

FAST-PVE: Extremely Fast Markov Random Field Based Brain MRI Tissue Classification

Jussi Tohka

Department of Signal Processing, Tampere University of Technology,
P.O. Box 553 FIN-33101 Tampere, Finland
`jussi.tohka@tut.fi`

Abstract. We present an extremely fast method named FAST-PVE for tissue classification and partial volume estimation of 3-D brain magnetic resonance images (MRI) using a Markov Random Field (MRF) based spatial prior. The tissue classification problem is central to most brain MRI analysis pipelines and therefore solving it accurately and fast is important. The FAST-PVE method is experimentally confirmed to tissue classify a standard MR image in under 10 seconds with the quantitative accuracy similar to other state of art methods. A key component of the FAST-PVE method is the fast ICM algorithm, which is generally applicable to any MRF-based segmentation method, and formally proven to produce the same segmentation result as the standard ICM algorithm.

Keywords: Segmentation, brain imaging, magnetic resonance imaging.

1 Introduction

In its simplest form, brain magnetic resonance (MR) image segmentation refers to classification of each voxel in the image to one of three principal tissue classes - white matter (WM), gray matter (GM), and cerebro-spinal fluid (CSF). This basic segmentation problem, termed tissue classification, is an essential pre-processing step for cortical surface extraction [1] and voxel-based morphometry [2], which are perhaps the most important techniques to analyze anatomical MR images in neuroscience. Complications in tissue classification include the intensity non-uniformities, partial volume effect and scanner specificity of MR imaging (MRI) as well as image noise [3]. These complications have been addressed in multiple works and it is not a purpose of the current paper to review these in detail.

Markov Random Fields (MRFs) are an elegant technique to incorporate spatial priors in medical image segmentation and, in particular, they are very often an essential component of the methods aiming to brain MR image segmentation. Examples of widely used segmentation methods for MRI based on MRFs include [4–7]. Based on MRFs, one can derive a spatial prior penalizing the segmentations that are noisy or otherwise unlikely. However, a drawback of MRFs is that their use leads to a complex optimization problem, often being the main computational bottleneck of the segmentation methods. On the other hand, speed of the

image segmentation is critical since, for example, human brain studies presently utilize a large number of brain images that all have to be segmented. This paper centers around the method of [7], which is used for tissue classification in several neuroimaging software packages and pipelines [8–10], and describes an extremely fast version, termed FAST-PVE, of it running in under 10 seconds for a high resolution 3-D MRI. The speed of FAST-PVE is maybe easiest to appreciate when compared to a recent Field Programmable Gate Array (FPGA) based implementation of the same method [11] for which computation times in the order of 1 minute for a 3-D image were reported. FAST-PVE combines several innovative techniques and approximations to achieve the speed up. The most central of these is the fast iterated conditional modes (ICM) algorithm used to solve the MRFs. Although this and other techniques are presented in the context of [7], many of them, including fast ICM, are general and could be used in conjunction with other methods as well.

Section 2 describes the FAST-PVE method and the most of speed-up techniques. Section 3 introduces the fast ICM algorithm. Section 4 presents experimental results with real and simulated 3-D MRI data confirming the high speed and the quantitative accuracy of FAST-PVE. Section 5 concludes the paper.

2 FAST-PVE Method

Given a 3-D MR image $X = \{x_i \in \mathbb{R} : i = 1, \dots, m\}$, where $i = (i_x, i_y, i_z)$ is the voxel index and m is the number of voxels, FAST-PVE computes an estimate of the fractional tissue content of each of the three main tissue types (WM, GM and CSF) in each voxel [7]. More specifically, it computes $W^* = \{w_{i_j}^* : i = 1, \dots, m, j \in \{CSF, GM, WM\}\}$, where $w_{i_j}^*$ is fraction of tissue type j in the voxel i . The partial volume coefficients (PVCs) satisfy $w_{i_j}^* \in [0, 1]$ and $\sum_j w_{i_j}^* = 1$. The process is termed partial volume (PV) estimation. The method assumes that the image X has been corrected for intensity non-uniformities and skull-stripped.

