Blind Retrospective Motion Correction of MR Images

Alexander Loktyushin,[1]* Hannes Nickisch,[2] Rolf Pohmann,[3] and Bernhard Schölkopf[1]

Purpose: Subject motion can severely degrade MR images. A retrospective motion correction algorithm, Gradient-based motion correction, which significantly reduces ghosting and blurring artifacts due to subject motion was proposed. The technique uses the raw data of standard imaging sequences; no sequence modifications or additional equipment such as tracking devices are required. Rigid motion is assumed.

Methods: The approach iteratively searches for the motion trajectory yielding the sharpest image as measured by the entropy of spatial gradients. The vast space of motion parameters is efficiently explored by gradient-based optimization with a convergence guarantee.

Results: The method has been evaluated on both synthetic and real data in two and three dimentions using standard imaging techniques. MR images are consistently improved over different kinds of motion trajectories. Using a graphics processing unit implementation, computation times are in the order of a few minutes for a full three-dimentional volume.

Conclusion: The presented technique can be an alternative or a complement to prospective motion correction methods and is able to improve images with strong motion artifacts from standard imaging sequences without requiring additional data. **Magn Reson Med 70:1608–1618, 2013. © 2013 Wiley Periodicals, Inc.**

Key words: retrospective motion correction; autofocusing; gradient-based optimization

INTRODUCTION

Subject motion during MRI scans can cause severe degradations of the image quality (1). Motion artifacts—usually appearing as ghosts and blur—can easily render an image unacceptable for medical analysis. Long scans used to acquire high-resolution images are particularly vulnerable. Many patients having difficulties in voluntarily constraining their motion could highly benefit from MR diagnostics made possible by motion correction.

Although there is a vast literature on motion correction, no universal solution to the problem yet exists. The available methods can be broadly classified into *prospective* and *retrospective* techniques. Prospective methods correct for motion during the scan by constantly adjusting the magnetic field gradients to follow the subject's pose as measured by external sensors, like tracking cameras (2,3). Retrospective methods remove motion artifacts after the image is fully acquired. This can be done by estimating the point spread function due to translation, and then using its phase for correction (4,5), by correlating the values in adjacent k-space lines to estimate the motion (6), by the method of generalized projections (7) in space-limited images with a known finite region of support, by special trajectories introducing redundancy into the acquired data that allows inferring the motion (8), or by navigator-based methods that measure the motion (9,10) using additional echoes in the pulse sequence.

Autofocusing (AF) methods represent a large and important class of retrospective techniques. AF does not require the use of special trajectories, or any other additional data, and relies on optimization-based refining of the image with respect to some quality measure. It was shown to be able to compete with navigator-based methods in terms of image quality (11). The origin of AF methods can be traced back to early attempts for solving image denoising and deblurring problems (12,13). AF is used for motion compensation in radar measurements (14), where the acquisition, like in MR, takes place in the frequency domain. The seminal paper of Atkinson (15) adapts AF to MR motion correction. In the last years, AF methods for MR were improved substantially to accommodate complex motions involving rotations (16,17), and to deal with three-dimentional (3D) acquisitions (18). Recently, retrospective approaches to correction of a non rigid body motion were proposed (19–21), however, to cope with the large number of unknown motion parameters, reference data (i.e., from navigator-based trackers) is needed.

We propose an AF method exploring the space of possible motions, and selecting the point in that space such that the motion corrected image yields the minimum value of a cost function. For high-resolution 3D volumes, the motion space is vast since we have six free parameters per phase encode step. Starting from an initial guess for the motion trajectory, e.g., no-motion or some small random initialization, our algorithm Gradient-based Motion Correction (GradMC) iteratively corrects for translation and rotation until no more progress in terms of image quality can be made. To objectively measure the image quality, we use the gradient entropy metric as cost function.

The major difference of GradMC to other AF approaches is the way the motion space is explored. To our knowledge, currently existing methods either perform greedy forward selection or rely on brute force search in motion space, which in high-resolution 3D scenarios fails due to the curse of dimensionality. Greedy forward selection (as in e.g. Ref. 15) locally optimizes a quality metric by altering only a few motion parameters at a time. We use the analytical gradient (see Appendix) to drive the optimization process, and thus are able to explore the parameter space consistently using quasi-Newton Broyden–Fletcher–Goldfarb–Shanno and obtain provable convergence to a local optimum, which is a desirable property of an optimization algorithm.

[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany.

[2]Philips Research Europe, Hamburg, Germany.

[3]Max Planck Institute for Biological Cybernetics, Tübingen, Germany.

*Correspondence to: Alexander Loktyushin, M.Sc., Max Planck Institute for Intelligent Systems, Heisenbergstr. 3, 70569 Stuttgart, Germany. E-mail: aloktyus@tuebingen.mpg.de

Received 28 August 2012; revised 30 November 2012; accepted 2 December 2012.

DOI 10.1002/mrm.24615

Published online 11 February 2013 in Wiley Online Library (wileyonlinelibrary.com).

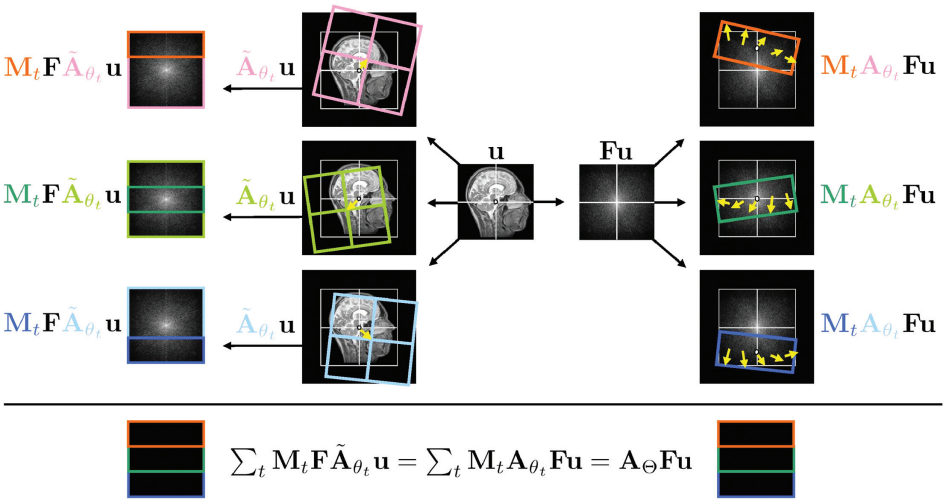

FIG. 1. Illustration of the forward model. An unknown underlying image $\mathbf{u}$ experiences rotation and translation during every phase encode step represented by a masking matrix $\mathbf{M}_t$. Motion can be represented in the pixel domain (left) and in the Fourier domain (right). The latter allows for efficient computations.

Interestingly, gradients can be computed quite cheaply by one single non uniform fast Fourier transformation per degree of freedom (DoF). We conjecture that derivative-based optimization makes it possible to achieve higher image quality due to efficient exploration of the high-dimensional objective landscape.

Similar to other approaches, we assume that the motion time-scale is longer than the repetition time TR, i.e., we model the motion trajectory as a piecewise constant function. We neglect second-order effects such as the influence of motion on the magnetic field. Like all other reference-free (no external information on motion) methods, we assume that the imaged object behaves as a rigid body. Under these assumptions, we can correct for arbitrary motion trajectories in six degrees of freedom for both two-dimentional (2D) and 3D acquisitions. We provide a Matlab implementation of GradMC along with four examples at: http://mloss.org/software/view/430/.

METHODS

In this section, we first explain how we model the motion-corrupted k-space data. Next, we construct the cost function based on the image gradient entropy metric. We then describe how translational and rotational motion is implemented, and deal with non linearities and local minima in our objective using multiscale optimization. Finally, we demonstrate that our method can be extended to multiple coils.

Model

MR scanners acquire Fourier coefficients along smooth trajectories as dictated by the gradient shapes of the MRI sequence. We restrict ourselves to the Cartesian case where k-space is densely sampled line by line. In the following, we describe a forward model of the measurement process that includes a moving subject. For an illustration, see Figure 1.

Let $\mathbf{F} \in \mathbb{C}^{N\times N}$ be an orthonormal Fourier matrix and $\mathbf{u} \in \mathbb{C}^N$ an unknown sharp image of size $N = n_x \cdot n_y \cdot n_z$ pixels. $\mathbf{M} = \mathrm{diag}(\mathbf{m}) \in [0,1]^{N\times N}$ is a diagonal masking matrix where $\mathbf{m} \in [0,1]^N$ are masking coefficients, and $\tilde{\mathbf{A}}_{\boldsymbol{\theta}_t} \in \mathbb{R}^{N\times N}$ denotes a rigid motion transformation matrix parameterized by a vector of translations and rotations $\boldsymbol{\theta}_t \in \mathbb{R}^3 \times [0, 2\pi)^3$ at time t.

