Registration

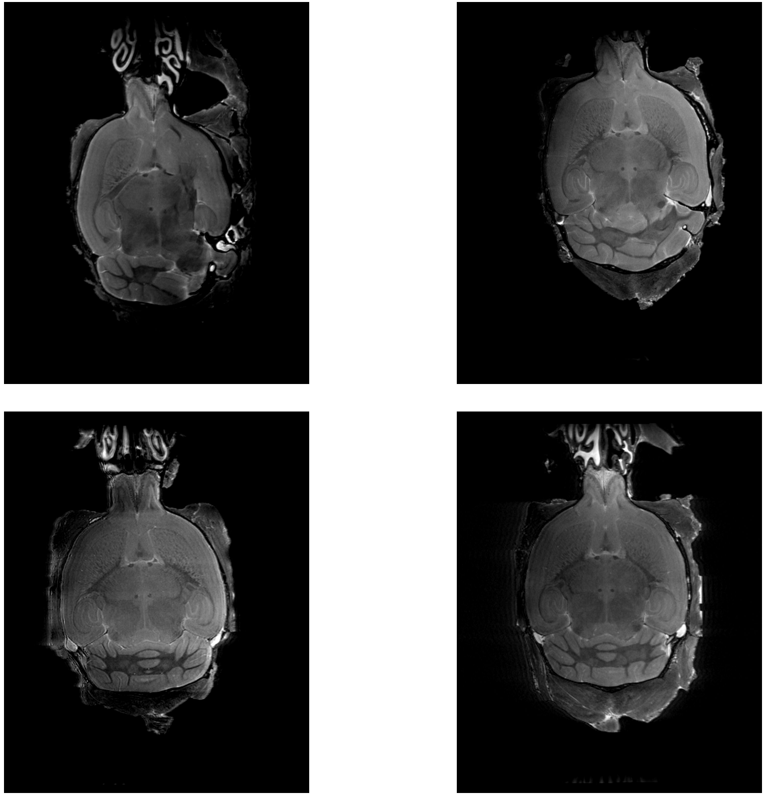
Darren Fernandes

Outline

1. What is registration?
   - Why we do it?
   - Consensus average
2. Aligning two images
   - Rigid transformations
   - Affine transformations
   - Nonlinear transformations
   - Relative and absolute determinants
3. Pydpiper registration pipeline
4. How transformations are encoded in *.xfm files

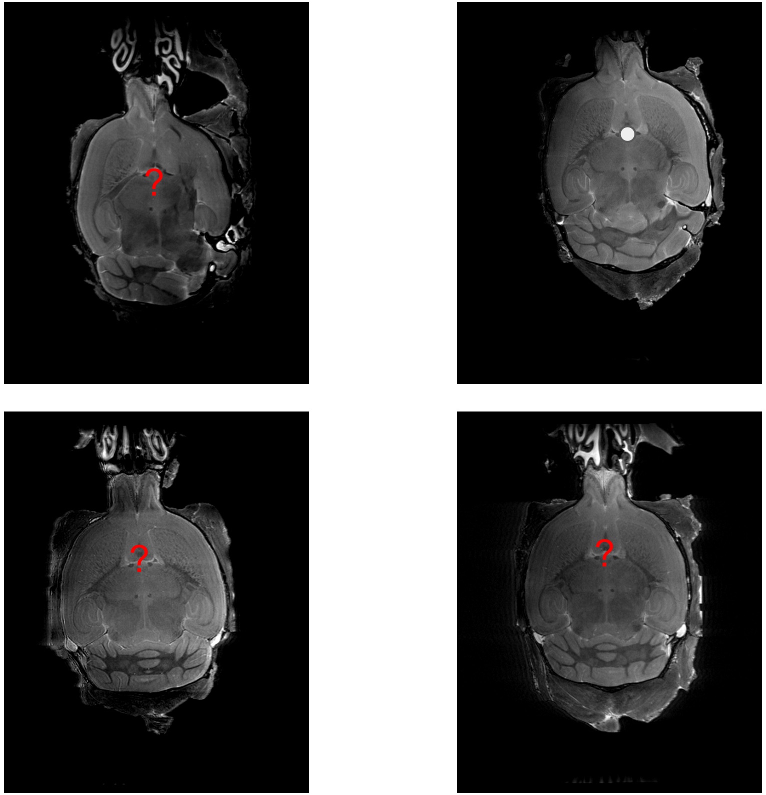
Why do we register?

Raw images
after scanning



Why do we register?

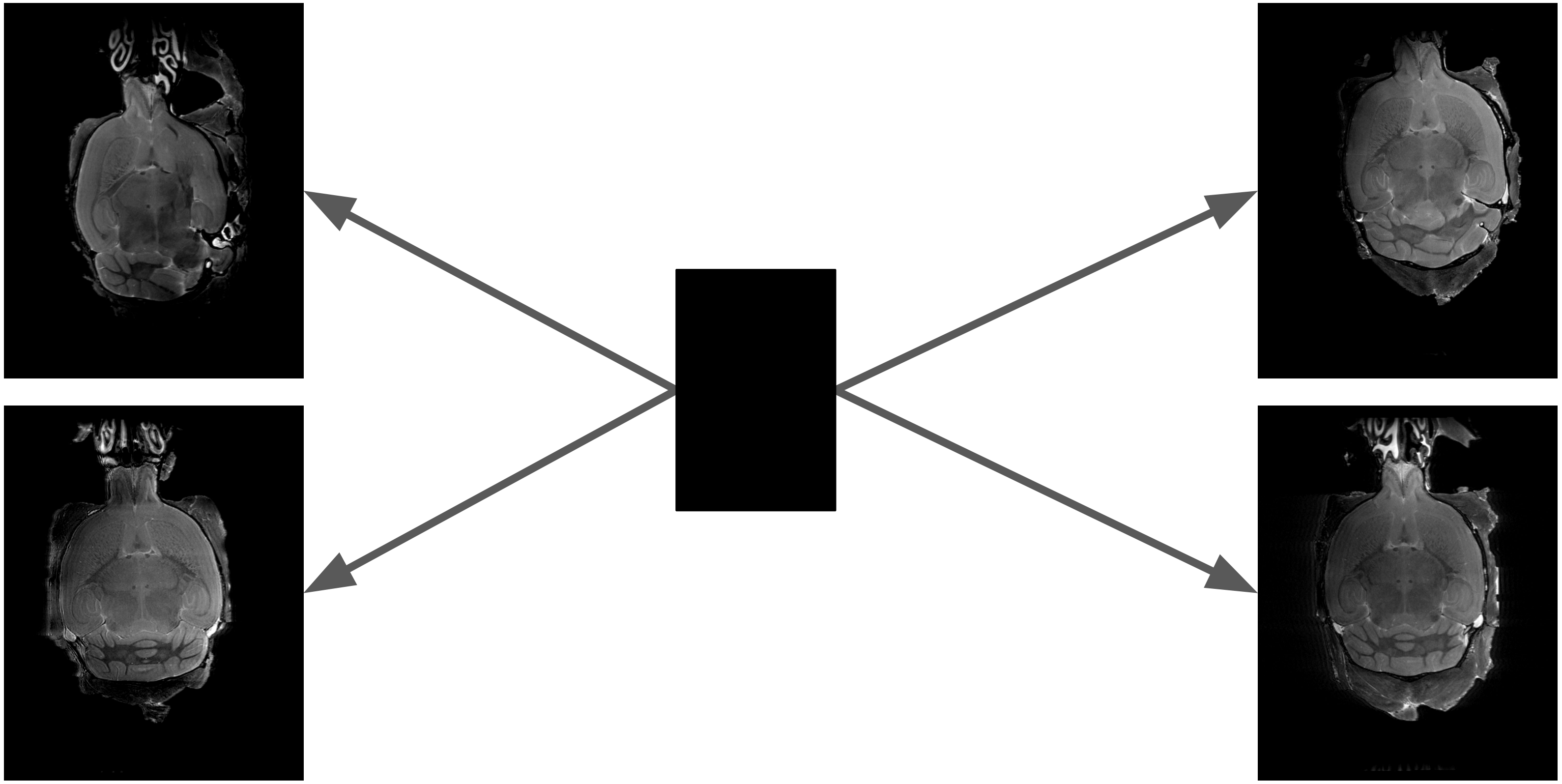
How do points in one image correspond to points in another image?



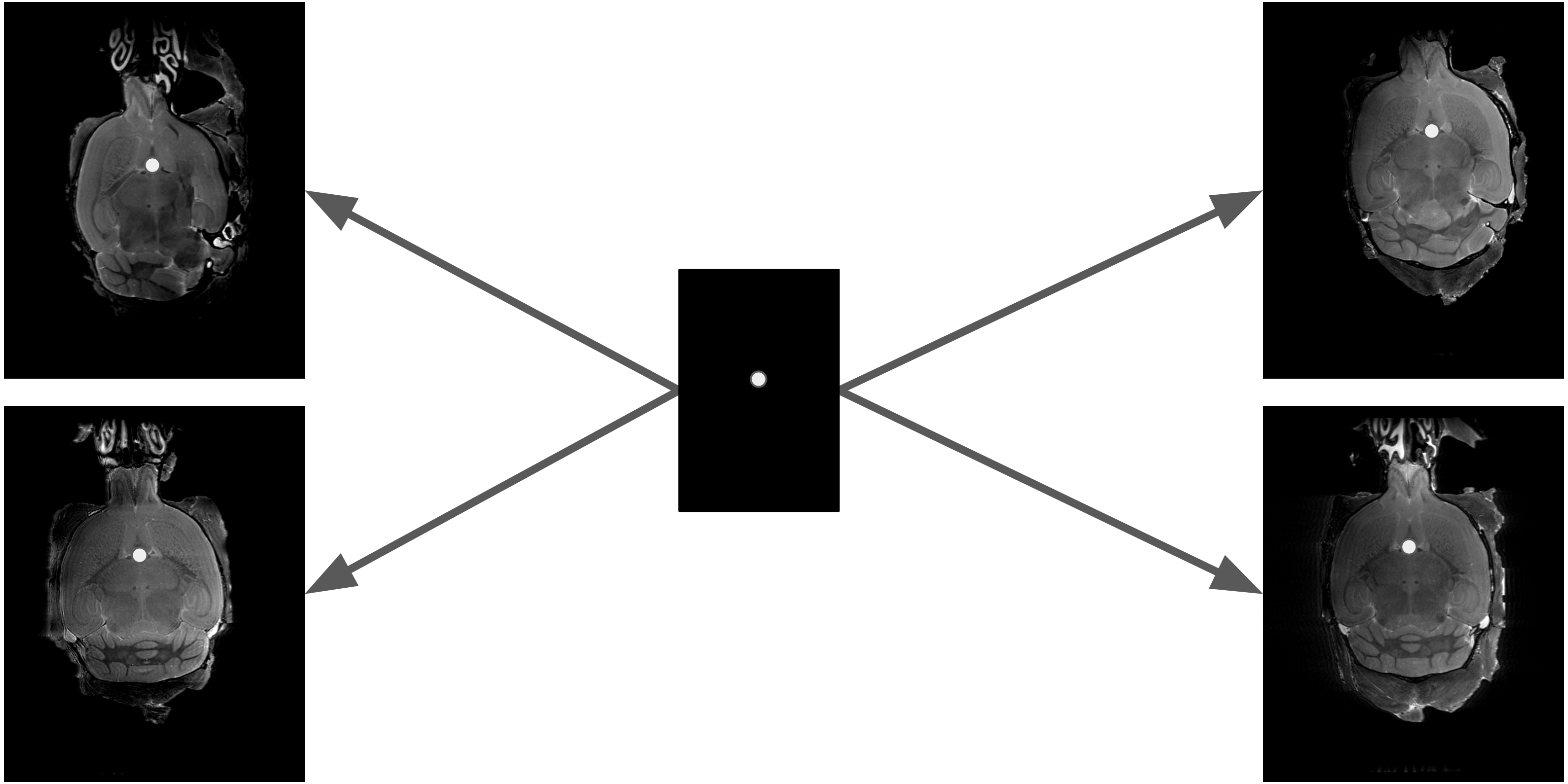
Why do we register?

1) Establishing point-by-point correspondence between all images in the goal of registration.

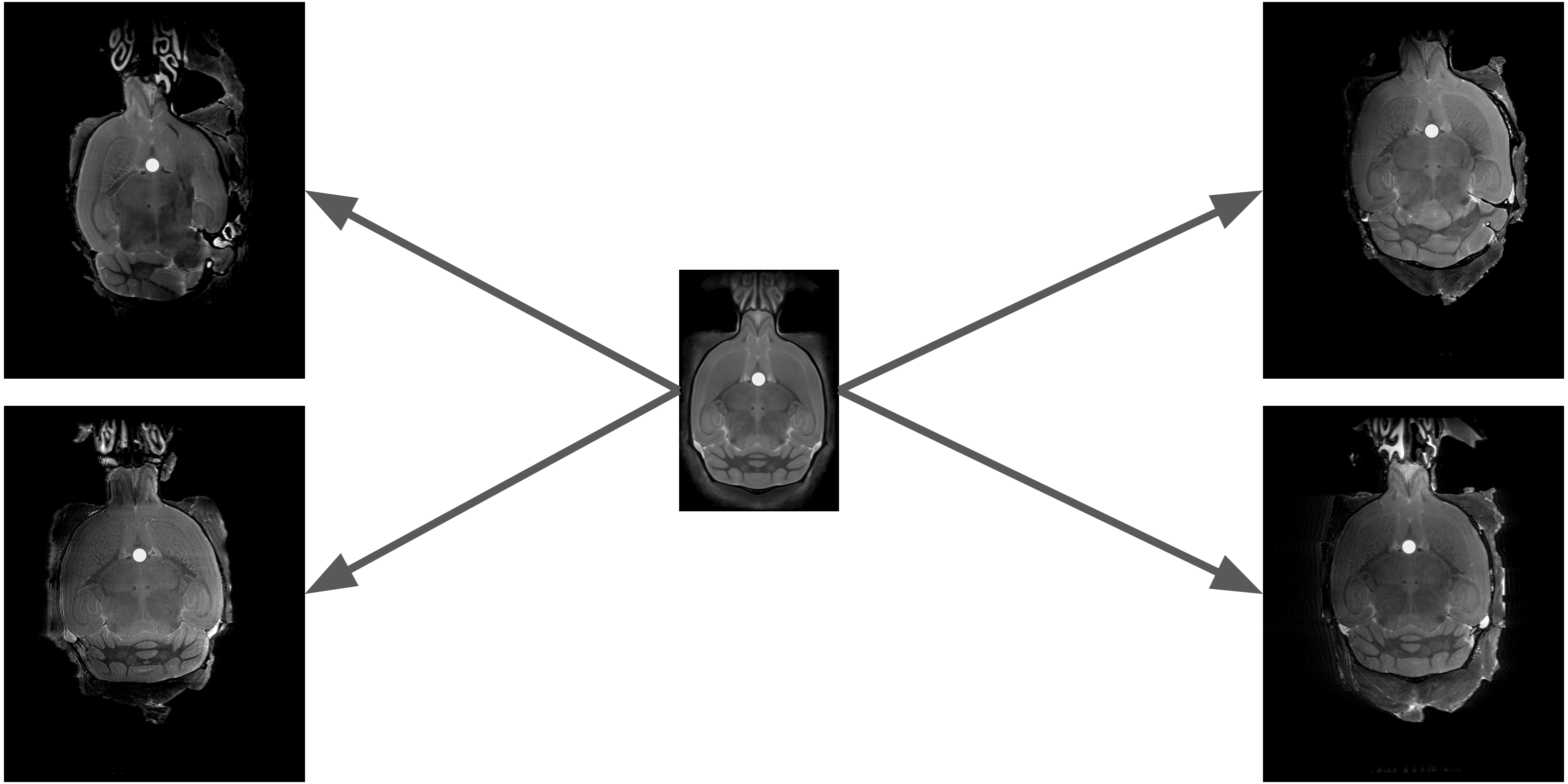
Why do we register?



