Published in final edited form as:
Neuroimage. 2012 February 15; 59(4): . doi:10.1016/j.neuroimage.2011.11.003.

Adaptive Kalman filtering for real-time mapping of the visual field

B. Douglas Ward[a,*], **John Janik**[b], **Yousef Mazaheri**[c], **Yan Ma**[d], and **Edgar A. DeYoe**[e]

[a]Department of Biophysics, 8701 Watertown Plank Road, Medical College of Wisconsin, Milwaukee, WI 53226 USA

[b]Stryker Incubation of Medical Technologies (IMT), 4100 E. Milham Rd, Kalamazoo, MI 49001 USA

[c]Department of Radiology and Medical Physics, Memorial Sloan-Kettering Cancer Center, 1275 York Avenue, New York, NY 10065 USA

[d]Biomedical Engineering Department, Marquette University, 1250 West Wisconsin Avenue, Milwaukee, WI 53233 USA

[e]Department of Radiology, 8701 Watertown Plank Road, Medical College of Wisconsin, Milwaukee, WI 53226 USA

Abstract

This paper demonstrates the feasibility of real-time mapping of the visual field for clinical applications. Specifically, three aspects of this problem were considered: (1) experimental design, (2) statistical analysis, and (3) display of results.

Proper experimental design is essential to achieving a successful outcome, particularly for real-time applications. A random-block experimental design was shown to have less sensitivity to measurement noise, as well as greater robustness to error in modeling of the hemodynamic impulse response function (IRF) and greater flexibility than common alternatives. In addition, random encoding of the visual field allows for the detection of voxels that are responsive to multiple, not necessarily contiguous, regions of the visual field.

Due to its recursive nature, the Kalman filter is ideally suited for real-time statistical analysis of visual field mapping data. An important feature of the Kalman filter is that it can be used for nonstationary time series analysis. The capability of the Kalman filter to adapt, in real time, to abrupt changes in the baseline arising from subject motion inside the scanner and other external system disturbances is important for the success of clinical applications.

The clinician needs real-time information to evaluate the success or failure of the imaging run and to decide whether to extend, modify, or terminate the run. Accordingly, the analytical software provides real-time displays of (1) brain activation maps for each stimulus segment, (2) voxel-wise spatial tuning profiles, (3) time plots of the variability of response parameters, and (4) time plots of activated volume.

[*]Corresponding author at: Department of Biophysics, Medical College of Wisconsin, 8701 Watertown Plank Road, Milwaukee, WI 53226, USA. Fax: +1 414 456 6512. ward@mcw.edu. .

Keywords

Kalman filtering; Nonstationary time series; Real-time imaging; fMRI; Retinotopy; Visual cortex; Human

Introduction

Although functional MRI (fMRI) has a long history of scientific applications in neuroscience, translating this research into clinical applications has lagged considerably. In order to make a successful transition from research lab to medical clinic, fundamental changes are required in: (1) experimental design, (2) statistical analysis, and (3) display of results.

Many clinical (as well as research) applications require experimental paradigms having multiple, simultaneous test conditions. An important example of this requirement is mapping of the visual field, where visual activation corresponding to multiple radial wedge locations, multiple concentric rings, etc., must be mapped onto the visual cortex. Therefore, any real-time algorithm must be capable of analyzing multiple simultaneous test conditions and allow the possibility that individual voxels may be responsive to multiple conditions, such as multiple visual field locations. Indeed, some voxels may be responsive to noncontiguous visual field locations (e.g., voxels straddling a sulcus), which further complicates the modeling process..

Proper experimental design is essential to achieving a successful outcome, particularly for real-time applications. We show that a random-block experimental design has less sensitivity to measurement noise, as well as greater robustness to error in modeling of the hemodynamic impulse response function (IRF) and greater flexibility than common alternatives. In addition, random encoding of the visual field allows for the detection of voxels that are responsive to multiple, not necessarily contiguous, regions of the visual field.

Due to the sequential nature of image acquisition and the large quantity of data, numerical calculations should be performed recursively for real-time display of results. That is, new data should be processed as it is acquired without the need to reprocess all of the previously acquired data. Also, to correct for subject motion, magnetic field perturbations, changes in subject's attention, etc., it is necessary to use statistical methods that can adapt to spurious signal perturbations, in real time, on a voxel-by-voxel basis. Due to its recursive nature, the Kalman filter is ideally suited for real-time statistical analysis of visual field mapping data. An important feature of the Kalman filter is that it can be used for nonstationary time series analysis. The capability of the Kalman filter to adapt, in real time, to abrupt changes in the baseline arising from subject motion inside the scanner and other external system disturbances is important for the success of clinical applications.

For clinical applications, it is desirable that brain images, functional activation maps, and statistical results be displayed in real time, so that the clinician can decide when sufficient data have been collected, if it is necessary to extend or to repeat the imaging run, or to change the parameters of the experiment itself Accordingly, the methodology described here provides real-time displays (updated approximately once per volume acquisition) of (1) brain activation maps for each stimulus segment, (2) voxel-wise spatial tuning profiles, (3) time plots of the variability of response parameters, and (4) time plots of activated volume.

Nonstationary time series analysis

A critical challenge for real-time analysis of fMRI data, and a major cause of data loss in clinical applications, is the *nonstationarity* of the data (i.e., the statistics of the random processes vary over time). Changes over time in the statistics of the time series may arise due to subject motion, variation in measurement noise level, “spikes” in the data, magnetic field distortions, change in subject attention level, slow changes in the vascular hemodynamic response due to physiological or neuronal processes, etc. Conventional fMRI time series analysis methods, such as cross-correlation, multiple regression, deconvolution, etc., assume that the underlying stochastic processes are stationary. Although attempts have been made to model temporally correlated error, such as using the auto-regressive moving average (ARMA) model, such models also assume that the time series are stationary (Chatfield, 2004; Shumway, 1988). An alternative method, the Kalman filter, is a powerful tool for analysis of *nonstationary* time series (Bozic, 1979; Gelb, 1974; Grewal and Andrews, 2001; Zarchan and Musoff, 2005). Furthermore, since the Kalman filter is a recursive algorithm, it has been successfully used in a wide variety of real-time applications. In this paper, we present an implementation of the adaptive Kalman filter tailored specifically for analysis of nonstationary fMRI time series data.

In order to illustrate the Kalman filter method, we present results from two fMRI visual field mapping experiments, which were conducted using random visual field excitation paradigms. For the first experiment, the visual field was mapped in one dimension using angular sectors (wedges). In the second experiment, the visual field was mapped in two dimensions (angle and eccentricity) simultaneously using radial sectors. Kalman filtering was implemented for simultaneous multiparameter estimation of the visual field response profile for each voxel of interest. In addition, real-time mapping of the visual field was simulated, post-experiment, along with a real-time graphical display of the visual field map. This was then used to explore optimization of the simulated scan duration, so that the run length is just sufficient to obtain statistically reliable activation maps. These experiments, and subsequent real-time simulations, demonstrated the feasibility of using Kalman filtering for real-time analysis and interpretation of the results from a visual field mapping paradigm for clinical applications.

A recurring theme of this paper is that it is advantageous to think of the fMRI imaging run not as a monolithic, *static* entity, but rather as an ongoing, *dynamic* process. From this point of view, run length is not a fixed constant but a variable, depending on the time required to achieve the desired parameter estimation accuracy. This has implications for the experimental design, data analysis methods and assumptions, and the graphical user interface, as explained below.

Application to visual field mapping

A conventional approach for fMRI mapping of the visual field uses a visual stimulus consisting of a rotating pie wedge or a set of expanding rings, each made up of an 8 Hz, flashing, black and white checkerboard pattern extending to 20° eccentricity. Typically, this method has been referred to as “temporal phase encoding” of the visual field. This is illustrated by the top row of Fig. 1, which shows the visual stimulus over 6 of 20 epochs separated by intervals of 2 s. In contrast, for “random-block encoding”, a particular region of the visual field, whether a “wedge” or a “ring”, is “ON” for fixed-length blocks of time, but these blocks are randomly placed within the allotted run time. This is illustrated by the bottom row of Fig. 1 for narrow wedge segments.

It is informative to consider the time sequence representation, or “stimulus functions”, for these respective designs. A stimulus function is a binary sequence that indicates the times

when a particular stimulus location is “ON”. For phase encoding of the visual field, the stimulus function for a particular wedge (or ring) location is a periodic function of time. This is illustrated in Fig. 2(a), which displays the time sequence of activations for each wedge location.

(Note: In this and some of the following illustrations, we have used particular color codings for different “segments” within the visual field, as indicated by the color wheel in Fig. 2. This is for illustrative purposes only, to help the reader in distinguishing and identifying different regions within the visual field. We reiterate that the subject in the scanner saw only a flashing, black and white checkerboard display, as illustrated in Fig. 1.)

From Fig. 2(a), we see that these stimulus functions are periodic, and more importantly, the stimulus functions for neighboring wedges are temporally phase shifted relative to each other. Therefore, a voxel is mapped to a single location in the visual field based on the time delay (or phase shift) of the measured fMRI time series relative to the input reference waveforms. Note, in particular, the similarity of the stimulus functions corresponding to neighboring wedge locations. This observation applies not only to the rotating hemi-field design but also to the expanding ring paradigm, as well as “moving bar” experiments (Dumoulin and Wandell, 2008). Although the “moving bar” paradigm does not use phase encoding, it is still the case that nearby locations in the visual field are activated by temporally similar stimulus functions.

One alternative to the phase encoding method is to use a randomized design. Typically, analysis of a randomized design is based on the assumption of system linearity (see, e.g., Boynton et al., 1996; Dale and Buckner, 1997). Here, we follow up on our previous work using a randomized design for fMRI mapping of the visual field (Ward et al., 2006). An important feature of the random-block encoding method is that stimulus functions corresponding to neighboring wedges are linearly independent. This is illustrated in Fig. 2(b). Note that this contrasts with the phase encoding method in which stimulus functions corresponding to neighboring wedges are related by a small, constant phase shift.

The advantages of the random-block encoding method, relative to the phase encoding method, are:

- Greater noise immunity. Since the reference waveforms for the phase encoding method are not as distinct as the random-block encoding waveforms, it is easier for measurement noise to make the response to one input stimulus sequence look like the response to another.
- The phase encoding method is more susceptible to modeling errors due to temporal variability, from voxel to voxel, in the hemodynamic response. Variability in the hemodynamic response has less effect on the random-block encoding method, which uses random code sequences, not time delay, for identifying the voxel visual field response.
- The phase encoding method is usually limited to a single “answer” for each voxel, due to the “winner-take-all” nature of the analysis that is typically used. That is, each voxel is identified as responding to a particular point location within the visual field. This method has been extended by using a two-dimensional circular Gaussian function to model voxel response to the visual field (the “population receptive field” (pRF) method; see Dumoulin and Wandell, 2008). However, this latter approach still assumes that there is a single center of response within the visual field for each voxel. Contrariwise, the random-block encoding method may indicate that a particular voxel is responsive to multiple (not necessarily contiguous) locations within the visual field. Furthermore, the iterative pRF method

is typically computationally intensive, making it unsuitable for real-time applications.

