Pre-Processing Neurohacking

Bui Dinh Ngoc

7/6/2016

Pre-Processing

This document is demo how to use R to Pre-Processing Neuro Image .We divide image pre-processing into four main conceptual steps

1. Inhomogeneity correction
2. Spatial interpolation
3. Skull stripping
4. Spatial registration

Download a NIfTI file from Neurohacking_data repository

```
options(fsl.path= "/usr/share/fsl/5.0")  # fsl-complete install 2 version 4.1 and 5.0 on ubuntu/debian
library(oro.nifti)
```

```
## oro.nifti 0.5.2
```

```
library(fslr)
```

```
## Loading required package: stringr
```

```
destfile <- "113-01-MPRAGE.nii.gz"
if(!file.exists(destfile))
{
  url <- "https://raw.githubusercontent.com/muschellij2/Neurohacking_data/master/kirby21/visit_1/113/113

  name <- file.path(getwd(), destfile)
  download.file(url, destfile,mode="wb") # NIfTI is binaryfile format
}
nim=readNIfTI("113-01-MPRAGE.nii.gz", reorient=FALSE)
```

Some statistics using FSL

```
Sys.setenv("LD_LIBRARY_PATH"="/usr/local/lib/") # R process maybe ignore LD_LIBRARY_PATH so i set it in
mean(nim)
```

```
## [1] 117377.2
```

```
fslstats(nim, opts= "-m")
```

```
## Warning in get.fsloutput(): Can't find FSLOUTPUTTYPE, setting to NIFTI_GZ
```

```
## FSLDIR='/usr/share/fsl/5.0'; PATH=${FSLDIR}/bin:${PATH};export PATH FSLDIR; sh "${FSLDIR}/etc/fslconf
```

```
## [1] "117377.191744"
```

```
fslstats("113-01-MPRAGE.nii.gz",opts="-m")
```

```
## FSLDIR='/usr/share/fsl/5.0'; PATH=${FSLDIR}/bin:${PATH};export PATH FSLDIR; sh "${FSLDIR}/etc/fslconf
```

```
## [1] "117377.191744"
```

```
fslstats.help()
```

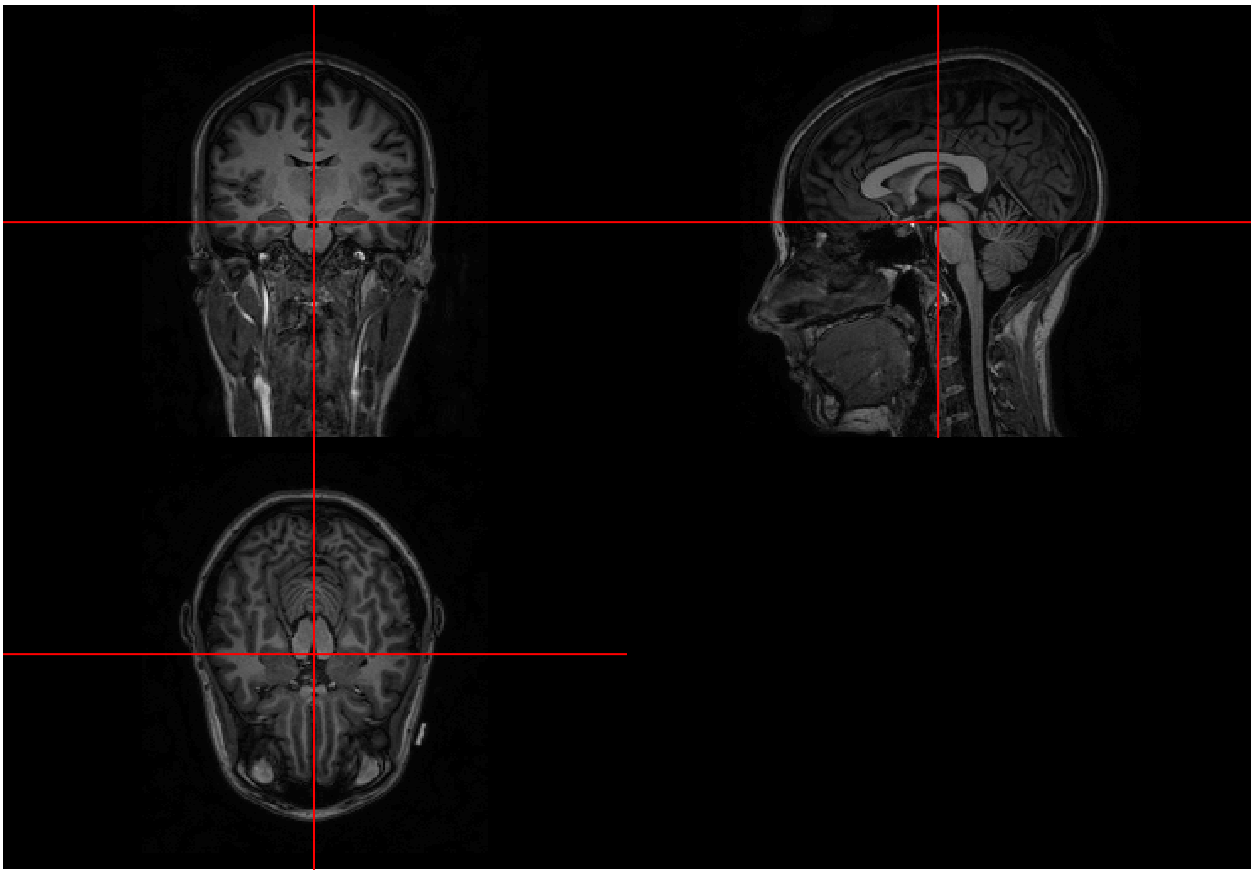
```
## Usage: fslstats [preoptions] <input> [options]preoption -t will give a separate output line for each
```