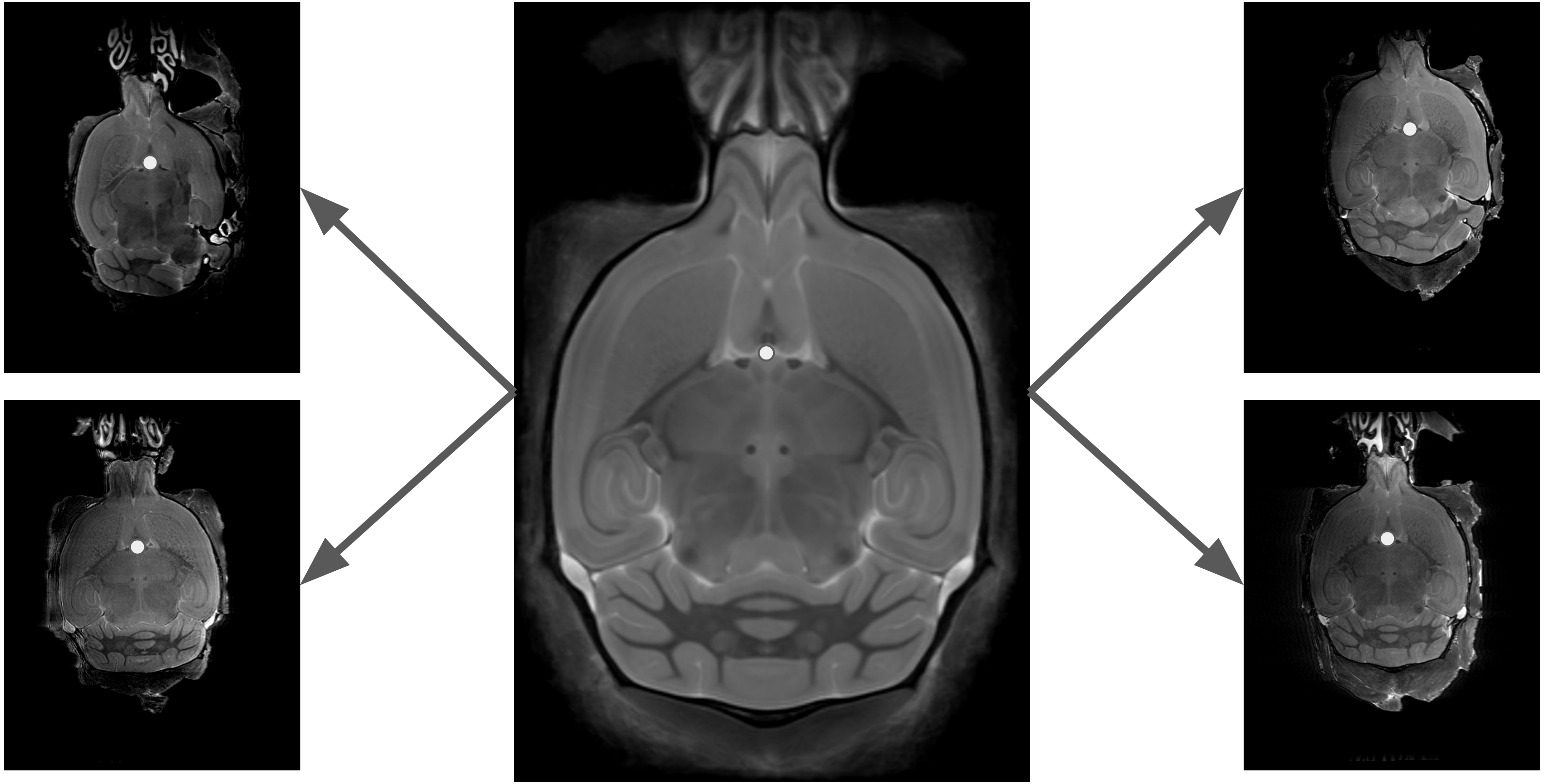
Why do we register?



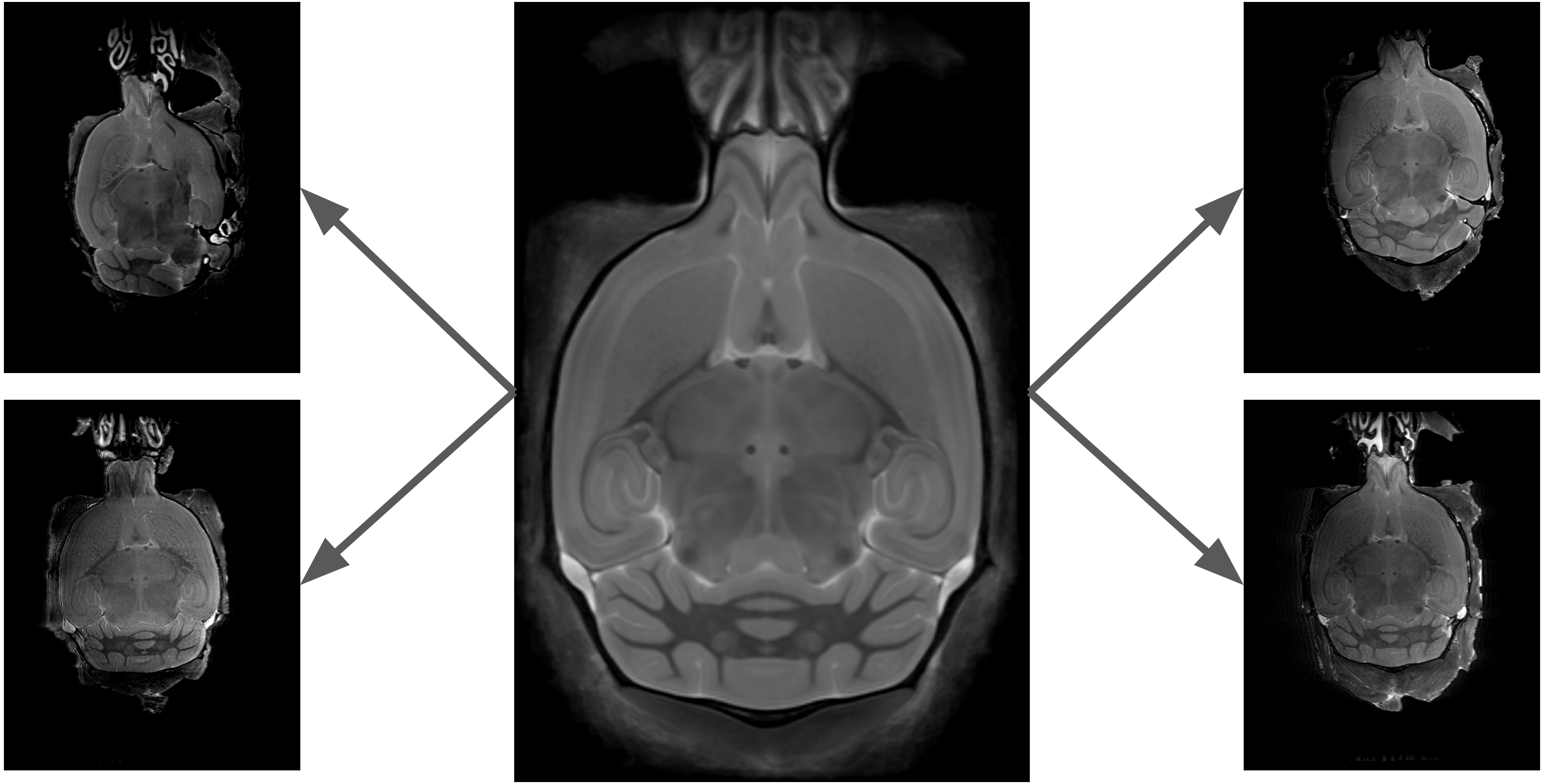
Why do we register?



Why do we register?



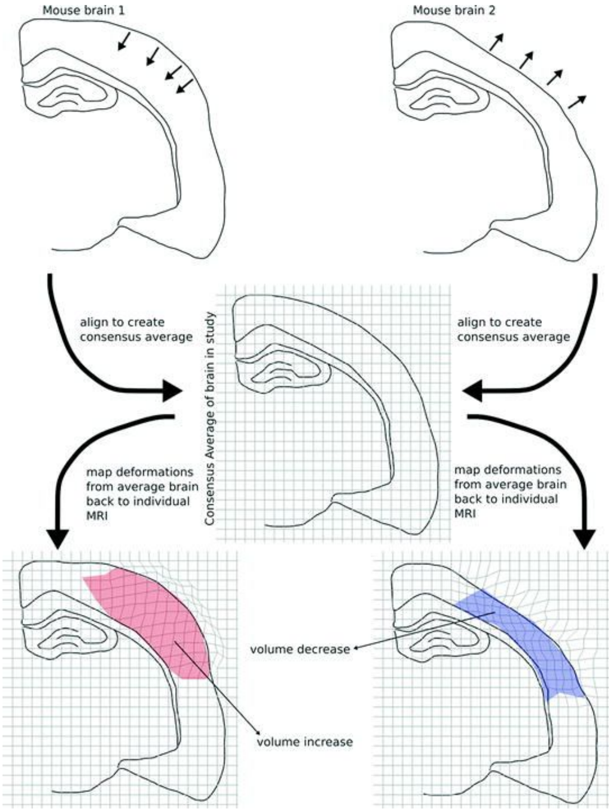
Why do we register?



Why do we register?

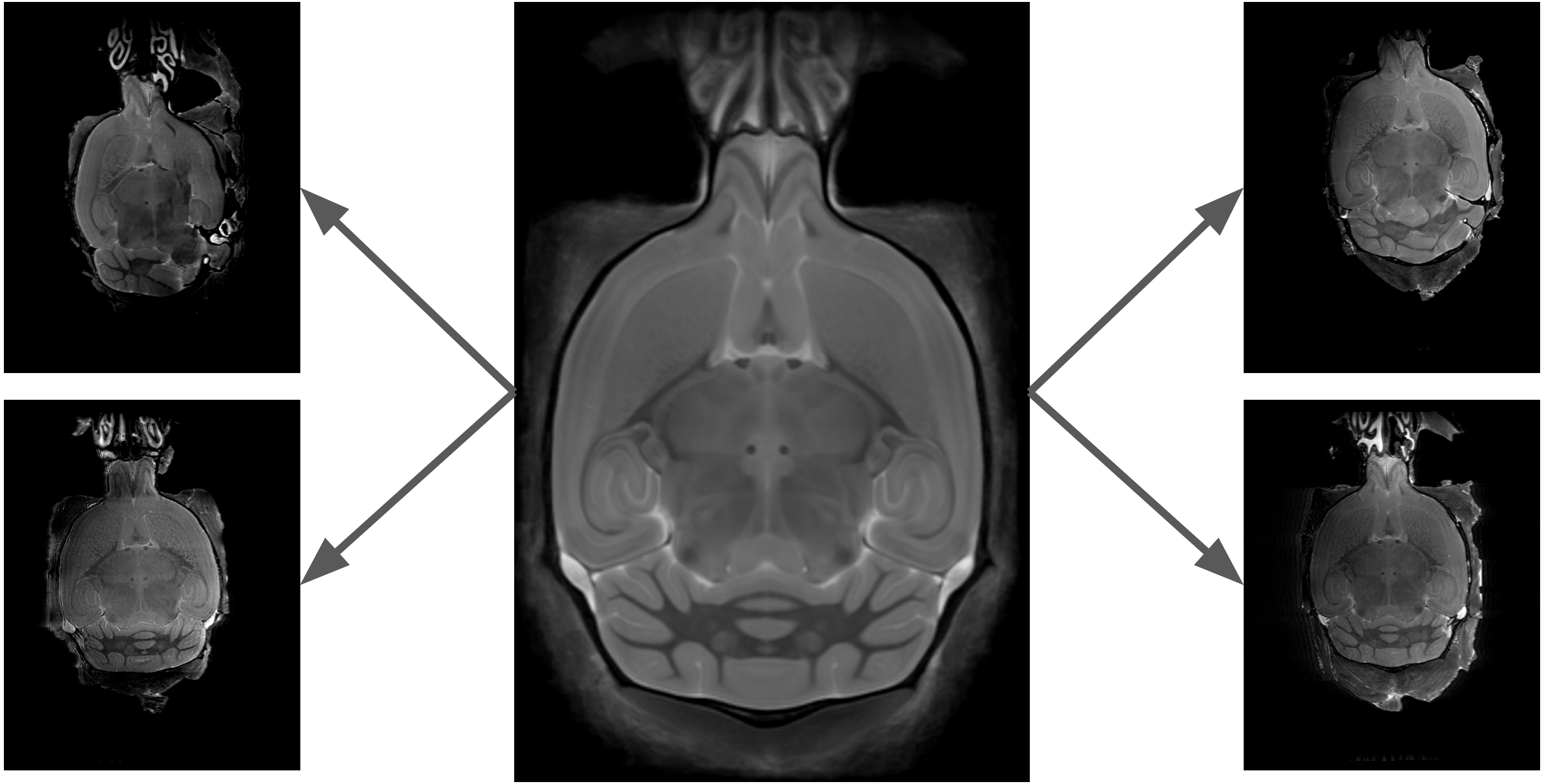
1) Establishing point-by-point correspondence between all images in the goal of registration.
2) High SNR consensus average

Why do we register?

