

Partial volume estimation using continuous representations

Mohammad-Reza Siadat^{*a,b}, Hamid Soltanian-Zadeh^{a,c}

^a Medical Image Analysis Laboratory, Henry Ford Health System
Detroit, MI 48202, USA

^b Computer Science Department, Wayne State University
Detroit, MI 48202, USA

^c Electrical and Computer Engineering Department, University of Tehran
Tehran 14399, Iran

ABSTRACT

This paper presents a new method for partial volume estimation using standard eigenimage method and B-splines. The proposed method is applied on the multi-parameter volumetric images such as MRI. The proposed approach uses the B-spline bases (kernels) to interpolate a continuous 2D surface or 3D density function for a sampled image dataset. It uses the Fourier domain to calculate the interpolation coefficients for each data point. Then, the above interpolation is incorporated into the standard eigenimage method. This incorporation provides a particular mask depending on the B-spline basis used. To estimate the partial volumes, this mask is convolved with the interpolation coefficients and then the eigenimage transformation is applied on the convolution result. To evaluate the method, images scanned from a 3D simulation model are used. The simulation provides images similar to CSF, white matter, and gray matter of the human brain in T1-, T2-, and PD-weighted MRI. The performance of the new method is also compared to that of the polynomial estimators.¹ The results show that the new estimators have standard deviations less than that of the eigenimage method (up to 25%) and larger than those of the polynomial estimators (up to 45%). The new estimators have superior capabilities compared to the polynomial ones in that they provide an arbitrary degree of continuity at the boundaries of pixels/voxels. As a result, employing the new method, a continuous, smooth, and very accurate contour/surface of the desired object can be generated. The new B-spline estimators are faster than the polynomial estimators but they are slower than the standard eigenimage method.

Keywords: partial volume estimation, B-spline interpolation, MRI, eigenimage method, Gram-Schmidt orthogonalization

1. INTRODUCTION

Partial volume averaging happens when there is more than one material within a scanned voxel/pixel. The applications of partial volume estimation in medical domain are identification of hidden abnormalities,² accurate volume calculation,³ accurate visualization and generating volume-rendered images.⁴ Other applications include urban ecosystem analysis, crop detection, fuel spill detection, wetland identification, waterway mapping, and forest management. The latter applications are far from the medical applications in that the images they use are not taken purely based on the properties of the materials themselves. Rather the lightening and the camera/scanner point of view play important roles on image formation.

A brief review of the work related to partial volume estimation methods can be found in reference.⁵ This reference proposes a histogram-based method that takes the information of neighboring pixels into account. The histogram method is time consuming and does not determine material distribution. The standard eigenimage method^{6,7} provides an optimal linear transformation, but it does not take advantage of the neighboring structure information. A method for improving the standard eigenimage method, using the information of fixed neighboring structures and polynomial estimators has been proposed in reference.¹ The choice of the neighboring structure can improve performance of the estimator in terms of its standard deviation. So one may choose different neighboring structures to get an optimal performance in an adaptive fashion. But the disadvantage of this method is that it does not provide a continuous estimation on the boundaries of the pixels/voxels. So, for the visualization purposes it is not useful. The new approach, presented in this paper, incorporates the neighboring information into the eigenimage method in such a way that the final estimation has an arbitrary degree of continuity for the entire data set. Details of the method are presented in Section 3. The eigenimage method is briefly described in Section 2. The results and discussions are reported in Section 4.

* Correspondence: Emails: msiadat@cs.wayne.edu, hamids@rad.hfh.edu

2. STANDARD EIGENIMAGE METHOD

The standard eigenimage is very fast at partial volume estimation.⁸ The optimality of this approach has been proved from a linear aspect of view.^{6,7} In this method, it is assumed that image intensities can be described as a linear combination of intensities of the materials being scanned, i.e.,

$$\vec{P}_{jk} = \alpha_d \vec{s}_d + \sum_{\substack{i=1 \\ i \neq d}}^m \alpha_i \vec{s}_i \quad (1)$$

where \vec{P}_{jk} is the intensity vector of the pixel located at (j, k) , α_d and α_i ($i = 1, \dots, m, i \neq d$) are partial volumes of the desired material and i -th material participating in the formation of \vec{P}_{jk} , respectively, \vec{s}_d is the signature vector of the desired material and \vec{s}_i ($i = 1, \dots, m, i \neq d$) is the signature vector corresponding to the i -th undesired material. The linear transformation presented in (2) is optimal for estimating α_d :

$$\vec{t}_d = \frac{\vec{t}_e}{\vec{s}_d \cdot \vec{t}_e} \quad (2)$$

where $\vec{t}_e = \vec{s}_d - \vec{s}_d^p$ and \vec{s}_d^p is the projection of the \vec{s}_d on the subspace spanned by $\{\vec{s}_k, k = 1, \dots, m, k \neq d\}$, i.e., the undesired subspace. Optimality of this approach is proven by showing that it maximizes the signal to noise ratio of the desired material given by:

$$SNR_d = \frac{\vec{t}_d \cdot \vec{s}_d}{\sigma(\vec{t}_d \cdot \vec{t}_d)^{1/2}}. \quad (3)$$

Using the above linear transformation, we can estimate the partial volume by:

$$\vec{\alpha} = T \vec{P}_{jk} \quad (4)$$

where $\vec{\alpha}$ contains the proportion of each material inside voxel (j, k) and T is a matrix whose rows are the normalized transformation vectors. This method does not use the information existing in the neighborhood of each pixel/voxel. In this paper we incorporate the information of neighboring points to improve the performance of the method in the presence of noise, as described in the following section.