Another alternative to the phase encoding method is to use binary sequences of fixed length that have almost zero cross-correlation; e.g., m-sequences (Buracas and Boynton, 2002; de Visme, 1971; Hansen et al., 2004). However, m-sequences have certain limitations for fMRI experimental design:

- For m-sequences, the criterion for optimization is the correlation structure of the binary sequence itself. However, in fMRI experiments, the measured signal is not simply a scaled version of the input binary stimulus; rather, it is the hemodynamic response to the input binary sequences. From a signal processing perspective, a "good" ensemble of binary sequences is not necessarily a "good" ensemble of corresponding hemodynamic responses.
- By their nature, m-sequences are limited to a given set of fixed lengths. Due to the fixed length of the sequences, the m-sequence method has limited flexibility for experimental design. This is particularly significant for real-time applications since it may be necessary to make real-time changes to the experimental design (e.g., censoring of noisy measurements, extension or early termination of the imaging run, etc.).

For these reasons, we have used random-block encoding for generating the experimental designs considered here.

Theory

Our objective is to provide a real-time display of the visual cortex showing the moment-by-moment evolution of each voxel's response as the visual stimulus segments are presented randomly over time. At any given moment, each voxel's blood oxygen level dependent (BOLD) time course can be modeled as a weighted sum of the ideal BOLD time courses that would be evoked by each of the stimulus segments (plus a baseline and noise). At any given moment, then, the "state" of each voxel is essentially captured by the set of weights (gain factors) that are adjusted to yield a sum that best matches the actual measured response for each voxel up to that point in the scan. Conceptualizing the voxel responses in this way permits use of a Kalman filter algorithm (Kalman, 1960) to optimally adjust the set (vector) of weights recursively each time a new fMRI data sample is acquired. Thus, the recursive nature of the Kalman filter algorithm makes it particularly useful for such a real-time application. Furthermore, the ability of the Kalman filter to handle nonstationary noise processes will allow the system to compensate for unwanted perturbations such as head movements or other nonstationary factors.

Fig. 3 provides a schematized flow diagram of our implementation of the Kalman filter algorithm. In the following sections, we first define the various components needed to construct the response model, the weighting vector, and estimate its variability. We then describe how these components are used in combination with empirical data samples to recursively update the model and to adjust the effect of the current data sample on the model depending on how deviant it is from expectation.

To assist the reader, the numbers in parentheses next to the various blocks of Fig. 3 refer to equation numbers listed below in the text.

Experimental design (upper left yellow box, Fig. 3)

We will use the term "segment" to refer to a generic sub-region within the visual field. The experimental objective is to determine, for each voxel, which segment or segments are

responsible for stimulating a voxel's hemodynamic response. We specifically allow for the possibility that a voxel is responsive to multiple, not necessarily contiguous, segments within the visual field.

Suppose that the entire visual field is partitioned into *S* nonoverlapping segments. If it is assumed that during each volume scan duration (or T_R), a particular segment is either "ON" or "OFF" (e.g., flashing checkerboard pattern is present or not), then the stimulus function for the ith segment is represented by a binary sequence $b_i(t)$ of length N, where N is the number of MRI image volumes. Obviously, the total scan time is then: $T = N \times T_R$. Since there are *S* distinct segments, each characterized by a different binary stimulus function, the complete experimental design **D** is specified by:

$$\mathbf{D} = [b_1(t)\, b_2(t) \ldots b_S(t)] \in \mathbf{B}^{N \times S}, \quad (1)$$

where $b_1(t)$, $b_2(t)$, …, $b_S(t)$, are *S* independent binary sequences, and **B** = { 0, 1 } (i.e., **D** is an N × S matrix, each of whose elements is a 0 or a 1. Unless indicated otherwise, the variable *t* will indicate discrete time; i.e., *t* = 1,2,…,N.

The response of a linear time-invariant (LTI) system to an external stimulus is determined completely by the system's impulse response function (IRF), which, in the context of fMRI, is often referred to as the hemodynamic response function (HDR). The output of the system (voxel) is modeled by the convolution of the known input with the IRF. There is an extensive literature on linear modeling of the voxel IRF (see, e.g., Cohen, 1997; Friston et al., 1998; Glover, 1999).

Therefore, the voxel hemodynamic response to each visual field segment, not including a constant amplification factor, can be modeled as the convolution of *h*(*t*), the voxel IRF, with the corresponding input binary sequence; i.e.,

$$\mathbf{Y} = [y_1(t)\, y_2(t) \ldots y_S(t)] \in \mathbf{R}^{N \times S}, \quad (2)$$

where

$$y_j(t) = b_j(t) \otimes \mathbf{h}(\mathbf{t}), \mathbf{j} = \mathbf{1}, \ldots, \mathbf{S}. \quad (3)$$

(The notation $\mathbf{R}^{N \times S}$ represents the set of N × S matrices having real-valued elements.) Here, $y_j(t)$ is the canonical hemodynamic response to the *j*th visual field segment, $b_j(t)$ is the binary stimulus function for the jth segment, and ⊗ is the convolution operator. Since we model the fMRI measurement from each voxel as a scaled, linear combination of the $y_j(t)$ functions, the rows of the matrix **Y** serve as the measurement "gain" factors, which relate the measured signal to the strength of the voxel response to each stimulus. As we will see later, the rows of the **Y** matrix will be used as inputs to the Kalman filter.

Hemodynamic response model (box upper middle of Fig. 3)

Both voxel input and IRF are usually modeled using discrete time approximations, and the convolution is then performed numerically. This approach works well for offline analysis. However, this method is not well-suited for real-time applications where the input stimulus is not necessarily known in advance. For maximum flexibility in the experimental design, we used a method that allows for possible real-time changes in the input stimulus. Although real-time changes can also be modeled using discrete-time convolution, it is neither efficient nor convenient to do so since the convolution integral must be calculated separately at each time point. Instead, we modeled the voxel hemodynamic response using a linear, 2nd order

ordinary differential equation (ODE) with constant coefficients. The equation for the ODE model can be written:

$$\frac{d^2 y_j(t)}{dt^2} + 2\zeta\omega_n \frac{dy_j(t)}{dt} + \omega_n^2 y_j(t) = \omega_n^2 b_j(t-\tau), \quad (4)$$

where $b_j(t)$ = input binary stimulus function for the jth segment, $y_j(t)$ = voxel hemodynamic response to the jth stimulus, and ζ, ω_n, and τ are physical constants. (In this section, t represents continuous time.) Therefore, given the input stimulus $b_j(t)$, the hemodynamic response $y_j(t)$ can be found by numerically solving Eq. (4).

The physical constants in the above equation were chosen to emulate the Cohen gamma variate model (Cohen, 1997). For this reason, we needed the impulse response function corresponding to the linear system described by Eq. (4). The IRF $h(t)$ can be determined by applying the Laplace transform to the above equation, after substituting $\delta(t)$ for $b_j(t)$, and then taking the inverse Laplace transform (D'Azzo et al., 2003; Ogata, 1970). Depending on the value for ζ, for $t \geq \tau$, we have:

$$h(t) = \frac{\omega_n}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n(t-\tau)} \sin\left(\omega_n \sqrt{1-\zeta^2}\,(1-\tau)\right), \quad 0 \leq \varsigma < 1, \quad (5a)$$

$$h(t) = \omega_n^2 (t-\tau)\, e^{-\omega_n(t-\tau)}, \quad \varsigma = 1, \text{and} \quad (5b)$$

$$h(t) = \frac{\omega_n}{2\sqrt{\zeta^2-1}} \left(e^{-\left(\zeta-\sqrt{\varsigma^2-1}\right)\omega_n(t-\tau)} - e^{-\left(\zeta+\sqrt{\varsigma^2-1}\right)\omega_n(t-\tau)}\right), \quad \varsigma > 1. \quad (5c)$$

Nonlinear optimization was used to estimate the unknown parameters (ζ, ω_n, and τ) by fitting the above equation to the Cohen gamma variate model; this resulted in canonical parameter values: $\zeta = 0.76$, $\omega_n = 0.55\ \text{s}^{-1}$, and $\tau = 2.41$ s.

Of course, the constants that we have used in the ODE model may not be appropriate for all subjects, nor for all voxels within a single subject. Furthermore, no system is perfectly linear under all conditions. These and other unspecified errors in the system model may impact the accuracy of the results. However, as will be shown in later sections, the adaptive Kalman filter has a mechanism for adapting to errors in the model specification.

State-space model of the fMRI experiment

Baseline—The measured fMRI time series for a voxel contains, in addition to the signal of interest, a time-varying baseline offset. Therefore, in order to estimate the signal, it is necessary to accurately model the baseline. Typically, the baseline is modeled as a low-order polynomial function of time; e.g.,

$$\text{baseline}(t) = c_0 + c_1 t + \ldots + c_n t^n, \quad (6)$$

where c_0, c_1, …, c_n are constants that are estimated from the complete time series for each voxel. However, for real-time applications, we do not have the complete time series available. Furthermore, there may be abrupt changes in the baseline (e.g., due to subject head motion). Therefore, we will use a more flexible model. Specifically, we will use two state variables to model the baseline: state variable b_0 will contain the current estimate of the baseline (i.e., $b_0(t)$ = baseline at time t) and state variable b_1 will represent the current

baseline drift rate (i.e., $b_1(t)$ = baseline drift rate at time t). We will model changes in the baseline drift rate as being driven by white Gaussian noise. As a function of continuous time t, the state-space model for the baseline is given by the system of differential equations:

$$\begin{array}{ll} \frac{d}{dt} b_0(t) & = b_1(t) \\ \frac{d}{dt} b_1(t) & = w(t), \end{array} \quad (7)$$

where $w(t)$ represents white Gaussian noise. Since fMRI measurements occur at discrete times, separated by intervals of T_R seconds (although different slices are acquired at different slice-offset times, consecutive acquisitions of the same slice are separated by T_R seconds), we will use the following discrete time approximation to the above equations:

$$\begin{bmatrix} b_0(t+1) \\ b_1(t+1) \end{bmatrix} = \begin{bmatrix} 1 & T_R \\ 0 & 1 \end{bmatrix} \begin{bmatrix} b_0(t) \\ b_1(t) \end{bmatrix} + \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}, \quad (8)$$

where t is the discrete time index (t = 1, 2, …, N). In more compact notation,

$$\mathbf{b}(t+1) = \Phi_B \mathbf{b}(t) + \mathbf{w_B}(t), \quad (9)$$

where **b** is the baseline state vector; $\mathbf{\Phi}_B = \begin{bmatrix} 1 & T_R \\ 0 & 1 \end{bmatrix}$ is the baseline state transition matrix; T_R is the fMRI repetition time (i.e., time interval between measurements of the BOLD signal in each voxel); and $\mathbf{w}_B$ is the baseline process noise vector.