Assuming additive measurement noise $\boldsymbol{\varepsilon}$, the acquisition in k-space can be written as

$$\mathbf{y} = \int_0^T \mathbf{M}_t\mathbf{F}\tilde{\mathbf{A}}_{\boldsymbol{\theta}_t}\mathrm{d}t\,\mathbf{u} + \boldsymbol{\varepsilon} \in \mathbb{C}^N, \int_0^T \mathbf{m}_t\mathrm{d}t = \mathbf{1}, \qquad [1]$$

with $\mathbf{M}_t$ selecting the part of k-space being filled by a scanner at time t.

At all times t, our subject is rigidly moving, i.e., Eq. [1] acts on the transformed image $\tilde{\mathbf{A}}_{\boldsymbol{\theta}_t}\mathbf{u}$ instead of $\mathbf{u}$. As measurements are acquired in k-space, it is more convenient to represent $\tilde{\mathbf{A}}_{\boldsymbol{\theta}_t}$ directly in k-space. Mathematically speaking, $\tilde{\mathbf{A}}_{\boldsymbol{\theta}_t}$ and $\mathbf{F}$ do not commute. However, there is an operator $\mathbf{A}_{\boldsymbol{\theta}_t}$ such that $\mathbf{F}\tilde{\mathbf{A}}_{\boldsymbol{\theta}_t} = \mathbf{A}_{\boldsymbol{\theta}_t}\mathbf{F}$, because rotation and translation have counterparts in k-space (22). Rotation (around the center) in pixel space corresponds to rotation (around the DC component) in k-space and translation in pixel space corresponds to pointwise multiplication of k-space data with linear phase ramps. This simplifies our measurement Eq. (1) because $\mathbf{F}$ can be moved outside the integral

$$\mathbf{y} = \int_0^T \mathrm{diag}(\mathbf{m}_t)\mathbf{A}_{\boldsymbol{\theta}_t}\mathrm{d}t\,\mathbf{Fu} + \boldsymbol{\varepsilon} \in \mathbb{C}^N, \int_0^T \mathbf{m}_t\mathrm{d}t = \mathbf{1}. \qquad [2]$$

In a Cartesian MRI measurement, there are two characteristic time intervals, the echo time TE and the repetition time TR. Our line-by-line setting is best described by a sequence of T repetitions of length TR each. In the following, we assume that the subject motion is piecewise constant, which means that the rigid transform matrix $\mathbf{A}_{\boldsymbol{\theta}_t}$ changes very little during one repetition. Thus, the

measurement integral (2) can be approximated by the sum and the masking becomes binary:

$$\mathbf{y} = \sum_{t=1}^{T} \mathrm{diag}(\mathbf{m}_t)\mathbf{A}_{\boldsymbol{\theta}_t}\mathbf{F}\mathbf{u} + \boldsymbol{\varepsilon} \in \mathbb{C}^N, \sum_{t=1}^{T} \mathbf{m}_t = \mathbf{1},\ \mathbf{m}_t \in \{0,1\}^N. \quad [3]$$

Assume for now that k-space lines for a volume of size n_x by n_y by n_z pixels are measured, hence $T = n_y \cdot n_z$ repetitions are needed. In every repetition, a noisy version of $\mathrm{diag}(\mathbf{m}_t)\mathbf{A}_{\boldsymbol{\theta}_t}\mathbf{F}\mathbf{u} \in \mathbb{C}^N$ is acquired. The binary mask $\mathbf{m}_t$ contains n_x nonzero components $[\mathbf{A}_{\boldsymbol{\theta}_t}]_{\mathbf{m}_t}\mathbf{F}\mathbf{u} \in \mathbb{C}^{n_x}$, where the shortcut $[\mathbf{A}_{\boldsymbol{\theta}_t}]_{\mathbf{m}_t} \in \mathbb{C}^{n_y \times N}$ represents the nonzero submatrix of $\mathrm{diag}(\mathbf{m}_t)\mathbf{A}_{\boldsymbol{\theta}_t} \in \mathbb{R}^{N\times N}$. As a result, we can pack the whole motion into the single matrix $\mathbf{A}_{\boldsymbol{\theta}} = \sum_{t=1}^{T} \mathrm{diag}(\mathbf{m}_t)\mathbf{A}_{\boldsymbol{\theta}_t}$ composed of blocks of size n_x each

$$\mathbf{y} = \mathbf{A}_{\boldsymbol{\theta}}\mathbf{F}\mathbf{u} + \boldsymbol{\varepsilon} \in \mathbb{C}^N,\ \mathbf{A}_{\boldsymbol{\theta}} := \begin{bmatrix} [\mathbf{A}_{\boldsymbol{\theta}_1}]_{\mathbf{m}_1} \\ [\mathbf{A}_{\boldsymbol{\theta}_2}]_{\mathbf{m}_2} \\ \vdots \\ [\mathbf{A}_{\boldsymbol{\theta}_T}]_{\mathbf{m}_T} \end{bmatrix} \in \mathbb{C}^{N\times N}, \quad [4]$$

where $\mathbf{A}_{\boldsymbol{\theta}}$ is parameterized by $\boldsymbol{\theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_T] \in \Theta = \mathbb{R}^{3T} \times [0, 2\pi)^{3T}$—the vector with all the motion parameters of the trajectory, and $T = n_y \cdot n_z$. Computationwise, a matrix vector multiplication with $\mathbf{A}_{\boldsymbol{\theta}}$ requires $\mathcal{O}(N)$ because $\mathbf{A}_{\boldsymbol{\theta}}$ can be decomposed into blocks, the matrix vector multiplications of which can be performed efficiently. Some additional details are provided in Appendix.

Cost Function

The key ingredient of any AF method is an image quality metric. Ideally, it is large for motion-corrupted images and small for motion-free ones. The second desirable property is that minima over such a metric correspond to diagnostically useful images according to medical experts. We are not aware of a theoretically-justified optimal metric fulfilling these requirements. As a surrogate, we use the gradient entropy

$$\phi(\mathbf{u}) = \mathcal{H}(\mathbf{D}_x\mathbf{u}) + \mathcal{H}(\mathbf{D}_y\mathbf{u}), \quad [5]$$

that was found to be superior to 24 estimators in an empirical study based on the expertise of radiologists (23). Here, $\mathbf{D}_x,\ \mathbf{D}_y \in \{0, \pm 1\}^{N\times N}$ are horizontal and vertical finite difference matrices, and $\mathcal{H}(\cdot)$ is a pixel entropy defined as:

$$\mathcal{H}(\mathbf{u}) = -\mathbf{v}^\top \ln \mathbf{v},\ \mathbf{v} = \sqrt{\frac{\mathbf{u} \odot \bar{\mathbf{u}}}{\mathbf{u}^{\mathsf{H}}\mathbf{u}}} \in \mathbb{R}_+^N,\ \mathbf{u} \in \mathbb{C}^N. \quad [6]$$

Further, $\bar{\mathbf{u}}$ denotes the complex conjugate and $\mathbf{a} \odot \mathbf{b}$ denotes the pointwise (Hadamard) product of two vectors $\mathbf{a}$ and $\mathbf{b}$. Note that the entropy is scale invariant $\mathcal{H}(\mathbf{u}) = \mathcal{H}(\alpha\mathbf{u})$, phase invariant $\mathcal{H}(\mathbf{u}) = \mathcal{H}(\exp(i\boldsymbol{\omega}) \odot \mathbf{u})$, and invariant with respect to a constant offset $\mathcal{H}(\mathbf{u}) = \mathcal{H}(\mathbf{u} + c\mathbf{1})$. The effect of such a focus criterion is to prefer images having uniform intensity regions separated by sharp edges. Indeed, motion corrupted images are known to be affected by blurring and ghosts, both leading to smooth intensity variations, and thus high-gradient entropy $\mathcal{H}$.

Our goal is to estimate the motion parameters $\hat{\boldsymbol{\theta}}$ which best describe the motion in the corrupted volume, and then to correct for motion. We consider a class of approximately invertible motions composed of translations and small angle rotations (<3°). For strong rotations, the information in high-frequency k-space regions is lost, and no artifact-free empirical inversion is possible. By empirical inversion, we mean the application of $\mathbf{A}_{\boldsymbol{\theta}}$ directly to a motion corrupted observation $\mathbf{y}$

$$\mathbf{u}_{\boldsymbol{\theta}} = \mathbf{F}^{\mathsf{H}}\mathbf{A}_{\boldsymbol{\theta}}\mathbf{y}, \quad [7]$$

where $\mathbf{u}_{\boldsymbol{\theta}}$ is the resulting image in the spatial domain. We are interested in motion parameters $\hat{\boldsymbol{\theta}}$ that correspond to the sharpest image according to the minimum of our cost function

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \phi\left(\mathbf{u}_{\boldsymbol{\theta}}\right). \quad [8]$$

It is clear that the estimation of $6 \cdot T$ free motion parameters from a single image constitutes a highly underdetermined problem. Therefore, we regularize the trajectory of the recovered motion parameters by putting a quadratic penalty on the differences of consecutive motion parameters

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \phi(\mathbf{F}^{\mathsf{H}}A_{\boldsymbol{\theta}}\mathbf{y}) + \lambda \left\|\mathbf{D}\boldsymbol{\theta}\right\|^2, \quad [9]$$

where $\mathbf{D} \in \{0, \pm 1\}^{6\cdot T \times 6\cdot T}$ is a finite difference matrix (respecting the temporal order of the phase encodes), and $\lambda \in \mathbb{R}_+$ is a regularization parameter controlling the smoothness of the trajectory. In practice, the value $\lambda = 0.1$ was found to give best results. In order for a regularization to make sense, we assume that the order in which k-space lines are recorded by the scanner is known (in 3D there is a phase/slice direction ambiguity with this respect). Otherwise, the temporal structure of the motion trajectory recovered by the algorithm might not match the actual temporal sequence of acquisition in k-space. As an alternative to the difference penalty term, a blocking approach can be used, where the motion parameters in consecutive views are hard-constrained to be optimized together. This makes it possible to obtain reasonable and probable motion estimates for those regions in k-space, where the accuracy of the algorithm is limited. While it may be worthwhile to take advantage of this complexity reduction approach to further improve the motion correction results, the spirit of our current research is slightly different. We reduce complexity by means of an adjustable regulariser instead, while keeping the number of parameters fixed. This leaves more freedom to the optimizer to find the best setting among the softly coupled parameters.