To solve the under-constrained estimation problem, FAST-PVE assumes that no voxel contains more than two-types of tissue and resorts to a two-stage optimization problem. In the first stage, PV classification, voxels are tissue classified into $K = 6$ tissue types representing the main tissue types (pure tissue classes $\mathcal{P} = \{CSF, GM, WM\}$) and partial volume mixtures of two tissue types (mixed classes $\mathcal{M} = \{\{Background, CSF\}, \{CSF, GM\}, \{GM, WM\}\}$). Certain mixed tissues are omitted because there are very few such voxels in the brain. In the second step, PVC estimation, FAST-PVE estimates the proportions of each tissue type within each voxel using the information from the PV classification step to make the estimation problem well-posed.

Partial Volume Classification. The PV classification is casted as an optimization problem:

$$C^* = \arg \max_C \log(p(x_i|c_i)) + U(C), \quad (1)$$

where $\log p(x_i|c_i)$ is the log-likelihood of x_i given that the tissue type of the voxel i is c_i and $U(C)$ is a spatial prior penalizing label configurations that are

unlikely. This optimization problem is derivable from the maximum a posteriori (MAP) principle [5]. The spatial prior $U(\cdot)$ is modeled by an MRF, or more accurately, $U(C)$ is the energy function of the Gibbs distribution defining the MRF [12]. Define

$$a_{ik} = \begin{cases} 2 & : c_i = c_k \\ 1 & : c_i \in \mathcal{P} \text{ and } c_k \in \mathcal{M} \text{ and } c_i \in c_k \text{ or vice versa} \\ -1 & : \text{otherwise} \end{cases} \quad (2)$$

Then,

$$U(C) = (\beta/2) \sum_{i=1}^m \sum_{k \in N_i} \frac{a_{ik}}{d(i, k)}, \quad (3)$$

where β is a parameter controlling the strength of the spatial prior, N_i is the 26-neighborhood around voxel i , $d(i, k)$ is the distance between centres of voxels i and k . In this work, $\beta = 0.1$ unless otherwise mentioned. The fast ICM algorithm to solve the optimization problem is presented in Section 3.

The likelihood $p(x_i|c_i)$ is assumed to follow Gaussian density $g(x_i|\mu_{c_i}, \Sigma_{c_i})$ with the mean μ_{c_i} and the variance Σ_{c_i} for pure tissue classes. For mixed classes, the likelihood is obtained by marginalizing the densities over all the possible values of the PVC w [13]: $p(x_i|c_i = \{j, k\}) = \int_0^1 g(x_i; \mu(w), \Sigma(w))dw$, where $\mu(w) = w\mu_j + (1-w)\mu_k$; $\Sigma(w) = w^2\Sigma_j + (1-w)^2\Sigma_k$. The integral has no closed form solution and hence numerical integration must be used to evaluate it. For the computational efficiency, it is important to select a numerical integration scheme as fast as possible without sacrificing too much accuracy. The numerical integration in FAST-PVE is based on the extended Simpson's rule with 8 equally sized intervals [14]. This selection was done by comparing different integration schemes numerically and selecting the fastest scheme yielding consistently under 0.1% error when compared to very accurate approximation obtained by the extended trapezoidal rule with 1000 equally sized intervals¹.

PVC Estimation. In the PVC estimation step, FAST-PVE estimates the proportions of each tissue type within each voxel. If $c_i^* = j$, that is labeling of voxel i indicates that it is a pure voxel, $w_{i_j}^* = 1$ and for all $k \neq j$ $w_{i_k}^* = 0$. If i is a mixed voxel, $c_i^* = \{j, k\}$, then we employ the maximum-likelihood principle:

$$w_{i_j}^* = \arg \max_{w \in [0,1]} \log g(x_i|\mu(w), \Sigma(w)), \quad (4)$$

where $\mu(w), \Sigma(w)$ are as above. Furthermore, $w_{i_k}^* = 1 - w_{i_j}^*$ and all the other PVCs are zero. The maximum-likelihood PVC-estimate (4) is solved by a simple grid search.

