AN ESTIMATOR FOR IMAGES WITH ORDERED COLORS

HÅVARD RUE

December 15, 1994

ABSTRACT. A common problem in Bayesian imaging is to find a point estimate of the true image based on the observed data and prior information through the posterior distribution. Common estimators for this purpose include the posterior mode and mean, which corresponds to the zero–one and (pixel–by–pixel) quadratic loss. However, both estimators have a fundamental weakness; the loss does not depend on the spatial structure of the errors. This is important because systematic structure in the errors can lead to misinterpretation of the estimated image. Typically, false objects may be introduced or true objects hidden. We propose a loss function that checks local linear dependency in the error. Furthermore, we obtain an estimator which has a low expected squared sample covariance in the errors. We present simulation results for some artificial data.

KEYWORDS. Bayesian inference; Loss functions; Image reconstruction; Image restoration.

Address. THE NORWEGIAN INSTITUTE OF TECHNOLOGY, DEPARTMENT OF MATHEMATICAL SCIENCES, N–7034 TRONDHEIM, NORWAY.

E–mail address. Havard.Rue@imf.unit.no

1. Introduction

The topic we address in this report is the following. Let $\boldsymbol{x}$ be a discrete random vector with probability distribution $\pi(\boldsymbol{x}|\text{data})$,

$$\boldsymbol{x} = (x_1, x_2, \dots, x_n) \quad \sim \quad \pi(\boldsymbol{x}|\text{data}) \tag{1}$$

The goal is to obtain a proper point estimate $\hat{\boldsymbol{x}}$ of $\boldsymbol{x}$. Intuitively, the mean of $\boldsymbol{x}$ and the mode in $\pi(\boldsymbol{x}|\text{data})$ are among the candidates. The mean has the well known property that it minimizes the sum of squared error for each $\hat{x}_i$

$$\mathrm{E}(\boldsymbol{x}) = \arg\min_{\boldsymbol{z}} \mathrm{E}(\|\boldsymbol{x} - \boldsymbol{z}\|_2^2), \tag{2}$$

which makes it optimal with respect to sum of squared error criteria or *loss function* $\|\boldsymbol{x} - \boldsymbol{z}\|_2^2$. This is appropriate if the x_i's have no specific or preferred ordering. However, the situation in image analysis is quite different as $\boldsymbol{x}$ has a regular geometrical organization like the elements of an matrix where each matrix element corresponds to a pixel. Each x_i corresponds to the greylevel at pixel i usually an integer value between 0 and 255 for 8 bits resolution images. The regular structure is usually accounted for by some neighborhood structure in $\pi(\boldsymbol{x}|\text{data})$, meaning that the conditional distribution for x_i given the rest depends only of x_j's in the neighborhood of pixel i. The concept of neighborhood is also relevant for the *interpretation* of $\hat{\boldsymbol{x}}$, as we interpret $\hat{x}_i$ together with $\hat{x}_j$'s in the neighborhood of pixel i. This moving–window interpretation of the estimate should, if possible, be the criteria for designing the estimator for our problem. This criterion is our loss function and we can define similar to (2) an optimal estimator with respect to this criterion. To summarize our own interpretation into one loss function is however a fundamental problem with perhaps no solution. In this report study a simple model for a such criterion or loss function that incorporate some information available from the regular geometrical structure of $\boldsymbol{x}$. The idea is as follows. The magnitude of the error is important to control, but in addition we study also the local spatial structure of the error in the neighborhood of each pixel. This because highly local dependent errors can lead to misinterpretation of $\hat{\boldsymbol{x}}$. By penalizing high local sample covariance of the error in the neighborhood of each pixel, we can design an estimator that is optimal with respect to low local linear dependency in the error in addition to a small magnitude.

The paper is organized as follows. The next section gives a brief introduction to the Bayesian approach to image analysis and introduce the notation needed. In Section 3 we discuss common estimators and how the corresponding loss functions can be improved. We present the proposed loss function (or estimator) in Section 4, while Section 5 describes the two–stage Markov chain Monte Carlo algorithm we use to compute the estimate. Finally, we present some simulation results using artificial data in Section 6.