Translation and Rotation Correction

The practical implementation of the $\mathbf{A}_{\boldsymbol{\theta}}$ operator involves the treatment of translational and rotational motion represented by $6 \cdot T$ components in $\boldsymbol{\theta}$. The translation correction amounts to a multiplication of each k-space line with a linear phase ramp $\exp(-2\pi i k_x \theta_t)$, a function of translation θ_t in the spatial domain at time t, and the Fourier coefficient k_x of the affected view.

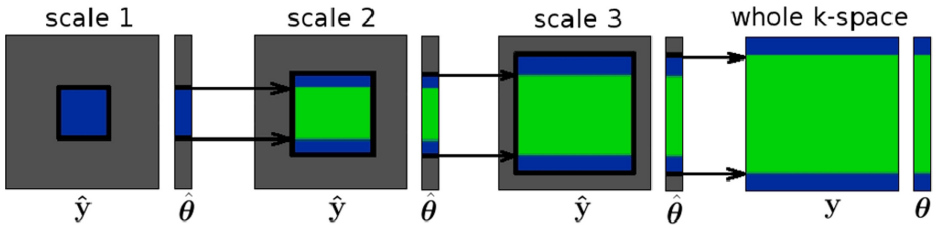

FIG. 2. Multiscale optimization of motion parameters $\boldsymbol{\theta}$ is performed by progressively growing the processed data $\hat{\boldsymbol{y}}$ from the low-frequency center until the whole k-space is covered. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

Rotation is more involved since it causes a rotation of one section of k-space relative to others. To compute its effect on the image, we first construct a deformed grid by rotating the points of each k-space line by their time-respective angles. Then, we do the interpolation in an oversampled (2X) k-space to estimate the values on the points of the rotated grid.

Multiscale Initialization

Our cost function is highly nonlinear since it involves logarithms, complex exponentials, and piecewise cubic interpolation polynomials. Minimizing such a function is difficult, due to the presence of local minima. We address this problem by using a multi scale coarse-to-fine approach—see Algorithm 1 and Figure 2 for illustration. It is based on the observation that in DC-centered k-space cubes containing low frequencies only, even strong motion will produce little offsets in the spatial domain. Furthermore, the signal intensity, and thus signal-to-noise ratio is usually higher in low-frequency regions. This means that estimation and correction of motion is easier in coarsely sampled versions of the image. Having estimated the motion parameters for the coarser scales, we proceed to the finer ones, for which we can use the motion parameters from coarser scales as an initialization. This way, we drive the optimizer towards better minima in the objective landscape.

There is a fundamental difference between 2D and 3D multiscale optimization. In 2D, a temporal sequence of acquired views in the spectrum most often goes in high-low-DC-low-high frequency order. In 3D, a sequence is more complex and involves multiple alternations of high- and low-frequency views (consider Cartesian line-by-line covering of the 3D k-space cube). In the first scale iteration, GradMC finds only the lowest frequency segments of the motion trajectory. These segments are surrounded by gaps corresponding to yet unknown motion parameters of higher frequency views. In each scale iteration the gaps shrink, until finally the whole trajectory is recovered. We exploit the alternation effect to our benefit—we initialize the motion parameters in the gaps by linearly interpolating from the values on their boundaries, where the algorithm has already determined the motion. This allows achieving higher accuracy in determining the motion in more problematic high-frequency views.

Multiple Coils

The approach can be extended to images acquired with multiple coils. Since we optimize with respect to motion parameters, there is no need to know the sensitivity profiles of each coil. We seek the motion parameters $\hat{\boldsymbol{\theta}}$ that invert the motion on the data from each coil, such that the sum of the cost functions over all C coils is minimal

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\Theta}{\arg\min} \sum_{c=1}^{C} \phi\left(\mathbf{F}^{\mathsf{H}} A_{\boldsymbol{\theta}} \mathbf{y}_c\right) + \lambda \left\|\mathbf{D}\boldsymbol{\theta}\right\|^2, \qquad [10]$$

where $\mathbf{y}_c$ is the raw data from coil c.

Algorithm 1 Multiscale optimization

Input: Corrupted volume $\mathbf{y}$ with $N = n_x \cdot n_y \cdot n_z$ k-space coefficients centered at $[c_x, c_y, c_z] = \left[\frac{n_x}{2}+1, \frac{n_y}{2}+1, \frac{n_z}{2}+1\right]$. Also, assume $n_y = n_z$.
Output: Restored volume $\mathbf{u}$ in spatial domain.
For $s \leftarrow \frac{64}{2}, ..., \frac{n_y}{2}$ **do**

- Use only k-space center of the raw data: $\hat{\mathbf{y}} \leftarrow \mathbf{y}(c_x,\ c_y{-}s\ ..\ c_y{+}s,\ c_z{-}s\ ..\ c_z{+}s)$.
- Compute best motion parameters: $\hat{\boldsymbol{\theta}} \leftarrow \arg\min_{\boldsymbol{\theta}}\ \phi(\mathbf{F}^{\mathsf{H}}\mathbf{A}_{\boldsymbol{\theta}}\hat{\mathbf{y}})$.
- Initialize central frequency part of $\hat{\boldsymbol{\theta}}$ on next finer scale: $\hat{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}(c_y{-}s\ ..\ c_y{+}s,\ c_z{-}s\ ..\ c_z{+}s)$.

End
Finally, obtain the sharp image: $\hat{\mathbf{u}} \leftarrow \mathbf{F}^{\mathsf{H}}\mathbf{A}_{\hat{\boldsymbol{\theta}}}\mathbf{y}$.
See Figure 2 for an illustration.

Data Acquisition

Imaging experiments were performed on a 3T scanner (Siemens Healthcare, Erlangen, Germany), using a single-channel birdcage coil for brain imaging. To obtain motion-free images and to inflict well-controlled motion and exclude non rigid motion, a fixed monkey brain embedded in agarose was used for initial experiments. It was placed on a special, MR-compatible holder that allowed well-defined motion in 3D (two translational, one rotational DoF) during the scan.

In vivo human images were acquired on four subjects as approved by the local ethics board. The subjects, whose heads were loosely fixed inside the coil, were told to move during some of the experiments. Standard imaging protocols in 2D and 3D were used, with sequences and parameters similar to those used in medical routine and neuroscience research. For 2D images, a multi slice RARE sequence was used. Acquiring eight echoes per echo train, four 4 mm slices were imaged with a TE of 40 ms and a TR of 2.5 s within 75 s. Including the scanner's automatic two-fold oversampling in read direction, a FOV of 400×160 mm^2 was covered with a matrix size of 384×160.

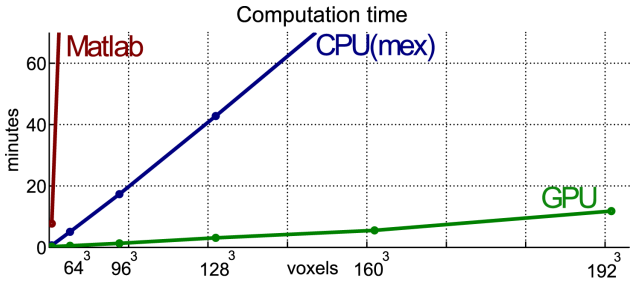

FIG. 3. Comparison of the computation times for motion correction of 3D volumes of different sizes N for implementations in Matlab, plain C with MEX interface, and CUDA (GPU). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

In addition, high-resolution data were obtained with a matrix size of 1024×666 over a FOV of 440×286 mm^2. Those scans acquired 18 echoes per echo train with a TE of 61 ms, a TR of 4 s and a refocusing flip angle of 120° within 2.5 min. 3D in vivo data had a matrix size of $384 \times 192 \times 96$ over a FOV of $440 \times 220 \times 115$ mm^3. FLASH images with a TE of 5.3 ms, a TR of 16 ms and a flip angle of 18° were acquired within 5 min. For the monkey brain images, the FOV was reduced to $100 \times 100 \times 24$mm^3 and only 16 slices were sampled with a flip angle of 35°. MP-RAGE scans had a similar duration (4.8 min) and used an inversion time of 1.1 s, a TE of 2.85 ms, a TR of 1.5 s and a flip angle of 9°.

