

# N3 Bias Field Correction Explained as a Bayesian Modeling Method

Christian Thode Larsen<sup>1</sup>, J. Eugenio Iglesias<sup>2,3</sup>, and Koen Van Leemput<sup>1,2,4</sup>

<sup>1</sup> Department of Applied Mathematics and Computer Science, Technical University of Denmark

<sup>2</sup> Martinos Center for Biomedical Imaging, MGH, Harvard Medical School, USA

<sup>3</sup> Basque Center on Cognition, Brain and Language, Spain

<sup>4</sup> Departments of Information and Computer Science and of Biomedical Engineering and Computational Science, Aalto University, Finland

**Abstract.** Although N3 is perhaps the most widely used method for MRI bias field correction, its underlying mechanism is in fact not well understood. Specifically, the method relies on a relatively heuristic recipe of alternating iterative steps that does not optimize any particular objective function. In this paper we explain the successful bias field correction properties of N3 by showing that it implicitly uses the same generative models and computational strategies as expectation maximization (EM) based bias field correction methods. We demonstrate experimentally that purely EM-based methods are capable of producing bias field correction results comparable to those of N3 in less computation time.

## 1 Introduction

Due to its superior image contrast in soft tissue without involving ionizing radiation, magnetic resonance imaging (MRI) is the *de facto* modality in brain studies, and it is widely used to examine other anatomical regions as well. MRI suffers from an imaging artifact commonly referred to as “intensity inhomogeneity” or “bias field”, which appears as low-frequency multiplicative noise in the images. This artifact is present at all magnetic field strengths, but is more prominent at the higher fields that see increasing use (e.g., 3T or 7T data). Since intensity inhomogeneity negatively impacts any computerized analysis of the MRI data, its correction is often one of the first steps in MRI analysis pipelines.

A number of works have proposed bias field correction methods that are integrated into tissue classification algorithms, typically within the domain of brain MRI analysis [1–7]. These methods often rely on generative probabilistic models, and combine Gaussian mixtures to model the image intensities with a spatially smooth, multiplicative model of the bias field artifact. Cast as a Bayesian inference problem, fitting these models to the MRI data employs expectation-maximization (EM) [8] optimizers to estimate some [7] or all [1, 3, 4, 6] of the model parameters. Specifically tailored for brain MRI analysis applications, these methods encode strong prior knowledge about the number and spatial distribution of tissue types present in the images. As such, they cannot be used out of the box to bias field correct imaging data from arbitrary anatomical regions.

In contrast, the popular N3 [9] bias field correction algorithm does not require any prior information about the MRI input. This allows N3 to correct images of various locations and contrasts, and even automatically handle images that contain pathology. However, despite excellent performance and widespread use, its underlying bias field correction mechanism is not well understood. Specifically, the original paper [9] presents N3 as a relatively heuristic recipe for increasing the “frequency content” of the histogram of an image, by performing specific iterative steps without optimization of any particular objective function.

This paper aims to demonstrate how N3 is in fact intimately linked to EM-based bias field correction methods. In particular, N3 uses the same generative models and bias field estimation computations; however, instead of using dedicated Gaussian mixture models that encode specific prior anatomical knowledge, N3 uses generic models with a very large number of components (200) that are fitted to the histogram by a regularized least-squares method.

The contribution of this paper is twofold. First, to the best of our knowledge, this is the first study offering theoretical insight into why the seemingly heuristic N3 iterations yield such successful bias field estimations. Second, we demonstrate experimentally on datasets of 3T and 7T brain scans that standard EM-based methods, using far less components, are able to produce comparable bias field estimation performance at reduced computational cost.

## 2 Methods

In this section, we first describe the N3 bias field correction method and its practical implementation. We then present EM-based bias field correction and the generative model it is based upon. Finally, we build an analogy between the two methods, thereby pointing out their close similarities.

### 2.1 The N3 method in its practical implementation

The following description is based on version 1.12<sup>1</sup> of the N3 method. In order to facilitate relating the method to a generative model in subsequent sections, we deviate from the notational conventions used in the original paper [9]. Furthermore, whereas the original paper only provides a high-level description of the algorithm (including integrals in the continuous domain, etc.), here we describe the actual implementation in which various discretization, interpolation, and other processing steps are performed.

Let  $\mathbf{d} = (d_1, \dots, d_N)^T$  be the intensities of the  $N$  voxels of a MRI scan, and let  $\mathbf{b} = (b_1, \dots, b_N)^T$  be the corresponding gains due to the bias field. As commonly done in the bias field correction literature [1, 3, 4, 6], N3 assumes that  $\mathbf{d}$  and  $\mathbf{b}$  have been log-transformed, such that the effect of  $\mathbf{b}$  is additive. The central idea behind N3 is that the histogram of  $\mathbf{d}$  is a blurred version of the histogram of the true, underlying image due to convolution with the histogram of  $\mathbf{b}$ , under the

<sup>1</sup> Source code freely available from <http://packages.bic.mni.mcgill.ca/tgz/>.

assumption that  $\mathbf{b}$  has the shape of a zero-mean Gaussian with known variance. The algorithm aims to reverse this by means of Wiener deconvolution and to estimate a smooth bias field model accordingly. This reversal process is repeated iteratively, because it was found to improve the bias field estimates [9].