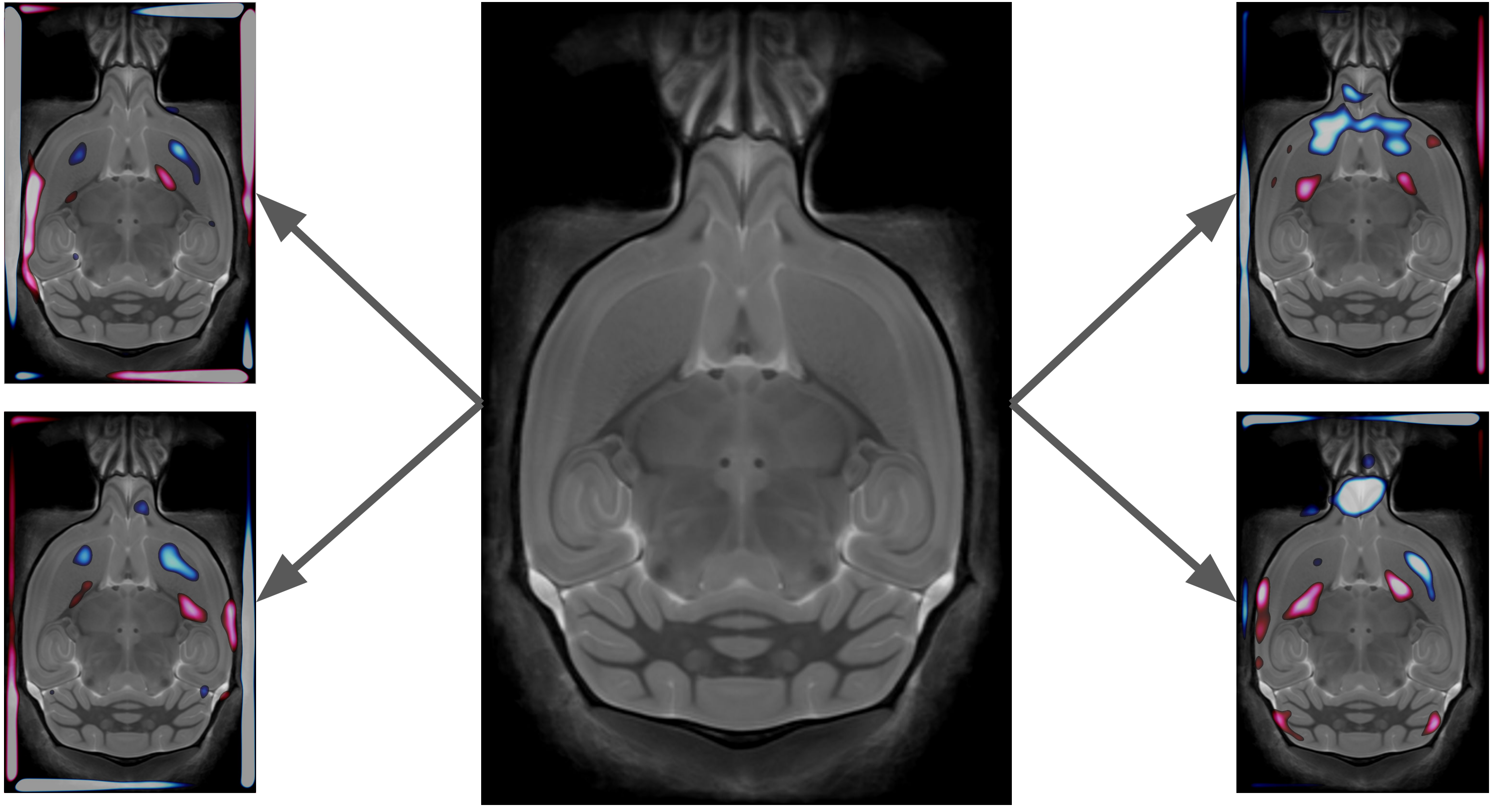

Mouse brain 1
Mouse brain 2
align to create consensus average
align to create consensus average
Consensus Average of brain in study
map deformations from average brain back to individual MRI
map deformations from average brain back to individual MRI
volume decrease
volume increase

Lerch, J. P., Sled, J. G., & Henkelman, R. M. (2011). MRI phenotyping of genetically altered mice. *Magnetic Resonance Neuroimaging: Methods and Protocols*, 349-361.

Why do we register?

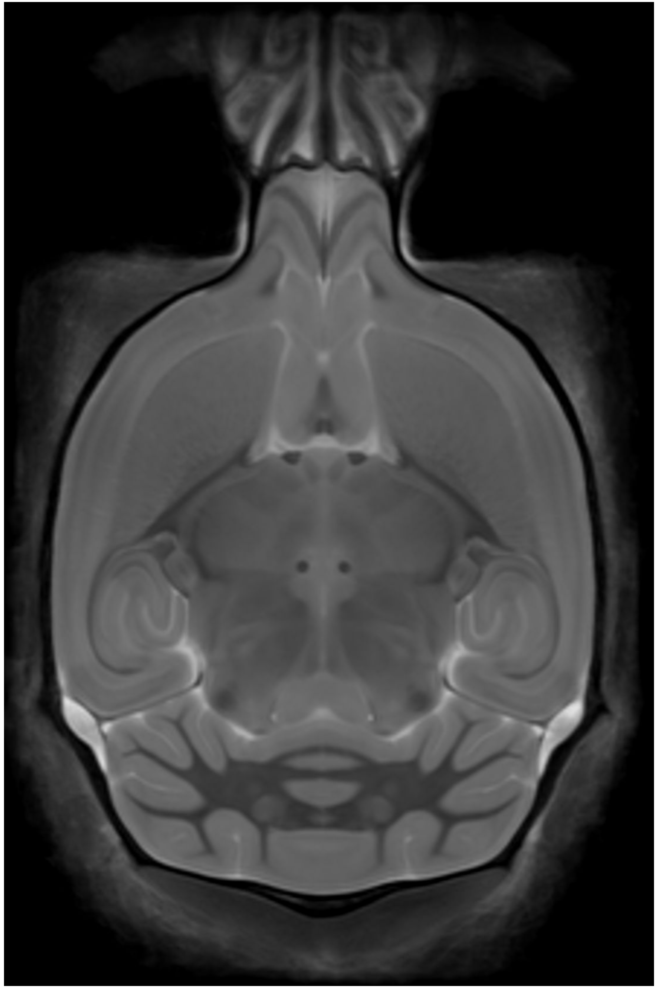


Why do we register?

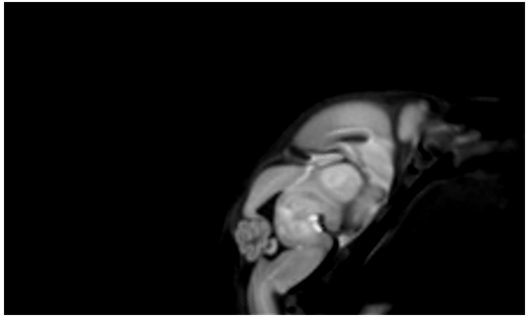


Why do we register?

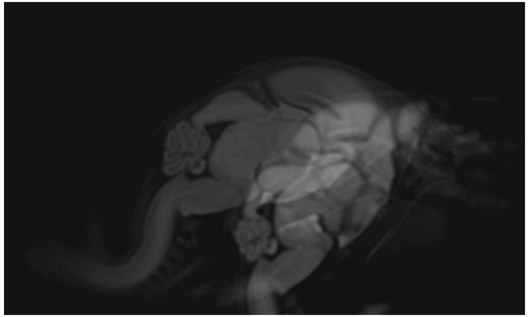
1) Establishing point-by-point correspondence between all images in the goal of registration.
2) High SNR consensus average
   - Consensus Average fixes the null measure of our volumetry analysis
3) Volumetric differences between images
   - Region volumes in subjects brains are measured in reference to the consensus average



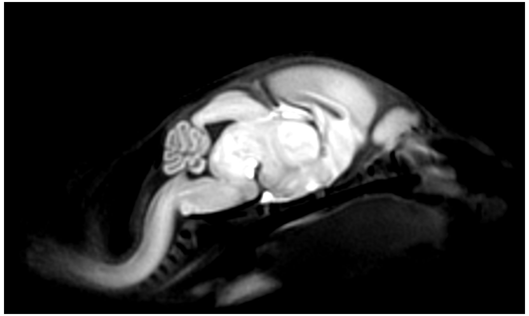
Aligning two images



source

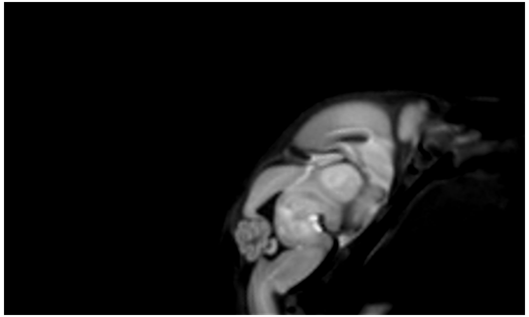


overlay

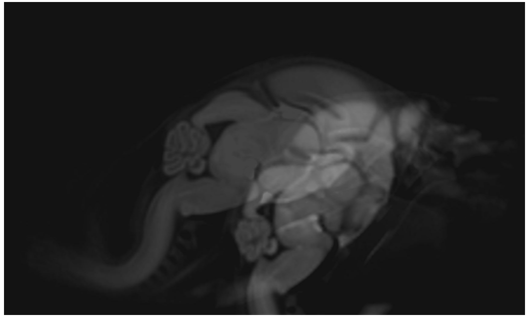


target

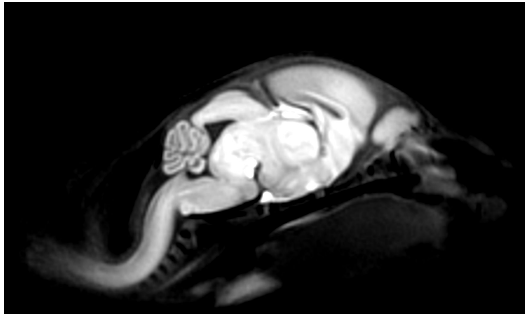
Aligning two images



source



overlay



target

What transformations should we do?

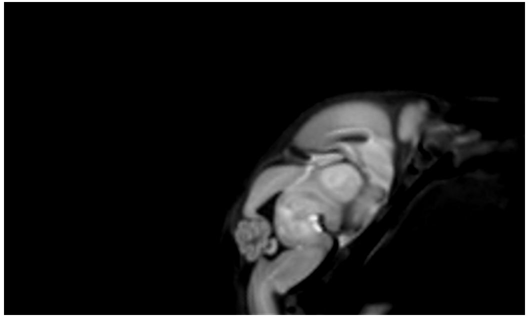
LSQ6 - Rigid transformation

- **L**east **SQ**uares with **6** degrees-of-freedom (dof).

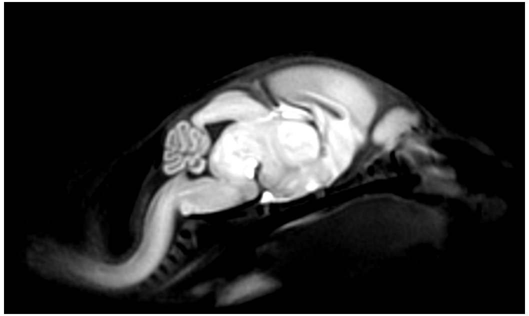
LSQ6

- **L**east **SQ**uares with **6** degrees-of-freedom (dof).
  - 3 dof for translations in each cartesian axis (X,Y,Z)
  - 3 dof for rotations about each cartesian axis (X,Y,Z)

LSQ6



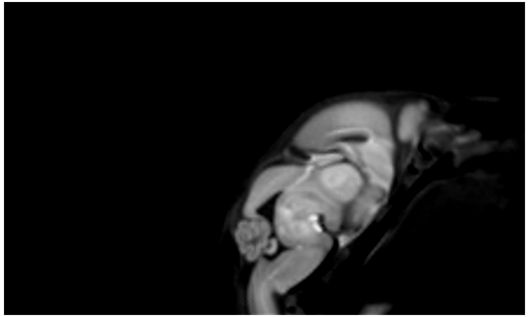
source



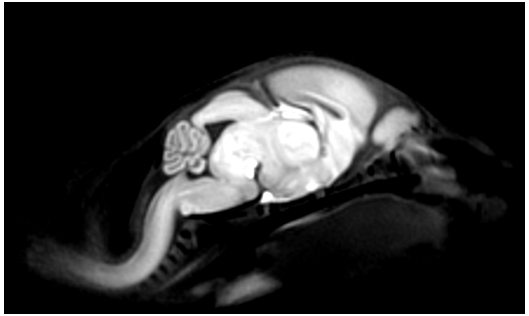
target

```
minctracc -lsq6 <source>.mnc <target>.mnc <transform>.xfm
```

LSQ6



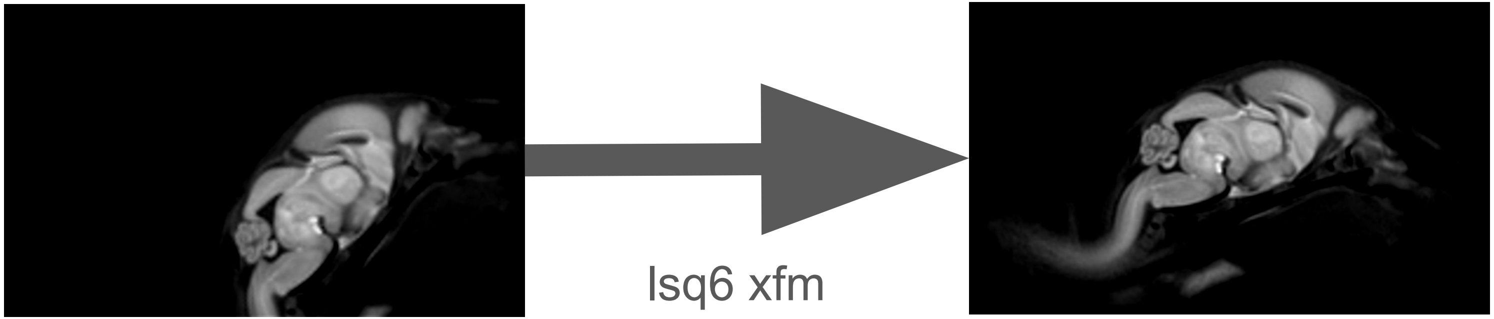
source



target

```
minctracc -lsq6 <source>.mnc <target>.mnc <lsq6>.xfm
```

LSQ6



lsq6 xfm

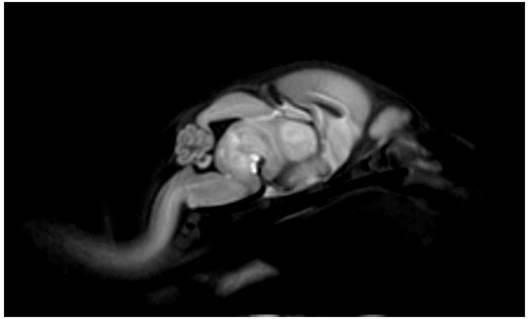
source

lsq6 source

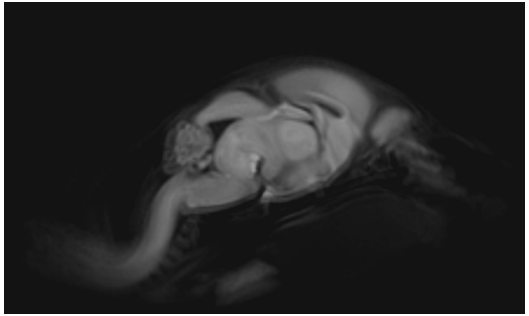
```
mincresample -like <target>.mnc <source>.mnc \

   -transform <lsq6>.xfm <lsq6 source>.mnc
```

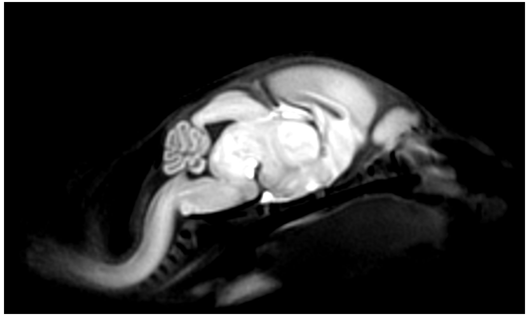
LSQ6



lsq6 source



overlay



target

What transformations should we do?

