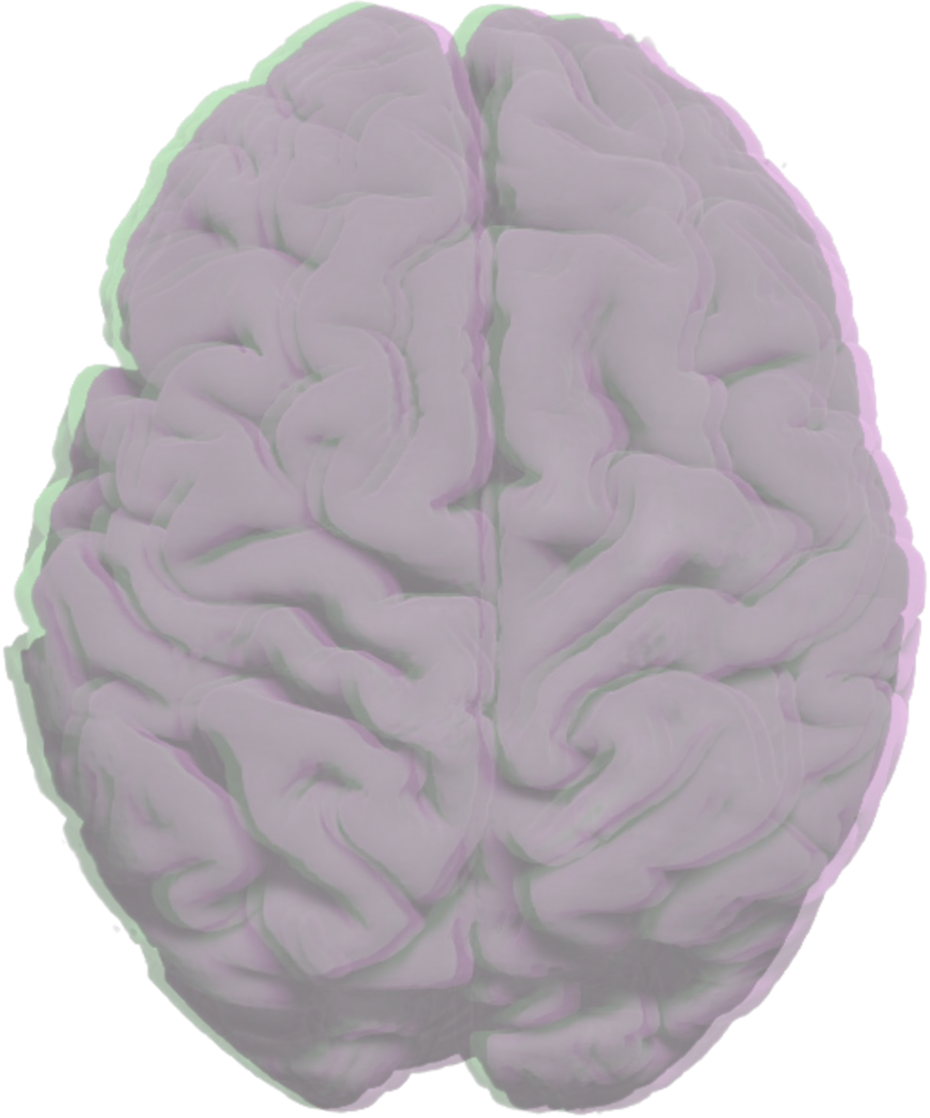
INC Summer Neuroimaging Bootcamp 2022

Coregistration

Tim Koscik, PhD
May 24, 2022



Outline

- What is coregistration?
- Why is coregistration important?
- Coregistration terms and nomenclature
- Coregistration types: rigid, affine, deformable
- Anatomy of an ANTs Registration
- Practical uses for coregistration:
  - tensor-based morphometry
  - joint label fusion

What is Coregistration?

- Coregistration is the process of aligning spatial representations of data to a shared coordinate system
- Usually done with the intent to combine data from different sensors, times, views, modalities, participants, etc., in a spatially meaningful way

Coregistration is Critical Component of Image Processing

- measure the same thing across time, samples, people, etc.
- merge and integrate data when spatial layout is important
- common spatial layout allows comparison across timepoints, individual, groups, etc.
- common spaces/regions can be labelled; labels can be mapped to individuals
- deviations from spatial layout provide information about shape differences and change, potential pathology

Terms and Nomenclature

everyone’s favourite

FIXED

- fixed coordinates, fixed space, fixed image, fixed volume, etc.
- fixed in place and form
- the target that you register to

MOVING

- moving coordinates, moving space, moving image, moving volume, etc.
- the component that moves to match the target

