Grunnleggende Bildebehandling

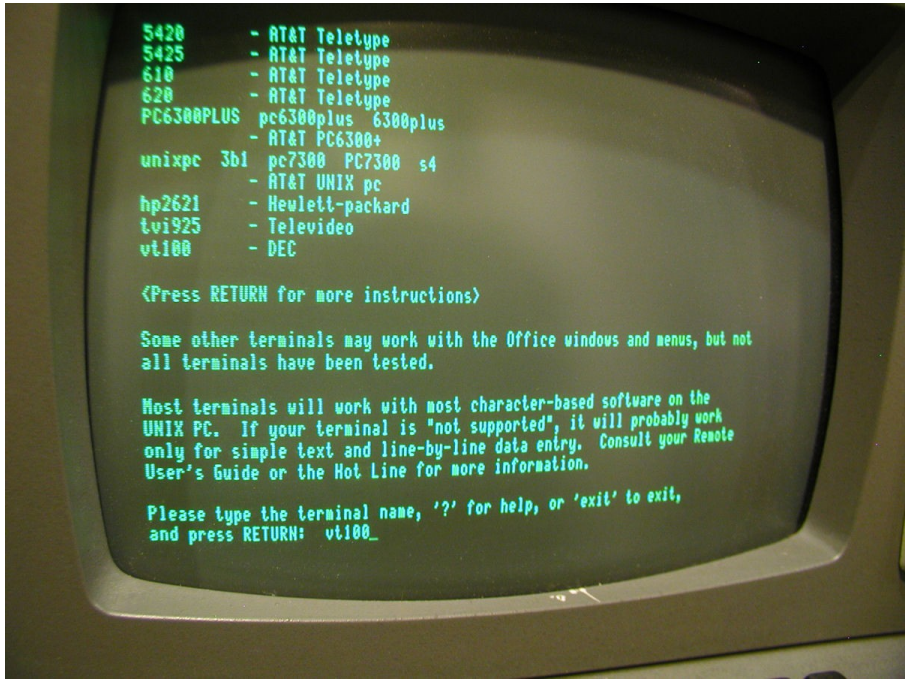
Forelesninger for RAD230
Ivan I. Maximov

Unix systemer

Terminal er noen historiske begrepet kom fra gamle unix (første) maskiner.
Hva trenger vi at en mulighet å skrive kommander på spesielle stedet hvilket er kjent som terminal.

Dere kan finne terminal i deres komputerer også.

Viktig å forstå: terminal gir oss en mulighet å jobbe med datamaskiner som vi bruker kommander.



Unix systemer

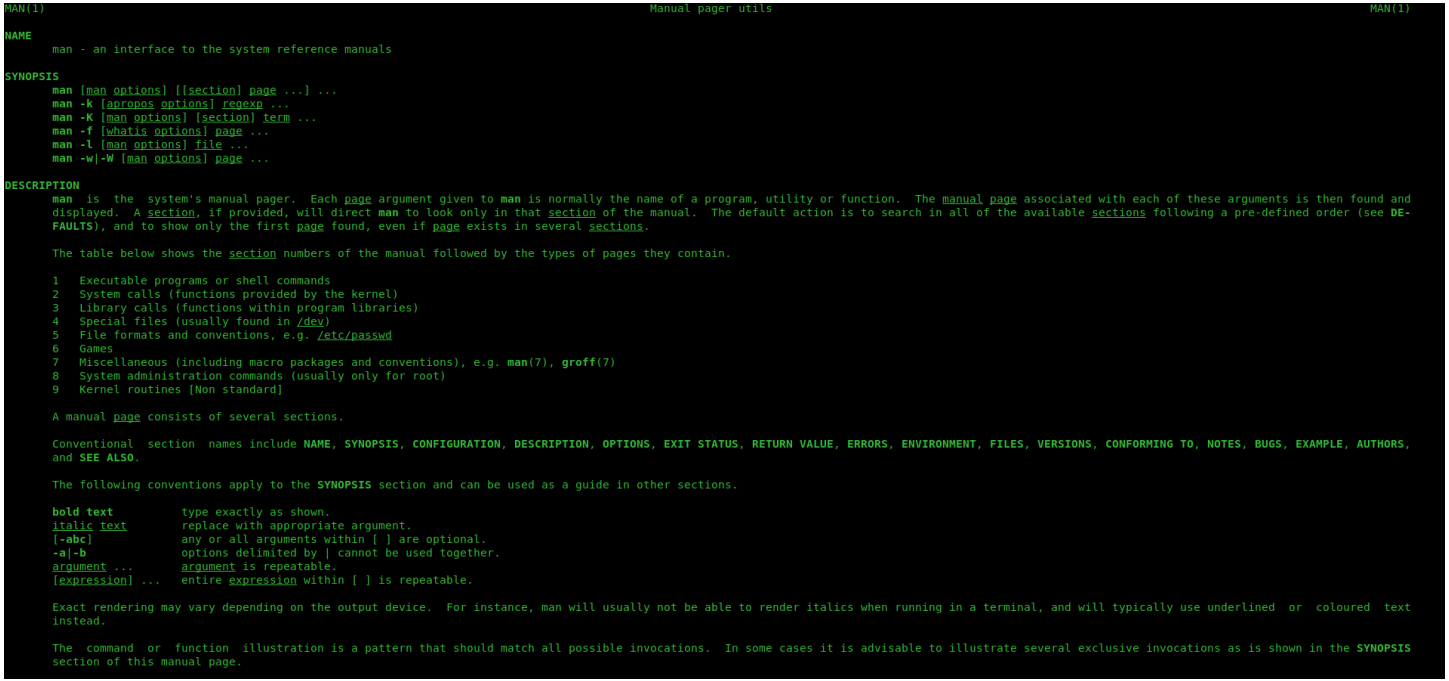
Man kan få hjelpen på terminalen hvis skriver:

>man man

Det fungerer for alle kommander

>man ls
>man cd

etc…



```
MAN(1)                                   Manual pager utils                                   MAN(1)

NAME
       man - an interface to the system reference manuals

SYNOPSIS
       man [man options] [[section] page ...] ...
       man -k [apropos options] regexp ...
       man -K [man options] [section] term ...
       man -f [whatis options] page ...
       man -l [man options] file ...
       man -w|-W [man options] page ...

DESCRIPTION
       man  is  the  system's manual pager.  Each page argument given to man is normally the name of a program, utility or function.  The manual page associated with each of these arguments is then found and
       displayed.  A section, if provided, will direct man to look only in that section of the manual.  The default action is to search in all of the available sections following a pre-defined order (see DE-
       FAULTS), and to show only the first page found, even if page exists in several sections.

       The table below shows the section numbers of the manual followed by the types of pages they contain.

       1   Executable programs or shell commands
       2   System calls (functions provided by the kernel)
       3   Library calls (functions within program libraries)
       4   Special files (usually found in /dev)
       5   File formats and conventions, e.g. /etc/passwd
       6   Games
       7   Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7)
       8   System administration commands (usually only for root)
       9   Kernel routines [Non standard]

       A manual page consists of several sections.

       Conventional  section  names include NAME, SYNOPSIS, CONFIGURATION, DESCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUE, ERRORS, ENVIRONMENT, FILES, VERSIONS, CONFORMING TO, NOTES, BUGS, EXAMPLE, AUTHORS,
       and SEE ALSO.

       The following conventions apply to the SYNOPSIS section and can be used as a guide in other sections.

       bold text          type exactly as shown.
       italic text        replace with appropriate argument.
       [-abc]             any or all arguments within [ ] are optional.
       -a|-b              options delimited by | cannot be used together.
       argument ...       argument is repeatable.
       [expression] ...   entire expression within [ ] is repeatable.

       Exact rendering may vary depending on the output device.  For instance, man will usually not be able to render italics when running in a terminal, and will typically use underlined  or  coloured  text
       instead.

       The  command  or  function  illustration is a pattern that should match all possible invocations.  In some cases it is advisable to illustrate several exclusive invocations as is shown in the SYNOPSIS
       section of this manual page.
```

Unix systemer

File Commands
ls – directory listing
ls -al – formatted listing with hidden files
cd *dir* - change directory to *dir*
cd – change to home
pwd – show current directory
mkdir *dir* – create a directory *dir*
rm *file* – delete *file*
rm -r *dir* – delete directory *dir*
rm -f *file* – force remove *file*
rm -rf *dir* – force remove directory *dir* *
cp *file1* *file2* – copy *file1* to *file2*
cp -r *dir1* *dir2* – copy *dir1* to *dir2*; create *dir2* if it doesn't exist
mv *file1* *file2* – rename or move *file1* to *file2* if *file2* is an existing directory, moves *file1* into directory *file2*
ln -s *file* *link* – create symbolic link *link* to *file*
touch *file* – create or update *file*
cat > *file* – places standard input into *file*
more *file* – output the contents of *file*
head *file* – output the first 10 lines of *file*
tail *file* – output the last 10 lines of *file*
tail -f *file* – output the contents of *file* as it grows, starting with the last 10 lines

Dette er korte "cheat" listen for flere kommander.

Man kan skrive kommander i filen og kjøre filen som bash-skript:

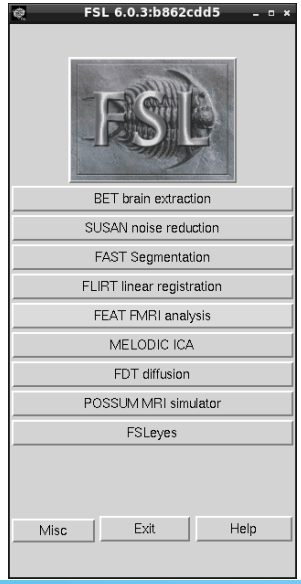
```
#!/bin/bash

echo "Start"
pwd
echo "pwd works"
touch MyFile.txt
echo "text in MyFile.txt">>MyFile.txt
echo "Done"
```

Unix systemer

https://fsl.fmrib.ox.ac.uk/fsl/fslwiki
Nettsiden med info om FSL.
FSL er om Neuroimaging men man kan bruke det samme teknikker for andre måte.

