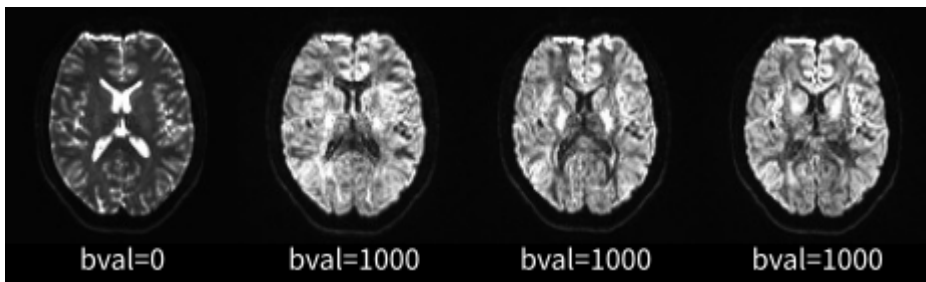




3. `bvals`和`bvecs`这两个文件表示什么意思？DWI的大致原理是在不同方向上施加一个梯度磁场，这个梯度磁场的强度用b-value来表示，b-value越大、水分子弥散的距离越大、图像信号变化越明显；DTI模型总共需要估计六个参数，所以至少需要在6个不同方向上施加梯度，梯度方向用b-vector来表示；还需要至少一个没有施加梯度的图像作为参考，常称为B0像。上面的文件总共1行65列，表示总共采集了65个图像，第1个图像的b-value是0，后64个图像的b-value是1000；`bvecs`文件总共有3行65列，表示在每个图像上所施加的梯度方向。

4. B0图像和DWI图像:



上图是数据中的前四个图像，第一个图像是B0像，后三个图像是DWI像，虽然b-value都是1000，但是梯度方向不同，因此图像信号也会有变化。

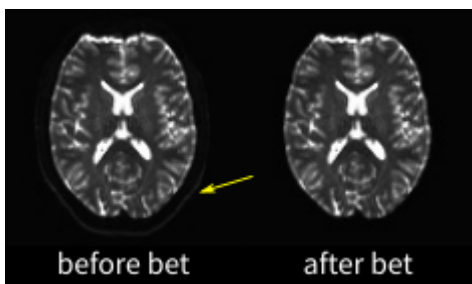
### 三、涡流/头动校正

在拟合张量模型之前，需要对图像进行一些涡流和头动校正，这里使用的是FSL 5.0.11。

#### 1. 脑提取

脑提取的目的就是为了获得一个（去除颅骨后）脑的mask，可以使用如下命令：

```
fslroi dti_data.nii.gz b0.nii.gz 0 1  
bet b0.nii.gz b0_brain.nii.gz -m -f 0.3
```



#### 2. `acqparams.txt`和`index.txt`

新建一个名为 `acqparams.txt` 的文本文件，内容为 `0 1 0 0.05`，该文件描述的是图像采集的信息。根据FSL官网的[文档](#)，该文件的正确性与否并不重要，在大多数情况下使用前面的设置即

可。另外，正确地填写这些信息需要对扫描参数有更深入的了解；新建一个名为 `index.txt` 的文本文件，方法为：

```
indx=""
for ((i=1; i<=65; i+=1))
do
    indx="$indx 1";
done
echo $indx > index.txt
```

上述代码的作用就是新建一个一列全为1的数，并保存到 `index.txt` 文件中；这个文件中的1表示该图像采集的参数对应于 `acqparams.txt` 文件的第一行。由于 `acqparams.txt` 里只有一行，所以所有图像都是1。

