof of Theorem 1, and remembering that $\partial(i)$ is an increasing set in c. Take the shortest path distance on a lattice with unit distance among neighboring sites. Then $S_i^c \subset S_i^{c+1}$. So given conditions S_i^c we just need to add condition (17) and (18) for j such that $\rho(i,j) = c$. These are the points in $\partial(i)$ for $c+1$ but not for c.

Since it is easy to generate local conditions for increasing values of c, we can obtain several images whose pixel have optimal colors $+1$ or -1, or are left undecided, when moments around i do not fulfill local conditions for the specified value of c. There is also a monotonicity structure in the set of sites that are left undecided at step c. Denote by $\Lambda_c^{\surd}$ the set of sites satisfying conditions (17) for a specified c. Then it holds that $\Lambda_{c'}^{\surd} \subseteq \Lambda_c^{\surd}$ for every $c' > c$. This means that only sites resolved under c need to be considered under $c+1$. Increasing c leaves more and more sites undecided (and for $c = 1$ all sites are decided). Hence the final part of the program, i.e. simulated annealing on the undecided sites, becomes more and more computationally demanding. There is a tradeoff between ability in reconstructing details (linked to the higher range of the loss function) and efficiency of the algorithm. In our practical experience, we have been satisfied with values of $c \leq 5$.

In Theorem 1 we have compared *single* terms in $L(\mathbf{x})$ in order to ensure its decrease (under suitable conditions) when putting site i to $+1$ or to -1. In fact it is stronger to compare their sum alltogether. Precisely, we look for conditions on μ_j for $j \in \partial(i)$ such that

$$\sum_{j\in\partial(i)} d_j^+(d_j^+ - 2\mu_j) < \sum_{j\in\partial(i)} d_j^-(d_j^- - 2\mu_j) \tag{22}$$

which assures $\mathbf{x}_i^* = +1$. And similarly, with reversed sign in order to fix site i to -1. (As seen before, deciding about site i will not affect further terms in L.) As compared to Theorem 1, this gives sufficient conditions that are easier to be fulfilled. Unfortunately the proof of such conditions depends on the value of c, the metric ρ and the embedding dimension of the lattice Λ. We have not found a general combinatorial argument and are only able to fully enumerate all local configurations (on $\partial(i)$ union all $\partial(j)$ for j on the boundary of $\partial(i)$) in order to determine that one that produces the most stringent constraints on $\mu_j, j \in \partial(i)$ guaranteeing $\mathbf{x}_i^* = +1$ or $\mathbf{x}_i^* = -1$. Note that also in the proof of Theorem 1 we have looked implicitly also to the color of the sites neighboring the boundary pixel of $\partial(i)$. But in the case of (22) we have to keep them into account explicitly. We state only the simplest possible set of extended conditions

Theorem 2 *Let Λ be a finite subset of Z^2, $c = 2$ and ρ be the shortest path metric. For every $i \in \Lambda$, let i_j, $j = 1, 2, 3, 4$ be its four nearest neighbors. It holds that:*

1. *If*
   - *(a)* $\mu_i \leq 1/2$
   - *(b)* $\mu_i + \mu_{i_j} \leq 2, \quad j = 1, 2, 3, 4$
   - *(c)* $\mu_i + \mu_{i_j} + \mu_{i_k} \leq 7/2, \quad j, k = 1, 2, 3, 4, \quad j \neq k$
   - *(d)* $\mu_i + \mu_{i_j} + \mu_{i_k} + \mu_{i_l} \leq 5, \quad j, k, l = 1, 2, 3, 4, \quad j \neq k \neq l$

   then $\mathbf{x}_i^* = +1$.
2. *If*
   - *(a)* $\mu_i \geq 1$
   - *(b)* $\mu_i + \frac{1}{2}\mu_{i_j} \geq 7/4, \quad j = 1, 2, 3, 4$
   - *(c)* $\mu_i + \frac{1}{2}(\mu_{i_j} + \mu_{i_k}) \geq 5/2, \quad j, k = 1, 2, 3, 4, \quad j \neq k$
   - *(d)* $\mu_i + \frac{1}{2}(\mu_{i_j} + \mu_{i_k} + \mu_{i_l}) \geq 13/4, \quad j, k, l = 1, 2, 3, 4, \quad j \neq k \neq l$
   - *(e)* $\mu_i + \frac{1}{2}\sum_{j=1}^{4} \mu_{i_j} \geq 4$

   then $\mathbf{x}_i^* = -1$.