**Deconvolution step:** The first step of the algorithm is to deconvolve the histogram. Given the current bias field estimate denoted  $\tilde{\mathbf{b}}$ , a normalized histogram with  $K = 200$  bins of bias field corrected data  $\mathbf{d} - \tilde{\mathbf{b}}$  is computed<sup>2</sup>. The bin centers are given by

$$\tilde{\mu}_1 = \min(\mathbf{d} - \tilde{\mathbf{b}}), \quad \tilde{\mu}_K = \max(\mathbf{d} - \tilde{\mathbf{b}}), \quad \tilde{\mu}_k = \tilde{\mu}_1 + (k-1)h, \quad (1)$$

where  $h = (\tilde{\mu}_K - \tilde{\mu}_1)/(K-1)$  is the bin width, and the histogram entries  $\{v_k, k = 1, \dots, K\}$  are filled using the following interpolation model:

$$v_k = \frac{1}{N} \sum_{i=1}^N \varphi \left[ \frac{d_i - \tilde{b}_i - \tilde{\mu}_k}{h} \right], \quad \varphi[s] = \begin{cases} 1 - |s| & \text{if } |s| < 1 \\ 0, & \text{otherwise.} \end{cases}$$

Defining  $\hat{\mathbf{v}}$  as a padded, 512-dimensional vector such that  $\hat{\mathbf{v}} = (\mathbf{0}_{156}^T, \mathbf{v}^T, \mathbf{0}_{156}^T)^T$ , where  $\mathbf{v} = (v_1, \dots, v_K)^T$  and  $\mathbf{0}_{156}$  is an all-zero 156-dimensional vector, the histogram is deconvolved by

$$\hat{\boldsymbol{\pi}} \leftarrow \mathbf{F}^{-1} \mathbf{D} \mathbf{F} \hat{\mathbf{v}}. \quad (2)$$

Here  $\mathbf{F}$  denotes the  $512 \times 512$  Discrete Fourier Transform matrix with elements

$$F_{n,k} = e^{-2\pi j(k-1)(n-1)/512}, \quad n, k = 1, \dots, 512$$

and  $\mathbf{D}$  is a  $512 \times 512$  diagonal matrix with elements

$$D_k = \frac{f_k^*}{|f_k|^2 + \gamma}, \quad k = 1, \dots, 512$$

where  $\gamma$  is a constant value set to  $\gamma = 0.1$ , and  $\mathbf{f} = (f_1, \dots, f_{512})^T = \mathbf{F} \mathbf{g}$ . Here  $\mathbf{g}$  denotes a 512-dimensional vector that contains a wrapped Gaussian kernel with variance

$$\tilde{\sigma}^2 = \frac{f^2}{8 \log 2}, \quad (3)$$

such that

$$\mathbf{g} = (g_1, \dots, g_{512})^T, \quad g_l = \begin{cases} h \mathcal{N}((l-1)h | 0, \tilde{\sigma}^2) & \text{if } l = 1, \dots, 256 \\ g_{512-l+1}, & \text{otherwise,} \end{cases} \quad (4)$$

where  $f$  denotes a user-specified full-width-at-half-maximum parameter (0.15 by default), and  $\mathcal{N}(\cdot | \mu, \sigma^2)$  denotes a Gaussian distribution with mean  $\mu$  and variance  $\sigma^2$ .

After  $\hat{\boldsymbol{\pi}}$  has been computed by means of Eq. (2), any negative weights are set to zero, and the padding is removed in order to obtain the central deconvolved 200-entry histogram  $\tilde{\boldsymbol{\pi}}$ .

<sup>2</sup> A flat bias field:  $\tilde{\mathbf{b}} = \mathbf{0}$  is assumed in the first iteration.

**Bias correction step:** When the histogram  $\tilde{\pi}$  has been deconvolved, the corresponding “corrected” intensity  $\tilde{d}_{\mu_l}$  in the deconvolved histogram is estimated at each bin center  $\tilde{\mu}_l, l = 1, \dots, K$  by

$$\tilde{d}_{\mu_l} = \sum_k w_k^l \tilde{\mu}_k \quad \text{with} \quad w_k^l = \frac{\mathcal{N}(\tilde{\mu}_l | \tilde{\mu}_k, \tilde{\sigma}_k^2) \tilde{\pi}_k}{\sum_{k'} \mathcal{N}(\tilde{\mu}_l | \tilde{\mu}_{k'}, \tilde{\sigma}_{k'}^2) \tilde{\pi}_{k'}},$$

and a “corrected” intensity  $\tilde{d}_i$  is found in every voxel by linear interpolation:

$$\tilde{d}_i = \sum_{l=1}^K \tilde{d}_{\mu_l} \varphi \left[ \frac{d_i - \tilde{b}_i - \tilde{\mu}_l}{h} \right], \quad \varphi[s] = \begin{cases} 1 - |s| & \text{if } |s| < 1 \\ 0, & \text{otherwise.} \end{cases}$$

