

Problemas con el procesamiento de FBB y como no morir en el intento

de DICOM a NiFTI

Necesitamos convertir los DICOM de FBB a formato NiFTI-1 para trabajar con FSL. El servidor DICOM corta el numero de slices por directorio a 500 por lo que inicialmente hubo muchisimos problemas de conversion y procesamiento.

Luego hay que buscar por todas las subcarpetas del DICOM las imagenes 4x5min que estan desordenadas. Para convertir uno de los sujetos ha de hacerse algo asi,

```
$ for x in `find /nas/raw_images/facehbi/fbb/FACEHBI-F001B/DICOM/ -type f`;
do if [[ `dckey -k "SeriesDescription" $x 2>&1 | grep "4x5min"` ]]; then cp
$x /nas/facehbi/tmp_2nifti/; fi; done; y=$(ls /nas/facehbi/tmp_2nifti/ |
head -n 1); dcm2nii -o /nas/facehbi/tmp/ /nas/facehbi/tmp_2nifti/$y;
```

que da una salida muy ruidosa, pero funciona. Para hacerlos todos de una tacada habria que iterar esto por todos los sujetos.

```
for s in `ls -d /nas/clinic/* | grep FACEHBI | grep "B$"`; do sbj=$(echo $s
| sed 's/.*-F(.*)B$/\1/'); for x in `find ${s}/DICOM/ -type f`; do if [[
`dckey -k "SeriesDescription" $x 2>&1 | grep "4x5min"` ]]; then cp $x
/nas/facehbi/tmp_2nifti/; fi; done; y=$(ls /nas/facehbi/tmp_2nifti/ | head -
n 1); dcm2nii -o /nas/facehbi/tmp/ /nas/facehbi/tmp_2nifti/$y; conv=$(ls
/nas/facehbi/tmp/ | head -n 1); fslsplit /nas/facehbi/tmp/${conv}
/nas/facehbi/fbb_first/smc0${sbj}s -t; rm -rf /nas/facehbi/tmp/*; rm -rf
/nas/facehbi/tmp_2nifti/*; done
```

si queremos aprovechar un archivo yet.txt ([ver aqui](#))

```
$ for sbj in `awk -F";" '{print $1}' yet.txt | sed 's/0//;`; do
s="/nas/clinic/FACEHBI-F${sbj}B"; for x in `find ${s}/DICOM/ -type f`; do
if [[ `dckey -k "SeriesDescription" $x 2>&1 | grep "4x5min"` ]]; then cp $x
/nas/facehbi/tmp_2nifti/; fi; done; y=$(ls /nas/facehbi/tmp_2nifti/ | head -
n 1); dcm2nii -o /nas/facehbi/tmp/ /nas/facehbi/tmp_2nifti/$y; conv=$(ls
/nas/facehbi/tmp/ | head -n 1); fslsplit /nas/facehbi/tmp/${conv}
/nas/facehbi/fbb/smc0${sbj}s -t; rm -rf /nas/facehbi/tmp/*; rm -rf
/nas/facehbi/tmp_2nifti/*; done
```

Correccion de movimiento

Ahora tenemos 4 archivos representando la integracion de 5 min y hay que corregirlos al espacio de usuario. Lo primero es traer el archivo de freesurfer y luego corregir cada uno de los fbb al user space. despues se ha de unirlos temporalmente y hacer un mcflirt.

lo primero seria traerse la mri del directorio de freesurfer

[get_fs_subj](#)

