

Preprocessing manual with fmriprep

Claire Zhang, 23/09/2024

In this manual, all commands calling modules are based on the setup of HPC cluster at DCCN. If you are using a different server, you need to install and load bidscoin and fmriprep accordingly.

In general, we will preprocess fmri data in 2 versions.

First, with fmriprep 24.0.0, we will not do ICAroma, but include fieldmap correction.

Second, with fmriprep 20.2.1, we will conduct ICAroma, but ignore fieldmap correction.

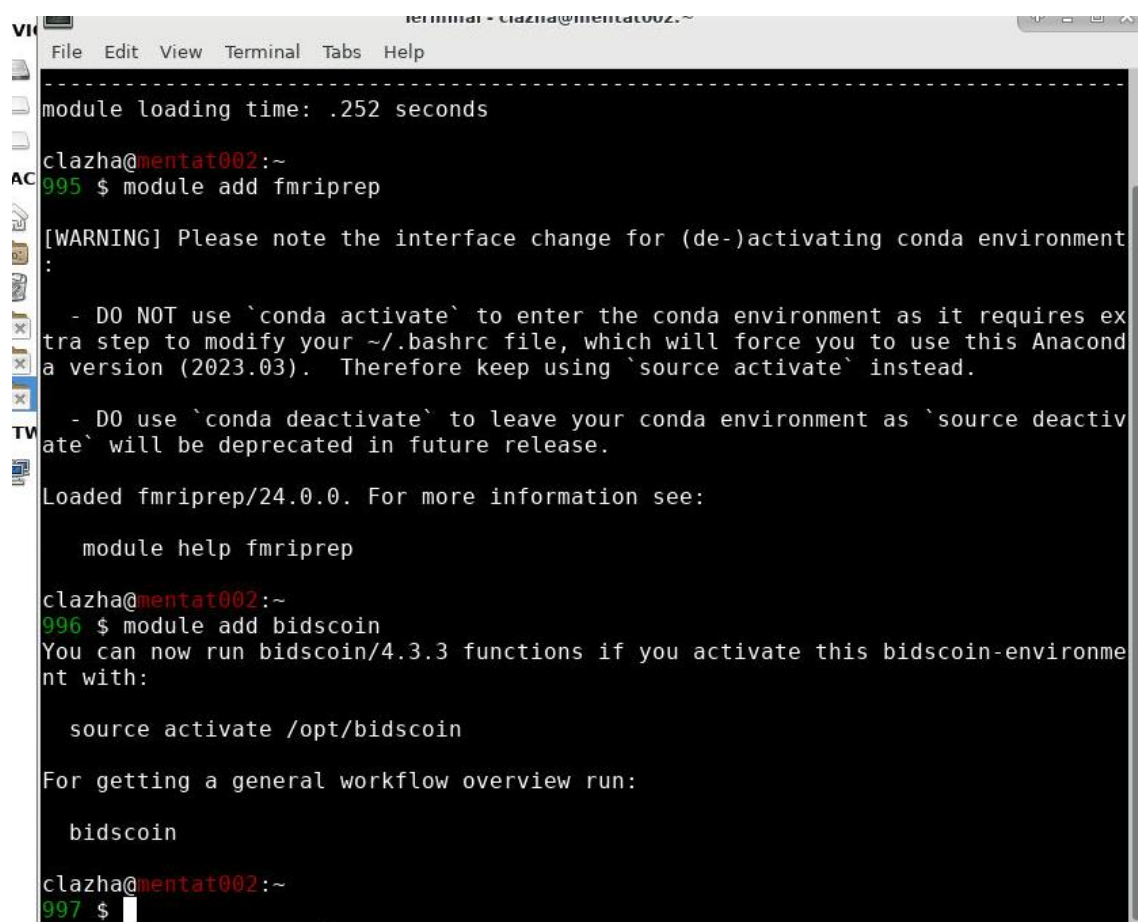
Fmriprep 24.0.0

1. Load fmriprep/24.0.0 & bidscoin

Codes:

```
module add fmriprep
module add bidscoin
source activate /opt/bidscoin
```

Note: fmriprep 24.0.0 at HPC cluster is the default.