Optimization Algorithm

We use the limited memory Brayden–Fletcher–Goldfarb–Shanne nonlinear optimizer (24) with 50 iterations per scale. Note that the gradient of the cost function can be computed similarly fast as the objective function itself (see Appendix for details). The computational bottleneck are the fast Fourier transformations requiring an effort of $\mathcal{O}(N \cdot \log N)$ each. Figure 3 shows the computation times for various implementations against the total number of voxels in a 3D volume. Our pure Matlab implementation[1] is clearly impractical for large volumes, but has an advantage of clear structure with the entire pipeline represented by dedicated matrix classes (25) handling all major operations of the forward model. The first order speed-up was achieved by implementing the heavy resampling and gridding routines in plain C via Matlab's MEX interface. Finally, we took advantage of modern graphic cards [graphical processing unit, (GPU)], benefiting from the fact that most heavy operations in our pipeline are massively parallel. GPU implementation was done in CUDA language and is also interfaced via MEX. For realistic volumes with 128^3 voxels, 3 min of processing are required. The current bottleneck of our GPU implementation is the need to transfer the image from the main memory to the device memory of the GPU for each function evaluation. Thus, merging our GPU code with the code of the optimizer will make it possible to achieve a further speed-up. Experiments were done on an Intel(R) Core(TM)2 Duo CPU 2.66 Ghz processor and a GeForce GTX 285 graphics card.

[1]Available from http://mloss.org/software/view/430/.

RESULTS

In the following, we benchmark GradMC algorithm on increasingly challenging motion correction tasks. Starting with 2D and 3D simulated data, we then proceed to 2D and 3D real world data.

Simulations

Starting with a motion-free 2D image, we use our forward motion model to create the corrupted data. This allows us to compare the motion trajectory recovered by our reconstruction algorithm against the ground truth. Both rotational and translational motion was simulated. The motion trajectory for each motion parameter was generated to be of a sinusoidal form (see Fig. 4). The sine-form motion was chosen for convenient analysis of the recovered motion trajectories against the ground truth in all view/frequency ranges.

In Figure 4, we show the results of the analysis of recovered trajectories. For a 2D image of the brain (matrix size $N = 192 \times 192$), we induced an out-of-phase sine-form motion simultaneously in all motion DoF (phase/frequency encode direction translations, and in-plane rotation). In 100 successive trials, the optimizer was challenged with different random initializations, which were in the range of the amplitude of the sinusoid. Plotted in solid are the ground truth sinusoidal trajectories. Dashed lines show the empirical mean of the recovered trajectories. The shaded gray tube around the empirical mean contains 95% of the probability mass computed by the respective quantiles. Analyzing the discrepancy of recovered mean trajectories to ground truth and the width of the error tubes, we observe that the accuracy is very high for translations in low- and mid-frequency ranges (DC component is the center of abscissa), whereas moderate discrepancies occur in high-frequencies, likely due to low power of the signal in these regions of the spectrum. A large discrepancy also appears in the estimation of translational motion in phase direction around the DC component that is consistent over all trials, thus having a systematic origin. However, this discrepancy was not observed to deteriorate the restored image, probably because close-to-DC regions are not that sensitive to motion, with DC itself being completely invariant to it. Doing the restoration with motion parameters $\boldsymbol{\theta}$ initialized to zeros, the unitless image metric values computed on corrupted/restored/ground-truth images were equal to 1318/1114/1032, respectively.

In Figure 5, the same analysis is carried out in 3D (matrix size $N = 192 \times 192 \times 48$). Since low- and high-frequency views are interleaved in 3D, the recovered trajectories have periodic large variance regions. These regions correspond to high-frequency views, and the intervals between their centers are equal to the matrix dimension in phase encode direction, so there are a total of 48 of such regions for the given matrix size. One beneficial consequence of this is that if the true motion trajectory has smooth variations, it is possible to actually constrain the variations in high-variance regions from the neighbouring low-frequency supports.

Measured Data

First, we show results of applying our method to 2D images acquired by a RARE sequence. The motion corrupted and

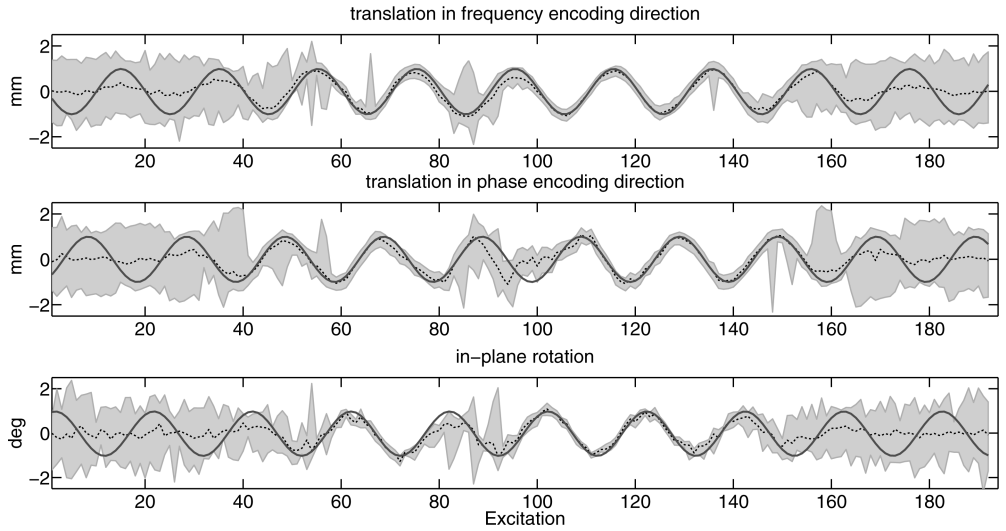

FIG. 4. Recovery of underlying ground truth motion parameters in a 2D forward simulation with three DoFs starting from a random initialization. **Solid line:** underlying sinusoidal motion trajectory. **Dotted line:** the empirical mean over 100 recovered trajectories. **Shaded gray:** error tube over 100 recovered trajectories capturing 95% of the probability mass (corresponds to $\pm 2\sigma$ for Gaussian).

restored images (matrix size $N = 384 \times 160$) are shown in Figure 6. The subject made sideways head movements with rotation around the support point, where the head touched the table. The echo train length was equal to eight, which in principle could allow us to reduce the number of motion parameters to be estimated by packing them into the train blocks. We did not exploit this advantage, and let the algorithm determine the motion parameters for

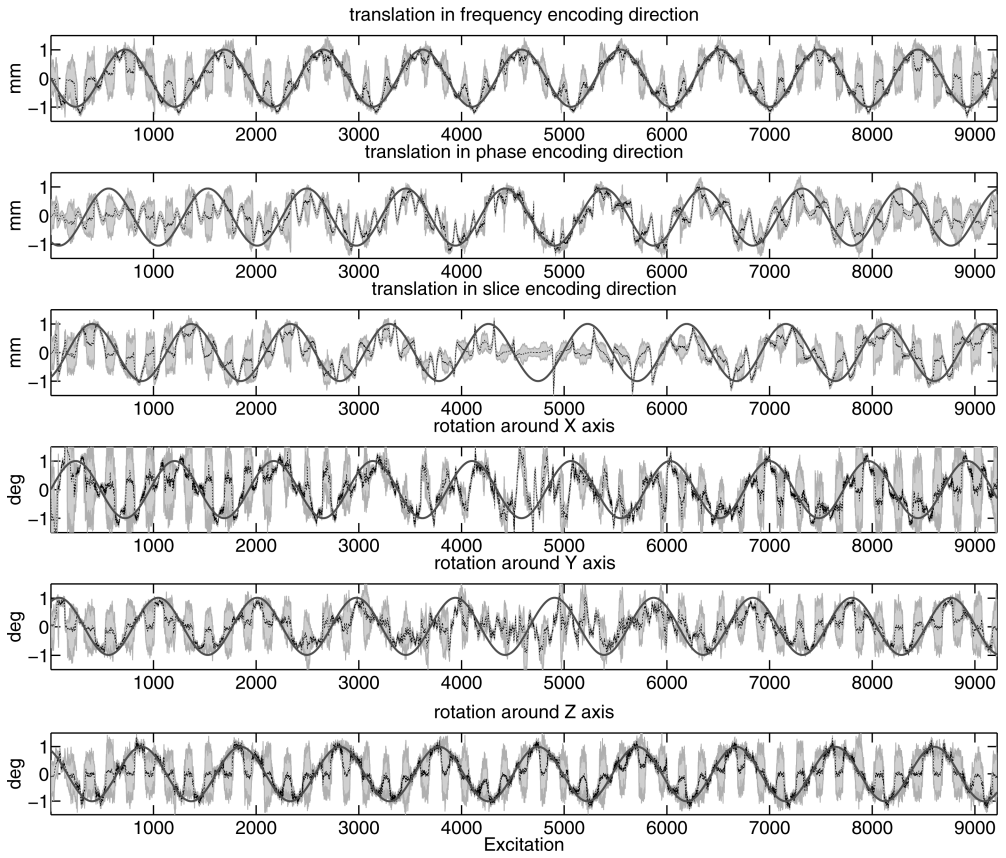

FIG. 5. Recovery of underlying ground truth motion parameters in a 3D forward simulation with six DoFs starting from random initializations. **Solid line:** underlying sinusoidal motion trajectory. **Dotted line:** the empirical mean over 100 recovered trajectories. **Shaded gray:** error tube over 100 recovered trajectories capturing 95% of the probability mass (corresponds to $\pm 2\sigma$ for Gaussian).