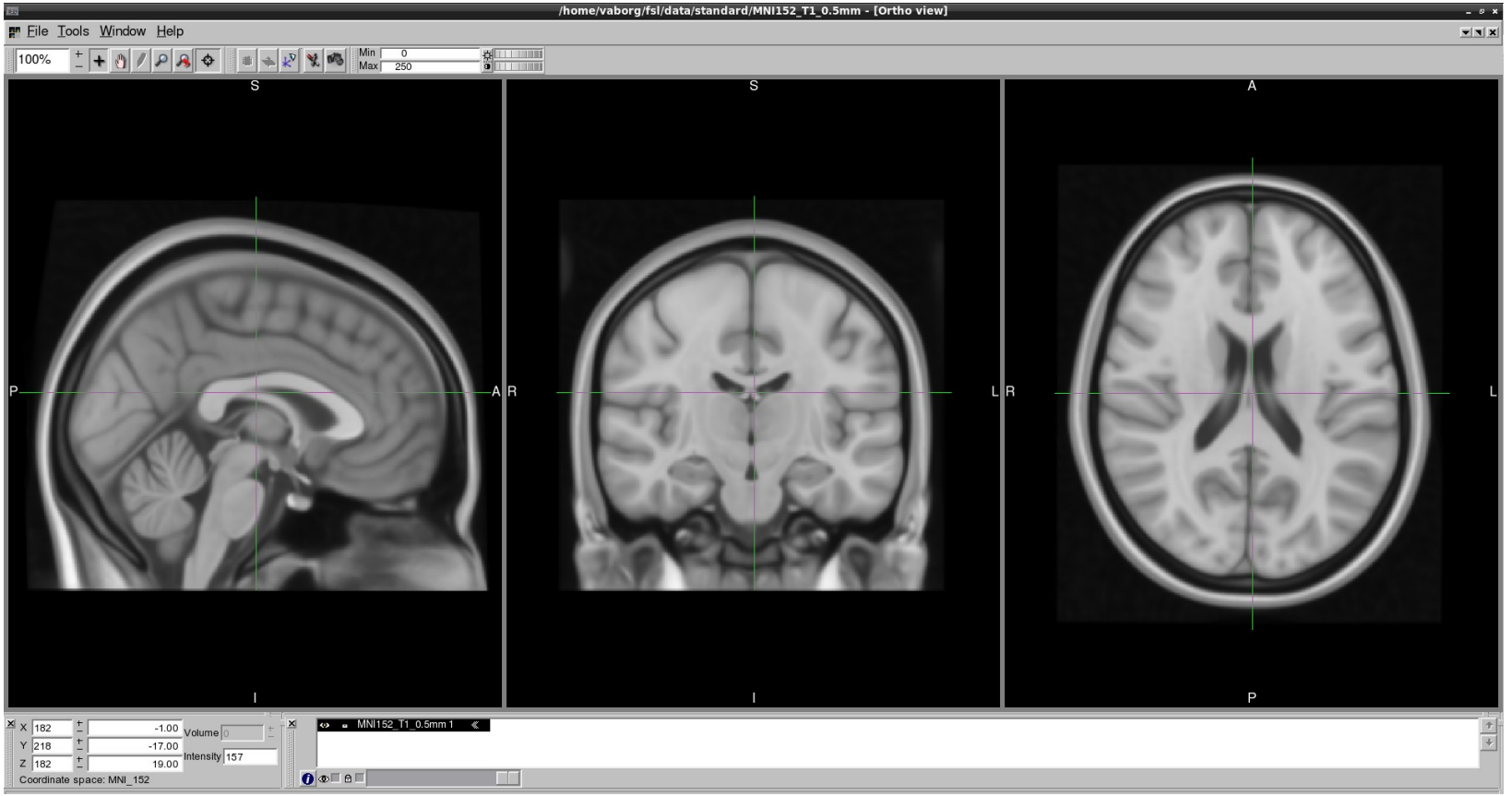
Det er ofte to typer of utiliter: med GUI og uten.
Å kjøre hoved vinduet, trykk: fsl



FSL

Hvordan kan man se bilder?