3. PROPOSED METHOD

In this section, we substitute the intensity-vector of each pixel/voxel with a vector of continuous functions, estimated based on the B-spline interpolations.⁹ We employ different B-spline bases and compare their results. The interpolation is done in the Fourier domain where the B-spline basis is considered as a system transfer function, the unknown interpolation coefficients as the input signal, and the image as the output. Since the neighboring interpolation coefficients are used to calculate the intensities within each pixel/voxel, the interpolation describes the underlying structure of the material distribution better than the original sampled data. The substitute description has the same values as the sampled data at the center of each pixel/voxel. As we will see in Section 4, it gives a superior estimation in actual (noisy) cases for the partial volume. In Section 3.1, we describe the interpolation. In Section 3.2, we incorporate the interpolations into the eigenimage method.

3.1. Interpolation

A 1D B-spline basis function of degree n can be presented as:

$$\beta^n(x) = \underbrace{\beta^0 * \dots * \beta^0}_{(n+1)\text{ times}} * \beta^0(x) \quad (5)$$

Where $*$ indicates the convolution operation and,

$$\beta^0(x) = \begin{cases} 1, & |x| < 0.5 \\ \frac{1}{2}, & |x| = 0.5 \\ 0, & \text{elsewhere} \end{cases} \quad (6)$$

The closed-form representation of the $\beta^2(x)$, $\beta^3(x)$ (the cubic B-spline, which is very popular for interpolation), and $\beta^4(x)$ are as follows:

$$\beta^2(x) = \begin{cases} \frac{3}{4} - x^2, & |x| \leq 0.5 \\ \frac{1}{2}(\frac{3}{2} - |x|)^2, & 0.5 < |x| \leq 1.5 \\ 0, & \text{elsewhere} \end{cases} \quad (7) \quad \beta^3(x) = \begin{cases} \frac{2}{3} - x^2 + \frac{|x^3|}{2}, & |x| \leq 1 \\ \frac{(2-|x|)^3}{6}, & 1 < |x| \leq 2 \\ 0, & \text{elsewhere} \end{cases} \quad (8)$$

$$\beta^4(x) = \begin{cases} \frac{115}{192} - \frac{5}{8}x^2 + \frac{x^4}{4}, & |x| < 0.5 \\ \frac{55}{96} + \frac{5}{24}|x| - \frac{15}{12}x^2 + \frac{5}{6}|x^3| - \frac{1}{6}x^4, & 0.5 \leq |x| < 1.5 \\ \frac{1}{384}(5 - 2|x|)^4, & 1.5 \leq |x| < 2.5 \\ 0, & \text{elsewhere} \end{cases} \quad (9)$$

Fig. 1 shows plots of the above B-spline bases.

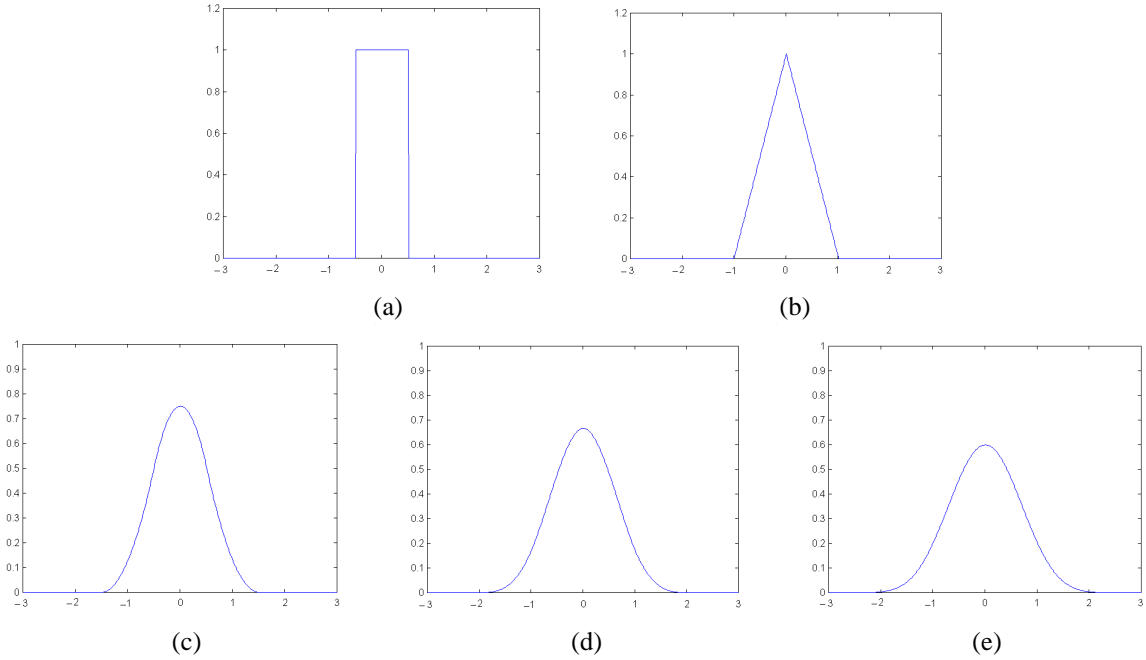


Fig. 1. Plots of (a) $\beta^0(x)$, (b) $\beta^1(x)$, (c) $\beta^2(x)$, (d) $\beta^3(x)$, (e) $\beta^4(x)$.