A screenshot of a terminal window titled 'terminal - clazha@mentat002'. The terminal shows the following sequence of commands and outputs:
- Initial output: 'module loading time: .252 seconds'
- Command: 'clazha@mentat002:~
995 \$ module add fmriprep'
- Output: '[WARNING] Please note the interface change for (de-)activating conda environment :
- DO NOT use `conda activate` to enter the conda environment as it requires extra step to modify your ~/.bashrc file, which will force you to use this Anaconda version (2023.03). Therefore keep using `source activate` instead.
- DO use `conda deactivate` to leave your conda environment as `source deactivate` will be deprecated in future release.'
- Output: 'Loaded fmriprep/24.0.0. For more information see:
module help fmriprep'
- Command: 'clazha@mentat002:~
996 \$ module add bidscoin'
- Output: 'You can now run bidscoin/4.3.3 functions if you activate this bidscoin-environment with:
source activate /opt/bidscoin'
- Output: 'For getting a general workflow overview run:
bidscoin'
- Command: 'clazha@mentat002:~
997 \$'

```

clazha@mentat002:~
1000 $ module add bidscoin
You can now run bidscoin/4.3.3 functions if you activate this bidscoin-environment with:

    source activate /opt/bidscoin.

For getting a general workflow overview run:

    bidscoin

clazha@mentat002:~
1001 $ source activate /opt/bidscoin
(/opt/bidscoin) clazha@mentat002:~
1002 $

```

2. Create bidsmap with the command bidsmapper

Codes:

```
bidsmapper /project/***/sourcedata /project/***/bids
```

Note: first entry, should be the source data; second entry, should be the bids folder where you want to put your niift files.

```

clazha@mentat002:~
1001 $ source activate /opt/bidscoin
(/opt/bidscoin) clazha@mentat002:~
1002 $ bidsmapper /project/2424007.01/dutch_data/sourcedata /project/2424007.01/dutch_data/bids
INFO |
INFO | ----- START BIDSmapper -----
INFO | >>> bidsmapper sourcefolder=/project/2424007.01/dutch_data/sourcedata bidsfolder=/project/2424007.01/dutch_data/bids bidsmap=bidsmap.yaml template=/home/predatt/clazha/.bidscoin/4.3.3+qt5/templates/bidsmap_dccn.yaml plugins=[] subprefix=None sesprefix=None store=False force=False
INFO | No existing bidsmap file found: /project/2424007.01/dutch_data/bids/code/bidscoin/bidsmap.yaml
INFO | Reading: /home/predatt/clazha/.bidscoin/4.3.3+qt5/templates/bidsmap_dccn.yaml
INFO | Checking the bidsmap run-items:
SUCCESS | All datatypes and options in the template bidsmap are valid
INFO | Mapping: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01 (subject 1/40)
VERBOSE | Executing plugin: dcm2niix2bids -> /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01
INFO | Discovered 'exclude' DICOM sample: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/001-localizer/00001_1.3.12.2.1107.5.2.43.66068.2022020710282395193396288.IMA
INFO | Discovered 'exclude' DICOM sample: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/002-AAHead_Scout_32ch-head-coil/00001_1.3.12.2.1107.5.2.43.66068.2022020710284447724296575.IMA
INFO | Discovered 'exclude' DICOM sample: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/003-AAHead_Scout_32ch-head-coil_MPR_sag/00001_1.3.12.2.1107.5.2.43.66068.2022020710284813391497597.IMA
INFO | Discovered 'exclude' DICOM sample: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/004-AAHead_Scout_32ch-head-coil_MPR_cor/00001_1.3.12.2.1107.5.2.43.66068.2022020710284813391497597.IMA

```

3. Edit bidsmap

Wait for a few seconds, then you will see this GUI. Now you can edit your json files in this GUI.

For fieldmap scans, magnitude 1 and 2:

The screenshot shows the 'Edit BIDS mapping' window with the following data:

DICOM input	
Properties	
filepath	/project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/006-...
filename	00001_1.3.12.2.1107.5.2.43.66068.2022020710315329704998053.IMA
filesize	164.94 kB
nrfiles	120
Attributes	
ProtocolName	field_map_2p4iso
EchoNumbers	1
ImageType	['ORIGINAL', 'PRIMARY', 'M', 'ND']
Modality	MR
SeriesDescription	field_map_2p4iso
SequenceName	.fm2d2
PulseSequenceName	
SequenceVariant	SP
ScanningSequence	GR
EchoPulseSequence	
MRAcquisitionType	2D
SliceThickness	2.4000000953674
FlipAngle	40
EchoTime	2.2
EffectiveEchoTime	
RepetitionTime	410
PhaseEncodingDirection	AP

BIDS output	
Data type	
fmap	
acq	
run	<<>>
chunk	
suffix	magnitude1
Data filename	
fmap/sub-11_ses-mri01_magnitude1.*	
Meta data	
B0FieldIdentifier	fieldmap
IntendedFor	

Note: Sometimes, bidsmapper only recognize 1 magnitude file. That is ok. When you run bidscoiner, it would extract magnitude 2 files as well. This is because at DCCN, the scanner is set up to put magnitude 1 and 2 original IMA files in the same folder.