2. Bayesian Image Analysis

We consider a standard problem in imaging; To estimate the true image or scene $\boldsymbol{x}$ from observed data $\boldsymbol{y}$ which is corrupted by noise. Let the true image $\boldsymbol{x}$ be a set of pixels or sites $\{x_i : i \in \mathcal{S}_x\}$ on a regular grid $\mathcal{S}_x$. Each x_i is one of C colors or grey levels, $x_i \in \Lambda = \{0, 1, \ldots, C-1\}$. We assume ordered colors meaning that two pixels with neighbor colors are similar in some sense and represent for example measurements of a physical quantity. Let Ω_x denote the configuration space $\Lambda^{\text{card}(\mathcal{S}_x)}$, where $\text{card}(\mathcal{S}_x)$ is the cardinality of $\mathcal{S}_x$. The observed data $\{y_i : i \in S_y\}$ are under perfect conditions some known function of $\boldsymbol{x}$, $y_i = \phi_i(\boldsymbol{x})$. It is common that ϕ_i has a local structure and depends only on x_j's in the neighborhood of x_i. Standard examples include the noise reduction case where $\phi_i(\boldsymbol{x}) = x_i$ and motion blurred and out–of–focus images. Imperfect measurement conditions introduce some noise in the observations. Common noise models involve the Gaussian and Possion distribution as conditioned on $\phi_i(\boldsymbol{x})$, y_i is a Gaussian or Poisson random variable with mean $\phi_i(\boldsymbol{x})$. In general we need to know the likelihood $\pi(\boldsymbol{y}|\boldsymbol{x})$ which gives the distribution for $\boldsymbol{y}$ conditioned on $\boldsymbol{x}$. The likelihood is often of the form $\prod_{i \in \mathcal{S}_y} \pi(y_i|\phi_i(\boldsymbol{x}))$, meaning that y_i and y_j are independent conditioned on $\boldsymbol{x}$.

The Bayesian approach to the estimation problem is to assign a prior $\pi(\boldsymbol{x})$ to the true image and thus model the true image as a realization from the prior distribution. This is not as critical as it might seem because only the part of the prior that is supported by the likelihood is important for the inference of $\boldsymbol{x}$. Hence, the inference should be based on the posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x})$. It seems sufficient that the prior models the local regularities in the true image such as local smoothness that are supposed to exist. A common way to model local regularities is through the use of Markov Random Fields (MRF) models introduced into image analysis by Geman and Geman [4]. The neighborhood system in the prior usually impose a similar or slightly larger neighborhood system in the posterior distribution. Note however, that this in general depends on the dependency structure in $\{\phi_i(\boldsymbol{x})\}$.

Any inference about $\boldsymbol{x}$ should be based on the posterior distribution, but there exist many different approaches in the present situation. One choice is to display realizations from the posterior distribution as a movie, possibly smoothed in time, and let the human mind draw the conclusions. If we want a point estimate of $\boldsymbol{x}$, we can use the *Iterated Conditioned Modes* (ICM) algorithm of Besag [1] or simply use *low temperature sampling*, which use a sample from the posterior at a low temperature as the point estimate [3]. In this report, we concentrate on estimators which are derived using decision theory and the concept of *loss functions*. Much attention has been paid to the modeling of the prior distribution, but little effort has been made to design a proper estimator through the loss function

$$l(\boldsymbol{x}, \boldsymbol{z}) : \Omega_x \times \Omega_x \longrightarrow [0, \infty). \tag{3}$$

The loss function measures a loss or the cost for estimating the true image as $\boldsymbol{z}$ when it really is $\boldsymbol{x}$. Usually, the expected value of the loss is denoted the risk. Once the loss has been specified the optimal estimate $\boldsymbol{x}^*$ is found by minimizing the risk. The result is

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{z}} \mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}} l(\boldsymbol{x}, \boldsymbol{z}). \tag{4}$$

One of the reasons for the limited experience with loss function modeling in Bayesian imaging is the practical problem of computing the estimate for complex loss function models. Based on some ideas of Geman [3], Rue [6] discussed modeling and interpretation of loss functions depending on $\boldsymbol{x}$ and $\boldsymbol{z}$ only through the misclassification vector $m_i = 1 - \delta(x_i - z_i)$ (where $\delta(a)$ is one when a is zero and zero otherwise), which is suitable for image with unordered colors or labels. Rue [6] also proposed a two–stage Markov chain Monte Carlo algorithm to compute the estimate for these loss functions. The presented simulation experiment with the Ising–model showed increased visual performance of the new estimators. In this paper we will consider images with ordered colors. Hence, the misclassification vector is now inappropriate as the size or the value of the error $\boldsymbol{e} = \boldsymbol{x} - \boldsymbol{z}$ is informative and of interest. We will therefore study a model of a loss function depending on $\boldsymbol{x}$ and $\boldsymbol{z}$ only through the error $\boldsymbol{e}$.

3. Loss Function Models

In this section we review common estimators for estimating images and discuss why the corresponding loss functions can be improved.