LSQ12 - affine transformations

- **L**east **SQ**uares with **12** degrees-of-freedom (dof).

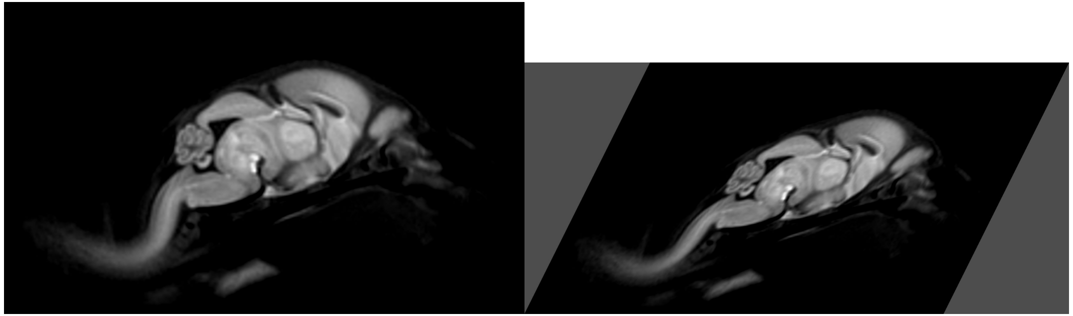
LSQ12

- **L**east **SQ**uares with **12** degrees-of-freedom (dof).
  - 3 dof for translations in each cartesian axis (X,Y,Z)
  - 3 dof for rotations about each cartesian axis (X,Y,Z)
  - 3 dof for scaling in each cartesian axis (X,Y,Z)

Scaling

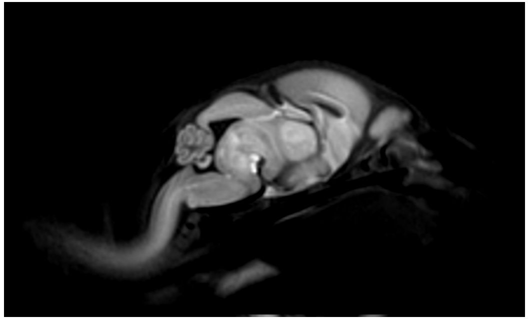
LSQ12

- **L**east **SQ**uares with **12** degrees-of-freedom (dof).
    - 3 dof for translations in each cartesian axis (X,Y,Z)
    - 3 dof for rotations about each cartesian axis (X,Y,Z)
    - 3 dof for scaling in each cartesian axis (X,Y,Z)
    - 3 dof for shearing about certain cartesian axis pairs (XY,XZ,YZ)

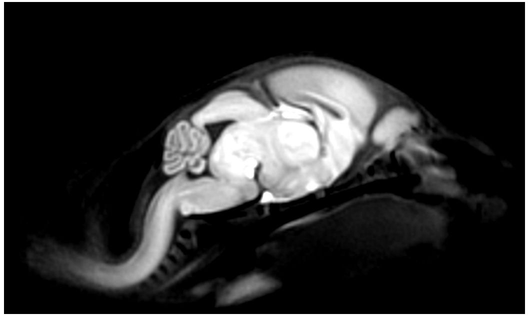


Shearing

LSQ12



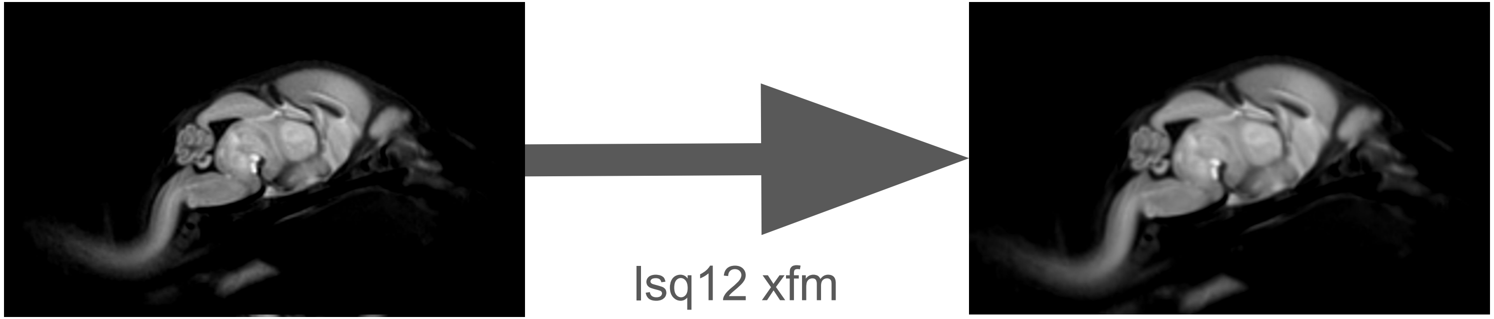
lsq6 source



target

```
minctracc -lsq12 <lsq6 source>.mnc <target>.mnc <lsq12>.xfm
```

LSQ12



lsq12 xfm

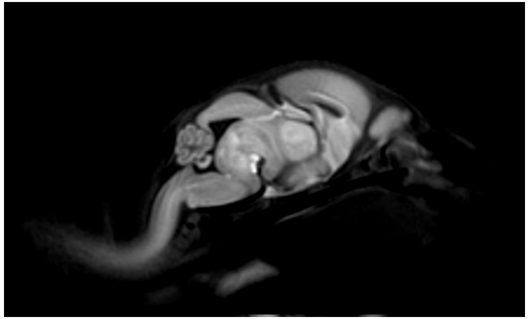
lsq6 source

lsq12 source

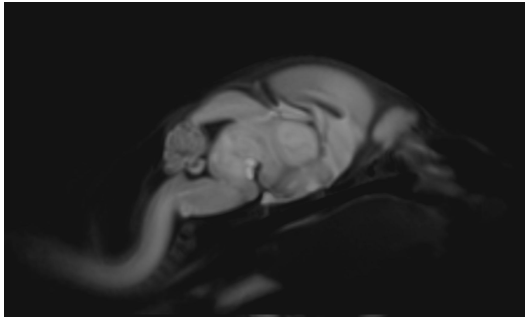
```
mincresample -like <target>.mnc <lsq6 source>.mnc \

   -transform <lsq12>.xfm <lsq12 source>.mnc
```

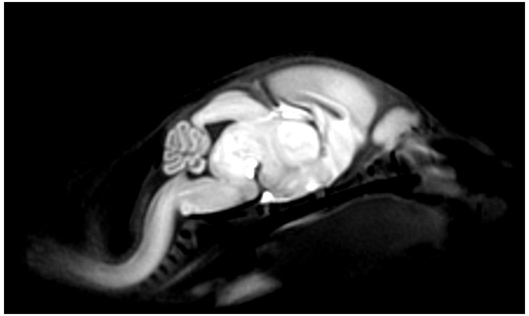
LSQ12



lsq12 source

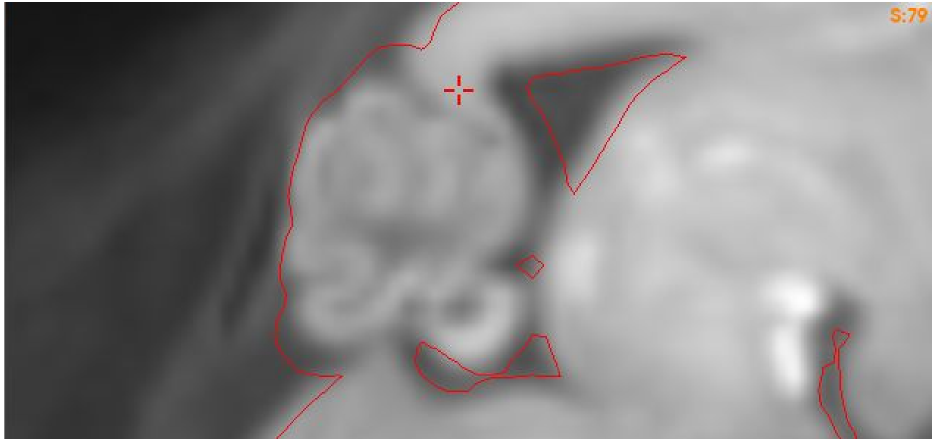


overlay

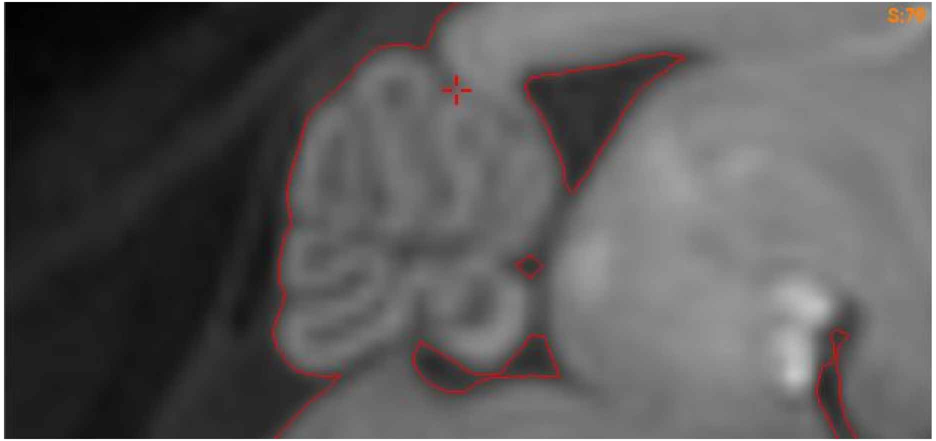


target

LSQ12

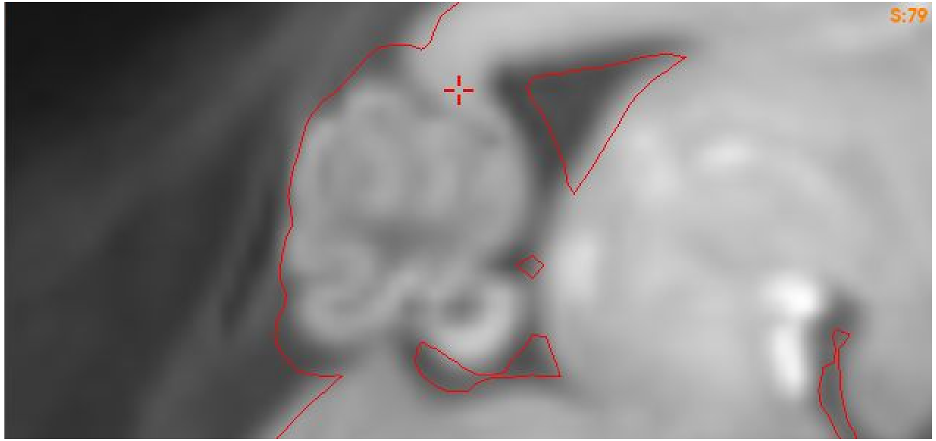


lsq12 source

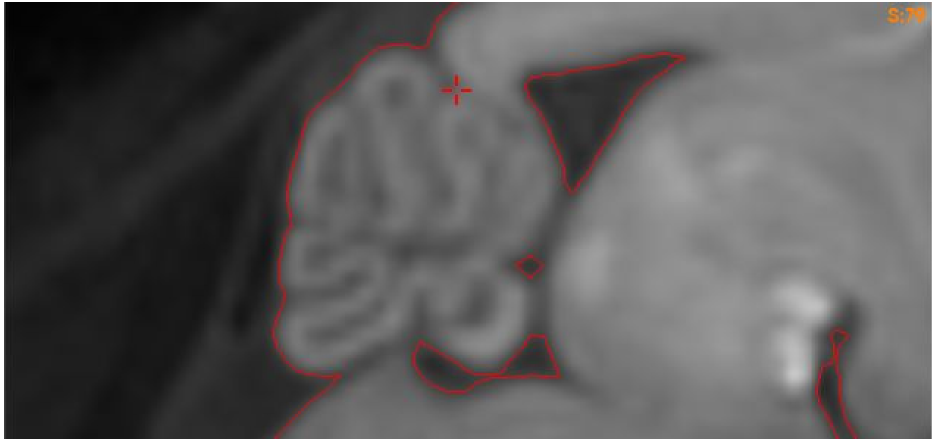


target

LSQ12



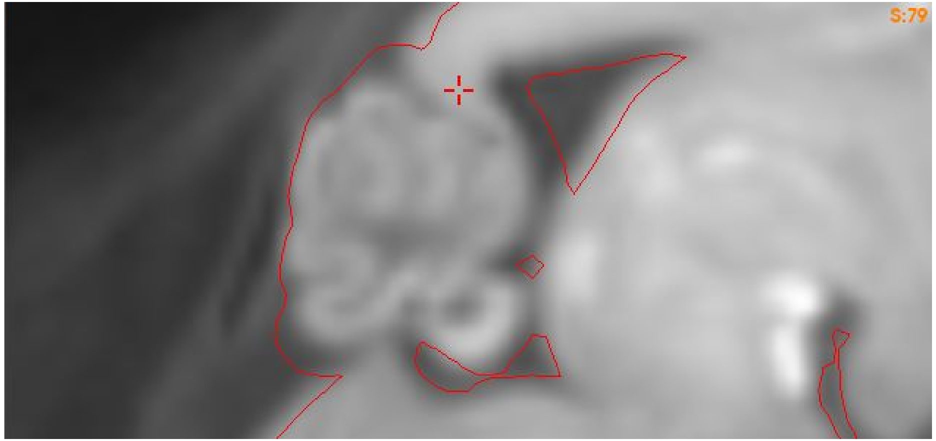
lsq12 source



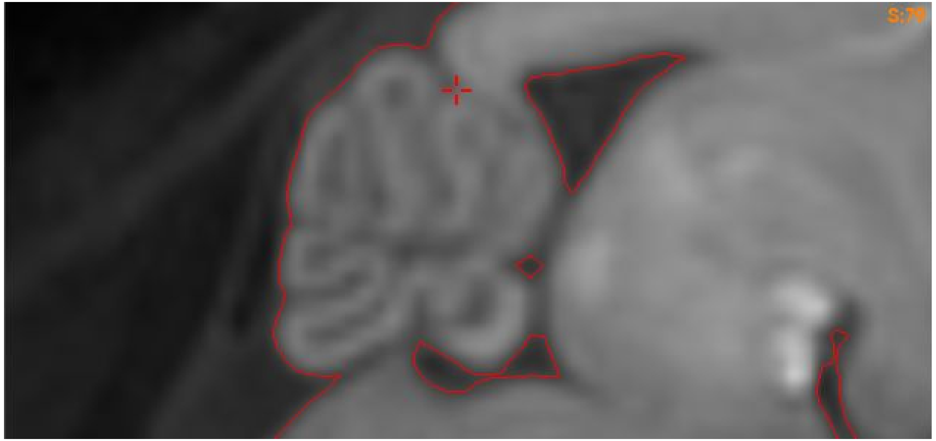
target

- Affine transformations are limited in alignment.
  - Changing alignment of one voxel, influence alignment in other voxels