For fieldmap scans, phase difference:

The screenshot shows the 'Edit BIDS mapping' window with the following data:

DICOM input	
Properties	
filepath	/project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/007-...
filename	00001_1.3.12.2.1107.5.2.43.66068.2022020710315373908698300.IMA
filesize	164.98 kB
nrfiles	60
Attributes	
ProtocolName	field_map_2p4iso
EchoNumbers	2
ImageType	['ORIGINAL', 'PRIMARY', 'P', 'ND']
Modality	MR
SeriesDescription	field_map_2p4iso
SequenceName	.fm2d2
PulseSequenceName	
SequenceVariant	SP
ScanningSequence	GR
EchoPulseSequence	
MRAcquisitionType	2D
SliceThickness	2.4000000953674
FlipAngle	40
EchoTime	4.66
EffectiveEchoTime	
RepetitionTime	410
PhaseEncodingDirection	AP

BIDS output	
Data type	
fmap	
acq	
run	<<>>
chunk	
suffix	phasediff
Data filename	
fmap/sub-11_ses-mri01_phasediff.*	
Meta data	
B0FieldIdentifier	fieldmap
IntendedFor	

For functional scans, single band:

DICOM input

Properties

- filepath: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/008-...
- filename: 00001_1.3.12.2.1107.5.2.43.66068.2022020710323526940499880.IMA
- filesize: 1.11 MB
- nrfiles: 1

Attributes

- SeriesDescription: Block1_SBRef
- Modality: MR
- ProtocolName: Block1
- ImageType: ['ORIGINAL', 'PRIMARY', 'M', 'ND', 'NORM', 'MOSAIC']
- SequenceName: epfid2d1_88
- PulseSequenceName: epfid2d1_88
- SequenceVariant: ['SK', 'SS']
- ScanningSequence: EP
- EchoPulseSequence: EP
- MRAcquisitionType: 2D
- SliceThickness: 2.4000000953674
- FlipAngle: 65
- EchoNumbers: 1
- EchoTime: 34.2
- EffectiveEchoTime: 34.2
- RepetitionTime: 1200
- PhaseEncodingDirection: AP

BIDS output

Data type

func

task: learn

acq: ce

rec: ce

dir: <<01>>

run: <<01>>

echo: <<01>>

part: <<01>>

chunk: <<01>>

suffix: sbref

Data filename

func/sub-11_ses-mri01_task-learn_run-01_sbref.*

Meta data

TaskName	learn
B0FieldSource	fieldmap
B0FieldIdentifier	

Buttons: Export, Cancel, OK, Validate, Save

For functional scans, bold:

DICOM input

Properties

- filepath: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/009-...
- filename: 00001_1.3.12.2.1107.5.2.43.66068.2022020710323559465400018.IMA
- filesize: 1.11 MB
- nrfiles: 520

Attributes

- Modality: MR
- ProtocolName: Block1
- SeriesDescription: Block1
- ImageType: ['ORIGINAL', 'PRIMARY', 'M', 'MB', 'ND', 'NORM', 'MOSAIC']
- SequenceName: epfid2d1_88
- PulseSequenceName: epfid2d1_88
- SequenceVariant: ['SK', 'SS']
- ScanningSequence: EP
- EchoPulseSequence: EP
- MRAcquisitionType: 2D
- SliceThickness: 2.4000000953674
- FlipAngle: 65
- EchoNumbers: 1
- EchoTime: 34.2
- EffectiveEchoTime: 34.2
- RepetitionTime: 1200
- PhaseEncodingDirection: AP

BIDS output

Data type

func

task: learn

acq: ce

rec: ce

dir: <<01>>

run: <<01>>

echo: <<01>>

part: <<01>>

chunk: <<01>>

suffix: bold

Data filename

func/sub-11_ses-mri01_task-learn_run-01_bold.*

Meta data

TaskName	learn
B0FieldSource	fieldmap
B0FieldIdentifier	

Buttons: Export, Cancel, OK, Validate, Save

IMPORTANT:

Put a key in B0FieldIdentifier in fieldmap scans. Put this same key in B0FieldSource in functional scans, both single band and bold files.