Finally, a residual  $\mathbf{r} = \mathbf{d} - \tilde{\mathbf{d}}$  is computed and smoothed in order to obtain a bias field estimate:

$$\tilde{\mathbf{b}} = \Phi \tilde{\mathbf{c}} \quad (5)$$

where

$$\tilde{\mathbf{c}} \leftarrow (\Phi^T \Phi + N\beta\Psi)^{-1} \Phi^T \mathbf{r}. \quad (6)$$

Here  $\Phi$  is a  $N \times M$  matrix of  $M$  spatially smooth basis functions, where element  $\Phi_{i,m}$  evaluates the  $m$ -th basis function in voxel  $i$ ;  $\Psi$  is a positive semi-definite matrix that penalizes curvature of the bias field; and  $\beta$  is a user-determined regularization constant (the default is  $\beta = 10^{-7}$ ).

**Post-processing:** N3 alternates between the deconvolution step and the bias field correction step until the standard deviation of the difference in bias estimates between two iterations drops below a certain threshold (default:  $\varsigma = 10^{-3}$ ). By default, N3 operates on a subsampled volume (factor 4). After convergence, the bias field estimate is exponentiated back into the original intensity domain, where it is subsequently fitted with Eq. (6), i.e., with  $\mathbf{r} = \exp(\tilde{\mathbf{b}})$ . The resulting coefficients are then used to compute a final bias field estimate by evaluation of Eq. (5) with  $\Phi$  at full image resolution. The uncorrected data is finally divided by the bias field estimate in order to obtain the corrected volume.

## 2.2 EM-based bias field estimation

In the following we describe the generative model and parameter optimization strategy underlying EM-based bias field correction methods<sup>3</sup>.

<sup>3</sup> Several well-known variants only estimate a subset of the parameters considered here – e.g., in [1] the mixture model parameters are assumed to be known, while [3] uses fixed, spatially varying prior probabilities of tissue types.

**Generative model:** Maintaining the notation  $\mathbf{d}$  to denote a log-transformed image and  $\mathbf{b} = \Phi\mathbf{c}$  to denote a parametric bias field model with parameters  $\mathbf{c}$ , the “true”, underlying image  $\mathbf{d} - \mathbf{b}$  is assumed to be a set of  $N$  independent samples from a Gaussian mixture model with  $K$  components – each with its own mean  $\mu_k$ , variance  $\sigma_k^2$ , and relative frequency  $\pi_k$  (where  $\pi_k \geq 0, \forall k$  and  $\sum_k \pi_k = 1$ ). Given the model parameters  $\boldsymbol{\theta} = (\mu_1, \dots, \mu_k, \sigma_1^2, \dots, \sigma_K^2, \pi_1, \dots, \pi_K, c_1, \dots, c_M)^T$ , the probability of an image is therefore

$$p(\mathbf{d}|\boldsymbol{\theta}) = \prod_{i=1}^N \left[ \sum_{k=1}^K \mathcal{N}(d_i - \sum_{m=1}^M c_m \Phi_{i,m} | \mu_k, \sigma_k^2) \pi_k \right]. \quad (7)$$

The generative model is completed by a prior distribution on its parameters, which is typically of the form

$$p(\boldsymbol{\theta}) \propto \exp[-\lambda \mathbf{c}^T \Psi \mathbf{c}],$$

where  $\lambda$  is a user-specified regularization hyperparameter and  $\Psi$  is a positive semi-definite regularization matrix. This model encompasses approaches where bias field smoothness is imposed either solely through the choice of basis functions (i.e.,  $\lambda = 0$ , as in [3]), or through regularization only (i.e.,  $\Phi = \mathbf{I}$ , as in [1]). The prior is uniform with respect to the mixture model parameters.