LSQ12



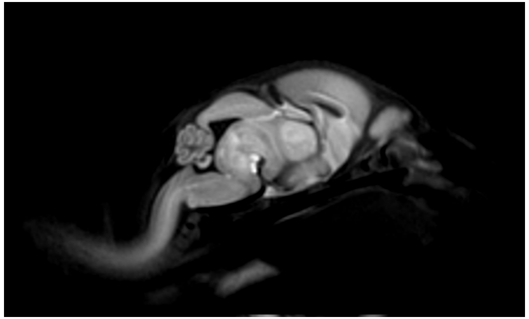
lsq12 source



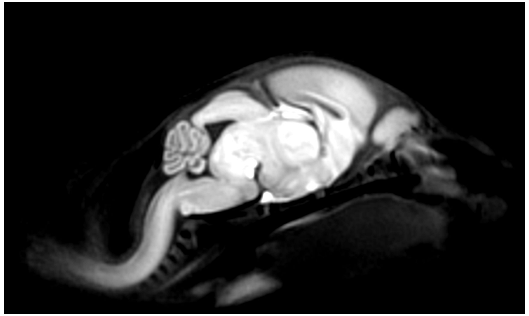
target

- Non-linear transformations address this issue.
  - Each voxel can gets its own personal transformation

Non-Linear transformations

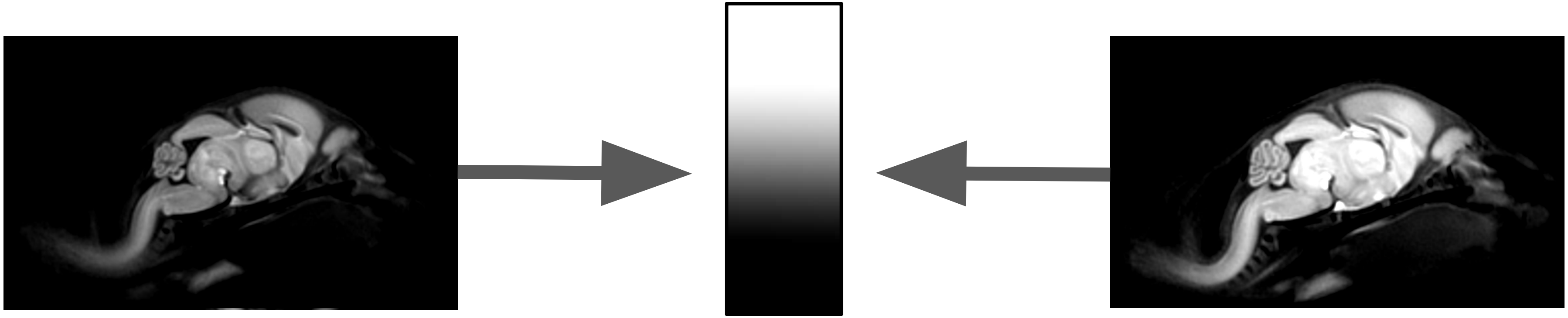


lsq12 source



target

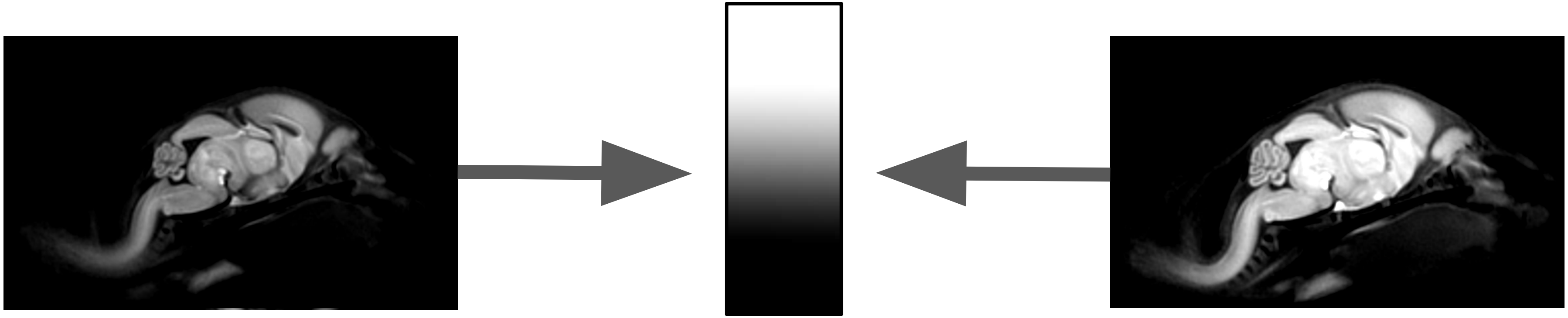
Non-Linear transformations



lsq12 source

target

Non-Linear transformations

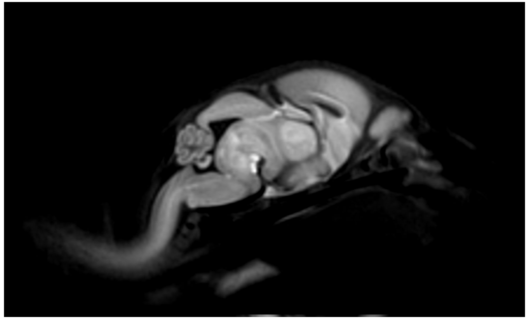


lsq12 source

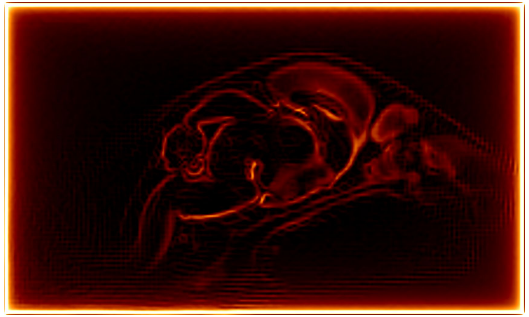
target

- Regularization - prevent non-sensical parameters when there are too many degrees-of-freedom. Demand voxels behave well:
    - Thou shall not move too far (elastic regularization)
    - Thou shall not move too dissimilar to thy neighbours (spatial regularization)
    - Thou shall be able to return to whence you came (invertibility)

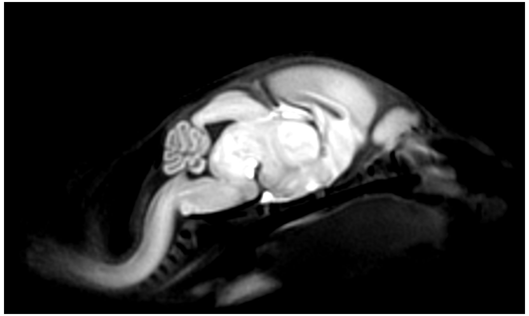
Non-Linear transformations



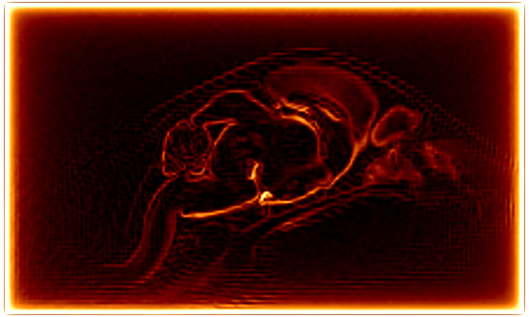
lsq12 source



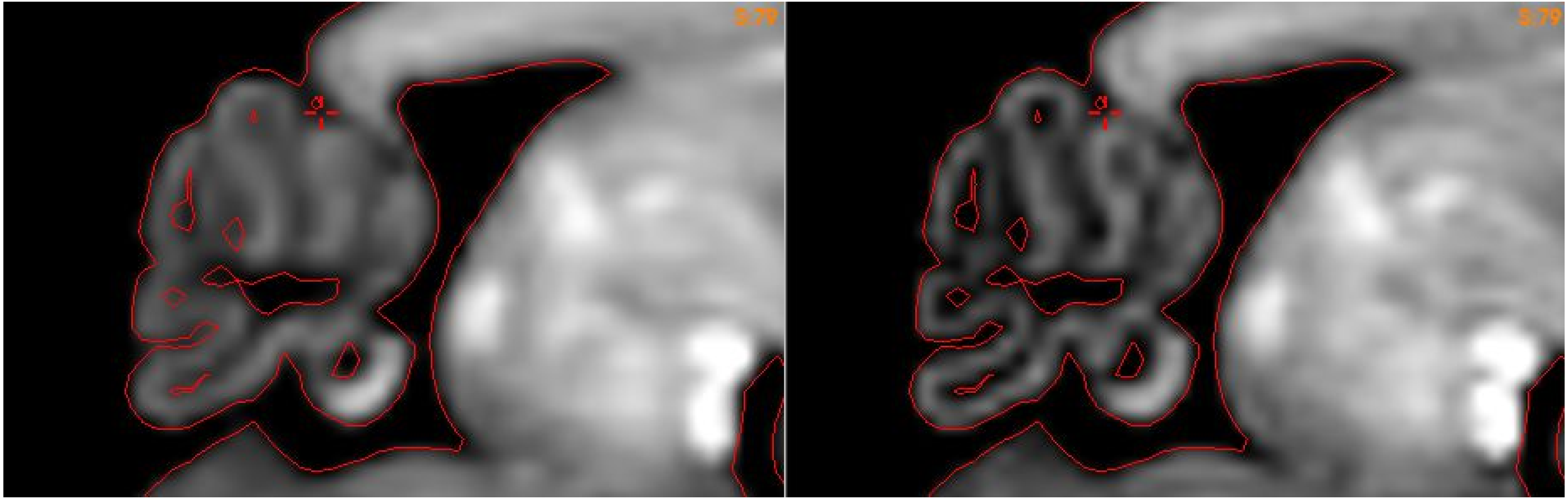
Alignment can also be performed on image gradients (i.e. edges)