Here, the key that used “*fieldmap*”. This key should be the same for the magnitude 1, magnitude 2, and phase difference files for the same fieldmap. You can use anything, as long as they match with each other and it is a unique string.

If you have several fieldmaps, and you want to apply different fieldmaps to different functional scans, then you should put different keys for different fieldmaps. As long as the Source and the Identifier match with each other, fmriprep will apply them accordingly.

If you have multiple sessions, add the dynamic value “_<<session>>” to your key (e.g. if the key is “fieldmap”, it should be “fieldmap_<<session>>”).

This is crucial. Otherwise, fmriprep will not run fieldmap correction.

```
File Edit View Terminal Help
[3] Van Der Walt, S., Colbert, S.C. & Varoquaux, G., 2011. The NumPy array: a structure for efficient numerical co
mputation. Computing in Science & Engineering, 13(2), pp.22–30.
(/opt/bidscoin) clazha@mentat002:~
1003 $ bidscoiner /project/2424007.01/dutch_data/sourcedata /project/2424007.01/dutch_data/bids
INFO |
INFO | ----- START BIDScoiner 4.3.3+qt5: BIDS 1.9.0 -----
INFO | >>> bidscoiner sourcefolder=/project/2424007.01/dutch_data/sourcedata bidsfolder=/project/2424007.01/dutch_
data/bids participant=None force=False bidsmap=bidsmap.yaml
INFO | Creating dataset description file: /project/2424007.01/dutch_data/bids/dataset_description.json
INFO | Creating a template README file (adjust it to your needs): /project/2424007.01/dutch_data/bids/README
INFO | Reading: /project/2424007.01/dutch_data/bids/code/bidscoin/bidsmap.yaml
INFO | Checking the bidsmap run-items:
SUCCESS | All run-items in the bidsmap are valid
INFO | ----- Subject 1/40 -----
INFO | >>> Coining datasources in: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01
VERBOSE | Executing plugin: dcm2niix2bids
INFO | --> Leaving out: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/001-localizer
INFO | --> Leaving out: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/002-AAHead_Scout_32ch-head-coil
INFO | --> Leaving out: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/003-AAHead_Scout_32ch-head-coil
INFO | --> Leaving out: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/004-AAHead_Scout_32ch-head-coil
INFO | --> Leaving out: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/005-AAHead_Scout_32ch-head-coil
INFO | --> Leaving out: /project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/006-field_map_2p4iso
VERBOSE | Command:
module add dcm2niix; dcm2niix -b y -z y -i n -l n -f "sub-11_ses-mri01_magnitude1" -o "/project/2424007.01/dutch_d
ata/bids/sub-11/ses-mri01/fmap" "/project/2424007.01/dutch_data/sourcedata/sub-11/ses-mri01/006-field_map_2p4iso"
VERBOSE | Output:
Chris Rorden's dcm2niix version v1.0.20240202 (JP2:OpenJPEG) (JP-LS:CharLS) GCC8.4.0 x86-64 (64-bit Linux)
Found 120 DICOM file(s)
Slices not stacked: echo varies (TE 2.2, 4.66; echo 1, 2). Use 'merge 2D slices' option to force stacking
convert 60 DICOM as /project/2424007.01/dutch_data/bids/sub-11/ses-mri01/fmap/sub-11_ses-mri01_magnitude1_e1 (88x8
```

After editing, you should click on save at GUI of bidsmapper at the default folder it recommends.

Usually, this map would be in this directory: /project/***/bids/code/bidscoin/. Then, close the GUI even if you see “Waiting for bidsmapper to process”. There is no notice of saying bidsmapper is saved...

4. Run bidscoiner

Codes:

```
bidscoiner /project/***/sourcedata /project/***/bids
```

Note: These entries should be the same as your bidsmapper command.

If you have multiple sessions, bidscoiner might not remove the arrow brackets from the fieldmap key stored in the json files.