It can be shown that the covariance matrix of $\mathbf{w}_B$ is given by (see Zarchan, p. 156):

$$\mathbf{Q}_B(t) \equiv \mathbf{Cov}(\mathbf{w}_B(t)) = q_B(t) \begin{bmatrix} \frac{T_R^3}{3} & \frac{T_R^2}{2} \\ \frac{T_R^2}{2} & T_R \end{bmatrix}, \quad (10)$$

where $q_B(t)$ represents the variance of a white noise input that is driving random changes to the fMRI baseline. Note that $q_B(t)$ is a scalar, but can vary with time. So, by varying $q_B(t)$, the baseline process noise covariance matrix $Q_B(t)$ can be varied with time as well. Since $Q_B(t)$ represents random variation in the location of the baseline, variable $q_B(t)$ can be used as a "control knob" to incorporate external knowledge about changes in the variability of the baseline over time.

Stimulus response—The evoked response of each voxel will be modeled by an array of scalars ***a***(t) representing the amplitude of the response of that voxel to the corresponding stimulus segments:

$$\mathbf{a}(t) = [a_1(t)\, a_2(t) \ldots a_S(t)]^{\mathrm{T}}. \quad (11)$$

Additionally, since the fMRI signal typically includes a constant offset plus a baseline drift, these baseline parameters will be included in the model as well. Therefore, the system (i.e., voxel) *state vector* **x** is:

$$\mathbf{x}(t) = \left[\mathbf{b}(\mathrm{t})^{\mathrm{T}} \mathbf{a}(\mathrm{t})^{\mathrm{T}}\right]^{\mathrm{T}} = [b_0(t)\, b_1(t)\, a_1(t)\, a_2(t) \ldots a_S(t)]^{\mathrm{T}}, \quad (12)$$

where t is the discrete time index (t = 1,2,…,N), $b_0(t)$ is the baseline, $b_1(t)$ is the baseline drift rate, and $a_j(t)$, j = 1,…,S, is the amplitude of response to the jth segment (e.g., jth

wedge or *j*th ring) of the visual field. In addition, the state vector can be augmented to include variables representing "nuisance" regressors; e.g.:

$$\begin{aligned}\mathbf{x}(t) &= \left[\mathbf{b}(\mathrm{t})^{\mathrm{T}}\mathbf{a}(\mathrm{t})^{\mathrm{T}}\mathbf{m}(\mathrm{t})^{\mathrm{T}}\right]^{\mathrm{T}} \\ &= \left[b_0(t)\, b_1(t)\, a_1(t) \ldots a_S(t)\, m_1(t) \ldots m_{\mathrm{n}}(t)\right]^{\mathrm{T}},\end{aligned} \quad (13)$$

where $m_i(t)$, i = 1,…,n, are the nuisance regressors (e.g., head motion parameter estimates). The state *covariance matrix* **P** is then defined by:

$$\mathbf{P}(\mathrm{t}) \equiv \mathbf{cov}(\mathbf{x}(\mathrm{t})) = \mathbf{E}\left[(\mathbf{x}(t) - \mathbf{E}(\mathbf{x}(t)))\ (\mathbf{x}(t) - \mathbf{E}(\mathbf{x}(t)))^{\mathrm{T}}\right], \quad (14)$$

where **E** is the expectation operator.

State updating—The state vector is propagated forward in time according to the *system model*:

$$\mathbf{x}(\mathrm{t}) = \mathbf{\Phi}(\mathbf{t}-\mathbf{1})\,\mathbf{x}(\mathbf{t}-\mathbf{1}) + \mathbf{w}(\mathbf{t}-\mathbf{1}),\ \mathbf{w}(\mathrm{t}) \sim \mathbf{N}(\mathbf{0}, \mathbf{Q}(\mathbf{t})), \quad (15)$$

where **Φ**(*t*) is the *state transition matrix*, and ***w***(*t*) represents the *process noise vector*. The symbol *N* signifies that vector **w**(t) is normally distributed, with mean **0**, and covariance matrix **Q**(*t*). The expected value (as opposed to the observed value) of a voxel's visual field response is assumed to be (approximately) constant over the time course of the experiment. Hence, if the state vector is given by Eq. (12), the state transition matrix, Φ, takes the form:

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_B & 0 \\ 0 & \mathbf{I}_S \end{bmatrix}, \quad (16)$$

where Φ_B is defined above (Eq. (9)), $\mathbf{I}_S$ is the S × S identity matrix, and **0** represents an all-zero matrix of the appropriate size.

Nonstationarity—The *process noise covariance matrix* **Q**(*t*) can be used to model the imaging sequence as a nonstationary stochastic process. Specifically, we define **Q**(*t*):

$$\mathbf{Q}(t) = \begin{bmatrix} \mathbf{Q}_B(t) & 0 \\ 0 & q_S\mathbf{I}_S \end{bmatrix}, \quad (17)$$

where $\mathbf{Q}_B(t)$ is defined above (Eq. (10)). The constant diagonal terms q_S are used to model possible (small) changes over time in the amplitude of the response of a voxel to different visual field segments. Note that the covariance matrix **Q**(*t*) describes the amount of random variation in the state of the system itself, distinct from, and independent of, measurement error. Thus, **Q**(*t*) can be used to model perturbations of the fMRI signal due to various factors, such as head motion, which are *not* measurement noise *per se*. Importantly, **Q**(*t*) can also be used to allow for error in our specification of the state-space model.

fMRI measurement model (upper right of Fig. 3)

Using the definition of the state vector (Eq. (12)), and the hemodynamic response $y_j(t)$ (Eqs. (2) and (4)) corresponding to the *j*th visual field segment, the fMRI measurement *z*(*t*), at discrete time *t*, can be modeled by:

$$z(t) = b_0(t) + \sum_{j=1}^{S} a_j(t)\, y_j(t) + v(t), \quad (18)$$

where $v(t)$ represents the measurement noise at time t. That is, the fMRI measurement, for a particular voxel, is the sum of the current baseline $b_0(t)$ and a linear combination of the canonical responses to the individual visual field segments, plus measurement noise. In matrix notation, we can write Eq. (18) as follows:

$$\mathbf{z}(t) = \mathbf{M}(t)\,\mathbf{x}(t) + \mathbf{v}(t), \mathbf{v}(t) \sim N(0, \mathbf{R}(t)), \quad (19)$$

where **M** is the *measurement matrix*, and **R** is the *measurement covariance matrix*. **M** (which, for this application, is a row vector) relates the state vector $\boldsymbol{x}(t)$ (Eq. (12)) to the fMRI measurement $\boldsymbol{z}(t)$:

$$\mathbf{M}(t) = [1 0 y_1(t)\ y_2(t) \ldots y_S(t)], \quad (20)$$

where $y_j(t)$ = canonical hemodynamic response to the jth visual field segment, as calculated from Eq. (4). The *measurement covariance matrix* $\mathbf{R}(t)$, which for the present application is actually a scalar constant, represents the fMRI measurement noise variance.

Kalman filter implementation

Our objective is to provide a real-time display of the anatomical locations within the visual cortex that are responsive to individual subregions of the visual field. The above state-space model of the fMRI experiment allows implementation of a Kalman filter (see Refs. Bozic, Gelb, Grewal, Shumway, and Zarchan for derivations of the equations in this section). The Kalman filter is an algorithm for optimal estimation of the state vector **x**, where the state vector contains the unknown parameters of interest which, to reiterate, are the mean and linear trend of the baseline plus the gain factors for each segment's canonical response function (Eq. (12)). The recursive nature of the Kalman filter makes it useful for real-time applications. Furthermore, the ability of the Kalman filter to handle nonstationary noise processes is most appropriate for fMRI experiments. The information flow for our implementation of the Kalman filter is presented schematically in the block diagram of Fig. 3. (Again, numbers in parentheses next to blocks in Fig. 3 refer to equation numbers listed in the text).

As indicated at the lower left of the light blue shaded region in Fig. 3, the state vector **x** at time t can be predicted from the previous state estimate at time t — 1 using the state transition matrix **Φ** (Eq. (16)). Using the notation $\mathbf{x}(t|t-1)$ to represent the predicted state vector at time t, given measurements up through time t — 1, the state prediction equation is given by:

$$\mathbf{x}(t|t-1) = \mathbf{\Phi}(\mathbf{t}-\mathbf{1})\,\mathbf{x}(\mathbf{t}-\mathbf{1}|\mathbf{t}-\mathbf{1}), \quad (21)$$

where $\mathbf{x}(t-1|t-1)$ represents the updated state estimate following the measurement at time t — 1. We can use the predicted state vector $\mathbf{x}(t|t-1)$ to predict the next measurement $\mathbf{z}(t|t-1)$,

$$\mathbf{z}(t|t-1) = \mathbf{M}(t)\,\mathbf{x}(t|t-1). \quad (22)$$

The *measurement residual* **res**(t) is the difference between the actual measurement $\mathbf{z}(t)$ and the predicted measurement $\mathbf{z}(t|t-1)$:

$$\mathbf{res}\,(t) = \mathbf{z}\,(t) - \mathbf{z}\,(t|t-1) = \mathbf{z}\,(t) - \mathbf{M}\,(t)\,\mathbf{x}\,(t|t-1)\,. \quad (23)$$

Now as indicated by the light orange shading in Fig. 3, we can use the state transition matrix **Φ** and the process noise covariance matrix **Q** to obtain the predicted error covariance matrix **P**(*t* — 1|*t* — 1) from the previous state covariance matrix **P**(*t* — 1|*t* — 1):

$$\mathbf{P}(t|t-1) = \mathbf{\Phi}\,(\mathbf{t}-\mathbf{1})\,\mathbf{P}(\mathbf{t}-\mathbf{1}|\mathbf{t}-\mathbf{1})\,\mathbf{\Phi}^{\mathrm{T}}\,(\mathbf{t}-\mathbf{1}) + \mathbf{Q}\,(\mathbf{t}-\mathbf{1}) \quad . \quad (24)$$

The covariance matrix **E**(t) for the measurement residual **res**(t) is:

$$\mathbf{E}\,(t) = \mathbf{M}\,(t)\,\mathbf{P}(t|t-1)\,\mathbf{M}^{\mathrm{T}}\,(t) + \mathbf{R}\,(t) \quad . \quad (25)$$

The Kalman gain matrix **K** is then calculated based on the relative sizes of the predicted state covariance matrix **P**(*t*|*t* — 1) and the residual error covariance matrix **E**(t):

$$\begin{aligned} \mathbf{K}\,(t) \quad &= \mathbf{P}(t|t-1)\,\mathbf{M}^{\mathrm{T}}\,(t) / [\mathbf{E}\,(t)] \\ &= \mathbf{P}(t|t-1)\,\mathbf{M}^{\mathrm{T}}\,(t) / [\mathbf{M}\,(t)\,\mathbf{P}(t|t-1)\,\mathbf{M}^{\mathrm{T}}\,(t) + \mathbf{R}\,(t)\ ]\ . \end{aligned} \quad (26)$$

The Kalman gain **K**(*t*) determines how much influence the current measurement **z**(*t*) will have on the updated state estimate **x**(*t*|*t*). From this equation, we see that if measurement covariance matrix, **R**(*t*), is large compared with the predicted error covariance matrix, **P**(*t*|*t* —1) (i.e., if the current measurement uncertainty is large compared with the uncertainty in the state estimate), then the Kalman gain **K**(*t*) will be small. However, if **R**(*t*) is small compared with **P**(*t*|*t* — 1) (i.e., if the uncertainty in the measurement is small compared with the present uncertainty in the state estimate), then the Kalman gain **K**(*t*) will be relatively large.

