Bayesian Analysis (2014) **9**, Number 3, pp. 733–758

Bayesian Multiscale Smoothing of Gaussian Noised Images

Meng Li [*] and Subhashis Ghosal [†]

Abstract. We propose a multiscale model for Gaussian noised images under a Bayesian framework for both 2-dimensional (2D) and 3-dimensional (3D) images. We use a Chinese restaurant process prior to randomly generate ties among intensity values at neighboring pixels in the image. The resulting Bayesian estimator enjoys some desirable asymptotic properties for identifying precise structures in the image. The proposed Bayesian denoising procedure is completely data-driven. A conditional conjugacy property allows analytical computation of the posterior distribution without involving Markov chain Monte Carlo (MCMC) methods, making the method computationally efficient. Simulations on Shepp-Logan phantom and Lena test images confirm that our smoothing method is comparable with the best available methods for light noise and outperforms them for heavier noise both visually and numerically. The proposed method is further extended for 3D images. A simulation study shows that the proposed method is numerically better than most existing denoising approaches for 3D images. A 3D Shepp-Logan phantom image is used to demonstrate the visual and numerical performance of the proposed method, along with the computational time. MATLAB toolboxes are made available online (both 2D and 3D) to implement the proposed method and reproduce the numerical results.

Keywords: Chinese Restaurant Process, MCMC-free computation, 3-dimensional image

1 Introduction

An observed 2-dimensional (2D) image can be viewed as a two-dimensional data matrix $\boldsymbol{X} = (\!(X_{(j,k)})\!)$, where $j, k = 1, \ldots, n$, as the sum of the underlying mean $\boldsymbol{\mu}$ and some random noise. The objective of image smoothing or denoising is to recover the underlying array of the means $\boldsymbol{\mu}$, so that the essential features in an image such as background, foreground and objects present in the image are visible clearly. This paper proposes using a Bayesian smoothing mechanism for Gaussian noised images based on a multiscale framework, where the prior encourages structure formation essential for image processing.

An obstacle in image processing is that the number of observations n^2 is typically extremely large. Therefore decomposition and transformation are necessary for denoising. Approaches based on wavelet-type transformations and thresholding to draw boundaries have been commonly used (Donoho, 1999; Sanyal and Ferreira, 2012). Multiscale methods, which decompose the image in a sequence of refining blocks of pixels to factorize

[*] Department of Statistcs, North Carolina State University, Raleigh, NC. mli9@ncsu.edu

[†] Department of Statistcs, North Carolina State University, Raleigh, NC. ghoshal@stat.ncsu.edu.

DOI:10.1214/14-BA871

the likelihood function, are proved to be particularly useful; see Kolaczyk and Nowak (2004), Willett and Nowak (2004) and Ferreira and Lee (2007). In addition, the monograph by Lindeberg (1993) comprehensively discussed multiscale representations using scale-space theory. A Bayesian approach enjoys the advantage of adjustability to multiscale structure, since the structural properties of an image such as local constancy and contrast across boundaries can be controlled naturally by a prior distribution (Kolaczyk, 1999). White and Ghosal (2011) showed a successful application of a Bayesian multiscale denoising method to Poisson noised images. In that paper, the authors proposed using a Chinese Restaurant Process (CRP) prior to probabilistically impose equality of relative intensity among neighboring pixels, which turned out to be extremely effective in detecting structures in an image.

The Gaussian distribution (assuming a known variance σ^2) is the other member amenable to the multiscale factorization among one-parameter exponential families (Kolaczyk and Nowak, 2004). While the Poisson distribution is a reasonable model for photon-limited images, a Gaussian additive noise model seems to be a reasonable representation of the stochastic variations of $\boldsymbol{X}$ when observations are measured continuously. Even for count observations, the model based on Poisson distributions involves calculation of large factorials, which is computationally intensive when the counts of photons are large. In this case, the Gaussianity assumption can be regarded as a good approximation. The Gaussianity assumption also allows the use of conditional conjugacy to analytically compute the posterior distribution, reducing the estimation procedure to elementary matrix operations without involving Markov chain Monte Carlo (MCMC) iterations, thus speeding up the computation. In this paper, we consider images with Gaussian noise and denoise these images using the multiscale framework and a prior based on the CRP.

The proposed Bayesian denoising method with Gaussian noise will use the basic ideas of White and Ghosal (2011) of assigning a prior distribution on relative intensities to randomly impose ties among neighboring pixels in each level of the multiscale decomposition. In a multiscale analysis, we can decompose the likelihood of the entire image into the product of conditional likelihoods appearing in various levels. At any level, a block of pixels (called a parent) is split into four neighboring smaller blocks of pixels (called children) to form a parent-child group. Starting from the image level, the process is continued until the pixel level is reached.

The grouping structure of the underlying means of the children in a parent-child group is modeled by a CRP to be described in details in Section 2.1. By the conjugacy of Gaussian distributions, we can obtain the posterior mean for each pixel using only simple matrix operations. The multiscale structure allows us to work with each level independently and pool all the estimation together to obtain the final reconstruction of the original image. The CRP and multiscale representation allow our method to preserve features of images instead of oversmoothing noisy observations. All parameters are estimated by the data, and thus the proposed procedure is completely data-driven.

Denoising of 3-dimensional (3D) images has important applications in magnetic resonance imaging (MRI). Colored images can also be considered as 3D images by considering information on wavelength. Higher dimensionality makes the problem much more challenging computationally. Benefiting from the flexibility of multivariate normal distributions and the CRP, the proposed method can easily handle 3D and colored image reconstruction.

The rest of the paper is organized as follows. In Section 2, we define the statistical model, along with the prior distribution. We also compute the posterior distribution, and estimate the smoothing parameters from the data. In Section 3, extensive simulation studies are conducted to demonstrate the performance of our model in various images. Section 4 generalizes the model to 3D images and Section 5 conducts a simulation study in this situation. Proofs of Theorem 1 and another two lemmas used in the main body of the paper are presented in the Appendix.

2 Bayesian multiscale model for 2D images

A Gaussian model for a noisy image assumes the observed image $\boldsymbol{X} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with the mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. For simplicity, we consider the image with the same row length and column length, in the form of $n = 2^L$. This is generally for convenience of notation, and it is possible to relax the setting. Starting with the pixel level, we can combine a group of four neighboring pixels into one block by summing them together, resulting in a coarse level of image with row (column) length 2^{L-1}. In this process, the block is known as the parent. The four neighboring pixels forming the group are called children, and the formed structure in this way is called a parent-child group. Continuing this grouping process until the whole image is obtained, we get a multiscale representation consisting of levels $l = L, L-1, \ldots, 1, 0$. Formally, the different scales of an image $\boldsymbol{X} = (\!(X_{(j,k)})\!)$ are defined as follows. In the lth scale of the image, the parent (j,k)th block pixel is split into 4 children of block-pixels at the $(l+1)$th scale, which can be formulated as

$$X_{l,(j,k)} = \sum_{j'=2j-1}^{2j} \sum_{k'=2k-1}^{2k} X_{l+1,(j',k')} \tag{1}$$

where $l = 0, 1, 2, \ldots, L-1$ and $j, k = 1, \ldots, 2^l$. Here $X_{L,(j,k)} = X_{(j,k)}$ and when $l = 0$, $X_{0,(1,1)}$ is the summation of the entire image.

While $X_{l,(j,k)}$ is the observation of the pixel (j,k) at level l, we use $\mathbf{X}^*_{l,(j,k)}$ to denote the vector of its children group

$$(X_{l+1,(2j-1,2k-1)}, X_{l+1,(2j-1,2k)}, X_{l+1,(2j,2k-1)}, X_{l+1,(2j,2k)}).$$

The similar convention of notation to distinguish a parent from the corresponding 4-children is followed consistently by denoting parameters such as $\boldsymbol{\mu}^*_{l,(j,k)}$ and $\boldsymbol{\Sigma}^*_{l,(j,k)}$ in the following context. The model $\boldsymbol{X} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ implies that $X_{l,(j,k)} \sim N(\mu_{l,(j,k)}, \sigma^2_{l,(j,k)})$, $l = 0, 1, \ldots, L$, where N stands for a univariate or multivariate normal distribution. A multiscale statistical model is then given by the factorization of the statistical model for the entire image into the following:

$$\mathrm{P}(\boldsymbol{X}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \mathcal{N}(X_{0,(1,1)};\mu_{0,(1,1)},\sigma^2_{0,(1,1)}) \times \prod_{l=0}^{L-1}\prod_{j=1}^{2^l}\prod_{k=1}^{2^l} \mathcal{N}(\boldsymbol{X}^*_{l,(j,k)};\boldsymbol{\mu}^*_{l,(j,k)},\boldsymbol{\Sigma}^*_{l,(j,k)}), \quad (2)$$

where $\mathcal{N}$ is the probability density function of the (multivariate) Gaussian distribution, and $\boldsymbol{\mu}^*$ and $\boldsymbol{\Sigma}^*$ are the mean vector and covariance matrix of the conditional distribution of the observation corresponding to the four children given their parent, and are computed by (4) and (3) below. We assume homogenous variance among a fixed level of images, which means $\sigma^2_{l,(j,k)} = \sigma^2_l$ for all j,k and $l = 0,1,\dots,L$, and thus

$$\boldsymbol{\Sigma}^*_{l,(j,k)} = \sigma^2_l \boldsymbol{\Sigma}_0. \quad (3)$$

Further, when we go from a higher level to lower, the group of four children merges to one parent pixel, therefore the variance of all the children pixels will be absorbed to one parent level, resulting in $\sigma^2_l = \frac{1}{4}\sigma^2_{l-1}$ for $l = 1,\dots,L$. Consequently, it leads to the relationship that $\sigma^2_l = \frac{1}{4^l}\sigma^2_0$ for $l = 0,1,\dots,L$. In addition, we reparameterize $\boldsymbol{\mu}$ by $\boldsymbol{\xi}$:

$$\boldsymbol{\mu}^*_{l,(j,k)} = \frac{1}{4}\mu_{l,(j,k)}\mathbf{1}_4 + \boldsymbol{\xi}^*_{l,(j,k)}, \quad (4)$$

where $\mathbf{1}_4 = (1,1,1,1)^T$. The reparameterization of the means emphasizes that we shall re-assign the weights of four children by $\boldsymbol{\xi}_{l,(j,k)}$ based on differences with $\frac{1}{4}\mu_{l,(j,k)}$.