**Parameter optimization:** According to Bayes’s rule, the maximum a posteriori (MAP) parameters are given by

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \log p(\boldsymbol{\theta}|\mathbf{d}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} [\log p(\mathbf{d}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta})]. \quad (8)$$

By exploiting the specific structure of  $p(\mathbf{d}|\boldsymbol{\theta})$  given by Eq. (7), this optimization can be performed conveniently using a generalized EM (GEM) algorithm [8, 3]. In particular, GEM iteratively builds a lower bound  $\varphi(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}})$  of the objective function that touches it at the current estimate  $\tilde{\boldsymbol{\theta}}$  of the model parameters (E step), and subsequently improves  $\varphi(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}})$  with respect to the parameters (M step) [8, 10]. This procedure automatically guarantees to increase the value of the objective function at each iteration. Constructing the lower bound involves computing soft assignments of each voxel  $i$  to each class  $k$ :

$$w_k^i = \frac{\mathcal{N}(d_i - \sum_m \tilde{c}_m \Phi_{i,m} | \tilde{\mu}_k, \tilde{\sigma}_k^2) \tilde{\pi}_k}{\sum_{k'} \mathcal{N}(d_i - \sum_m \tilde{c}_m \Phi_{i,m} | \tilde{\mu}_{k'}, \tilde{\sigma}_{k'}^2) \tilde{\pi}_{k'}}, \quad (9)$$

which yields the following lower bound:

$$\varphi(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}}) = \sum_i \left[ \sum_k w_k^i \log \left( \frac{\mathcal{N}(d_i - \sum_m c_m \Phi_{i,m} | \mu_k, \sigma_k^2) \pi_k}{w_k^i} \right) \right] - \lambda \mathbf{c}^T \Psi \mathbf{c}. \quad (10)$$

Optimizing Eq. (10) simultaneously for the Gaussian mixture model parameters and bias field parameters is difficult. However, optimization with respect to the

mixture model parameters for a given set of bias field parameters is closed form:

$$\tilde{\mu}_k \leftarrow \frac{\sum_i w_k^i (d_i - \sum_m \tilde{c}_m \Phi_{i,m})}{\sum_i w_k^i}, \quad \tilde{\sigma}_k^2 \leftarrow \frac{\sum_i w_k^i (d_i - \sum_m \tilde{c}_m \Phi_{i,m} - \tilde{\mu}_k)^2}{\sum_i w_k^i} \quad (11)$$

$$\tilde{\pi}_k \leftarrow \frac{\sum_i w_k^i}{N}. \quad (12)$$

Similarly, for a given set of mixture model parameters the optimal bias field parameters are given by

$$\tilde{\mathbf{c}} \leftarrow (\Phi^T \mathbf{S} \Phi + 2\lambda \Psi)^{-1} \Phi^T \mathbf{S} \mathbf{r}, \quad (13)$$

with

$$s_k^i = \frac{w_k^i}{\tilde{\sigma}_k^2}, \quad s_i = \sum_k s_k^i, \quad \mathbf{S} = \text{diag}(s_i), \quad \tilde{d}_i = \frac{\sum_k s_k^i \tilde{\mu}_k}{\sum_k s_k^i}, \quad \mathbf{r} = \mathbf{d} - \tilde{\mathbf{d}}.$$

Valid GEM algorithms solving Eq. (8) are now obtained by alternately updating the voxels' class assignments (Eq. (9)), the mixture model parameters (Eqns. (11) and (12)), and the bias field parameters (Eq. (13)), in any order or arrangement.

### 2.3 N3 as an approximate MAP parameter estimator

Having laid out the details of both N3 and EM-based bias field correction, we are in a position to illustrate parallels between these two methods. In particular, as we describe below, *N3 implicitly uses the same generative model as EM methods* and shares the exact same bias field parameter update (up to numerical discretization aspects). The only difference is that, whereas EM methods fit their Gaussian mixture models by maximum likelihood estimation, N3 does so by regularized least-squares fitting of the mixture model to the histogram entries. Thus, whereas N3 was conceived as iteratively deconvolving Gaussian bias field histograms from the data without optimizing any particular objective function, its successful performance can be readily understood from a standard Bayesian modeling perspective.

Considering the generative model described in Section 2.2, we postulate that N3 uses  $K = 200$  Gaussian distributions that are equidistantly spaced between the minimum and maximum intensity, i.e., the parameters  $\{\mu_k\}$  are fixed (Eq. (1)). Furthermore, all Gaussians are forced to have an identical variance that is also fixed:  $\sigma_k^2 = \tilde{\sigma}^2, \forall k$ , where  $\tilde{\sigma}^2$  is given by Eq. (3). Thus, the only free parameters in N3 are the relative class frequencies  $\pi_k, k = 1, \dots, K$  and the bias field parameters  $\mathbf{c}$ . We start by analyzing the update equations for  $\mathbf{c}$ .

For the specific scenario where  $\sigma_k^2 = \tilde{\sigma}^2, \forall k$ , the EM bias field update equation (Eq. (13)) simplifies to

$$\tilde{\mathbf{c}} \leftarrow (\Phi^T \Phi + 2\tilde{\sigma}^2 \lambda \Psi)^{-1} \Phi^T \mathbf{r}, \quad \text{with} \quad \tilde{d}_i = \sum_k w_k^i \tilde{\mu}_k, \quad \mathbf{r} = \mathbf{d} - \tilde{\mathbf{d}},$$

where  $w_k^i$  is given by Eq. (9). When the hyperparameter  $\lambda$  is set to the value  $\lambda = N\beta/2/\bar{\sigma}^2$  this corresponds directly to the N3 bias field update equation Eq. (6), where the only difference is that N3 explicitly computes  $\tilde{d}_{\mu_i}$  for just 200 discrete intensity values and interpolates to obtain  $\tilde{d}_i$ , instead of computing  $\tilde{d}_i$  directly for each individual voxel.

