Unwarping Echo Planar Images Using CMTK[1]

Torsten Rohlfing

August 22, 2012

Neuroscience Program, SRI International, Menlo Park, CA

Abstract

This document describes the workflow for unwarping echo planar MR images (EPI), in particular diffusion-weighted images, using an acquisition method with reversed phase encoding and the tools of the Computational Morphometry Toolkit (CMTK). This technique requires an additional acquisition (additional $b = 0$ image for diffusion imaging) with reversed phase encoding direction, but no field map. Simple 6-direction diffusion data are provided with this article for demonstration.

Contents

---

1 Introduction

Echo-planar images are essential for the acquisition of diffusion weighted as well as functional (i.e., BOLD) MR images. Unfortunately, these images are subject to significant spatial distortions to due susceptibility changes at, for example, air-tissue boundaries.

The most common technique to correct these distortions is via the acquisition of a field map, from which the distortion field can be computed. This has several disadvantages, such as the need to acquire the field maps, and the need to compute unwrapped phase maps. Also, the resulting distortion correction is not necessarily of good quality.

Holland *et al.* [1] have recently introduced a field map-free undistortion method, which makes use of the fact that the distortion behaves "symmetrically" when the phase encoding direction of the acquisition is reversed.

Thus, from a pair of images acquired with opposing phase encoding direction, one can compute a symmetric deformation field pair that undistorts both input images. If this is done for a pair of opposing-phase-encode $b = 0$ images in diffusion images, and the actual diffusion-weighted images are acquired analogous to one of the two $b = 0$ images, then the entire set of diffusion images can be unwarped using the deformation field computed for the appropriate $b = 0$ image.

We describe herein the tools of the Computational Morphometry Toolkit (CMTK), which implement the numerical computational part of this unwarping strategy. Demonstration data are provided with this article to test the described tools and workflows.

1.1 CMTK

The Computational Morphometry Toolkit (CMTK) is **free software**, and that's as in both free beer and free speech. CMTK is available both in source code, licensed under the GPLv3, and as pre-compiled binary distributions from http://nitrc.org/projects/cmtk/. If you are using NeuroDebian, you can also install CMTK directly.

We shall assume that CMTK has been installed such that its tools can be run as

```
cmtk <tool> <arg1> <arg2> ...
```

To perform the processing stages described in this article, you will need CMTK release 2.2.4[1] or later. Versions prior to 2.2.0 do not support EPI unwarping, and versions 2.2.0 through 2.2.3 contain a bug that prevented proper initialization of the deformation field in some cases.

2 Step-by-Step

2.1 Imaging

On your MR scanner of choice, you will need to implement an acquisition like the one described in Ref. [1]. In short, before the $b = 0$ image of your favorite DWI acquisition sequence, you need to acquire an additional $b = 0$ image with phase-encoding direction reversed, thus resulting in an image that is flipped in the phase-encoding direction and also distorted by the "opposite" distortion field.

---

[1]Has not been released yet. Please obtain latest CMTK source code from Subversion repository, https://www.nitrc.org/svn/cmtk/trunk/core/.

Note that CMTK will not help you set up the acquisition - this is outside the scope of the toolkit and also outside the scope of this manual. Also note that the CMTK developers cannot offer any assistance with the imaging itself. Please consult with experts, such as the authors of Ref. [1], or the capable MR physicist of your choice.

2.2   DICOM Image Stacking

Assume that the DICOM files containing the EPI data are stored in the "`dicom/`" directory. These are stacked into 3D images in NIFTI-format using the following CMTK command:

```
cmtk dcm2image -vx -O dwi/image%n.nii dicom/
```

This will result in a series of continuously numbered images in NIFTI format, all stored in the `dwi/` directory.