$\beta^n(x)$ is $(n-1)$ -time differentiable while it is limited in space within the interval $[-\frac{n+1}{2}, \frac{n+1}{2}]$. Differentiability of the interpolation basis function satisfies the continuity requirement. The property of being limited in the space reduces the effects of the far-zone data points. This is an acceptable feature since there is very little correlation between a data point and its far-zone neighborhood. The B-spline bases also provide very smooth interpolations (referred to as minimum curvature property⁹).

We assume the sampled data are spread evenly over the Cartesian space. The estimated function in general is calculated as:

$$\hat{g}(x_1, x_2, \dots, x_m) = \sum_{k_1 \in \mathbb{Z}} \sum_{k_2 \in \mathbb{Z}} \dots \sum_{k_m \in \mathbb{Z}} c(k_1, k_2, \dots, k_m) \beta^n(x_1 - k_1, x_2 - k_2, \dots, x_m - k_m), \quad (x_1, x_2, \dots, x_m) \in \mathbb{R}^m \quad (10)$$

where x_1, x_2, \dots, x_m are the axes of an m -dimensional real space and c is the coefficient function which is unknown. We can set up the following set of linear equations to calculate the interpolation coefficients.

$$\hat{g}(x_1, x_2, \dots, x_m) = g(x_1, x_2, \dots, x_m), \quad (x_1, x_2, \dots, x_m) \in \mathbb{Z}^m \quad (11)$$

where g is the sampled image dataset. Equation (11) employs the measured image data points to determine the same number of interpolation coefficients. Therefore, the above interpolation has the same value as the sampled data at the center of each voxel/pixel (exact interpolation¹⁰). According to (10) and (11), for a 256×256 2D image we have to solve a set of 65536 linear equations. The situation gets worst when we are interpolating a density function for a 3D dataset. To avoid this huge equation set, we formulate interpolation problem in terms of a system shown in Fig. 2. This figure presents a simulation of (10) and (11) in which the system impulse response is represented by B-spline basis function, the input and output signals are represented by the coefficient function, c , and the image data set, g , respectively.

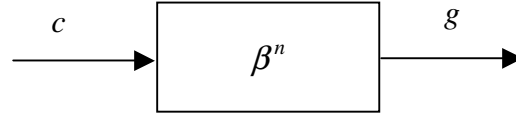


Fig. 2. Linear system-based representation for interpolation.

The following equation describes the behavior of the above system in the Fourier domain.

$$G = B \times C \quad (12)$$

where G and C are the Fourier transforms of the output image, g , and input coefficient, c , respectively, and B is the Fourier transform of the B-spline basis function, β^n . To calculate C , we divide G by B element by element.

3.2. Partial volume

In an m -dimensional space using “ p ” image modalities, we propose the following formula to calculate the partial volume in a voxel that includes the point (x_1, x_2, \dots, x_m) :

$$\begin{aligned} \alpha_d &= \int_{x_m - \Delta x_m / 2}^{x_m + \Delta x_m / 2} \dots \int_{x_2 - \Delta x_2 / 2}^{x_2 + \Delta x_2 / 2} \int_{x_1 - \Delta x_1 / 2}^{x_1 + \Delta x_1 / 2} \vec{t}_d \cdot \hat{g}(x_1, x_2, \dots, x_m) dx_1 dx_2 \dots dx_m \\ &= \int_{x_m} \dots \int_{x_2} \int_{x_1} \left(\sum_{l=1}^p t_d^l \hat{g}^l(x_1, x_2, \dots, x_m) \right) dx_1 dx_2 \dots dx_m \\ &= \int_{x_m} \dots \int_{x_2} \int_{x_1} \left(\sum_{l=1}^p t_d^l \sum_{k_1 \in \mathbb{Z}} \sum_{k_2 \in \mathbb{Z}} \dots \sum_{k_m \in \mathbb{Z}} c(k_1, k_2, \dots, k_m) \beta^n(x_1 - k_1, x_2 - k_2, \dots, x_m - k_m) \right) dx_1 dx_2 \dots dx_m \quad (13) \end{aligned}$$

where $\Delta x_1, \Delta x_2, \dots, \Delta x_m$ are the voxel sizes, t_d^l is the l -th component of the transformation vector \vec{t}_d , and \hat{g} is the estimated function vector whose elements are m -dimensional function \hat{g}^l . So, the estimated function \hat{g}^l (the same as \hat{g} in 3.1) is the continuous representation of the l -th image modality of the data set. \vec{t}_d is calculated in Section 2, i.e., standard eigenimage method, c is determined in section 3.1, and β^n is an m -dimensional B-spline basis function of degree n . Since summation and integral are linear operations, we can switch them and derive (14).