Sketch of proof: Let i be the origin 0, and enumerate its four first neighbors as $1,2,3,4$. We consider first each configuration on $1,2,3,4$. This fixes the left hand side of (22). The other term is fixed once the neighbors (excluding 0) of $1,2,3,4$ are decided. The reader can think of two nested *do* cycles. For example, start with all sites $1,2,3,4$ set to -1; then the left side of (22) is equal to $\sum_{k=1}^{4}(1-2\mu_k)$ with obvious notation. The right side can be equal to $2(2-2\mu_0)$ if all neighbors of $1,2,3,4$ are $+1$. In this first case we get condition $\mu_0 \leq \frac{1}{2}$ assuring $\mathbf{x}_0^* = +1$ as the better choice, $\mu_0 \geq 1$ in the other case. We start to collect conditions. Now we proceed and put to -1 at neighboring site of, say, 1, which is not a neighbor of the other three sites $2,3,4$. Computing again from (22) we get the condition $\mu_0 + \mu_1 \leq 2$, and $\mu_0 + \mu_1/2 \geq 7/4$ for the two cases. Etc., until we get a condition involving all μ_k, $k = 0,1,2,3,4$. Then we take the next configuration of $1,2,3,4$, that one with one $+1$ and three -1's. And we perform the inside *do*–loop again. Here we get a set of conditions involving only either μ_0, or pairs (μ_0,μ_k), triplets etc. We compare the conditions found with the previous ones. We are interested in the intersection. It turns out that the conditions involving the pairs are more stringent than those obtained starting from $1,2,3,4$ all equal to -1. While the first condition obtain for μ_0 will survive all loops. In the next step we will find that the condition involving the triplets will be relevant. And the reader will allow us to stop here: the proof is basically trivial, just very pedantic.

As already clear in Theorem 1, conditions regarding $+1$ (the "object") and -1 (the "background") are different. In particular the sufficient conditions for the background given in Theorem 2 are now improved more than those for the object in the image.

For larger values of c the enumeration technique becomes soon unfeasible. We have developed another easier method that allows to obtain sufficient conditions that are better than those in Theorem 1, although in general weaker than those of Theorem 2. These conditions are based on (automatic) enumeration of all configurations on $\partial(i)$ only. This is important for higher values of c. Without going into details, here is a sketch of this method. For every $\mathbf{x} \in \Omega$ let $l_i(\mathbf{x})$ be the set of sites $j \in \Lambda$ whose corresponding d_j^+ and d_j^- change if x_i is flipped. Of course $l_i(\mathbf{x}) \subset \partial(i)$. Checking (22) for every $\mathbf{x}$ is implied by the same check done for every subset s of $\partial(i)$ and every $\mathbf{x}$ whose $l_i(\mathbf{x}) = s$. This gives possibly weaker conditions than optimal ones yet stronger than those obtained by termwise comparison. Note that one needs only to check "legal" sets s, for instance only simply connected ones. By induction on the number of sites in s one can show the following condition for the background. If

$$\mu_i + \sum_{j\in s}\left(1 - \frac{\rho(i,j)}{c}\right)\mu_j > \frac{1}{2c}\left(c^2 + \sum_{j\in s}\left(c^2 - \rho(i,j)^2\right)\right) \tag{23}$$

for every s in $\partial(i)$, then $\mathbf{x}_i^* = -1$ for any metric ρ. For the shortest path metric we can give also the following condition for the object in the image. If

$$\mu_i + \sum_{j\in s}(\min\{c, r(i,j)\} - \rho(i,j))\mu_j < \frac{1}{2}\left(1 + \sum_{j\in s}\left(\min\{c, r(i,j)\}^2 - \rho(i,j)^2\right)\right) \tag{24}$$

for every s in $\partial(i)$, then $\mathbf{x}_i^* = +1$. Note that (23) and (24) reduces to the conditions in Theorem 2 for $c = 2$ and the shortest path metric. In practical experiments we suggest to use condition (17) to check for $\mathbf{x}_i^* = +1$, and (23) to check for $\mathbf{x}_i^* = -1$ if $|\partial(i)| \leq 12$ and condition (18) otherwise.

We end this section with a comment on the number N of undecided sites for which the conditions of Theorem 1 or (23) do not hold. N can be thought of as a random variable, if the data image is considered as randomly generated through prior and likelihood. Similarly as in Ferrari, Frigessi, and Gonzaga de Sá (1995, Theorem 2 in Section 4), we can show that the expected number of undecided sites is still of the same order as the number of sites n of Λ, asymptotically in n: $\mathrm{E}(N) \sim cn$. This is of course disappointing, but, as we show in the next section, in practice a good portion, often around 50% of sites can be fixed through our sufficient conditions.

