Bayesian Kernel Methods for Analysis of Functional Neuroimages

Ana S. Lukic, *Member, IEEE*, Miles N. Wernick*, *Senior Member, IEEE*, Dimitris G. Tzikas, Xu Chen, Aristidis Likas, *Senior Member, IEEE*, Nikolas P. Galatsanos, *Senior Member, IEEE*, Yongyi Yang, *Senior Member, IEEE*, Fuqiang Zhao, and Stephen C. Strother, *Member, IEEE*

***Abstract*—We propose an approach to analyzing functional neuroimages in which 1) regions of neuronal activation are described by a superposition of spatial kernel functions, the parameters of which are estimated from the data and 2) the presence of activation is detected by means of a generalized likelihood ratio test (GLRT). Kernel methods have become a staple of modern machine learning. Herein, we show that these techniques show promise for neuroimage analysis. In an on-off design, we model the spatial activation pattern as a sum of an unknown number of kernel functions of unknown location, amplitude, and/or size. We employ two Bayesian methods of estimating the kernel functions. The first is a maximum *a posteriori* (MAP) estimation method based on a Reversible-Jump Markov-chain Monte-Carlo (RJMCMC) algorithm that searches for both the appropriate model complexity and parameter values. The second is a relevance vector machine (RVM), a kernel machine that is known to be effective in controlling model complexity (and thus discouraging overfitting). In each method, after estimating the activation pattern, we test for local activation using a GLRT. We evaluate the results using receiver operating characteristic (ROC) curves for simulated neuroimaging data and example results for real fMRI data. We find that, while RVM and RJMCMC both produce good results, RVM requires far less computation time, and thus appears to be the more promising of the two approaches.**

***Index Terms*—Functional neuroimaging, kernel methods, relevance vector machine (RVM), reversible-jump Markov-chain Monte-Carlo (RJMCMC).**

Manuscript received December 28, 2006; revised March 14, 2007. This work was supported by the National Institutes of Health/National Institute of Neurological Disorder and Stroke (NIH/NINDS) under Grant NS34069 and Grant NS35273, in part by the National Institutes of Health/National Institute of Biomedical Imaging and BioEngineering (NIH/NIBIB) under Grant EB02013, the NIH under Grant EB002013 and Grant MH072580. *Asterisk indicates corresponding author.*

A. S. Lukic was with the Department of Biomedical Engineering, Illinois Institute of Technology, Chicago, IL, 60616, USA. She is now with Predictek, Inc., Chicago, IL 60616 USA.

*M. N. Wernick is with the Department of Electrical and Computer Engineering and Medical Imaging Research Center, Illinois Institute of Technology, Chicago, IL 60616 USA (e-mail: wernick@iit.edu).

D. G. Tzikas, A. Likas, and N. P. Galatanos are with the Department of Computer Science, University of Ioannina, Ioannina, GR 45110, Greece.

X. Chen and S. C. Strother are with the Rotman Research Institute, Baycrest and University of Toronto, Toronto, M6A 2E1 ON, Canada.

Y. Yang is with the Department of Electrical and Computer Engineering and Medical Imaging Research Center, Illinois Institute of Technology, Chicago, IL 60616 USA.

F. Zhao is with the Department of Neurobiology, University of Pittsburgh, Pittsburgh, PA 15203 USA.

Digital Object Identifier 10.1109/TMI.2007.896934

I. Introduction

The aim of a two-state neuroimaging study, using positron emission tomography (PET) or functional magnetic resonance imaging (fMRI), is to compare two groups of images (acquired in two different brain states) to identify brain regions that exhibit changes in response to some task or drug. The result is an activation pattern indicating the task- or drug-affected regions. One of the most important components of a neuroimaging study is the statistical method used to detect the activation pattern (see reviews in [1]–[4]).

Traditionally, these statistical methods aim to classify each pixel in the image as either activated or not. This is most commonly done by thresholding a statistical parametric map (SPM) which is often a t- or F- statistic calculated for each pixel. The main task then is to choose the appropriate threshold for a selected significance level. A popular approach to this problem is to apply results from random field theory [5]. In some methods, inferences are made on a pixel-by-pixel basis using only the properties of the null distribution and no attempts are made to include assumptions about the activation pattern [6]. More-advanced approaches, which consider clusters of activated pixels, have been proposed (e.g., [7]–[10]). Still, with no assumption about the distribution under the alternative hypothesis, these methods can yield the probability of the observed data in the absence of activation, but cannot estimate the probability that activation is present.

More recently, statistical methods for neuroimaging have been developed within the Bayesian framework (e.g., [11]–[14]). These methods typically require a model for the alternative hypothesis. In [15], parametric distributions were used to model a single pixel under the two hypotheses, but no prior spatial information was included. This work was extended in [16] wherein a model was formulated for a small region in the image (e.g., a 3×3 pixel window). A potential advantage of Bayesian methods is that they make it possible to estimate posterior probabilities, not just class labels. This comes with a certain computational cost, because most data models are not tractable analytically and some type of iterative procedure must be used. Posterior probability maps have been defined for the hierarchical linear observation model in [12] and [13] wherein the expectation-maximization algorithm was used to estimate the covariance of residuals at each level. A Markov random field model was proposed in [11] in which simulated annealing was used to find the maximum *a posteriori* (MAP) estimate of the activation map.

In this paper, we propose a Bayesian approach in which we model the activation pattern as a sum of *kernel functions*. We investigate two methods of estimating the parameters of these kernel functions: 1) a MAP estimation method based on a reversible-jump Markov-chain Monte-Carlo (RJMCMC) algorithm and 2) a relevance vector machine (RVM) [17], [18].

The RJMCMC approach was proposed by our group several years ago [19], and a similar formulation was independently developed by Hartvig [14]. The present paper expands on our initial work using the RJMCMC [19] and RVM methods [20], and compares both methods to other existing techniques.

Although the algorithm that was developed by Hartvig in [14] is based on the same principle as our RJMCMC method, the implementation is not the same. The method in [14] uses different priors from ours, and uses Gaussian-shaped kernels. In addition, the transition probabilities in [14] are different and follow the Geyer and Moller methodology [21], whereas our method follows more closely the methodology proposed by Green [22], [23]. The RVM approach, to our knowledge, has not before been applied to this problem in any way, except for our earlier work [20].

As we will explain, our approach consists of estimating the activation pattern using either the RJMCMC or RVM method, and then substituting the estimated pattern into a generalized likelihood ratio test (GLRT) [24]. The GLRT is a standard decision theory approach, which has been used before in various ways in functional neuroimaging. The t-test [25], [26] is itself a GLRT for making binary decisions from univariate data in the presence of signal-independent Gaussian noise. In [29], we showed that a GLRT based on kernels can perform exceedingly well in neuroimaging if provided with an appropriate data model. Different forms of GLRTs have been proposed in [27] and [28] for analyzing complex fMRI data. We have also successfully employed the GLRT strategy in object detection algorithms [30].

In the next section, we introduce the GLRT framework and data model. In Section III, we introduce each kernel method, then provide details of the algorithms in Section IV. In Section V, we describe our experimental results, and provide conclusions in Section VI.

II. Generalized Likelihood Ratio Test

Likelihood ratio tests (LRTs) are well known to be the optimal approach to hypothesis testing when the probability density functions (PDFs) of the observations are completely known under all the hypotheses [24]. For example, the Bayes-risk, Neyman-Pearson, and minimum-probability-of-error decision rules all have the form of a LRT, i.e.,

$$\Lambda(\mathbf{x}) = \frac{p(\mathbf{x}; H_1, \theta_1)}{p(\mathbf{x}; H_0, \theta_0)} \begin{array}{c} d_1 \\ > \\ < \\ d_0 \end{array} T \tag{1}$$

where $\mathbf{x}$ is a vector containing the observed data, d_j denotes deciding in favour of hypothesis H_j, θ_j is a vector of parameters of the PDF for H_j, and T is the decision threshold selected based on the decision strategy that has been adopted (e.g., to set a particular false-positive probability).

When the parameters θ_j of the PDFs are unknown (as in neuroimaging), the LRT cannot be specified exactly. In this case it is common instead to perform a GLRT, in which the unknown parameters are replaced with statistical estimates, i.e.,

$$\Lambda(\mathbf{x}) = \frac{p(\mathbf{x}; H_1, \hat{\theta}_1)}{p(\mathbf{x}; H_0, \hat{\theta}_0)} \begin{array}{c} d_1 \\ > \\ < \\ d_0 \end{array} T \tag{2}$$

where $\hat{\theta}_j$ is an estimate of θ_j. For example, the student t-test is a univariate GLRT for the case of signal-independent Gaussian noise when the unknown parameters are the means and (equal) variances of the PDFs. In the t-test [25], [26], these unknown population statistics θ_j are replaced by values $\hat{\theta}_j$ estimated from the data.

We now frame the problem of detecting the activation pattern in an on–off neuroimaging study as a GLRT. We assume that two sets of N images are acquired, one set representing a "control state" and the other representing a potentially "activated state." The test is whether to reject the null hypothesis that the activated state is the same as the control state. Denoting images by vectors composed by lexicographic ordering of the voxel values, we represent the two hypotheses as follows:

$$\begin{aligned} H_0 :\ & g_j^{(c)}(\mathbf{r}) = b(\mathbf{r}) + n_j^{(c)}(\mathbf{r}), \quad j = 1, \ldots, N \\ & g_j^{(a)}(\mathbf{r}) = b(\mathbf{r}) + n_j^{(a)}(\mathbf{r}), \quad j = 1, \ldots, N \\ H_1 :\ & g_j^{(c)}(\mathbf{r}) = b(\mathbf{r}) + n_j^{(c)}(\mathbf{r}), \quad j = 1, \ldots, N \\ & g_j^{(a)}(\mathbf{r}) = b(\mathbf{r}) + s(\mathbf{r}) + n_j^{(a)}(\mathbf{r}), \quad j = 1, \ldots, N \end{aligned} \tag{3}$$

where $\mathbf{r}$ is a vector representing the spatial coordinates in the image, $g_j^{(c)}(\mathbf{r})$ and $g_j^{(a)}(\mathbf{r})$ denote the control- and activation-state images, $b(\mathbf{r})$ represents the baseline spatial pattern, $n_j^{(c)}(\mathbf{r})$ and $n_j^{(a)}(\mathbf{r})$ represent the noise contributions to the control- and activation-state images, respectively, and $s(\mathbf{r})$ represents the spatial activation pattern that we are attempting to learn from the study.