The Kalman gain is then used in the following equation, which updates the state estimate **x** following receipt of the latest measurement **z**(*t*):

$$\mathbf{x}\,(t|t) = \mathbf{x}\,(t|t-1) + \mathbf{K}\,(t)\ [\mathbf{z}\,(t) - \mathbf{z}\,(t|t-1)]\,. \quad (27)$$

The notation **x**(*t*|*t*) means that this is the estimate of the state vector **x** at time *t*, given all fMRI measurements up to, and including, time *t*. For large Kalman gain, the state estimate is greatly affected by the empirical measurement. Conversely, if the Kalman gain is small, then the measurement has little influence on the state estimate. The updated state covariance **P**(*t*|*t*) is calculated using **K**(*t*) and **M**(*t*),

$$\mathbf{P}\,(t|t) = [\mathbf{I} - \mathbf{K}\,(t)\,\mathbf{M}\,(t)]\,\mathbf{P}\,(t|t-1)\,, \quad (28)$$

where **I** = identity matrix. The updated state vector **x**(*t*|*t*) and updated state covariance **P**(*t*|*t*) are then used as inputs to the next cycle of calculations. The block diagram shown in Fig. 3 summarizes the algorithm and illustrates the recursive nature of the Kalman filter.

Kalman filter initialization

To initialize the Kalman filter, it is necessary to specify the initial state vector **x**(0) and the initial state covariance matrix **P**(0). However, the algorithm is not sensitive to the choice of initial values **x**(0) and **P**(0, since their effect on the state estimate fades rapidly as more

fMRI measurements (i.e., images) are acquired. (Although P(0) should not be set so small that the algorithm is not able to accept new measurements.) If the fMRI data for each voxel is scaled by a (voxel-specific) constant, so that the initial baseline measurement is equal to 100, then subsequent visual segment response amplitude estimates will approximate "percent change relative to baseline". Therefore, if we subtract 100 from the baseline level, the initial state estimate consists of all zeros:

$$\mathbf{x}(0) = \left[0\,0\,\overbrace{0 \quad \cdots \quad 0}^{\mathrm{S}}\right]^{T}. \quad (29)$$

The initial state covariance matrix, **P**(0), is specified thus:

$$\mathbf{P}(0) = \begin{bmatrix} \sigma_p^2 & 0 & 0 & \cdots & 0 \\ 0 & \sigma_v^2 & 0 & \cdots & 0 \\ 0 & 0 & \sigma_a^2 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \sigma_a^2 \end{bmatrix}, \quad (30)$$

where the initial baseline, drift rate, and visual response amplitude variances are σ_p^2, σ_v^2, and σ_a^2, respectively. Note that the initial visual response amplitude variance σ_a^2 is the same for all segments since, prior to any measurements, our uncertainty about the amplitude of the visual response is the same for all segments. However, these diagonal elements will not, in general, be equal during the run since their respective binary stimulus functions will differ. Also note that although **P**(0) is a diagonal matrix, **P**(*t*) (*t*>0) is not, due to the cumulative effect of successive measurements.

Adaptive Kalman filter

One advantage of the Kalman filter approach is that it can be configured to adapt to nonstationary changes in the statistical properties of the fMRI signal through manipulation of the process noise covariance matrix, **Q**(*t*). Note that **Q**(*t*) and the measurement error covariance matrix **R**(*t*) must be specified at each discrete time point (i.e., for each volume acquisition), and have to be specified for each voxel. For the current application, **R**(*t*) is a scalar constant, i.e., **R**(*t*) = σ_r^2, where σ_r^2 is the estimated scanner noise variance (relative to a signal baseline level of 100).

The results presented in this paper used a constant measurement variance **R**, across all voxels, of **R** = 1.64 (relative to a baseline of 100, estimated from previous data). However, our current practice is to use a short null period of 8–12 TR, prior to the start of the external (visual) stimulus, to estimate the measurement variance on a voxelby-voxel basis. This seems to yield a minor improvement, by reducing false positives from "noisy" voxels.

From Eq. (23) above, the measurement residual, **res**(*t*) = **z**(*t*) — **M**(*t*)**x**(*t*|*t* — 1), is the difference between the actual measurement, and the measurement as predicted by the Kalman filter. The residual error covariance matrix **E**(*t*) (which, in this case, is a scalar) is the estimated uncertainty in the residual error. If the residual error is large relative to **E**(*t*), this may indicate an incorrect model specification. One way to compensate for this is to increase the process noise covariance (see Zarchan and Musoff, 2005). Since $q_B(t)$ (Eq. (10)) determines the baseline estimate portion of the process noise covariance matrix, we can use the residual error to adjust $q_B(t)$ in an *ad hoc* fashion, as follows:

$$\begin{array}{rl} \text{if} |res\,(t-1)\,| / \sqrt{\mathrm{E}\,(t-1)} > p99 & \text{then } \mathrm{q_B}\,(t) = 3.0 q_B\,(t-1) \\ \text{elseif} |res\,(t-1)\,| / \sqrt{\mathrm{E}\,(t-1)} > p95 & \text{then } \mathrm{q_B}\,(t) = 1.5 q_B\,(t-1) \\ \text{elseif} |res\,(t-1)\,| / \sqrt{\mathrm{E}\,(t-1)} > p50 & \text{then } \mathrm{q_B}\,(t) = 1.1 q_B\,(t-1)\,, \\ \text{elseif} |res\,(t-1)\,| / \sqrt{\mathrm{E}\,(t-1)} > p01 & \text{then } \mathrm{q_B}\,(t) = q_B\,(t-1)\,/3.0 \\ \text{elseif} |res\,(t-1)\,| / \sqrt{\mathrm{E}\,(t-1)} > p05 & \text{then } \mathrm{q_B}\,(t) = q_B\,(t-1)\,/1.5 \\ \text{elseif} |res\,(t-1)\,| / \sqrt{\mathrm{E}\,(t-1)} > p50 & \text{then } \mathrm{q_B}\,(t) = q_B\,(t-1)\,/1.1 \end{array} \quad (31)$$

where *p*99, *p*95, etc., are the corresponding z-scores. Note that the actual performance of the Kalman filter is not very sensitive to the details of the above schedule of adjustments. The purpose of the above is to make incremental changes to the process noise covariance matrix **Q**. If the residual error is “too large”, then the **Q** matrix is increased. If the residual error is “too small”, then the **Q** matrix is decreased.

Another method for adapting the Kalman filter to abrupt changes in the baseline is to use the (real time) estimated head motion parameters to determine the elements of the process noise covariance matrix **Q** (Ward et al., 2005). It should be emphasized that perturbation of the fMRI signal due to subject motion is not measurement noise per se, but rather represents a system disturbance, resulting in a long-term change in the baseline of the fMRI signal. After thresholding, the motion parameter estimates can be used to adjust elements of matrix **Q** in order to adapt to changes in the process noise.

The head motion parameters are available in real time from the AFNI (Cox, 1996) real-time plugin. After each volume acquisition, and real-time motion registration, the set of 6 motion parameters are (optionally) written to a separate data file. Therefore, after each TR, our real-time program (called SIRTAIN) reads the motion parameters (in addition to the registered volume of MRI data). Presently, the motion parameters are not used as nuisance regressors (since this would involve additional computational overhead). However, the program does calculate the finite-difference derivatives of the motion parameters. If any of the derivatives exceeds a fixed threshold, then the Q matrix baseline elements are set (for the current time point) to very large values. In other words, the Kalman filter is told that there may have been a large change in the signal baseline. Hence, the Kalman filter gain is increased, allowing the filter to adapt to any sudden change in the signal. This has the effect of “censoring” the current measurement, so that it does not influence the visual segment response amplitude estimates.

Other external information, in addition to the estimated motion parameters, can be used to adapt the Kalman filter to temporal changes in the process noise. For example, estimates of B-field distortion. Or, if an eye-tracker is available, the output from this device could be used as input to the adaptive Kalman filter in order to compensate for changes in the subject’s center-of-gaze.

Integrated fractional volume

One diagnostic application for real-time mapping of the visual field is the detection of scotomata, local regions of blindness. If a scotoma arises from damage to the visual system peripheral to, or within one or more, visual areas, then this will produce a corresponding “hole” in the cortical representation of the visual field. To provide a numerical criterion for detecting such holes, we define *integrated fractional volume* (*IFV*(*s*)) as the total volume of neural tissue in the visual cortex that is responsive to segment *s* of the visual field. This quantity can be estimated by summing the fractional volumes over all voxels *v*:

$$IFV\,(s) = \sum_{v} SRF\,(s,v) \cdot GMF\,(v) \cdot VoxVol, \quad s = 1, \ldots, S, \tag{32}$$

where *VoxVol* = volume of a single voxel; *GMF*(*v*) = gray matter fraction (i.e., fraction of volume of voxel *v* that consists of gray matter [as opposed to white matter, cerebrospinal fluid, blood, etc.]); and *SRF*(*s*, *v*) = segment response fraction (i.e., fraction of gray matter in voxel *v* that is responsive to segment *s*). Of course, the value for *VoxVol* is known from the fMRI imaging parameters. Estimation of *GMF* is important in order not to overestimate the volume of neural tissue in a voxel that is responsive to a particular visual field segment. For example, if a voxel contains a large draining vessel, then it would be desirable to reduce the contribution of that voxel to the total sum of responsive neuronal volume. Work is currently underway to estimate *GMF* for each voxel in real-time (Shefchik et al., 2011). Until such an estimate becomes available, the *GMF* term in the above formula will be replaced with the constant *one*.