For the remaining parameters  $\boldsymbol{\pi} = (\pi_1, \dots, \pi_K)^T$ , N3 implicitly uses a regularized least-squares fit of the resulting mixture model to the zero-padded normalized histogram  $\hat{\boldsymbol{v}}$ :

$$\hat{\boldsymbol{\pi}} \leftarrow \underset{\boldsymbol{x}}{\operatorname{argmax}} \|\hat{\boldsymbol{v}} - \mathbf{A}\boldsymbol{x}\|^2 + \gamma\|\boldsymbol{x}\|^2, \quad (14)$$

where  $\mathbf{A}$  is a  $512 \times 512$  matrix in which each column contains the same Gaussian-shaped basis function, shifted by an offset identical to the column index:

$$\mathbf{A} = \begin{pmatrix} g_1 & g_{512} & \cdots & g_2 \\ g_2 & g_1 & \cdots & g_3 \\ \vdots & \vdots & \ddots & \vdots \\ g_{512} & g_{511} & \cdots & g_1 \end{pmatrix},$$

i.e., the first column contains the vector  $\boldsymbol{g}$  defined in Eq. (4), and the remaining columns contain cyclic permutations of  $\boldsymbol{g}$ . To see why Eq. (14) is equivalent to Eq. (2), consider that because  $\mathbf{A}$  is a circulant matrix, it can be decomposed as

$$\mathbf{A} = \mathbf{F}^{-1} \boldsymbol{\Lambda} \mathbf{F} \quad \text{with} \quad \boldsymbol{\Lambda} = \operatorname{diag}(\boldsymbol{f}),$$

where  $\mathbf{F}$  and  $\boldsymbol{f}$  were defined in Section 2.1. The solution of Eq. (14) is given by

$$\begin{aligned} \hat{\boldsymbol{\pi}} &\leftarrow (\mathbf{A}^T \mathbf{A} + \gamma \mathbf{I})^{-1} \mathbf{A}^T \hat{\boldsymbol{v}} = (\mathbf{F}^{-1} \boldsymbol{\Lambda}^H \mathbf{F} \mathbf{F}^{-1} \boldsymbol{\Lambda} \mathbf{F} + \gamma \mathbf{I})^{-1} \mathbf{F}^{-1} \boldsymbol{\Lambda}^H \mathbf{F} \hat{\boldsymbol{v}} \\ &= (\mathbf{F}^{-1} \boldsymbol{\Lambda}^H \boldsymbol{\Lambda} \mathbf{F} + \gamma \mathbf{F}^{-1} \mathbf{F})^{-1} \mathbf{F}^{-1} \boldsymbol{\Lambda}^H \mathbf{F} \hat{\boldsymbol{v}} = \mathbf{F}^{-1} \underbrace{(\boldsymbol{\Lambda}^H \boldsymbol{\Lambda} + \gamma \mathbf{I})^{-1}}_D \boldsymbol{\Lambda}^H \mathbf{F} \hat{\boldsymbol{v}}, \end{aligned}$$

where  $\boldsymbol{\Lambda}^H$  denotes the *Hermitian transpose* of  $\boldsymbol{\Lambda}$  and where we have used the properties that  $\mathbf{A}^T = \boldsymbol{\Lambda}^H$  and  $\mathbf{F}^H = 512 \cdot \mathbf{F}^{-1}$ .

An example of N3's mixture model fitted this way will be shown in Figure 1. The periodic end conditions in  $\mathbf{A}$  have no practical impact on the histogram fit, as the support of the Gaussian-shaped basis functions is limited, and only the parameters of the 200 central basis functions are retained after fitting. Although this is clearly an *ad hoc* approach, the results are certainly not unreasonable, and N3 thereby maintains a close similarity to purely EM-based bias field correction methods.

### 3 Experiments

**Implementation:** In order to experimentally verify our theoretical analysis and quantify the effect of replacing the N3 algorithm of Section 2.1 with the EM

algorithm described in Section 2.2 and *vice versa*, we implemented both methods in Matlab. For our implementation of N3, we took care to mimic the original N3 implementation (a Perl script binding together a number of C++ binaries) as faithfully as possible. Specifically, we used identically placed cubic B-spline basis functions  $\Phi$ , identical regularizer  $\Psi$ , and the same sub-sampling scheme and parameter settings as in the original method. Our EM implementation shares the same characteristics and preprocessing steps where possible, so that any experimental difference in performance between the two methods is explained by algorithmic rather than technological aspects.