[get_fs_subj.sh](#)

```
#!/bin/sh
study=$1
shift

id=$1
shift

dir=$1
shift

debug=0

#First get the freesurfer processed MRIs
${FREESURFER_HOME}/bin/mri_vol2vol --mov
${SUBJECTS_DIR}/${study}_${id}/mri/nu.mgz --targ
${SUBJECTS_DIR}/${study}_${id}/mri/rawavg.mgz --regheader --o
${dir}/${id}_tmp_nu_in_rawavg.mgz
${FREESURFER_HOME}/bin/mri_convert --in_type mgz --out_type nii
${dir}/${id}_tmp_nu_in_rawavg.mgz ${dir}/${id}_tmp.nii.gz
${FSLDIR}/bin/fslreorient2std ${dir}/${id}_tmp ${dir}/${id}_struc
${FREESURFER_HOME}/bin/mri_vol2vol --mov
${SUBJECTS_DIR}/${study}_${id}/mri/brain.mgz --targ
${SUBJECTS_DIR}/${study}_${id}/mri/rawavg.mgz --regheader --o
${dir}/${id}_tmp_brain_in_rawavg.mgz
${FREESURFER_HOME}/bin/mri_convert --in_type mgz --out_type nii
${dir}/${id}_tmp_brain_in_rawavg.mgz ${dir}/${id}_tmp_brain.nii.gz
${FSLDIR}/bin/fslreorient2std ${dir}/${id}_tmp_brain ${dir}/${id}_brain

if [ $debug = 0 ] ; then
    rm ${dir}/${id}_tmp*
fi
```

y luego registrar cada imagen al espacio del sujeto. Hay 4 variantes para esto.

1.- Intentando registrar cada imagen independientemente.

[fbb_reg](#)

[fbbtemp_reg.sh](#)

```
#!/bin/sh
```

```

study=$1
shift

id=$1
shift

tdir=$1
shift

wdir=$1
shift

debug=1

#get the uncorrected PETs and register to user space MRI
for i in {0..3}; do
    tf=`printf "${id}s%04d" $i`
    ${FSLDIR}/bin/imcp ${tdir}/${tf} ${tdir}/${id}_tmp
    ${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in
${tdir}/${id}_tmp -omat ${tdir}/${tf}_pet2struc.mat -out
${tdir}/${tf}_reg
done
a=`for i in {0..3}; do printf " ${tdir}/${id}s%04d_reg " $i; done`
${FSLDIR}/bin/fslmerge -t ${wdir}/${id}_tmp_mvc $a
${FSLDIR}/bin/mcflirt -in ${wdir}/${id}_tmp_mvc -out
${wdir}/${id}_tmp_corr
${PIPEDIR}/bin/4dmean.pl ${wdir}/${id}_tmp_corr
${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in ${wdir}/${id}_mean -
omat ${wdir}/${id}_fbb2struc.mat -out ${wdir}/${id}_fbb

if [ $debug = 0 ] ; then
    rm ${tdir}/${id}_tmp*
    rm ${wdir}/${id}_tmp*
fi

```

2.- Usando la informacion de un registro para el resto

[fbb_regc](#)

[fbbtemp_regc.sh](#)

```

#!/bin/sh

study=$1
shift

id=$1
shift

```

```

tdir=$1
shift

wdir=$1
shift

sok=$1
shift

debug=1

#Now get the uncorrected PETs and register to user space MRI
bsc=`printf "${id}s%04d" $sok`
${FSLDIR}/bin/imcp ${tdir}/${bsc} ${tdir}/${id}_tmp
${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in ${tdir}/${id}_tmp -
omat ${tdir}/${bsc}_pet2struc.mat -out ${tdir}/${bsc}_reg
for i in {0..3}; do
    if [ "$i" != "$sok" ]; then
        tf=`printf "${id}s%04d" $i`
        ${FSLDIR}/bin/imcp ${tdir}/${tf} ${tdir}/${id}_tmp
        ${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in
${tdir}/${id}_tmp -init ${tdir}/${bsc}_pet2struc.mat -omat
${tdir}/${tf}_pet2struc.mat -out ${tdir}/${tf}_reg
    fi
done
a=`for i in {0..3}; do printf " ${tdir}/${id}s%04d_reg " $i; done`
${FSLDIR}/bin/fslmerge -t ${wdir}/${id}_tmp_mvc $a
${FSLDIR}/bin/mcflirt -in ${wdir}/${id}_tmp_mvc -out
${wdir}/${id}_tmp_corr
${PIPEDIR}/bin/4dmean.pl ${wdir}/${id}_tmp_corr
${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in ${wdir}/${id}_mean -
omat ${wdir}/${id}_fbf2struc.mat -out ${wdir}/${id}_fbf

if [ $debug = 0 ] ; then
    rm ${tdir}/${id}_tmp*
    rm ${wdir}/${id}_tmp*
fi