Instead of choosing a loss function one might choose the estimator and derive the corresponding loss function. This can be said to be the case with the *maximum a-posteriori* estimator $\boldsymbol{x}_{\mathrm{MAP}}$ which maximize the posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{y})$. At first glance $\boldsymbol{x}_{\mathrm{MAP}}$ seems to be a reasonable estimator. Note however, that the corresponding loss function is the zero–one loss meaning a loss which equals to zero when $\boldsymbol{e} = \boldsymbol{0}$ (for all pixels) and on otherwise. Clearly this loss function in inappropriate and there are two main reasons. Firstly, the loss should depend on both the size and the type of error because some spatial error structure reduces the visual information more than others. Secondly, the range of dependency in the loss function is global. Suppose that we fix $\boldsymbol{x}$ and all but one pixel i in the estimated image $\boldsymbol{z}$. Then a plausible requirement for the loss as a function of z_i would be that the loss does not depend on z_j's in pixels far away from pixel i. This suggests a loss function with a local structure. Although $\boldsymbol{x}_{\mathrm{MAP}}$ has been used with good results in many applications it may lead to over–smoothing, gross mislabelings and the obliteration of small structures in problems like image restoration, reconstruction and classification [3].

Another approach is to transfer the complex multidimensional problem to a set of more intuitive and easier problems and then estimate x_i from the marginal distribution $\pi(x_i|\boldsymbol{y})$ from for instance the marginal mean or median estimator. The corresponding loss function is of the form

$$l(\boldsymbol{x}, \boldsymbol{z}) = \sum_{i \in \mathcal{S}_x} l_{\text{pixel}}(x_i, z_i), \tag{5}$$

where l_{pixel} is e_i^2 and $|e_i|$ for the mean and median estimator, respectively. The loss function (5) considers the loss to be invariant to any permutation of the error $\boldsymbol{e}$. If there is strong spatial structure in $\boldsymbol{e}$ the loss is the same as if we rearrange the indices in the best possible way to remove the systematic structure in $\boldsymbol{e}$. This is of course not a reasonable behavior. We also should be concerned about the structure of the error and not only in magnitude. This is because systematic (local) structure in the error can lead to misinterpretations of the image as new and false objects may be introduced or true objects can be hidden, compare the difference in the loss of information by adding iid Gaussian noise and local correlated Gaussian noise with the same variance to a true image. Both the mean and the median estimators have nice interpretations if we only study the marginal distributions. Unfortunately, the marginal distribution may not provide enough information. Recall that our estimation problem has a regular geometric structure as $\{x_i\}$ are pixels on a regular grid. Therefore, neighborhood information may be informative as illustrated by the following example. Assume that $\pi(x_i|\boldsymbol{y})$ has two separate distinct modes with equal probability. The mean and median give an estimate in the valley between the two peaks. This is reasonable as we cannot decide which peak is to be preferred without more information. By studying the joint distribution of pixel i and its neighbors we may obtain additional information which can be used for selecting the correct peak.

Although local MRF prior models have been applied with reasonable success in many applications [3], the prior model itself provides limited information of the true scene. It is the part of the prior that is supported by the likelihood which gives a reasonable posterior MRF model for $\boldsymbol{x}$ conditioned on the data. It is therefore important to have a flexible prior model that can handle both smooth regions and discontinuities when combined with the data through the likelihood. A prior model with a small or moderate neighborhood system is usually preferred to a model where the neighborhood system is large, because a large neighborhood system can reduce the flexibility of the model. The same remark also applies to loss functions. As our loss function model is only a crude approximation to the "real" loss, we should be conservative in the following way. The posterior distribution should be used to obtain knowledge of the risk for various error configurations. We should not let the loss function control the risk without proper support from the posterior. For example, we should not attempt to control the simultaneous occurrence of the squared error at pixel i and j, i.e. $(e_i e_j)^2$ if $\pi(x_i, x_j|\boldsymbol{y}) = \pi(x_i|\boldsymbol{y})\,\pi(x_j|\boldsymbol{y})$ or approximately so. The consequence of doing this anyway, is that the estimator may be influenced by terms supported only by the loss function and not from the posterior distribution (and hence the data) itself. For a MRF posterior distribution with a local neighborhood structure this suggests a loss function with a similar local structure. Then the decision at pixel i depends on the decisions in the neighborhood of pixel i for which the decisions are supported by the posterior and then also the data.