For the covariance matrix, instead of the identity covariance $\boldsymbol{I}$, which means that observations at all four children pixels are independent and identically distributed (i.i.d.), we force their sum to be that of their parent, so that we can preserve the total exposure of the original image. With this condition and Lemma 1 in the Appendix, we obtain that

$$\boldsymbol{\Sigma}_0 = \boldsymbol{I} - \mathbf{1_4(1_4'1_4)^{-1}1_4'} = \boldsymbol{I} - \frac{1}{4}\mathbf{1}_4\mathbf{1}_4'. \quad (5)$$

In summary, the likelihood can be factorized as follows:

$$\begin{aligned}\mathrm{P}(\boldsymbol{X}|\boldsymbol{\mu},\boldsymbol{\Sigma}) &= \mathcal{N}(X_{0,(1,1)};\mu_{0,(1,1)},\sigma^2_0) \\ &\times \prod_{l=0}^{L-1}\prod_{j=1}^{2^l}\prod_{k=1}^{2^l} \mathcal{N}(\boldsymbol{X}^*_{l,(j,k)};\frac{1}{4}X_{l,(j,k)}\mathbf{1}_4 + \boldsymbol{\xi}^*_{l,(j,k)},\frac{\sigma^2_0}{4^l}\boldsymbol{\Sigma_0}). \quad (6)\end{aligned}$$

For each level l, we estimate $\boldsymbol{\xi}_l$ of each pixel by the posterior mean $\mathrm{E}(\boldsymbol{\xi}_l|\boldsymbol{X}_l)$. The estimation of $\boldsymbol{\xi}$ can be obtained by pooling all the estimation of ξ's at all levels of the image together, which is

$$\widehat{\xi}_{(j,k)} = \sum_{l=1}^{L}\frac{1}{4^{L-l}}\mathrm{E}(\xi_{l,(j_l,k_l)}|\boldsymbol{X}_l), \quad (7)$$

where $\boldsymbol{X}_l$ is the entire image at level $l, l = 1, 2, \ldots, L$, and $j_l = \lceil j/2^{L-l} \rceil$, $k_l = \lceil k/2^{L-l} \rceil$; here $\lceil x \rceil$ is the ceiling function meaning the smallest integer not less than x. The final estimation of the pixel (j,k) in the original image is

$$\widehat{\mu}_{(j,k)} = \widehat{\xi}_{(j,k)} + \frac{1}{4^L} X_{0,(1,1)}, \quad j,k = 1, \ldots, n. \tag{8}$$

2.1 Prior distributions

While the multiscale structure allows to consider each parent-child group independently, it is important to induce local constancy in parameters through a prior distribution. The CRP is a one-parameter family of distributions on partitions that helps create ties between $\boldsymbol{\xi}$'s in each parent-child group.

When splitting the parent into four children pixels, we use the one-step quad splitting (White and Ghosal, 2011), rather than a two-step procedure (Kolaczyk, 1999) of first vertical and then horizontal, because the former is rotationally invariant. We use the 4-person CRP model to specify the prior probabilities for quad splits, which corresponds to the tie of $\boldsymbol{\xi}$'s. The configuration of $\boldsymbol{\xi}$'s formed by subgrouping the four children is denoted by $\mathcal{C}$, and let $\mathscr{C}$ be the collection of all 15 possible configurations. These possibilities are given by a CRP with parameter M. For example, the configuration $\mathcal{C} = (123)4$ means $\xi_1 = \xi_2 = \xi_3$, where the order of (1,2,3,4) is given below:

1	2
3	4

Given a smoothing parameter M in the CRP, the prior probability of each configuration $\mathrm{P}(\mathcal{C}|M)$ is given by a modified version of $\mathrm{CRP}(M)$, the CRP with parameter M. A possible modification is to remove 3 of the total 15 configurations: (14)23, (23)14 and (14)(23), which are only diagonally tied and unlikely to appear in a real image especially in the finest level of images. In this case, we re-scale the probabilities of other configurations of the same type to ensure that the total probability is 1. Table 1 displays the distribution under the modified $\mathrm{CRP}(M)$. The simulation results given later in Table 4 show that the removal of diagonal ties generally improves the accuracy slightly, but it can save substantial computation time, especially for 3D images (see Section 4) where computation is a major concern.

Conditionally on the grouping, a prior for $\boldsymbol{\xi}$ is given by a normal distribution. For notational convenience, we focus the discussion for one particular parent-child group. Let $\boldsymbol{X} = (X_1, X_2, X_3, X_4)$ be the observation of four children and $X = X_1 + X_2 + X_3 + X_4$ be the observation corresponding to the parent. The prior distribution of the parameters $\boldsymbol{\xi} = (\xi_1, \xi_2, \xi_3, \xi_4)$ can start with $N(\boldsymbol{0}, \tau^2 \boldsymbol{I})$. One natural constraint is that $\boldsymbol{\xi}$ should be summed to zero. Further, each configuration $\mathcal{C}$ corresponds to some linear constraints for $\boldsymbol{\xi}$, which can be uniquely represented by a constraint matrix $\boldsymbol{A}$ such that $\boldsymbol{A}\boldsymbol{\xi} = \boldsymbol{0}$. See Table 1 for all the constraint matrices associated with given configurations.

By Lemma 1, presented in the appendix, the prior distribution of $\boldsymbol{\xi}$ given each configuration $\mathcal{C}$ which is equivalent to the condition $\boldsymbol{A}\boldsymbol{\xi} = \boldsymbol{0}$, is given by:

$$\boldsymbol{\xi}|\mathcal{C} \sim N(\boldsymbol{0}, (\boldsymbol{I} - \boldsymbol{A}'(\boldsymbol{A}\boldsymbol{A}')^{-1}\boldsymbol{A})\tau^2), \tag{9}$$

where $\boldsymbol{A}$ is the constraint matrix corresponding to each configuration.

Table 1: Illustration of the corresponding constraint matrix $\boldsymbol{A}$ and prior probability $\mathrm{P}(\mathcal{C}|M)$ for given configuration $\mathcal{C}$. The column of $\mathcal{C}$ contains all possible configurations belonging to the same tie structures, while the diagonally tied ones are crossed out. The constraint matrix $\boldsymbol{A}$ is for the first $\mathcal{C}$, while the other constraint matrices can be obtained by permuting the columns of $\boldsymbol{A}$ according to the tieing structures. The last column is the prior probability $\mathrm{P}(\mathcal{C}|M)$, which is shared by all configurations in the same category.

$\mathcal{C}$	$\boldsymbol{A}$	$P(\mathcal{C}\|M)$
$\{1234\}$	$\begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}$	$\frac{M}{M}\frac{M}{M+1}\frac{M}{M+2}\frac{M}{M+3}$
$\{(123)4\}$, $\{(234)1\}$, $\{(134)2\}$, $\{(124)3\}$	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \end{bmatrix}$	$\frac{M}{M}\frac{1}{M+1}\frac{2}{M+2}\frac{M}{M+3}$
$\{(12)(34)\}$, $\{(13)(24)\}$, ~~$\{(14)(23)\}$~~	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$	$\frac{3}{2}\frac{M}{M}\frac{1}{M+1}\frac{M}{M+2}\frac{1}{M+3}$
$\{12(34)\}$, $\{(12)34\}$,$\{(13)24\}$, $\{(24)13\}$, ~~$\{(14)23\}$~~, ~~$\{(23)14\}$~~	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$	$\frac{3}{2}\frac{M}{M}\frac{M}{M+1}\frac{M}{M+2}\frac{1}{M+3}$
$\{(1234)\}$	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix}$	$\frac{M}{M}\frac{1}{M+1}\frac{2}{M+2}\frac{3}{M+3}$

2.2 Posterior distributions

We shall derive the posterior distribution of $\mathcal{C}$ given (M, τ) and the observation $\boldsymbol{X}$ assuming the model parameter $\sigma_0^2 = \sigma^2$ is known. The estimation of the model parameter σ_0^2 and smoothing parameters (M, τ) will be discussed in the Section 2.3. By the Bayes rule, we obtain the posterior probability of the configuration $\mathcal{C}$:

$$\mathrm{P}(\mathcal{C}|M, \tau, \boldsymbol{X}) \propto \mathrm{P}(\mathcal{C}|M)\mathrm{P}(\boldsymbol{X}|\mathcal{C}, \tau, X), \tag{10}$$

where $\boldsymbol{X}$ is the observation vector for the four children, and X is that of the parent, i.e., X is the summation of all elements in $\boldsymbol{X}$.

This is a discrete probability distribution with 12 distinct values, so they need to be scaled to sum to one. The first factor, $\mathrm{P}(\mathcal{C}|M)$, is given by the modified CRP. The second factor, $\mathrm{P}(\boldsymbol{X}|\mathcal{C},\tau,X)$ can be obtained from $\boldsymbol{\xi}|\mathcal{C}$ and $\boldsymbol{X}|\boldsymbol{\xi}$ by applying Lemma 2, presented in the appendix. From the discussion about the model assumptions and prior distributions, we have

$$\begin{aligned}\boldsymbol{X}|\boldsymbol{\xi} &\sim N(\frac{1}{4}X\mathbf{1}_4+\boldsymbol{\xi},\boldsymbol{\Sigma}_1),\\ \boldsymbol{\xi}|\mathcal{C} &\sim N(\mathbf{0},\boldsymbol{\Sigma}_2),\end{aligned}$$

where

$$\boldsymbol{\Sigma_1}=\sigma^2\boldsymbol{\Sigma}_0,\quad \boldsymbol{\Sigma}_2=\tau^2(\boldsymbol{I}-\boldsymbol{A}'(\boldsymbol{A}\boldsymbol{A}')^{-1}\boldsymbol{A}),$$

and $\boldsymbol{\Sigma}_0=\boldsymbol{I}-\mathbf{1_4'(1_4 1_4')^{-1}1_4'}$; here $\boldsymbol{A}$ is the constraint matrix corresponding to the configuration $\mathcal{C}$. Applying Lemma 2, we can obtain

$$\boldsymbol{X}|(\mathcal{C},\tau,X)\sim N(\frac{1}{4}X\mathbf{1}_4,\boldsymbol{\Sigma}_1+\boldsymbol{\Sigma}_2). \tag{11}$$

The enforced constraint $X_1+X_2+X_3+X_4=X$ makes the joint distribution rank deficient, which can be reduced to a lower 3-dimensional multivariate normal by dropping one of the co-ordinates. We shall drop the last one X_4 to make the covariance matrix nonsingular in the computation. However, we shall keep all of them in the formulas to make them symmetric, and just remind the reader of the singularity issue when necessary.