### 3. 涡流/头动校正

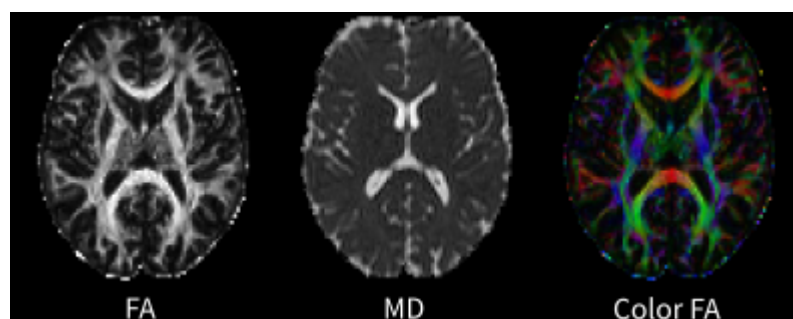
```
eddy_openmp --imain=dti_data.nii.gz --mask=b0_brain_mask.nii.gz \
    --acqp=acqparams.txt --index=index.txt --bvecs=bvecs --bvals=bvals \
    --out=eddy_corrected_data
```

`eddy_corrected_data.nii.gz` 即为涡流/头动校正后的数据，  
`eddy_corrected_data.eddy_rotated_bvecs` 为头动校正后的b-vector文件。

## 四、张量拟合

```
dtifit -k eddy_corrected_data.nii.gz -o dti -m b0_brain_mask.nii.gz \
    -r eddy_corrected_data.eddy_rotated_bvecs -b bvals --save_tensor
```

张量拟合结束后就可以得到一些定量指标，比如FA（Fractional Anisotropy）和MD（Mean Diffusivity）：



FA反映的是组织结构的方向性，如果水分子的弥散运动在某些方向上受到阻碍，在另一些方向上不受阻碍，则FA较大；FA较大的地方主要是在白质部分。MD表征水分子在所有方向上的平均弥散距离，如果没有受到阻碍，则MD较大；MD较大的地方主要在脑脊液。Color FA的计算方法是第一个特征向量（first eigenvector, 文件名为 `dti_v1.nii.gz`）乘上FA；第一个特征向量被认为反映（一个体素内）纤维束的平均方向，所以Color FA表示FA较大的区域的纤维束的方向。一般地，红色表示左右（X轴），绿色表示前后（Y轴），蓝色表示上下（Z轴）。

## 五、确定性纤维束追踪

纤维束追踪就是重建出神经纤维束，分为确定性纤维束追踪和概率纤维束追踪两种类型，纤维束追踪需要对人脑的纤维束有相当的先验知识（而我没有这样的先验知识）。这里使用TrackVis进行确定性纤维束追踪。由于纤维束追踪是根据张量拟合的结果进行的，我现在还不清楚是否可以用FSL的拟合结果来作为TrackVis的输入，因此需要用TrackVis重新进行张量拟合并在此基础上进行确定性纤维束追踪。TrackVis实际上包含两个相对独立的软件，一个是TrackVis用于可视化，一个是Diffusion Toolkit用于张量拟合和纤维束追踪，这里不做区分。

### 1. 准备文件

由于TrackVis不能进行涡流/头动校正，使用FSL的校正后的结果作为输入。另外，TrackVis的b-vector的格式不同于FSL，首先FSL的b-vector的格式是每列表示一个梯度方向，TrackVis是每行表示一个梯度方向，其次TrackVis需要去掉b-value为0所对应的梯度方向。下面是一个简单的R脚本，将FSL的bvecs转换为TrackVis的格式：

```
tmp <- as.matrix(read.table('eddy_corrected_data.eddy_rotated_bvecs'));
tmp <- t(tmp);    ## transpose the matrix
tmp <- tmp[-1,]; ## remove the b0 rows
write.table(tmp, 'trackvis_bvecs', row.names=FALSE, col.names=FALSE);
```

```
0.9998748468 0.004242372498 0.01524116265
-0.001985023882 0.9910491508 -0.1334827345
-0.01714178242 0.5413708134 -0.8406091849
0.5961638117 -0.791111436 -0.136862725
-0.2242421927 -0.6313897744 -0.7423357676
-0.8876333838 -0.3109156389 -0.3397623308
0.8043029464 0.05242739946 -0.5919021357
0.2315036004 0.8851403207 -0.4036492234
0.9390192453 0.1024770035 -0.3282397306
0.5073860481 -0.8099956096 0.2940518843
0.3455705748 -0.7824944927 0.5179607579
0.4519189437 -0.5368017987 0.7124697167
-0.4761718816 -0.4953545886 -0.7265563782
-0.6173832129 0.607957979 -0.4992244628
-0.5849877179 0.001667913561 0.8110404355
-0.821492133 -0.5509134299 -0.1471226298
0.8882048035 0.02649528646 0.4586831442
0.2835720181 -0.4271348528 0.8585701649
```