Forming paired difference images $x_j(\mathbf{r}) = g_j^{(a)}(\mathbf{r}) - g_j^{(c)}(\mathbf{r})$, we can express the hypotheses as follows:

$$\begin{aligned} H_0 :\ & x_j(\mathbf{r}) = n_j(\mathbf{r}), \quad j = 1, \ldots, N \\ H_1 :\ & x_j(\mathbf{r}) = s(\mathbf{r}) + n_j(\mathbf{r}), \quad j = 1, \ldots, N' \end{aligned} \tag{4}$$

where $n_j(\mathbf{r})$ is a combined-noise image. If we knew the activation pattern $s(\mathbf{r})$, we might be able to perform an LRT, and thus obtain optimal detection performance. Of course, this is not possible in practice. However, we can perform a GLRT by first estimating $s(\mathbf{r})$ and then substituting this estimate into the likelihood ratio. We will see that this procedure is similar to a standard t-test, except that the method of estimating $s(\mathbf{r})$ using kernels is more sophisticated and appears to perform better.

III. Estimating the Activation Pattern Using Kernels

Estimation of the spatial activation pattern $s(\mathbf{r})$ is the principal goal of an on–off neuroimaging study. In this paper, we approximate this spatial pattern as a superposition of *kernel functions*. In essence, we are estimating the spatial activation pattern as a regression problem in the space domain.

In this paper, we study two kernel methods: the RVM and a MAP method based on reversible jump Markov chain Monte

Carlo (RJMCMC) estimation. In both methods, we model the activation pattern $s(\mathbf{r})$ as a superposition of kernel functions, i.e.,

$$s(\mathbf{r};\boldsymbol{\theta}) = \sum_{p=1}^{P} w_p K(\mathbf{r};\mathbf{c}_p, d_p) = \mathbf{K}^T(\mathbf{r};\mathbf{c},\mathbf{d})\mathbf{w} \tag{5}$$

where

$$\mathbf{K}(\mathbf{r};\mathbf{c},\mathbf{d}) = [K(\mathbf{r};\mathbf{c}_1, d_1),\\ K(\mathbf{r};\mathbf{c}_2, d_2)\cdots\cdots K(\mathbf{r};\mathbf{c}_M, d_M)]^T,$$

in which $K(\mathbf{r};\mathbf{c}_p, d_p)$ is the pth kernel function, $p = 1,\ldots,P$. The parameters associated with the pth kernel function are as follows: d_p is a kernel's width parameter, $\mathbf{c}_p = [\mathbf{c}_p^x\ \mathbf{c}_p^y]^T$ contains the coordinates of the kernel's center, and w_P is the kernel's weight (amplitude). For notational simplicity, these values are concatenated to form vectors as follows: $\mathbf{w} = [w_1\ w_2\ \cdots\ w_P]^T, \mathbf{d} = [d_1\ d_2\ \cdots\ d_P]^T$, and $\mathbf{c} = [\mathbf{c}_1^T\ \mathbf{c}_2^T\ \cdots\ \mathbf{c}_P^T]^T$; thus, the complete parameter vector is denoted by $\boldsymbol{\theta} = [\mathbf{w}^T\ \mathbf{c}^T\ \mathbf{d}^T\ P]^T$. In general, we do not know *a priori* the locations of the kernels, nor do we know how many there are. Therefore, these parameters must be estimated from the data. One can assume, as we do in our RJMCMC method, that the sizes of the kernels are unknown as well. However, this is not essential, because it is always possible to represent larger "blobs" as the superposition of several small ones.

One of the main challenges in this formulation is to avoid overfitting, i.e., a situation in which excessive small kernel functions are used to represent the activation pattern, thus slavishly fitting the noise. Due to their Bayesian approach, the RVM and RJMCMC methods are both very effective in limiting the number of kernel functions, thus leading to stable, reproducible patterns.

In the following sections, we describe the RJMCMC and RVM methods for estimating the parameters of the kernel representation of the spatial activation pattern.

A. RJMCMC Approach

In the RJMCMC approach, we assume that the number of kernel functions in the model is unknown, as are the kernels' weights, locations, and width parameters. We estimate these unknowns by maximizing their *a posteriori* probability distribution, i.e.,

$$\hat{\theta} = \arg\max_{\theta} L(\mathbf{X}|\theta)p(\theta) \tag{6}$$

where $p(\theta)$ is the prior distribution of $\theta, \mathbf{X} = [\mathbf{x}_1^T\ \mathbf{x}_2^T\ \cdots\ \mathbf{x}_N^T]$ is a concatenation of the observed difference images $\mathbf{x}_i = \mathbf{g}_i^{(a)} - \mathbf{g}_i^{(c)}$, where $\mathbf{g}_j^{(a)}$ are the activation-state images and $\mathbf{g}_j^{(c)}$ are control-state images, and $L(\mathbf{X}|\theta)$ is the likelihood of observing data $\mathbf{X}$ given the parameters in θ. The pixels in each image are rearranged into column vectors using lexicographic ordering so that $\mathbf{x}_i = [x_i(\mathbf{r}_1)\ x_i(\mathbf{r}_2)\ \cdots\ x_i(\mathbf{r}_M)]^T$, where $x_i(\mathbf{r}_m) = g_i^{(a)}(\mathbf{r}_m) - g_i^{(c)}(\mathbf{r}_m), m = 1,\ldots,M$, and M is the number of pixels in each image. Assuming the noise is Gaussian and independent across observed images we write the likelihood term as:

$$L(\mathbf{X}\,|\,\boldsymbol{\theta}) = \prod_{i=1}^{N}(2\pi)^{-M/2}|\mathbf{C}_n|^{-1/2}\\ \times \exp\left\{-\frac{1}{2}(\mathbf{x}_i - \mathbf{s}(\boldsymbol{\theta}))^T\mathbf{C}_n^{-1}(\mathbf{x}_i - \mathbf{s}(\boldsymbol{\theta}))\right\} \tag{7}$$

where $\mathbf{C}_n$ is the noise covariance matrix.

In RJMCMC, $\mathbf{C}_n$ is considered known; therefore, it must be estimated separately before the estimation of $\boldsymbol{\theta}$. In this work, we choose fixed priors for $\boldsymbol{\theta}$. Assuming that the parameters of the P kernel functions are mutually independent (both between and within kernels), we write the prior distribution of the parameter vector θ as

$$p(\theta) = p_p(P)\prod_{p=1}^{P} p_c(\mathbf{c}_p)p_d(d_p)p_w(w_p) \tag{8}$$

where $p_p(P)$ is a prior on the number of kernels used to approximate the activation map, $p_c(\mathbf{c})$ is a prior for kernel locations, and $p_d(d)$ and $p_w(w)$ are priors for diameter and weights, respectively. We assume uniform priors on the number of kernels P over the range $[0, P_{\max}]$ and a uniform prior for the locations within the set of all image pixels. As prior distributions for widths d and weights w, we use truncated Gaussian distributions having mean, variance, and support that are prespecified to reflect our expectation of "reasonable" estimates. Besides enforcing our prior knowledge about the unknown parameters, priors must also have a role as a complexity penalty term to ensure that we avoid overfitting.

Since we cannot maximize the posterior probability in (6) analytically, we turn to an algorithm that allows us to sample from this distribution even though the direct generation of samples from it is not possible. For this purpose, we make use of the MCMC methodology [31], and add a "reversible jump" feature that permits jumps between spaces of different dimension [22], [23], [31]. The details of the RJMCMC algorithm are given in Section IV-A.

B. RVM Approach

In our second approach, based on RVM, we assume there is one kernel function of a fixed, known width at every pixel in the image, i.e., $P = M$ and $\mathbf{c}_p = \mathbf{r}_p, p = 1,\ldots,P$. To avoid overfitting, we construct priors in such a way as to enforce sparse estimates of the unknown weights in $\mathbf{w}$, resulting in many weights being estimated as zero, thereby pruning the number of kernels appearing in the spatial pattern.

In RVM, we average all observed difference images and rewrite the likelihood as

$$L(\bar{\mathbf{x}}\,|\,\mathbf{w}) = (2\pi)^{-M/2}\,|\mathbf{C}_{\bar{n}}|\\ \times \exp\left\{-\frac{1}{2}(\bar{\mathbf{x}} - \mathbf{s}(\mathbf{w}))^T\mathbf{C}_{\bar{n}}^{-1}(\bar{\mathbf{x}} - \mathbf{s}(\mathbf{w}))\right\} \tag{9}$$

where $\bar{\mathbf{x}} = (1/N)\sum_{i=1}^{N}\mathbf{x}_i$ and $\mathbf{C}_{\bar{n}}$ denotes the covariance matrix of the noise in the average observed image $\bar{\mathbf{x}}$.

Direct estimation of the parameters of this model is not possible due to their large number as compared to the available data. Thus, we use a Bayesian methodology that considers many of these parameters as random variables, allowing us to impose priors on them.

More specifically, we assume a Gaussian prior distribution over the weight vector $\mathbf{w}$ as

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{p=1}^{M} N\left(w_p\,|\,0, \alpha_p^{-1}\right) \tag{10}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_M]$ is a vector of M hyperparameters determining the strength of the prior distribution on each basis function's weight. The hyperparameters in $\boldsymbol{\alpha}$ are also considered to be random variables and, since they are scale parameters, they are assigned gamma prior distributions

$$p(\alpha_p) = \prod_{p=1}^{M} \mathrm{Gamma}(\alpha_p\,|\,a, b). \tag{11}$$

Typically, no prior knowledge is available for the hyperparameters, thus we make the assigned hyperpriors noninformative by choosing small values for the parameters a and b, (e.g., $a = b = 10^{-4}$).