During the course of our experiments, we observed that N3’s final basis function fitting operation in the original intensity domain (described in Section 2.1, “Post-processing”) actually hurts the performance of the bias field correction. Also, we noticed that N3’s default threshold value to detect convergence ( $\varsigma = 10^{-3}$ ) tends to stop the iterations prematurely. To ensure a fair comparison with the EM method, we henceforth report the performance of N3 (Matlab) with the final fitting operation switched off, and with a more conservative threshold value that guarantees full convergence of the method ( $\varsigma = 10^{-5}$ ).

For our EM implementation, we report results for mixture models of  $K = 3$ ,  $K = 6$ , and  $K = 9$  components. We initialize the algorithm with the bias field coefficients set to zero:  $\mathbf{c} = \mathbf{0}$  (no bias field); with equal relative class frequencies:  $\pi_k = 1/K, \forall k$ ; equidistantly placed means given by Eq. (1) and equal variances given by  $\sigma_k^2 = ((\max(\mathbf{d}) - \min(\mathbf{d}))/K)^2, \forall k$ . For a given bias field estimate, the algorithm alternates between re-computing  $w_k^i, \forall i, k$  (Eq. (9)) and updating the mixture model parameters (Eqns. (11) and (12)), until convergence in the objective function is detected (relative change between iterations  $< 10^{-6}$ ). Subsequently, the bias field is updated (Eq. 13) and the whole process is repeated until global convergence is detected (relative change in the objective function  $< 10^{-5}$ ).

**MRI data and brain masking:** We tested both bias field correction methods on two separate datasets of T1-weighted brain MR scans. The first dataset was acquired on several 3T Siemens Tim Trio scanners using a multi-echo MPRAGE sequence with a voxel size of  $1.2 \times 1.2 \times 1.2 \text{ mm}^3$ . It consists of 38 subjects scanned twice with varying intervals for a total of 76 volumes. The second dataset consists of 17 volumes acquired on a 7T Siemens whole-body MRI scanner using a multi-echo MPRAGE sequence with a voxel size of  $0.75 \times 0.75 \times 0.75 \text{ mm}^3$ . Since N3 bias field correction of brain images is known to work well only on scans in which all non-brain tissue has been removed [11], both datasets were skull-stripped using FreeSurfer<sup>4</sup>.

**Evaluation metrics:** Since the true bias field effect in our MR images is unknown, we compare the two methods using a segmentation-based approach. In particular, we use the coefficient of joint variation [12] in the white and gray matter as an evaluation metric, measured in the original (rather than logarithmic)

<sup>4</sup> <https://surfer.nmr.mgh.harvard.edu/>

domain of image intensities, after bias field correction. This metric is defined as  $CJV = \frac{\sigma_1 + \sigma_2}{|\mu_1 - \mu_2|}$ , where  $(\mu_1, \sigma_1)$  and  $(\mu_2, \sigma_2)$  denote the mean and standard deviation of intensities within the white and the gray matter, respectively. Compared to the coefficient of variation defined as  $CV = \sigma_1/\mu_1$ , which is also commonly used in the literature [11, 13] and which measures only the intensity variation within the white matter, the CJV additionally takes into account the remaining separation between white and gray matter intensities.

In order to compute the CJV, we used FreeSurfer to obtain automatic white and gray matter segmentations, which we then eroded once in order to limit the influence of boundary voxels, which are typically affected by partial volume effects. We observed that the segmentation performance of FreeSurfer was sub-optimal in the 7T data because this software has problems with field strengths above 3T. This problem was ameliorated by bias field correcting the 7T scans with SPMS<sup>5</sup> prior to feeding them to FreeSurfer.

In addition to reporting CJV results for the two methods, we also report their run time on a 64bit CentOS 6.5 Linux PC with 24 gigabytes of RAM, an Intel(R) Xeon(R) E5430 2.66GHz CPU, and with Matlab version R2013b installed. For the sake of completeness, we also include the CJV and run time results for the original N3 software (default parameters, with the exception of the spacing between the B-spline control points – see below).

**Stiffness of the bias field model:** The stiffness of the B-spline bias field model is determined both by the spacing between the B-spline control points (affecting the number of basis functions in  $\Phi$ ) and the regularization parameter of  $\Psi$  that penalizes curvature ( $\beta$  in N3, and  $\lambda$  in the EM method).

As recommended in [13], we used a spacing of 50 mm instead of the N3 default<sup>6</sup>, as it is known to be too large for images obtained at higher-field strengths. Finding a common, matching value for the regularization parameter in both methods proved difficult, since we observed that the methods perform best in different ranges. Therefore, for the current study we computed average CJV scores for both methods over a wide range of values. We report results for the setting that worked best for each method and for each dataset separately<sup>7</sup>.