SRF is estimated from the Kalman filter output as follows: For a given voxel, a *z*-score (*Z*(*s*)) expressing the statistical significance of the response to segment *s* of the visual field is obtained by dividing the estimated response amplitude parameter a_s (Eqs. (12) and (27)) by the square root of the corresponding element of the state covariance matrix (Eqs. (14) and (28)). Recall that the first two elements of the state vector are associated with the baseline mean and drift rate, hence element s + 2 in Eq. (12) is associated with the segment *s* response amplitude, so we have:

$$Z\,(s) = \frac{a_s}{\sqrt{P\,(s+2, s+2)}}, \quad s = 1, \ldots, S. \tag{33}$$

For this voxel, let *I* be the collection of segment indices whose *z*-scores are greater than some threshold Z_{thr}:

$$I = \{s | Z\,(s) > Z_{thr}\}\,. \tag{34}$$

Then, the segment response fraction *SRF*(*s*,*v*) of voxel *v* for segment *s* is the fraction of the summed amplitudes in voxel *v* corresponding to segment *i*:

$$SRF\,(s,v) = \begin{cases} \frac{a_s}{\sum_{i \in I} a_i} & s \in I \\ 0\ otherwise & \end{cases}. \tag{35}$$

Substituting Eq. (35) into Eq. (32), and summing over all voxels yields the *IFV* for segment *s*.

Note that the quantity *IFV*(*s*) has actual physical units of measurement (i.e., mm^3). Previously, results have been presented in relative terms, using arbitrary units. However, for both research and clinical applications, using fixed units of measurement is an important step forward since objective analysis and comparison of results is difficult otherwise. Of course, this requires the assumption that response amplitude for each segment is determined by the fractional volume (number of neurons) that is activated by that segment. However, the volume (number of neurons) could be activated more or less strongly by a particular segment depending on the particular selectivities of the neurons for, say, check size, flicker rate, segment overlap with the receptive field, etc. Furthermore the conventional BOLD signals are spatially biased due to the presence of large vessels. For this reason, we are

working on real-time estimation of the GMF, so that only gray matter will be included in the IFV estimate. However, this is only a partial solution, since the accuracy of the IFV estimate may be affected by imaging artifacts due to nearby large draining vessels. Nevertheless, we feel that in spite of these limitations, it is advantageous to have fixed units of measurement in order to facilitate comparison of results across subjects, test conditions, and imaging platforms.

Monte Carlo simulations

Monte Carlo simulations: methods

Monte Carlo simulations were used to compare the sensitivity of the phase encoding and random-block visual field mapping paradigms to different noise levels and errors in the hemodynamic impulse response model. Two ensembles (S = 20) of binary stimulus functions (Fig. 2) corresponding to the two experimental design paradigms (phase encoding and random encoding) were generated. The simulation was simplified by assuming that each voxel responds to one and only one segment (wedge or ring) location in the visual field. The fMRI signal for a voxel responding to a particular visual field location was simulated by using the model for the hemodynamic response (Eq. (4)), to which was added white Gaussian noise (to simulate measurement noise). So, if a particular voxel responded to visual excitation in segment i, then the fMRI signal was modeled as:

$$z(t) = y_i(t) + \varepsilon(t), \quad (36)$$

where $\varepsilon(t) \sim N\left(0, \sigma_{\mathrm{n}}^2\right)$. We have previously shown that signal-to-noise ratio (SNR) is not an appropriate parameter for comparison of different experimental designs (Ward and Mazaheri, 2006). Parametric comparison of different experimental designs requires a control parameter that is independent of the experimental design itself. Experimental designs having block lengths that are short relative to the IRF are temporally filtered and may not reach full amplitude. Therefore, SNR is not an appropriate parameter since the amplitude of the response varies with the block length. Alternatively, the area-to-noise ratio (ANR) is defined:

$$ANR \equiv \frac{AUC}{2\sigma_n} \quad (37)$$

(Ward and Mazaheri, 2008), where AUC = area-under-the-curve of the hemodynamic IRF, provides a suitable parameter that is independent of the experimental design itself. (For long block lengths, SNR and ANR are approximately equal). The constant σ_{n} in the above equation was chosen to yield the desired ANR level (i.e., σ_{n} = 1/(2 ANR)) since the differential equation model (Eq. (4)) has the property that AUC = 1 for all values of the physical constants ζ, ω_{n}, and τ.

(Note: The ANR measure was *not* used to evaluate the experimental design. Rather, it was used to parameterize the different designs, in order to allow a “fair” comparison between random encoding and phase encoding. If we had used SNR to parameterize the different designs, there would be a small bias in favor of the randomized design. However, to evaluate and compare the performances of the alternative designs, we used cross-correlation as the performance metric.)

The “preferred segment” for each voxel was estimated by finding the maximum correlation coefficient ρ between the fMRI time course $z(t)$ and each of the canonical response functions $r_j(t)$:

$$k=\mathrm{argmax}_j \rho\left(z\left(t\right), r_j\left(t\right)\right), \quad j=1,\ldots,S. \quad (38)$$

The canonical response functions $r_j(t)$ were generated by solving Eq. (4) using the set of canonical physical constants $\zeta = 0.76$, $\omega_n = 0.55\ s^{-1}$, and $\tau = 2.41$ s. (as mentioned earlier). The fMRI "signal" was generated by solving Eq. (4), but using the specified "true" physical constants in place of the canonical values. Referring to Eq. (38), if a given voxel's preferred segment is i, then a classification error has occurred if $k \neq i$. Classification error probability was estimated by repeating the simulation 2000 times for each scenario and accumulating the number of classification errors.

Monte Carlo simulations: results

The Monte Carlo simulation results are presented in Fig. 4. The top row illustrates the "true" IRFs used for generating the data, with corresponding analysis results displayed in the bottom row. The IRFs in the top row were calculated from Eq. (5a,b,c), using the "true" values for ζ, ω_n, and τ, whereas the analysis used the canonical values for these parameters.

In the first column, we consider the case where the canonical and true IRFs are the same. In Fig. 4(a), we display the canonical IRF, obtained using Eq. (5a,b,c), with the canonical physical constants. In Fig. 4(e), we plot the visual segment identification error probability vs. ANR (area-to-noise ratio) for phase encoding (blue curve) and random encoding (red curve). As may be seen, the segment identification error for the phase encoding method is greater than that for the random-block encoding method for small ANR values (i.e., large noise levels). Decoding errors are more likely to occur when using phase encoding since the individual stimulus functions are more similar to each other (Fig. 2(a)) than is the case with random-block encoding (Fig. 2(b)). Most of the error from the phase encoding method consists in misidentifying the true segment as one of its two nearest-neighbor segments.

For the remaining scenarios, we hold the ANR fixed at 1.0, which represents a relatively high noise condition. In the second column, we show the results obtained when the true value for parameter ζ differs from its assumed canonical value. In Fig. 4(b), we plot the "true" IRF's when ζ varies from 0.30 to 1.52. As can be seen from this figure, parameter ζ determines the "shape" of the IRF; specifically, the amount of overshoot or oscillation in the response; small values of ζ correspond to greater overshoot. In Fig. 4(f), we plot the segment identification error probability when the simulated data are analyzed using the canonical model with assumed $\zeta = 0.76$. With phase encoding (blue curve), the error probability is at a minimum when the "true" and canonical models agree; however, the error probability increases when the "true" ζ diverges from the canonical value. On the other hand, random encoding (red curve) is very robust to errors in the modeling assumptions with virtually no errors over the range of ζ values tested.

In the third column, we show the results obtained when the true value for parameter ω_n differs from its assumed canonical value. In Fig. 4(c), we plot the "true" IRFs when ω_n varies from 0.22 to 1.1. As can be seen from this figure, parameter ω_n determines the time scale of the response; for small values of ω_n, the response is stretched-out, whereas for large values of ω_n, the response is more compressed in time. In Fig. 4(g), we plot the segment identification error probability when the simulated data are analyzed using the canonical model with assumed $\omega_n = 0.55\ s^{-1}$. With phase encoding (blue curve), the error probability, as before, is at a minimum when the "true" and canonical models agree; however, the error probability increases when the "true" ω_n diverges from the canonical value. Again, as before, random encoding (red curve) is very robust to errors in the modeling assumptions.

In the fourth column, we show the results obtained when the true value for the time delay parameter τ differs from its assumed canonical value. The results are plotted in Fig. 4(h), which shows the visual segment identification error probability vs. IRF modeling parameter (true time delay τ) for phase encoding (blue curve) and random encoding (red curve). As would be expected, the minimum classification error occurs when the model for generating the data coincides with the model for analyzing the data (i.e., near $\tau = 2.4$ s). Also, we see that the phase encoding method is particularly susceptible to modeling error in the assumed time lag (since phase encoding is based on relative timing), whereas the random encoding is very robust to error in the time lag model.

Monte Carlo simulations: discussion

The above Monte Carlo simulations demonstrate that the random block encoding method is very robust to errors in modeling of the voxel IRF (vis a vis the phase encoding approach). This is indicated in Fig. 4(f)–(h), which show that the segment identification error probability remains negligibly small, even when the "true" IRF differs greatly from the modeled IRF. Also, from Fig.4(e), the random block encoding method shows greater resilience to noise than the phase encoding method. These results are not unexpected since the random block encoding method relies on random sequences rather than phase delay for identifying voxel responsiveness. That is, the random block binary sequences are more distinct from each other than the phase shifted binary sequences.

The Monte Carlo simulations have several limitations. First, these simulations use independent errors for modeling the measurement noise. This may lead to overly optimistic results since temporally correlated errors may yield more misclassifications. A second limitation is the assumption that the ANR level is independent of the experimental design. There is some evidence that phase encoding tends to yield higher amplitude responses relative to random encoding (E. DeYoe, personal observation). If so, then the current analysis would underestimate the benefits of phase encoding for experimental design.

A third limitation is the assumption that voxels do not have multiple-segment responses. It is assumed that if a particular voxel responds to the visual stimulus, then it is responding to only one segment in the visual field. Of course, this assumption is not true in general, as there is no reason for the voxel boundaries to be aligned with the visual field segment boundaries. As discussed in the Application to visual field mapping section, we believe that the capability to determine whether a voxel is responding to multiple, not-necessarily contiguous, segments in the visual field is a major consideration in favor of random-block encoding.

fMRI experiments

Here, we present results from two types of fMRI visual field mapping experiments, which were conducted using random visual field excitation paradigms. For the first experiment, the visual field was mapped in one dimension using angular sectors (wedges). In the second experiment, the visual field was mapped in two dimensions (angle and eccentricity) simultaneously using radial sectors.

fMRI 1-D visual field mapping experiment

Materials and methods—While fixating a central marker, subjects viewed flickering (8 Hz), black and white checkered wedges presented in independent, random, ON/OFF sequences at each of 20 equally-spaced clock positions within the subject's field of view (FOV) extending to 30° eccentricity (see Fig. 1(b)). During each 240 s fMRI run,

approximately half of the wedges were ON at any one time, with minimum ON or OFF epochs of 10 s (see Fig. 2(b)).