Let us assume that we are using a 6-gradient-direction DWI setup with a single reverse phase-encoded $b = 0$ image acquired first, followed by the standard $b = 0$ image, followed by six diffusion images with different gradient directions. In this case, we should find the following files:

`image1.nii` – reverse phase-encoded $b = 0$ image

`image2.nii` – standard phase-encoded $b = 0$ image

`image3.nii` – 1st diffusion image

`image4.nii` – 2nd diffusion image

`image5.nii` – 3rd diffusion image

`image6.nii` – 4th diffusion image

`image7.nii` – 5th diffusion image

`image8.nii` – 6th diffusion image

This image naming system is easy to achieve but less than transparent. For the sake of clarity, we shall assume below that the images have been renamed to `b0_rev.nii` for the reverse-encoded $b = 0$ image, `b0_fwd.nii` for the standard $b = 0$ image, and `b1.nii` through `b6.nii` for the diffusion-weighted images.

Demonstration images following this naming convention are distributed with the article. They can be found in the `inputs/` folder of the attached data archive in zip format, or obtained from the CMTK Subversion repository at

https://www.nitrc.org/svn/cmtk/trunk/doc/UnwarpEchoPlanar/data/inputs/

2.3   Unwarping EPI

First, we compute the deformations to unwarp the two opposite-direction phase encoded $b = 0$ images. We also write out the Jacobian determinant map of the forward-encoded image, which we will need to correct for signal pile up:

(a) b0_rev.nii.gz

(b) b0_fwd.nii.gz

(c) b1.nii.gz

(d) b2.nii.gz

(e) b3.nii.gz
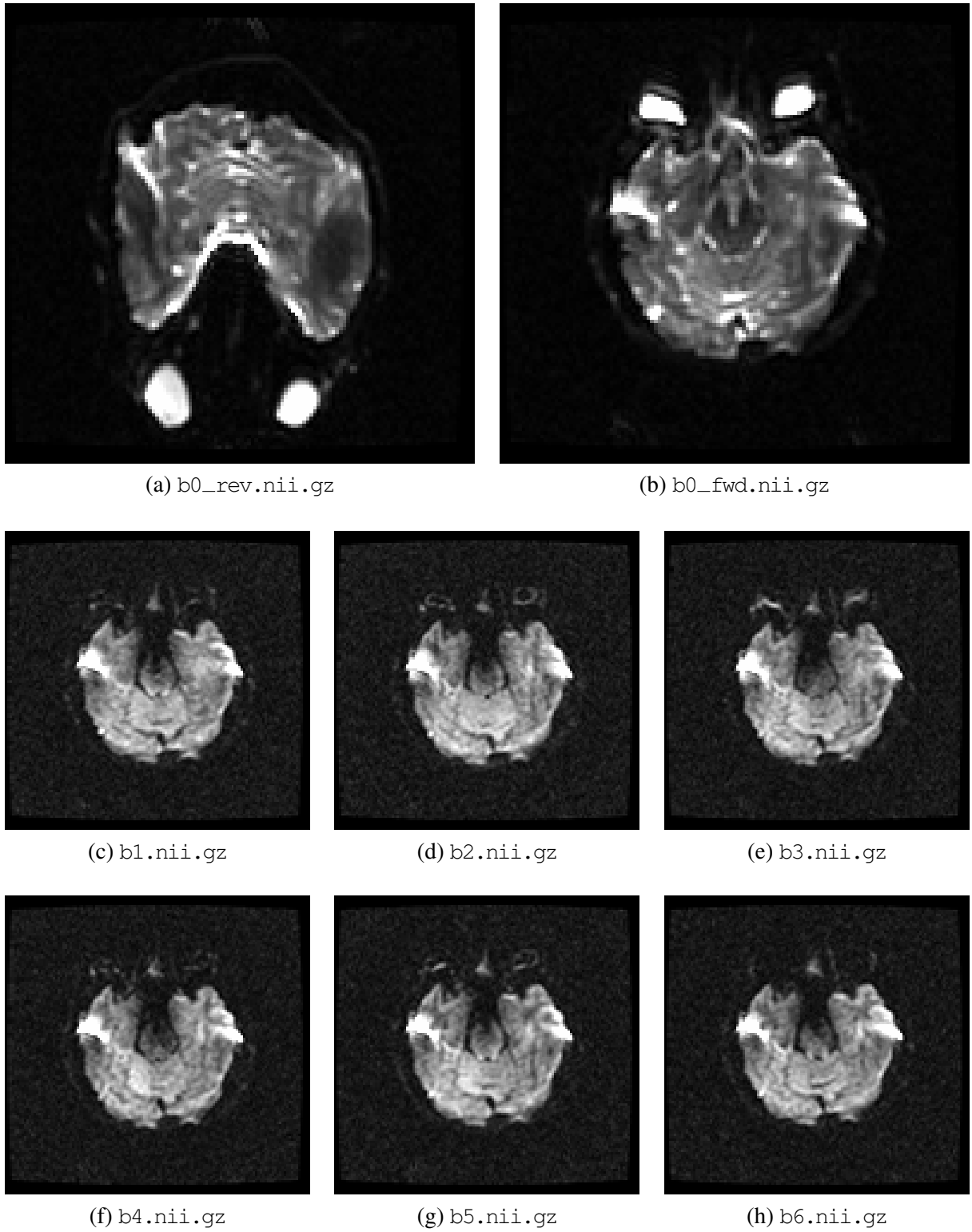
(f) b4.nii.gz

(g) b5.nii.gz

(h) b6.nii.gz

Figure 1: Demonstration data provided with this paper (file names given in parentheses). (a) Axial slice from $b = 0$ image acquired with reversed phase-encoding direction. (b) Same slice from $b = 0$ image acquired with standard phase-encoding direction. (c) through (h) Same slice from six different diffusion-weighted images acquired with standard phase-encoding direction.

```
cmtk epiunwarp --write-jacobian-fwd epiunwarp/jacobian_fwd.nii \
  inputs/b0_fwd.nii.gz inputs/b0_rev.nii.gz \
  epiunwarp/b0_fwd.nii epiunwarp/b0_rev.nii epiunwarp/dfield.nrrd
```

Second, apply the computed deformation to the first diffusion-weighted image (analogous to the remaining diffusion images) to create an unwarped reformatted image:

```
cmtk reformatx --floating inputs/b1.nii --linear -o epiunwarp/b1.nii \
  epiunwarp/b0_fwd.nii epiunwarp/dfield.nrrd
```

Third, compute the pixel-wise multiplication of the unwarped diffusion image with the Jacobian of the deformation:

```
cmtk imagemath --in epiunwarp/b1.nii epiunwarp/jacobian_fwd.nii --mul \
  --out epiunwarp/b1.nii
```

The result is the final, distortion-corrected image. For the demonstration data distributed with this article the unwarped images are shown in Fig. 2. The unwarped images in NIFTI format are also contained in the unwarp/ folder of the data archive that accompanies this article, and they can be obtained from the CMTK Subversion repository under

https://www.nitrc.org/svn/cmtk/trunk/doc/UnwarpEchoPlanar/data/unwarp/

The exact commands used to create the unwarped images are also contained in the make_unwarp shell script in the accompanying data archive.

2.4   Unwarping EPI with Eddy-Current Correction

The distortion caused by the B_0 field inhomogeneity and corrected by the epiunwarp tool is not the only type of distortion that spatially affects echo-planar images. A second source of distortions are eddy currents, which primarily lead to shear transformations within each acquired slice. To avoid unnecessary multiple interpolation, which would degrade the already low-quality image data further, CMTK provides a script, correct_dwi_distortion, which performs correction of both eddy current and B_0 field distortions.

The script is called as follows:

```
cmtk correct_dwi_distortion unwarp_eddy \
  inputs/b0_rev.nii.gz inputs/b0_fwd.nii.gz inputs/b?.nii.gz
```

The first argument, unwarp_eddy, is a directory, which will be created by the script and will hold all output files, most importantly the corrected images. The second and third argument are the reverse-encoded and standard-encoded $b = 0$ image files, and the remaining arguments, here abbreviated by the "?" wildcard, are the diffusion weighted images, b1.nii through b6.nii.

Upon completion of the script, the newly created output directory will contain the unwarped image files, each named with the same file name as its corresponding input image file.