Bias Field Correction Using fslr

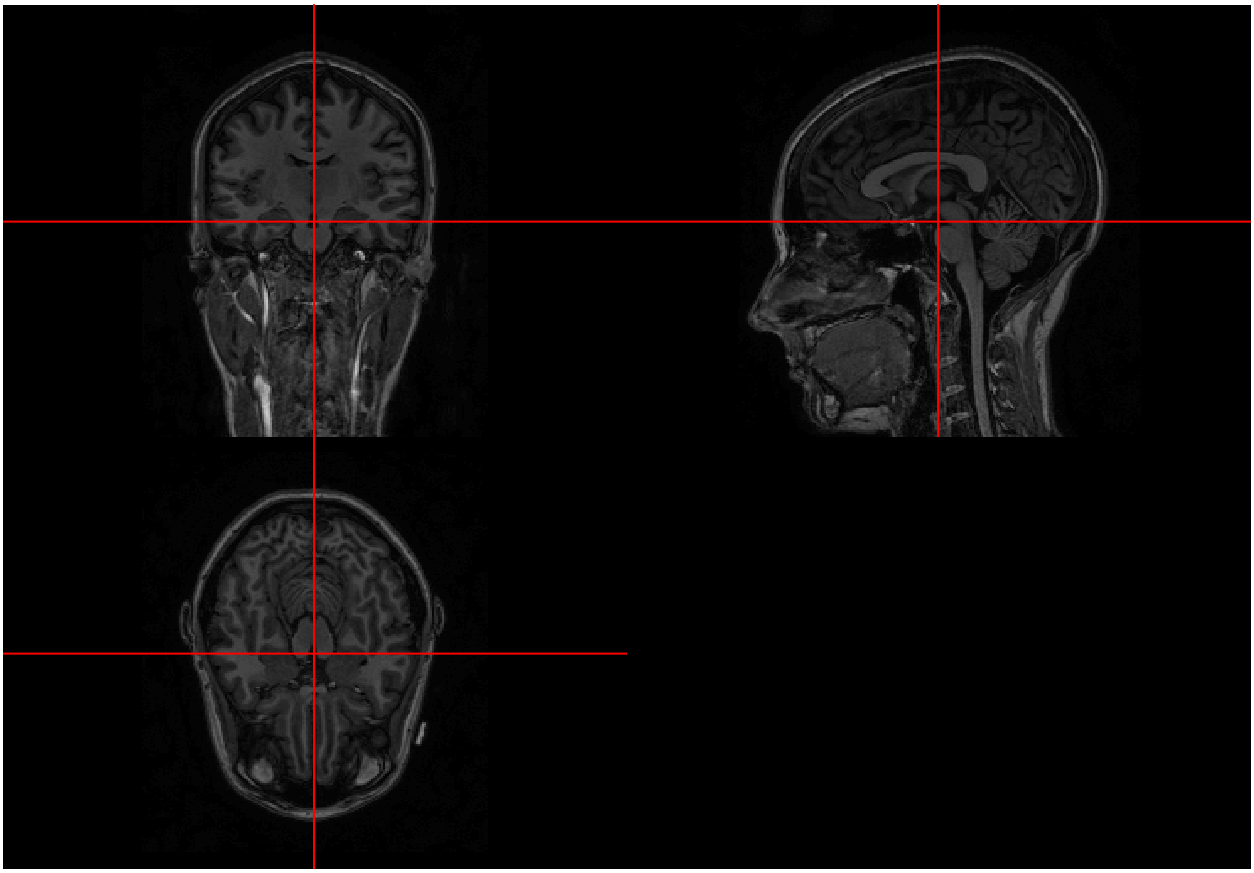
```
fast_img = fsl_biascorrect(nim, retimg=TRUE)
```

```
## FSLDIR='/usr/share/fsl/5.0'; PATH=${FSLDIR}/bin:${PATH};export PATH FSLDIR; sh "${FSLDIR}/etc/fslconf
```

```
orthographic(nim)
```