5 First Experiments

In this section we show two typical classification experiments that show how the OBE based on Baddeley's delta metric gives better results than the more classical MPM. Recall that such an estimator corresponds to the choice $c = 1$ in (11). Our experiments also quantify the advantage of the algorithms presented in Section 4 and show some properties of the undecided sites, which do not satisfy the sufficient conditions.

We consider binary images on a squared 32×32 lattice. Black sites are $+1$, white sites are -1. We assume the Ising model with inverse temperature $\beta > 0$ as prior distribution for $\mathbf{X}$. The likelihood $P(\mathbf{Y}|\mathbf{x})$ is assumed to be sitewise independent Gaussian with standard deviation σ. This gives a posterior distribution

$$\pi(\mathbf{x}) \propto \exp\left(\beta \sum_{i \sim j} x_i x_j - \frac{1}{2\sigma^2} \sum_{i \in \Lambda} (y_i - x_i)^2\right), \tag{25}$$

where the first sum is over nearest neighboring sites.

We first run an MCMC chain converging to the posterior distribution (25) with Metropolis transitions and uniform proposals. From it we estimate the moments μ_i using 500 samples from the Markov chain run for 5000 iterations after a "burn–in" period of 500. We then check the sufficient conditions (17) for the foreground and (23) for the background when $|\partial(i)| \leq 12$ and (18) otherwise. Then a simulated annealing is run on the clusters of undecided pixels. Its temperature is decreased with the schedule $T_t = (1 - 0.01)^t$ ending when $T_t = 0.01$. The MPM estimate is the initial configuration. The computing time is always very reasonable, especially because the Metropolis updates (according to the pseudocode in the Appendix) are very fast.

In our first displayed simulation, we take a true classification which corresponds to the word *is* in a white background. This is shown in Figure 1a. That image is then corrupted by Gaussian noise with $\sigma = 0.5$ and the data are shown in Figure 1b. We take $\beta = 0.5$, because we can display well in this case the difference of the various reconstructions methods. Other values of β shows similar but less evidently behavior: never the OBE based on Baddeley's loss behaves worse than MPM. The MPM estimated classification is displayed in Figure 1c. As expected the classification is quite good, but oversmoothed, in the sense that small details are not restored properly. Next we obtain the OBE based on Baddeley's delta metric for $p = 2$, shortest path distance on the lattice, $c = 3$ and $c = 4$. This is displayed in Figure 1d and Figure 1e. Even the moderate number of 100 samples used to estimate the μ_i's and a fast decreasing temperature schedule yields good results, as show in Figure 1f. We can notice how the new estimator can better restore the fine structure of the letters, which are thought of being of main interest here. In our experiments values of c around 3 give a good improvement with respect to MPM, and are very fast to compute.

Another striking experiment is shown in Figure 2. The true unknown classification is built up by hand (Figure 2a). It has several fine structures, a circle with four rays sitting in a half square. Of course the Ising model is again quite off as a prior, as it often happens in practice. The noise is intense ($\sigma = 0.7$), and from the data image (Figure 2b) the MPM restoration in Figure 2c (we assume $\beta = 0.45$) misses most fine parts: the circle cannot be seen, nor the half square. Using Baddeley's delta metric allows to improve things seriously. Figure 2d displays classifications obtained for $c = 5$ and the shortest path metric, Figure 2e and Figure 2f displays the classifications obtained for $c = \sqrt{5}$ for the Euclidean and shortest path metric. (This value of c gives $|\partial(i)| = 12$ for both metrics.) Notice that the circle is now visible, and so is the half square and the vertical and horizontal lines.

In Figures 2g and 2h we display the situation of the pixels after the sufficient conditions (17) and (23) are applied. The grey pixels are the sites that are left undecided (for which we run the Simulated Annealing algorithm), 48% when the Euclidean distance is adopted, 36% for the shortest path distance case. Also it is very clear how the undecided pixels mimic the foreground structure, in the sense that they appear where the data may suggest the presence of black pixels, but the noise is so high to prevent a local decision. Undecided pixels are present at the border between black and white areas and along fine structure that has been blurred by the noise. So looking to the pattern of undecided pixels can also be informative, since it allows to isolate the hard part of the image classification problem. Possibly more data need to be taken to proceed towards confident decisions in such areas. Variance and credibility bounds should of course be also produced in order to quantify the quality of the estimate.

We have performed many experiments, including the case when the true scene is generated from the Ising prior. In all cases either all restorations behave similarly or the one based on the delta loss are better. The experiments displayed here are not at all exceptions.

References

Baddeley, A. (1992). Errors in binary images and a L^p version of the Hausdorff metric. *Nieuw Archief coor Wiskunde 10*, 157–183.