Parameter Estimation. The model parameters $\mu_j, \Sigma_j, j = 1, \dots, 3$ must be estimated before the PV estimation. An efficient way to do this is to first label

¹ In our experiments, the Simpson's rules consistently outperformed the trapezoidal rules, the Simpson's rule completely broke down with lesser number than 8 intervals, and also more advanced Simpson's 3/8 and Boole's rules failed to attain the a priori set error bound when the number of intervals was less than 8.

each voxel in the image X by a pure label belonging to \mathcal{P} , and then to estimate the parameters based on the intensity distribution of the labeled voxels for each tissue class j denoted here as I_j . However, one does not want to assume that the initial labeling is particularly accurate (as it has to be computed rapidly) and, secondly, also PVE causes intensities in I_j to contain outliers. From this it follows that the set of samples I_j contains samples that are not drawn from the distribution describing the tissue type j which leads to problems when applying conventional techniques, such as the maximum-likelihood principle, for parameter estimation. The problem is solved by first detecting the possible outliers by a simple analysis of the spatial configuration of the initial labeling, i.e., excluding the intensities of the voxels for which one or more their 26-neighbours do not have label j from the set I_j ². After initial trimming robust point estimators are applied to estimate μ_j, Σ_j based on trimmed I_j . In particular, we use the maximum covariance determinant (MCD) method (also termed least trimmed squares (LTS) in one dimension) for the task [15]. The accuracy of this kind of parameter estimation has been confirmed in our earlier studies [7, 16] and the reasons for selecting MCD as the parameter estimation method (instead of, e.g., more conventional M-estimators) has been explained in detail in [7], where the method just described is abbreviated as trimmed MCD (TMCD).

The initial labeling is generated by the incremental k-means technique proposed in [16] which is very fast and accurate enough. The algorithm itself is a specific multistart adaptation of the standard k-means algorithm. A specific point in the algorithm is that only those image voxels whose gradient's magnitude after low-pass filtering is smaller than a certain threshold are retained in the update of the means step of k-means. More specifically, if $D = \{d_i\} = \|\nabla X\|$ denotes the gradient magnitude image, the objective function to be optimized is

$$\sum_{j \in \mathcal{P}} \sum_{k \in V} \min_j \|x_k - \mu_j^{init}\|^2; V = \{i : d_i < 2std(D)\}, \quad (5)$$

where $std(D)$ is the standard deviation of gradient magnitude values and μ_j^{init} are the mean values respect to which the optimization is done. By excluding the voxels with a high gradient magnitude, we hope to eliminate some bias which is due to PVE as after low-pass filtering, these voxels are likely to be situated near tissue type boundaries. To speed up the initial labeling, we utilize a technique proposed for the expectation maximization (EM) algorithm in [17]. The k-means objective function is approximated by

$$\sum_{j \in \mathcal{P}} \sum_{l=0}^{255} \min_j h_l \|c_l - \mu_j^{init}\|^2, \quad (6)$$

where c_l and h_l are the bin-centre and the number of elements of the bin l in the 256-bin histogram of $\{x_i : i \in V\}$. The idea behind this technique is to re-cast

² Here 26-connectivity is used instead of original 6-connectivity of [7] because this typically leads to slightly better performance.

the clustering problem of hundreds of thousands data-points into a smaller 256 data-point clustering problem. If there are 256 intensity values (or less) in the image (6) equals (5), otherwise (6) approximates (5).