Brain images were obtained with a 1.5 T General Electric Signa scanner equipped with a custom three-axis, shielded head coil designed for rapid gradient field switching. A 64 × 64 voxel matrix covering a 24 × 24 cm FOV was used to obtain voxels of 3.75 × 3.75 × 4.0 mm. For gradient-recalled echo-planar fMRI, the imaging parameters were as follows: an initial 90° RF pulse, a TR of 1000 ms, and TE of 40 ms. The first four images of each scan were discarded to allow brain tissue magnetization to achieve steady state. At the end of the scanning session, a high resolution, T1-weighted, spoiled GRASS anatomical image was also collected. The anatomical images covered the whole brain with a voxel resolution of 1 × 1 × 1.1 mm (flip angle = 30°, TR = 25 ms, FOV = 24 cm).

Results and discussion

Kalman filter time series analysis: Since the Kalman filter is a recursive algorithm, the optimal estimate of the state vector, using all current and past fMRI measurements, is available in real-time. This is illustrated in Fig. 5, which shows the output of the Kalman filter for a single voxel during the imaging run. In Fig. 5(a), the Kalman filter estimate of the fMRI signal (red line) and the baseline (green line) are displayed. These are plotted with the actual fMRI time series (black line), normalized to have an initial baseline value of 100. In Fig. 5(b), the Kalman filter estimate of the state vector (i.e., the 20 elements of the state vector corresponding to the amplitude of the response of this particular voxel to the visual stimulation from each of the 20 wedge locations) and the standard deviation of that estimate (Fig. 5(c)) are plotted. Here, the same color coding is used as before to represent different wedge locations. At each point in time, we see the best estimate of the voxel's response to each of the 20 wedge locations, using all of the fMRI data up to that point in time. From Fig. 5(b), it becomes evident at about 130 s that this particular voxel is responding to two distinct wedge locations, wedges #1 and #20, and not to any of the other wedges. In fact, wedges #1 and #20 are spatially contiguous (see color wheel inset for corresponding color code map). As the imaging run continues (that is, as more measurements become available), there is little change in these parameter estimates.

Visual field activation map and voxel response profiles: Fig. 6(a) shows the visual field brain activation map for a single slice through occipital visual cortex. The color coding for each voxel corresponds to the visual field wedge location for which the voxel was maximally responsive (i.e., highest z-value from the Kalman filter state estimate), here using a z-threshold of $z_{thr} = 2.25$. When viewed over the time evolution of the experiment (not shown), the set of active voxels grows gradually during the run, although the colors of the active voxels (i.e., the most responsive wedge locations) tend toward constant values.

In Fig. 6(b), we see the voxel visual field response profiles for the 3 × 3 array of voxels contained in the small square in (a). These bar charts display the output from the Kalman filter for each of these 9 voxels. Along the vertical axis is plotted the z-score corresponding to the amplitude estimate for each of the 20 wedge locations. Close inspection of the visual field response profiles reveals that the results are more subtle than the simple "winner-take-all" map of Fig. 6(a). That is, each graph in Fig. 6(b) shows the complete response profile to all stimulus segments for a given voxel. Also, we see the gradual shift in the response profile from voxel-to-voxel. Going down the left-hand column of Fig. 6(b), we see a gradual counter-clockwise shift in preferred location (from #7 to #9) for each voxel. We see a similar counter-clockwise shift in preference going up the right-hand column of voxels. The voxels in the center column, which may be straddling the hemispheric midline, have a combination of the activation patterns from both left- and right-hand columns of voxels. The

ability to detect multiple visual field responses for individual voxels is a distinct advantage of the random-block encoding method over the phase encoding method.

An expanded view of the voxel response profiles is provided by Fig. 7, which shows the voxel visual field response profiles for the 9 × 9 array of voxels contained in the large rectangle of Fig. 6(a). This figure shows that the general pattern of Fig. 6(b) extends to the rest of the visual cortex. That is, we can see the systematic shift in the visual field response profile over space, from voxel-to-voxel, that is the hallmark of retinotopically organized visual cortex. Note that it is often the case that a single voxel responds to multiple (usually, but not always, neighboring) locations in the visual field.

Simultaneous random mapping of eccentricity and polar angle in real-time

Materials and methods—For this experiment, the visual field was divided into a dartboard-like array of 12 checkered segments (illustrated schematically by the color wheel inset in Fig. 8) scaled according to the retinocortical mapping function of Polimeni et al. (2006) to produce activation of roughly equal areas of primary visual cortex. The 8 Hz flickering stimulus segments were presented as random uncorrelated temporal sequences with approximately 50% duty cycle during a 600 s fMRI scan. Conventional BOLD echo planar fMRI acquisition was obtained using a 3.0 T General Electric Signa scanner with TR = 2 s, TE = 20 ms, and 2.5 mm cubic voxels.

Real-time parameter estimation and display of the visual field response map was simulated offline in MATLAB using the data from each time point (TR period) of the fMRI imaging run as input to the recursive Kalman filter algorithm. The platform for the real-time simulation consisted of a Dell Precision T7500 64-bit dual quad-core processor running at 2.66 GHz with 24 GB memory, with two 24-inch monitors. All of the real-time analysis software was written in MATLAB and used the Parallel Processing toolbox.

A single volume of imaging data was input to a bank of Kalman filters at the rate of once every 2 s. Approximately 16,000 Kalman filters were set up to run in parallel, one for each voxel inside the masked region. The color of a voxel in the visual field map was determined by the segment having the highest z-score, thresholded at $z_{Thr} = 3.0$ (see Figs. 8 and 11).

Results and discussion

Run termination criterion: As mentioned in Introduction, one of the reasons for performing real-time analysis is to allow the clinician to determine whether an imaging run has been successful (i.e., yielded diagnostically useful information). This will allow the clinician to determine whether the run should be terminated, extended, repeated, rescheduled, etc.

But this raises the question: What criteria should be used for deciding when to terminate the run? One possibility is to display the evolution of the visual field brain activation map over time and to terminate the run when the activation map reaches an approximate steady-state condition. This is illustrated in Fig. 8, which shows a sequence of visual field activation maps from one run, separated by intervals of 100 s of image acquisition time. Qualitatively, there is little change in the visual field brain activation map in Fig. 8(c) through (f) (i.e., the activation map stabilized after approximately 300 s into the run).

A more quantitative run-termination criterion is obtained by displaying the parameter estimation accuracy as a function of time. The Kalman filter is continuously updating the estimate of the state covariance matrix **P** (Eq. (28)). The square root of a diagonal element of **P** represents the standard deviation of the corresponding state parameter estimate. These values can be displayed graphically in real time. This is illustrated by Fig. 9, which shows

the maximum standard deviation of the 12 visual field parameter estimates from a set of 25 voxels as a function of time. Fig. 9 illustrates the fact that increasing increments of imaging time has diminishing returns, so far as parameter estimation accuracy is concerned.

Fig. 10 shows a real-time plot of the integrated fractional volume (Eq. (32)) for each of the 12 visual field segments, integrated across all responsive voxels, as a function of imaging run time. For each segment s, we see that the $IFV(s)$ increases rapidly at the beginning of the run, but eventually converges to a more-or-less constant value. Again, we see that after a certain point, continuing the run yields little additional information. There are two possible requirements for using IFV as the run-termination criterion: (1) $IFV(s)$ must reach a stable level, for a specified period, for each segment s. (2) The value of $IFV(s)$ must be above some minimum volume threshold (to be determined). The minimum volume threshold would be set to assure that there is sufficient responsive neuronal volume for each part of the visual field. Failure to achieve this minimum threshold may be indicative of a visual field deficit.

Real-time displays: As should be evident from the foregoing discussion, there are many alternatives for presenting diagnostic information in real time to the clinician. The objective is to enable the clinician to decide when sufficient data have been collected to allow termination of the run, or, if necessary, to repeat or reschedule the imaging run.

Fig. 11 is a snapshot of one possible video display configuration during a (simulated) real-time imaging run. Fig. 11(a), (b), and (c) show brain activation maps for segments, wedges, and rings, respectively, plotted on top of the current fMRI image data. The wedge and ring maps are computed by combining the response to the individual segments composing each wedge or ring. Fig. 11(d), (e), and (f) depict the color coding of the visual field for segments, wedges, and rings, along with the integrated fractional volume bar graphs for segments, wedges, and rings, respectively. All of these plots are updated continuously during the run. Indicators for successful completion of the run would be the spatial extent and stability of the brain activation maps, as well as the attainment of prespecified minimum values for the IFV corresponding to different segments of the visual field.

Conclusions

In this paper, we have demonstrated the feasibility of real-time mapping of the visual field for clinical applications. Translation of this technology from research lab to medical clinic presents several challenges. We have examined three aspects of this problem: (1) experimental design, (2) statistical analysis, and (3) display of results.

Proper choice of the experimental design (i.e., the visual field stimulation paradigm) is critical to the success of the diagnostic imaging run. We have shown that a random-block encoding design has less sensitivity to measurement noise, as well as greater robustness to error in modeling of the hemodynamic IRF, than alternatives such as the phase encoding method. Also, random encoding has more inherent flexibility than “fixed” designs such as m-sequences (Buracas and Boynton, 2002; Hansen et al., 2004), which is important in allowing the possibility of variable run length, as well as real-time censoring of measurements. Other alternative stimulus designs, such as “drifting bars” (Dumoulin and Wandell, 2008), impose computational requirements that are less optimal for real-time applications. In addition, random encoding of the visual field allows the detection of voxels that are responsive to multiple, not necessarily contiguous, regions of the visual field.

Due to its recursive nature, the Kalman filter is ideally suited for real-time statistical analysis of the visual field mapping data. An important feature of the Kalman filter is that it can be used for nonstationary time series analysis. The capability of the Kalman filter to adapt, in

real-time, to abrupt changes in the baseline arising from subject motion inside the scanner, and other external system disturbances, is important for the success of clinical applications.

Of course, our ultimate goal is to provide the clinician with information that will make it possible to evaluate the success or failure of the imaging run and to decide, in real time, whether to extend, modify, or terminate the run. Real-time displays of (1) brain activation maps for each stimulus segment, (2) voxel-wise spatial tuning profiles, (3) time plots of the variability of response parameters, and (4) time plots of activated volume can provide ample information on which to make such decisions.

In summary, we have shown that using a random stimulus presentation and a recursive Kalman filter for multiple parameter estimation, it is possible to display, in real time, clinically relevant information. We believe that this is an important development for enabling the transfer of fMRI technology to clinical applications.

Acknowledgments

This work was supported by grants R01 EB007827 and R01 EB001417 from the National Institutes of Health.

References

Boynton GM, Engel SA, Glover GH, Heeger DJ. Linear systems analysis of functional magnetic resonance imaging in human V1. J. Neurosci. 1996; 16:4207–4221. [PubMed: 8753882]

Bozic, SM. Digital and Kalman Filtering. John Wiley & Sons; New York, NY: 1979.