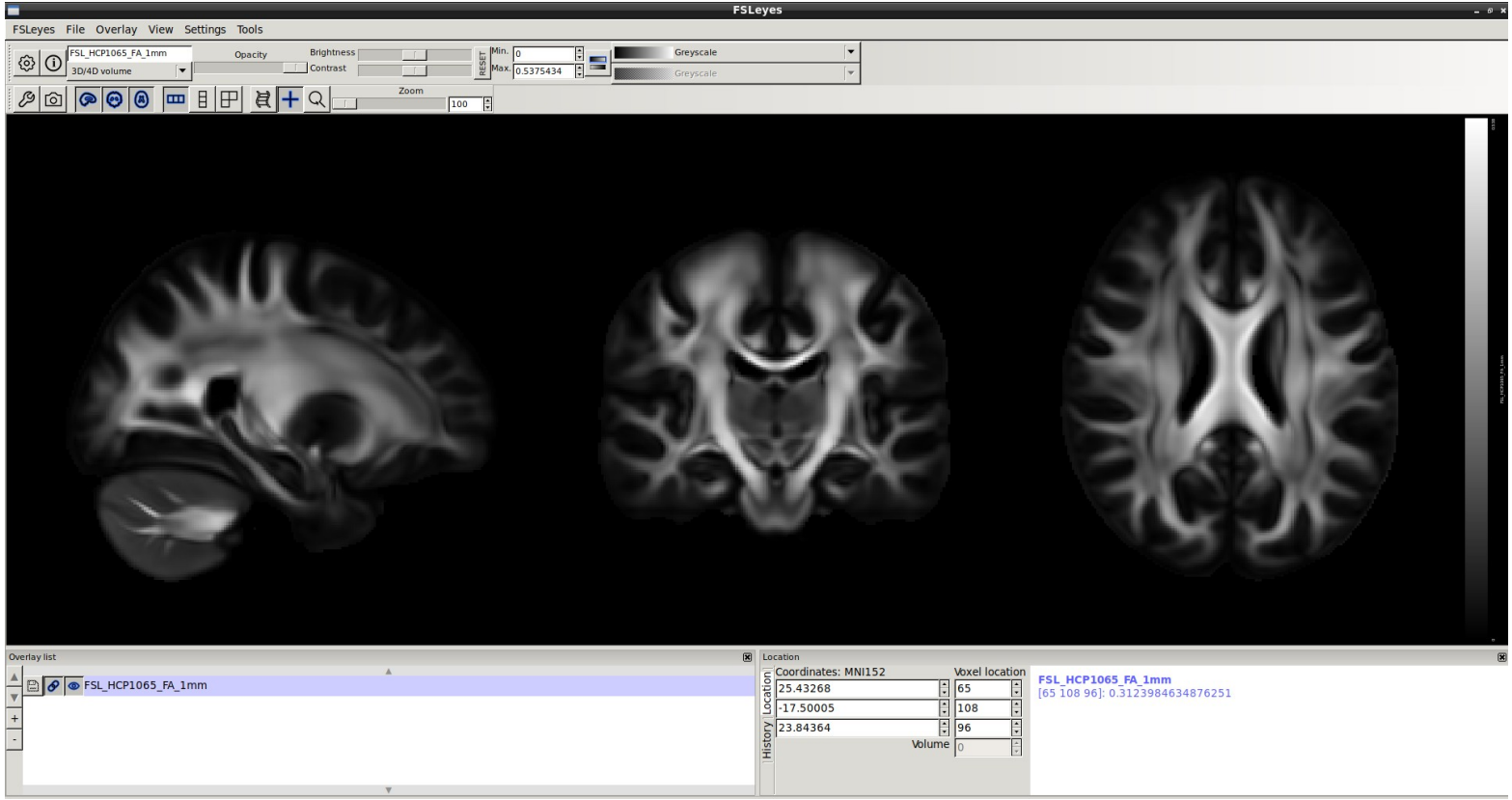
1. fslview_depricated



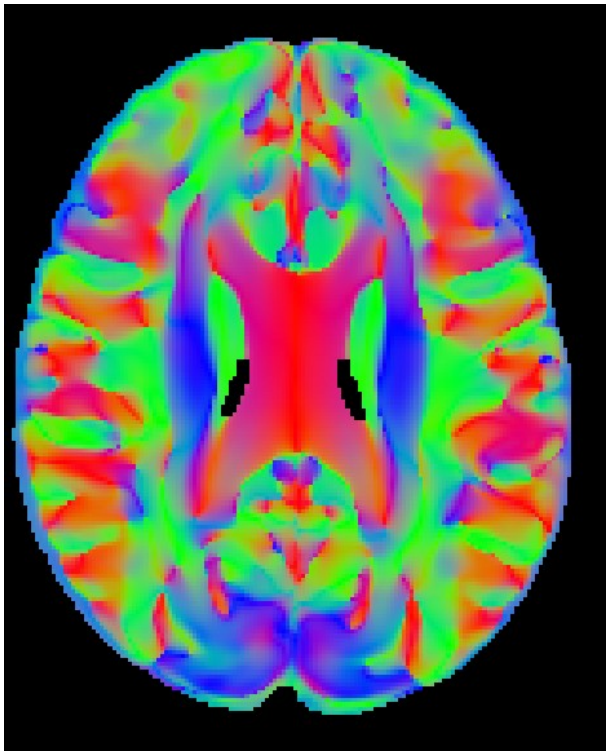
FSL

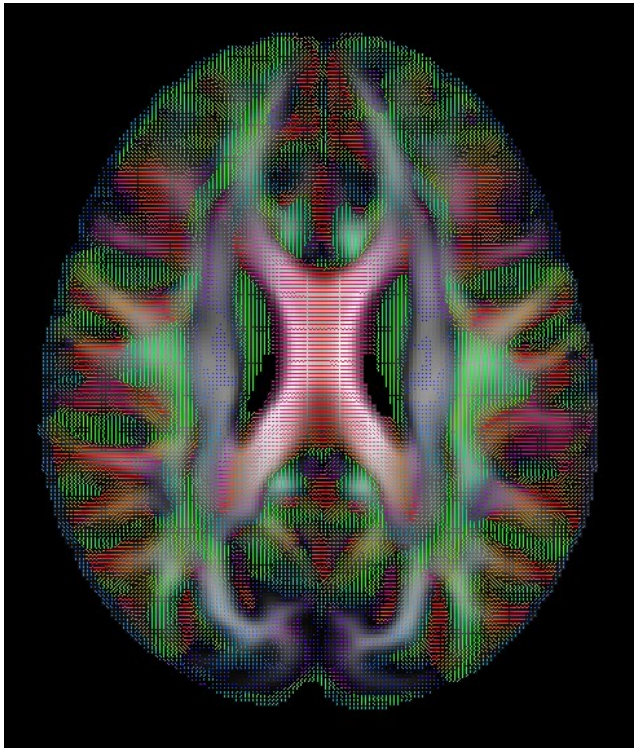
Hvordan kan man se bilder?

1. fslview_depricated
2. fsleyes



FSL

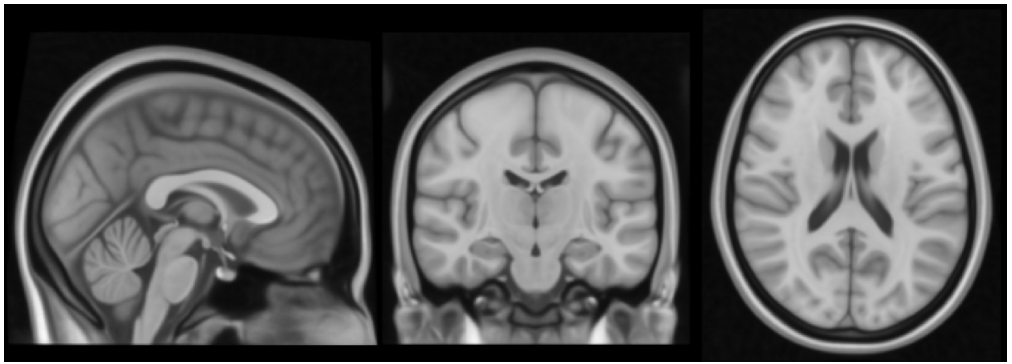