Given the Gaussian prior on the weights $p(\mathbf{w}|\alpha)$, it is not immediately obvious that the suggested model will result in sparse solutions. However, by integrating over the hyperparameters, we can compute the "true" weight prior $p(\mathbf{w}) = \int p(\mathbf{w}|\alpha)p(\alpha)d\alpha$. This integral yields a Student-t prior, which is well known to produce sparse representations since most of its mass is concentrated close to the origin or the axes of definition [17], thus encouraging the estimate of $\mathbf{w}$ to have a large number of near-zero elements.

Performing this integration and substituting the resulting Student- t prior for $\mathbf{w}$ into the posterior $p(\mathbf{w}|\bar{\mathbf{x}}) = L(\bar{\mathbf{x}}|\mathbf{w})p(\mathbf{w})$ would yield an approach that is very similar to the RJMCMC method, except that here we know the number of unknown parameters. In principle, we could use the MCMC algorithm to estimate these. However, in RVM, we instead exploit the hyperparameter structure by rewriting the parameter posterior as

$$p(\mathbf{w}, \boldsymbol{\alpha}\,|\,\bar{\mathbf{x}}) = p(\mathbf{w}|\bar{\mathbf{x}}, \boldsymbol{\alpha})p(\alpha|\bar{\mathbf{x}}) \tag{12}$$

where we explicitly acknowledge $\boldsymbol{\alpha}$ as an unknown to be estimated. The first term on the right-hand side of (12) is known and given by

$$\begin{aligned} p(\mathbf{w}|\bar{\mathbf{x}}, \boldsymbol{\alpha}) &= \frac{L(\bar{\mathbf{x}}|\mathbf{w})p(\mathbf{w}|\boldsymbol{\alpha})}{p(\bar{\mathbf{x}}|\boldsymbol{\alpha})} \\ &= \frac{L(\bar{\mathbf{x}}|\mathbf{w})p(\mathbf{w}|\alpha)}{\int L(\bar{\mathbf{x}}|\mathbf{w})p(\mathbf{w}|\alpha)d\mathbf{w}} \end{aligned} \tag{13}$$

where $L(\bar{\mathbf{x}}|\mathbf{w})$ and $p(\mathbf{w}|\boldsymbol{\alpha})$ are specified in (9) and (10), respectively. The second term on the right-hand side of (12) cannot be expressed analytically and it is approximated by a delta function at its mode [17], i.e.,

$$p(\boldsymbol{\alpha}|\bar{\mathbf{x}}) \approx \delta(\boldsymbol{\alpha}_{\mathrm{MP}}) \tag{14}$$

where $\boldsymbol{\alpha}_{\mathrm{MP}}$ is the mode of $p(\boldsymbol{\alpha}|\bar{\mathbf{x}})$. The details of the algorithm for estimating $\boldsymbol{\alpha}_{\mathrm{MP}}$ are given in Section IV-B.

IV. Algorithms

A. RJMCMC Algorithm

In this section, we describe our implementation of the RJMCMC algorithm for estimating the vector of model parameters θ by maximizing its *a posteriori* probability distribution in (6). Since we cannot maximize it analytically, we use a stochastic algorithm to draw samples from the posterior, then use these samples to estimate the mode (and thus the MAP estimate).

For convenience, we find the MAP solution by maximizing the natural logarithm of the likelihood, i.e.,

$$\hat{\boldsymbol{\theta}} = \arg\min_{0} \left\{ \frac{1}{2}\sum_{j=1}^{N}[\mathbf{x}_j - \mathbf{s}(\boldsymbol{\theta})]^T \mathbf{C}_n^{-1}[\mathbf{x}_j - \mathbf{s}(\boldsymbol{\theta})] - \log p(\boldsymbol{\theta}) \right\}. \tag{15}$$

By solving this optimization problem, we search for an activation map common to all N difference images. Our first choice for a kernel was a Gaussian function since it is a well-known fact that combinations of isotropic Gaussian functions can model arbitrarily shaped activations [41]. Unfortunately, this did not work very well and we decided to use a blurred pillbox function

$$K(\mathbf{r}; \mathbf{c}, d) = \xi * h(\mathbf{r}; \mathbf{c}, d) \tag{16}$$

where

$$h(\mathbf{r}; \mathbf{c}, d) = \begin{cases} 1, & \|\mathbf{c} - \mathbf{r}\| \leq d \\ 0, & \text{otherwise} \end{cases}. \tag{17}$$

In (16), $*$ denotes convolution, and ξ is the imaging system point spread function which can be assumed known or estimated from data. The parameters $\mathbf{c}$ and d will be estimated by RJMCMC. The imaging point spread function ξ is a Gaussian whose width we estimated separately in the previous study [35] and is equal to 6.2 mm.

We estimate the noise covariance matrix $\mathbf{C}_n$ based on estimates of the variance of the noise at each pixel and an estimate of the noise autocorrelation function. The details are given in the Appendix.

The RJMCMC method is an iterative algorithm for generating samples of random vectors of unknown length from a possibly complicated multivariate probability distribution. We will use this algorithm to generate samples of parameter vector θ from its posterior for the purpose of maximizing it.

The algorithm proceeds by randomly choosing one of the following operations at each iteration: 1) creation of a new kernel ("birth"), 2) deletion of a kernel ("death"), 3) merger of two kernels into one ("merge"), 4) splitting of a kernel into two ("split"), or 5) improvement of the parameter estimates without changing the parameter vector length ("update"). At each iteration of RJMCMC, a new parameter sample vector is proposed.

The acceptance ratio that governs the probability of acceptance of a proposed sample at iteration l is

$$R^l = \underbrace{\frac{\pi(\boldsymbol{\theta}^{l+1}, \mathbf{X})}{\pi(\Theta^l, \mathbf{X})}}_{\text{Target ratio}} \times \underbrace{\frac{\zeta(\boldsymbol{\theta}^l, step^{-1}\,|\,\boldsymbol{\theta}^{l+1}, \mathbf{X})}{\zeta(\boldsymbol{\theta}^{l+1}, step|\boldsymbol{\theta}^l, \mathbf{X})}}_{\text{Proposal ratio}} \times \underbrace{\left|\frac{\partial \boldsymbol{\theta}^{l+1}}{\partial \boldsymbol{\theta}^l}\right|}_{\text{Jakobian}} \tag{18}$$

where π is the so-called *target distribution* from which we wish to sample. In our application, this is the posterior distribution. Therefore, the target ratio is composed of likelihood-ratio and prior-ratio terms as follows:

$$\frac{\pi(\boldsymbol{\theta}^{l+1}, \mathbf{X})}{\pi(\boldsymbol{\theta}^l, \mathbf{X})} = \underbrace{\frac{L(\boldsymbol{\theta}^{l+1}|\mathbf{X})}{L(\boldsymbol{\theta}^l|\mathbf{X})}}_{\text{Likelihood ratio}} \times \underbrace{\frac{p(\boldsymbol{\theta}^{l+1})}{p(\boldsymbol{\theta}^l)}}_{\text{Prior ratio}} \tag{19}$$

where $\boldsymbol{\theta}^l$ is the value of the parameter vector at iteration l. In (18), $\zeta(\boldsymbol{\theta}^{l+1}, \text{step}|\boldsymbol{\theta}^l, \mathbf{X})$ is the probability that $\boldsymbol{\theta}^{l+1}$ will be proposed by selecting a certain *step* given the current state of the chain $\boldsymbol{\theta}^l$ and the observations $\mathbf{X}$. Finally, $step^{-1}$ denotes the inverse of a *step*, e.g., $birth^{-1} = death$. The proposal ratio in (18) is given by

$$\begin{aligned}&\frac{\zeta(\boldsymbol{\theta}^l, step^{-1}\,|\,\boldsymbol{\theta}^{l+1}, \mathbf{X})}{\zeta(\boldsymbol{\theta}^{l+1}, step\,|\,\boldsymbol{\theta}^l, \mathbf{X})} \\ &\quad = \frac{q(\boldsymbol{\theta}^l\,|\,\boldsymbol{\theta}^{l+1}, \mathbf{X}, step^{-1})p_s(step^{-1}\,|\,\boldsymbol{\theta}^{l+1})}{q(\boldsymbol{\theta}^{l+1}\,|\,\boldsymbol{\theta}^l, \mathbf{X}, step)p_s(step\,|\,\boldsymbol{\theta}^l)}\end{aligned} \tag{20}$$

where $q(\boldsymbol{\theta}^{l+1}|\boldsymbol{\theta}^l, \mathbf{X}, step)$ is the proposal distribution from which new parameters are sampled and $p_s(step\,|\,\boldsymbol{\theta}^l)$ is the probability that, out of the five possible steps, a particular one will be chosen given the current state of the chain.

All steps are equi-probable with the following exceptions: 1) if the current number of kernels in $\boldsymbol{\theta}^l$ is zero, only a birth step is possible, 2) if the current number of kernels in $\boldsymbol{\theta}^l$ is one, a merge step is not possible, or 3) if the current number of kernels in $\boldsymbol{\theta}^l$ is equal to some predefined maximum number, then birth and split steps are not possible.