$$\alpha_d = \sum_{l=1}^p t_d^l \sum_{k_1 \in \mathbb{Z}} \sum_{k_2 \in \mathbb{Z}} \dots \sum_{k_m \in \mathbb{Z}} c(k_1, k_2, \dots, k_m) \int_{x_m} \dots \int_{x_2} \int_{x_1} \beta^n(x_1 - k_1, x_2 - k_2, \dots, x_m - k_m) dx_1 dx_2 \dots dx_m \quad (14)$$

The integral in (14) provides an m -dimensional mask. This mask depends only on the choice of the B-spline basis. The width of the mask at each direction also depends on the choice of the basis function. For instance, $\beta^2(x)$, $\beta^3(x)$, $\beta^4(x)$ in a separable B-spline provide 3-, 3-, and 5-element-wide masks, respectively. To calculate the overall desired volumes we have to sum up α_d over the entire data set. When (14) is applied to the entire image, it represents convolution of the mask and the interpolation coefficient, c . Note that we have not specified the interval of the interpolation summations yet. One may realize that the way of mask calculation is different for $\beta^n(x)$ with even and odd n . This calculation is straightforward for even n 's but a little tricky for the odd n 's. The last summation in (14) implements the dot product of the linear transformation, generated by the eigenimage method based on the Gram-Schmidt orthogonalization.

For the visualization purposes, we use (14) without the integrals over the pixel/voxel. So we derive the desired material distribution as a continuous function of (x_1, x_2, \dots, x_m) :

$$\alpha_d(x_1, x_2, \dots, x_m) = \sum_{l=1}^p t_d^l \sum_{k_1 \in \mathbb{Z}} \sum_{k_2 \in \mathbb{Z}} \dots \sum_{k_m \in \mathbb{Z}} c(k_1, k_2, \dots, k_m) \beta^n(x_1 - k_1, x_2 - k_2, \dots, x_m - k_m) \quad (15)$$

Equation (15) is $(n-1)$ -time differentiable continuous estimation over the entire dataset. As we will see in the Result Section, it preserves the continuity of the contours/surfaces of the desired object at the pixel/voxel boundaries.

4. RESULTS AND DISCUSSION

In order to evaluate the new approach we have simulated a 3D model using three spheres whose centers are the same. The first sphere has the intensity vector [147, 887, 959], similar to that of CSF in the human brain MRI for T1-, T2-, and PD-weighted images. We use [388, 240, 605] for regions outside of the first sphere and inside of the second one, and [323, 427, 833] for the regions outside the second sphere and inside of the third one. These two intensity vectors are similar to those of white matter (WM) and gray matter (GM) in T1-, T2-, and PD-weighted images, respectively. In this model, similar to the human brain, CSF is located in the central regions, GM is located in outer regions and WM is in between. We image this model with a thickness of 2 units and obtain 3 different images for each slice. Sample images are shown in Fig. 3.

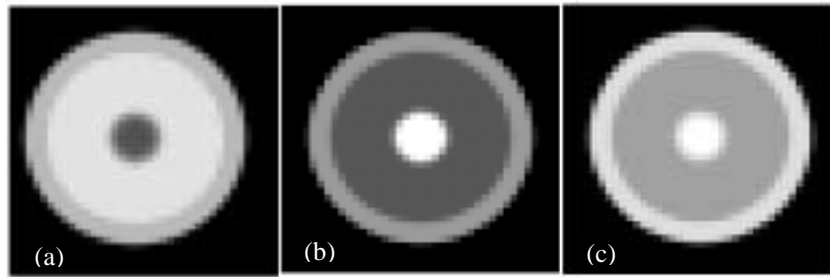


Fig. 3. Three images with signatures similar to T1, T2, and PD in MRI, obtained from the model, a) T1-weighted, b) T2-weighted, c) PD-weighted. Note that the display gray-scales are not the same for the above three images.

In the next step, we add noise to the images. As mentioned in references,^{6,7} we can assume a normally distributed noise for MR images. Also noise is added to each pixel, and its standard deviation (SD) is similar for different materials. The noisy images with $SD = 20$ of the same slice are shown in Fig. 4.

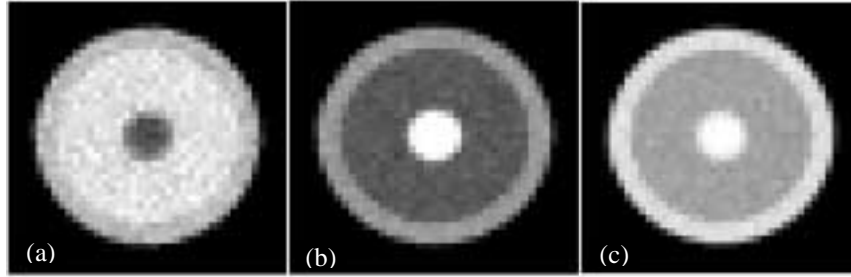


Fig. 4. Three images of the same slice as Fig. 3 with additive noise (SD = 20), (a) T1-weighted, (b) T2-weighted, (c) PD-weighted.

We have applied the standard eigenimage method, polynomial estimators, and the proposed B-spline estimators to the above dataset mixed with several noise levels, SD = 10, 20, 30, 40 and the original image set (SD = 0). We report the estimated volume for each material, and the biases and standard deviations of estimations for the simulation study in Tables II-VI.