Forskjellige typer av modaliter kan man visualisere med hjelpe av fsleyes. Man kan bruke mange layer og oppløsninger.

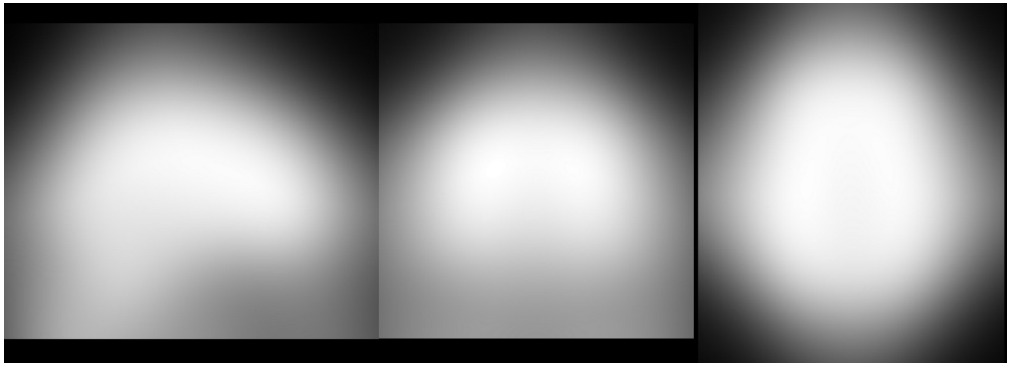
Vi trenger bare å se resultater for våre manipulasjoner.
For mer, se FSL kurs materialer på nett.