上图即为修改后的bvecs

## 2. 张量拟合

```
dti_recon "eddy_corrected_data.nii.gz" "trackvis" -gm "trackvis_bvecs" \
-b 1000 -b0 1 -ot nii.gz
```

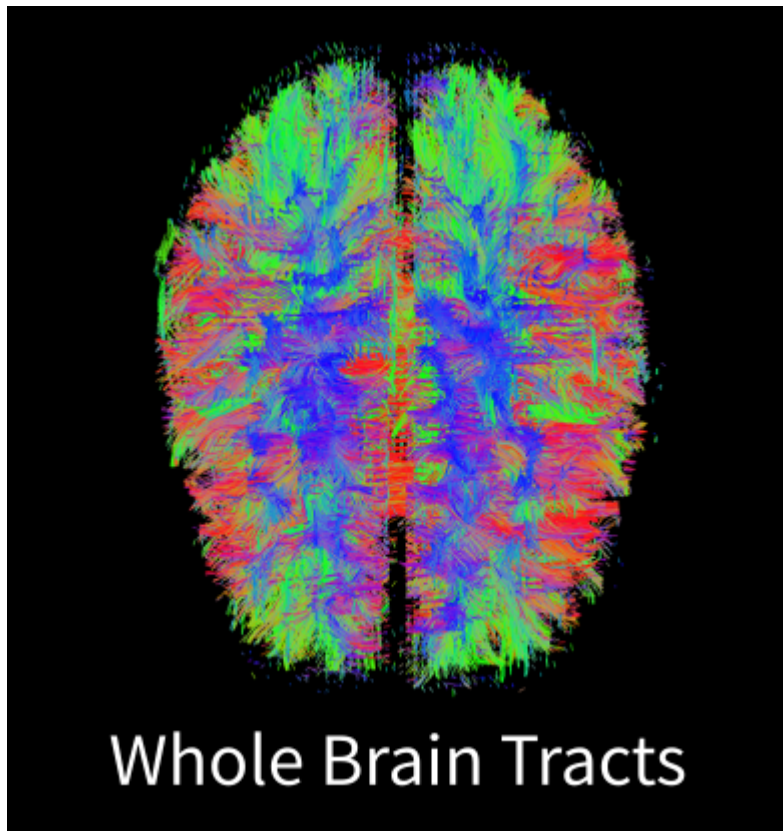
-gm 选项指定b-vector文件， -b 选项表示b-value， -b0 选项表示有多少个b0图像。FSL和TrackVis的张量拟合结果有微小的差异（大约小数点后第三位）。

## 3. 全脑纤维束追踪

```
dti_tracker "trackvis" "tmp.trk" -at 35 -m "b0_brain_mask.nii.gz" \
-m2 trackvis_fa.nii.gz 0.2 -sl -it nii.gz
spline_filter "tmp.trk" 1 "trackvis.trk"
```

-m 选项表示将追踪限制在脑mask内， -m2 选项表示将追踪限制在FA大于0.2的区域；  
spline\_filter 命令对追踪结果做一些平滑。在命令行输入 trackvis trackvis.trk，即可查看追踪结果：