Besag, J., P. Green, D. Higdon, and K. Mengersen (1995). Bayesian computation and stochastic systems. *Statistical Science 10.*

Borgefors, G. (1986). Distance transformations in digital images. *Computer vision, graphics and image processing 34*, 344–371.

Chicán, A. (Ed.) (1991). *Progress in Decision, Utility and Risk Theory.* Kluwer Academic Publishers.

Ferrari, P. A., A. Frigessi, and P. Gonzaga de Sá (1995). Fast approximate maximum a posteriori restoration of multicolor images. *Journal of the Royal Statistical Society, Series B 57*(3).

Frigessi, A., F. Martinelli, and J. Stander (1993). Computational complexity of Markov chain Monte Carlo methods. Technical Report Quaderno n. 32, Istituto per le applicazioni del calcolo "Mauro Picone", Rome, Italy.

Geyer, C. (1992). Practical Markov chain Monte Carlo (with discussion). *Statistical Science 7*, 473–511.

Gidas, B. (1991). *Metropolis–type Monte Carlo simulation algorithms and simulated annealing.* Trends in Contemporary Probability.

Greenwood, P. E., I. W. McKeague, and W. Wefelmeyer (1995). Outperforming the Gibbs sampler empirical estimator for nearest neighbor random fields. *Preprint.*

Mengersen, K. and R. L. Tweedie (1995, January). Rates of convergence of the Hastings and Metropolis algorithms. Technical report, University of Queensland, Australia.

Rosenfeld, A. and A. C. Kak (1982). *Digital picture processing* (2 ed.). Academic Press.

Rosenthal, J. S. (1994, December). Theoretical rates of convergence for Markov chain monte carlo. Technical report, University of Toronto, Canada.

Rue, H. (1994). An estimator for images with ordered colors. Technical Report Statistics no. 6/1994, Departments of Mathematics, The University of Trondheim, Trondheim, Norway.

Rue, H. (1995, September). New loss functions in Bayesian imaging. *Journal of the American Statistical Association.*

Serra, J. (1982). *Image analysis and mathematical morphology.* Academic press.

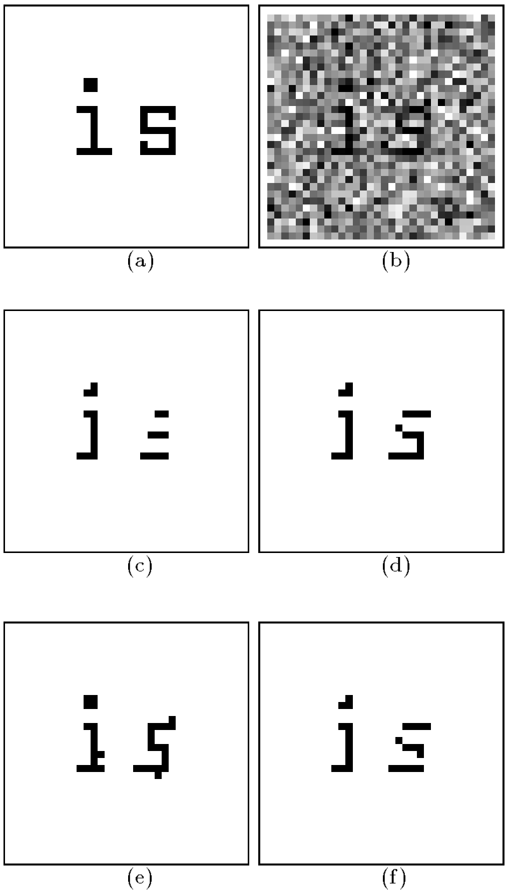

Figure 1: The results from simulation experiment 1. Figure (a) displays the true scene and Figure (b) displays the data when independent Gaussian noise with variance 0.5^2 is added. Figure (c) displays the MPM classification. Figure (d) and (e) display the classification with Baddeley's delta loss with the shortest path metric and using 3 and 4 as values of c. Figure (f) displays the classification where $c = 3$ and both the MCMC and Simulated Annealing step in run for a shorter time to obtain the estimate.