FSL

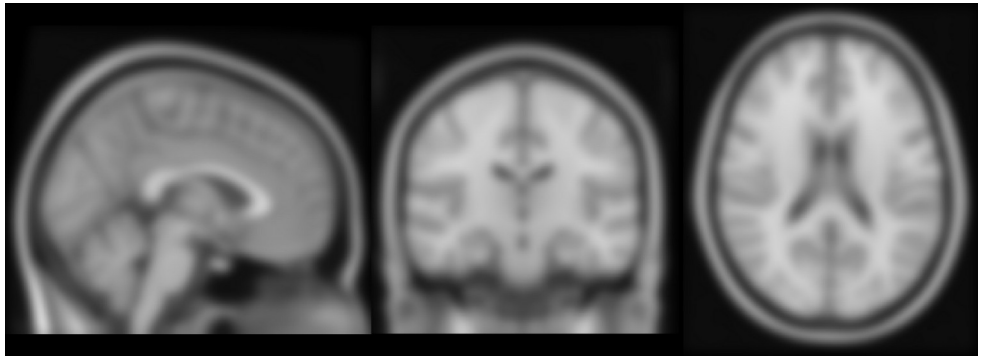


fslmaths utiliti gir oss en mulighet å jobbe med bilder som vi jobber med matriser før. Alle algorithmer er allerede implementert in fslmaths og vi bør lære hvordan skal vi bruke det.

