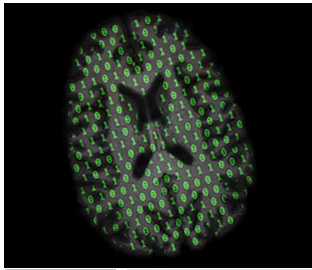
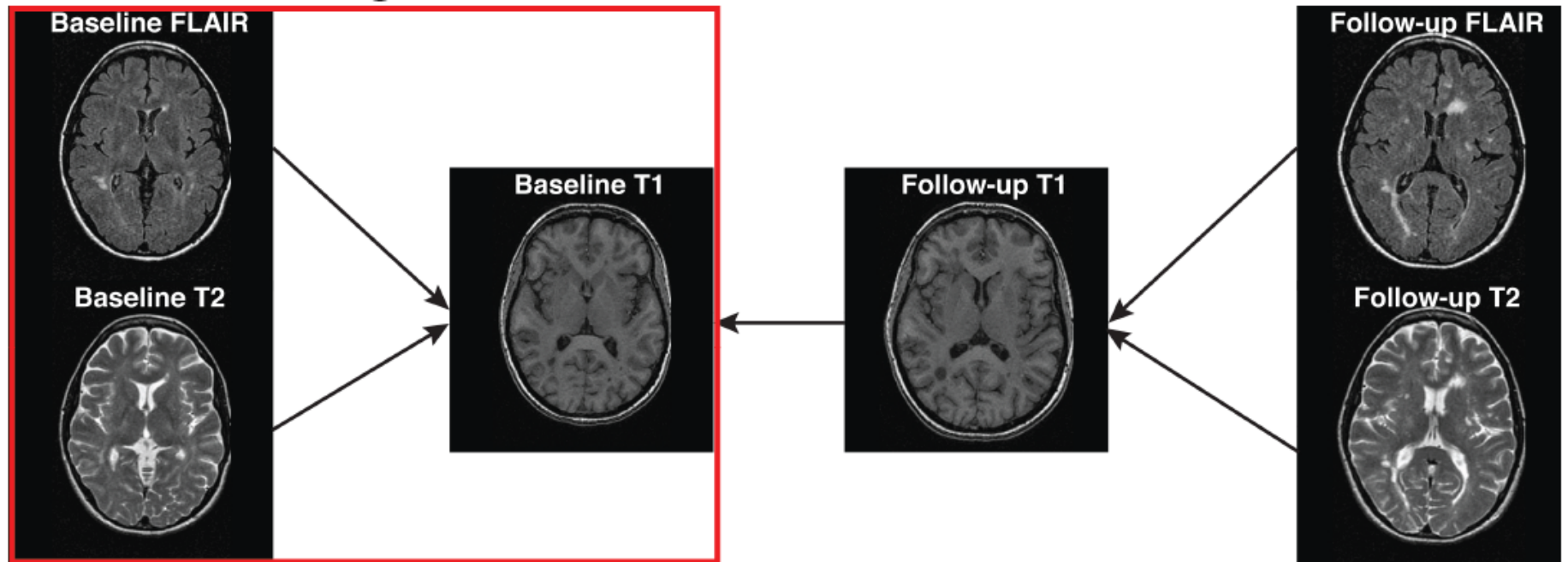


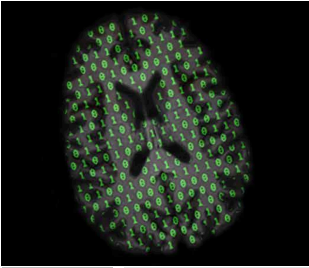


# THEME 4 / LECTURE 4: WRAPPER FUNCTIONS



# Wrapper Functions

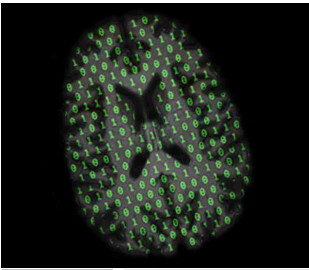




# Wrapper Functions

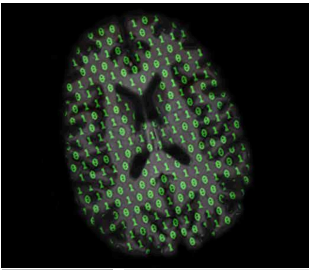
The `extrantsr` function `preprocess_mri_within` will do the following steps:

- Inhomogeneity correction
- Registration of the files to the first filename



# Within Visit 1 N4 Correction and Co-Registration

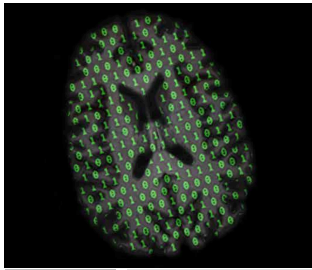
```
files = c("113-01-MPRAGE.nii.gz",  
          "113-01-T2w.nii.gz",  
          "113-01-FLAIR.nii.gz")  
  
files = file.path(mridir, files)  
outfiles = c("113-01-MPRAGE_processed.nii.gz",  
             "113-01-T2w_processed.nii.gz",  
             "113-01-FLAIR_processed.nii.gz")  
  
outfiles = file.path(mridir, outfiles)  
  
preprocess_mri_within(files = files, retimg = FALSE,  
outfiles = outfiles, correction = "N4", skull_strip = FALSE)
```