Buracas GT, Boynton GM. Efficient design of event-related fMRI experiments using M-sequences. NeuroImage. 2002; 16:801–813. [PubMed: 12169264]

Chatfield, C. The Analysis of Time Series: An Introduction. CRC Press; Boca Raton, FL: 2004.

Cohen MS. Parametric analysis of fMRI data using linear systems methods. NeuroImage. 1997; 6:93–103. [PubMed: 9299383]

Cox RW. AFNI: Software for analysis and visualization of functional magnetic resonance neuroimages. Comput. Biomed. Res. 1996; 29:162–173. [PubMed: 8812068]

Dale AM, Buckner RL. Selective averaging of rapidly presented individual trials using fMRI. Hum. Brain Imaging. 1997; 5:329–340.

D'Azzo, JJ.; Houpis, CH.; Sheldon, SN. Linear Control System Analysis and Design with MATLAB. Marcel Dekker; New York, NY: 2003.

Dumoulin SO, Wandell BA. Population receptive field estimates in human visual cortex. NeuroImage. 2008; 39:647–660. [PubMed: 17977024]

Friston KJ, Fletcher P, Josephs O, Holmes A, Rugg MD, Turner R. Event-related fMRI: characterizing differential responses. NeuroImage. 1998; 7:30–40. [PubMed: 9500830]

Gelb, A., editor. Applied Optimal Estimation. MIT Press; Cambridge, MA: 1974.

Glover GH. Deconvolution of impulse response in event-related BOLD fMRI. NeuroImage. 1999; 9:416–429. [PubMed: 10191170]

Grewal, MS.; Andrews, AP. Theory and Practice Using MATLAB®. John Wiley & Sons; New York, NY: 2001. Kalman filtering.

Hansen KA, David SV, Gallant JL. Parametric reverse correlation reveals spatial linearity of retinotopic human V1 BOLD response. NeuroImage. 2004; 23:233–241. [PubMed: 15325370]

Kalman RE. A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic. Engineering. 1960; 82:35–45.

Ogata, K. Modern Control Engineering. Prentice-Hall; Englewood Cliffs, NJ: 1970.

Polimeni JR, Balasubramanian M, Schwartz EL. Multi-area visuotopic map complexes in macaque striate and extra-striate cortex. Vision Res. 2006; 46:3336–3359. [PubMed: 16831455]

Shefchik DL, Nencka AS, Jesmanowicz A, Hyde JS. Accelerated gradient-recalled echo, asymmetric spin-echo (GREASE-II) for production of high-resolution human T1, T2, and T2* maps (abstract). Proc. Int. Soc. Magn. Reson. Med. 2011; 19:4510.

Shumway, RH. Applied Statistical Time Series Analysis. Prentice-Hall; Englewood Cliffs, NJ: 1988.

de Visme, GH. Binary Sequences. English Universities Press; London: 1971.

Ward BD, Mazaheri Y, Roopchansingh V. Application of Kalman filtering to detection and estimation of nonstationary fMRI signals (abstract). Proc. Int. Soc. Magn. Reson. Med. 2005; 13:1596.

Ward BD, Janik J, Mazaheri Y, DeYoe EA. Real-time mapping of the visual field using random stimulus presentation (abstract). Proc. Int. Soc. Magn. Reson. Med. 2006; 14:3256.

Ward BD, Mazaheri Y. State-space estimation of the input stimulus function using the Kalman filter: a communication system model for fMRI experiments. J. Neurosci. Methods. 2006; 158:271–278. [PubMed: 16828876]

Ward BD, Mazaheri Y. Information transfer rate in fMRI experiments measured using mutual information theory. J. Neurosci. Methods. 2008; 167:22–30. [PubMed: 17919734]

Zarchan, P.; Musoff, H. Fundamentals of Kalman Filtering: A Practical Approach. second edition. American Institute of Aeronautics and Astronautics (AIAA); Reston, VA: 2005.

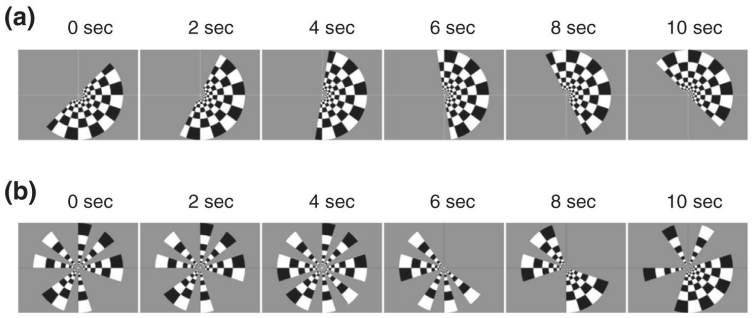

Fig. 1.

Comparison of visual stimulus designs. First row: Phase encoding of the visual stimulus. Second row: Random block encoding of the visual stimulus. Consecutive columns show the visual stimuli as they would appear when separated by 2 s intervals.

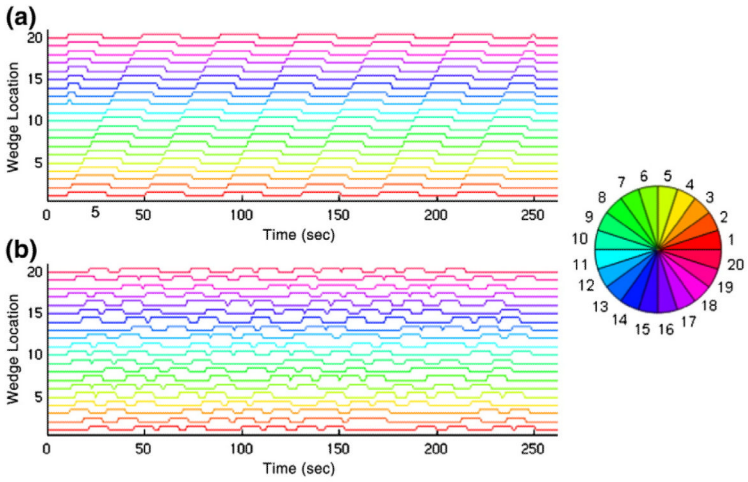

Fig. 2.

Time series representation of activation sequence for different parts of the visual field for the (a) phase encoding and (b) random block encoding methods. The color wheel at the right indicates the color coding for individual wedge locations.

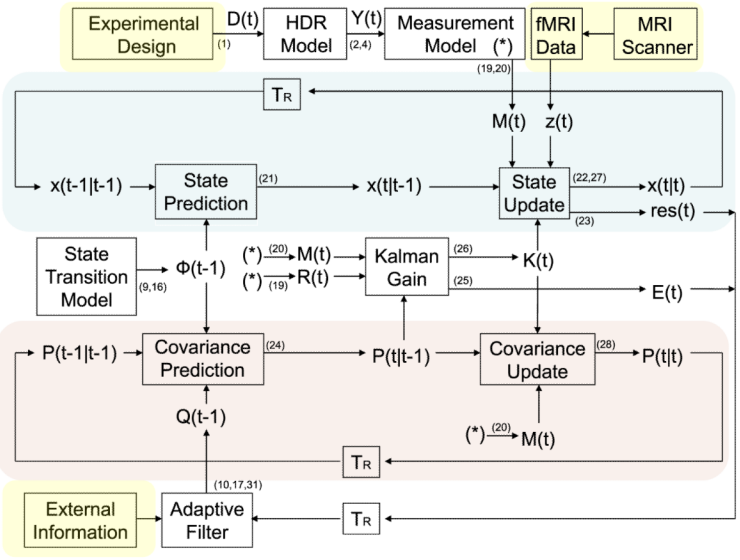

Fig. 3.

Block diagram of implementation of Kalman filter for an fMRI experiment. See the text for explanation of the processing blocks. Numbers in parentheses refer to equations in the text. T_R represents a discrete time increment delay based on the fMRI imaging sequence sampling rate. The blue-shaded region indicates the processing steps involved in recursive estimation of the state vector (voxel stimulus-evoked response). The orange-shaded region indicates the processing steps for recursive estimation of the state covariance matrix. The yellow-shaded regions indicate the external inputs to the system.

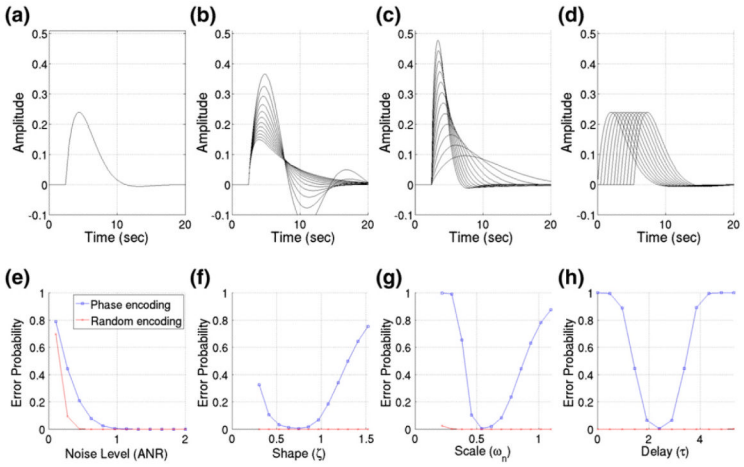

Fig. 4.

Monte Carlo simulation results for comparison of phase encoding vs. random encoding experimental designs. First row: "True" IRFs used for generating the simulated data. Second row: Visual field segment classification error probability obtained from analyzing data simulated by using the first row IRF's, along with phase encoding (blue curve) and random encoding (red curve) experimental designs. (a) The canonical IRF model ($\zeta = 0.76$, $\omega_n = 0.55$, $\tau = 2.41$). (b) "True" IRFs, same as (a), but IRF "shape" parameter ζ varies from 0.30 to 1.52. (c) "True" IRFs, same as (a), but IRF "scale" parameter ω_n varies from 0.22 to 1.10. (d) "True" IRFs, same as (a), but IRF "delay" parameter τ varies from 0.00 to 5.31. Plot (e) Error probability vs. "area-to-noise ration" ANR, shows that random encoding is more robust to noise than phase encoding. Plots (f) error probability vs. IRF "shape", (g) error probability vs. IRF "scale", and (h) error probability vs. IRF "delay", indicate that random encoding is very robust to error in the assumed IRF model, whereas phase encoding is very sensitive to modeling errors.