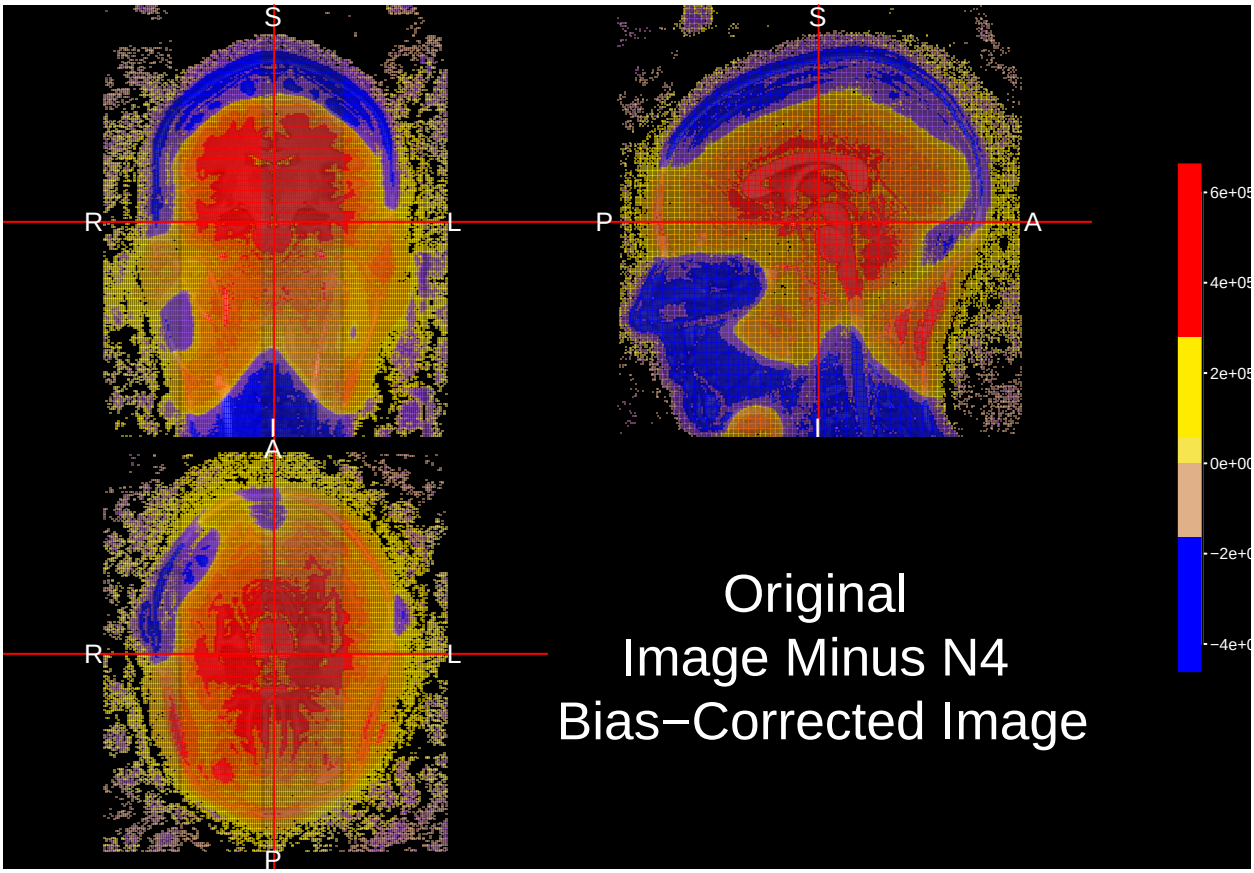


```
orthographic(fast_img)
```



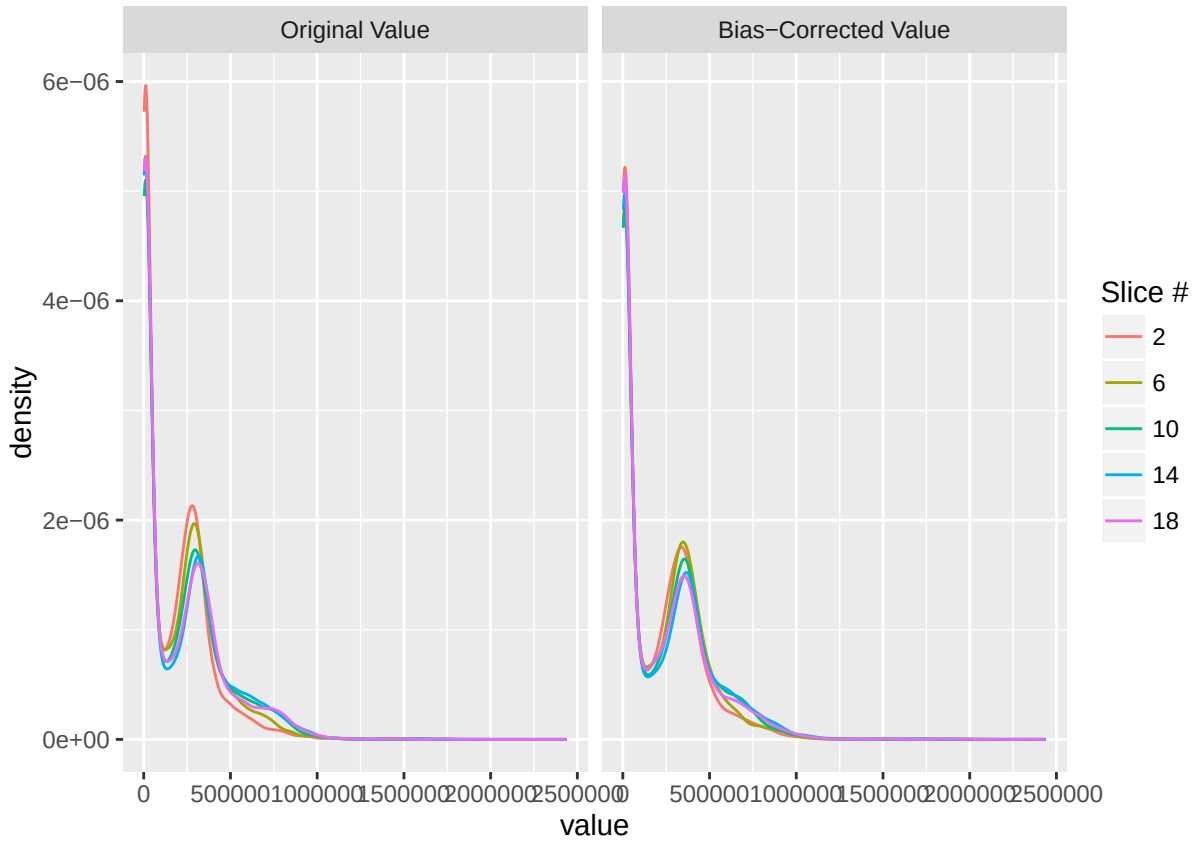
Original Image Minus N4 Bias-Corrected Image

```
library(scales)
sub.bias <- niftiarr(nim, nim-fast_img)
# quantile the difference image using these as breaks
q=quantile(sub.bias[sub.bias !=0],probs = seq(0,1,by=0.1))

# get a diverging gradient palette
fcol=div_gradient_pal(low="blue",mid="yellow",high ="red")
ortho2(nim,sub.bias,col.y = alpha(fcol(seq(0,1, length=10)),
0.5), ybreaks = q, ycolorbar=TRUE, text = paste0("Original
Image Minus N4", "\n Bias-Corrected Image"))
```