Given (M,τ), we now have the discrete distribution $\mathrm{P}(\mathcal{C}|M,\tau,X)$ for a parent-child group. The other posterior distribution $\mathrm{P}(\boldsymbol{\xi}|\mathcal{C},X)$ is another multivariate normal distribution by conjugacy, namely,

$$\boldsymbol{\xi}|\mathcal{C},\boldsymbol{X}\sim N(\boldsymbol{\Sigma}_2(\boldsymbol{\Sigma}_1+\boldsymbol{\Sigma}_2)^{-1}(\boldsymbol{X}-\frac{1}{4}X\mathbf{1}_4),\boldsymbol{\Sigma}_2(\boldsymbol{\Sigma}_1+\boldsymbol{\Sigma}_2)^{-1}\boldsymbol{\Sigma}_1); \tag{12}$$

here the vector $\boldsymbol{\xi}$ is summed to be 0, thus is lower dimensional as is the case in (11). Similarly we can drop the last one ξ_4 to address the singularity issue. The final estimate of each pixel in one parent-child group can be obtained by

$$\widehat{\boldsymbol{\xi}}=\sum_{\mathcal{C}\in\mathscr{C}}\mathrm{P}(\mathcal{C}|M,\tau,\boldsymbol{X})\mathrm{E}(\boldsymbol{\xi}|\mathcal{C},\boldsymbol{X})=\sum_{\mathcal{C}\in\mathscr{C}}\mathrm{P}(\mathcal{C}|M,\tau,\boldsymbol{X})\boldsymbol{\Sigma}_2(\boldsymbol{\Sigma}_1+\boldsymbol{\Sigma}_2)^{-1}(\boldsymbol{X}-\frac{1}{4}X\mathbf{1}_4). \tag{13}$$

Similarly,

$$\widehat{\mathrm{E}}(\boldsymbol{\xi}\boldsymbol{\xi}'|\boldsymbol{X})=\sum_{\mathcal{C}\in\mathscr{C}}\mathrm{P}(\mathcal{C}|M,\tau,\boldsymbol{X})[\widehat{\boldsymbol{\xi}\boldsymbol{\xi}}'+\boldsymbol{\Sigma}_2(\boldsymbol{\Sigma}_1+\boldsymbol{\Sigma}_2)^{-1}\boldsymbol{\Sigma}_1]. \tag{14}$$

Adding (13) across levels gives the estimate of the posterior mean of $\mu_{j,k}$ (see equation (7) and (8)). Note that $\boldsymbol{\xi}$'s over different levels l are a priori independent and their likelihood factorizes, so they are a posteriori independent too. This allows to obtain the estimate of $\mathrm{Var}(\mu_{j,k}^2|\boldsymbol{X})$ by adding variances of the appropriate ξ variables which add to $\mu_{j,k}$ in view of their posterior independence.

2.3 Estimation of parameters

When the variance σ^2 is known, we have two smoothing parameters: M and τ for each level, which determine the tieing structure via the CRP probability allocation and the prior distribution of distinct values in a parent-child group. In general, smaller M or τ encourages more ties and less variation respectively, thus resulting in smoother estimation. For a higher level, where the image is split in more pixels, the true intensities of the neighboring pixels are more likely to be equal or close, since smoothness in an image is formed by some neighboring pixels with similar intensities. Therefore, it makes sense to let (M, τ) decrease along with increasing level size. We shall use all the data at each level to determine the common (M, τ) for that level separately among different levels of the image. We obtain their values by maximizing the marginal likelihood. The nice structure of multiscale analysis makes it possible to estimate (M, τ) independently for each level. We shall also apply the decreasing constraint, which actually makes the estimated values more stable in the finer levels.

For each parent-child group, the marginal likelihood of the sample given (M, τ) and the group sum is

$$\mathrm{P}(\boldsymbol{X}|M, \tau, X) = \sum_{\mathcal{C} \in \mathscr{C}} \mathrm{P}(\mathcal{C}|M)\mathrm{P}(\boldsymbol{X}|\mathcal{C}, \tau, X). \tag{15}$$

Before maximizing (15), we pass to the logarithmic scale to make the algorithm more stable. Since the optimization is conducted for the entire level, we need to formulate the target function pooling all the parent-child groups together. For level $l = 1, 2, \ldots, L$, the length of a row or column is 2^l, and thus the number of such groups is 4^{l-1}. Using z as the index for the children groups, we can derive the target function as:

$$\sum_{z=1}^{4^{l-1}} \log \left\{ \sum_{\mathcal{C}_z \in \mathscr{C}} \mathrm{P}(\mathcal{C}|M)\mathrm{P}(\boldsymbol{X}|\mathcal{C}, \tau, X) \right\}. \tag{16}$$

The Newton-Raphson algorithm or grid search type algorithms can be applied with the decreasing constraint in (M, τ). We use a simplex search algorithm (Lagarias et al., 1998) which gives stable estimates. The selection of parameters is more critical for finer scales of the image. In practice, we use multiple starting points to ensure global maximization.

For real image data, the variance σ^2 at each pixel is unknown. The parameter σ^2 can also be estimated by maximizing the marginal likelihood similar to the estimation of (M, τ). However, unlike (M, τ), which are estimated separately for each level of data, σ^2 is fixed across different levels. Thus the optimization of the log-likelihood is much more computationally intensive, especially for images with large sizes. The method of moments estimation has a computational advantage and will be used here. For the (j, k)th children group in the $(L-1)$th level of the image, denote the indexes corresponding to the (j, k)th block as $C(j, k) = \{(j', k') : j' = 2j-1, 2j, k' = 2k-1, 2k\}$. Let $s^2_{j,k}$ be the sample variance for the data $X_{j',k'}$, where $(j', k') \in C(j, k)$, then $s^2_{j,k}$ is an unbiased estimate for σ^2 if $\mu_{j',k'}(j' = 2j-1, 2j; k' = 2k-1, 2k)$ are all the same.

Then an estimate of σ^2 can be obtained by averaging all the sample variances:

$$\widehat{\sigma}^2 = \frac{1}{4^{L-1}} \sum_{j=1}^{2^{L-1}} \sum_{k=1}^{2^{L-1}} s_{j,k}^2. \tag{17}$$

Obviously, not all four children pixels have intensities coming from the Gaussian distribution with the same means, such as if the children block contains a part of a boundary in the image. In that case, $s_{j,k}^2$ will be inflated, since the difference among the means is added to the overall variation. But the effect will be not significant if the non-boundary pixels dominate the whole image, as showed by the following theorem. The proof of Theorem 1 is deferred to the appendix.

Theorem 1. *Suppose that we observe an image $\boldsymbol{X}$ of size $n = 2^L$ in each direction. Let the true image arise from the realization of a function $g : (s,t) \mapsto \mu(s,t)$ on the domain $\mathcal{D} = [0,1] \times [0,1]$ and be corrupted by independent Gaussian noise with mean 0 and variance σ^2 at each pixel.*

Assume that the true surface $g(\cdot,\cdot)$ is bounded by a constant m. Further assume that $\mathcal{D} = \mathcal{D}_1 \cup \cdots \cup \mathcal{D}_k, k < \infty$, where $\mathcal{D}_i^0 = \mathcal{D}_i \setminus \partial\mathcal{D}_i$ is a convex set such that $g(\cdot,\cdot)$ is Lipschitz continuous on $\mathcal{D}_i^0, i = 1, \ldots, k$. Then $\widehat{\sigma}^2$ defined in (17) is asymptotically unbiased and is consistent for σ^2 as $n \to \infty$.

We can improve the finite sample performance of $\widehat{\sigma}^2$ by the following modifications. Consider the sample variances $s_{i,j}^2, i,j = 1, \ldots, 2^{L-1}$, as the new scalar responses, and denote them as $z_t, t = 1, \ldots, 4^{L-1}$. We know that the majority of z_t have mean σ^2 but the others have means larger than σ^2, for example the blocks containing boundaries. Therefore we could classify all $s_{j,k}^2$'s into two groups via commonly used clustering methods such as K-means (Hartigan and Wong, 1979) with $K = 2$. The two clusters are boundary-containing or boundary-free groups and we can use the mean of $s_{j,k}^2$'s in the boundary-free group to estimate σ^2. As a more sophisticated alternative, a Gaussian mixture model can be used to classify z_t's to the various groups:

$$z_t = p_1 f_1 + p_2 f_2 + \cdots + p_K f_K \tag{18}$$

where $p_1 + \cdots + p_K = 1$, and $f_1, \ldots, f_K$, are densities for normal distributions. An Expectation Maximization (EM) algorithm is used to estimate parameters (McLachlan and Peel, 2000) and has already been implemented in MATLAB. The number of components K can be selected by the Bayesian information criterion (BIC) using data. A simpler alternative could be to use just the fixed value $K = 2$, where the mean of the component with larger proportion is used as the estimate of σ^2. All the modifications improve the performance of $\widehat{\sigma}^2$ by accounting for the inflation effect. The K-means with $K = 2$ and the Gaussian mixture model with two components lead to straightforward computation, which is important for large image data.

2.4 Asymptotic properties

The proposed Bayesian denoising method enjoys some good convergence properties. Let $\boldsymbol{\mu} = (\mu_{j,k} : j, k = 1, \ldots, 2^L)$ be the mean parameter of the image in the Bayesian model, and $\boldsymbol{\mu}^0 = (\mu^0_{j,k} : j, k = 1, \ldots, 2^L)$ be the true value of the underlying mean of the observed image. Define a structure $\mathcal{M}$ by equality among neighboring values of the split parameters at any level. For example, the full model for the observed image is that the components of the underlying mean $(\mu_{j,k} : j, k = 1, \ldots, 2^L)$ are not at all tied among intensities. With more specifications of ties, a structure becomes more restricted. Let the true structure be $\mathcal{M}_0$, and call any model that is broader than $\mathcal{M}_0$ a compatible model; otherwise call it incompatible. A compatible model has fewer assumptions of ties than the true model, and thus will never contain any incorrect specification of ties but it may miss some correct ties. White and Ghosal (2011) showed that under the Poisson model for the image, as the total intensity tends to infinity, the posterior distribution of relative intensities is consistent and the posterior probability of the true model converges to one. A similar result holds in our setting of Gaussian noised images.