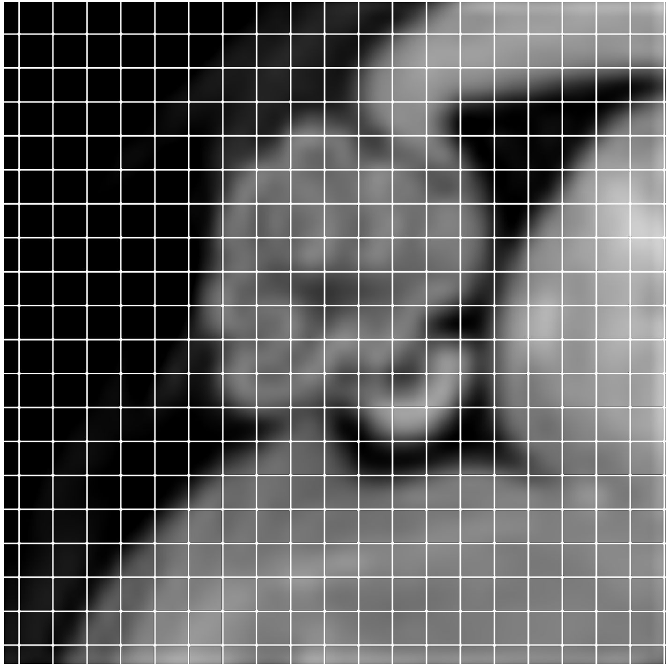
target



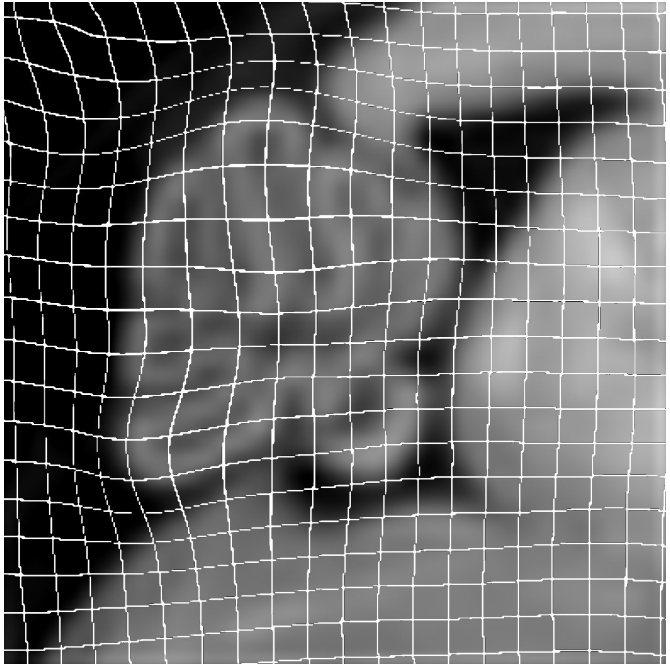
Non-Linear transformations



Non-Linear transformations

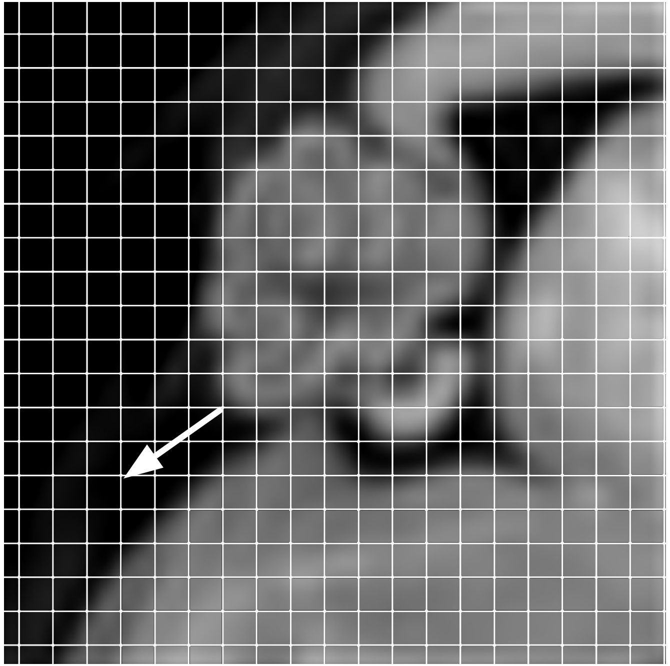


source

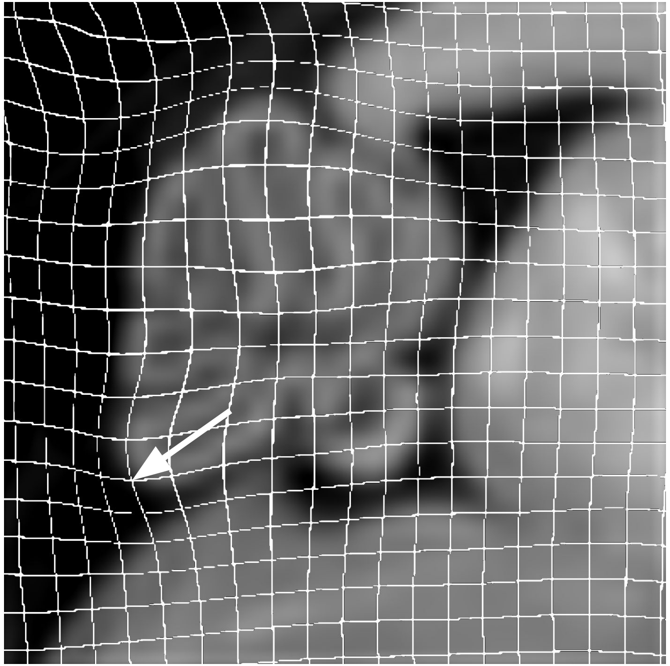


target

Non-Linear transformations



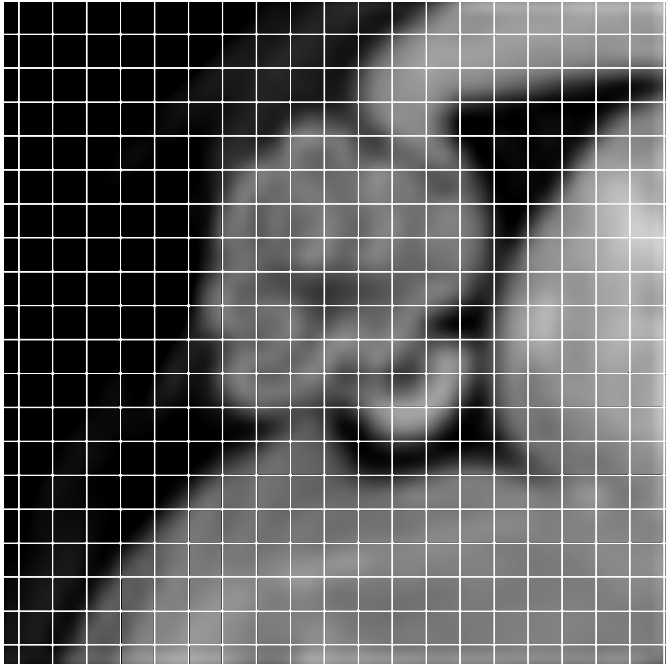
source



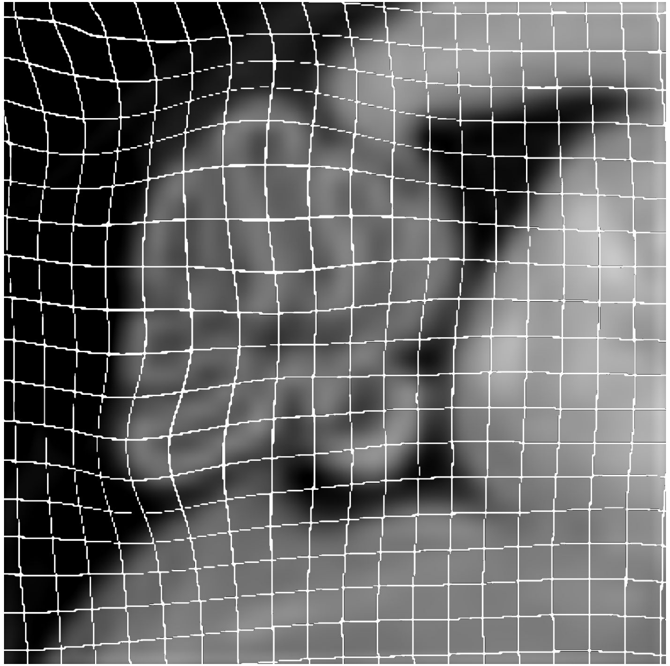
target

Deformations fields: at every source point, vector indicates target point

Non-Linear transformations



source



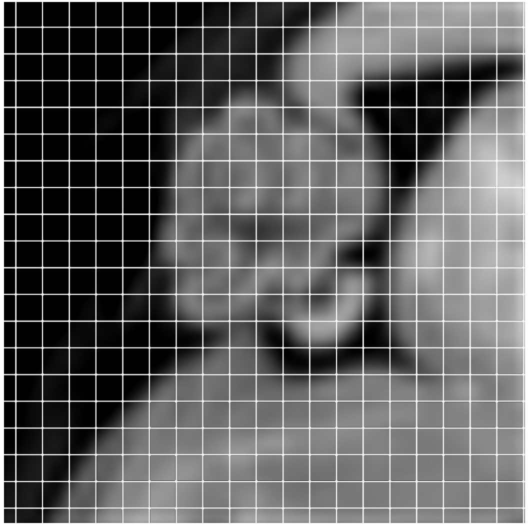
target

How do we quantify volumetric changes caused by transformations?

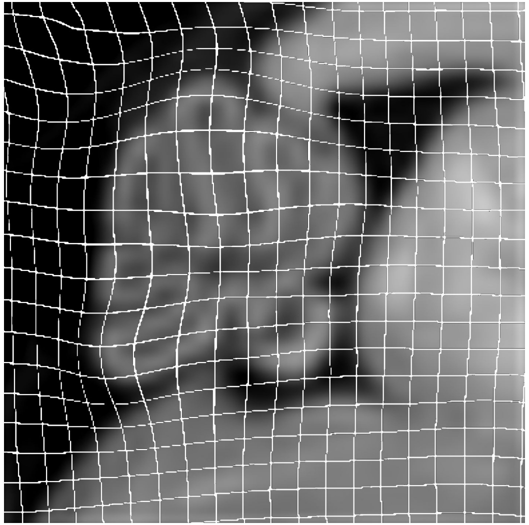
Determinants

- Deformations are vector fields
    - At every coordinate, there is a 3-component vector telling the coordinate where to go after transformation.
- Determinants are scalar fields
    - For every voxel, there is a scalar value indicating how much the volume has expanded (value>1) or contracted (value<1)