3 Fast ICM

Algorithm 1. Fast ICM

```

1: for  $i = 1$  to  $m$  do
2:   for  $k = 1$  to  $K$  do
3:     Compute  $l_{ik} \leftarrow \log p(x_i|c_i)$ 
4:   end for
5:   Initialize  $c_i^0 \leftarrow \arg \max_k l_{ik}$ 
6:   Set  $b_i \leftarrow 1$ 
7: end for
8: Set  $t \leftarrow 0$ 
9: repeat
10:  Set  $t \leftarrow t + 1$ 
11:  for  $i = 1$  to  $m$  do
12:    if  $b_i = 1$  then
13:      Set  $c_i^t \leftarrow \arg \max_k l_{ik} + U(c_i = k|C_{N_i}^t)$ 
14:      if  $c_i^t \neq c_i^{t-1}$  then
15:        for all  $j \in N_i$  do
16:          Set  $b_j \leftarrow 1$ 
17:        end for
18:      end if
19:      Set  $b_i \leftarrow 0$  // Remove this line to get the standard ICM
20:    end if
21:  end for
22: until  $C^t = C^{t-1}$ 
23: return  $C^t$ 

```

The most central component to the speed of FAST-PVE is the fast ICM algorithm for solving the optimization problem (1). The pseudocode of fast ICM is given in Algorithm 1, where c_i^t denotes the label of voxel i at the iteration t and the ordered set ($|N_i|$ -tuple) of the labels of the neighbours of the voxel i at the iteration t is $C_{N_i}^t = \{c_j^t : j \in N_i, j < i\} \cup \{c_j^{t-1} : j \in N_i, j > i\}$ ³. Although the algorithm is here presented in the context of MRI segmentation, it is general and applicable to any MRF based segmentation problem. As we will formally show, the fast ICM algorithm converges to the same local maximum as the original ICM algorithm by Besag [18] but requiring considerably less

³ We use a set notation for this tuple because this way the meaning of it becomes more easily apparent. $C_{N_i}^t$ needs to be defined as a tuple rather than a set for the proof of the correctness of the fast ICM algorithm.

computation time. We want to note that the ICM algorithm is not as effective as more modern algorithms [19]. However, it is computationally efficient and, as we have argued before [3], local solutions obtainable by it can in certain situations compare favourably to (more) global solutions available, e.g., based on graph cuts. Also, the majority of MRI segmentation methods based on MRFs use this local algorithm.

The ICM algorithm is based on the observations that (1) can be minimized locally by considering each voxel in turn and the label of the voxel i depends only on the labels of its neighbouring voxels $C_{Ni} = \{c_j : j \in N_i\}$ and the intensity x_i [18]. The difference between the Fast and standard ICM algorithms is the book keeping variables $b_i, i = 1, \dots, m$, which track whether a label c_i can change during the iteration t . Especially, if we omit line 19 of the below algorithm, we obtain the standard ICM algorithm (b_i always equals one in this case).

In the Algorithm 1, the line 13 is the most expensive one requiring $|N_i| + 1 = 27$ additions and multiplications for each label and each voxel using a look-up table based implementation. The key point is that in practice in later stages of the algorithm very few voxels change their label during an iteration. Thus, keeping track of those voxels which cannot change their label can lead to massive savings in computation time by avoiding the computations of line 13. The trick is simple: voxels whose neighbours have not changed their label since the previous evaluation round must have the same label as in the previous round. More formally, if $C_{Ni}^t = C_{Ni}^{t-1}$, then $U(c_i = k | C_{Ni}^t) = U(c_i = k | C_{Ni}^{t-1})$, and consequently $c_i^t = \arg \max_k l_{ik} + U(c_i = k | C_{Ni}^t) = \arg \max_k l_{ik} + U(c_i = k | C_{Ni}^{t-1}) = c_i^{t-1}$.

To see that the modification is correct, i.e., it leads to the same result as the standard ICM algorithm, we introduce the notation b_i^t to refer to the value of b_i at line 12 during the iteration t . Note that 1) $j \in N_i$ if and only if $i \in N_j$ and 2) $j \notin N_j$. These are basic properties of the neighborhood systems in MRFs [12], which we use repeatedly below. Now we are ready to prove the following proposition from which the correctness of the algorithm follows since $c_i^t = c_i^{t-1}$ if $C_{Ni}^t = C_{Ni}^{t-1}$ as pointed out above:

Proposition 1. For all voxels $v = 1, \dots, m$, it holds that a) if $C_{Nv}^t \neq C_{Nv}^{t-1}$ then $b_v^t = 1$ and b) if $C_{Nv}^t = C_{Nv}^{t-1}$ then $b_v^t = 0$.