Theorem 2. *For the Bayesian smoothing method with modified CRP, we have that*

(a) the posterior distribution of $\boldsymbol{\mu}$ *is consistent at* $\boldsymbol{\mu}^0$ *as* $\sigma \to 0$*;*

(b) for any incompatible model $\mathcal{M}^*$*, the posterior model probability* $\Pi(\mathcal{M}^*|\boldsymbol{X}) \leq \exp(-c/\sigma^2)$ *for some constant* $c > 0$ *almost surely for all sufficiently small* σ*;*

(c) for any compatible model $\mathcal{M}^*$ *that is different from the true model, the posterior model probability* $\Pi(\mathcal{M}|\boldsymbol{X}) = O_p(\sigma^d)$*, where* d *is a constant standing for the redundancy of* $\mathcal{M}^*$*.*

Note that the asymptotic regime here is $\sigma \to 0$, which is different from that in Theorem 1. The setting $\sigma \to 0$ can be interpreted as taking repeated independent observations on the same image, therefore the resulting mean image can be thought of as a noisy image with standard deviation approaching zero. The proof of Theorem 2 relies on the same arguments given by White and Ghosal (2011), which only use the finite dimensionality and the regularity of the Poisson family, and non singularity of the prior distribution. As the Gaussian model meets these general conditions, their arguments go verbatim.

3 Simulation results for 2D images

In this section, we conduct a simulation study to judge the practical performance of the proposed Bayesian smoothing method using the Chinese restaurant process (Bayesian CRP). We compare with five other existing approaches, which are translation-invariant Haar (TI-Haar) estimation (Willett and Nowak, 2004), coarse-to-fine wedgelet (Castro et al., 2004), platelet (Willett and Nowak, 2003), nonparametric Bayesian dictionary learning (BPFA) proposed by Zhou et al. (2012) and the conventional running median method. All the implementations are completed in MATLAB.

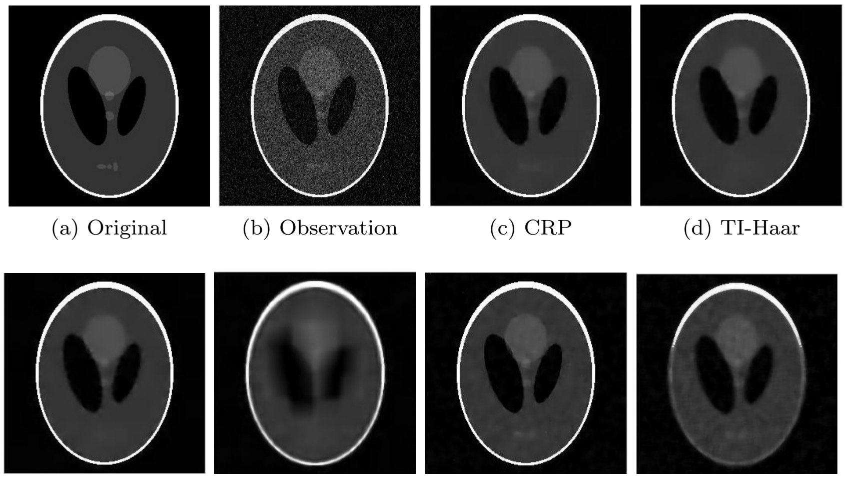
The Gaussian model can be applied to a wide range of images, regardless of the quantity being measured for each pixel. They are much more applicable to the large photon images, or images based on a continuous quantity like intensity. The essential differences can be summarized according to the number of unique values of intensities in the image. Call them discrete images when there are a limited number of unique values and continuous when there are a large number of unique values. In our simulations, typical images from both discrete and continuous cases will be used. The image of the Shepp-Logan phantom (Jain, 1989) is discrete, while the Lena image is continuous. The Shepp-Logan phantom image contains ellipses with various absorption properties to mimic the outline of a head, which is widely used to test reconstruction algorithms. The Lena image is a real image typically used to measure the performance of smoothing algorithms. True intensity values in both images are within the range 0 to 1.

Cycle spinning is a common technique to remove visual artifacts in image reconstruction (Coifman and Donoho, 1995; Willett and Nowak, 2004) and can be completed by averaging random or local shifts. We average 121 local shifts (11×11, which means a step size up to 5 in each possible direction) for the methods of Bayesian CRP, wedgelet, platelet and running median. The TI-Haar is translation invariant and hence it is not necessary to apply local shifts any more, while the BPFA method already includes cycle spinning in terms of patches automatically. The tuning parameter for platelet is hard to specify. We use the value 0.1, which is the best in terms of the mean squared errors (MSE) for the Lena image when $\sigma = 0.5$. The length of the window for the running median method is fixed at 5. We estimate σ^2 by equation (17) with 2-means or Gaussian mixture models since the results are similar. All simulations are run by MATLAB on Dual Processor Xeon Twelve Core 3.6 GHz machines with 80GB RAM running 64Bit CentOS Linux 5.0. The method of wedgelet is the only one to use compiled code (Castro et al., 2004), which speeds up the computation.

The performance of various methods are compared both visually and numerically. The visual performances to the two images are shown in Figure 1 and Figure 2 for observations with light noise ($\sigma = 0.1$). We can see that the smoothed images obtained by the proposed Bayesian approach are able to identify most of the features present in the true image. For example, the small ellipses are still visible after smoothing by the Bayesian CRP method, as well as Bayesian dictionary learning (BPFA) and running median. Both the Shepp-Logan phantom and Lena images show that TI-Haar, wedgelet and platelet tend to over-smooth, which miss features present in the true images. The platelet method depends on the selection of a tuning parameter, which may have caused its problematic performance. Figure 3 demonstrates the performance of all six methods for the Lena image with heavier noise ($\sigma = 0.6$). We can see that the Bayesian CRP method reconstructs the key features such as the nose and the mouth, while achieving smoothing even though the observed image is heavily noised. TI-Haar and wedgelet tend to oversmooth and miss some features such as the boundary between the face and the arm. The platelet, BPFA and running median methods capture features but tend to overfit since the smoothed images are still grainy.

Numerical comparisons confirm our visual observations. Two common criteria are used: the mean absolute errors (MAD) and the mean squared errors (MSE). In addi-

Figure 1: Comparison of the Bayesian smoothing method with other approaches for the 256 × 256 Shepp-Logan phantom shown in (a). The noisy observation with standard deviation $\sigma = 0.1$ and a constant background noise 0.01 is shown in (b). The six denoising methods (c)–(h) are respectively Bayesian CRP, TI-Haar, wedgelet, platelet, Bayesian dictionary learning and running median. All methods except TI-Haar and Bayesian dictionary learning use $121 = 11 \times 11$ local shifts to remove artifacts.



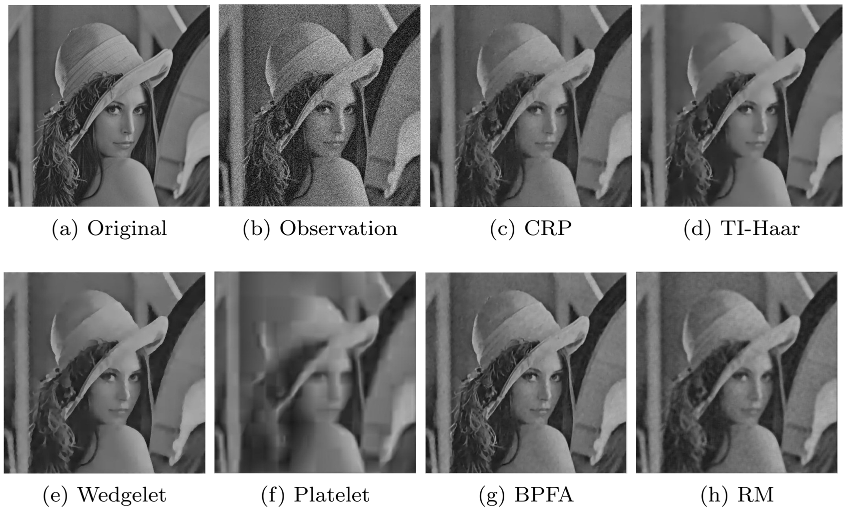
(a) Original (b) Observation (c) CRP (d) TI-Haar

(e) Wedgelet (f) Platelet (g) BPFA (h) RM

Table 2: Numerical comparison of smoothing methods for the phantom image when noise standard deviation $\sigma = 0.1$. The mean MSE ($\times 10^{-2}$), MAD ($\times 10^{-2}$) and HD ($\times 10^{-2}$) of 100 simulations are reported. The maximum standard error for each criterion is given by the last row. All methods except TI-Haar and Bayesian dictionary learning use 121 local shifts to remove artifacts. The running time for each local shift and the total time are reported in the last two columns.

Methods	MSE	MAD	HD	Time (sec)	
Observation	1.01	8.02	181.21	per shift	total
Bayesian CRP	0.04	1.42	48.53	0.27	32.55
TI-Haar	0.05	1.51	56.08	0.28	0.28
Wedgelet	0.04	1.38	47.15	4.21	509.01
Platelet	0.46	3.27	187.49	36.46	4411.19
BPFA	0.04	1.38	46.75	1042.13	1042.13
Running Median	0.13	2.55	89.27	0.03	3.78
SE (max)	0.00	0.00	0.54	–	–

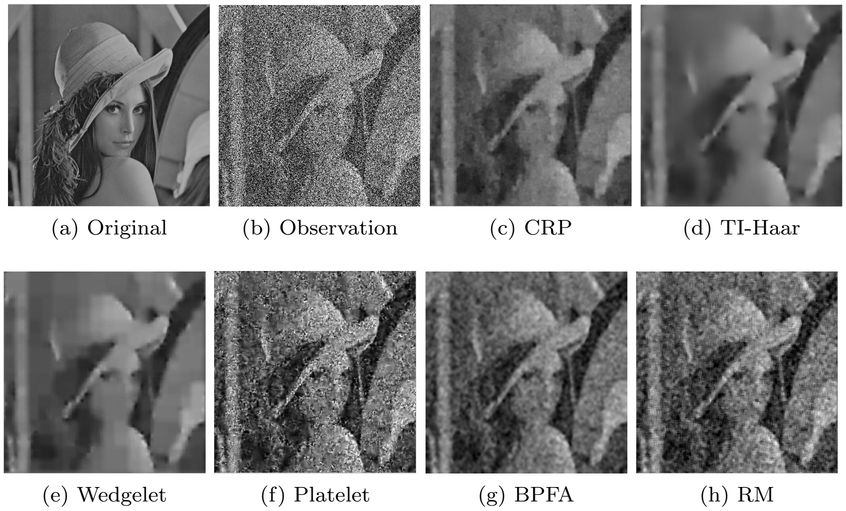
Figure 2: Comparison of the Bayesian smoothing method with other approaches for the 512 × 512 Lena image shown in (a). The noisy observation with standard deviation $\sigma = 0.1$ is shown in (b). The six denoising methods (c)–(h) are respectively Bayesian CRP, TI-Haar, wedgelet, platelet, Bayesian dictionary learning and running median. All methods except TI-Haar and Bayesian dictionary learning use $121 = 11 \times 11$ local shifts to remove artifacts.