The design of a proper loss function for estimating grey level images is a difficult problem whose importance depends on the situation. Loosely speaking, good data give reasonable and similar estimates for any decent loss function because the posterior has most of the probability mass centered around a single peak. For moderate (quality) data, a better loss function can yield a more sensible estimator as it minimizes the loss over several reasonable states in Ω_x with posterior probability not near zero. The important part of the loss function is its behavior for small and moderate errors since it is this part of the loss function controls the estimator. Starting from (5) we could add a spatial term to the loss function and then introduce local dependency into the estimator. The decision at pixel i will then depend on the (pending) decisions in the neighborhood of i similar to a Markov property. This introduces information from distributions like $\pi(x_i, x_j|\boldsymbol{y})$ and $\pi(x_i, x_j, x_k|\boldsymbol{y})$ (where pixel i, j and k are neighbors) into the estimator in addition to properties from the marginal distributions. Presumably, the additional information is informative for the decisions in e.g. situations with a two–mode marginal.

We can also include in the loss function a correction for the human eye's ability to detect small changes in the grey level at pixel i. It is well known that sensitivity of the eye depends both on the grey level at pixel i and on the grey levels in the neighborhood of pixel i. We can define an visual error $\tilde{\boldsymbol{e}}$ as $\tilde{e}_i \propto e_i V(x_i, x_{\partial i})$ where $V(x_i, x_{\partial i})$ measures ability of the eye to detect small changes in the grey level at level x_i in the neighborhood $x_{\partial i}$. The visual error $\tilde{\boldsymbol{e}}$ can then be used instead of $\boldsymbol{e}$ to define loss functions. The algorithm described in Section 5 can still be used. We consider this approach to be an interesting open problem but do not pursue it further here.

4. The Proposed Loss Function Model

In this section we present a simple loss function model for the point estimation of images with ordered colors. The model does not incorporate specific terms regarding structure or properties of the true image, but depends only on $\boldsymbol{x}$ and $\boldsymbol{z}$ through the error $\boldsymbol{e}$. In addition to controlling the magnitude of the error at each pixel we also consider the spatial structure of the error. As a first approximation we consider local linear dependency in the error and add terms into the loss function that penalize high local sample covariance in the error. We model the loss function as a weighted sum of two terms,

$$l(\boldsymbol{x}, \boldsymbol{z}) = l_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{z}) + \lambda\, l_{\mathrm{S}}(\boldsymbol{x}, \boldsymbol{z}). \tag{6}$$

The first term controls the magnitude of the error at each pixel and is invariant for any permutation of $\boldsymbol{e}$. The second term controls the spatial structure of the error and is invariant of the shift, translation and rotation of $\boldsymbol{e}$. This ensures the corresponding estimator to be invariant for rotation and translation of the problem (apart from the boundary region) as argued in [6].

The term that controls the local magnitude of the error is

$$l_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{z}) = \sum_{i \in \mathcal{S}_x} l_{\text{pixel}}(x_i, z_i) \tag{7}$$

and the expected value of (7) depends of $\pi(\boldsymbol{x}|\boldsymbol{y})$ only through the marginal distributions. Proper choices for $l_{\text{pixel}}(x_i, z_i)$ include e_i^2 and $|e_i|$ i.e. the loss functions for the mean and median estimator. The loss increases as a function of $|e_i|$ for both choices. A more appropriate choice is to let the loss tend to a finite value as $|e_i|$ increases since small changes for large $|e_i|$ do not influence our loss. The idea is that if an error is large it does not matter how large it is. One of many ways to incorporate this is to choose

$$l_{\text{pixel}}(x_i, z_i) = \frac{e_i^2}{\kappa^2 + e_i^2}, \tag{8}$$

where κ is a scale parameter for the error. The loss in (8) behaves like $(e_i/\kappa)^2$ near $e_i = 0$ and like $1 - (\kappa/e_i)^2$ when $|e_i|$ is large. If $\lambda = 0$ in (6), we obtain the marginal mode and mean as the estimator for small and large values of κ respectively. There is another reason why a choice like (8) is appropriate when the number of colors is large. The loss function (8) is uniformly bounded. Hence, it is easier for the spatial term l_{S} to dominate the expected loss and thereby force x_i^* away from the estimate with $\lambda = 0$ when needed.