The output directory will also contain dfield_fwd.nrrd, the deformation field for the standard-encoded $b = 0$ image (and the diffusion-weighted images); jacobian.nii, the Jacobian determinant map for intensity correction; and the eddy/ sub-directory, which contains the eddy current-correcting coordinate transformations between the $b = 0$ image and each of the diffusion-weighted images.
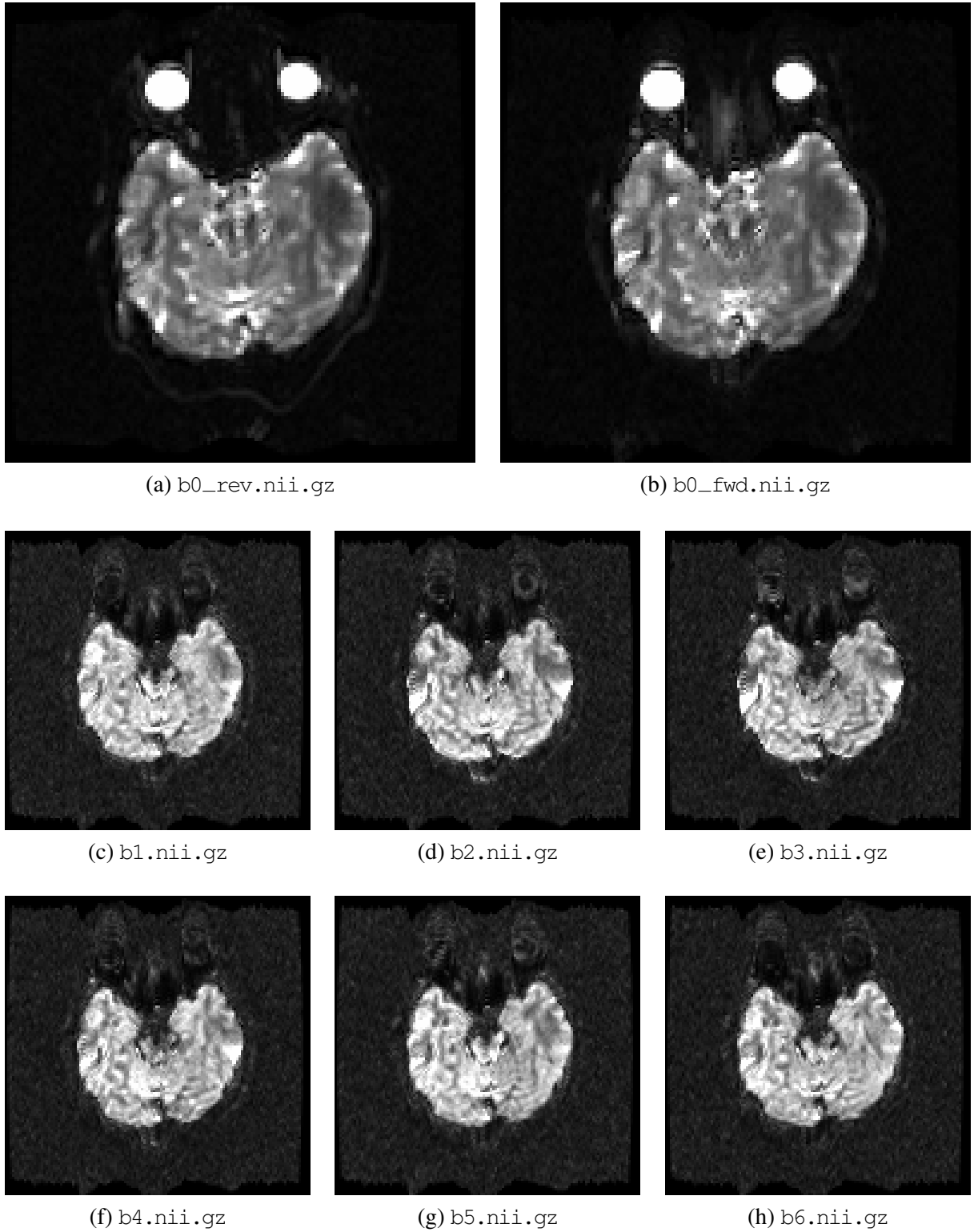
(a) b0_rev.nii.gz

(b) b0_fwd.nii.gz

(c) b1.nii.gz

(d) b2.nii.gz

(e) b3.nii.gz

(f) b4.nii.gz

(g) b5.nii.gz

(h) b6.nii.gz

Figure 2: Unwarped echo planar images. These images were unwarped and intensity corrected using the deformation and Jacobian determinant fields computed by the epiunwarp tool. The image files in NIFTI format are provided with this article in the data/unwarp/ directory.