Any choice of the proposal distribution q will produce samples from the desired target distribution, but the convergence time of the chain will not be the same for every choice. To create a new kernel in the birth step, we sampled the location, diameter and amplitude parameters independently, i.e.,

$$\begin{aligned} q_b(\boldsymbol{\lambda}\,|\,\boldsymbol{\theta}, \mathbf{X}) = &\; q_c(\mathbf{c}\,|\,\theta, \mathbf{X}, birth) \\ &\times q_w(w\,|\,\boldsymbol{\theta}, \mathbf{X}, birth) \\ &\times q_d(d\,|\,\theta, \mathbf{X}, birth) \end{aligned} \tag{21}$$

where $\boldsymbol{\lambda} = [w \quad \mathbf{c}^T \quad d]^T$ are parameters describing a new kernel. The location parameter $\mathbf{c}$ was sampled from the distribution that is proportional to the blurred current residual similarly to the method proposed in [23]

$$q_c(\mathbf{c}\,|\,\boldsymbol{\theta}^l, \mathbf{X}, birth) \propto \mathbf{y_c}\frac{1}{N}\left|\sum_{j=1}^{N}(\mathbf{x}_j - \mathbf{s}(\theta^l))\right| \boldsymbol{I}_{(\mathbf{c},\theta^l)} \tag{22}$$

where $\mathbf{y_c}$ is a row of the 2-D blurring matrix $\mathbf{Y}$ corresponding to the pixel at location $\mathbf{c}$, $I_{(\mathbf{c},\boldsymbol{\theta}^l)}$ is an indicator function equal to zero if location $\mathbf{c}$ is already a center of a kernel defined by $\boldsymbol{\theta}^l$ or if the value of blurred residual at $\mathbf{c}$ is smaller then the 75% of the maximum blurred average residual value. This last condition is introduced to speed the convergence of the chain by sampling only from locations with high residual.

The diameter d and amplitude a were sampled from proposal distributions equal to their prior distributions. In the death step, each kernel had an equal chance to be proposed for deletion

$$q(\boldsymbol{\lambda}|\left[\boldsymbol{\theta}_u^l \quad \boldsymbol{\lambda}\right], \mathbf{X}, \text{ death}) = \frac{1}{P^l} \tag{23}$$

where $\boldsymbol{\lambda}$ is a parameter vector of a kernel to delete, $\boldsymbol{\theta}_u^l$ are the other parameters not to be changed, and P^l is the number of kernels at iteration l. For both birth and death steps, the determinant of the Jacobian is equal to one.

If a split step is chosen, we select one of the current kernels for splitting. We calculate the parameters of the new kernels in the following way:

$$\begin{aligned} & w_1 = \frac{w^*}{4} + \frac{w^*}{2}u_1, \quad w_2 = \frac{w^*}{4} + \frac{w^*}{2}(1 - u_1) \\ & \mathbf{c}_1 = \mathbf{c}^* - [u_2 \quad u_3]\frac{\gamma w^*}{w_1}, \quad \mathbf{c}_2 = \mathbf{c}^* + [u_2 \quad u_3]\frac{\gamma w^*}{w_2} \\ & d_1 = u_4 d^* \frac{(w^*)^2}{(w_1)^2}, \quad d_2 = (1 - u_4)d^*\frac{(w^*)^2}{(w_2)^2} \end{aligned} \tag{24}$$

where $[w^* \quad \mathbf{c^*}^T \quad d^*]^T$ are the parameters of the kernel selected to be split, $[w_1 \; \mathbf{c}_1^T \; d_1]^T$ and $[w_2 \; \mathbf{c}_2^T \; d_2]^T$ are the parameters of two resulting kernels, $[u_1, u_2, u_3, u_4]$ are random numbers sampled independently from the uniform distribution $U_{[0,1]}$, and γ is a predefined coefficient. We chose $\gamma = 3$ in all our experiments. In the merge step two kernels to be merged are selected, and the parameters of the resulting kernel are calculated as follows:

$$\begin{aligned} w^m &= w_1 + w_2 \\ w^m \mathbf{c}^m &= w_1\mathbf{c}_1 + w_2\mathbf{c}_2 \\ a^{m^2} d^m &= a_1^2 d_1 + a_2^2 d_2 \end{aligned} \tag{25}$$

where $[w_1 \; \mathbf{c}_1^T \; d_1]^T$ and $[w_2 \; \mathbf{c}_2^T \; d_2]^T$ are parameters of the two kernels selected to be merged and $[w^m \; \mathbf{c}^{mT} \; d_2]^T$ are the parameters of the resulting kernel.

Unlike birth and death steps, split and merge steps require calculation of the Jacobian to maintain the equilibrium in probability during these transitions. For the split step, the determinant of the Jacobian is equal to

$$J_s = \frac{w^{*^9} d^* \alpha^2}{2(w_1 w_2)^4} \tag{26}$$

and the inverse of (26) is used in the merge step

$$J_m = \frac{2(w_1 w_2)^4}{w^{*^9} d^* \alpha^2}. \tag{27}$$

The update step makes no change in the parameter-space dimensionality. Its purpose is to improve the current estimate of the parameters. The parameters are updated one by one, dividing the update step in a number of substeps equal to the total current number of parameters to update, which is $3P^l$. At each of these substeps an update is proposed for only one parameter and the change is accepted or not according to the acceptance ratio

$$R_u = \underbrace{\frac{\pi([\boldsymbol{\theta}_u^l \quad \tilde{\lambda}])}{\pi([\boldsymbol{\theta}_u^l \quad \lambda])}}_{\text{Target ratio}} \times \underbrace{\frac{q_u(\lambda|[\boldsymbol{\theta}_u^l \quad \tilde{\lambda}])}{q_u(\tilde{\lambda}|[\boldsymbol{\theta}_u^l \quad \lambda])}}_{\text{Proposal ratio}} \times \underbrace{1}_{\text{Jacobian}} \tag{28}$$

where $\boldsymbol{\theta}_u^l$ is the part of θ_u that is kept constant while element λ is being updated, $\tilde{\lambda}$ is the proposed value for λ and is sampled from $q_u(\cdot|[\boldsymbol{\theta}_u^l \quad \lambda])$. To update a location, we sampled again from the distribution proportional to the residual but we restricted the possible choices only to the neighborhood of the current value

$$q_c^i(\mathbf{c}|\boldsymbol{\theta}^l, \mathbf{X}, \text{update}) \propto \mathbf{y_c}\frac{1}{N}\left|\sum_{j=1}^{N}(\mathbf{x}_j - \mathbf{s}(\theta^l))\right| I_{(\mathbf{c},\mathbf{c}^i,\varepsilon)} \tag{29}$$

where $I_{(\mathbf{c},\mathbf{c}^i,\varepsilon)}$ is the indicator function equal to one if location $\mathbf{c}$ is in the neighborhood of the location $\mathbf{c}^i$ being updated, ε is the parameter defining the neighborhood, and i is the index of the kernel for which the parameters are being updated. The proposed value for the update of the diameter and amplitude were sampled from their respective prior distributions centered around the current value of the parameter being updated.

At each step of the RJMCMC algorithm, one sample of $\boldsymbol{\theta}$ is generated. We allow the algorithm to run long enough for the sample distribution to converge to the target posterior distribution. We then choose the sample that has the maximum posterior probability. To determine the number of iterations, we experimented with different chain lengths and determined that the maximum almost always occurs within the first 3000 iterations. Since we run the algorithm for 50 times to estimate the receiver operating characteristic (ROC) curve we are also limited by the computational time needed to run longer chains. Therefore we fixed the chain length to 3000 iterations.

B. RVM Algorithm

In the RVM approach, we use Gaussian kernel functions of the form

$$K(\mathbf{r};\mathbf{c},d) = \exp\left\{-\frac{1}{2}d\|\mathbf{r}-\mathbf{c}\|^2\right\}. \tag{30}$$

We place one kernel at each pixel, thus the kernel locations in $\mathbf{c}$ are known. All the kernels are assumed to have the same width.

We start by looking at the terms that constitute the parameter posterior

$$p(\mathbf{w},\boldsymbol{\alpha}\,|\,\bar{\mathbf{x}}) = p(\mathbf{w}\,|\,\bar{\mathbf{x}},\boldsymbol{\alpha})p(\boldsymbol{\alpha}\,|\,\bar{\mathbf{x}}) \tag{31}$$

As shown earlier the first term is known

$$\begin{aligned} p(\mathbf{w}\,|\,\bar{\mathbf{x}},\boldsymbol{\alpha}) &= \frac{L(\bar{\mathbf{x}}\,|\,\mathbf{w})p(\mathbf{w}\,|\,\boldsymbol{\alpha})}{p(\bar{\mathbf{x}}\,|\,\boldsymbol{\alpha})} \\ &= \frac{L(\bar{\mathbf{x}}\,|\,\mathbf{w})p(\mathbf{w}\,|\,\boldsymbol{\alpha})}{\int L(\bar{\mathbf{x}}\,|\,\mathbf{w})p(\mathbf{w}\,|\,\boldsymbol{\alpha})d\mathbf{w}} \\ &= N(\mathbf{w}\,|\,\boldsymbol{\mu},\Sigma) \end{aligned} \tag{32}$$

in which

$$\begin{aligned} \Sigma &= \left(\boldsymbol{\Phi}^T\mathbf{C}_{\bar{n}}^{-1}\boldsymbol{\Phi} + \mathbf{A}\right)^{-1} \\ \boldsymbol{\mu} &= \Sigma\boldsymbol{\Phi}^T\mathbf{C}_{\bar{n}}^{-1}\bar{\mathbf{x}} \end{aligned} \tag{33}$$

where $\boldsymbol{\Phi}$ is the so-called "design matrix" of dimensions $M \times M$ and $[\boldsymbol{\Phi}]_{p,t} = K(\mathbf{r}_p;\mathbf{r}_t,d), p = 1,\ldots,M, t = 1,\ldots,M$ with $K(\mathbf{r}_p;\mathbf{r}_t,d)$ defined in (30), and $\mathbf{A} = \mathrm{diag}[\alpha_1 \ \alpha_2 \ \cdots \ \alpha_M]$.