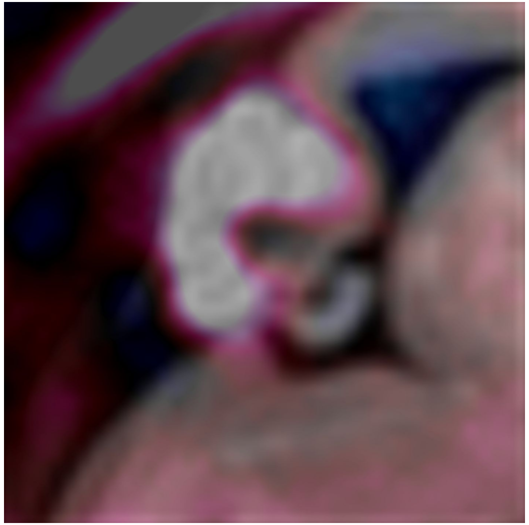
Determinants



source

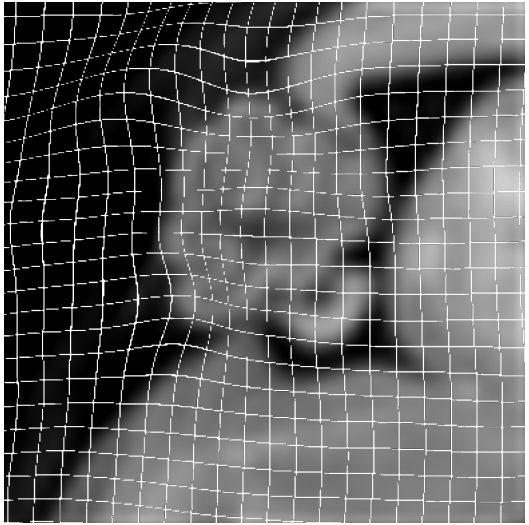


target

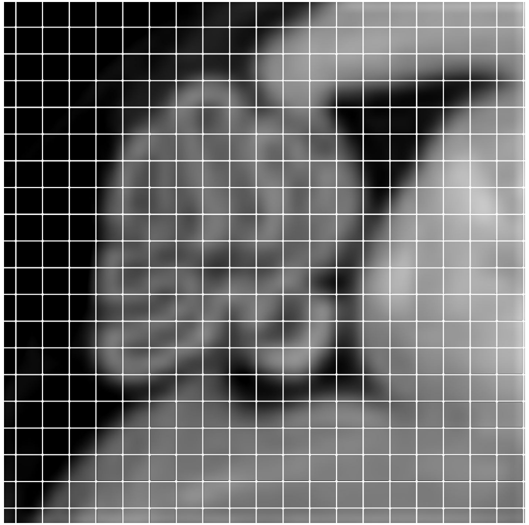


absolute determinant

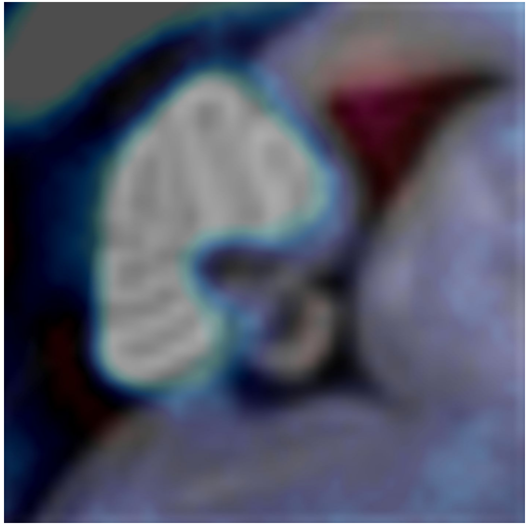
Inverse Determinants



source

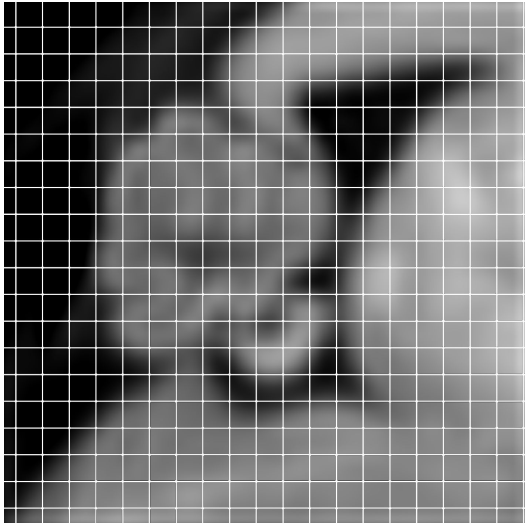


target

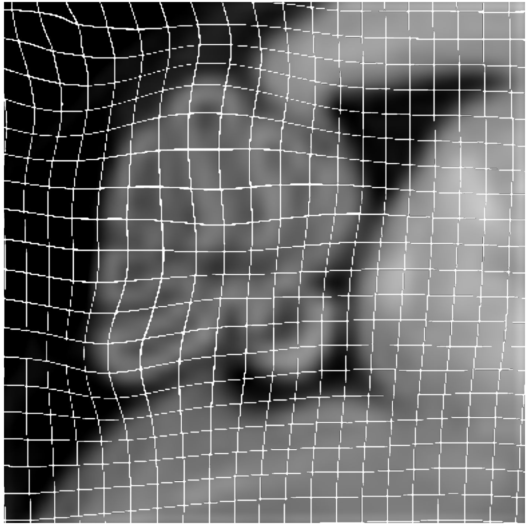


Inverse absolute determinant

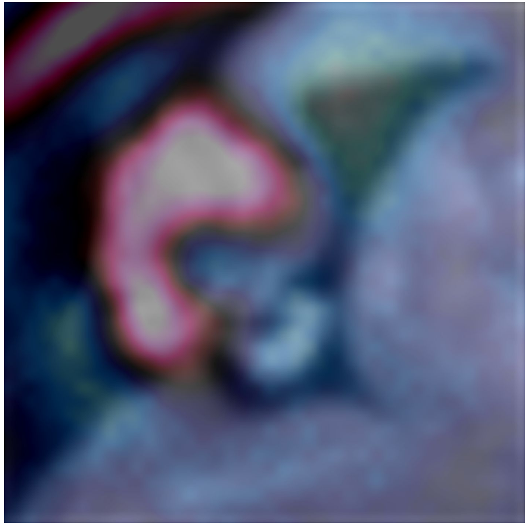
Determinants



lsq12 source

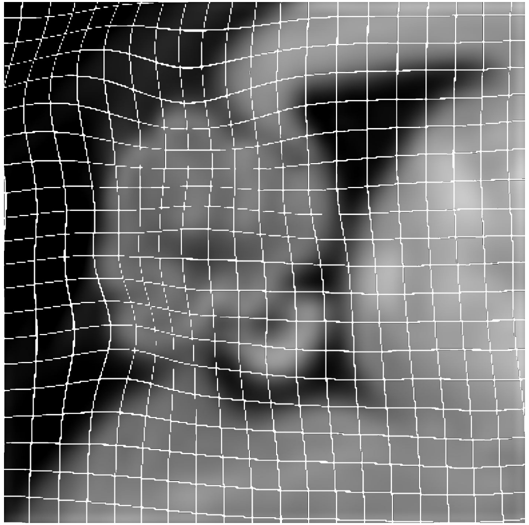


target

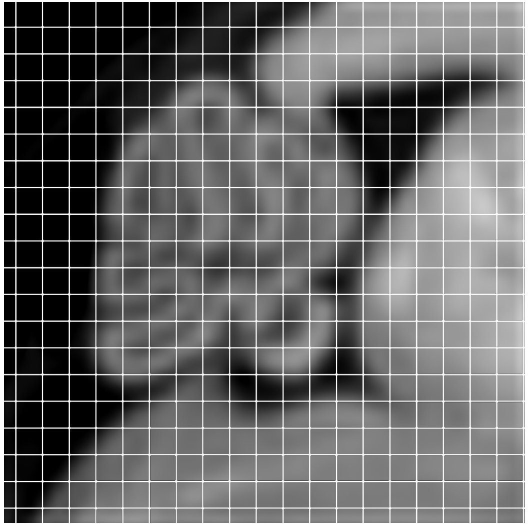


relative determinant

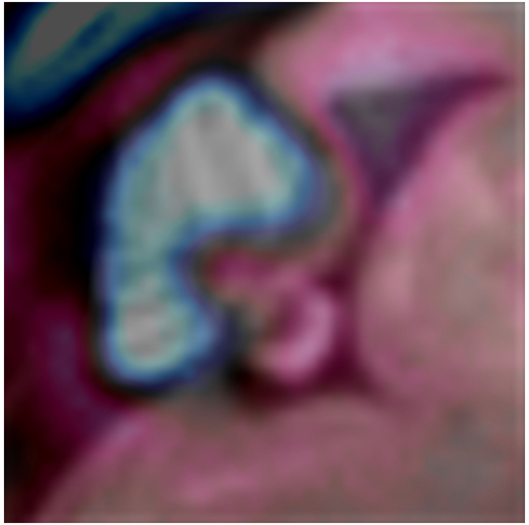
Inverse Determinants



lsq12 source

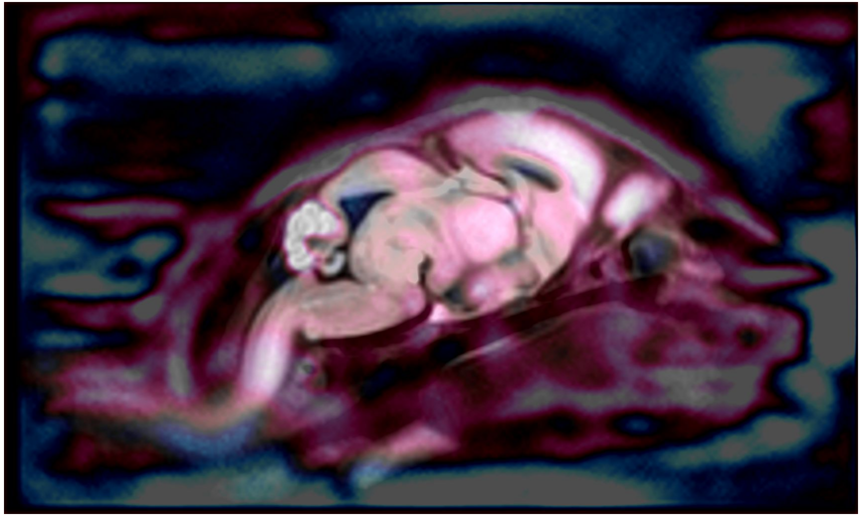


target



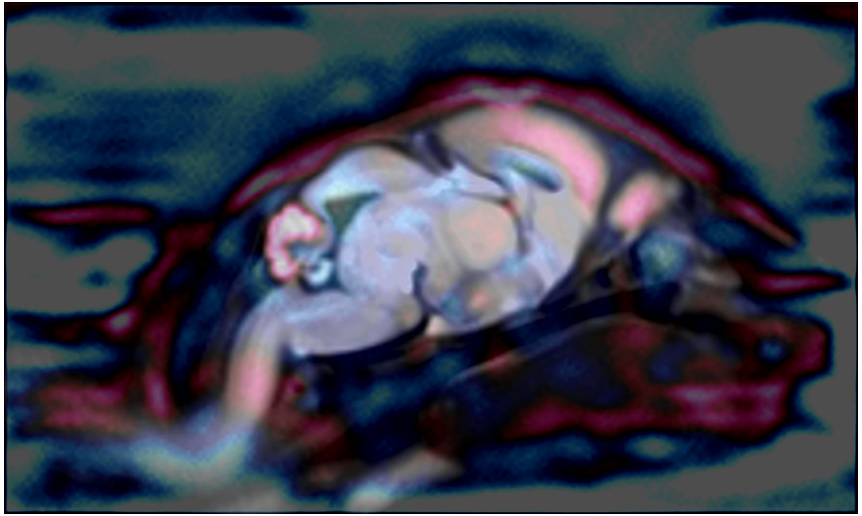
Inverse relative determinant

Determinants



absolute determinant

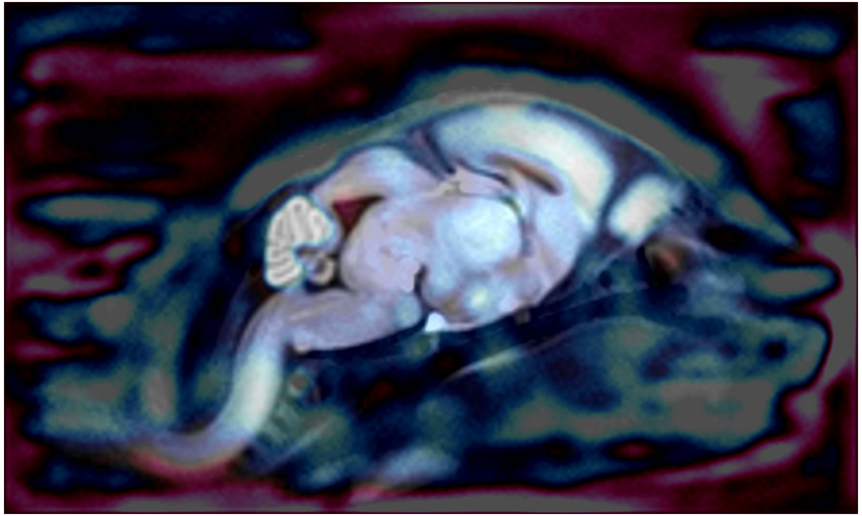
(source space)



relative determinant

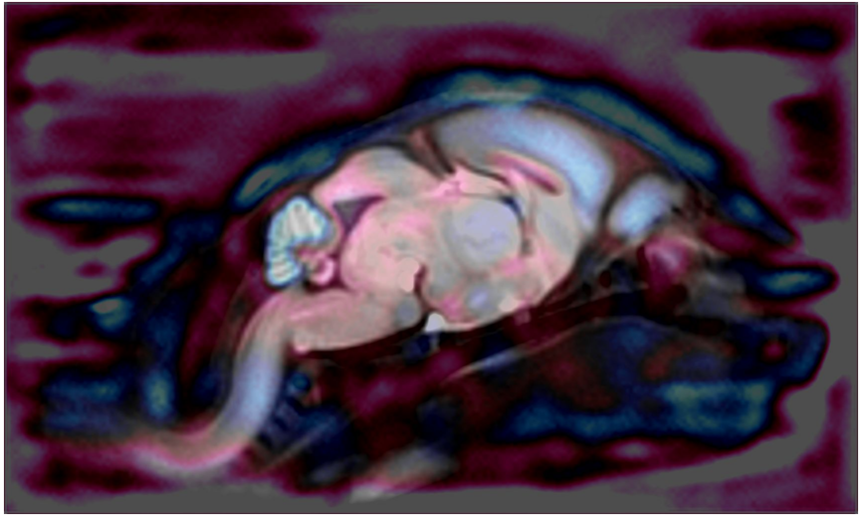
(source lsq12 space)

Inverse Determinants



Inverse absolute determinant

(target space)



Inverse relative determinant

(target space)

Determinants

- Rigid (lsq6) transformations have determinant 1 everywhere
- Affine (lsq12) transformations have constant determinant everywhere
- Determinants of overall transformations (lsq12+nlin) are called **absolute determinants**:
    - Sensitive to overall brain size differences
    - Contain all information regarding volumetry
    - Volume = (resolution)^3 * [absolute determinants]
- Determinants of non-linear transformations are called **relative determinants**:
    - Sensitive to subtle neuroanatomy differences
    - Contains only information relative to overall brain size
    - Volume/[Brain volume] ≅ (resolution)^3 * [relative determinants]

Pydpiper

- Run registration
  - Input *.mnc files with images you will to align
  - Output:
    - consensus average
    - Determinants
    - Transformations
    - Resampled images
    - … important intermediate files
    - and plenty of tmp files

```
MBM.py \
--pipeline-name=[pipeline name]      \
--init-model=[init model]            \
--registration-method=ANTS           \
--lsq6-centre-estimation             \
--lsq6-protocol=[proc_6.csv]         \
--num-executors=[x]                  \
--lsq12-protocol=[proc_12.csv]       \
--no-run-maget                       \
--maget-no-mask                      \
--no-common-space-registration       \
--files [file1] [file2] …
```

Pydpiper

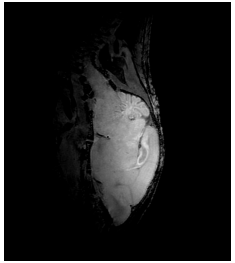
- Init (Initial) models are useful for standarization.
- Typically, scanner data is unmasked, in a different orientation, slightly anisotropic.
- We can use init models to apply transformations to native scanner images.

```
MBM.py \
--pipeline-name=[pipeline name]       \
--init-model=[init model]             \
--registration-method=ANTS            \
--lsq6-centre-estimation              \
--lsq6-protocol=[proc_6.csv]          \
--num-executors=[x]                   \
--lsq12-protocol=[proc_12.csv]        \
--no-run-maget                        \
--maget-no-mask                       \
--no-common-space-registration        \
--files [file1] [file2] …
```

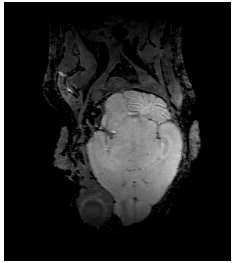
Initial Models

Bruker Space

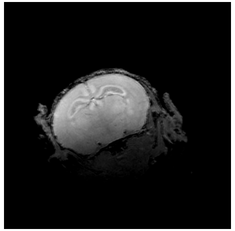
X left-right (res -0.07483)



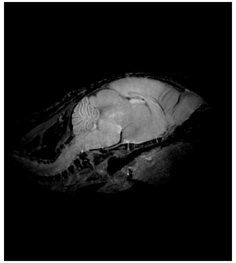
Y inf-sup (res -0.07483)



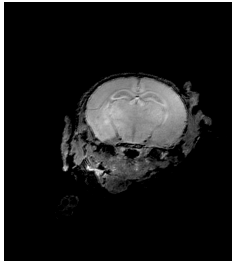
Z ant-post (res -0.07485)



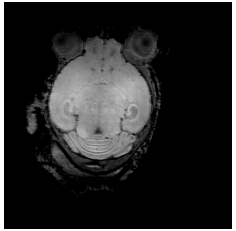
MICe Space



X left-right (res 0.075)



Y ant-post (res 0.075)



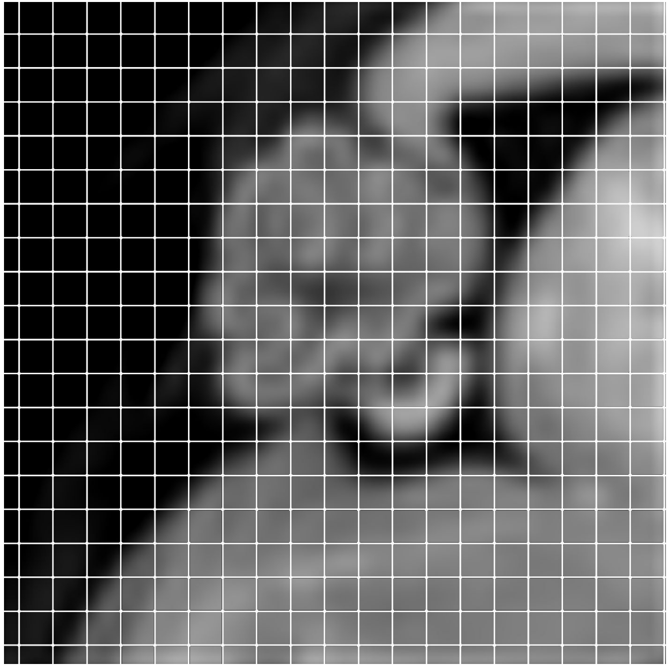
Z inf-sup (res 0.075)

Pydpiper

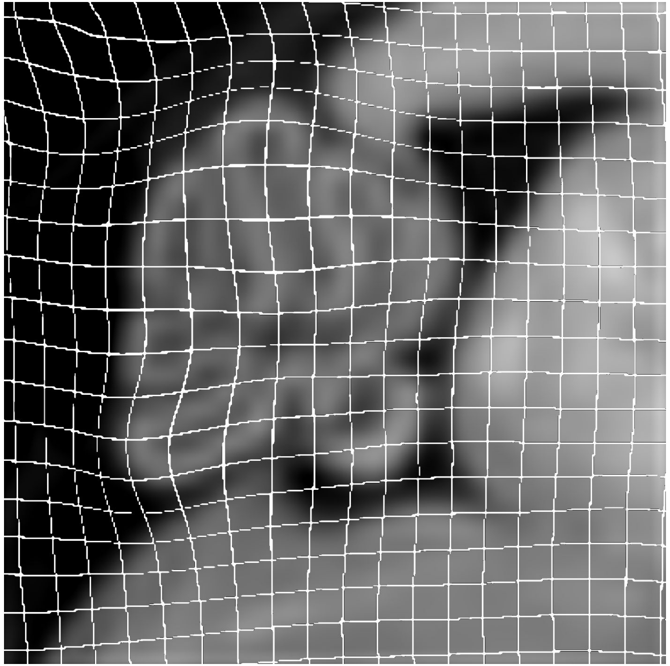
- Registration methods can be:
  - ANTS
  - minctracc/ANIMAL

```
MBM.py \
--pipeline-name=[pipeline name]       \
--init-model=[init model]             \
--registration-method=ANTS            \
--lsq6-centre-estimation              \
--lsq6-protocol=[proc_6.csv]          \
--num-executors=[x]                   \
--lsq12-protocol=[proc_12.csv]        \
--no-run-maget                        \
--maget-no-mask                       \
--no-common-space-registration        \
--files [file1] [file2] …
```