TRANSFORMATION

- the mapping FROM moving space TO fixed space
- Instructions for converting the moving image to the same location, shape, orientation, etc. as the fixed image

DEGREES OF FREEDOM

The number of ways the transform is allowed to move

ORIGIN

- The zero points in a coordinate system
- A reference point for transformations

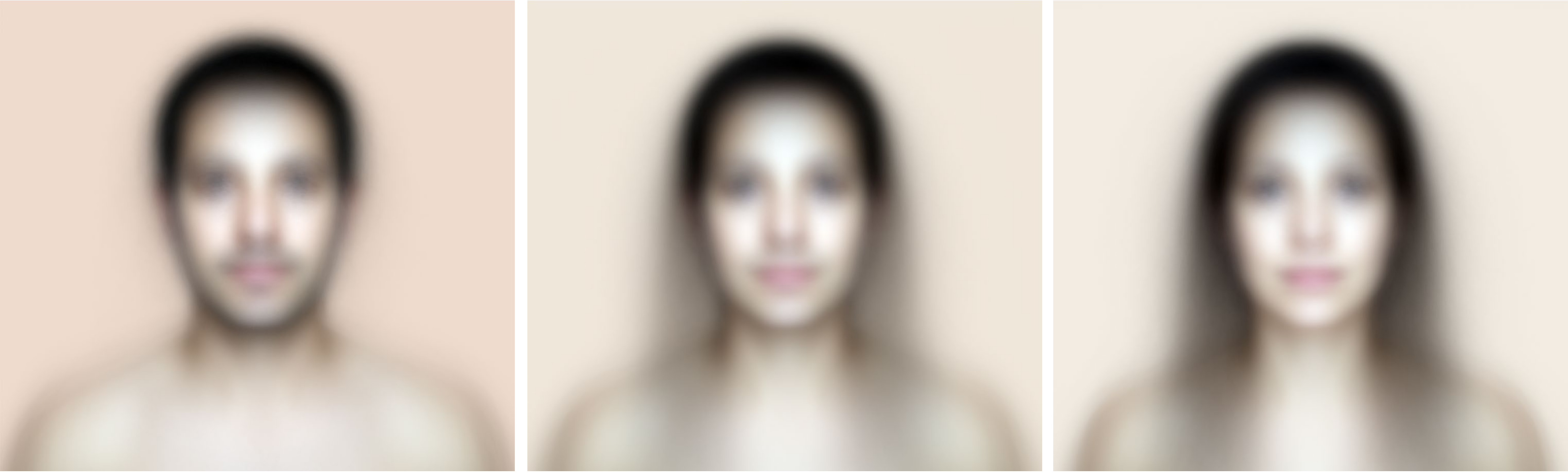
RESAMPLING

- Recalculating the values at each point using the instructions in a transformation

Example: Faces

- Human faces provide an excellent example of how registration works.
- Common features, individual layouts
  - vastly more variability in faces than brains
  - human facial features evolved to signal individual identity
    - Sheehan, M., Nachman, M. Morphological and population genomic evidence that human faces have evolved to signal individual identity. *Nature Communications, 5*, 4800 (2014). https://doi.org/10.1038/ncomms5800
- Examples from *Humanae* used with permission
  - face images remain the copyright of Angélica Dass
  - https://angelicadass.com/photography/humanae/
  - copying face images from this presentation is not permitted

- No registration, except for aligning the camera by the photographer

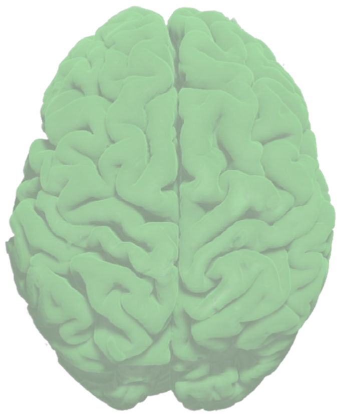


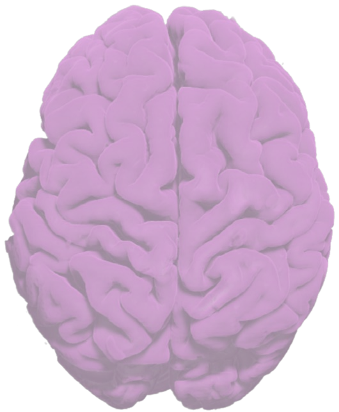
RIGID Registration

- simply moving and aligning NO shape changes
- ALL PIXELS AND VOXELS MOVE THE SAME