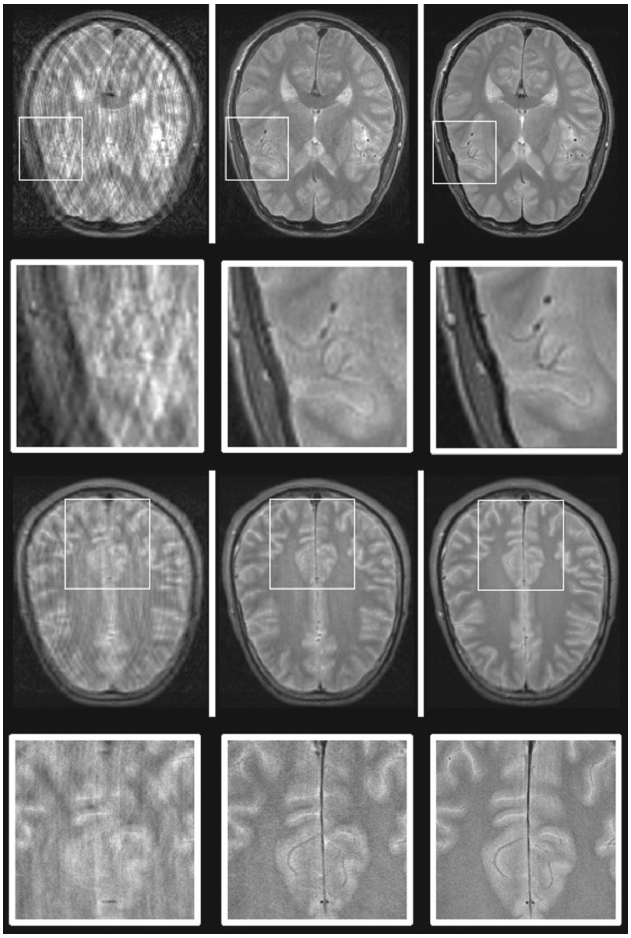

FIG. 6. Motion correction of 2D RARE images. Freely moving human subject. **Left:** motion corrupted image. **Middle:** reconstruction. **Right:** no motion image. **Top:** low resolution. **Bottom:** high resolution (matrix size $N = 1024 \times 640$).

each view. The reconstructed result is of high quality compared to the observed image, where the anatomical details are almost fully occluded by ghosting artifacts. Additionally, the bottom part of Figure 6 shows the reconstruction of high resolution (matrix size $N = 1024 \times 640$) RARE image, for which the method is also capable of improving the image quality.

We did additional experiments in 2D to test the multicoil procedure as described in the methods section. The results are shown in Figure 7, which compares the reconstruction using the data from each coil separately (middle) to simultaneous treatment (right). The reconstruction quality is slightly better if the coil data is motion corrected jointly.

Next, we present the reconstruction results on more challenging 3D volumes. To completely avoid the effects of non rigid motion, we first imaged the monkey brain. In Figure 8, we show the reconstruction results for 3D volume (matrix size $N = 384 \times 192 \times 16$) acquired with FLASH sequence. From left to right two-degraded/restored/nomotion slices are shown. GradMC was able to find an almost artifact-free solution.

Finally, we acquired 3D motion-affected volumes of the brain of a freely moving human subject. The top rows of Figure 9 show the results from the FLASH sequence (matrix size $N = 384 \times 192 \times 96$), whereas the bottom rows display images acquired with an MPRAGE sequence (matrix size $N = 304 \times 192 \times 94$).

DISCUSSION

In recent years, most successful techniques have been based on prospective motion correction (see Refs. (26–28) for a comprehensive review). The obvious and very strong advantage of our method (and retrospective methods in general) is that it can be applied to any already acquired dataset (subject to our model assumptions), and does not require the use of any tracking equipment, or special imaging sequences. By applying additional constraints, it may also be extendable to handle non-rigid body motion, for which prospective correction is highly challenging. Conversely, large rotations and motion of the object out of the FOV lead to irreversible loss of information, which is problematic for retrospective methods, while prospective ones are immune to such problems.

We have shown that gradient-based optimization is an efficient way to uncover motion parameters from motion-corrupted MRI scans and to reconstruct undistorted images, even though the optimization is challenging due to nonlinearity, high dimensionality, and heavy computational demand. In previous AF approaches, motion parameters were obtained by application of trial translational/rotational displacements throughout the views in k-space and calculation of the image quality metric to find the best displacement for each view. Using complicated heuristics to select and group views to be optimized, it is possible to make the procedure quite fast, but at the cost of complexity, accuracy and robustness of the algorithm. Dealing with high-resolution 3D volumes such methods run into all sorts of trouble due to the curse of dimensionality, leading to an enormous combinatorial search.

When starting to develop our method, we faced two major challenges: the first one was that our highly nonconvex image quality metric contains multiple local minima. We addressed this problem by using a multiscale algorithm to initialize the motion parameters on each scale iteration to the vicinity of supposedly good local minima found from the previous coarser scale. We exploited that it is much easier to accurately determine the motion parameters of low-frequency regions in k-space because of higher signal power and less sensitivity to strong motion. The second problem was computation time—an important limitation for many retrospective motion correction methods. We developed a GPU version of GradMC based on a highly parallel implementation of our forward model. For volumes of realistic sizes, only a few minutes are needed for the reconstruction. There is still room for improvement with respect to computation time, as in the current implementation it is necessary to move the data to the memory of the GPU and back for each function evaluation. We have used cubic interpolation to do the gridding, which might seem inferior to recent efficient gridding methods based on Kaiser–Bessel convolution kernels or jincs (29,30). The reason we are using cubic interpolation is that it can be implemented simply and efficiently on GPUs. Better

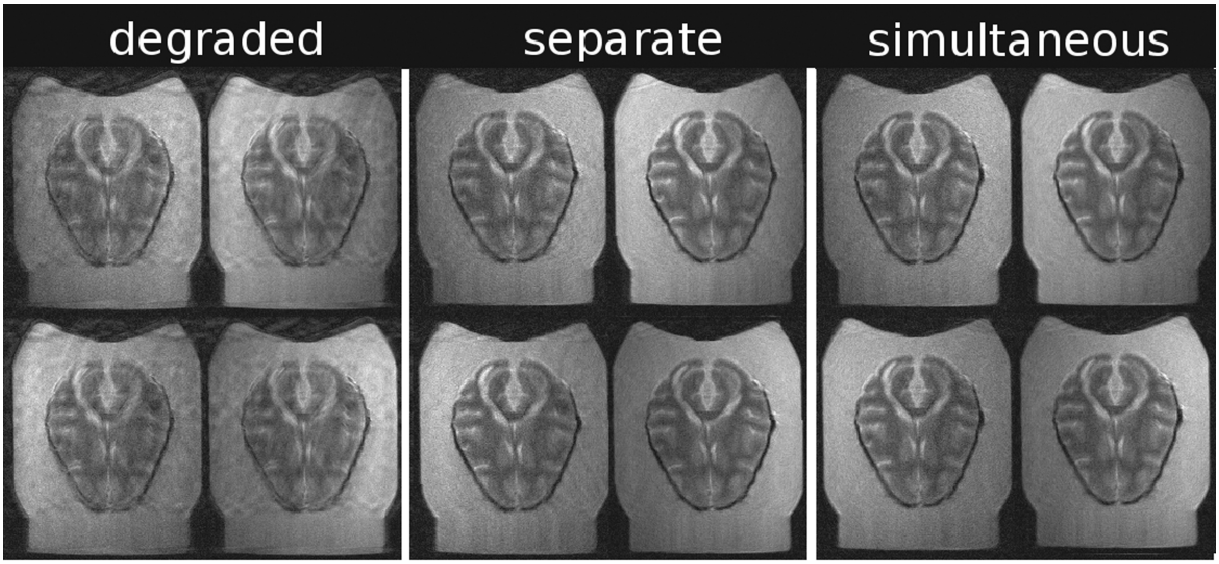

FIG. 7. Motion correction of data recorded by four coils. Monkey brain in fixation gel. **Left:** motion corrupted data. **Middle:** reconstruction using the data from each coil separately. **Right:** reconstruction with simultaneous optimization of image metric over the data from all coils.

interpolation kernels would need specialized mathematical functions like Bessel functions either not available on GPUs or quite slow to compute.