Proof. We prove the part a) first. Since $C_{Nv}^t \neq C_{Nv}^{t-1}$, there is such $j \in N_v$ for which $c_j^{t^*} \neq c_j^{t^*-1}$, where $t^* = t$ if $j < v$ and $t^* = t - 1$ if $j > v$. Since $j \in N_v$, also $v \in N_j$. Thus, b_v is set to 1 at line 16 when the voxel counter i is at j and the iteration counter is at t^* . Now, b_v 's value can be changed to 0 only at line 19 and only when the voxel counter i is at v . The voxel counter will be at v next time during the iteration t (since $t^* + 1 = t$ if $j > v$ and $t^* = t$ if $j < v$) and thus $b_v^t = 1$ at the line 12 during the iteration t as required and the proof is complete. The part b) of the proposition is equivalent to the statement: if $b_v^t = 1$ then $C_{Nv}^t \neq C_{Nv}^{t-1}$ and we prove this statement instead of the original one. At line 19 during the iteration $t - 1$, b_v is set to 0. Now, since $b_v^t = 1$ at line 12 during the iteration t , this means that line 16 must be visited at least once between the two

consecutive visits at the voxel v (during the iterations $t-1$ and t). Thus, v must be a neighbour of a voxel $j > v$ for which $c_j^{t-1} \neq c_j^{t-2}$ or a voxel $u < v$ for which $c_u^t \neq c_u^{t-1}$. In the former case, since $v \in N_j$, also $j \in N_v$. Further, c_j^{t-1} is the y^{th} element of $C_{N_v}^t$ and c_j^{t-2} is the y^{th} element of $C_{N_v}^{t-1}$, where y is an arbitrary integer from 1 to $|N_v|$. Thus, $C_{N_v}^t \neq C_{N_v}^{t-1}$. In the latter case, the reasoning is equivalent. \square

Only the part a) is needed for the correctness of the algorithm. The part b) shows that the algorithm is 'optimal', i.e., the book keeping burden cannot be (in practice) reduced while maintaining the correctness of the algorithm ⁴. An algorithm based on a similar idea has been previously presented in [20] for inhomogeneity modelling. However, our presentation of the algorithm and the proof of its correctness are much more concise. Moreover, the correctness proof in [20] required to track the voxel behaviour two iterations back while a single iteration is sufficient for our proof.

4 Implementation, Experiments and Results

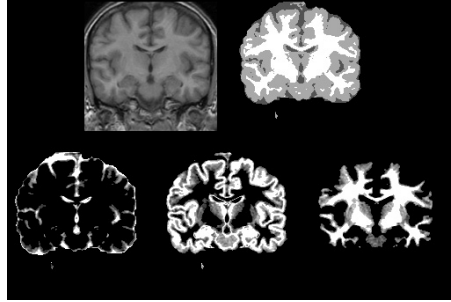
The implementation of the FAST-PVE is Matlab based with the most significant bottlenecks, partial volume classification and PVC estimation, coded in C and compiled into mex-files. The computation times were measured with a standard laptop (Intel Core Duo CPU running at 2.26 GHz with 64 bit Windows 7 operating system, Matlab version 2010b). The source-code is available at http://www.cs.tut.fi/~jupeto/matlab_code/pvemri2.zip

We demonstrate the speed and accuracy of the method with two datasets, which have become standards in evaluating tissue classification/partial volume estimation algorithms: BrainWeb and IBSR2. With simulated data from the BrainWeb database [21, 22], we demonstrate that the PVC estimates by the new faster method are of similar quantitative accuracy than those by previous, much slower, implementations, reported in [7, 23]. A skull stripped (only CSF, GM and WM labelled voxels) T1-weighted BrainWeb phantom of 1 mm^3 voxel size with no inhomogeneity was used with different levels of Rician distributed random noise (1%, 3%, 5%, 7% and 9% of the maximum image intensity). The image dimensions were $181 \times 217 \times 181$. The accuracy of the PVC estimates was measured for each tissue type j separately by root mean square error $RMSE_j = \sqrt{(1/m) \sum_{i=1}^m (t_{i_j} - w_{i_j}^*)^2}$, where t_{i_j} is the ground truth PVC and $w_{i_j}^*$ the estimated one. $RMSE$ values and computation times are reported in Table 1. The $RMSE$ values of FAST-PVE were almost identical to the ones by the earlier C-based implementation of the method (see [23], the average $RMSE$ difference was 0.007), but the computation time was reduced from over 3 minutes to under 10 seconds. As already mentioned a recent FPGA-based implementation of the method [11] tissue classified comparable images in approximately 1