To approximate the second term, we estimate $\boldsymbol{\alpha}_{\mathrm{MP}}$ as

$$\boldsymbol{\alpha}_{\mathrm{MP}} = \arg\max_{\boldsymbol{\alpha}} p(\alpha\,|\,\bar{\mathbf{x}}) = \arg\max_{\boldsymbol{\alpha}} p(\bar{\mathbf{x}}\,|\,\boldsymbol{\alpha})p(\boldsymbol{\alpha}) \tag{34}$$

where $p(\bar{\mathbf{x}}\,|\,\boldsymbol{\alpha})$ is known as the marginal, or type-II, likelihood [32] and is computed by marginalizing over the weights according to

$$p(\bar{\mathbf{x}}\,|\,\boldsymbol{\alpha}) = \int p(\bar{\mathbf{x}}|\mathbf{w})p(\mathbf{w}|\boldsymbol{\alpha})d\mathbf{w} \tag{35}$$

yielding

$$p(\bar{\mathbf{x}}\,|\,\boldsymbol{\alpha}) = N(0,\boldsymbol{\Psi}), \quad \boldsymbol{\Psi} = \mathbf{C}_{\bar{n}} + \Phi\mathbf{A}^{-1}\boldsymbol{\Phi}^T. \tag{36}$$

Unfortunately, $\boldsymbol{\alpha}_{\mathrm{MP}}$ cannot be computed analytically, so we use an iterative formula for its re-estimation. We perform the following minimization which is equivalent to the maximization in (34):

$$\begin{aligned} \boldsymbol{\alpha}_{\mathrm{MP}} &= \arg\min_{\boldsymbol{\alpha}}(-2\log p(\bar{\mathbf{x}}|\boldsymbol{\alpha})p(\boldsymbol{\alpha})) \\ &= \arg\min_{\boldsymbol{\alpha}}\Bigg(\log|\boldsymbol{\Psi}| + \bar{\mathbf{x}}^T\boldsymbol{\Psi}^{-1}\bar{\mathbf{x}} \\ &\quad + 2\sum_{p=1}^{M}(a\log\boldsymbol{\alpha}_p - b\boldsymbol{\alpha}_p)\Bigg) \end{aligned} \tag{37}$$

leading to the following iterative update equation [17]

$$\boldsymbol{\alpha}_p^{\mathrm{new}} = \frac{1+2a}{\mu_p^2 + \Sigma_{\mathrm{pp}} + 2b}, \tag{38}$$

where μ_p is the pth element of the posterior mean weight and $\Sigma_{p,p}$ is the pth diagonal element of the posterior weight covariance. Both μ_p and $\Sigma_{p,p}$ are evaluated from (33) using the current estimate for $\boldsymbol{\alpha}_{\mathrm{MP}}$.

A drawback of the above optimization method is the complexity of computing matrix $\boldsymbol{\Sigma}$ if the number of basis functions is large. Some of these computations can be avoided by pruning basis functions having amplitude that is estimated to be zero. However, initially there are N basis functions, and computation of $\boldsymbol{\Sigma}$ is time-consuming.

One can bypass this difficulty by initially assuming only one basis function, and then adding or deleting basis functions at each iteration [34]. It has been shown that this algorithm increases the marginal likelihood at each step. This is a very effective way to implement RVM because all quantities can be computed incrementally using their value from the previous iteration and a small update which is computed very efficiently.

Once we estimate $\boldsymbol{\alpha}_{\mathrm{MP}}$, we find the signal estimates from (5) using the maximum posterior estimates of $\mathbf{w}$. According to (36),

TABLE I
SUMMARY OF RJMCMC AND RVM ALGORITHM STEPS

	RJMCMC	**RVM**
Initialize:	• Choose a kernel function K which can have an arbitrary number of parameters (e.g. (16) and (17)). • Choose the maximum allowed number of kernel functions and the number of iterations to run. • Choose prior distribution p (e.g. Fig. 4) and proposal distributions q for each step (e.g. (21) – (25)). • Calculate difference images, (4) • Estimate the covariance matrix of individual difference image noise (Appendix A).	• Choose a kernel function K with all parameters except the amplitude fixed (e.g. (30)). • Choose the hyper-prior parameters a and b (11). • Calculate the average difference image $\overline{\mathbf{x}}$. • Estimate the covariance matrix of the average difference image noise (Appendix A). • Choose an initial value for the hyper-parameters $\boldsymbol{\alpha}$.
In Each Iteration:	• Randomly chose a step (*birth*, *death*, *split*, *merge* or *update*) • Sample new parameters from the proposal distribution q for the selected step. • Calculate the acceptance ratio in (18) and decide whether to accept proposed parameters.	• Update the estimate of hyper-parameters $\boldsymbol{\alpha}$, (38). See [34] for the details of a very efficient algorithm.
Calculate signal estimate:	• Select the sample with the maximum posterior probability. • Use (5)to estimate the signal.	• Calculate the estimate of $\mathbf{w}$ using (33) • Use (5) to estimate the signal.
Use (2) to calculate the likelihood ratio map		

TABLE II
PARAMETERS OF THE PHANTOM

Parameter	Value
Image dimensions	60×60 pixels
Pixel size	3.1mm
Phantom size	18 cm×15 cm
FWHM of noise correlation function induced by spatial smoothing*	6.2 mm
Standard deviation of the noise, relative to the baseline value	5%
Diameter of activations[#]	12.5mm
Mean activation amplitude, M	5% above baseline
Ratio of physiological variance to noise variance, V	0.1
Total number of images, $2N$[+]	20

*In PET this is induced by the reconstruction filter and in fMRI by post-reconstruction spatial smoothing.
#In PET, this should be interpreted as the apparent size of the activations after blurring by the imaging system.
+We studied simple block designs with N activation and N control images.

the maximum posterior estimate of $\mathbf{w}$ is μ given by (33) and evaluated using $\boldsymbol{\alpha}_{\mathrm{MP}}$.

C. Summary of the RJMCMC and RVM Algorithm Steps

Table I summarizes the steps of the RJMCMC and RVM algorithms as we have implemented them.

V. EXPERIMENTAL RESULTS

A. Synthetic Data

To evaluate the performance of the proposed methods and compare them with existing techniques, we developed a simple brain phantom. The values of the parameters used to construct the phantom, given in Table II, are based on a positron emission tomography (PET) neuroimaging study performed at the VA Medical Center, Minneapolis, MN [35]. Though the phantom parameters were deduced from a PET study, the values used are also representative of whole-brain, blood-oxygenation-level-dependent (BOLD) functional magnetic resonance imaging (fMRI) studies that have been spatially smoothed [36].

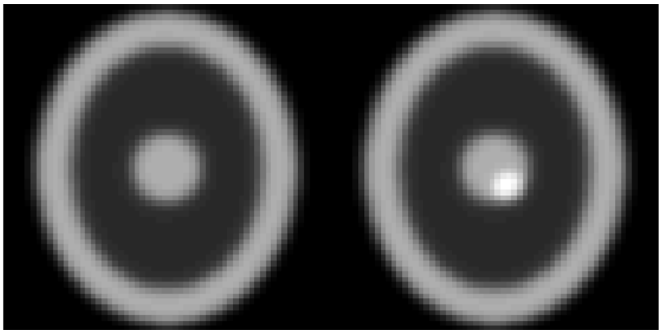

Fig. 1. Baseline (left) and activated phantom (right). Brighter areas of the baseline represent gray matter; darker areas simulate white matter. In the baseline image, the ratio of gray matter activity to white matter activity is 4:1.

In the phantom, the ratio of baseline activity in "gray matter" to that in "white matter" is 4:1 [37]. "Activated" brain images were obtained by introducing a circular-shaped "activation" with fixed size and with random, Gaussian-distributed amplitudes. A noise-free example of an activated image is shown in Fig. 1 (right).

The amplitude of the simulated activation was varied across images to simulate physiological variability between subjects or scans. The amplitude mean (activation strength) was specified

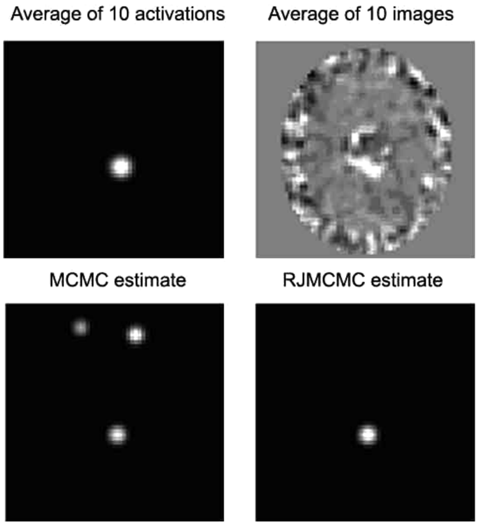

Fig. 2. RJMCMC synthetic data example showing: the average of 10 simulated noise-free activation patterns (upper left), the average of 10 noisy "activated" images (upper right), the activation pattern estimated by simple MCMC (without reversible jumps; lower left), and the activation pattern estimated by RJMCMC (lower right). Without reversible jumps, RJMCMC yields two false positive activations, whereas RJMCMC correctly detects a single activated region.

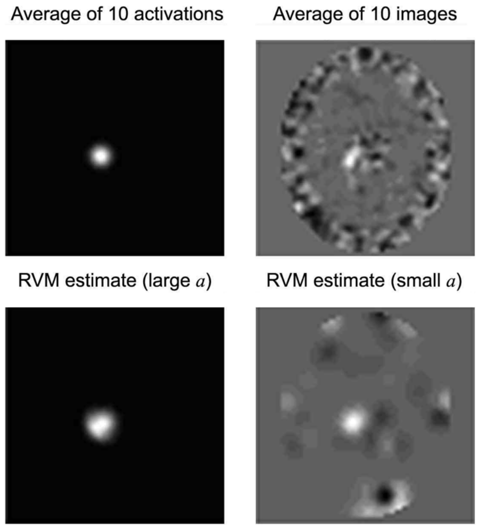

Fig. 3. RVM synthetic data example showing: the average of 10 simulated noise-free activation patterns (upper left), the average of 10 noisy "activated" images (upper right), the activation pattern estimated by RVM with $a = 1$ and $b = 0$ (lower left), and the activation pattern estimated by RVM with $a = 0.01$ and $b = 0$ (lower right). The RVM result obtained with a smaller value of a (flatter prior) is more noisy. The RVM result with the larger value of a correctly detects a single activated region.

in relation to the local value of the baseline, with proportionality constant M, i.e.,

$$E[\tilde{a}] = M\tilde{b} \tag{39}$$

where $\tilde{a}$ is the amplitude of the kernel, $\tilde{b}$ is the value of noise-free baseline image at the center pixel of activation, and $E[\cdot]$ denotes the expected value. The amplitude variance, denoted by σ_a^2, was specified in relation to the local noise variance ν^2 with proportionality constant V, so that

$$V = \frac{\sigma_a^2}{\nu^2} = \frac{\text{variance due to physiological variability}}{\text{variance due to image noise}}. \tag{40}$$

Unlike in our previous study in which the location of kernels was kept constant across realizations, in these experiments we introduced a small variation in the locations by allowing for either c_x or c_y to change by ±1 pixel independently with 50% probability.