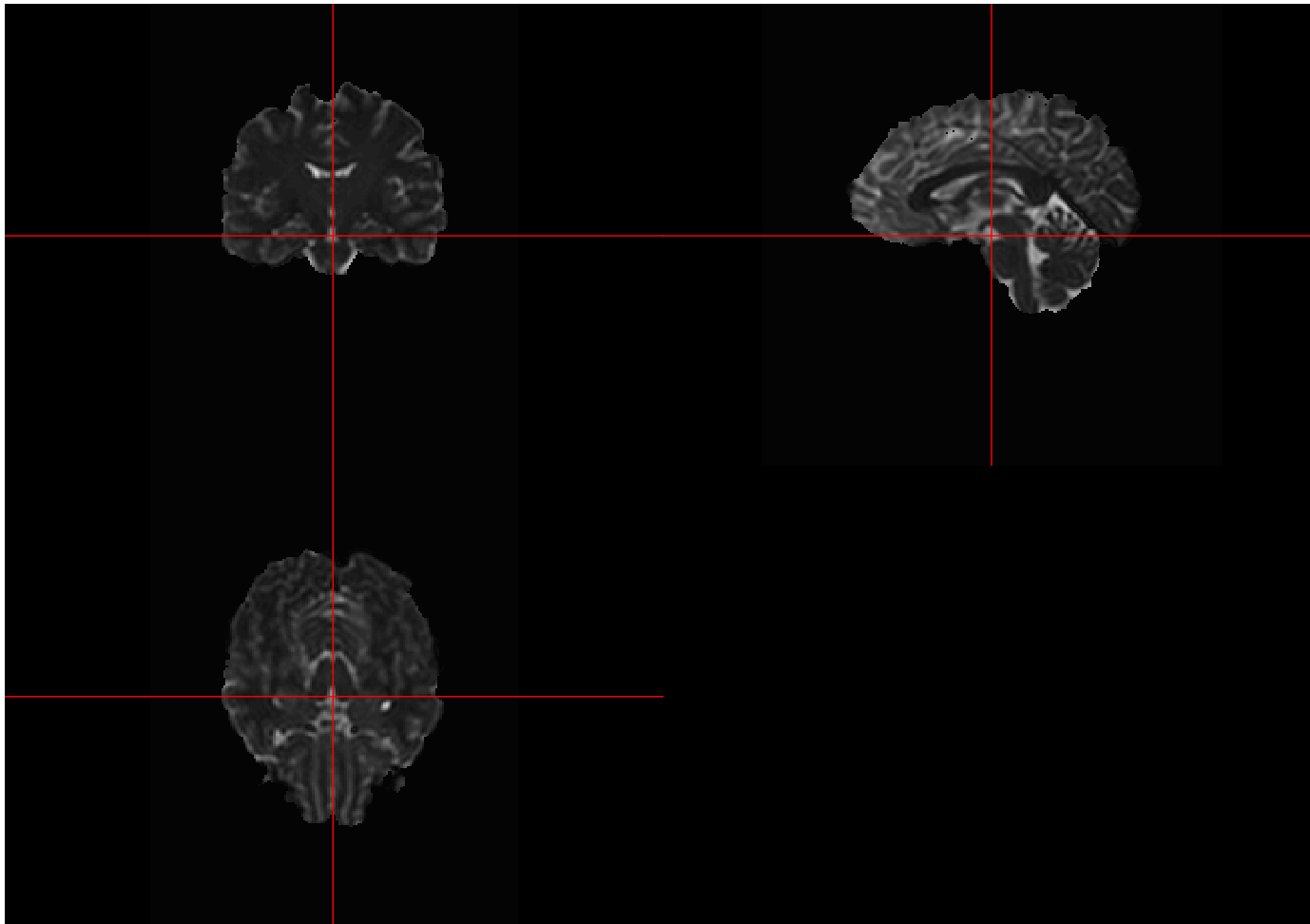
# Applying a Brain Mask to All Registered Images

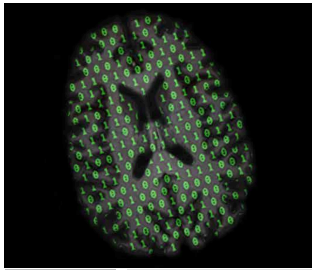
- Images from visit 1 are all in the same space as T1
- If we skull strip the T1 image then the mask can be applied to the other images to extract brain tissue

```
brain = fslbet_robust(img = outfiles[1],  
                      correct = FALSE, verbose = FALSE)  
mask = brain > 0  
  
masked_imgs = lapply(outfiles, fslmask,  
                      mask = mask, verbose = FALSE)  
  
orthographic(masked_imgs[[2]])
```