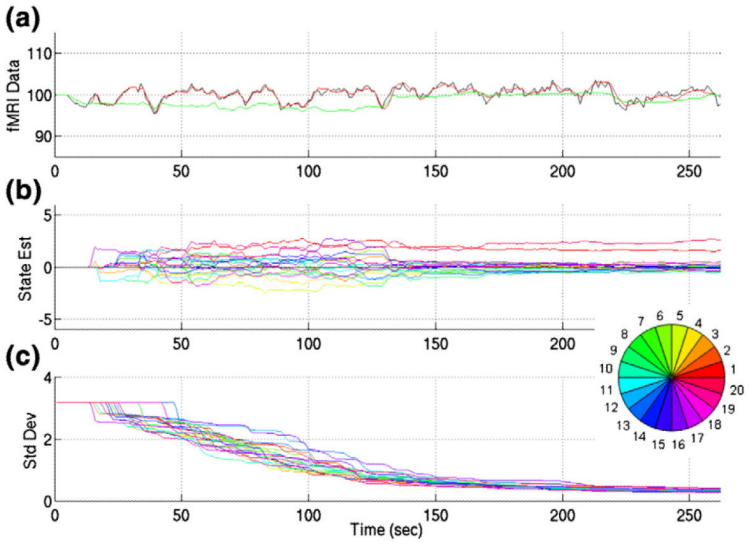

Fig. 5.

Output from Kalman filter for a single voxel during the imaging run: (a) Kalman filter estimate of signal (red line) and baseline (green line), plotted on top of actual fMRI time series data (black line). (b) Kalman filter estimate of the 20 element state vector as a function of time during the run. Each line represents the estimated amplitude of the response of this voxel to one of the 20 segments of the visual field. (c) Standard deviation of the parameter estimates for each of the 20 elements of the state vector. In (b) and (c), the color coding for the individual parameters corresponds to that for the visual field segments, indicated by the color wheel (inset). After approximately 130 s, it is apparent from (b) that this particular voxel is responding to segments (or wedges) #1 and #20. The state estimate of the voxel response amplitude for the other segments stays close to zero.

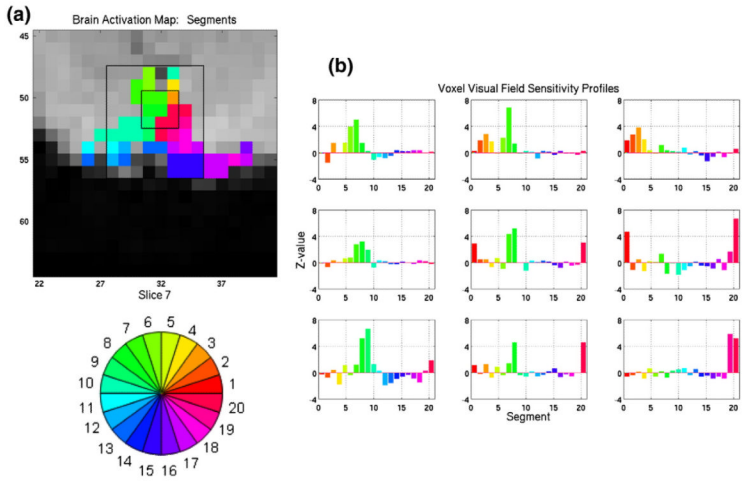

Fig. 6.

(a) Voxel-based, visual field activation map for a single slice through the visual cortex, plotted on top of fMRI image data. The color-coding indicates the visual field location for which each active voxel has greatest response. (b) Voxel visual field response profiles for the 3 × 3 array of voxels contained in the small square in (a). Individual visual field segments are plotted along the horizontal axis; z-values corresponding to the segments are plotted along the vertical axis. The color wheel inset indicates the color coding of the individual segments (wedges) within the visual field.

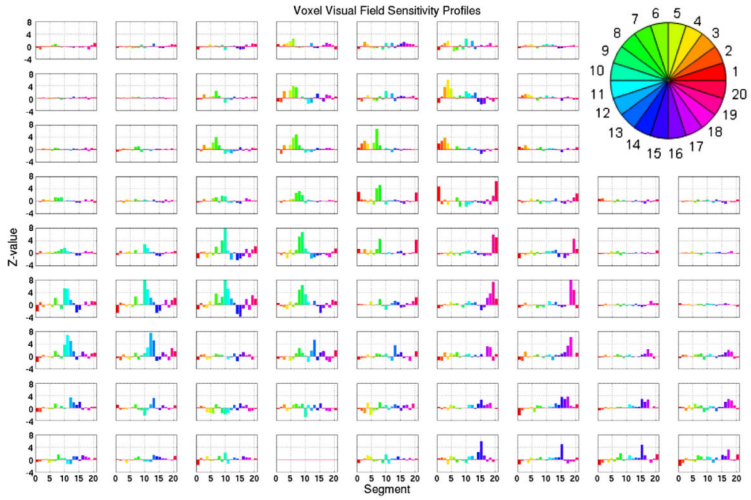

Fig. 7.

Stimulus response profiles for the 9 × 9 array of voxels contained in the large square of Fig. 6(a). Each box corresponds to a single voxel. Visual field segment indices are indicated along the horizontal axes; corresponding z-values are plotted along the vertical axes. Color coding for the visual field segments is indicated by the color wheel (inset).

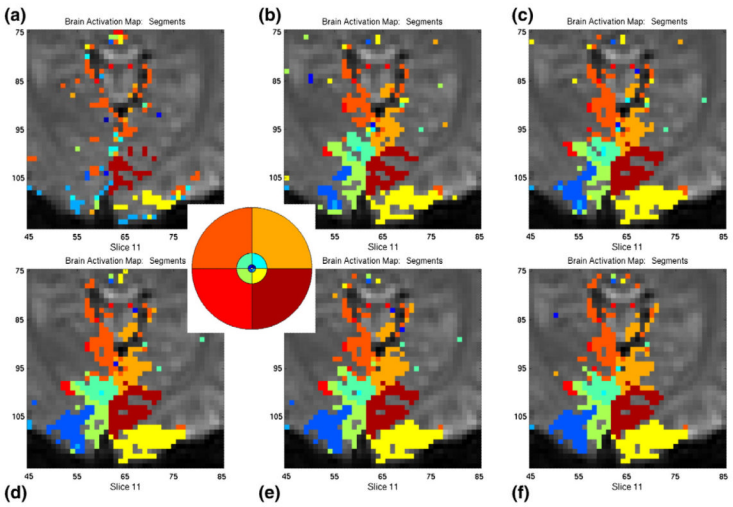

Fig. 8.

The brain activation map indicates to which segment of the visual field each voxel is most responsive. Here, the color coded brain activation maps, obtained after t = 100, 200, 300, 400, 500, and 600 s, are displayed in subplots (a) through (f), respectively. Note that there is little change in the brain activation map after the first 200 or 300 s. The color coding for the different stimulus segments is indicated by the color wheel (inset).

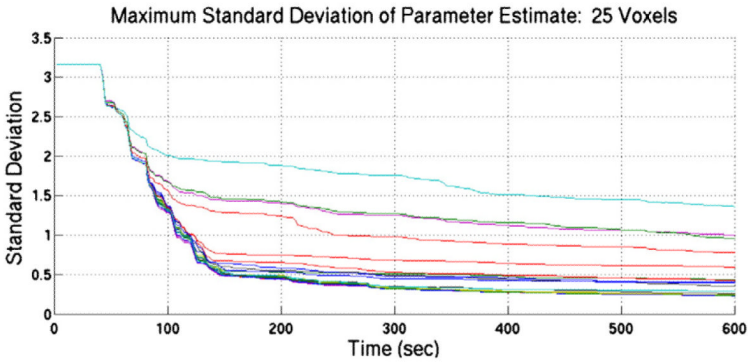

Fig. 9.

This is a plot of the maximum standard deviation of stimulus parameter estimates from 25 centrally located voxels as a function of time. Different colors correspond to different voxels. Note that there is little improvement in the parameter estimation accuracy after the first 200 or 300 s.

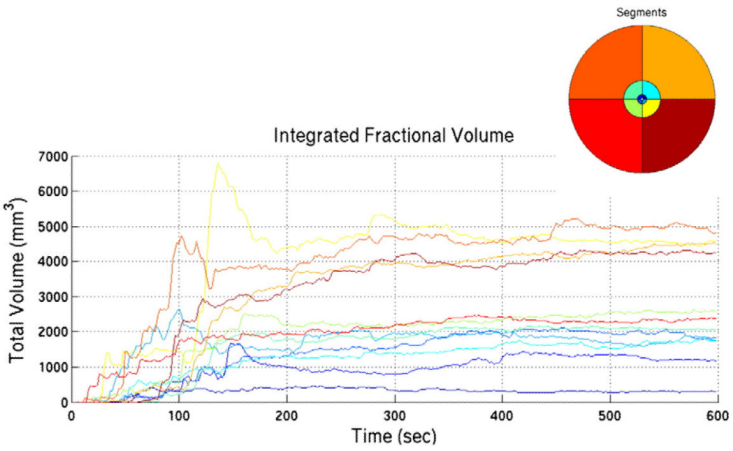

Fig. 10.

Plot of Integrated Fractional Volume (i.e., estimate of total gray matter tissue volume responsive to an external stimulus) for each of the 12 visual field segments vs. time. Again, there is little change in the IFV estimates after the initial 200 or 300 s of run time. The inset indicates the color coding of the 12 visual field segments for a two-dimensional (polar angle plus eccentricity) mapping of the visual field.

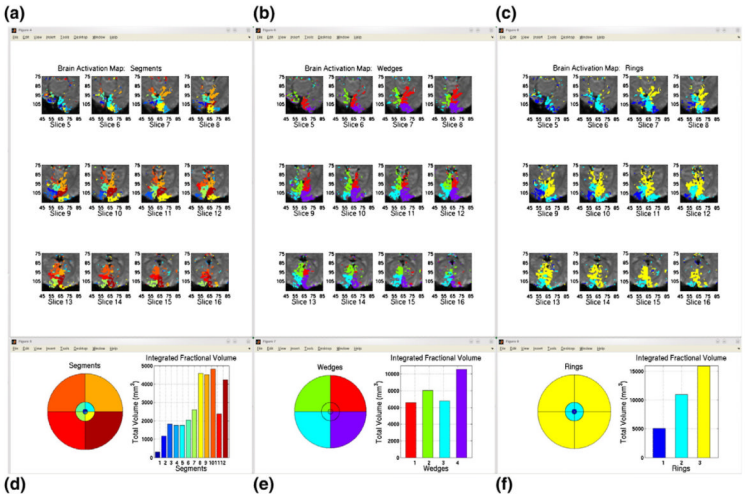

Fig. 11.

Snapshot of video monitor display during a (simulated) real-time imaging run. Fig. 11(a), (b), and (c) are the brain activation maps for segments, wedges, and rings, respectively. The left-halves of (d), (e), and (f) depict the color coding of the visual field for segments, wedges, and rings, respectively. The right-halves of (d), (e), and (f) show the integrated fractional volume bar graphs for segments, wedges, and rings, respectively. Each of the plots (a) through (f) is updated continuously during the run.