```

3.- Usando la informacion solo del cerebro extraido para hacer el corregistro

[fbf_regb](#)

[fbftemp_regb.sh](#)

```

#!/bin/sh

study=$1

```

```

shift

id=$1
shift

tmdir=$1
shift

wdir=$1
shift

sok=$1
shift

debug=1

#Now get the uncorrected PETs and register to user space MRI
bsc=`printf "${id}s%04d" $sok`
${FSLDIR}/bin/imcp ${tmdir}/${bsc} ${tmdir}/${id}_tmp
${FSLDIR}/bin/flirt -ref ${wdir}/${id}_brain -in ${tmdir}/${id}_tmp -
omat ${tmdir}/${bsc}_pet2struc.mat -out ${tmdir}/${bsc}_reg
for i in {0..3}; do
    if [ "$i" != "$sok" ]; then
        tf=`printf "${id}s%04d" $i`
        ${FSLDIR}/bin/imcp ${tmdir}/${tf} ${tmdir}/${id}_tmp
        ${FSLDIR}/bin/flirt -ref ${wdir}/${id}_brain -in
${tmdir}/${id}_tmp -init ${tmdir}/${bsc}_pet2struc.mat -omat
${tmdir}/${tf}_pet2struc.mat -out ${tmdir}/${tf}_reg
    fi
done
a=`for i in {0..3}; do printf " ${tmdir}/${id}s%04d_reg " $i; done`
${FSLDIR}/bin/fslmerge -t ${wdir}/${id}_tmp_mvc $a
${FSLDIR}/bin/mcflirt -in ${wdir}/${id}_tmp_mvc -out
${wdir}/${id}_tmp_corr
${PIPEDIR}/bin/4dmean.pl ${wdir}/${id}_tmp_corr
${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in ${wdir}/${id}_mean -
omat ${wdir}/${id}_fbb2struc.mat -out ${wdir}/${id}_fbb

if [ $debug = 0 ] ; then
    rm ${tmdir}/${id}_tmp*
    rm ${wdir}/${id}_tmp*
fi

```

4.- el mas complicado, usando una mascara. Este metodo se probó con el [registro de PiBs](#) y usa un umbral de intensidad

[fbb_regm](#)

[fbbtemp_regm.sh](#)

```
#!/bin/sh

study=$1
shift

id=$1
shift

tdir=$1
shift

wdir=$1
shift

#sok=$1
#shift

treshold=$1
shift

debug=1

#Now get the uncorrected PETs and register to user space MRI
bsc=`printf "${id}s%04d" $sok`
${FSLDIR}/bin/fslreorient2std ${tdir}/${bsc} ${tdir}/${bsc}_tmp
if [ $treshold = 0 ] ; then
    ${FSLDIR}/bin/fslmaths ${tdir}/${bsc}_tmp -thr $treshold -bin
    ${tdir}/${bsc}_tmp_fbb_mask
else
    ${FSLDIR}/bin/fslmaths ${tdir}/${bsc}_tmp -thrP $treshold -
    bin ${tdir}/${bsc}_tmp_fbb_mask
fi
${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in
${tdir}/${bsc}_tmp_fbb_mask -omat ${tdir}/${bsc}_tmp_fbb -dof 9
${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in ${tdir}/${bsc} -
applyx_fm -init ${tdir}/${bsc}_tmp_fbb -out ${tdir}/${bsc}_reg -omat
${tdir}/${bsc}_pet2struc.mat

for i in {0..3}; do
    if [ "$i" != "$sok" ]; then
        tf=`printf "${id}s%04d" $i`
        ${FSLDIR}/bin/fslreorient2std ${tdir}/${tf}
        ${tdir}/${id}_tmp
        if [ $treshold = 0 ] ; then
            ${FSLDIR}/bin/fslmaths ${tdir}/${id}_tmp -thr
            $treshold -bin ${tdir}/${tf}_tmp_fbb_mask
        else
            ${FSLDIR}/bin/fslmaths ${tdir}/${id}_tmp -thrP
            $treshold -bin ${tdir}/${tf}_tmp_fbb_mask
```

```

        fi
        ${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in
${tdir}/${tf}_tmp_fbb_mask -omat ${tdir}/${tf}_tmp_fbb -init
${tdir}/${bsc}_pet2struc.mat -dof 9
        ${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in
${tdir}/${tf} -applyxfm -init ${tdir}/${tf}_tmp_fbb -out
${tdir}/${tf}_reg
    fi
done
a=`for i in {0..3}; do printf " ${tdir}/${id}s%04d_reg " $i; done`
${FSLDIR}/bin/fslmerge -t ${wdir}/${id}_tmp_mvc $a
${FSLDIR}/bin/mcflirt -in ${wdir}/${id}_tmp_mvc -out
${wdir}/${id}_tmp_corr
${PIPEDIR}/bin/4dmean.pl ${wdir}/${id}_tmp_corr
${FSLDIR}/bin/flirt -ref ${wdir}/${id}_struc -in ${wdir}/${id}_mean -
omat ${wdir}/${id}_fbb2struc.mat -out ${wdir}/${id}_fbb

if [ $debug = 0 ] ; then
    rm ${tdir}/${id}_tmp*
    rm ${wdir}/${id}_tmp*
fi