The analytic and the actual (ground truth) volume of our simulated model are shown in Table I. It is observed that the model is accurate and its error is about 0.002%. In Tables II-VI, the standard eigenimage method is referred to as eigenimage. The polynomial methods with fixed 3×3, 1×3×5, 3×5, 5×5 neighboring structures are referred to as ply_3×3, ply_1×3×5, ply_3×5, ply_5×5, respectively. The separable B-spline estimators with basis functions β^2 and β^4 are referred to as spl_beta2 and spl_beta4, respectively. We examine separable, spl_beta3 (sep), and bell-shape, spl_beta3 (bel), cubic B-spline (β^3) estimators where the bell-shape basis function has the same form in all directions (radially symmetric).

Table I. The accuracy of the simulation model: a comparison between volumes of the materials within the model.

	Vol. Material #1	Vol. Material #2	Vol. Material #3	total volume
Analytical	200.1	4401.5	3976.1	8577.7
Actual (Ground Truth)	200.1	4401.6	3976.1	8577.8

Table II. The estimated volumes for each material and the total volume obtained by the eigenimage, polynomial, and B-spline estimators. The biases and standard deviations of the estimations are also given. Here SD = 0.

Estimator	Vol. Material #1	Vol. Material #2	Vol. Material #3	Total Volume	Estimation Bias	Standard Dev.
eigenimage	200.1	4401.6	3976.1	8577.8	-7.33×10^{-17}	3.44×10^{-15}
ply_3×3	200.1	4401.6	3976.1	8577.8	-7.24×10^{-17}	0.0075
spl_beta2	200.1	4401.6	3976.2	8577.9	1.01×10^{-6}	0.0111
spl_beta3 (sep)	200.1	4401.6	3976.1	8577.8	-5.75×10^{-7}	0.0112
spl_beta3 (bel)	200.1	4401.5	3976.0	8577.7	-3.52×10^{-6}	0.0110
spl_beta4	200.1	4401.6	3976.0	8577.8	-7.08×10^{-7}	0.0105
ply_1×3×5	200.1	4401.6	3977.1	8578.8	1.26×10^{-5}	0.0315
ply_3×5	200.1	4401.6	3976.9	8578.6	1.026×10^{-5}	0.0352
ply_5×5	200.1	4401.6	3977.0	8578.7	1.194×10^{-5}	0.0427

Table III. The estimated volumes for each material and the total volume obtained by the eigenimage, polynomial, and B-spline estimators. The biases and standard deviations of the estimations are also given. Here SD = 10.

Estimator	Vol. Material #1	Vol. Material #2	Vol. Material #3	Total Volume	Estimation Bias	Standard Dev.
eigenimage	206.3	4411.6	3961.8	8579.7	2.394×10^{-5}	0.0689
ply_3×3	206.3	4411.6	3961.8	8579.7	2.394×10^{-5}	0.0587
spl_beta2	206.3	4411.5	3961.9	8579.7	2.484×10^{-5}	0.0536
spl_beta3 (sep)	206.3	4411.6	3961.8	8579.7	2.326×10^{-5}	0.0527
spl_beta3 (bel)	206.3	4411.4	3961.7	8579.4	2.033×10^{-5}	0.0529
spl_beta4	206.3	4411.5	3961.8	8579.6	2.311×10^{-5}	0.0536
ply_1×3×5	206.3	4411.5	3962.9	8580.6	3.634×10^{-5}	0.0466
ply_3×5	206.3	4411.5	3962.7	8580.4	3.403×10^{-5}	0.0468
ply_5×5	206.3	4411.5	3962.8	8580.6	3.569×10^{-5}	0.0502

Table IV. The estimated volumes for each material and the total volume obtained by the eigenimage, polynomial, and B-spline estimators. The biases and standard deviations of the estimations are also given. Here SD = 20.

Estimator	Vol. Material #1	Vol. Material #2	Vol. Material #3	Total Volume	Estimation Bias	Standard Dev.
eigenimage	205.2	4424.4	3954.0	8583.6	7.523×10^{-5}	0.1340
ply_3×3	205.2	4424.4	3954.0	8583.6	7.523×10^{-5}	0.1137
spl_beta2	205.2	4424.3	3954.2	8583.7	7.596×10^{-5}	0.1033
spl_beta3 (sep)	205.2	4424.3	3954.0	8583.6	7.454×10^{-5}	0.1014
spl_beta3 (bel)	205.2	4424.2	3953.9	8583.3	7.160×10^{-5}	0.1018
spl_beta4	205.2	4424.4	3954.0	8583.5	7.444×10^{-5}	0.1033
ply_1×3×5	205.2	4424.2	3955.2	8584.5	8.707×10^{-5}	0.0742
ply_3×5	205.2	4424.2	3955.0	8584.4	8.487×10^{-5}	0.0698
ply_5×5	205.2	4424.2	3955.1	8584.5	8.644×10^{-5}	0.0673

Table V. The estimated volumes for each material and the total volume obtained by the eigenimage, polynomial, and B-spline estimators. The biases and standard deviations of the estimations are also given. Here SD = 30.