To control the spatial structure of the error we study the sample covariance of $\boldsymbol{e}$ in a set ω. We use for ω a local set as we want a local structure in our loss function. In the forthcoming simulation experiments we used ω equal a 3×3 square. Fix a site in ω to be the center in the set. Define ω^c as the set ω translated to a new center site c and $\omega + c$ as the set ω where each element is translated by c and ω_θ as the set rotated with an angle $\theta \in \{0, \pm\pi/2, \pi\}$ around its center. The sample covariance of $\boldsymbol{e}$ at lag k in ω is

$$R_\omega(k) = \frac{1}{\text{card}(\omega \cap (\omega - k))} \sum_{j \in \omega \cap (\omega - k)} (e_j - m_\omega)(e_{j+k} - m_\omega) \tag{9}$$

where m_ω is the sample mean of $\boldsymbol{e}$ in ω,

$$m_\omega = \frac{1}{\text{card}(\omega)} \sum_{j \in \omega} e_j. \tag{10}$$

We define $R_\omega(k)$ to be zero if ω is affected by the boundary of $\mathcal{S}_x$. For a set of predefined lags L, we can define a penalty term for the error structure in ω by the squared sample covariance for each lag. The global penalty term l_{S} consists of the contribution from ω centered at each pixel in $\mathcal{S}_x$, plus a similar contribution from the rotated set ω_θ (if ω is not already invariant for rotation),

$$l_{\mathrm{S}}(\boldsymbol{x}, \boldsymbol{z}) = \sum_{c \in \mathcal{S}_x} \sum_{\theta \in \{0, \pm\pi/2, \pi\}} \sum_{k \in L} R^2_{\omega^c_\theta}(k). \tag{11}$$

We can use more than one basis set by adding up terms like (11).

5. Algorithm

From the loss function (6) we can obtain an approximation to the true Bayes estimate $\boldsymbol{x}^*$ by a two–stage Markov chain Monte Carlo (MCMC) and simulated annealing algorithm [6]. This section gives only a brief overview of the algorithm and we refer the reader to [6] for a more comprehensive discussion.

The first critical step in the algorithm is to estimate the expected loss in terms of $\boldsymbol{z}$,

$$\widehat{U}(\boldsymbol{z}) = \widehat{\mathrm{E}}\, l(\boldsymbol{x}, \boldsymbol{z}), \tag{12}$$

which in theory is trivial. We just estimate the right hand side for each $\boldsymbol{z}$ we need. In practice, this is an expensive evaluation and which we want to do only once. Here is a trivial example to illustrate the idea on how the requirement can be met. To compute $\mathrm{E}_w((w-c)^2)$ as a function of c we do

$$\mathrm{E}((w-c)^2) = \mathrm{E}(w^2 - 2wc + c^2) = \mathrm{E}(w^2) - 2c\,\mathrm{E}(w) + c^2. \tag{13}$$

We then need to express $l(\boldsymbol{x}, \boldsymbol{z})$ as $\sum_k g_k(\boldsymbol{z}) h_k(\boldsymbol{x})$, for some functions g_k and h_k. This is always possible with a finite state space Ω_x but to be practical with respect to CPU and memory storage, the number of terms in the sum should not be too large. The expected loss can now be estimated by estimating $\mathrm{E}\, h_k(\boldsymbol{x})$ using MCMC. We construct an irreducible and aperiodic Markov chain with transition probabilities such that the equilibrium distribution is $\pi(\boldsymbol{x}|\boldsymbol{y})$ and $\mathrm{E}\, h_k(\boldsymbol{x})$ is estimated as the average of h_k using samples from the (ergodic) Markov chain. The law of large numbers still holds even if the samples are weakly dependent. We refer the reader not familiar with MCMC to [2] for an introduction and review.

The term (7) in the loss function which involves only the marginal distributions, can be handled in several ways. For a small number of colors C we estimate the expected value of $l_{\mathrm{pixel}}(x_i, z_i)$ as a function of z_i as

$$\widehat{\mathrm{E}}\, l_{\mathrm{pixel}}(x_i, z_i) = \sum_{x_i} \widehat{\pi}(x_i|\boldsymbol{y})\, l_{\mathrm{pixel}}(x_i, z_i), \tag{14}$$

where the marginal distribution is estimated from MCMC. This corresponds to a complete representation of $l_{\mathrm{pixel}}(x_i, z_i)$ in terms of powers of e_i. For more than a few colors, we can approximate l_{pixel} using a truncated series expansion in e_i and thereby get a polynomial approximation in z_i of the estimated expected local loss.

