THY COMPUTE

- So is this solved?
- Far from it

$$\mathbf{B}^T\mathbf{B} = \mathbf{B}_z^T\mathbf{B}_z \bar{\otimes} \mathbf{B}_y^T\mathbf{B}_y \otimes \mathbf{B}_x^T\mathbf{B}_x$$

$$(\mathbf{B}_z \otimes \mathbf{B}_y \otimes \mathbf{B}_x)^T \, \mathbf{DD} (\mathbf{B}_z \otimes \mathbf{B}_y \otimes \mathbf{B}_x)$$

REGISTRATION TIME

DIPY

Home Workshops Tutorials Gallery CLI Documentation Community Login

Symmetric Diffeomorphic Registration in 3D

Symmetric Diffeomorphic Registration in 3D

- Symmetric Diffeomorphic Registration in 3D
  - References

This example explains how to register 3D volumes using the Symmetric Normalization (SyN) algorithm proposed by Avants et al. [Avants09] (also implemented in the ANTs software [Avants11])

We will register two 3D volumes from the same modality using SyN with the Cross Correlation (CC) metric.

```
import numpy as np
from dipy.align.imwarp import SymmetricDiffeomorphicRegistration
from dipy.align.imwarp import DiffeomorphicMap
from dipy.align.metrics import CCMetric
from dipy.core.gradients import gradient_table
from dipy.data import get_fnames
from dipy.io.image import load_nifti, save_nifti
from dipy.io.gradients import read_bvals_bvecs
import os.path
from dipy.viz import regtools
```

Let's fetch two b0 volumes, the first one will be the b0 from the Stanford HARDI dataset

```
hardi_fname, hardi_bval_fname, hardi_bvec_fname = get_fnames('stanford_hardi')

stanford_b0, stanford_b0_affine = load_nifti(hardi_fname)
stanford_b0 = np.squeeze(stanford_b0)[..., 0]
```

The second one will be the same b0 we used for the 2D registration tutorial