Estimator	Vol. Material #1	Vol. Material #2	Vol. Material #3	Total Volume	Estimation Bias	Standard Dev.
eigenimage	194.7	4411.3	3975.9	8581.9	5.339×10^{-5}	0.2063
ply_3×3	194.7	4411.3	3975.9	8581.9	5.339×10^{-5}	0.1749
spl_beta2	194.7	4411.3	3976.0	8582.0	5.459×10^{-5}	0.1587
spl_beta3 (sep)	194.7	4411.3	3975.8	8581.9	5.291×10^{-5}	0.1560
spl_beta3 (bel)	194.7	4411.2	3975.7	8581.7	4.996×10^{-5}	0.1565
spl_beta4	194.7	4411.4	3975.8	8581.9	5.278×10^{-5}	0.1589
ply_1×3×5	194.7	4411.4	3976.9	8582.9	6.654×10^{-5}	0.1086
ply_3×5	194.7	4411.4	3976.7	8582.8	6.410×10^{-5}	0.0997
ply_5×5	194.7	4411.4	3976.8	8582.9	6.585×10^{-5}	0.0909

Table VI. The estimated volumes for each material and the total volume obtained by the eigenimage, polynomial, and B-spline estimators. The biases and standard deviations of the estimations are also given. Here SD = 40.

Estimator	Vol. Material #1	Vol. Material #2	Vol. Material #3	Total Volume	Estimation Bias	Standard Dev.
eigenimage	237.2	4522.7	3846.6	8606.5	3.736×10^{-4}	0.2747
ply_3x3	237.2	4522.7	3846.6	8606.5	3.736×10^{-4}	0.2314
spl_beta2	237.2	4522.6	3846.9	8606.6	3.745×10^{-4}	0.2079
spl_beta3 (sep)	237.2	4522.6	3846.7	8606.5	3.728×10^{-4}	0.2043
spl_beta3 (bel)	237.2	4522.5	3846.6	8606.4	3.699×10^{-4}	0.2050
spl_beta4	237.2	4522.5	3846.7	8606.4	3.727×10^{-4}	0.2083
ply_1x3x5	237.1	4522.6	3847.9	8607.5	3.862×10^{-4}	0.1368
ply_3x5	237.1	4522.6	3847.6	8607.3	3.838×10^{-4}	0.1249
ply_5x5	237.1	4522.6	3847.8	8607.4	3.855×10^{-4}	0.1123

The behaviors of the standard deviation for the above estimators with regard to noise power are shown in Fig. 5.

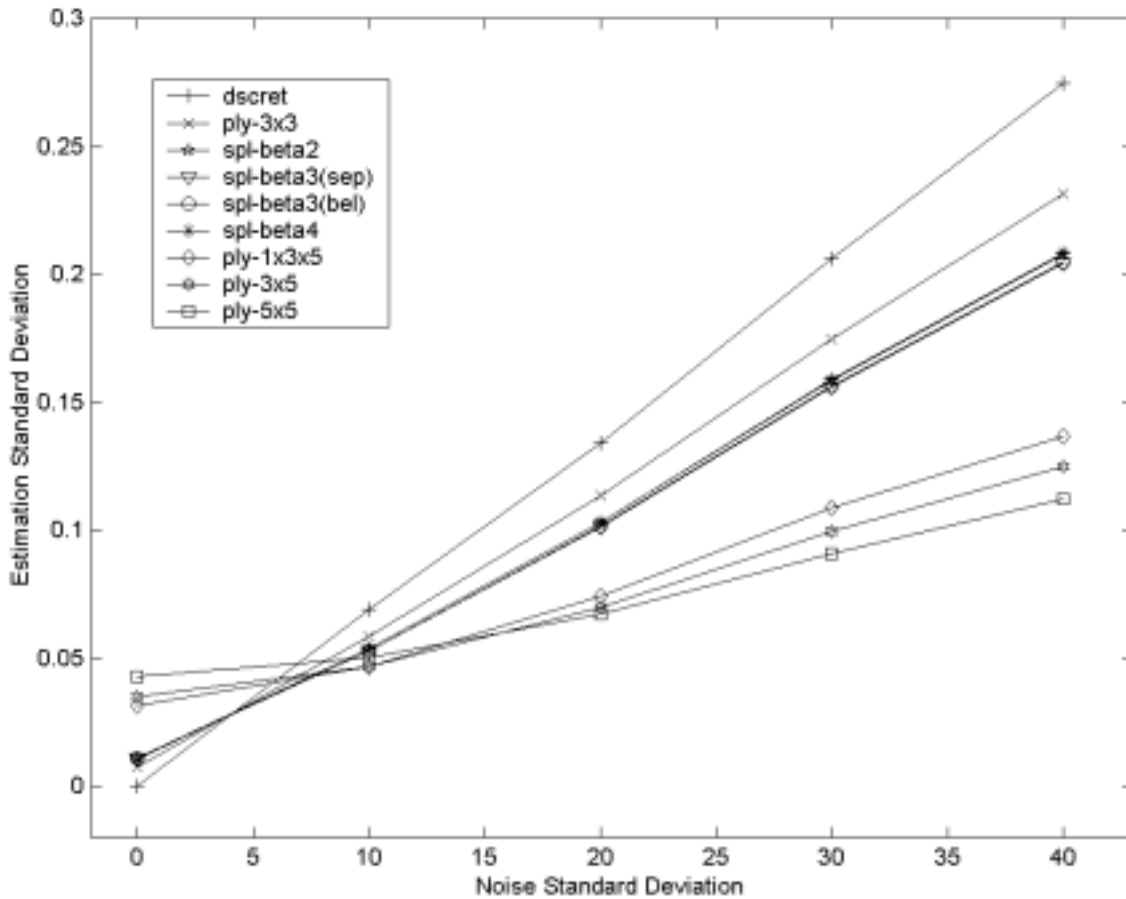


Fig. 5. The behaviors of the estimators versus SD of noise.