minctracc



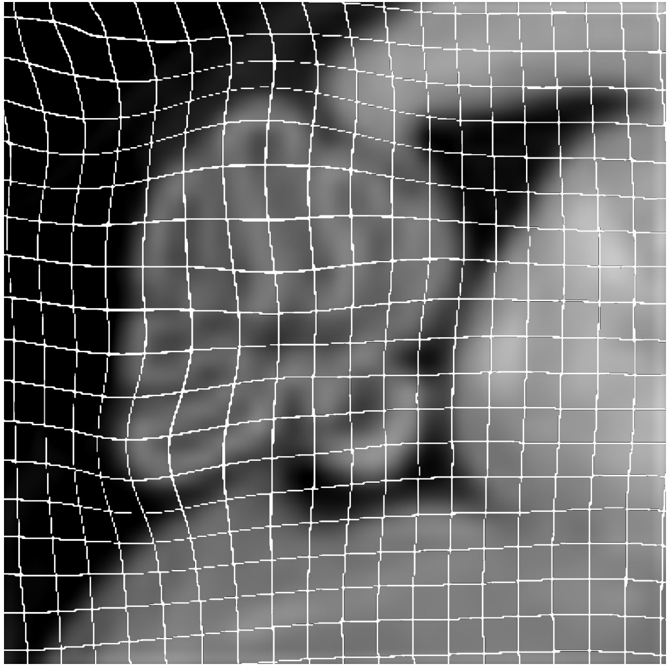
source



target

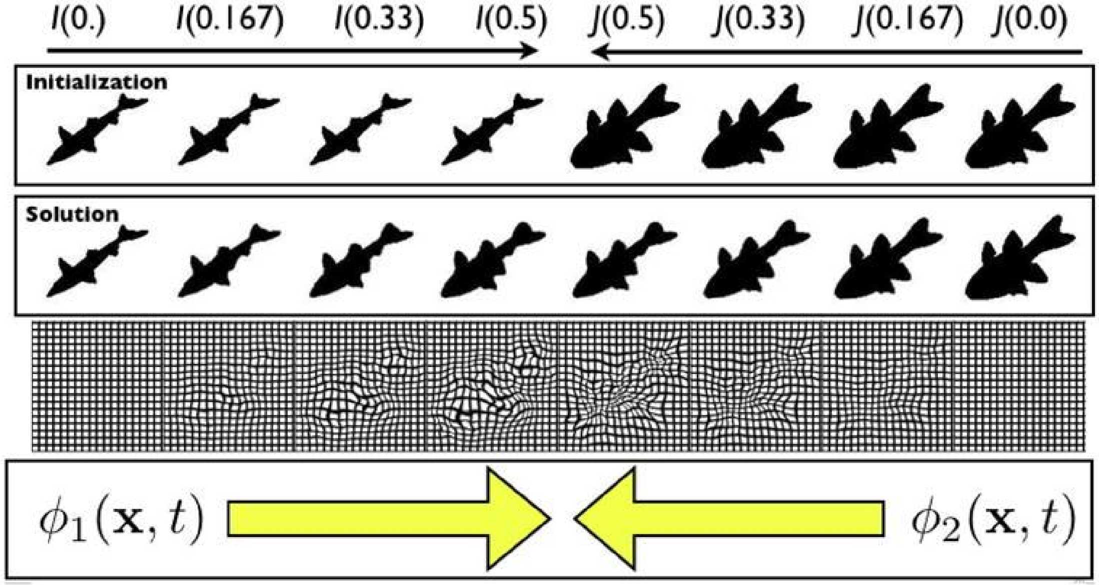
minctracc

- Penalize deformation based on stiffness



target

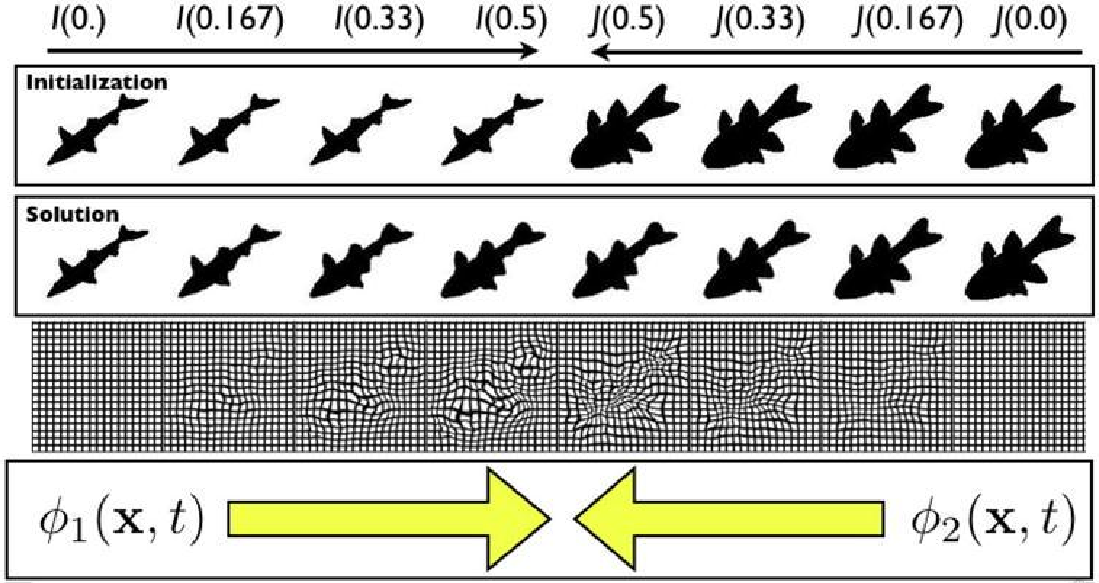
mincANTS



Avants, B. B., Epstein, C. L., Grossman, M., & Gee, J. C. (2008). Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Medical image analysis*,*12*(1), 26-41.

mincANTS

- Deform both source and target and meet in the middle.
- The resulting total transformation is guaranteed to be invertible.



Avants, B. B., Epstein, C. L., Grossman, M., & Gee, J. C. (2008). Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Medical image analysis*,*12*(1), 26-41.

Pydpiper

- --lsq6-protocol requires csv listing important parameters for lsq6 registration
- Parameters include:
    - blur: the fwhm of the gaussian convolved to blur images
    - step: resolution of intermediate files for alignment
    - gradient: whether to use gradient information
    - simplex: Volume of the tetrahedron
    - w_translations: to what extent do you prefer translations over rotations

```
MBM.py \
--pipeline-name=[pipeline name]      \
--init-model=[init model]            \
--registration-method=ANTS           \
--lsq6-centre-estimation             \
--lsq6-protocol=[proc_6.csv]         \
--num-executors=[x]                  \
--lsq12-protocol=[proc_12.csv]       \
--no-run-maget                       \
--maget-no-mask                      \
--no-common-space-registration       \
--files [file1] [file2] …
```

Pydpiper

- --lsq12-protocol requires csv listing important parameters for lsq12 registration
- Parameters include:
    - blur: the fwhm of the gaussian convolved to blur images
    - step: resolution of intermediate files for alignment
    - gradient: whether to use gradient information
    - simplex: Volume of the tetrahedron
    - w_translations: to what extent do you prefer translations over rotations

```
MBM.py \
--pipeline-name=[pipeline name]      \
--init-model=[init model]            \
--registration-method=ANTS           \
--lsq6-centre-estimation             \
--lsq6-protocol=[proc_6.csv]         \
--num-executors=[x]                  \
--lsq12-protocol=[proc_12.csv]       \
--no-run-maget                       \
--maget-no-mask                      \
--no-common-space-registration       \
--files [file1] [file2] …
```

Pydpiper

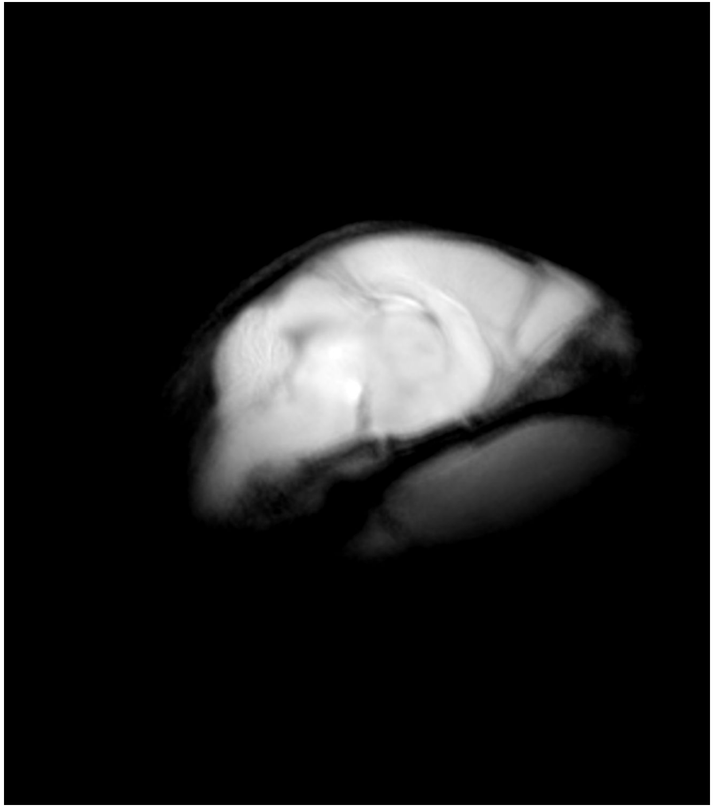
- Minc Files!

```
MBM.py \
--pipeline-name=[pipeline name]       \
--init-model=[init model]             \
--registration-method=ANTS            \
--lsq6-centre-estimation              \
--lsq6-protocol=[proc_6.csv]          \
--num-executors=[x]                   \
--lsq12-protocol=[proc_12.csv]        \
--no-run-maget                        \
--maget-no-mask                       \
--no-common-space-registration        \
--files [file1] [file2] …
```

Pydpiper outputs

Outputs:

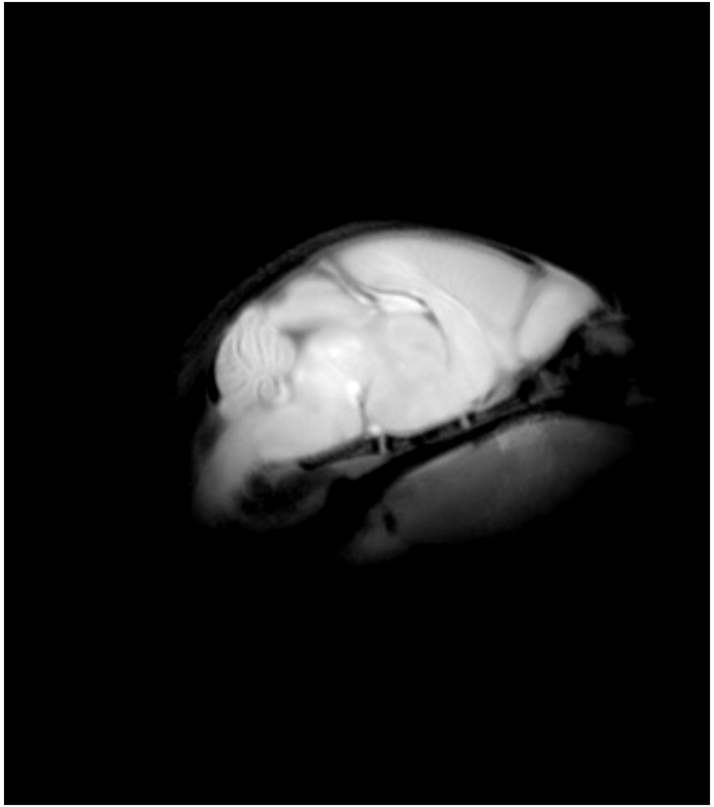
- Averages
  - lsq6 average
  - lsq12 average
  - nlin1 average
  - nlin2 average
  - nlin3 average



Pydpiper outputs

Outputs:

- Averages
  - lsq6 average
  - lsq12 average
  - nlin1 average
  - nlin2 average
  - nlin3 average



Pydpiper outputs

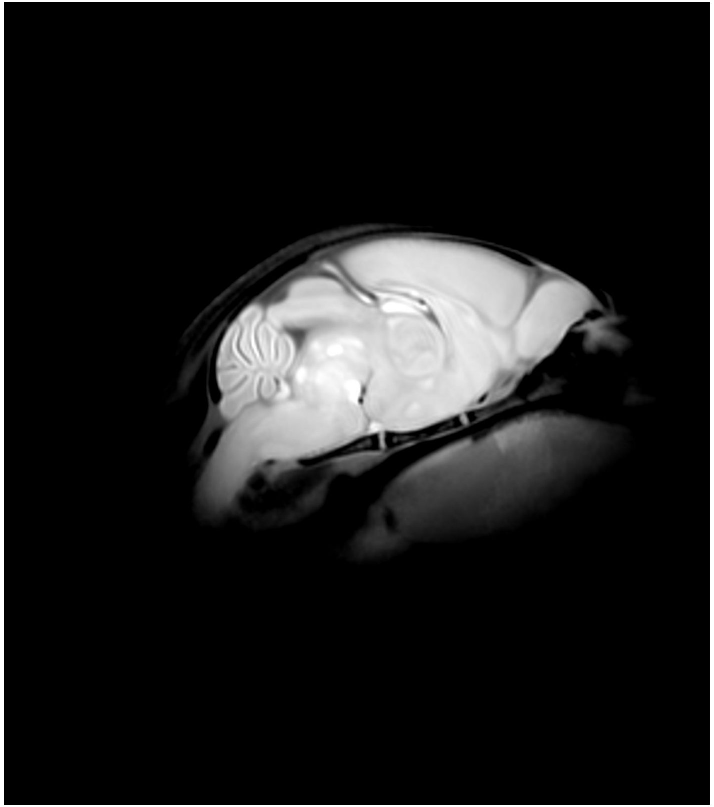
Outputs:

- Averages
  - lsq6 average
  - lsq12 average
  - nlin1 average
  - nlin2 average
  - nlin3 average

Pydpiper outputs

Outputs:

- Averages
  - lsq6 average
  - lsq12 average
  - nlin1 average
  - nlin2 average
  - nlin3 average



Pydpiper outputs

Outputs:

- Averages
  - lsq6 average
  - lsq12 average
  - nlin1 average
  - nlin2 average
  - nlin3 average

Pydpiper outputs

Outputs:

- Averages
  - lsq6 average
  - lsq12 average
  - nlin1 average
  - nlin2 average
  - nlin3 average
- transformations
- determinants

XFM Files

- Transformations are saved in XFM files
- These are simple text files and can be opened in any text editor
- For linear transformations, xfm files can be generated using the ‘param2xfm’ utility
    - You can supply scales, shears, translations, and rotations
- Conversely, for linear transformations, the ‘xfm2param’ utility outputs the transformations in terms of scales, shears, translations, and rotations.

XFM Files

- Example xfm file

```
MNI Transform File
%Sun Aug 20 11:04:15 2017>>> param2xfm -center
3.5925 5.035 3.03 -rotations 30 0 0 perturb_p03.xfm
%(mni_autoreg 0.99.60)

Transform_Type = Linear;
Linear_Transform =
 1 0 0 0
 0 0.866025388240814 -0.5 2.18956184387207
 0 0.5 0.866025388240814 -2.11155676841736;
```

XFM Files

```
1 0 0 0
0 0.866025388240814 -0.5 2.18956184387207
0 0.5 0.866025388240814 -2.11155676841736
```

XFM Files

1	0	0	0
0	0.866	-0.5	2.19
0	0.5	0.866	-2.11

XFM Files

Translation

1	0	0	0
0	0.866	-0.5	2.19
0	0.5	0.866	-2.11

XFM Files

	x’	y’	z’	
x	1	0	0	0
y	0	0.866	-0.5	2.19
z	0	0.5	0.866	-2.11

XFM Files

- Example xfm file (lsq6)

```
MNI Transform File
%Sun Aug 20 11:04:15 2017>>> param2xfm -center
3.5925 5.035 3.03 -rotations 30 0 0 perturb_p03.xfm
%(mni_autoreg 0.99.60)

Transform_Type = Linear;
Linear_Transform =
 1 0 0 0
 0 0.866025388240814 -0.5 2.18956184387207
 0 0.5 0.866025388240814 -2.11155676841736;
```

XFM Files

- Example xfm file (lsq12)

```
MNI Transform File
% Created from tag file /var/tmp/lin_from_nlin_13903//tags.tag
% using 12 parameter linear least-squares

Transform_Type = Linear;
Linear_Transform =
 1.08337691794832 -0.0027806279034992 -0.00872909807204995 0.0373192850693193
 -0.00277798126367675 1.08036686284692 -0.0459407761578367 -0.144161788375067
 0.019892234154679 0.00542618478733193 1.09645683943094 -0.170295415034178;
```

XFM Files

1.08	0	-0.01	0.04
0	1.08	-0.05	-0.14
0.02	0.01	1.10	-0.17

Guess the transformation

1	0	0	0
0	1	0	0
0	0	1	0

Guess the transformation

1	0	0	0
0	1	0	0
0	0	1	-1

Guess the transformation

1	0	0	2
0	1	0	0
0	0	1.5	0

Guess the transformation

0.5	0	0	0
0	2	0	0
0	0	1	0

Guess the transformation

-1	0	0	0
0	1	0	0
0	0	1	0

XFM Files

- Example xfm file (w/ nlin)

```
MNI Transform File
%Sun Aug 20 17:43:45 2017>>> xfmconcat invert_lsq12.xfm
/hpf/largeprojects/MICe/dfernandes/lqiu_long_redone/lvl2reg/diff_transforms/p05_to_p03/p05_to_p03_
inverse.xfm nlin.xfm

Transform_Type = Linear;
Linear_Transform =
 0.923089701727597 0.00234289353670122 0.00745579735871461 -0.0314413241084889
 0.00165853773793207 0.92559817572826 0.0388037665860721 -0.22991538847729
 -0.01676208982699 -0.00463223062722248 0.911875208259508 -0.0233336349928444;
Transform_Type = Grid_Transform;
Displacement_Volume = nlin_grid_0.mnc;
```

XFM Files

```
> mincinfo nlin_grid_0.mnc

file: two_img_registration/nlin_grid_0.mnc
image: signed__ float -0.71406149864196777344 to 1.0232181549072265625
image dimensions: vector_dimension zspace yspace xspace
    dimension name         length         step        start
    --------------         ------         ----        -----
    vector_dimension            3      unknown      unknown
    zspace                    134         0.09           -6
    yspace                    223         0.09          -10
    xspace                    159         0.09       -7.125
```

Thank You