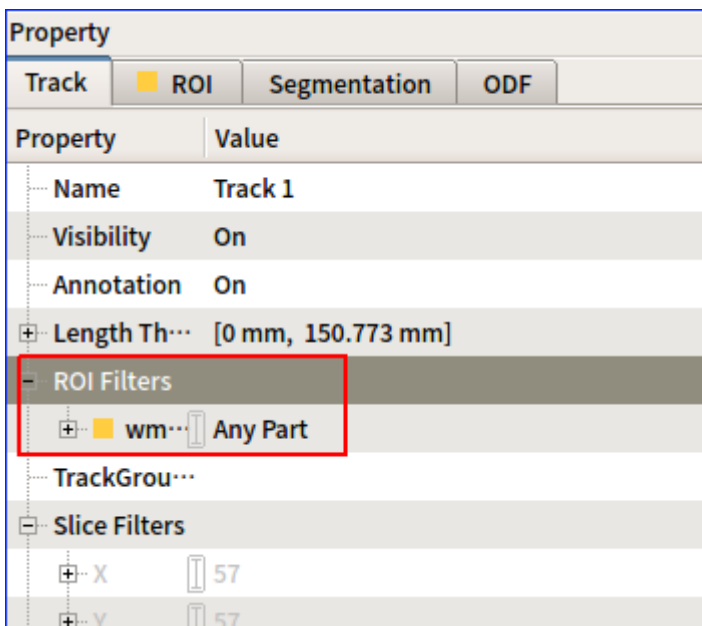
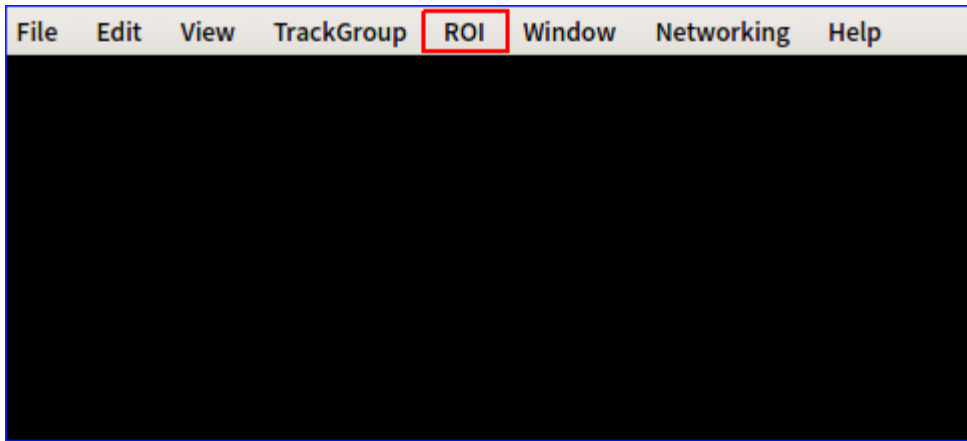