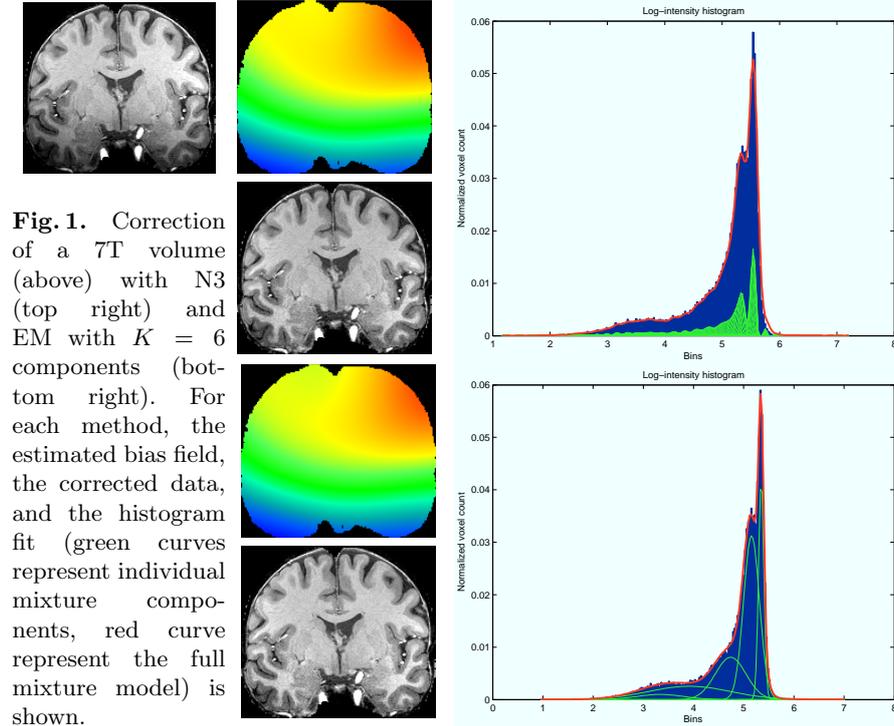
## 4 Results

Figure 1 shows the histogram fit and the bias field estimate of both our N3 implementation and the EM method with  $K = 6$  Gaussian components on a representative scan from the 7T dataset. In general, the histogram fit works well for both methods; however for N3 a model mismatch can be seen around the high-intensity tail. This is the result of zeroing negative weights after Wiener filtering.

<sup>5</sup> <http://www.fil.ion.ucl.ac.uk/spm/>

<sup>6</sup> 200 mm, appropriate for the 1.5T data the method was originally developed for.

<sup>7</sup> A more elaborate validation study would determine the optimal values on a separate training dataset; however, this is outside the scope of the current workshop paper.



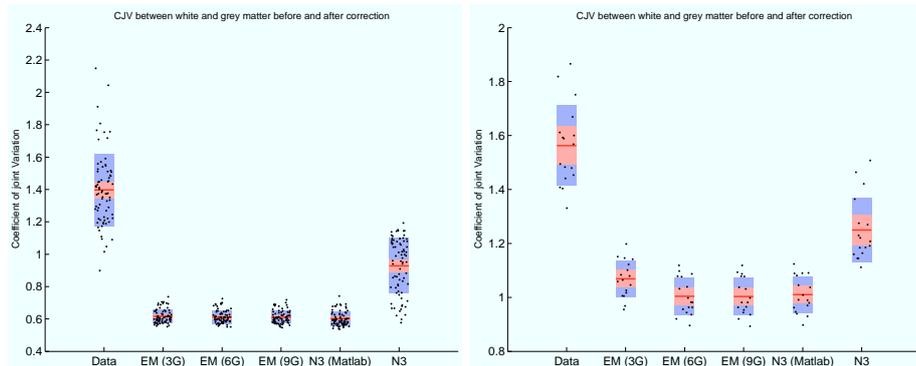
**Fig. 1.** Correction of a 7T volume (above) with N3 (top right) and EM with  $K = 6$  components (bottom right). For each method, the estimated bias field, the corrected data, and the histogram fit (green curves represent individual mixture components, red curve represent the full mixture model) is shown.

Dataset	Average computation time (seconds)				
	EM (3G)	EM (6G)	EM (9G)	N3 (Matlab)	N3
3T	12.7	20.7	29.7	86.0	53.5
7T	50.6	79.2	102.0	415.5	170.8

**Table 1.** Average computation time for correcting a volume within each dataset.

Figure 2 shows the CJV in the two test datasets, before bias field correction as well as after, using the EM method (for  $K = 3$ ,  $K = 6$ , and  $K = 9$  components), our Matlab N3 implementation, and the original N3 software. Overall, the EM and N3 (Matlab) methods perform comparably, except for EM with  $K = 3$  components which seems to have too few degrees of freedom in the 7T dataset. The original N3 implementation is provided as a reference only; its underperformance compared to our own implementation is to be expected since its settings were not tuned the same way.