The spatial term in the loss function (11) is more complicated as it involves also the joint distribution of x_i and its neighbors. By using that $e_i = x_i - z_i$ we get

$$\widehat{\mathrm{E}}\, R^2_\omega(m) = c + \sum_{i\in\omega} \widehat{a}^{(m)}_i z_i + \sum_{ij\in\omega} \widehat{a}^{(m)}_{ij} z_i z_j + \sum_{ijk\in\omega} \widehat{a}^{(m)}_{ijk} z_i z_j z_k + \sum_{ijkl\in\omega} a^{(m)}_{ijkl} z_i z_j z_k z_l, \qquad (15)$$

where the coefficients $a^{(m)}_i$ depends on the third order moments $\{\mathrm{E}(x_i x_j x_k), ijk \in \omega\}$, $a^{(m)}_{ij}$ depends on the second order moments $\{\mathrm{E}(x_i x_j), ij \in \omega\}$ and $a^{(m)}_{ijk}$ depends on the first order moments $\{\mathrm{E}(x_i), i \in \omega\}$. The coefficients $a^{(m)}_{ijkl}$ are constants. The constant c is of no interest as $\boldsymbol{x}^*$ is not influenced by an additive constant in the loss function. The actual expressions for the $a^{(m)}$–coefficients are somewhat complicated but are easily obtained by the use of a proper tool for symbolic computation.

The second step is to minimize the estimated expected loss which is easily found from (14) and (15). We use the simulated annealing approach and interpret $\boldsymbol{z}$ as an MRF with distribution

$$\pi(\boldsymbol{z}; t) \propto \exp\left(-\widehat{U}(\boldsymbol{z})/(U_0 T(t))\right) \qquad (16)$$

where U_0 is a proper normalization constant and $T(t)$ is the temperature as a function of the time t. The neighborhood system of $\pi(\boldsymbol{z}; t)$ is controlled by the local set ω and not by the neighborhood system in $\pi(\boldsymbol{x}|\boldsymbol{y})$. As an example, let ω be a 3×3 square. The conditional distribution for z_i given $\boldsymbol{z}_{-i}$ depends then only of z_j's in a 5×5 square centered at i. By annealing the temperature tends to zero slowly while we do the sampling from $\pi(\boldsymbol{z}; t)$. If the cooling is done sufficiently slow we end up in the global minima for $\widehat{U}(\boldsymbol{z})$ [4]. With a faster cooling schedule we are not guaranteed to find the global minima.

6. Experiments

This section presents simulation experiments with the proposed loss function on some artificial. We use a MRF prior model which model local smoothness as a sum of local contributions from each pixel and its eight nearest neighbors,

$$\pi(\boldsymbol{x}) \propto \exp\left(-U_x(\boldsymbol{x})\right) = \exp\left(-\beta \sum_{\langle i,j\rangle} \Upsilon(x_i - x_j) - \beta \frac{1}{\sqrt{2}} \sum_{[i,j]} \Upsilon(x_i - x_j)\right). \qquad (17)$$

The sum $\langle i, j\rangle$ are over the four nearest neighbors and the sum $[i, j]$ are over the four nearest diagonal neighbors. Note that $U_x(\boldsymbol{x})$ is a function of the difference between neighbor pixels only. For the potential function Υ we follow Geman and McClure [5] and use $\Upsilon(d) = d^2/(d^2 + \kappa_x^2)$ where κ_x is a scale parameter. The prior model now favors nearby pixels to be equal but also incorporate presence of edges or discontinuities in an implicit way since the derivative of Υ tends to zero as $|d/\kappa_x|$ increases. Hence, $\pi(\boldsymbol{x})$ will hardly change with $|d|$ for large $|d/\kappa_x|$.

The aim of the experiments is to check the ability of the new estimator to detect high local sample covariance in the true error. We choose a test scene with size 16×16 that consist of nine 2×2–squares with pixel value three on a background with pixel value zero. We allow for six different colors with pixel values from zero to five.

As the set L of predefined lags we use $\{(1,0),(0,1),(1,1),(1,-1)\}$, i.e. the four nearest lags. The loss function is (6) where l_{M} equals the quadratic loss and ω is a 3×3–square. We estimate the expected loss using 5000 samples from a Markov chain with the posterior distribution as its equilibrium distribution. Between each sample we did 50 full random sweeps with the sampler described below. This gave a reasonable balance between the work required to sample from the posterior and to sample the a–coefficients. Each full sweep visit all pixels in a random order. At each pixel we propose a new candidate uniformly from the available colors. The new state $\boldsymbol{x}''$ is then accepted with probability $\min(1, \pi(\boldsymbol{x}''|\boldsymbol{y})/\pi(\boldsymbol{x}'|\boldsymbol{y}))$, otherwise we stay in the old state $\boldsymbol{x}'$. To allow for sufficient mixing among possible modes in the posterior distribution while sampling from the posterior, we propose once in a while to change the four pixel values in the 2×2–square at pixel i with a new color selected uniformly among the available colors.

In the annealing step we use an exponential cooling schedule $T(t)/T(0) = \eta^t$ with $\eta = 0.99$, and 10 and 0.001 as the initial and final temperature.