B. RJMCMC and RVM Examples

In Fig. 2, we show results of an experiment that illustrates the value of the reversible jump feature of the RJMCMC when the complexity of the model is unknown. Fig. 2 (upper left) shows an image of the average of ten simulated noise-free activation patterns. We formed each pattern using only one kernel. We randomly varied the location and amplitude of the kernel from image to image to represent physiological variability between subjects or scans. Fig. 2 (upper right) shows the average of ten simulated "activated" images, which were obtained from the activation pattern in Fig. 2 (lower left) with colored noise added to simulate functional neuroimaging data. Fig. 2 (lower left) shows the activation pattern estimated by a MCMC method (without reversible jumps), assuming that the number of kernels was three. Finally, in Fig. 2 (lower right), we show the activation pattern estimated from the same data by the RJMCMC method, which clearly demonstrates the value of the ability of RJMCMC to "jump" between spaces of different dimensions. When the number of kernels is set incorrectly, simple MCMC (without reversible jumps) can produce erroneous activation patterns by fitting the noise in the data. RJMCMC is in comparison relatively immune to such problems.

Fig. 3 shows examples of RVM results. Fig. 3 (upper left) shows the average of 10 realizations of a simulated focal activation, and Fig. 3 (upper right) shows the average of 10 simulated noisy images. Fig. 3 (bottom row) shows the activation pattern estimated by the RVM method when the hyperparameters are $a = 1$ and $b = 0$ (lower left) and $a = 0.01$ and $b = 0$ (lower right). The lower value of a (flatter prior) gives a noisier result.

In the RJMCMC method, the kernel widths are estimated within the algorithm. In the RVM method, they must be selected in advance. In the simulation experiments described later, cross validation was used to optimize the RVM kernel width. In the real data experiments that follow, the RVM kernel width was fixed at the same value.

C. Prior Distributions

Prior distributions used for the kernel amplitude and width in all RJMCMC synthetic data experiments were truncated Gaussian distributions as shown in Fig. 4. The prior for the amplitude was a Gaussian centered at the true value of 0.2, with variance 0.05 and truncated at zero to avoid detection of negative activations. The prior for the diameter was a Gaussian centered at the true value of 12.5 mm, with variance 4 and truncated at 11 mm to prevent the algorithm from overfitting

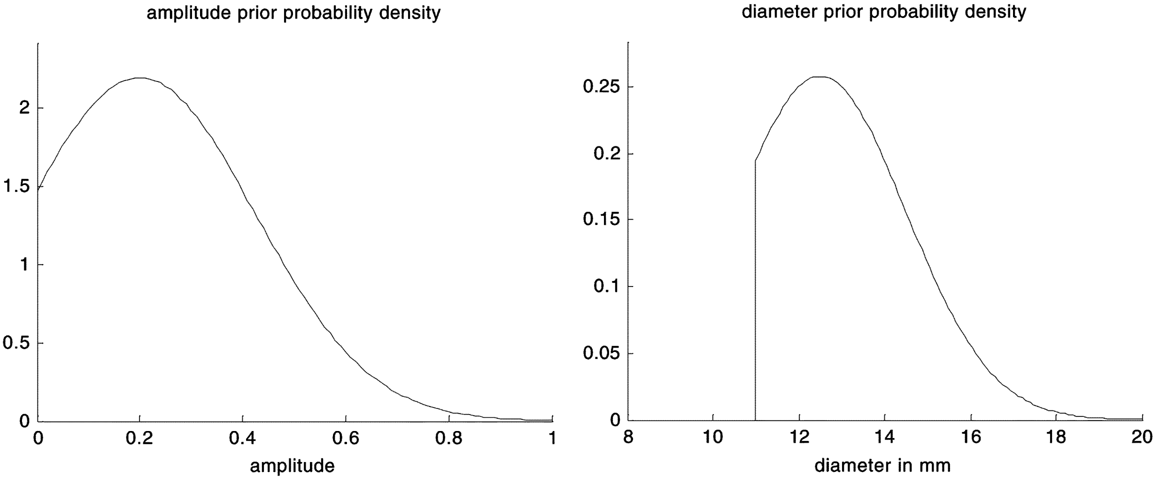

Fig. 4. Prior distributions of the kernel width parameter d in mm and amplitude a. The prior probability distribution of the width parameter is zero for values less than 11 mm. In this way, we prevented the algorithm from overfitting (i.e., using a large number of tiny basis functions).

(i.e., using a large number of tiny basis functions). The noise covariance matrix was assumed known.

D. Detection Performance Evaluation

Next, we provide the results of a comparison study that demonstrates the potential value of the proposed methods in the context of functional neuroimaging. To evaluate and compare performance, we used the area under the portion of the ROC curve where the false positive fraction (FPF) is between 0.0 and 0.1. We restricted our attention to this portion of the operating region so as to exclude the region of high FPF, which is not generally useful for neuroimaging. We normalize the area to the maximum possible value, which is 0.1, and express the value as a percentage, i.e.,

$$\tilde{A}_Z = 100 \times \frac{\int_{0.0}^{0.1} \mathrm{TPF(FPF)dFPF}}{0.1} \qquad (41)$$

where TPF and FPF denote true-positive fraction and false-positive fraction, respectively.

Each ROC curve was estimated using the LABROC1 software package [38] based on two groups of 50 samples that were obtained under null- and alternative-hypothesis conditions as given by (3). Each sample was generated from two groups of $N = 10$ images.

For each of these 10 image pairs, we formed the difference image, then used the RJMCMC algorithm to search all 10 difference images collectively for the presence of a common activation pattern. We then recorded the value of the RJMCMC output (which can be thought of as a fitted activation pattern) at location (33,27), where we knew the true activation to be located when it is present. To evaluate RVM, we calculated the average of all 10 difference images and recorded the value of the RVM signal estimate at the same location.

A comparison of detection performance is shown in Table III, which shows the value of $\tilde{A}_Z$ achieved by various methods, which are reviewed in detail in [29]. Table II shows that RJMCMC and RVM produced very similar performance, and significantly outperformed all of the other methods tested.

TABLE III
COMPARISON OF PERFORMANCES

Method	$\tilde{A}_z$ (%)*
RJMCMC	81.8
RVM	80.4
t-test, single-pixel variance estimates	63.4
SVD thresholding, column centering	62.4
t-test, pooled variance estimate	43.9
SVD thresholding, Fisher, row centering	38.7
SVD thresholding, Fisher, column centering	31.8
SVD thresholding, Fisher, double centering	31.1
SVD thresholding, row centering	25.2
SVD thresholding, double centering	16.0

* Normalized area under the ROC curve for false positive fraction between 0 and 0.1.

E. fMRI Cat Data

In this section, we present some preliminary results computed from actual functional magnetic-resonance imaging (fMRI) data to demonstrate that the RVM and RJMCMC methods can compute reasonable spatial patterns from real data. Thorough performance evaluations will be left for a future paper; our aim here is simply to establish the feasibility of kernel methods when applied to real data.

The data set was obtained by scanning an isoflurane-anesthetized cat [39] using gradient-echo data collection at 9.4T after injection of MION contrast. Images were obtained in a 1-mm-thick slice tangential to the surface of the cortex containing the visual area with in-plane resolution of 0.15 × 0.15 mm, $\mathrm{TE} = 10$ ms, and $\mathrm{TR} = 2$s.

Stimuli consisted of square-wave, high-contrast, moving gratings with low spatial frequency at two orthogonal orientations (45° versus 135°). Each epoch consisted of 10 baseline (20 s), 10 stimulus (20 s), and nine baseline scans. Baselines contained stationary grating patterns with the same orientation. Interleaved 45° and 135° epochs were repeated 40 times, each with a ~30 s break between epochs. Prior to the analysis, three transitional scans were removed from each segment of every epoch to ensure that we only use the scans acquired after the hemodynamic response (HDR) has reached the steady-state. Forty pairs of baseline-stimulus images were then obtained

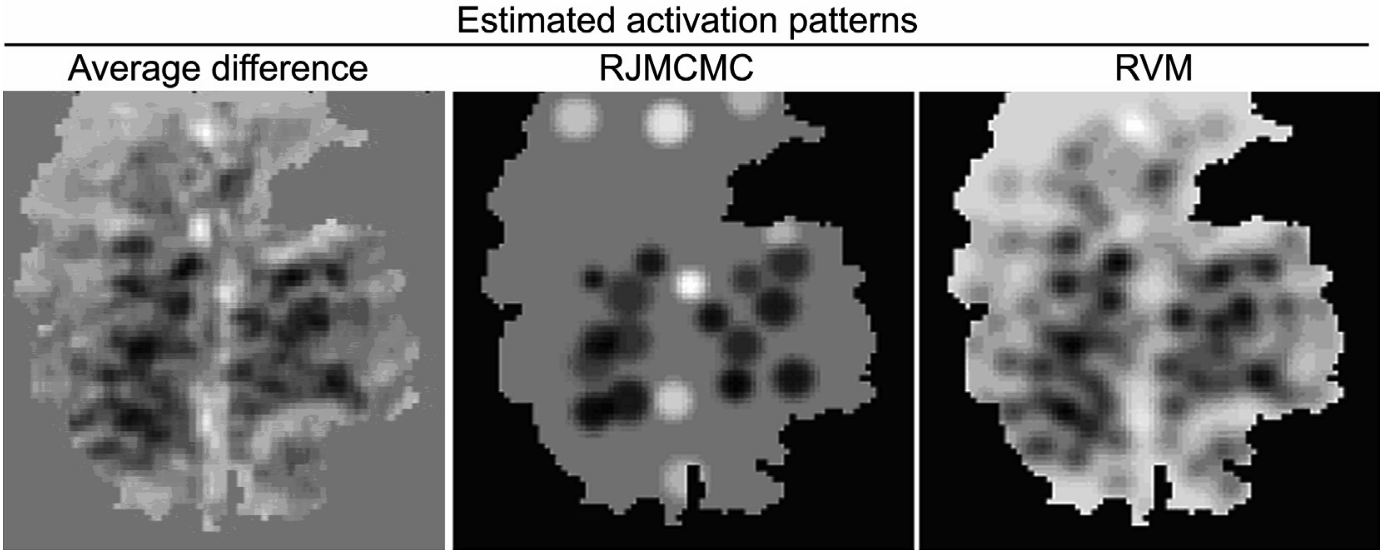

Fig. 5. Spatial activation patterns estimated as the average difference (left), and by RJMCMC (center) and RVM (right) methods.

by averaging over the remaining seven prestimulus baseline images and seven stimulus images in each epoch. Finally, 40 difference images were calculated and averaged to obtain a single average difference image.