Here’s a bash script that you can use to edit all json files (fmap, and func) and remove arrow brackets from B0FieldIdentifier and B0FieldSource tags:

```
#!/bin/bash
```

```
# Base directory to search for JSON files
```

```
BASE_DIR="/project/***/bids/"
```

```

# Find all .json files in subdirectories
find "$BASE_DIR" -type f -name "*.json" | while read -r file; do

    # Process the B0FieldIdentifier field

    jq 'if has("B0FieldIdentifier") and (.B0FieldIdentifier | type == "string") then .B0FieldIdentifier |=
gsub("<<|>>"; "") else . end' "$file" > "${file}.tmp" && mv "${file}.tmp" "$file"

    # Process the B0FieldSource field

    jq 'if has("B0FieldSource") and (.B0FieldSource | type == "string") then .B0FieldSource |=
gsub("<<|>>"; "") else . end' "$file" > "${file}.tmp" && mv "${file}.tmp" "$file"

    # Check for errors
    if [[ $? -eq 0 ]]; then
        echo "Processed: $file"
    else
        echo "Error processing: $file"
    fi
done

echo "Processing complete."

```

This is to ensure that fmriprep is able to read fieldmap tags.

5. Run fmriprep

```

004 $ fmriprep_sub.py /project/2424007.01/dutch_data/bids -o /project/2424007.01/dutch_data/derivatives/fmriprep -p sub-5 sub-
6 sub-7 sub-8 sub-9 sub-11 sub-14 sub-15 --nthreads 3 --mem_mb 28000 --args " --output-spaces MNI152NLin6Asym --ignore slicetim
ing" --workdir /project/2424007.01/cla_tryout/work
>>> Submitting job (1/8):
qsub -l nodes=1:ppn=3,walltime=72:00:00,mem=28000mb -N fmriprep_sub-5 <<EOF
#!/bin/bash

ulimit -v unlimited
echo using: TMPDIR=${TMPDIR}
cd /home/predatt/clazha
apptainer run --cleanenv --bind ${TMPDIR}:/tmp,${TMPDIR}:/var/tmp /opt/fmriprep/24.0.0/fmriprep-24.0.0.simg /project/2424007.01/d
utch_data/bids /project/2424007.01/dutch_data/derivatives/fmriprep participant -w /project/2424007.01/cla_tryout/work/sub-5 --p
articipant-label 5 --fs-license-file /opt_host/fmriprep/license.txt --mem_mb 28000 --omp-nthreads 3 --nthreads 3 --output-spa
ces MNI152NLin6Asym --ignore slicetiming
EOF

>>> Submitting job (2/8):
qsub -l nodes=1:ppn=3,walltime=72:00:00,mem=28000mb -N fmriprep_sub-6 <<EOF
#!/bin/bash
sleep 1m

```

Codes:

24.0.0

```

fmriprep_sub.py /project/***/bids -o
/project/***/derivatives/fmriprep -p sub-5 sub-6 sub-7 sub-8 sub-9

```



```
sub-11 sub-14 sub-15 --nthreads 3 -r 'slurm' --mem_mb 28000 --args "  
--output-spaces MNI152NLin6Asym --ignore slicetiming" --workdir  
/project/2424007.01/cla_tryout/work
```

```
fmrip_prep_sub.py /project/***/bids -o /project/***/  
derivatives/fmrip_prep --nthreads 3 -r 'slurm' --mem_mb 28000 --args  
" --output-spaces MNI152NLin6Asym --ignore slicetiming" --workdir  
/project/2424007.01/cla_tryout/work
```

Note:

For fmrip_prep 24.0.0 the “args” used are the following: *--output-spaces MNI152NLin6Asym --ignore slicetiming*

If you are running it at local computing cluster, put this to the fmrip_prep docker. You can choose to run fmrip_prep in participant batches. Then, you should add -p to call for specific participants.

Remove -r 'slurm' if you are still using torque.

20.2.1

```
fmrip_prep_sub.py /project/***/bids -o /project/***/derivatives/fmrip_prep --nthreads 3 --mem_mb  
28000 --participant_label 6 --args " --use-aroma --return-all-components --ignore slicetiming --  
output-spaces MNI152NLin6Asym --ignore fieldmap" --workdir  
/project/2424007.01/cla_tryout/work
```

Note:

For fmrip_prep 20.2.1 args, it is just this: *--use-aroma --return-all-components --ignore slicetiming --output-spaces MNI152NLin6Asym --ignore fieldmap*

If you are running it at local computing cluster, put this to the fmrip_prep docker. You can choose to run fmrip_prep in participant batches. Then, you should add -p to call for specific participants.

HPC cluster code to check job status:

```
qstat -u username
```