Table 1 shows the average computation time of each method. Due to the much higher resolution of the 7T data, computation time increased for all methods when correcting this dataset. In all cases, the EM correction ran three to six times faster than the N3 Matlab implementation, depending on the number of components in the mixture. As before, results for the original N3 method are provided for reference only.



**Fig. 2.** Scatter plots showing the CVJ between white and gray matter in the 3T (left) and 7T (right) datasets. Lower CVJ equates to better performance. The red line represents the mean, while the blue box covers one standard deviation of the data and the red box covers the 95% confidence interval of the mean.

## 5 Discussion

In this paper we have explained the successful bias field correction properties of the N3 method by showing that it implicitly uses the same type of generative models and computational strategies as EM-based bias field correction methods. Experiments on MRI scans of healthy brains indicate that, at least in this application, purely EM-based methods can achieve performance similar to N3 at a reduced computational cost.

Future work should evaluate how replacing N3’s highly constrained 200-component mixture model with more general mixture models affects bias field correction performance in scans containing pathology. Conversely, while N3’s idiosyncratic histogram fitting procedure was found to work well in our experiments, it is worth noting that it precludes N3 from taking advantage of specific prior domain knowledge when such is available. For instance, the skull stripping required to make N3 work well in brain studies [11] typically involves registration of the images into a standard template space, which means that probabilistic brain atlases are available at no additional cost. It is left as further work to evaluate whether this puts N3 at a potential disadvantage compared to EM-based methods, which can easily take this form of extra information into account [3, 7]. Future validation studies should also include comparisons with the publicly available N4ITK implementation [14], which employs a more elaborate but heuristic B-spline fitting procedure in the bias field computations.

## Acknowledgments

This research was supported by the NIH NCRR (P41-RR14075), the NIH NIBIB (R01EB013565), TEKES (ComBrain), the Danish Council for Strategic Research (J No. 10-092814) and financial contributions from the Technical University of

Denmark. The authors would like to thank Jonathan Polimeni for supplying 7T data for our tests.

## References

1. W. M. Wells, I., Grimson, W.E.L., Kinikis, R., Jolesz, F.A.: Adaptive segmentation of MRI data. *IEEE Transactions on Medical Imaging* **15**(4) (August 1996) 429 – 442
2. Held, K., Kops, E., Krause, B., Wells, W., Kikinis, R., Muller-Gartner, H.: Markov random field segmentation of brain MR images. *IEEE Transactions on Medical Imaging* **16**(6) (Dec 1997) 878–886
3. Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P.: Automated model-based bias field correction of MR images of the brain. *IEEE Transactions on Medical Imaging* **18**(10) (October 1999) 885 – 896
4. Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P.: Automated model-based tissue classification of MR images of the brain. *IEEE Transactions on Medical Imaging* **18**(10) (October 1999) 897 – 908
5. Pham, D., Prince, J.: Adaptive fuzzy segmentation of magnetic resonance images. *IEEE Transactions on Medical Imaging* **18**(9) (Sept 1999) 737–752
6. Zhang, Y., Brady, M., Smith, S.: Segmentation of brain MR images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging* **20**(1) (2001) 45–57
7. Ashburner, J., Friston, K.J.: Unified segmentation. *NeuroImage* **26**(3) (July 2005) 839 – 851
8. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* **39**(1) (1977) pp. 1–38
9. Sled, J.G., Zijdenbos, A.P., Evans, A.C.: A nonparametric method for automatic correction of intensity nonuniformity in MRI data. *IEEE Transactions on Medical Imaging* **17**(1) (February 1998) 87 – 97
10. Minka, T.P.: Expectation-maximization as lower bound maximization (1998)
11. Boyes, R.G., Gunter, J.L., Frost, C., Janke, A.L., Yeatman, T., Hill, D.L., Bernstein, M.A., Thompson, P.M., Weiner, M.W., Schuff, N., Alexander, G.E., Killiany, R.J., DeCarli, C., Jack, C.R., Fox, N.C.: Intensity non-uniformity correction using N3 on 3-T scanners with multichannel phased array coils. *NeuroImage* **39**(4) (February 2008) 1752 – 1762
12. Likar, B., Viergever, M.A., Pernus, F.: Retrospective correction of MR intensity inhomogeneity by information minimization. *IEEE Transactions on Medical Imaging* **20**(12) (Dec 2001) 1398–1410
13. Zheng, W., Chee, M.W., Zagorodnov, V.: Improvement of brain segmentation accuracy by optimizing non-uniformity correction using N3. *NeuroImage* **48**(1) (2009) 73 – 83
14. Tustison, N., Avants, B., Cook, P., Zheng, Y., Egan, A., Yushkevich, P., Gee, J.: N4ITK: Improved N3 bias correction. *IEEE Transactions on Medical Imaging* **29**(6) (2010) 1310–1320