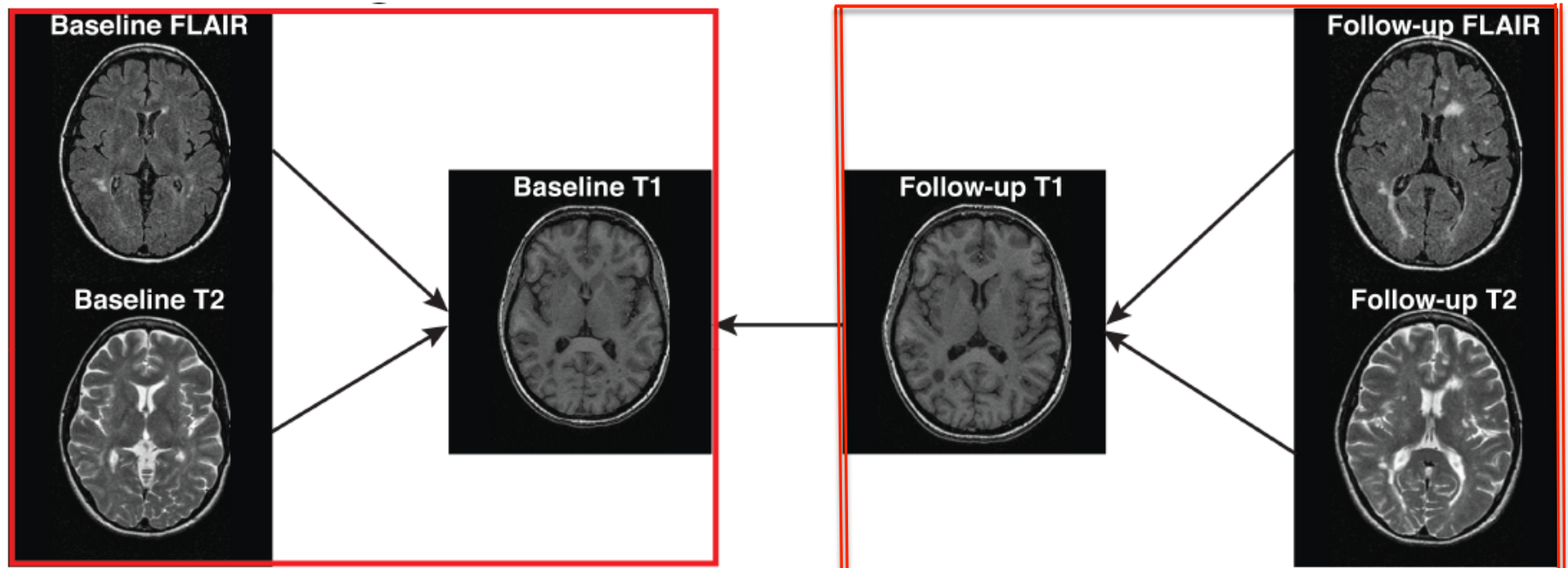


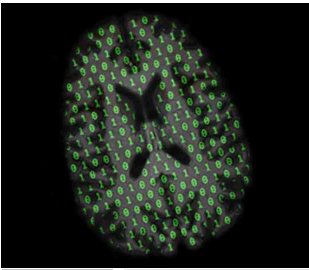
# Results for Baseline Masking





# Multi-Sequence Within-visit Co-Registration





## Within Visit 2 N4 Correction and Co-Registration

```
mridir2=file.path(kirbydir, "visit_2", "113")

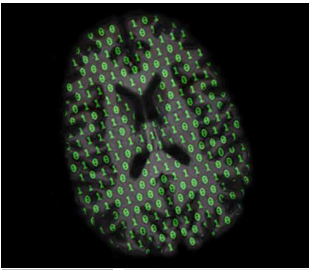
files2 = c("113-02-MPRAGE.nii.gz",
           "113-02-T2w.nii.gz",
           "113-02-FLAIR.nii.gz")

files2 = file.path(mridir2, files2)
outfiles2 = c("113-02-MPRAGE_processed.nii.gz",
              "113-02-T2w_processed.nii.gz",
              "113-02-FLAIR_processed.nii.gz")

outfiles2 = file.path(mridir2, outfiles2)

preprocess_mri_within(files = files2, retimg = FALSE,
outfiles = outfiles2, correction = "N4", skull_strip = FALSE)
```

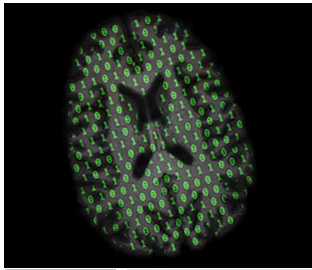




# Applying a Brain Mask to All Registered Images

- Images from visit 1 are all in the same space as T1
- If we skull strip the T1 image then the mask can be applied to the other images to extract brain tissue

```
brain2 = fslbet_robust(img = outfiles2[1],  
                      correct = FALSE, verbose = FALSE)  
mask2 = brain2 > 0  
  
masked_imgs2 = lapply(outfiles2, fslmask,  
                      mask = mask2, verbose = FALSE)  
  
orthographic(masked_imgs2[[3]])
```



# Results for Follow-up Masking