⁴ We could try, in addition, to utilize symmetries of the prior, but this would result in a significantly more complicated book-keeping strategy and the cost of the book-keeping would erase potential savings by the book-keeping.

Table 1. *RMSE* values and computation times of FAST-PVE with the Brainweb data

noise level	1%	3%	5 %	7%	9%
$RMSE_{WM}$	0.09	0.10	0.11	0.13	0.16
$RMSE_{GM}$	0.08	0.11	0.16	0.20	0.26
$RMSE_{CSF}$	0.06	0.09	0.13	0.17	0.22
time (seconds)	8.90	7.60	7.64	7.76	7.86

**Fig. 1.** Counter-clockwise: An example of a coronal slice of an IBSR image, the PVC estimates of CSF, GM, and WM, and the hardened segmentation

minute per image, thus the FAST-PVE offers over six-fold speed-up compared to it. The break-down computation time spend on each step of the method was on average: 1.89 s (24 % of the total computation time) for initial segmentation by k-means, 1.33 s (17 %) for TMCD parameter estimation, 2.21 s (28 % for computations of log-likelihoods, 1.81 s (23 %) for fast ICM and 0.65 s (8 %) for PVC estimation. This demonstrates that no significant speed ups were possible by implementing the initial segmentation and parameter estimation in C instead of Matlab.

The IBSR2 dataset consists of 18 T1-weighted MR brain images and their manual segmentations. These are provided by the Center for Morphometric Analysis at Massachusetts General Hospital <http://www.cma.mgh.harvard.edu/ibsr/>. The image size was $256 \times 256 \times 128$ with the voxel sizes varying between $0.84 \times 0.84 \times 1.5 \text{ mm}^3$ and $1 \times 1 \times 1.5 \text{ mm}^3$. The images have been non-uniformity corrected by the CMA 'autoseg' biasfield correction routines. We skull-stripped them using Brain Surface Extractor (BSE) [5].

With IBSR2 data, we demonstrate that the method works equally well and fast with real data and provide also timing comparisons to BrainSuite09 [5]. The BrainSuite, implemented in C++, was selected as a reference method due to its methodological similarity to FAST-PVE and since the computational efficiency has been an important factor in its design. With real data, quantitative evaluation of the PV estimation is restricted to evaluation of the quality of the subsequent hard segmentations. Therefore, we hardened the PVCs by selecting the tissue type (CSF, GM, or WM) with the maximal PVC $w_{i_j}^*$ to represent voxel

Table 2. Average Dice coefficients, standard deviations of Dice coefficients, average computation time per image and maximum computation time per image with the IBSR2 dataset. FAST-PVE (S-ICM) corresponds to FAST-ICM coupled with the standard ICM algorithm instead of the fast ICM algorithm. FAST-PVE with fast ICM computed the segmentations over 2 times faster than the other methods.