(a) Original (b) Observation (c) CRP (d) TI-Haar

(e) Wedgelet (f) Platelet (g) BPFA (h) RM

Figure 3: Comparison of the Bayesian smoothing method with other approaches for the 512 × 512 Lena image shown in (a). The noisy observation with standard deviation $\sigma = 0.6$ is shown in (b). The six denoising methods (c)–(h) are respectively Bayesian CRP, TI-Haar, wedgelet, platelet, Bayesian dictionary learning and running median. All methods except TI-Haar and Bayesian dictionary learning use $121 = 11 \times 11$ local shifts to remove artifacts.



(a) Original (b) Observation (c) CRP (d) TI-Haar

(e) Wedgelet (f) Platelet (g) BPFA (h) RM tion, the Hausdorff distance (HD) (Huttenlocher et al., 1993) is used to measure the similarity between smoothed images and the true images. For $A = \{a_1, \ldots, a_p\}$, and $B = \{b_1, \ldots, b_p\}$, the Hausdorff distance is defined as

$$H(A, B) = \max\left\{\max_{a \in A} \min_{b \in B} \|a - b\|, \min_{b \in B} \max_{a \in A} \|a - b\|\right\}, \tag{19}$$

where $\|\cdot\|$ can be taken to be the usual Euclidean norm. Distance-based metrics such as the Hausdorff distance and Baddeley's delta metric (Baddeley, 1992; Wilson et al., 1997) can often measure the similarity between two images in a more intelligent way. The Hausdorff distance is relatively easier to compute and hence is used here. In addition to the accuracy, the computing time for each method is noted. Table 2 shows the numerical performance of all six methods for the phantom image when $\sigma = 0.1$. We can see that our method has comparable performances with wedgelet and BPFA, but uses much less time to compute the estimates. This advantage in computation is due to the avoidance of the MCMC algorithm by exploiting the conditional conjugacy structure of Gaussian distributions given the partitions. Table 3 shows the MSEs for the Lena image with various noise levels. We observe that the Bayesian CRP method tends to outperform all the other approaches when σ increases. For smaller σ's, the Bayesian CRP method is still comparable with the best ones, but incurs much less computational cost. The platelet is the best when $\sigma = 0.5$, but may suffer a lot at the other noise levels. This indicates that the platelet is sensitive to the tuning parameter and has the potential to perform well, when the tuning parameter is selected appropriately. Unfortunately, the platelet algorithm does not provide a data-driven selection of its tuning parameters, which must be subjectively chosen by the user without knowing the true image. The TI-Haar is the most computationally efficient approach among the six methods but the Bayesian CRP method outperforms it in terms of all other criteria in both Table 2 and Table 3. BPFA is another well-performing method for light noise, but it suffers from heavy noise and it is the one of the most computationally intensive methods.

In addition, we compare the modified CRP prior and the original CRP prior using the phantom images in terms of MSEs and computational time (Table 4). We can see that the two priors lead to estimates with similar accuracy. Typically the modified CRP prior is more computationally efficient, because fewer configurations lead to fewer operations.

We use various sizes of phantom images to demonstrate the scalability of the proposed method (Table 5). Figure 4 shows that the computational time is approximately linear in the total number of pixels n^2. In fact, the number of operations in expression (16) is $12 \times 4^{l-1}$ where l is the scale of the image, thus linear in the total number of pixels at lth scale; similarly the estimation procedure described by equation (13) is also linear in 4^{l-1}. As a result, the entire procedure requires $O(n^2)$ operations, where n^2 is the total number of pixels.

The proposed approach can address colored images as well. For example, a colored image in MATLAB can be represented using (red, green, blue) (RGB) representation (the so-called Truecolor images in MATLAB). Therefore, for noisy observations, we can apply the Bayesian CRP method to each slice of the color representation and then

Table 3: Numerical comparison of smoothing methods for the Lena image at various noise levels in terms of MSE ($\times 10^{-2}$). 100 simulations are run and the maximum standard errors are given by the last row. All methods except TI-Haar and Bayesian dictionary learning use 121 local shifts to remove artifacts. The running time for each local shift and the total time when $\sigma = 0.2$ are reported; the running time is similar for the other noise levels.

Method	Noise level σ						Time (sec)	
	0.2	0.3	0.4	0.5	0.6	0.7	per shift	total
Bayesian CRP	0.15	0.21	0.27	0.32	0.37	0.42	0.99	119.97
TI-Haar	0.16	0.22	0.30	0.36	0.42	0.49	2.46	2.46
Wedgelet	0.15	0.21	0.29	0.35	0.40	0.47	54.77	6627.11
Platelet	0.27	0.26	0.27	0.30	1.85	6.55	150.03	18153.47
BPFA	0.12	0.19	0.27	0.34	0.43	0.54	9993.91	9993.91
Running Median	0.22	0.32	0.46	0.63	0.86	1.13	0.17	21.11
SE (max)	0.00	0.00	0.00	0.00	0.00	0.00	–	–

Figure 4: Scalability of the Bayesian CRP. We plot the total time with the number of levels $L = \log_2(n)$, and fit a straight line when the number of pixels $n^2 = 4^L$ is the predictor. The fitted line is: Total Time $= 0.00 + 10.45\ n^2$ thus linear in the total number of pixels.

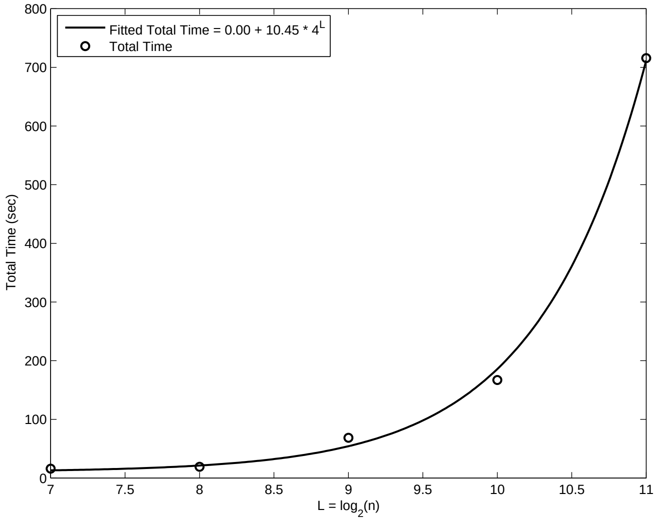


Table 4: Comparison of the modified CRP prior and the CRP prior using noisy phantom images with the standard deviation $\sigma = 0.1$ and a constant background noise 0.01 for various image sizes. We report the MSE($\times 10^{-4}$) for each of them, the differences (modified CRP − CRP) and the standard errors of the differences. The time taken by each prior is reported by the last two rows. The results are based on 121 local shifts and 100 simulations.

Summary	Priors	128	256	512	1024	2048
MSE ($\times 10^{-4}$)	Modified CRP	7.20	3.94	2.46	1.76	1.38
	CRP	7.22	3.97	2.46	1.77	1.39
	Difference	-0.02	-0.02	-0.01	-0.01	-0.01
	SE	0.04	0.02	0.01	0.01	0.01
Time (sec)	Modified CRP	15.98	19.12	68.63	167.18	715.66
	CRP	14.21	28.64	86.05	255.39	1111.25

Table 5: Scalability of the Bayesian CRP using noisy phantom images with the standard deviation $\sigma = 0.1$ and a constant background noise 0.01. The result is based on 121 local shifts and 100 simulations.

Time (sec)	128	256	512	1024	2048
Optimization	7.12	8.9031	31.39	76.65	335.43
Estimation	8.86	10.21	37.22	90.49	380.10
Total	15.98	19.12	68.63	167.18	715.66

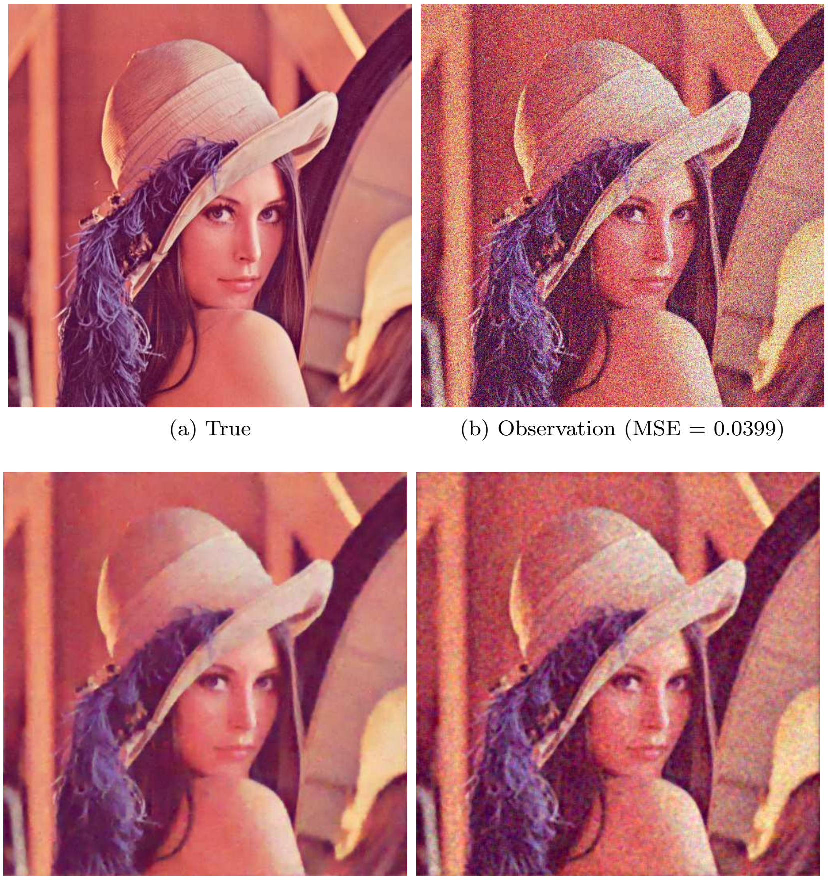
combine the three smoothed slices together. In Figure 5, we apply the Bayesian CRP to a 512 × 512 colored Lena image with Gaussian noise ($\sigma = 0.2$). The Bayesian CRP method appears to give a better smoothed image compared with the running median approach both visually and numerically (smaller MSE).