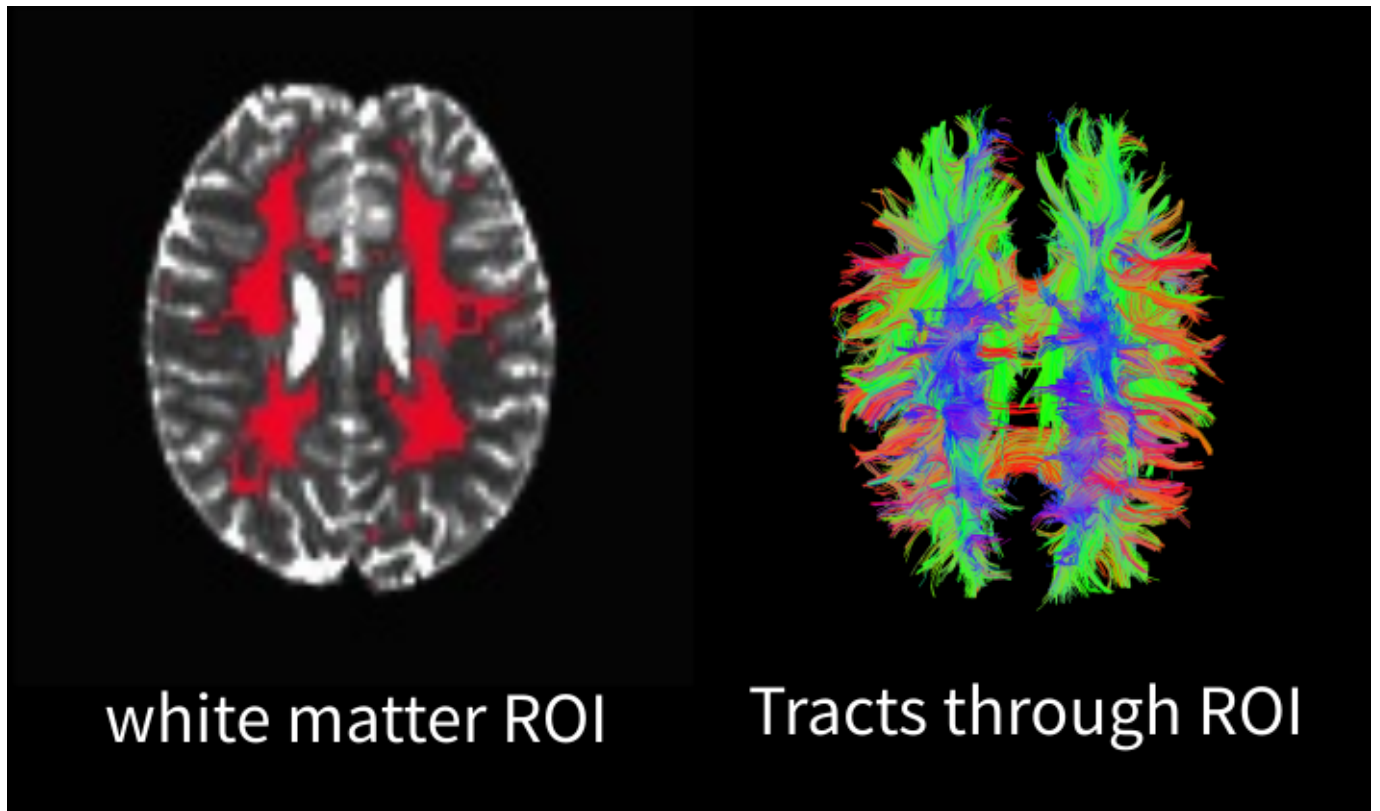
#### 4. 基于ROI纤维束追踪

上图是全脑的纤维束追踪结果，对于纤维束追踪的原理我现在还是模糊的，我当前的理解是，以每一个体素为起点（seed），都可以得到一条重建的纤维，那么以所有体素为起点的结果叠加在一起即为全脑的追踪结果。如果只想看到通过特定ROI的纤维束，可以使用该ROI作为过滤器，去除不感兴趣的纤维束。这里假设我感兴趣的区域是大脑白质，我通过对T1加权像进行分割和配准得到白质的mask，即白质ROI，实现代码如下：

```
## 去除T1像的颅骨
bet t1.nii.gz t1_brain.nii.gz
## 将T1像分割成灰质、白质和脑脊液三类
fast -t 1 -n 3 -H 0.1 -I 4 -l 20.0 -o t1 t1_brain.nii.gz
fslmaths t1_pve_2.nii.gz -thr 0.5 -bin wm_mask.nii.gz
## 使用BBR配准，将B0像配准到T1像
flirt -ref t1_brain.nii.gz -in b0_brain.nii.gz -dof 6 -omat b02t1_init.mat
flirt -ref t1_brain.nii.gz -in b0_brain.nii.gz -dof 6 -cost bbr \
      -wmseg wm_mask.nii.gz -init b02t1_init.mat -omat b02t1.mat -out b02t1Warp
## 将T1的白质mask转换到B0像空间
convert_xfm -omat t12b0.mat -inverse b02t1.mat
flirt -in wm_mask.nii.gz -ref b0_brain.nii.gz -out wm_maskWarp -init t12b0.mat \
      -applyxfm -interp nearestneighbour
## 将白质mask缩小，减少配准误差的影响
fslmaths wm_maskWarp -ero wm_mask_ero
```