Method	Dice GM	Dice WM	ave time	max time
FAST-PVE	0.78 ± 0.08	0.86 ± 0.04	3.96 s	5.62 s
FAST-PVE (S-ICM)	0.78 ± 0.08	0.86 ± 0.04	9.26 s	13.7 s
BrainSuite09	0.72 ± 0.09	0.83 ± 0.08	19.2 s	51.0 s

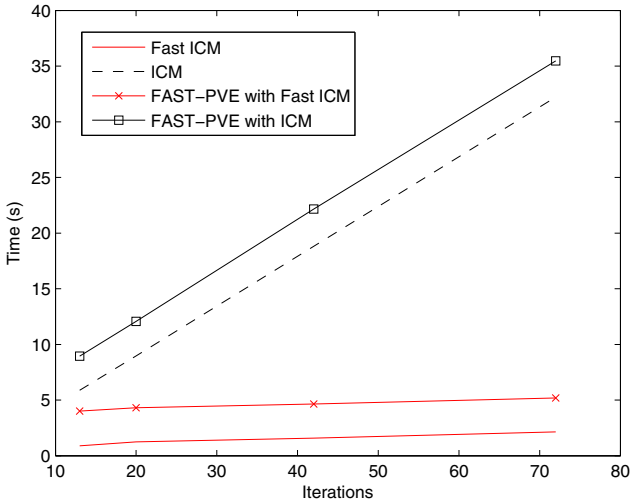


Fig. 2. Computation times of FAST-PVE with (standard) ICM and Fast ICM with four different values of β (0.1, 0.3, 0.5 and 1.0) leading to different numbers of ICM iterations. The lines labeled 'ICM' and 'Fast ICM' give the computation times of the ICM part of FAST-PVE. The computation time increased linearly with the number of ICM iterations with standard ICM, but was almost constant with Fast ICM. Savings in computation time due to Fast ICM become more apparent with higher β .

i (see [3]). An example of PVC estimates and the hardened 3-class segmentation is shown in Fig. 1. The Dice coefficient between the manual and automatic segmentation was used as the performance measure. It was computed separately for each tissue class (CSF,GM,WM). The Dice coefficient between the sets U and V is defined as $2|U \cap V|/(|U| + |V|)$. The value 1.0 corresponds to the perfect match and the value 0.0 corresponds to no match. The average Dice values over 18 images of FAST-PVE and Brainsuite09 are presented in Table 2 which shows also the average and maximal computation times (over 18 images) of the methods. FAST-PVE was, on average, more accurate than Brainsuite09 and also approximately 5 times faster. The Dice coefficients of FAST-PVE are similar to other state of the art methods - although it should be noted that the purpose

of the method is to obtain PVC estimates, not the hard segmentations, which might call for a different kind of MRF model (see [3] for details).

The table 2 shows also the computation times of FAST-PVE but with the standard ICM instead of the fast ICM algorithm. The segmentations by these two methods are the same by Proposition 1, and hence the Dice coefficients are equal. However, FAST-PVE with fast ICM was over 2 times faster than FAST-PVE with standard ICM which demonstrates the importance of the fast ICM algorithm. The savings in computation times due to fast ICM become more apparent when the number of ICM iterations increase which in practice occurs with the increase in β parameter. This is demonstrated in Fig. 2, where the computation times are plotted against the number of ICM iterations. The computation time of standard ICM increased linearly with the number of ICM iterations while the computation time of fast ICM was almost constant.

5 Conclusions

We have presented an extremely fast method named FAST-PVE for tissue classification and partial volume estimation of 3-D brain MRI. The method was confirmed to segment a standard MR image in under 10 seconds with the quantitative accuracy similar to other state of art methods. A key component of the FAST-PVE method is the fast ICM algorithm, which is generally applicable to any MRF-based segmentation method, and formally proven to produce the same segmentation result as the standard ICM algorithm. The source-code of FAST-PVE is available from http://www.cs.tut.fi/~jupeto/matlab_code/pvemri2.zip under a free software licence.

Acknowledgments. Financially supported by Academy of Finland grants 130275 and 263785. Thanks to Claude Lepage for suggesting to take a careful look into the numerical integration scheme used in FAST-PVE.