In some situations, the observed image consists of counts of photons hitting the pixels along with their energy levels. In this case, colors can be represented by energy levels of photons hitting pixels, which essentially lead to another dimension standing for the color; see White and Ghosal (2013). Then we can treat the observation as a 3D input and use the 3D Bayesian CRP in Section 4 to denoise the image.

4 Extension to 3D images

Due to the adjustability of the CRP and multivariate Gaussian distributions to higher dimensions, our method can be extended to data structures with higher dimensions such as 3-dimensional images. A colored image can sometimes be viewed as a 3D image. For a 3D image, our method can be extended using the following modification. The multiscale levels of the data are obtained by grouping $8 = (2 \times 2 \times 2)$ children

Figure 5: Denoising for the colored Lena image with size 512 × 512 shown in (a). The noisy observation with standard deviation 0.2 is shown in (b). The denoised images are shown in (c) and (d) using the proposed Bayesian CRP method and the running median approach with the window length 5. When plotting the colored image, we truncate both the noisy and smoothed images by [0,1] as required by MATLAB. Both methods use 121 local shifts to remove artifacts.



(a) True

(b) Observation (MSE = 0.0399)

(c) Bayesian CRP (MSE = 0.0020)

(d) Running Median (MSE = 0.0038)

pixels into 1 parent block and the corresponding Gaussian assumptions are made on a vectorized array of means with length 8 instead of 4 as in the 2D case. An eight person CRP will be used instead of a four person CRP to create ties. We modify the CRP prior by removing the diagonally tied configurations as in the 2D case, which reduces the number of configurations from 4140 to 958. The posterior distributions and the final estimates are calculated by the same procedure as in the 2D case with minor changes. We shall summarize the modeling and estimation procedures for the 3D case as follows.

Let the observed data be $\boldsymbol{X} = (\!(X_{(j,k,p)})\!)$ where $j,k,p = 1,\ldots,2^n$. It is only for notational convenience that we assume the three dimensions have the same length. The different scales of an image $\boldsymbol{X}$ are defined as follows. In the lth scale of the image, the parent (j,k,p)th block pixel is split into 8 children of block-pixels at the $(l+1)$th scale, which can be formulated as

$$X_{l,(j,k,p)} = \sum_{j'=2j-1}^{2j} \sum_{k'=2k-1}^{2k} \sum_{p'=2p-1}^{2p} X_{l+1,(j',k',p')}, \tag{20}$$

where $l = 0,1,2,\cdots,L-1$, and $j,k,p = 1,\cdots,2^l$. Here $X_{L,(j,k,p)} = X_{(j,k,p)}$ and when $l = 0$, $X_{0,(1,1,1)}$ is the summation of all the entire image.

We use $\mathbf{X}^*_{l,(j,k,l)}$ to denote the vector of its children group

$$\begin{aligned}(X_{l+1,(2j-1,2k-1,2p-1)}, X_{l+1,(2j-1,2k,2p-1)}, X_{l+1,(2j,2k-1,2p-1)}, X_{l+1,(2j,2k,2p-1)},\\ X_{l+1,(2j-1,2k-1,2p)}, X_{l+1,(2j-1,2k,2p)}, X_{l+1,(2j,2k-1,2p)}, X_{l+1,(2j,2k,2p)}),\end{aligned}$$

and similarly for the μ and $\boldsymbol{\Sigma}$ parameters. Similar analysis as in the 2D case, we can establish the following multiscale statistical model given by:

$$\begin{aligned}\mathrm{P}(\boldsymbol{X}|\boldsymbol{\mu},\boldsymbol{\Sigma}) &= \mathcal{N}(X_{0,(1,1,1)}; \mu_{0,(1,1,1)}, \sigma_0^2)\\ &\times \prod_{l=0}^{L-1} \prod_{j=1}^{2^l} \prod_{k=1}^{2^l} \prod_{p=1}^{2^l} \mathcal{N}(\boldsymbol{X}^*_{l,(j,k,p)}; \frac{1}{8} X_{l,(j,k,p)} \mathbf{1}_8 + \boldsymbol{\xi}^*_{l,(j,k,p)}, \frac{\sigma_0^2}{8^l}\Sigma_0)\end{aligned} \tag{21}$$

where $\boldsymbol{\Sigma}_0 = \boldsymbol{I} - \mathbf{1_8}(\mathbf{1_8'}\mathbf{1_8})^{-\mathbf{1}}\mathbf{1_8'} = \boldsymbol{I} - \frac{1}{8}\mathbf{1}_8\mathbf{1}_8'$ and $\mathbf{1}_8 = (1,1,1,1,1,1,1,1)'$.

The prior distribution is a modified 8-person CRP. We remove all the configurations that have only diagonal ties and thus end up with 958 admissible configurations out of 4140. We rescale the probability for each configuration to make them sum to one. The constraint matrix $\boldsymbol{A}$ is obtained the same way as in the 2D case, and the estimation procedure is the same except that we need to replace $\frac{1}{4}$ by $\frac{1}{8}$ and $\mathbf{1}_4$ by $\mathbf{1}_8$ wherever they show up.

The 3D case uses 8-dimensional Gaussian vectors instead of 4-dimensional as used in the 2D case. This generalization makes computations much more challenging. However, the asymptotic convergence property of the posterior distribution established in Theorem 2 continues to hold by the same arguments, since the resulting model still belongs to a finite dimensional regular parametric family and the prior distribution is non-singular.

5 Simulation results for 3D images

The methods of TI-Haar, wedgelet, platelet, and BPFA are not yet developed for 3D images. Therefore we compare our method with some other approaches in this simulation. Mukherjee and Qiu (2011) proposed a 3D image denoising method via local smoothing and nonparametric regression (LSNR), and compared with other approaches through an extensive simulation. We follow the same simulation settings and compare the Bayesian CRP method with the simulation results presented in Mukherjee and Qiu (2011). Particularly, we use two artificial 3D images as follows. The two true image intensity functions are:

$$f_1(x,y,z) = -(x-\frac{1}{2})^2 - (y-\frac{1}{2})^2 - (z-\frac{1}{2})^2 + \mathbb{1}\{(x,y,z) \in R_1 \cup R_2\}, \tag{22}$$

where $R_1 = \{(x,y,z) : |x-\frac{1}{2}| \le \frac{1}{4},\ |y-\frac{1}{2}| \le \frac{1}{4},\ |z-\frac{1}{2}| \le \frac{1}{4}\}$ and $R_2 = \{(x,y,z) : (x-\frac{1}{2})^2 + (y-\frac{1}{2})^2 \le 0.15^2,\ |z-\frac{1}{2}| \le 0.35\}$;

$$f_2(x,y,z) = \frac{1}{4}\sin(2\pi(x+y+z)+1) + \frac{1}{4} + \mathbb{1}\{(x,y,z) \in S_1 \cup S_2\}, \tag{23}$$

where $S_1 = \{(x,y,z) : (x-\frac{1}{2})^2 + (y-\frac{1}{2})^2 \le \frac{1}{4}(z-\frac{1}{2})^2,\ 0.2 \le z \le 0.5\}$ and $S_2 = \{(x,y,z) : 0.2^2 \le (x-\frac{1}{2})^2 + (y-\frac{1}{2})^2 + (z-\frac{1}{2})^2 \le 0.4^2,\ z < 0.45\}$; here $\mathbb{1}$ is the indicator function.

We shall compare our method with LSNR (Mukherjee and Qiu, 2011), anisotropic diffusion (AD) in Perona and Malik (1990), total variation minimization (TV) in Rudin et al. (1992), optimized non-local means (ONLM) method Coupé et al. (2008) and the conventional running median (RM) method. The TV method is modified by Mukherjee and Qiu (2011) by minimizing a 3D-version of the TV criterion. The estimation of σ^2 in the Bayesian CRP method is conducted by the 2-means modification described in section 2.3. For all three cases, we consider two cases when noise $\epsilon \sim N(0, \sigma^2)$ is added with $\sigma = 0.1$ and 0.2. We consider two image sizes with $n = 64$ and $n = 128$. We use 100 replications for each setting.

From Table 6, we see that the proposed Bayesian CRP method is one of the best approaches among all six methods for both settings in terms of MSE. In fact, when $n = 64$ (both $f = f_1$ and $f = f_2$) and $n = 128$ ($f = f_2$), the Bayesian CRP method is significantly better than all the other methods presented here; for the scenario that $f = f_1$ and $n = 128$, the LSNR method has the same MSE as the Bayesian CRP method. The LSNR method is the second best performing approach here, but it is based on the vectorization of the 3D image thus probably destroys the spatial structure of a 3D image. In contrast, the Bayesian CRP method is based on 8-person blocks to take into account the spatial association, and is also invariant under rotations in each dimension.

In addition to the two functions f_1 and f_2, we apply the Bayesian CRP method to a 3D Shepp-Logan phantom image. It is simulated as a 3D version of the commonly used 2D Shepp-Logan phantom, for which a MATLAB code is available (Schabel, 2005). This code creates an arbitrary number of ellipsoids in a 3D image, and is particularly useful as a standard 3D test image. We shall use the 3D Shepp-Logan phantom image to demonstrate both the visual and numerical performance of the proposed Bayesian CRP method, along with its computational efficiency.

Figure 6 shows five selected cross-sections of the 3D Shepp-Logan image. We can see that the Bayesian CRP method successfully recovers most of the key features of the

Table 6: 3D denoising for two images f_1, f_2 in terms of MSE ($\times 10^{-2}$). Bayesian CRP (the first row) uses $5 \times 5 \times 5$ local shifts and is based on 100 replications. The mean of 100 MSEs is reported. The maximum standard error for each column is reported in the last row. The numerical records for the other five methods to estimate f_1 and f_2 are from Mukherjee and Qiu (2011).