6.1. Experiment 1. The data $\boldsymbol{y}$ is the true scene ($\phi_i(\boldsymbol{x}) = x_i$) with iid Gaussian noise with mean zero and variance 1.5^2 added at each pixel. The true and the noisy image are shown in Figure 1a and Figure 1b respectively. We use $\beta = 0.55$ and $\kappa_x = 0.5$ as the parameters in the prior model. The estimates $\widehat{\mathrm{E}}\,\boldsymbol{x}$ and $\widehat{\boldsymbol{x}^*}$ with $\lambda = 25$, are shown in Figure 1c and Figure 1d.

6.2. Experiment 2. In this experiment, we use the same true scene as in Experiment 1. The data is the true scene blurred with the 3×3 mask h, where $h_{ij} = 1/9$. We then add iid Gaussian noise with mean zero and variance 0.8^2 to each pixel (Figure 2b). The parameters in the prior model is $\beta = 0.6$ and $\kappa_x = 0.5$. Figure 2c and Figure 2d shows the estimates $\widehat{\mathrm{E}}\,\boldsymbol{x}$ and $\widehat{\boldsymbol{x}^*}$ with $\lambda = 100$.

6.3. Discussion. The results demonstrate the ability of the new estimator to detect high local sample covariance in the error and how the estimator make the necessary adjustments to reduce the risk. It is surprising that the new estimator allows for block–adjustment compared to the posterior mean meaning that four pixels in a 2×2–square change color. We take this as an indication for the squared local sample covariance to be a reasonable measure for local structure in the error. Both the estimators fail to restore the upper left square correctly in Experiment 1, but this appears to be true if we study the data by eye alone. Note that the difference between $\widehat{\mathrm{E}}\,\boldsymbol{x}$ and $\widehat{\boldsymbol{x}^*}$ are especially prominent with so few colors.

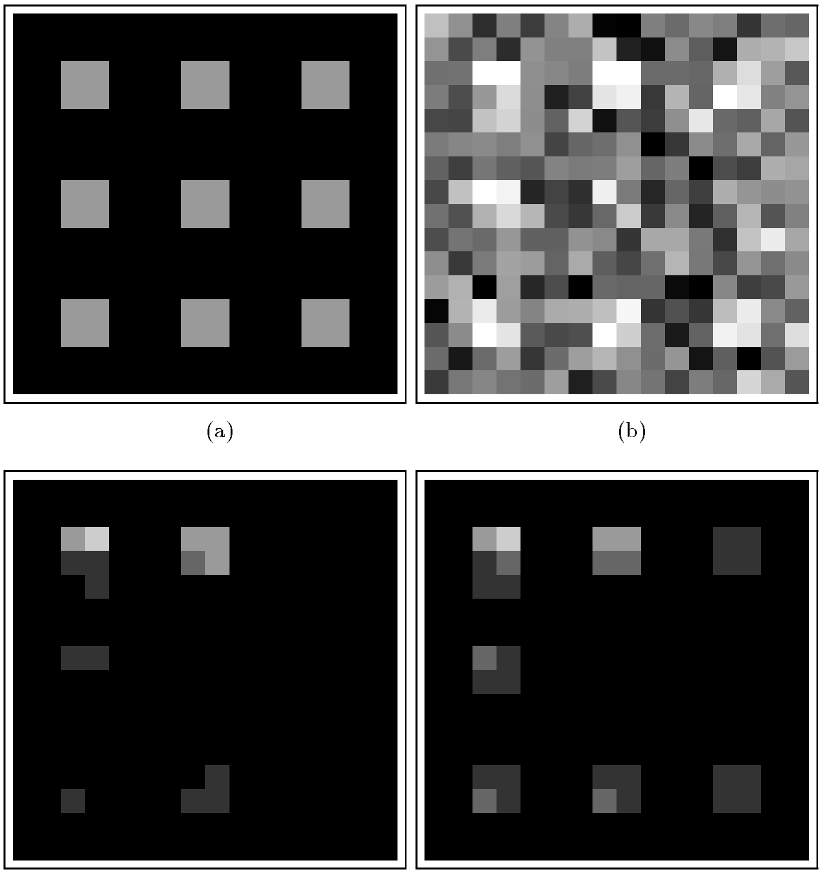
7. Conclusion

In this report, we have discussed common loss function models used in Bayesian imaging for estimation of images with ordered colors. Even if the ultimate goal, the "true" loss function seems far from realistic, we can improve common loss functions models by introducing an additional terms that handle missing features. The characteristic regular structure of the image make the spatial structure of the error important an not only the magnitude of the error. This because structure in the error can hide objects in the image or introduce new ones. As a first approximation we study a loss function that penalize high local sampler covariance in the error, and thus reduce the expected local linear dependency in the error. We demonstrate the new estimator on artificial noisy images with and without blur. The new estimator demonstrate the ability to detect high local sample covariance in the true error and gives an estimate with improved visual information compared to the posterior mean.