References

1. Dale, A., Fischl, B., Sereno, M.: Cortical surface-based analysis. I. segmentation and surface reconstruction. *Neuroimage* 9, 179–194 (1999)
2. Ashburner, J., Friston, K.J.: Voxel-based morphometry—the methods. *Neuroimage* 11(6 pt. 1), 805–821 (2000)
3. Tohka, J., Dinov, I.D., Shattuck, D.W., Toga, A.W.: Brain MRI tissue classification based on local Markov random fields. *Magn. Reson. Imaging* 28(4), 557–573 (2010)
4. Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P.: Automated model-based tissue classification of MR images of the brain. *IEEE Trans. Med. Imaging* 18(10), 897–908 (1999)
5. Shattuck, D.W., Sandor-Leahy, S.R., Schaper, K.A., Rottenberg, D.A., Leahy, R.M.: Magnetic resonance image tissue classification using a partial volume model. *Neuroimage* 13(5), 856–876 (2001)

6. Zhang, Y., Brady, M., Smith, S.: Segmentation of brain mr images through a hidden random markov field model and the expectation-maximization algorithm. *IEEE Trans. Med. Imaging* 20(1), 45–57 (2001)
7. Tohka, J., Zijdenbos, A., Evans, A.: Fast and robust parameter estimation for statistical partial volume models in brain MRI. *Neuroimage* 23(1), 84–97 (2004)
8. Evans, A.C.: Large-scale morphometric analysis of neuroanatomy and neuropathology. *Anat. Embryol (Berl)* 210(5-6), 439–446 (2005)
9. Bellec, P., Carbonell, F.M., Perlberg, V., Lepage, C., Lyttelton, O., Fonov, V., Janke, A., Tohka, J., Evans, A.C.: A neuroimaging analysis kit for matlab and octave. In: *Proceedings of the 17th International Conference on Functional Mapping of the Human Brain* (2011)
10. Gaser, C.: VBM8 toolbox, <http://dbm.neuro.uni-jena.de/vbm/>
11. Koo, J., Evans, A., Gross, W.: 3-D brain tissue classification on FPGAs. *IEEE Trans. Image Proc.* 18, 2735–2746 (2009)
12. Li, S.: *Markov Random Field Modeling in Image Analysis*. Springer, London (2009)
13. Santago, P., Gage, H.D.: Quantification of MR brain images by mixture density and partial volume modeling. *IEEE Trans. Med. Imaging* 12(3), 566–574 (1993)
14. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: *Numerical Recipes, 3rd edn. The Art of Scientific Computing*. Cambridge University Press (2007)
15. Rousseeuw, P.J., Leroy, A.M.: *Robust Regression and Outlier Detection*. Wiley-Interscience, New York (1987)
16. Manjon, J., Tohka, J., Garcia-Marti, G., Carbonell-Caballero, J., Lull, J., Marti-Bonmati, L., Robles, M.: Robust MRI brain tissue parameter estimation by multi-stage outlier detection. *Magn. Reson. Med.* 59, 866–873 (2008)
17. Wang, Y., Adah, T., Xuan, J., Szabo, Z.: Magnetic resonance image analysis by information theoretic criteria and stochastic site models. *IEEE Trans. Inform. Tech. Biomed.* 5(2), 150–158 (2001)
18. Besag, J.: On the statistical analysis of dirty pictures. *J. R. Stat. Soc. Ser. B* 48(3), 259–302 (1986)
19. Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., Rother, C.: A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Trans. Patt. Analysis Mach. Intell.* 30, 1068–1080 (2008)
20. Huang, F., Narayan, S., Wilson, D., Johnson, D., Zhang, G.Q.: A fast iterated conditional modes algorithm for water-fat decomposition in mri. *IEEE Trans. Med. Imaging* 30, 1480–1492 (2011)
21. Collins, L., Zijdenbos, A., Kollokian, V., Sled, J., Kabani, N., Holmes, C., Evans, A.: Design and construction of a realistic digital brain phantom. *IEEE Trans. Med. Imaging* 17(3), 463–468 (1998)
22. Kwan, R.S., Evans, A., Pike, G.: MRI simulation-based evaluation of image-processing and classification methods. *IEEE Trans. Med. Imag.* 18(11), 1085–1097 (1999)
23. Manjon, J.V., Tohka, J., Robles, M.: Improved estimates of partial volume coefficients from noisy brain MRI using spatial context. *Neuroimage* 53(2), 480–490 (2010)