```

Esto lo he intentado agrupar en el script *fbb_correct.pl*,

[fbb_correct.pl \(chunk\)](#)

```

my $study;
my $cfile;
my $wbrain = 0;
my $wguide = 0;
my $wmask = 0;
my $sok = 0;

@ARGV = ("-h") unless @ARGV;
while (@ARGV and $ARGV[0] =~ /^-/) {
    $_ = shift;
    last if /^--$/;
    if (/^-e/) { $study = shift; chomp($study);}
    if (/^-cut/) { $cfile = shift; chomp($cfile);}
    if (/^-useg/) { $sok = shift; $wguide = 1;}
    if (/^-useb/) { $sok = shift; $wbrain = 1;}
    if (/^-usem/) { $sok = shift; $wmask = 1;}
    if (/^-h/) { print_help $ENV{'PIPEDIR'}.'/doc/fbb_reg.hlp'; exit;}
}

```

y despues,

```

foreach my $subject (@ok_pets){

```

```

    $pm->start and next;
    my $order;
    if($wmask){
        $order = "fbftemp_regm.sh ".$study."
".$pets{$subject}.$subject." ".$petnc_dir." ".$w_dir." ".$sok;
    }elseif($wguide){
        $order = "fbftemp_regc.sh ".$study."
".$pets{$subject}.$subject." ".$petnc_dir." ".$w_dir." ".$sok;
    }elseif($wbrain){
        $order = "fbftemp_regb.sh ".$study."
".$pets{$subject}.$subject." ".$petnc_dir." ".$w_dir." ".$sok;
    }else{
        $order = "fbftemp_reg.sh ".$study."
".$pets{$subject}.$subject." ".$petnc_dir." ".$w_dir;
    }
    print "$order\n";
    system($order);
    $pm->finish;
}
$pm->wait_all_children;

```

Resumiendo

si todo va bien basta con hacer:

```

$ fbb_correct.pl -e facehbi
$ parallel_fbb_rois_metrics.pl facehbi

```

Postprocessing (carrot)

```

$ cd /home/osotolongo/Documents/ACE/facehbi
$ R CMD BATCH selected_mri.r
$ sed 's//g' facehbi_mri_selected.csv | awk -F"," '{if($1!="Subject")
printf("%04d;%s;%s;%s;%s;%s;%s\n",$1,$2,$3,$4,$5,$6,$7); else
printf("%s;%s;%s;%s;%s;%s;%s\n",$1,$2,$3,$4,$5,$6,$7)}' >
facehbi_mri_selected_reformatted.csv
$ join -t";" -j 1 facehbi_fbb_fs_suvr_predef.csv
facehbi_mri_selected_reformatted.csv > facehbi_fbb_mri.csv
$ sed 's// /g' facehbi_fbb_mri.csv > facehbi_fbb_mri.dat

```

From:
<http://imagen.fundacioace.com/wiki/> - Detritus Wiki

Permanent link:
<http://imagen.fundacioace.com/wiki/doku.php?id=neuroimagen:fbf&rev=1443691222>

Last update: **2020/08/04 10:47**