fslmaths opp_bildet -s 20mm ned_bildet

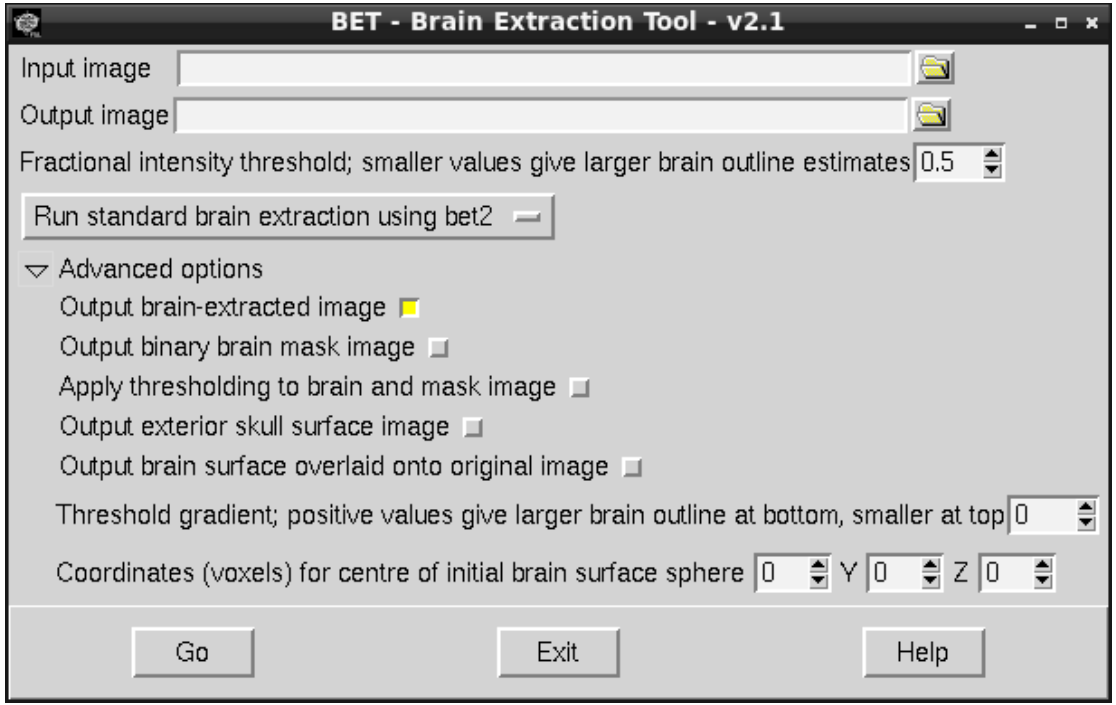


fslmaths opp_bildet -s 3mm ned_bildet

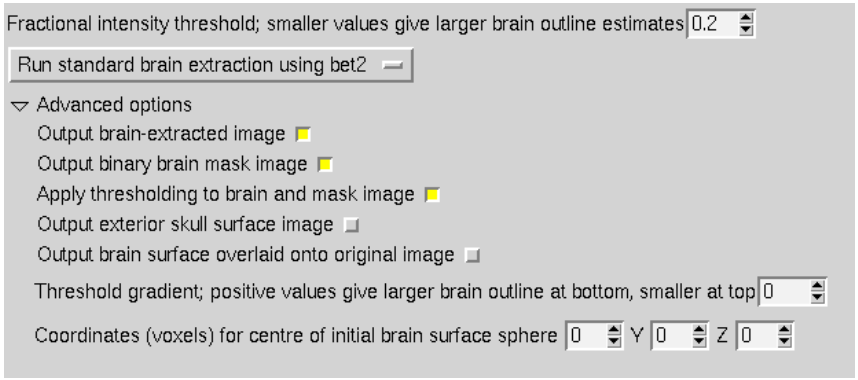


BET

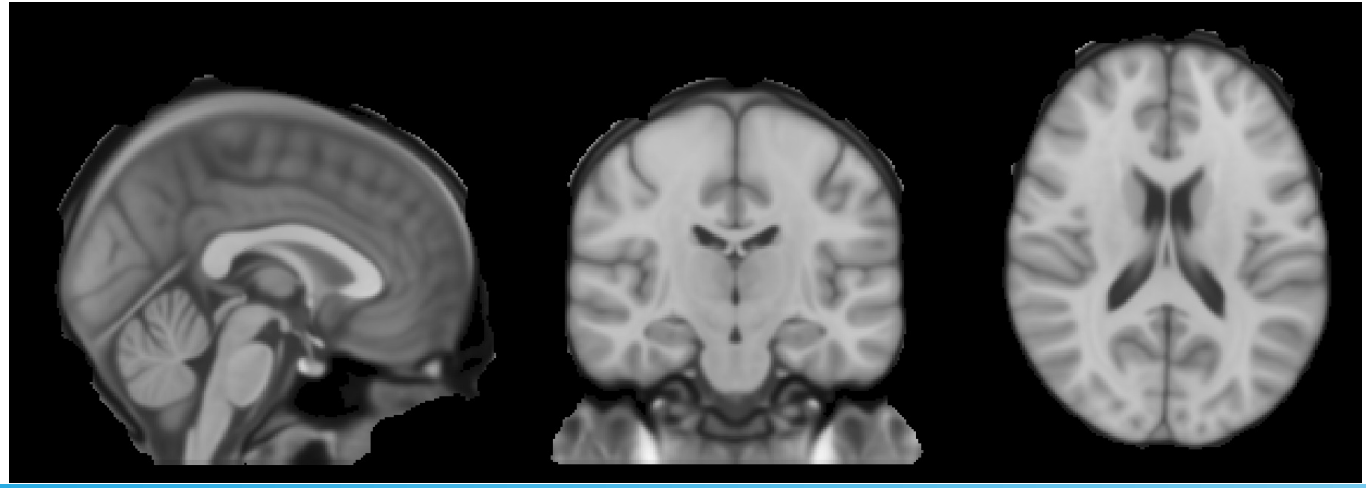
Å kjøre BET GUI trykk: Bet



BET



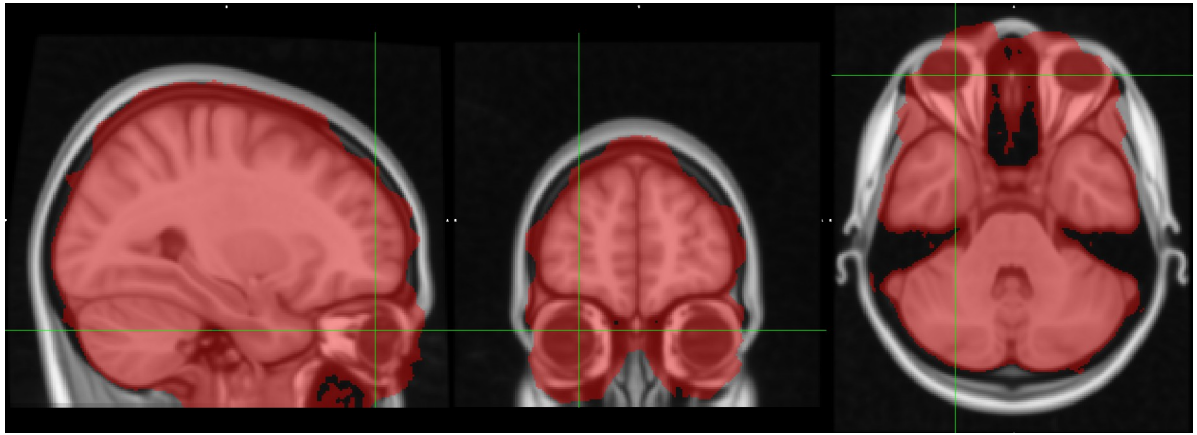
Parameter brukte vi for hjern extraction