For the demonstration data distributed with this article the unwarped and eddy current-corrected images are shown in Fig. 3. The unwarped images files in NIFTI format are also contained in the unwarp_eddy/ folder of the data archive that accompanies this article, and they can be obtained from the CMTK Subversion repository under

https://www.nitrc.org/svn/cmtk/trunk/doc/UnwarpEchoPlanar/data/unwarp_eddy/

The exact commands used to create the unwarped images are also contained in the make_unwarp_eddy shell script in the accompanying data archive.

2.5   Unwarping EPI with Eddy Current and Subject Motion Correction

A third common source of artifacts in imaging in general is subject motion. In the context of this article, subject motion can cause considerable problems because changes of the subject position over the course of a diffusion-weighted imaging session mean that the same unwarping deformation can no longer be applied to all diffusion-weighted images.

Thus, we note, in the presence of significant subject motion, the correction of image distortion will necessarily be compromised (this is equally true for correction using field maps, as in the presence of motion the acquired field maps will not in general be aligned with the acquired echo-planar images).

With this in mind, we still would like to correct subject motion as well as B_0-field distortion and eddy current effects as much as possible. CMTK provides a script for this purpose as well, which is called entirely analogous to the aforementioned undistortion and eddy current correction script:'

```
cmtk correct_dwi_distortion_and_motion unwarp_eddy_motion \
  inputs/b0_rev.nii.gz inputs/b0_fwd.nii.gz inputs/b?.nii.gz
```

Internally, both scripts are organized very similarly, and indeed the correct_dwi_distortion_and_motion script begins by correcting eddy current effects and B_0-field distortion in exactly the same way as does the correct_dwi_distortion script. Subsequently, however, the undistorted images are all aligned using 3D rigid transformations, to correct for subject motion.

For the demonstration data distributed with this article the unwarped, eddy current-corrected, and motion-corrected images are shown in Fig. 4. The image files in NIFTI format are also contained in the unwarp_eddy/ folder of the data archive that accompanies this article, and they can be obtained from the CMTK Subversion repository under

https://www.nitrc.org/svn/cmtk/trunk/doc/UnwarpEchoPlanar/data/unwarp_eddy/

The exact commands used to create the unwarped images are also contained in the make_unwarp_eddy_motion shell script in the accompanying data archive.