Method	$n = 64$				$n = 128$			
	$f = f_1$		$f = f_2$		$f = f_1$		$f = f_2$	
	$\sigma = 0.1$	0.2	$\sigma = 0.1$	0.2	$\sigma = 0.1$	0.2	$\sigma = 0.1$	0.2
Bayesian CRP	0.01	0.05	0.02	0.09	0.01	0.03	0.01	0.04
LSNR	0.03	0.08	0.06	0.13	0.01	0.03	0.02	0.06
TV	0.03	0.09	0.06	0.15	0.01	0.04	0.03	0.06
AD	0.06	0.35	0.07	0.38	0.03	0.20	0.04	0.22
ONLM	0.03	0.12	0.06	0.14	0.01	0.06	0.03	0.06
RM	0.22	0.33	0.11	0.26	0.08	0.19	0.06	0.14
SE (max)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

phantom; see the 3rd row for example. The Bayesian CRP method appears to recover boundaries since the CRP ties in the prior do not let it oversmooth. Some of the small ellipsoids are also recovered clearly, for example, the 5th row when $\sigma = 0.1$. When the noise becomes heavier such that features in the noisy observations are not clearly visible, the method cannot reconstruct all of them and may miss some of the small features (5th row when $\sigma = 0.2$). We also observe that when the feature is close to the background and with small size, then even light noise can make it difficult to be recognized (the top ellipsoids in the 4th row).

The numerical performance is presented in Table 7, with both the MSEs and computational time listed. It is clear that the Bayesian CRP method decreases the MSEs dramatically when applied to noisy phantoms. The computational cost is important for a method when addressing 3D images, because of the large size of the data. We can see that given smoothing parameters, the estimation for one local shift takes less than 50 seconds when the size $n = 64$, and about 4 minutes when $n = 128$. The optimization step is computationally intensive, which is typical for most methods to select tuning parameters. We use a simplex search to select M and τ in a wide range to make the algorithm completely data-driven and flexible for various data types. There are several variants we can use to improve the computational efficiency. For example, in practice we can specify a large value of M and τ for the first level of the image, and decrease the values by a factor 1/4 or 1/8 for each level. Another alternative could be adjusting the range to search in and making sure that the optimal values are not achieved at the boundary. In addition, at least two different parallel computing techniques are applicable to make it more computationally efficient benefiting from the characteristics of the Bayesian CRP method. First, given smoothing parameters, all local shifts are parallel to each other, therefore the *parfor* loops in MATLAB can be used to parallelize different shifts with very minor change in the original code. Second, the method is based on a

Figure 6: Performance of the Bayesian CRP approach for a 3D Shepp-Logan phantom image ($n = 128$). Each row corresponds to a cross-section of the image; the five rows are the 40th, 50th, 60th, 65th and 80th slice, and are selected to represent various features of a typical phantom image. The first column is the original image. Then 2nd column is the noisy observation with noise level 0.1, followed by the smoothed version in the 3rd column. The 4th and 5th column are the noisy observation and the smoothed image when the noise level is 0.2. Here the Bayesian CRP approach uses $5 \times 5 \times 5$ local shifts.

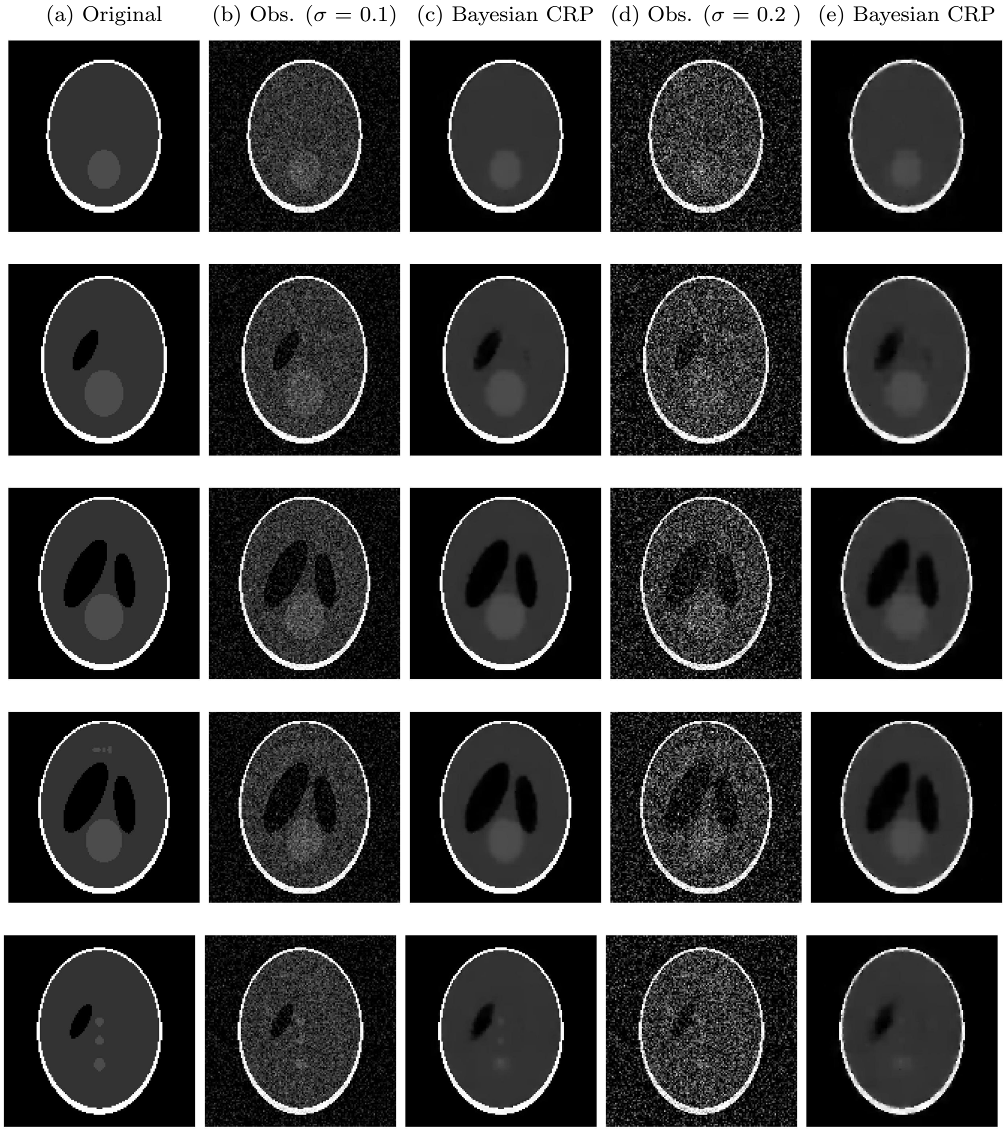


Table 7: Numerical performance of the Bayesian CRP on the 3D Shepp-Logan phantoms. The mean MSEs and average computational times are reported. The average computational time includes the Optimization time to select the smoothing parameters M and τ, Estimation time per shift given the selected smoothing parameters, and the Total time when using $5 \times 5 \times 5$ local shifts. The total time = the optimization time + the estimation time per shift × the number of shifts. 100 simulations are run and the standard errors of MSEs are given in the parentheses. Results are obtained without using any parallel computing techniques.

		$n = 64$		$n = 128$	
		$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.1$	$\sigma = 0.2$
MSE	Observation	0.0100 (0.0000)	0.0400 (0.0000)	0.0100 (0.0000)	0.0400 (0.0000)
	Bayesian CRP	0.0002 (0.0000)	0.0025 (0.0000)	0.0001 (0.0000)	0.0013 (0.0000)
Time	Optimization (h)	2.24	1.98	12.12	13.12
	Estimation/shift (s)	49.53	42.50	245.46	259.03
	Total (h)	3.96	3.46	20.64	22.11

large number of independent 4-person (2D) or 8-person (3D) blocks, which makes graphics processing unit (GPU) computing possible. GPU computing is applicable for both the optimization and estimation step. We report the results without exploiting parallel computing techniques, mainly for the convenience of the readers to make straight comparison, and also because parallel computing facilities may not be available to all. Our online MATLAB toolboxes have an option to incorporate the *parfor* loops to take advantage of parallel computing when available.

Acknowledgments

The authors wish to thank the reviewers and editors for their comments, which substantially improved the paper. The research reported here is partially supported by NSF grant number DMS-1106570.

References

Baddeley, A. (1992). "An error metric for binary images." *Proceedings of the IEEE Workshop on Robust Computer Vision*, 59–78. 746

Castro, R., Willett, R., and Nowak, R. (2004). "Coarse-to-fine manifold learning." In *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. (ICASSP'04)*. 742, 743

Coifman, R. and Donoho, D. (1995). "Translation-Invariant De-Noising." In Antoniadis,

A. and Oppenheim, G. (eds.), *Wavelets and Statistics*, volume 103 of *Lecture Notes in Statistics*, 125–150. Springer New York. 743

Coupé, P., Yger, P., Prima, S., Hellier, P., Kervrann, C., and Barillot, C. (2008). "An optimized blockwise nonlocal means denoising filter for 3-D magnetic resonance images." *IEEE Transactions on Medical Imaging*, 27(4): 425–441. 751

Donoho, D. L. (1999). "Wedgelets: Nearly Minimax Estimation of Edges." *The Annals of Statistics*, 27(3): pp. 859–897. 733

Ferreira, M. A. and Lee, H. K. (2007). *Multiscale modeling: a Bayesian perspective*. Springer Verlag. 734

Hartigan, J. A. and Wong, M. A. (1979). "Algorithm AS 136: A k-means clustering algorithm." *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1): 100–108. 741

Huttenlocher, D., Klanderman, G., and Rucklidge, W. (1993). "Comparing images using the Hausdorff distance." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9): 850–863. 746

Jain, A. (1989). *Fundamentals of Digital Image Processing*. Prentice-Hall Information and System Sciences Series. Prentice Hall. 743

Kolaczyk, E. D. (1999). "Bayesian Multiscale Models for Poisson Processes." *Journal of the American Statistical Association*, 94: 920–933. 734, 737

Kolaczyk, E. D. and Nowak, R. D. (2004). "Multiscale likelihood analysis and complexity penalized estimation." *Annals of Statistics*, 32: 500–527. 734

Lagarias, J. C., Reeds, J. A., Wright, M. H., and Wright, P. E. (1998). "Convergence properties of the Nelder–Mead simplex method in low dimensions." *SIAM Journal on Optimization*, 9(1): 112–147. 740