BET

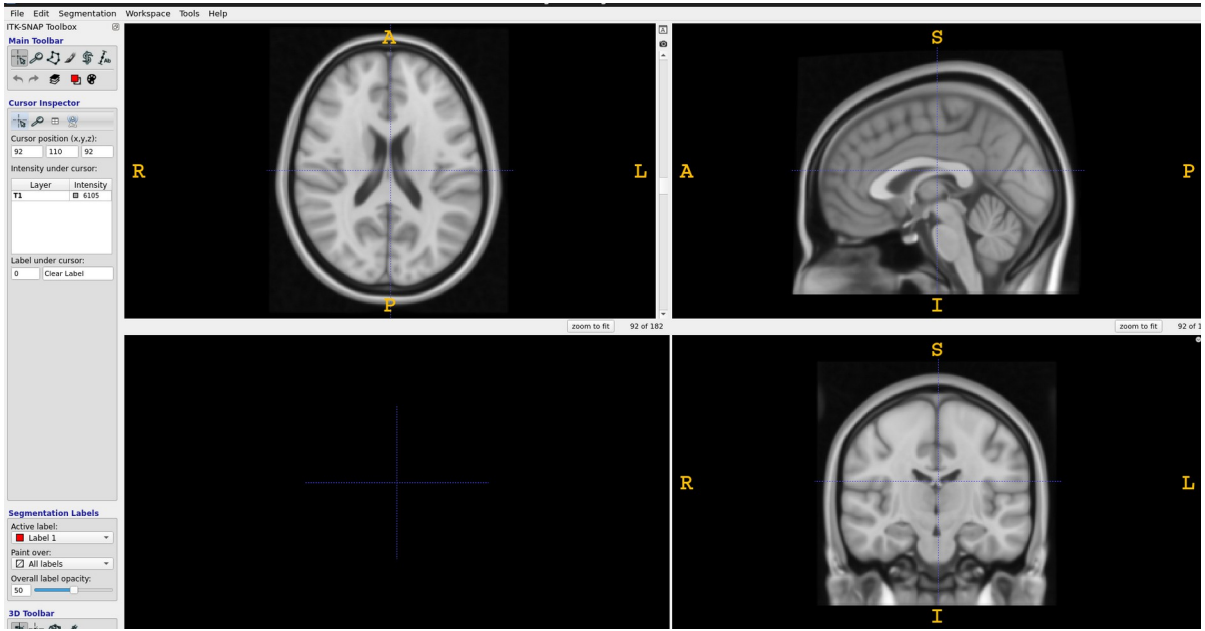
Det er viktig å se hvilke threshold (-f eller fractional intensity threshold) verdi brukte vi for hjern extraction. Som en bivirkning har vi MASK også.

Nyttig å vite: hvis vi har noen binary mask, da kan vi “extract” hjernen med fslmaths:
fslmaths input_brain -mul mask output_brain



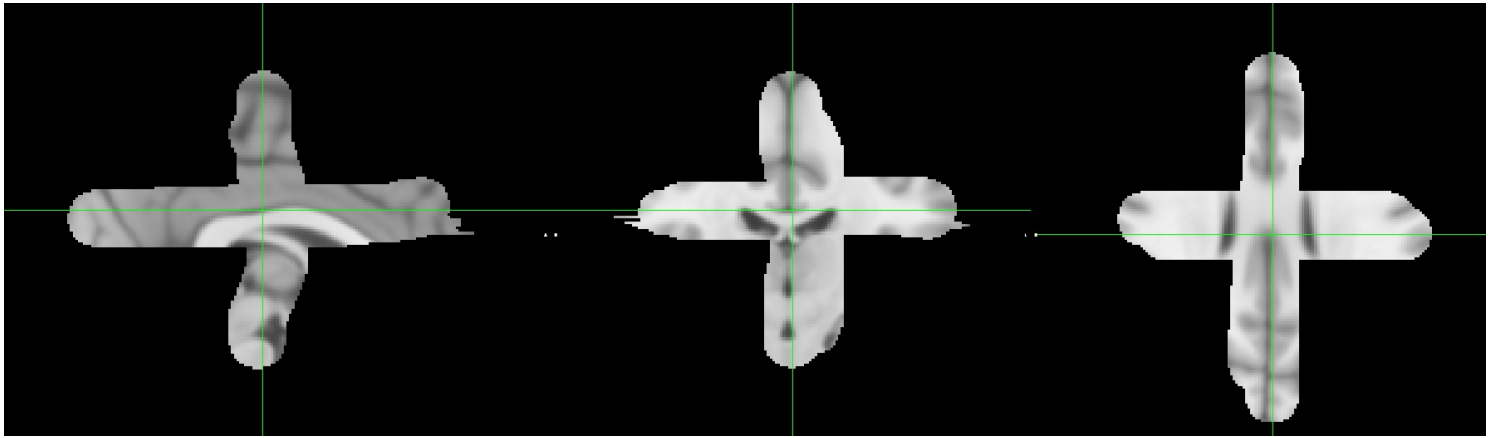
BET

Som eksampel, kunne vi jobbe med manuell masking. For dette man kan bruke andre vertøy kjent som ITK-SNAP.



BET

"Fancy" eksampel med manuell masking og *fslmaths*



FSL

Oppgave 1: å vurdere en kvalitet for BET hjerns utdrag (extraction) avhengig av Gaussian smoothing for 0.5, 1, 2, 3 og 5mm.
Bruk forskjellige thresholding (-f 0.2, 0.3 og 0.5)

Oppsummer resultater og si når kan man få den beste kvaliteten for hjerns utdrag.