To test the stability of the reconstruction algorithm, we used our forward model to generate motion corrupted data. The ground truth motion trajectory had a sinusoidal form, which conveniently allows controlling the strength of motion (amplitude) and its time variation (frequency). Doing a motion correction on this simulated data, we challenged the optimizer with random initializations of motion parameters. With respect to image quality, there was always a substantial improvement for all initializations. Comparing recovered trajectories to ground truth, we observed that the mismatch in motion parameters was small for low- and mid-frequency views, and increasingly large towards high-frequency views. We expected to see such a pattern which is due to an inverse power law drop of the signal strength from low- to high-frequencies. Additionally, the simulation experiment showed that multiple differing motion trajectories, when used to invert the motion, lead to similarly good reconstruction with respect to both image metric and visual quality. For 3D volumes, we also observed "cross-talk" effects between different DoFs, most strongly distinct between translational DoF in phase/slice encoding directions. This effect is most strongly observed if the data is simulated with motion in some DoF only. On reconstruction of such data, recovered motion trajectories show non zero motion in "silent" DoF. Thus, artifacts produced by motion in one DoF can—up to a certain extent—be mimicked by motion in another DoF. The bottom line is that the good match between the recovered trajectory and ground truth, although being desirable is not always possible and this is especially true for high-frequency views. It seems, though, that the objective landscape has a nice property of allowing for multiple local minima, which are also good in terms of image quality.

The current major limitation of our method is the inability to correct for motion involving strong rotations (angles larger than 3°). The visual quality of a reconstructed image never gets worse than that of a degraded image, however, the stronger the rotation, the less improvement can be achieved. For rotation angles larger than 10°, the corrected image looks essentially the same as the degraded one. This limitation stems from the reduced form of our objective, where instead of a data fidelity term (difference between the observation and the forward model applied to a motion-free estimate), we use an empirical inverse of the observed motion. An exact empirical inversion is not possible since for strong rotations the amount of missing k-space information is significant. Losses occur in high-frequency corners, and in k-space holes that appear due to rotation of neighbouring views against each other. The inversion-free treatment of the problem capable of dealing with arbitrary rotations involves only the forward model, taking the form of alternations between the estimation of motion parameters and recovery of the sharp image. Both alternating steps have the data fidelity and regularization terms. Regularization terms are to be based on prior knowledge of the properties of motion parameters and the image. We experimented with the outlined alternating procedure, but experienced problems with convergence. An alternative to empirical inversion in a final image reconstruction step would be to use a proper mathematical inverse, i.e., computed by conjugate gradients. In respective experiments, we did not find the results obtained with a mathematical inverse to be better in terms of accuracy compared to the empirical inverse. The reason for this is that the objective function used to estimate motion parameters has an empirical inverse in its core resulting in inherently suboptimal estimates of motion parameters. A related effect, also due to an empirical inversion in the objective, is that the pose of the object in the reconstructed image can be different compared to the ground truth. Indeed, translating/rotating the whole object does not influence the value of the quality metric. We do not consider this to be a flaw of the method, though.

A strong assumption we make is that the imaged object behaves as a rigid body. Indeed, this is what allows us

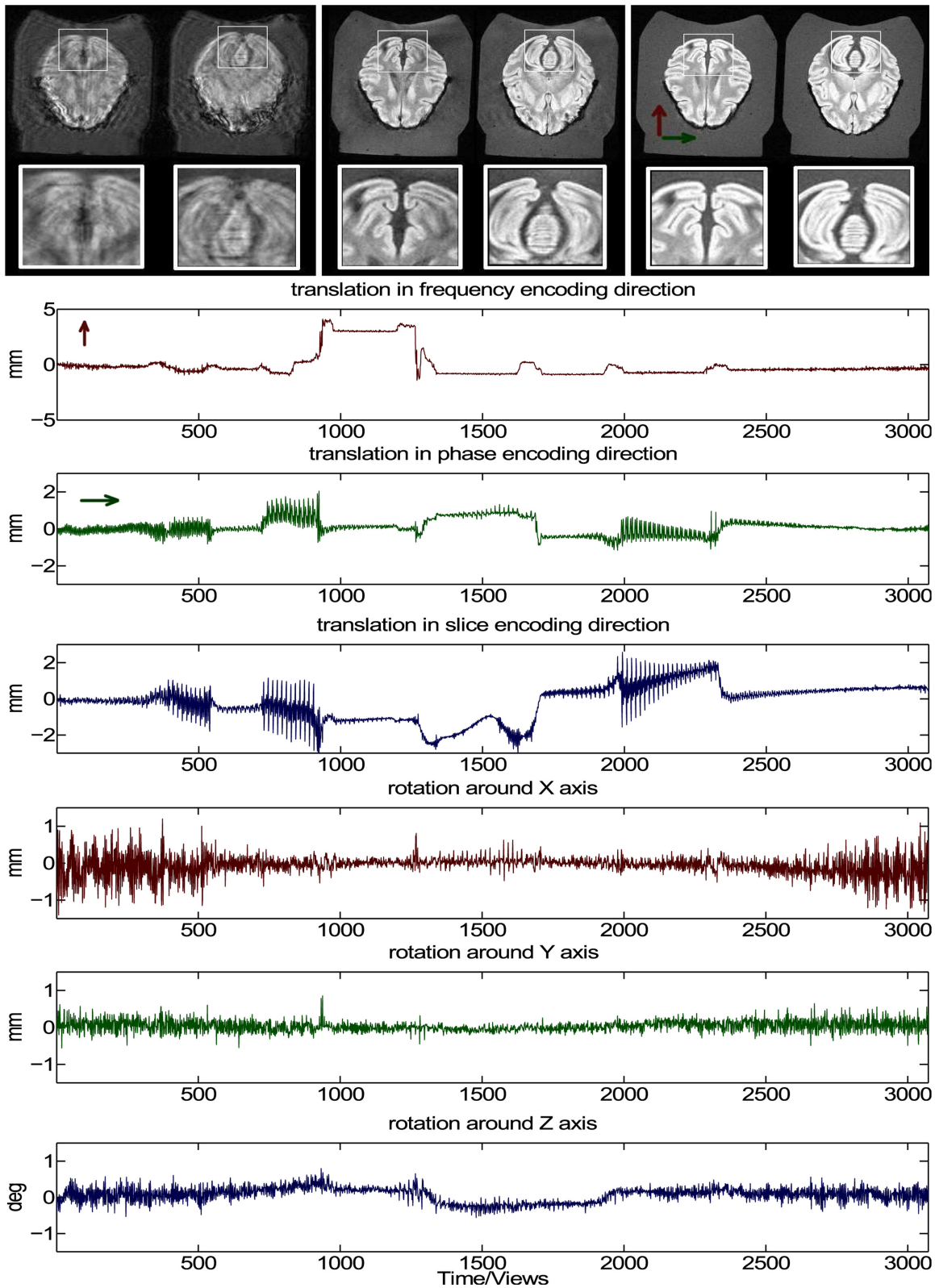

FIG. 8. Motion correction of 3D FLASH images. Monkey brain in fixation gel. Motion was induced by manually moving the probe in the scanner by means of specialized device. **Left:** motion corrupted slices. **Middle:** reconstruction. **Right:** no motion image. **Bottom:** recovered trajectories (from top to bottom—translations, then rotations). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

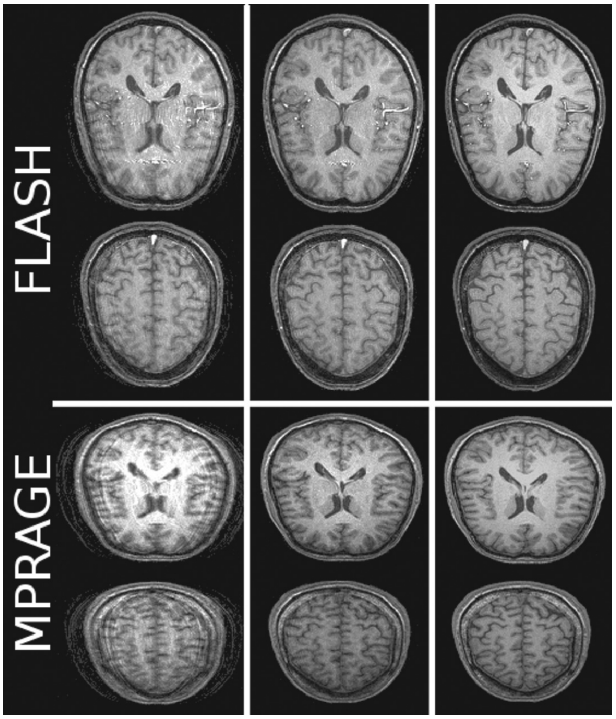

FIG. 9. Motion correction of 3D FLASH and MPRAGE images. Freely moving human subject. **Left:** motion corrupted slices. **Middle:** reconstruction. **Right:** no motion image.

to carry out fast multiplications with the matrix $\mathbf{A}_{\boldsymbol{\theta}}$ in the Fourier domain. Slight deviations from non rigidity are well-tolerated, however gross effects (e.g., due to movement of the tongue during the acquisition) make an artifact-free correction difficult, as we observed in a dedicated experiment. We also assume that the input to GradMC is the raw k-space data along with the order in which the k-space was sampled. In a clinical setting, such data is not always available, because often only magnitude images are preserved. Since real data from the scanner often has non uniform spatial phase, taking the modulus of complex-valued pixels in the spatial domain results in distortions of the motion structure in the Fourier domain. It does not longer hold that the same motion affects all coefficients in each k-space view. An interesting direction for future work is to understand what clues about motion still survive in magnitude images, with the hope of exploiting them for the correction.