3   Limitations

Currently, the correct_dwi_distortion and correct_dwi_distortion_and_motion scripts assume axially acquired data with phase encoding in the anterior/posterior direction. The epiunwarp tool does not have this limitation, but its support for other acquisition schemes is simply not yet exposed by the wrapper scripts. Likewise, the registration tool used to compute eddy current corrections can be used in a way
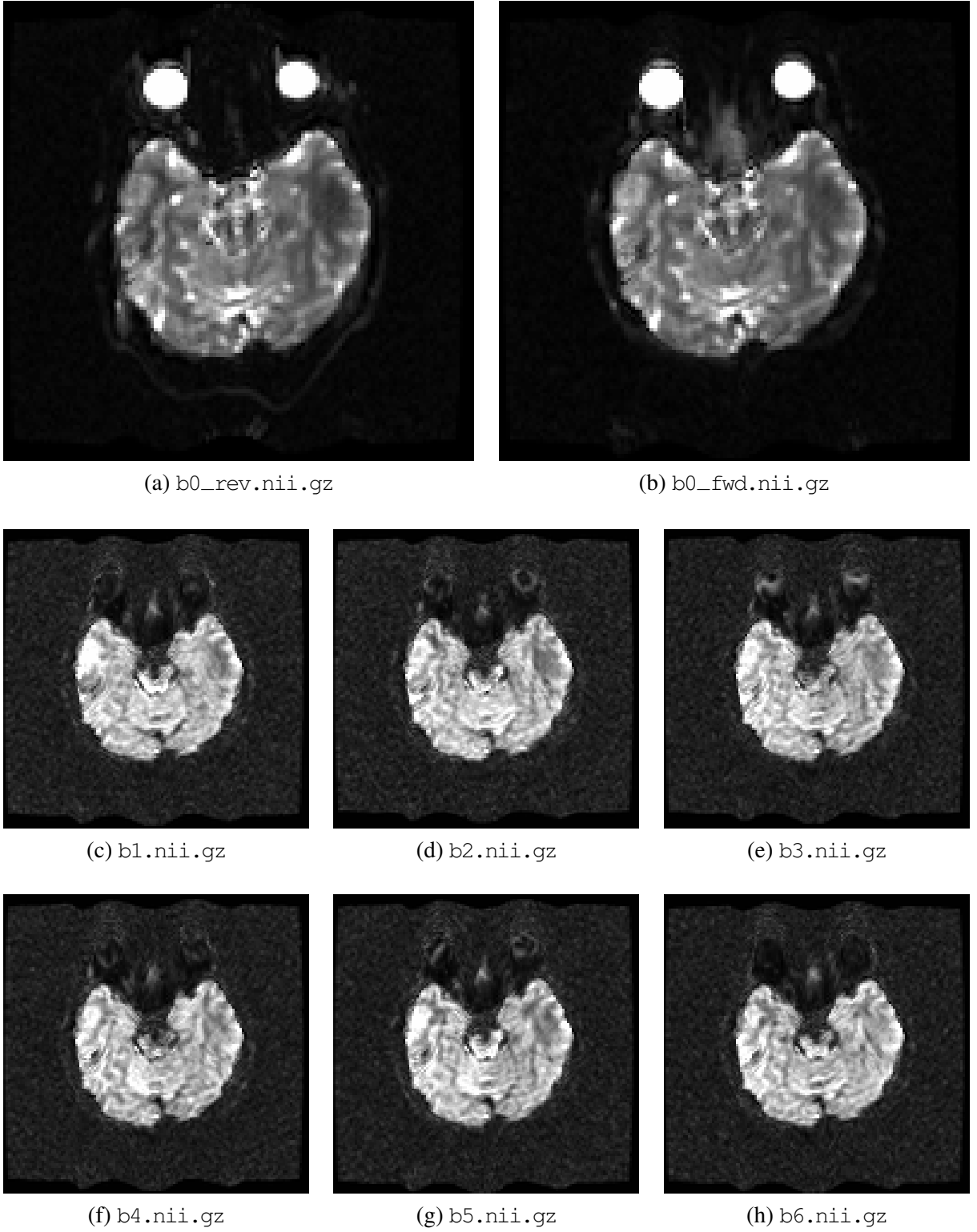
(a) b0_rev.nii.gz

(b) b0_fwd.nii.gz

(c) b1.nii.gz

(d) b2.nii.gz

(e) b3.nii.gz

(f) b4.nii.gz

(g) b5.nii.gz

(h) b6.nii.gz

Figure 3: Unwarped and eddy current-corrected images. These images were computed using the correct_dwi_distortion script. The image files in NIFTI format are provided with this article in the data/unwarp_eddy/ directory.