As it was mentioned in Section 3, formula (15) provides continuous estimation over the whole data set including the pixel/voxel boundaries. Fig. 6 shows the material distribution estimated by spl_beta4 and ply-5x5. The estimations are performed for the inner sphere (of the used simulation model) as the desired object on a single slice. In this experiment the

resolution of the actual space is increased 10 times. The intensities are proportional to the amount of the desired material estimated in each sub-voxel.

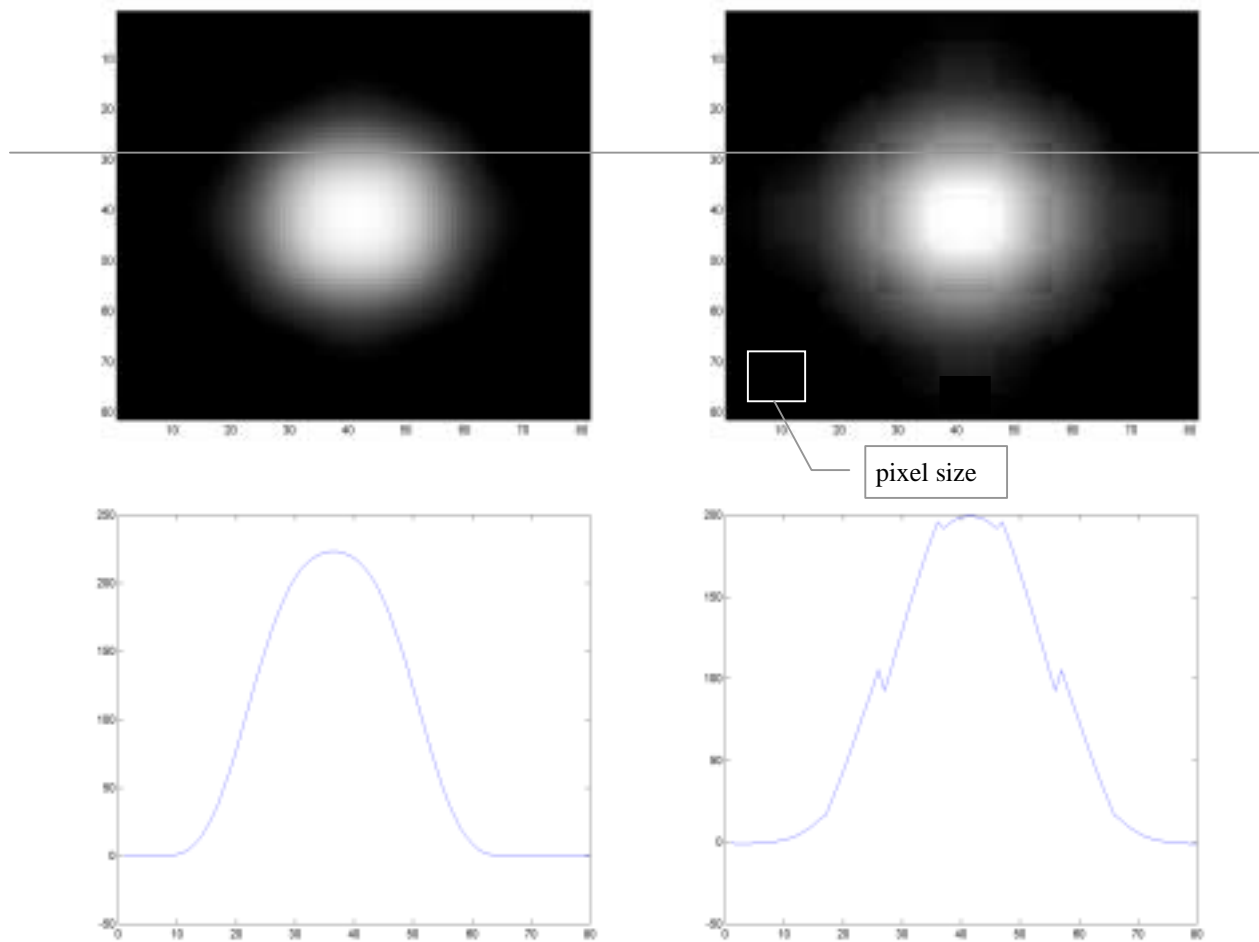


Fig. 6. The results of the inner sphere estimation by (a) spl_beta4, and (b) ply-5x5. The plots of a single row of the estimation shown on images (a) and (b), (c) corresponds to (a), (d) corresponds to (b).