In our approach, we assume that the k-space is acquired in a Cartesian line-by-line fashion. Extending our approach to radial/spiral trajectories is straight forward—i.e., for radial trajectories we expect our method to achieve higher accuracy, since each k-space view will contain low-frequency coefficients, which are crucial for estimation of motion parameters.

We have shown how to extend our method to handle data from multiple coils. To do the correction, we do not require sensitivity profiles of coils, however, if sensitivity profiles are known, GradMC can be modified to make many interesting correction scenarios possible. For example, inverting the sensitivity mask by subtracting it from the unity mask, and then penalizing intensities in the inverted region, it is possible to estimate the motion and correct for it. The underlying mechanism for this is that high-intensity pixels in out-of-profile regions of coils likely appear due to motion artifacts. Unfortunately, we cannot simply extend our method to include parallel imaging, because this would require a proper mathematical inverse to estimate the motion parameters. This, however, would mean that the analytical gradient is computationally out of reach.

In summary, we presented a retrospective method capable of blind estimation and correction of motion. We exploit the analytic formulation of our degradation model to search for motion parameters in a high-dimensional space being guided by the gradient of our cost function. Numerous experiments substantiate the great potential of the proposed method.

APPENDIX

Derivatives

In the following section, we show how to compute the derivatives $\partial\phi(\mathbf{F}^{\mathsf{H}}\mathbf{A}_{\boldsymbol{\theta}}\mathbf{y})/\partial\boldsymbol{\theta} \in \mathbb{R}^N$ of an image quality metric ϕ with respect to a parameter vector $\boldsymbol{\theta} \in \mathbb{R}^N$. Here, $\mathbf{A}_{\boldsymbol{\theta}} \in \mathbb{C}^{N\times N}$ is a parameterized matrix such that every row $[\mathbf{A}_{\boldsymbol{\theta}}]_i$, $i = 1 \dots N$ depends on a single scalar parameter θ_i only. Furthermore, $\mathbf{y} \in \mathbb{C}^N$ is an arbitrary vector, $\mathbf{F} \in \mathbb{C}^{N\times N}$ represents an arbitrary matrix and $\phi : \mathbb{C}^N \to \mathbb{R}$ is a continuously differentiable function.

We start off by computing the derivative $\partial\phi/\partial\mathbf{u}$ of our entropy criterion $\phi(\mathbf{u}) = -\mathbf{v}^\top \ln \mathbf{v},\ \mathbf{u} \in \mathbb{C}^N,\ \mathbf{v} \in \mathbb{R}_+^N$ from Eq. [6] which turns out to be

$$\frac{\partial\phi}{\partial\mathbf{u}} = \mathbf{v} \odot \mathbf{p}/\bar{\mathbf{u}} - \mathbf{u}\frac{\mathbf{v}^\top\mathbf{p}}{\mathbf{u}^{\mathsf{H}}\mathbf{u}},\ \mathbf{v} = \sqrt{\frac{\mathbf{u}\odot\bar{\mathbf{u}}}{\mathbf{u}^{\mathsf{H}}\mathbf{u}}}$$

where $\mathbf{p} := \partial\phi/\partial\mathbf{v} = -(\mathbf{1} + \ln\mathbf{v})$ is the derivative of the objective with respect to $\mathbf{v}$, $\operatorname{sign}(\mathbf{u})_i = u_i/|u_i|$ for $|u_i| \neq 0$ and 0 otherwise is the complex modulus and $|\mathbf{u}|_i = |u_i| = \sqrt{\Re[u_i]^2 + \Im[u_i]^2}$ is the absolute value. By noting that $\lim_{v\to 0} v\log v = 0$ for $v \in \mathbb{R}_+$, we see that $\phi(\mathbf{u})$ is well-behaved for all values except for $\mathbf{u} = \mathbf{0}$. In practice, one can circumvent that situation by using the stabilization $\mathbf{v}_\epsilon = \sqrt{\mathbf{u}\odot\bar{\mathbf{u}} + \epsilon^2\mathbf{1}}/\sqrt{\mathbf{u}^{\mathsf{H}}\mathbf{u} + \epsilon^2}$, $0 < \epsilon \ll 1$ such that $\phi_\epsilon(\mathbf{u}) = -\mathbf{v}_\epsilon^\top \ln\mathbf{v}_\epsilon$ and $\phi_\epsilon(\mathbf{0}) = 0$. Similarly, the stabilized derivative is given by

$$\frac{\partial\phi_\epsilon}{\partial\mathbf{u}} = \mathbf{u}\odot\mathbf{v}\odot\mathbf{p}/(\mathbf{u}\odot\bar{\mathbf{u}} + \epsilon^2\mathbf{1}) - \mathbf{u}\frac{\mathbf{v}_\epsilon^\top\mathbf{p}}{\mathbf{u}^{\mathsf{H}}\mathbf{u} + \epsilon^2}$$

with the limit $\partial\phi_\epsilon(\mathbf{0})/\partial\mathbf{u} = \mathbf{0}$. In our implementation, we did not observe any trouble even when using $\epsilon = 0$. The following derivations do not depend on the particular functional form of the vector $\partial\phi/\partial\mathbf{u} \in \mathbb{C}^N$.

Let us denote by $\mathbf{A}'_{\boldsymbol{\theta}}$ the matrix such that $[\frac{\partial}{\partial\theta_i}\mathbf{A}_{\boldsymbol{\theta}}\mathbf{y}]_i = [\mathbf{A}'_{\boldsymbol{\theta}}\mathbf{y}]_i$ i.e., $\mathbf{A}'_{\boldsymbol{\theta}}$ contains all the information needed for the Jacobian $\partial\mathbf{A}_{\boldsymbol{\theta}}\mathbf{y}/\partial\boldsymbol{\theta}^\top = \operatorname{dg}(\mathbf{A}'_{\boldsymbol{\theta}}\mathbf{y})$, which is diagonal because every row i depends on θ_i only, since the cross terms vanish. Now, let $\partial\phi(\mathbf{u}_{\boldsymbol{\theta}})/\partial\mathbf{u}_{\boldsymbol{\theta}}$ (where $\mathbf{u}_{\boldsymbol{\theta}} = \mathbf{F}^{\mathsf{H}}\mathbf{A}_{\boldsymbol{\theta}}\mathbf{y}$) denote the derivative of the objective function $\phi : \mathbb{C}^N \to \mathbb{R}$. By means of the chain rule, we have

$$\frac{\partial\phi(\mathbf{u}_{\boldsymbol{\theta}})}{\partial\boldsymbol{\theta}} = \frac{\partial\mathbf{u}_{\boldsymbol{\theta}}^\top}{\partial\boldsymbol{\theta}}\frac{\partial\phi(\mathbf{u}_{\boldsymbol{\theta}})}{\partial\mathbf{u}_{\boldsymbol{\theta}}} = \Re\left[\overline{(\mathbf{A}'_{\boldsymbol{\theta}}\mathbf{y})}\odot\left(\mathbf{F}\frac{\partial\phi(\mathbf{u}_{\boldsymbol{\theta}})}{\partial\mathbf{u}_{\boldsymbol{\theta}}}\right)\right], \quad [11]$$

where $\Re[\mathbf{z}]$ denotes the real part of the complex vector $\mathbf{z}$. For 2D images, we have one rotational and two translational parameters and for 3D images, we have both three rotational and translational parameters for each time step, so that we end up with either $\boldsymbol{\theta} = [\boldsymbol{\theta}_x, \boldsymbol{\theta}_y, \boldsymbol{\theta}_\alpha] \in \mathbb{R}^{T\times 3}$ or $\boldsymbol{\theta} = [\boldsymbol{\theta}_x, \boldsymbol{\theta}_y, \boldsymbol{\theta}_z, \boldsymbol{\theta}_a, \boldsymbol{\theta}_b, \boldsymbol{\theta}_c] \in \mathbb{R}^{T\times 6}$. In the following, we look at the general case $\boldsymbol{\theta} \in \mathbb{R}^{N\times 6}$, where the parameter vector is expanded to the same size as the matrix by replicating the single parameter per phase encode.