Compare using Histogram

```
library(ggplot2)
library(reshape2)

slices = c(2, 6, 10, 14, 18)
 vals = lapply(slices, function(x) {
 cbind(img = c(nim[,,x]), fast = c(fast_img[,,x]),
 slice = x)
 })
 vals = do.call("rbind", vals)
 vals = data.frame(vals)
 vals = vals[ vals$img > 0 & vals$fast > 0, ]
 colnames(vals)[1:2] = c("Original Value", "Bias-Corrected Value")
 v = melt(vals, id.vars = "slice")
 g = ggplot(aes(x = value, colour = factor(slice)), data = v) + geom_line(stat = "density") + facet_wra
 g = g + scale_colour_discrete(name = "Slice #")
 print(g)
```

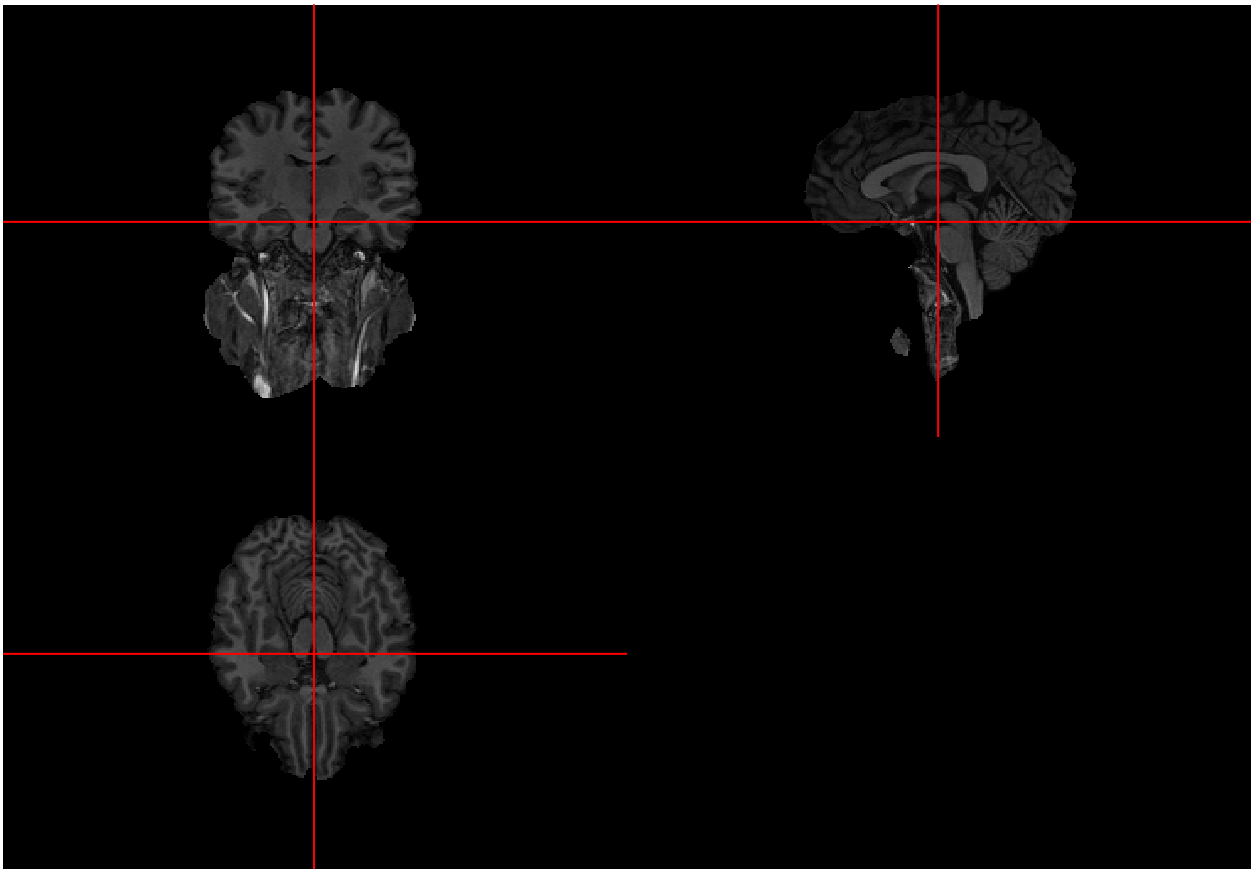


Brain Extraction Using fslr

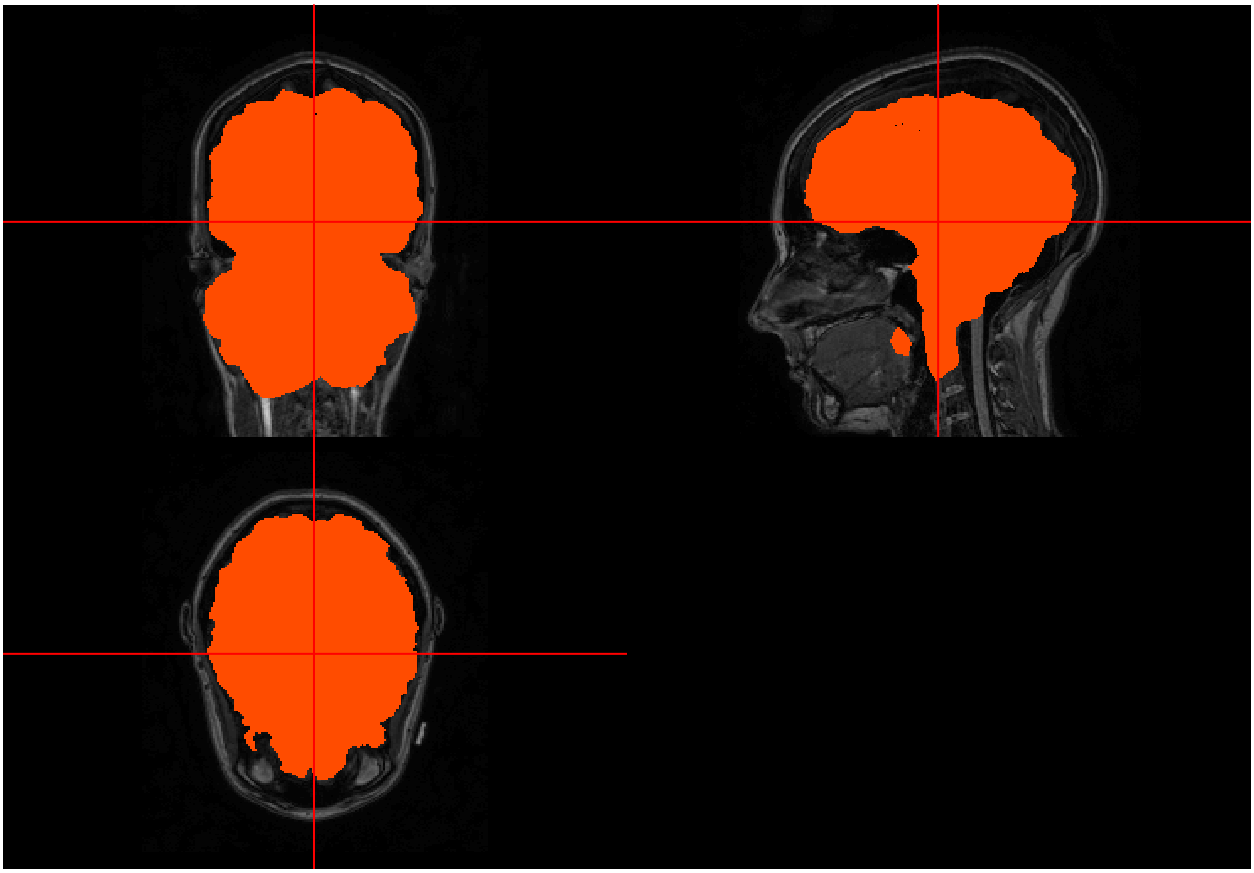
```
bet_fast = fslbet(infile=fast_img, retimg=TRUE)
```

```
## FSLDIR='/usr/share/fsl/5.0'; PATH=${FSLDIR}/bin:${PATH};export PATH FSLDIR; sh "${FSLDIR}/etc/fslcon
```

```
bet_fast_mask <- niftiarr(bet_fast, 1)
is_in_mask = bet_fast>0
bet_fast_mask[!is_in_mask]<-NA
orthographic(bet_fast)
```



```
orthographic(fast_img,bet_fast_mask)
```