The new estimator has a local structure so it will not do global changes in the estimate compared to the result when $\lambda = 0$. The difference will be on a local scale, which must be of importance if the new estimator should be used.

References

1. J. Besag, *On the statistical analysis of dirty pictures (with discussion)*, J. R. Statist. Soc. B **48** (1986), 259–302.
2. J. Besag, P. Green, D. Higdon, and K. Mengersen, *Bayesian computation and stochastic systems*, Statisical Science **10** (1995).
3. D. Geman, *Random fields and inverse problems in imaging*, Ecole d'ete de probabilites de Saint-Flour XVIII, 1988 (P. L. Hennequin, ed.), Springer–Verlag, 1990, Lecture notes in mathematics; 1427.
4. S. Geman and D. Geman, *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*, IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI–6** (1984), 721–741.
5. S. Geman and D. E. McClure, *Statistical methods for tomographic image reconstruction*, Bulletin of the International Statistical Institute **Bk. 4** (1987), 5–21.
6. H. Rue, *New loss functions in Bayesian imaging*, Journal of the American Statistical Association (1995), to appear in the September issue.



(a) (b) (c) (d)

FIGURE 1. The results from simulation experiment 1. Figure (a) shows the true scene and Figure (b) shows the data with additive Gaussian noise with variance 1.5^2. Figure (c) shows the estimated posterior mean and Figure (d) the estimate with the proposed loss function.

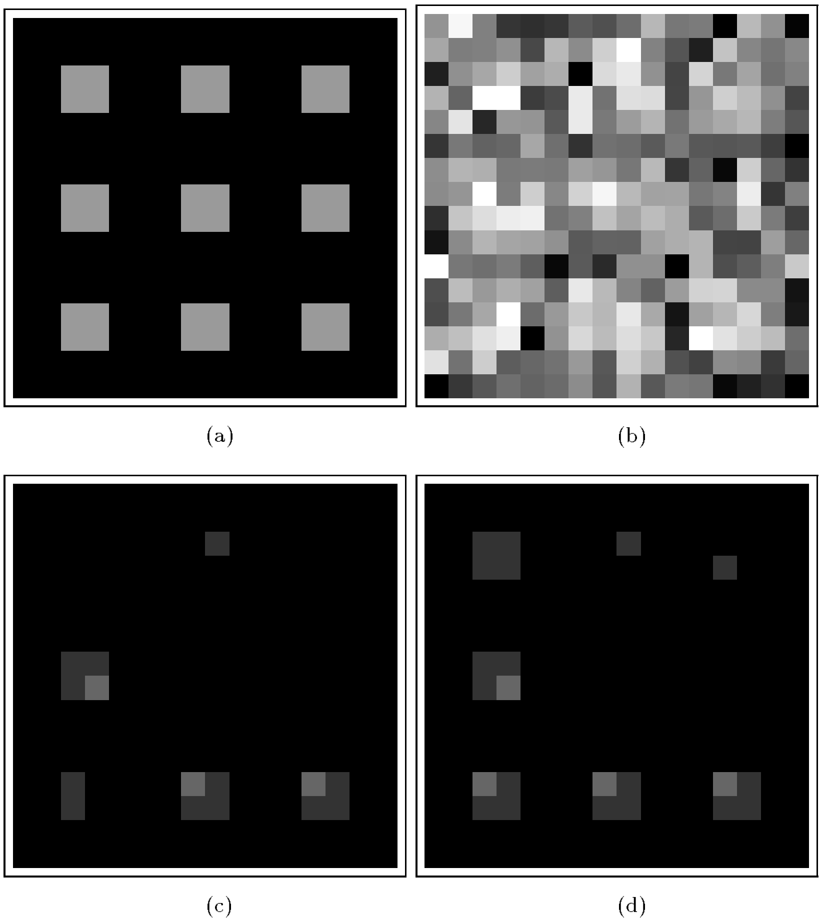


FIGURE 2. The results from simulation experiment 2. Figure (a) shows the true scene and Figure (b) shows the blurred image with additive Gaussian noise with variance 0.8^2. Figure (c) shows the estimated posterior mean and Figure (d) the estimate with the proposed loss function.