To compute (matrix vector multiplications with) $\mathbf{A}'$ in general, we need to take a closer look at how the MVM $\mathbf{Ay}$ (we write $\mathbf{A}$ and $\mathbf{A}'_\#$ where $\# \in \{x, y, z, a, b, c\}$ in the following) is computed and how it depends on the motion parameters. First of all, $\mathbf{Ay}$ can be decomposed into a translational part $\mathbf{t} = \mathbf{t}_x \odot \mathbf{t}_y \odot \mathbf{t}_z \in \mathbb{C}^N$ by pointwise multiplication with a location specific phase shift and a rotational part $\mathbf{R} \in \mathbb{R}^{N\times N}$ that is in fact a resampling matrix transforming the Cartesian grid $\mathbf{g} \in \mathbb{R}^{N\times 3}$ into the (phase-encode-wise) rotated grid $\mathbf{k} \in \mathbb{R}^{N\times 3}$, such that $\mathbf{Ay} = \mathbf{t}\odot(\mathbf{Wy})$. In particular, we have $\mathbf{t}_x = \exp(-2\pi i \mathbf{k}_x \odot \boldsymbol{\theta}_x)$, where the derivatives are given by $\mathbf{t}'_x = -2\pi i \boldsymbol{\theta}_x \odot \mathbf{t}_x$. Rotated grid vectors $\mathbf{k}^i \in \mathbb{R}^3$ are obtained from the Cartesian grid vectors by a local rotation $\mathbf{R}^i\mathbf{g}^i$ as achieved by a rotation matrix $\mathbf{R}^i \in \mathbb{R}^{3\times 3}$, depending on three rotation parameters a_i, b_i, c_i from $\boldsymbol{\theta}_i \in \mathbb{R}^6$. Furthermore, entries of the gridding matrix W_{ij} are obtained by application of a windowing function $w : \mathbb{R} \to \mathbb{R}$ (we use cubic resampling windows for ease of GPU implementation) to the new k-space location k_j and the old Cartesian location g_i, such that $W^{ij} = w(k^i_x - g^j_x)w(k^i_y - g^j_y)w(k^i_z - g^j_z)$. Using the derivative of the window function w', we define the matrix $\mathbf{W}'_a$ (and similarly $\mathbf{W}'_b$ and $\mathbf{W}'_c$) containing the derivatives $[\mathbf{W}'_a]_{ij} = w'(k^i_x - g^j_x)w(k^i_y - g^j_y)w(k^i_z - g^j_z)$. Also, the derivative $\mathbf{k}'_\# \in \mathbb{R}^{N\times 3}$ of the rotated k-space grid with respect to rotation parameters $\# \in \{a, b, c\}$ can be written as $(\mathbf{k}^i{}_\#)' = (\mathbf{R}^i_\#)'\mathbf{g}^i$, where $\mathbf{g}^i$ is the Cartesian location and $(\mathbf{R}^i_\#)' = \partial\mathbf{R}^i_\#/\partial\#_i \in \mathbb{R}^{3\times 3}$ is the derivative of the rotation matrix in with respect to the rotation parameters. We use the axis-angle representation for the matrices $\mathbf{R}$ but other representation such as Euler angles are possible, too. Putting everything together, we obtain

$$\mathbf{A}'_x\mathbf{y} = (\mathbf{t}'_x \odot \mathbf{t}_y \odot \mathbf{t}_z) \odot (\mathbf{Wy}), \text{ and}$$
$$\mathbf{A}'_a\mathbf{y} = \mathbf{t} \odot \sum_{\#\in\{a,b,c\}} (\mathbf{W}'_\#\mathbf{y}) \odot \mathbf{k}'_a.$$

Computationwise, matrix vector multiplications with $\mathbf{A}'$ have a complexity of $\mathcal{O}(N)$ each so that the derivatives with respect to each of the six DoFs of the general 3D case can be computed efficiently.

REFERENCES

1. Wood M, Henkelman, R. MR image artifacts from periodic motion. Med Phys 1985;12:143–151.
2. Zaitsev M, Dold C, Sakas G, Hennig J, Speck, O. Magnetic resonance imaging of freely moving objects: prospective real-time motion correction using an external optical motion tracking system. Neuroimage 2006;31:1038–1050.
3. Qin L, van Gelderen P, Derbyshire JA, Jin F, Lee J, de Zwart JA, Tao Y, Duyn JH. Prospective Head-Movement Correction for High-Resolution MRI Using an In-Bore Optical Tracking System. Magn Reson Med 2009;62:924–934.
4. Lee TY, Su HR, Lai SH, chiun Chang T. Compensation of mostion artifacts in MRI via graph-based optimization. In IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida, USA, 2009. pp. 2192–2199.
5. Lin W, Wehrli F, Song, H. Correcting bulk in-plane motion artifacts in MRI using the point spread function. IEEE Trans Med Imaging 2005;24:1170–1176.
6. Lin W, Song H. Extrapolation and correlation (EXTRACT): a new method for motion compensation in MRI. IEEE Trans Med Imaging 2009;28:82–93.
7. Hedley M, Yan H, Rosenfeld D. Motion artifact correction in MRI using generalized projections. IEEE Trans Med Imaging 1991;10:40–46.
8. Johnson KO, Robison RK, Pipe JG. Rigid Body Motion Compensation for Spiral Projection Imaging. IEEE Trans Med Imaging 2011;30:655-665.
9. Fu Z, Wang Y, Grimm R, Rossman P, Felmlee J, Riederer S, Ehman R. Orbital navigator echoes for motion measurements in magnetic resonance imaging. Magn Reson Med 1995;34:746–753.
10. Welch EB, Manduca A, Grimm RC, Ward HA, Jack CR Jr. Spherical navigator echoes for full 3-D rigid body motion measurement in MRI. Med Reson Med 2002;47:32–41.
11. Manduca A, McGee K, Welch E, Felmlee J, Grimm R, Ehman R. Autocorrection in MR Imaging: adaptive motion correction without navigator echoes. Radiology 2000;215:904–909.
12. Frieden B. Restoring with maximum likelihood and maximum entropy. J Opt Soc Am 1972;62:511–518.
13. Muller R, Buffington A.Real-time correction of atmospherically degraded telescope images through image sharpening. J Opt Soc Am 1974;64:1200–1210.
14. Bocker R, Jones S. ISAR motion compensation using the burst derivative measure as a focal quality indicator. Int J Imaging Syst Tech 1992;4:285–297.
15. Atkinson D, Hill D, Stoyle P, Summers P, Keevil S. Automatic correction of motion artifacts in magnetic resonance images using an entropy focus criterion. IEEE Trans Med Imaging 1997;16:903–910.
16. Atkinson D, Hill DLG, Stoyle PNR, Summers PE, Clare S, Bowtell R, Keevil SF. Automatic Compensation of Motion Artifacts in MRI. Magn Reson Med 1999;41:163–170.
17. Lin W, Ladinsky G, Wehrli F, Song H. Image metric-based correction (autofocusing) of motion artifacts in high-resolution trabecular bone imaging. J Magn Reson Imaging 2007;26:191–197.
18. McGee K, Felmlee J, Jack C, Manduca A, Riederer S, Ehman R. Autocorrection of three-dimensional time-of-flight MR angiography of the circle of Willis. Am J Roentgenol 2001;176:513–518.
19. Batchelor P, Atkinson D, Irarrazaval P, Hill D, Hajnal J, Larkman D. Matrix description of general motion correction applied to multishot images. Magn Reson Med 2005;54:1273–1280.
20. Cheng JY, Alley MT, Cunningham CH, Vasanawala SS, Pauly JM, Lustig M. Nonrigid motion correction in 3D using autofocusing with localized linear translations. Magn Reson Med 2012;68:1785–1797.
21. Odille F, Vuissoz P, Marie P, Felblinger J. Generalized Reconstruction by Inversion of Coupled Systems (GRICS) applied to free-breathing MRI. Magn Reson Med 2008;60:146–157.
22. Bracewell RN, The Fourier Transform and its Applications. 2nd ed. New York: McGrawHill; 1986.
23. McGee K, Manduca A, Felmlee J, Riederer S, Ehman R. Image metric-based correction (Autocorrection) of motion effects: analysis of image metrics. J Magn Reson Imaging 2000;11:174–181.
24. Byrd RH, Lu P, Nocedal J, Zhu C. A limited memory algorithm for bound constrained optimization. SIAM J Sci Stat Comput 1995;16:1190–1208.
25. Nickisch H. glm-ie: generalised linear models inference & estimation toolbox. J Machine Learn Reson 2012;13:1699–1703.
26. White N, Roddey C, Shankaranarayanan A, Han E, Rettmann D, Santos J, Kuperman J, Dale A. PROMO: real-time prospective motion correction in MRI using image-based tracking. Magn Reson Med 2010;63:91–105.
27. Tisdall MD, Hess AT, Reuter M, Meintjes EM, Fischl B, van der Kouwe AJW. Volumetric navigators for prospective motion correction and selective reacquisition in neuroanatomical MRI. Magn Reson Med 2012;68:389–399.
28. Maclaren J, Herbst M, Speck O, Zaitsev M. Prospective motion correction in brain imaging: a review. Magn Reson Med 2012. doi: 10.1002/mrm.24314.
29. Beatty P, Nishimura D, Pauly J. Rapid gridding reconstruction with a minimal oversampling ratio. IEEE Trans Med Imaging 2005;24:799–808.
30. Johnson KO, Pipe JG. Convolution kernel design and efficient algorithm for sampling density correction. Magn Reson Med 2009;61:439–447.