(a) b0_rev.nii.gz

(b) b0_fwd.nii.gz

(c) b1.nii.gz

(d) b2.nii.gz
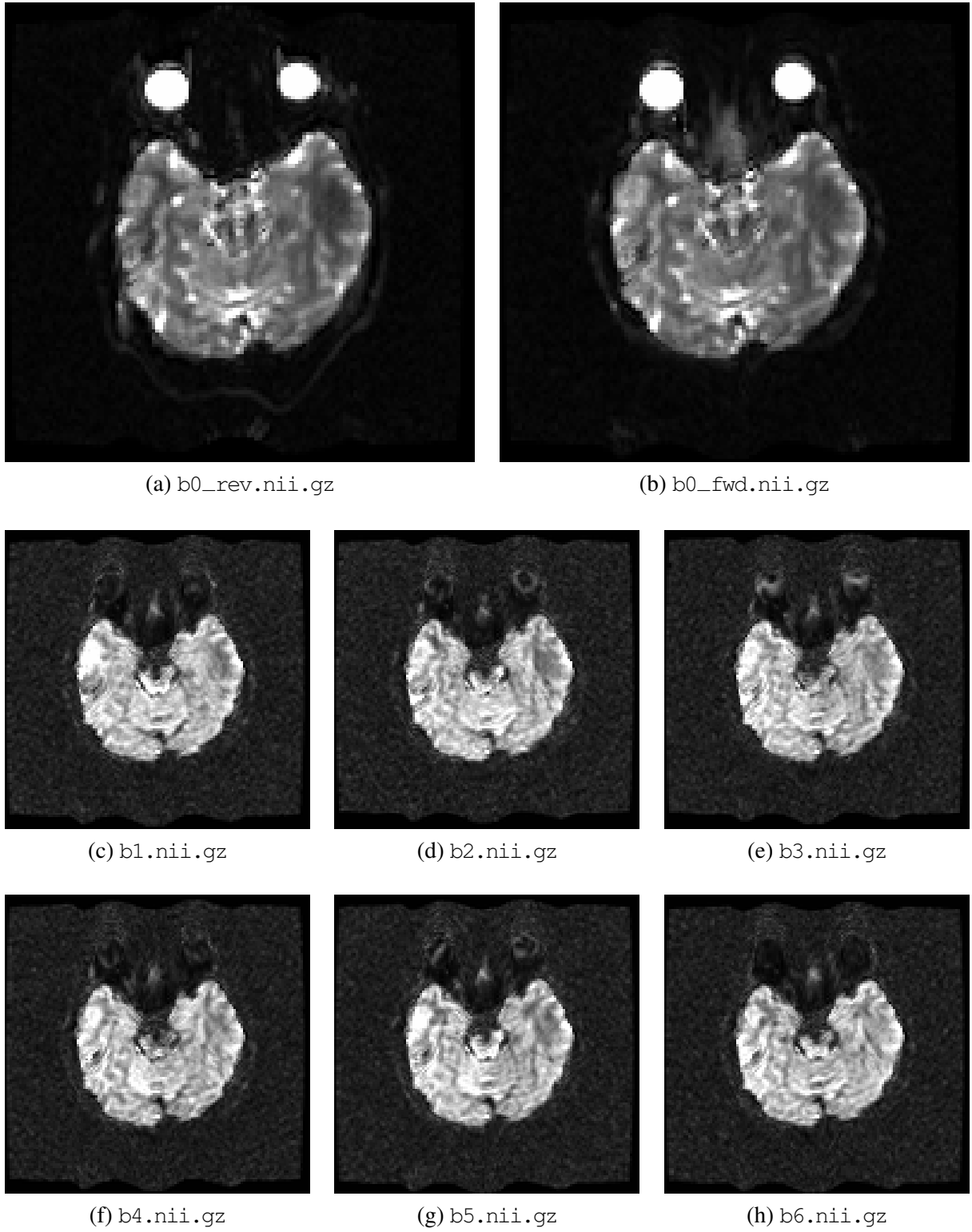
(e) b3.nii.gz

(f) b4.nii.gz

(g) b5.nii.gz

(h) b6.nii.gz

Figure 4: Unwarped, eddy current-corrected, and motion-corrected images. These images were computed using the correct_dwi_distortion_and_motion script. The image files in NIFTI format are provided with this article in the data/unwarp_eddy/ directory.

to support non-axial acquisitions, but again, use of this ability by the two wrapper scripts has yet to be implemented.

References

[1] D. Holland, J. M. Kuperman, and A. M. Dale. "Efficient correction of inhomogeneous static magnetic field-induced distortion in Echo Planar Imaging." *NeuroImage*, **50**(1):175–183, 2010. http://dx.doi.org/10.1016/j.neuroimage.2009.11.044. 1, 2.1