Lindeberg, T. (1993). *Scale-Space Theory in Computer Vision*. Kluwer International Series in Engineering and Computer Science: Robotics: Vision, manipulation and sensors. Springer. 734

McLachlan, G. and Peel, D. (2000). *Finite Mixture Models*. John Wiley & Sons. 741

Mukherjee, P. and Qiu, P. (2011). "3-D image denoising by local smoothing and nonparametric regression." *Technometrics*, 53(2): 196–208. 750, 751, 752

Perona, P. and Malik, J. (1990). "Scale-space and edge detection using anisotropic diffusion." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7): 629–639. 751

Rudin, L., Osher, S., and Fatemi, E. (1992). "Nonlinear total variation based noise removal algorithms." *Physica D: Nonlinear Phenomena*, 60(1): 259–268. 751

Sanyal, N. and Ferreira, M. A. (2012). “Bayesian hierarchical multi-subject multiscale analysis of functional MRI data.” *NeuroImage*, 63(3): 1519–1531. 733

Schabel, M. (2005). “3D Shepp-Logan phantom.” http://www.mathworks.com/matlabcentral/fileexchange/9416-3d-shepp-logan-phantom. MATLAB Central File Exchange. 751

White, J. T. and Ghosal, S. (2011). “Bayesian smoothing of photon-limited images with applications in astronomy.” *Journal of the Royal Statistical Society, Series B* , 73: 579–599. 734, 737, 742

— (2013). “Denoising three-Dimensional and colored images using a Bayesian multiscale model for photon counts.” *Signal Processing*, 93: 2906–2914. 748

Willett, R. and Nowak, R. (2003). “Platelets: a multiscale approach for recovering edges and surfaces in photon-limited medical imaging.” *IEEE Transactions on Medical Imaging*, 22(3): 332–350. 742

— (2004). “Fast multiresolution photon-limited image reconstruction.” In *IEEE International Symposium on Biomedical Imaging: Nano to Macro, 2004*, 1192–1195. 734, 742, 743

Wilson, D., Baddeley, A., and Owens, R. (1997). “A new metric for grey-scale image comparison.” *International Journal of Computer Vision*, 24(1): 5–17. 746

Zhou, M., Chen, H., Paisley, J., Ren, L., Sapiro, G., and Carin, L. (2012). “Nonparametric Bayesian dictionary learning for sparse image representations.” *IEEE Transactions on Image Processing*, 21: 13—144. 742

Appendix

Proof of Theorem 1.

Proof. Denote the mean intensity at the (j,k)th pixel by $\mu_{j,k} = g(j/n, k/n)$. Note that the total number of blocks is $B = n^2/4$. By the definition, $s^2_{j,k}$ is the sample variance of $\{X_{j',k'}\}$, where $(j',k') \in C(j,k) = \{j' = 2j-1, 2j; k' = 2k-1, 2k\}$. Denote $\bar{X}_{j,k}$ as the sample mean of $X_{j',k'}$, where $j',k' \in C(j,k)$, then $X_{j',k'} - \bar{X}_{j,k}$ is distributed as $N(\mu_{j',k'} - \bar{\mu}_{j,k}, 3\sigma^2/4)$, leading to the equation that $\mathrm{E}(s^2_{i,j}) = \sigma^2 + \frac{1}{3}\sum_{j'=2j-1}^{2j}\sum_{k'=2k-1}^{2k}(\mu_{j',k'} - \bar{\mu}_{j,k})^2$, where $\bar{\mu}_{j,k}$ is the mean of $\mu_{j',k'}(j',k' \in C(j,k))$.

Notice that all the parent-child blocks can be categorized as two types: contained by one of the $\mathcal{D}^0_i$'s (type 1) or contained by more than one $\mathcal{D}_i$ (type 2). For any (j,k)th block belonging to type 1, the distance in a block is always less than or equal to the distance between the two diagonal elements, which is $2\sqrt{2}/n$. Because of the Lipschitz continuity on each $\mathcal{D}^0_i$ and the existence of only finitely many $\mathcal{D}_i$'s, there exists a constant c such that $|\mu_{j',k'} - \bar{\mu}_{j,k}| \le c/n$, where $j',k' \in C(j,k)$. Therefore, $|\mathrm{E}(s^2_{i,j}) - \sigma^2| \le 4/3 \times c^2/n^2 = O(n^{-2})$. For each (j,k)th block that belongs to type 2, $|\mu_{j',k'} - \bar{\mu}_{j,k}| \le 2m$ for $j',k' \in C(j,k)$, therefore $|\mathrm{E}(s^2_{i,j}) - \sigma^2| \le 16m^2/3 = O(1)$. Because $\mathcal{D}_i$ is convex, we have at most $k \times 4 \times n/2 = 2kn$ blocks to contain the boundaries of the $\mathcal{D}_i$'s, corresponding to a proportion of $2kn/B = O(n^{-1})$. Therefore, $|\mathrm{E}(\widehat{\sigma}^2) - \sigma^2| \le B^{-1}\sum_{j=1}^{n/2}\sum_{k=1}^{n/2}|\mathrm{E}(s^2_{j,k}) - \sigma^2| \le 1 \cdot O(n^{-2}) + O(n^{-1}) \cdot O(1) = O(n^{-1})$, indicating that $\widehat{\sigma}^2$ is an asymptotically unbiased estimator of σ^2.

We shall show that the variance of $\widehat{\sigma}^2$ converges to 0 to obtain consistency. It is sufficient to show that $\mathrm{Var}(s^2_{i,j})$ is bounded since $\widehat{\sigma}^2$ is the average of all $s^2_{i,j}$'s, which are independent. Recall that $X_{j',k'} - \bar{X}_{j,k}$ is normally distributed with uniformly bounded mean and constant variance $3\sigma^2/4$ for all blocks. Consequently $\mathrm{E}(X_{j',k'} - \bar{X}_{j,k})^4$ is bounded uniformly for all blocks. Notice that

$$3^2 s^4_{i,j} = \left(\sum_{j'=2j-1}^{2j}\sum_{k'=2k-1}^{2k}(X_{j',k'} - \bar{X}_{j,k})^2\right)^2 \le \sum_{j'=2j-1}^{2j}\sum_{k'=2k-1}^{2k}(X_{j',k'} - \bar{X}_{j,k})^4,$$

according to the Cauchy-Schwarz inequality. Hence it follows that $\mathrm{Var}(s^2_{i,j}) \le \mathrm{E}(s^4_{i,j})$ is bounded uniformly for all blocks, and thus $\mathrm{Var}(\widehat{\sigma}^2) = B^{-1}\mathrm{Var}(s^2_{i,j}) = O(n^{-2})$. Combining the facts that $\mathrm{E}(\widehat{\sigma}^2) - \sigma^2 = O(n^{-1})$ and $\mathrm{Var}(\widehat{\sigma}^2) = O(n^{-2})$, we obtain consistency of $\widehat{\sigma}^2$. □

The following two lemmas were used in the paper:

Lemma 1. Let $\boldsymbol{X}$ be an n-dimensional random vector, $\boldsymbol{A}$ is an $m \times n$ matrix and $\boldsymbol{c} \in \mathbb{R}^n$. If $\boldsymbol{X} \sim N(\boldsymbol{\mu}, \sigma^2\boldsymbol{I})$, then $\boldsymbol{X}|\{\boldsymbol{AX} = \boldsymbol{c}\} \sim N(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$, where

$$\begin{aligned}\boldsymbol{\mu}^* &= \boldsymbol{A}'(\boldsymbol{AA}')^{-1}\boldsymbol{c} + (\boldsymbol{I} - \boldsymbol{A}'(\boldsymbol{AA}')^{-1}\boldsymbol{A})\boldsymbol{\mu} \\ \boldsymbol{\Sigma}^* &= (\boldsymbol{I} - \boldsymbol{A}'(\boldsymbol{AA}')^{-1}\boldsymbol{A})\sigma^2.\end{aligned}$$

Lemma 2. Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be two n-dimensional multivariate normal random vectors such that $\boldsymbol{X}|\boldsymbol{Y} \sim N(\boldsymbol{c}+\boldsymbol{Y}, \boldsymbol{\Sigma}_1)$, and $\boldsymbol{Y} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_2)$, where $\boldsymbol{c} \in \mathbb{R}^n$ and $\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2$ are both $n \times n$ nonnegative definite matrices such that $\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2$ is positive definite. Then the marginal distribution of $\boldsymbol{X}$ is $N(\boldsymbol{c}, \boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma_2})$, and the conditional distribution of $\boldsymbol{Y}$ given $\boldsymbol{X}$ is $N(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$, where

$$\begin{aligned} \boldsymbol{\mu}^* &= \boldsymbol{\Sigma_2}(\boldsymbol{\Sigma_1}+\boldsymbol{\Sigma_2})^{-1}(\boldsymbol{X}-\boldsymbol{c}) \\ \boldsymbol{\Sigma}^* &= \boldsymbol{\Sigma_2}-\boldsymbol{\Sigma_2}(\boldsymbol{\Sigma_1}+\boldsymbol{\Sigma_2})^{-1}\boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma_2}(\boldsymbol{\Sigma_1}+\boldsymbol{\Sigma_2})^{-1}\boldsymbol{\Sigma}_1. \end{aligned}$$

Proofs. It is well known that if $\begin{pmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \end{pmatrix} \sim N\left(\begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix}\right)$ and $\boldsymbol{\Sigma}_{22}$ is nonsingular, then

$$\boldsymbol{X}_1|\boldsymbol{X}_2 \sim N(\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{X}_2-\boldsymbol{\mu}_2), \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}). \tag{24}$$

Lemma 1 can be obtained by first deriving the joint distribution of $\begin{pmatrix} \boldsymbol{X} \\ \boldsymbol{AX} \end{pmatrix}$ and then applying (24). For Lemma 2, the joint distribution of $\begin{pmatrix} \boldsymbol{X} \\ \boldsymbol{Y} \end{pmatrix}$ can be obtained by the multiplication of the conditional density of $\boldsymbol{X}|\boldsymbol{Y}$ and the density of the marginal distribution of $\boldsymbol{Y}$. When $\boldsymbol{\Sigma}_2$ is singular, we can just drop the dependent variable to make the covariance matrix nonsingular and apply the same argument. An alternative approach for a singular covariance matrix is to consider the density function with respect to the Lebesgue measure on the column space of $\boldsymbol{\Sigma}_2$, rather than the full n-dimensional Lebesgue measure. □