The plots of estimations over a single line drawn on Figs. 6(a), (b) are shown in Figs. 6(c), (d), respectively. The plot in Fig. 6(d) clearly shows the discontinuities at the boundaries of the estimation performed by the polynomial method (ply-5x5). On the other hand, the plot shown in Fig. 6(c) indicates a continuous estimation provided by the B-spline estimator (spl_beta4). As a matter of fact, this plot, as well as the estimation itself, has third-derivative continuity at the boundaries. Although this property does not have a significant benefit for the volume calculation, it is valuable for the visualization applications.

Based on the experimental results we conclude that:

1. The overall volume estimated by the eigenimage, polynomial, and B-spline estimators are very similar to the ground of truth (actual volumes) in the absence of noise (Tables I, II). Also, with noise, the volumes calculated by the estimators are very similar.
2. In a high quality human brain MRI, the mean value to standard deviation ratio (mean/SD) is around 260 which is almost the same as our choice of $SD = 20$. Lower mean/SD ratios can be found in the fast scan MRI like fMRI. As it is shown in Table IV and Fig. 5, the standard deviation of the eigenimage is about twice that of the cnt_5x5 and 1.3 times of the spl_beta3, where $SD = 20$. Tables III–VI and more clearly Fig. 5, show that the standard deviations of the B-spline estimators in the presence of noise are smaller than that of the standard eigenimage method and larger than those of the polynomial estimators. So the new method has superior performance compared to the standard eigenimage.

3. Although most of the polynomial estimators have smaller standard deviation compared to the B-splines as the noise power increases, B-spline estimators have the superior capability of providing a continuous estimation over the entire dataset. This fact, shown by Fig. 5, is important for visualization purposes. It is also beneficial for 3D deformable models, and warping.^{11, 12}
4. The spline interpolation provides the minimum curvature among different interpolation methods (minimum curvature property⁹). Since we incorporated spline interpolation to our estimation, the new method provides the minimum curvature estimation as well. This feature is also important for visualization purposes.
5. The standard deviation curves behave very similarly for several B-spline estimators. Compared to the polynomial estimators that have different performance at different noise levels, with the examined B-spline estimators we get almost the same behavior. This fact indicates that there is little room for improvement of the results by switching between different B-splines in an adaptive manner based on the noise level. This may be considered as the second disadvantage of the new method compared to the polynomials.
6. In spite of the above fact, we expected to get lower standard deviations when using wider B-spline bases. Tables III-VI show that this does not hold in practice.
7. The polynomial estimators, based on our definition,¹ use the information of a fixed neighborhood. On the other hand, the B-spline estimators, introduced in this paper, use the information of the entire image for coefficient calculation. This is a part of our future work to determine how the choice of neighborhood structure affects the partial volume estimation. Also the B-spline interpolation method we used in this study is an exact interpolation. In future work, we will examine the non-exact version of it to study the smoothing effects.

REFERENCES

1. M. R. Siadat, H. Soltanian-Zadeh, "Partial volume estimation: an improvement for eigenimage method," *Proceedings of SPIE*, vol. 3979, pp. 646-655, 2000.
2. J. P. Windham, D. J. Peck, "Identification of abnormalities hidden by partial volume averaging effects," *Med. Phys.*, vol. 17, no. 3, pp. 550, May/June 1990.
3. D. J. Peck, J. P. Windham, H. Soltanian-Zadeh, J. R. Poebuck, "A fast and accurate algorithm for volume determination in MRI," *Med. Phys.*, vol. 19, no. 3, pp. 599-605, May/June 1992.
4. M. Levoy, "Display of surfaces from volume data," *IEEE Comput. Graphics, Applicat.*, vol. 8, no. 3, pp. 29-37, May 1988.
5. D. H. Laidlaw, K. W. Fleischer, A. H. Barr, "Partial-volume Bayesian classification of material mixtures in MR volume data using voxel histograms," *IEEE Trans. on Medical Imaging*, vol. 17, no. 1, February 1998.
6. J. P. Windham, M. A. Abd-Allah, D. A. Reimann, J. W. Froelich, A. M. Hagggar "Eigenimage filtering in MRI imaging," *Journal of Computer Assisted Tomography*, vol. 12, no. 1, pp. 1-9, Jan./Feb. 1988.
7. H. Soltanian-Zadeh, J. P. Windham, A. E. Yagle, "Optimal transformation for correcting partial volume averaging effects in magnetic resonance imaging," *IEEE Trans. on Nuclear Science*, vol. 40, no. 4, 1993.
8. D. H. Laidlaw, "Reply to: comments on partial-volume Bayesian classification of material mixtures in MR volume data using voxel histograms," *IEEE Trans. on Medical Imaging*, vol. 17, no. 6, December 1998.
9. M. Unser, "Splines a perfect fit for signal and image processing," *IEEE Signal Processing Magazine*, pp. 22-36, Nov. 1999.
10. P. Thevenaz, T. Blu, M. Unser, "Interpolation revisited," *IEEE Trans. On Medical Imaging*, vol. 19, no. 7, pp. 739-758, July 2000.
11. A. Ghanei, H. Soltanian-Zadeh, J. P. Windham, "A 3D deformable surface model for segmentation of objects from volumetric data in medical images," *Computers in Biology and Medicine*, vol. 28, pp. 239-253, 1998.
12. A. Ghanei, H. Soltanian-Zadeh, "A new method for homothetical warping of brain MRI images," *SPIE*, vol. 3661, pp. 1448-1454, 1999.