As in the artificial data RJMCMC experiment, we used a truncated Gaussian with mean of 0.8 and variance of 5 as a prior for amplitude. The maximum amplitude was limited to 1.5. The positive part of this truncated Gaussian was then reflected about the vertical axis to allow for negative amplitudes with the same prior probability. The support of the prior for kernel width was restricted to the range from 2 to 8 pixels, within which it had a Gaussian shape with mean of 3 and variance 20. The maximum number of kernels was limited to 30 and the algorithm ran for 3000 iterations, which was found empirically to provide good results.

The output of the RVM and RJMCMC methods is an estimated spatial activation pattern $\hat{s}(\mathbf{r})$, which is a superposition of kernel functions having the parameters contained within the vector $\hat{\theta}$. Examples of these patterns for the cat data set are shown in Fig. 5, along with the average difference image for comparison.

After these patterns are estimated, the estimated parameter vector $\hat{\boldsymbol{\theta}}$ is substituted into the likelihood ratio in (2) using the signal model in (5). The result is a likelihood ratio value at every pixel, which can be displayed as an image. Images of the likelihood ratio from RJMCMC and RVM, and the t-statistic image from the t-test, are shown in Fig. 6.

In the t-statistic image in Fig. 6, we display only the values having $|t| > 5$. We determined that 62% of the pixel values within the brain mask region exceeded this threshold, and, in this image, all the surviving t-values were negative. To facilitate comparison to the likelihood ratio images (which are, by definition, nonnegative) we inverted the grayscale of the t-statistic image, so that black denotes $t = 0$ and white denotes the largest negative value of t. To display the RJMCMC and RVM likelihood ratio images in Fig. 6 in a comparable way, we set a threshold in each case that placed 62% of the pixels above threshold, which is the same fraction of activated pixels as in the t-image.

Comparing the results in Fig. 6, we see that the RJMCMC and RVM produced highly peaked activation regions, whereas the t-test produced a very dispersed pattern of activation. In these data, we expect activation in cortical columns, which would be difficult to identify in the t-test result, because of the broad extent of the activation regions. Therefore, one would need to rely mainly on further thresholding to identify the locations of the columns.

It is interesting to note that RJMCMC and RVM produced almost the same likelihood ratio image, with RVM giving somewhat higher emphasis to some of the activated regions. Thus, in both the simulated experiment summarized in Table I, and the real-data experiment shown in Fig. 6, the two methods produced very similar results. As we will discuss next, the RVM method requires a great deal less computation time than RJMCMC; therefore, it appears to be the more promising of the two algorithms.

F. Computation Time

A major advantage of the RVM method over the RJMCMC method is the relatively short computation time that RVM requires. The following are the computation times required to obtain the estimated activation patterns in the real data example. Using a MATLAB implementation of both algorithms on a computer with dual 3.2-GHz Xeon processors, the RVM analysis required 10 min to complete, whereas the RJMCMC method required more than one day (25 h, 15 min). Therefore, the RVM method is clearly the more practical approach, and the results appear to indicate that RVM performs about as well as, if not better than, the RJMCMC method.

VI. Conclusion

In this paper, we presented a Bayesian approach for analysis of functional neuroimages in which we model the activation pattern as a sum of kernel functions. We formulate a MAP estimation problem to determine the parameters of the model. We apply two different techniques, RJMCMC and RVM, to estimate the activation pattern, then use a GLRT to quantify the relatively likelihood of activation at each pixel.

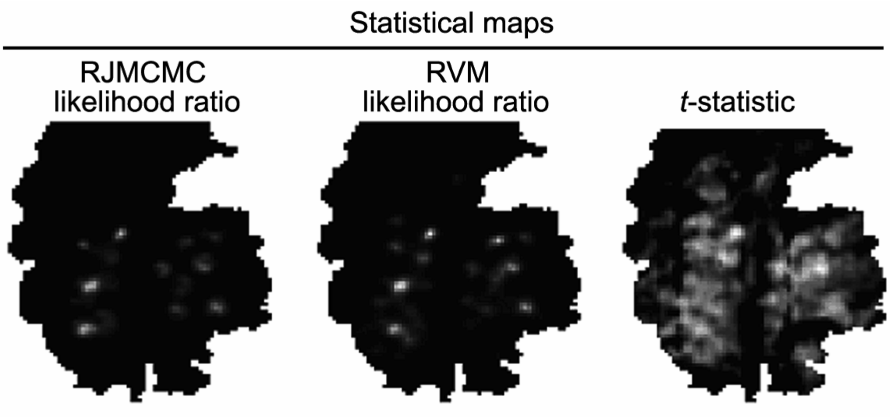

Fig. 6. Likelihood ratio images computed by RJMCMC and RVM, and the t-image (displayed on an inverted scale for ease of comparison). Each map shows the upper 64% of pixels, which corresponds to the fraction of pixels in the t-image having $|t| > 5$.

Using ROC analysis of simulated data, we compared the performance of these two methods to the others evaluated in a previous study [29]. In this experiment, the RJMCMC and RVM methods performed well, outperforming more-traditional approaches, such as t-test and SVD thresholding. However, further investigations will be needed to determine whether this finding generalizes to other data sets.

To demonstrate feasibility of the proposed methods, we applied them to real fMRI data and obtained satisfactory results. In future work, we will quantify performance of these techniques on real data by evaluating reproducibility and predictive power of the activation patterns using the NPAIRS [40] resampling framework. This should shed further light on the relative merits of the various techniques. This will also provide us a basis for optimizing the hyperparameters used.

We would like to point out that for the RJMCMC Gaussian kernels were initially tested that did not work well. Since the RJMCMC methodology gives us the capability to incorporate very easily the estimation of parameters, we changed the kernel to a blurred pillbox function where we estimate the width of the pillbox using the data. Clearly this can be viewed both as a strong point and as a shortcoming of the RJMCMC methodology. For the RVM methodology there is no simple and easy way to perform an analogous step. As stated in [17], to estimate the kernel width, cross validation methods could be employed which are computationally intensive, thus negating the main advantages of the RVM approach (speed and ease of implementation).

In our application, one might consider the use of more-complex kernels. However, in our current RJMCMC formulation, it is already difficult to estimate the parameter vectors; therefore, we expect that more complex kernels (with greater numbers of parameters) may not improve performance. The current RVM formulation does not include parameter other than the kernel weight, so flexible kernels cannot be used without a significant modification of the procedure.

Based on these initial studies, RVM appears to be a more promising approach than RJMCMC. RVM produced comparable performance to RJMCMC in simulations, and produced spatial patterns from real data that appear more plausible. RVM is also clearly favoured from a practical standpoint, as it requires much less computation time than RJMCMC (more than two orders of magnitude less time in our experiments).

Appendix
Estimating the Noise Covariance Matrix $\mathbf{C}_n$

We estimate the noise covariance matrix $\mathbf{C}_n$ based on estimates of the noise autocorrelation function $\mathbf{\Psi}$ given by

$$\mathbf{\Psi}_{p,q} = E[n_{i,j} n_{i+p,j+q}] \tag{42}$$

where $n_{i,j}$ denotes noise in the row i and column j in the image. We assume spatially stationary noise, therefore $\mathbf{\Psi}_{p,q}$ is independent of i and j. We model the noise as white, blurred by some unknown blurring kernel h

$$n_{i,j} = \sum_{m,n} z_{m,n} h_{i-m,j-n} \tag{43}$$

where $z_{m,n}$ is a unit variance Gaussian random variable in the row m and column n and h is a 2-D blurring kernel. In this model, all $z_{m,n}$ are independent, i.e., $E[z_{m,n} z_{m+k,n+p}] = \delta(k,p), \forall m,n$. If the pixels in the image are rearranged using lexicographical ordering, the blurring operation in (43) can be expressed as a matrix-vector multiplication

$$\mathbf{n} = H\mathbf{z} \tag{44}$$

where H is a matrix containing the elments of h reranged so that (43) is equivalent to (44). We can now express the noise covariance matrix $\mathbf{C}_n$ as

$$\mathbf{C}_n = E[\mathbf{n}\mathbf{n}^T] = E[H\mathbf{z}\mathbf{z}^T H^T] = HH^T. \tag{45}$$

Therefore, to estimate $\mathbf{C}_n$ we need to estimate the blurring kernel h. By substituting (43) into (42), it can be shown that the noise autocorrelation function is a convolution of h with itself, i.e., $\mathbf{\Psi} = h * h$, assuming that h is symmetric, i.e., $h_{m,n} = h_{-m,-n}$. Therefore, we can estimate h and in turn $\mathbf{C}_n$ by estimating $\mathbf{\Psi}$.

We estimate the elements of the noise autocorrelation function $\mathbf{\Psi}$ by averaging over local windows of size 3 × 3 pixels and over all N images

$$[\hat{\mathbf{\Psi}}]_{p,q} = \frac{1}{N|S_{p,q}|} \sum_{S_{p,q}} \sum_{j=1}^{N} x_j(\mathbf{r}_{m1}) x_j(\mathbf{r}_{m2}) \qquad p,q = -1,0,1 \tag{46}$$

where $S_{p.q}$ is a set of pixel pairs $m1$ and $m2$ such that the difference in their corresponding rows is equal to p and the difference in their corresponding columns is equal to q.