在TrackVis的菜单栏选项 ROI -> New ROI From Nifti/Analyze Image，选择上一步得到的白质 ROI mask，即 `wm_mask_ero.nii.gz`；在TrackVis的右侧的Property界面，选择 ROI Filters -> Toggle Existing ROI，即得到通过白质ROI的纤维束：

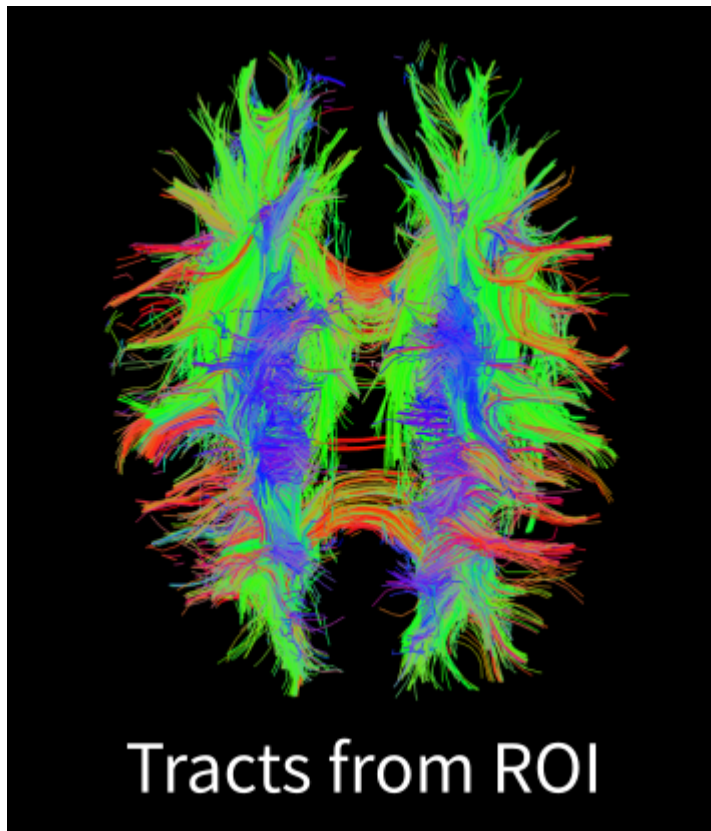




上述做法是先做出全脑的纤维束，然后使用ROI过滤掉不感兴趣的部分；也可以在做纤维束追踪的时候，将起点限制在ROI里，也就是只从ROI里的区域出发来重建纤维束，代码如下：

```
dti_tracker "trackvis" "tmp.trk" -at 35 -m "b0_brain_mask.nii.gz" \  
           -m2 trackvis_fa.nii.gz 0.2 -sl -it nii.gz -sm wm_mask_ero.nii.gz  
spline_filter "tmp.trk" 1 "wm_seed.trk"
```





## 六、小结

---

总结了使用FSL进行涡流/头动校正、张量拟合，以及使用TrackVis进行确定性纤维束追踪的基本方法。这些方法都是来自于FSL和TrackVis的官方文档（TrackVis的文档不够详细）。对于原理的理解上，还非常粗浅，有些理解完全是个人的猜测，需要进一步确认。因此，多有谬误，敬请指正。