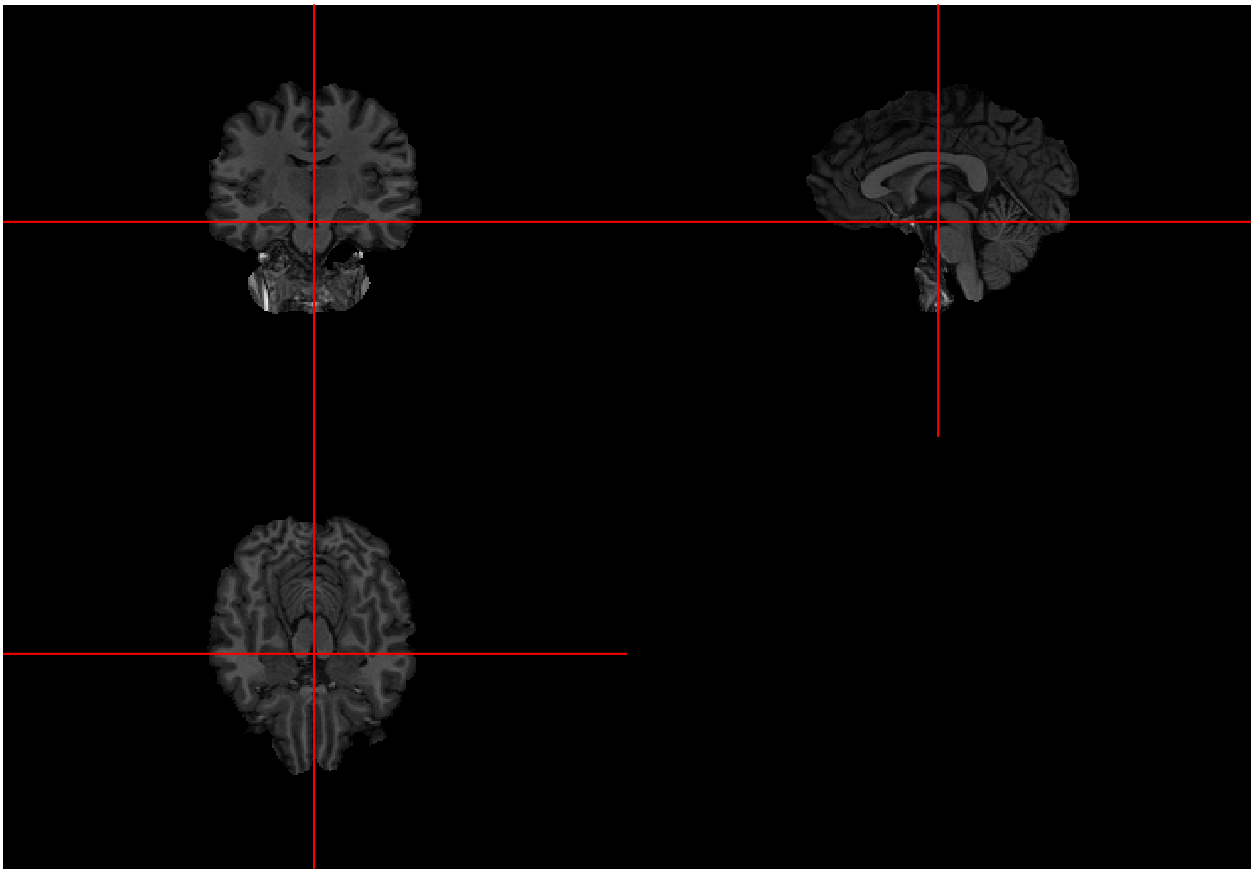
FSL's Brain Extraction Tool (BET) can be used for skull stripping BET: fast, robust, and popular

Improving Brain Segmentation

```
cog = cog(bet_fast, ceil=TRUE)
cog = paste("-c", paste(cog, collapse= " "))
bet_fast2 = fslbet(infile=fast_img,retimg=TRUE,opts=cog)
```

```
## FSLDIR='/usr/share/fsl/5.0'; PATH=${FSLDIR}/bin:${PATH};export PATH FSLDIR; sh "${FSLDIR}/etc/fslcon
```

```
orthographic(bet_fast2)
```



```
dim(bet_fast2)
```

```
## [1] 170 256 256
```

In Summary

With FSL easy to pre-processing Neuro image and perform some task as Brain Extraction .

Full genarated file can see here : http://rpubs.com/ngocbd/Pre-Processing-Neurohacking