To estimate h from $\mathbf{\Psi}$, we recall that $\mathbf{\Psi} = h * h$ and use the convolution property of the Fourier transform:

$$F(\mathbf{\Psi}) = F(h)F(h) = (F(h))^2 \tag{47}$$

where F denotes the Fourier transform operator. Therefore ,h can be estimated as

$$\hat{h} = F^{-1}(\sqrt{F(\hat{\mathbf{\Psi}})}) \tag{48}$$

where the square root is calculated at each pixel and F^{-1} denotes the inverse Fourier transform operator. In practice, to inforce the symmetry of h, we estimate it as

$$\hat{h} = F^{-1}(\sqrt{|F(\hat{\mathbf{\Psi}})|}). \tag{49}$$

We then construct the matrix H from the elements of $\hat{h}$ and estimate $\mathbf{C}_n$ according to (45). This procedure guarantees that the estimate of $\mathbf{C}_n$ is positive definite.

Acknowledgment

The authors would like to thank Dr. S.-G. Kim for generously providing the CAT fMRI data used in this paper.

References

[1] K. J. Friston, "Imaging neuroscience: Principles or maps?," *Proc. Nat. Acad. Sci.*, vol. 95, no. 3, pp. 796–802, Feb. 3, 1998.

[2] J. Marchini and A. Presanis, "Comparing methods of analyzing fmri statistical parametric maps," *NeuroImage*, vol. 22, no. 3, pp. 1203–1213, 2004.

[3] K. M. Petersson, T. E. Nichols, J.-B. Poline, and A. P. Holmes, "Statistical limitations in functional neuroimaging. i. non-inferential methods and statistical models," *Phil. Trans. Roy. Soc.—Series B: Biol. Sci.*, vol. 354, no. 1387, pp. 1239–1260, 1999.

[4] K. J. Worsley, "An overview and some new developments in the statistical analysis of PET and fmri data," *Hum. Brain Mapp.*, vol. 5, no. 4, pp. 254–258, 1997.

[5] R. J. Adler, *The Geometry of Random Fields.*. New York: Wiley, 1981.

[6] K. J. Worsley, A. C. Evans, S. Marrett, and P. Neelin, "A three-dimensional statistical analysis for CBF activation studies in human brain," *J. Cereb. Blood Flow Metab.*, vol. 12, no. 6, pp. 900–918, 1992.

[7] K. J. Friston, K. J. Worsley, R. S. J. Frackowiak, J. C. Mazziotta, and A. C. Evans, "Assessing the significance of focal activations using their spatial extent," *Hum. Brain Mapp.*, vol. 1, pp. 214–220, 1994.

[8] J.-B. Poline and B. M. Mazoyer, "Analysis of individual positron emission tomography activation maps by detection of high signal-to-noise pixel clusters," *J. Cerebral Blood Flow Metabolism*, vol. 13, pp. 425–437, 1993.

[9] J.-B. Poline, K. J. Worsley, A. C. Evans, and K. J. Friston, "Combining spatial extent and peak intensity to test for activations in functional imaging," *NeuroImage*, vol. 5, no. 2, pp. 83–96, Feb. 1997.

[10] K. Worsley, S. Marrett, P. Neelin, A. Vandal, K. Friston, and A. Evans, "A unified statistical approach for determining significant signals in images of cerebral activation," *Human Brain Mapp.*, vol. 4, pp. 58–73, 1996.

[11] X. Decombes, F. Kruggel, and D. Y. v. Cramon, "FMRI signal restoration using a spatio-temporal markov random field preserving transitions," *NeuroImage*, vol. 8, no. 4, pp. 340–349, Nov. 1998.

[12] K. J. Friston and W. Penny, "Posterior probability maps and spms," *Neuroimage*, vol. 19, no. 3, pp. 1240–1429, 2003.

[13] K. J. Friston, W. Penny, C. Phillips, S. Kiebel, G. Hinton, and J. Ashburner, "Classical and bayesian inference in neuroimaging: Theory," *NeuroImage*, vol. 16, no. 2, pp. 465–483, Jun 2002.

[14] N. V. Hartvig, "A stohastic geometry model for functional magnetic resonance images," *Scand. J. Stat.*, vol. 29, no. 3, pp. 333–353, 2002.

[15] B. S. Everitt and E. T. Bullmore, "Mixture model mapping of brain activation in functional magnetic resonance images," *Hum. Brain Mapp.*, vol. 7, pp. 1–14, 1999.

[16] N. V. Hartvig and J. L. Jensen, "Spatial mixture modeling of fmri data," *Hum. Brain Mapp.*, vol. 11, pp. 233–248, 2000.

[17] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 2001.

[18] L. K. Hansen and C. E. Rasmussen, "Pruning from adaptive regularization," *Neural Comput.*, vol. 6, no. 6, pp. 1223–1232, 1994.

[19] A. S. Lukic, M. N. Wernick, N. P. Galatsanos, Y. Yang, and S. C. Strother, "A signal-detection approach for analysis of functional neuroimages," in *IEEE Nucl. Sci. Symp. Conf. Rec.*, 2001, vol. 3, pp. 1394–1398.

[20] D. Tzikas, A. Likas, N. P. Galatsanos, A. S. Lukic, and M. N. Wernick, "Relevance vector machine learning of functional neuroimages," in *2004 2nd IEEE Int. Symp. Biomed. Imag.: Macro Nano*, Arlington, VA, 2004, pp. 1004–1007.

[21] C. J. Grayer and J. Moeller, "Simulation procedures and likelihood inference for spatial point processes," *Scand. J. Statist.*, vol. 21, pp. 359–373, 1994.

[22] P. J. Green, "Reversible jump markov chain monte carlo computation and bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, Dec. 1995.

[23] G. Stawinski, A. Doucet, and P. Duvaut, "Reversible jump markov chain monte carlo for bayesian deconvolution of point sources," in *Proc. SPIE, Bayesian Inference Inverse Problems*, 1998, vol. 3459, pp. 179–190.

[24] S. M. Kay, *Fundamentals of Statistical Signal Processing—Detection Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1998.

[25] K. V. Mardia, J. T. Kent, and J. M. Bibby, *Multivariate Analysis.*. New York: Academic.

[26] K. J. Worsley, A. C. Evans, S. Marrett, and P. Neelin, "A three-dimensional statistical analysis for CBF activation studies in human brain," *J. Cereb. Blood Flow Metab.*, vol. 12, no. 6, pp. 900–918, 1992.

[27] J. Sijbers and A. J. den Dekker, "Generalized likelihood ratio tests for complex fMRI data: A simulation study," *IEEE Trans. Med. Imag.*, vol. 24, no. 5, pp. 604–611, May 2005.

[28] F. Y. Nan and R. D. Nowak, "Generalized likelihood ratio detection for fmri using complex data," *IEEE Trans. Med. Imag.*, vol. 18, no. 4, pp. 320–329, 1999.

[29] A. S. Lukic, M. N. Wernick, and S. C. Strother, "An evaluation of methods for detection of brain activations from PET or fMRI images," *Artificial Intell. Med.*, vol. 25, no. 1, pp. 69–88, 2002.

[30] A. Abu Naser, N. P. Galatsanos, and M. N. Wernick, "Methods of detecting objects in photon-limited images," *J. Opt. Soc. Am.*, vol. 23, no. 2, pp. 272–278, 2006.

[31] C. Andrieu, N. de Freitras, A. Doucet, and M. Jordan, "An introduction to MCMC for machine learning," *Mach. Learn.*, vol. 50, pp. 5–43, 2003.

[32] J. Berger, *Statistical Decision Theory and Bayesian Analysis*, 2nd ed. New York: Springer-Verlag, 1985.

[33] D. Mackay, "Bayesian interpolation," *Neural Comput.*, vol. 4, pp. 415–447, 1992.

[34] M. E. Tipping and A. Faul, "Fast marginal likelihood maximisation for sparse bayesian models," presented at the 9th Int. Workshop Artif. Intell. Stat., Key West, FL, 2003, unpublished.

[35] S. C. Strother and M. N. Wernick, Deducing statistical properties of brain activation from real data for use in constructing phantoms , 2001, Tech. Rep. [Online]. Available: http://www.iit.edu/~wernick/phantom-params.pdf

[36] M. R. Zaini, S. C. Strother, J. R. Anderson, J.-S. Liow, U. Kjems, C. Tegeler, and S.-G. Kim, "Comparison of matched bold and FAIR 4.0t-fMRI with [150] water PET brain volumes," *Med. Phys.*, vol. 26, no. 8, pp. 1559–1567, 1999.

[37] S. C. Strother, J. R. Anderson, X.-L. Xu, J.-S. Liow, D. C. Bonar, and D. A. Rottenberg, "Quantitative comparisons of image registration techniques based on high-resolution MRI of the brain," *J. Comput. Assist. Tomogr.*, vol. 18, no. 6, pp. 954–962, 1994.

[38] C. E. Metz, B. Herman, P.-L. Wang, J.-H. Shen, and B. Kronman, *LABROC1*. Chicago, IL: Dept. Radiol., Franklin McLean Memorial Research Inst., Univ. Chicago, 1993.

[39] F. Zhao, P. Wang, K. Hendrich, and S.-G. Kim, "Spatial specificity of cerebral blood volume-weighted fMRI responses at columnar resolution," *NeuroImage*, vol. 27, no. 2, pp. 416–424, 2005.

[40] S. C. Strother, J. R. Anderson, L. K. Hansen, U. Kjems, R. Kustra, J. Siditis, S. Fruitiger, S. Muley, S. LaConte, and D. A. Rottenberg, "The quantitative evaluation of functional neuroimaging experiments: The NPAIRS data analysis framework," *NeuroImage*, vol. 15, no. 4, pp. 747–771, Apr. 2002.

[41] C. Bishop, *Neural Networks For Pattern Recognition*. Oxford, U.K.: Oxford Univ. Press, 1995.