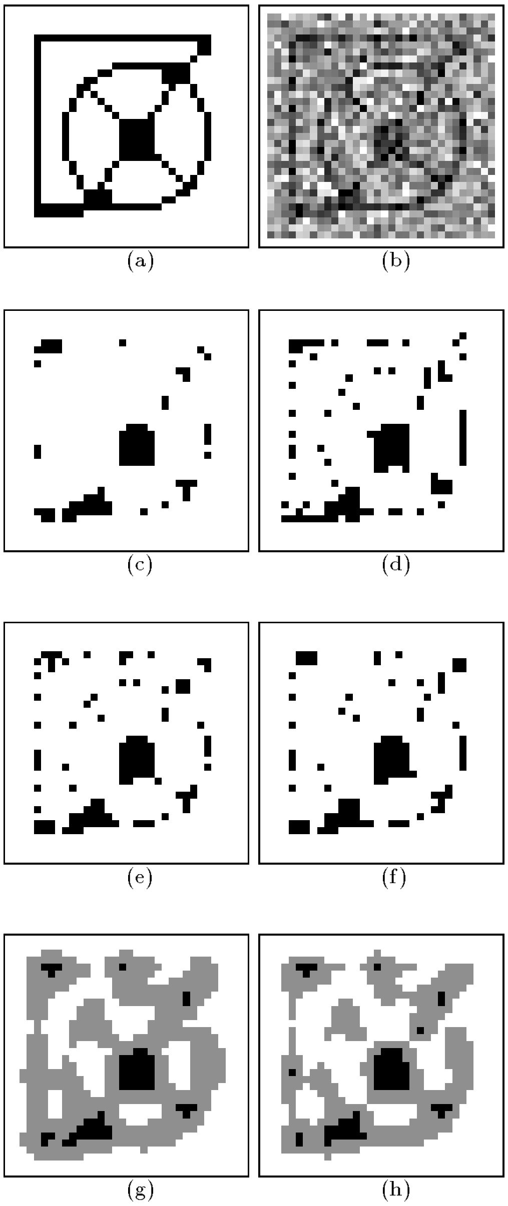

Figure 2: The results from simulation experiment 2. Figure (a) displays the true scene, and Figure (b) displays the data when independent Gaussian noise with variance 0.7^2 is added. Figure (c) displays the MPM estimate and Figure (d) displays the estimate using Baddeley's delta loss with $c = 5$ and the shortest path metric. Figure (e) and (f) display the estimates using $c = \sqrt{5}$ and the Euclidean and shortest path metric, respectively. Figure (g) and (h) displays the results after the sufficient conditions (23) and (17) are checked for the estimates in (e) and (f), respectively. The grey pixels displays the undecided sites (48% and 36%, respectively), for which the Simulated Annealing algorithm is run.

A Pseudocode for computing $\hat{\mathbf{x}}$

Here is a pseudocode for computing the Bayes estimate $\hat{\mathbf{x}}$ using the simulated annealing algorithm and the Metropolis update. It shows how easy it is to compute $L(\hat{\mathbf{x}}_t) - L(\hat{\mathbf{x}}_t^{(i)})$ if we keep a record of the current values of $w(d(j, B(\hat{\mathbf{x}}_t)))$.

$x \longleftarrow x_{\text{initial}}$
$T \longleftarrow T_0$
for all i **in** Λ **do** $d_i \longleftarrow \min\{c, d(i, B(x))\}$
while $(T > T_{\text{final}})$ **do**
  for all i **in** Λ **do**
    $U_{\text{diff}} \longleftarrow 0$
    $\partial(i) \longleftarrow \{j \in \Lambda : \rho(i,j) < c\}$
    for all j **in** $\partial(i)$ **do**
      if $(x_j$ **equal** -1 **and** $d_j > \rho(i,j))$ **then**
        $d_j^{\text{move}} \longleftarrow \rho(i,j)$
      elseif $(x_i$ **equal** $+1$ **and** d_j **equal** $\rho(i,j))$ **then**
        $x_i \longleftarrow -1$
        $d_j^{\text{move}} \longleftarrow \min\{c, d(j, B(x))\}$
        $x_i \longleftarrow +1$
      else
        $d_j^{\text{move}} \longleftarrow d_j$
      end if
      if $(d_j$ **different from** $d_j^{\text{move}})$ **then**
        $U_{\text{diff}} \longleftarrow U_{\text{diff}} + d_j^{\text{move}}(d_j^{\text{move}} - 2\mu_j) - d_j(d_j - 2\mu_j)$
      end if
    end for
    $p^{\text{move}} \longleftarrow \min\{1, \exp(-U_{\text{diff}}/T)\}$
    $u \longleftarrow \text{uniform}(0,1)$
    if $(u < p^{\text{move}})$ **then**
      $x_i \longleftarrow -x_i$
      for j **in** $\partial(i)$ **do** $d_j \longleftarrow d_j^{\text{move}}$
    end if
  end for
  $T \longleftarrow T(1-\epsilon)$
end while
$x_{\text{estimate}} \longleftarrow x$