RIGID Registration

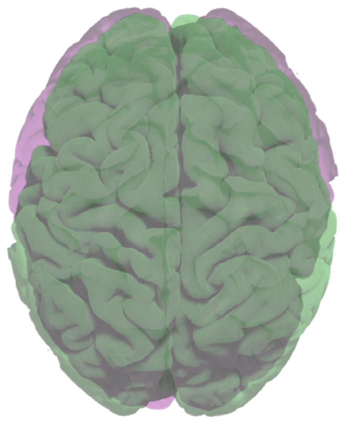
- TRANSLATION
  - shift along each linear axis in the dataset
  - all data points shift by the same amount
- 2D images (3 df)
  - Translation (X,Y): Left/Right and Up/Down
- 3D volumes (6 df)
  - Translation (X, Y, Z): Left/Right, Anterior/Posterior, Superior/Inferior



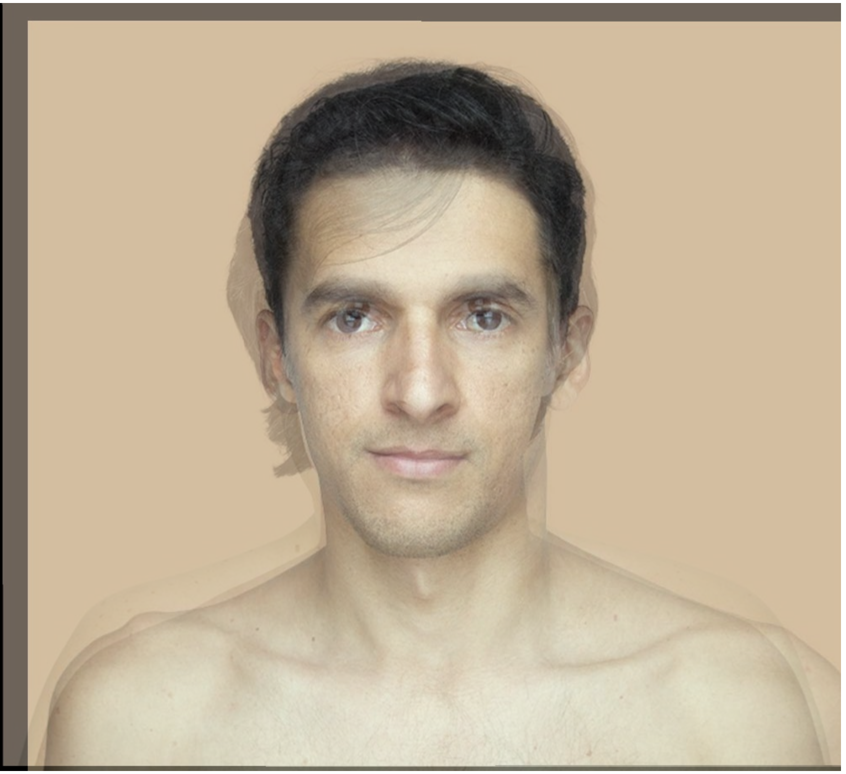


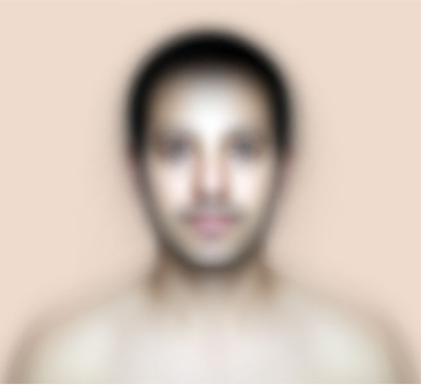
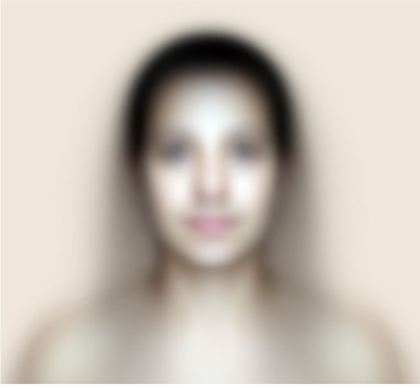
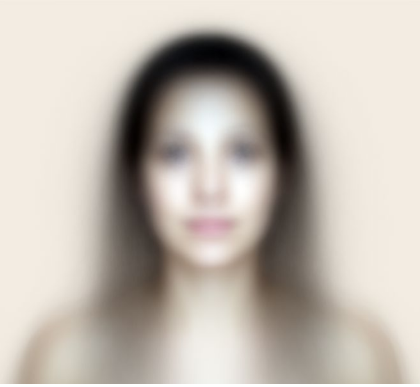
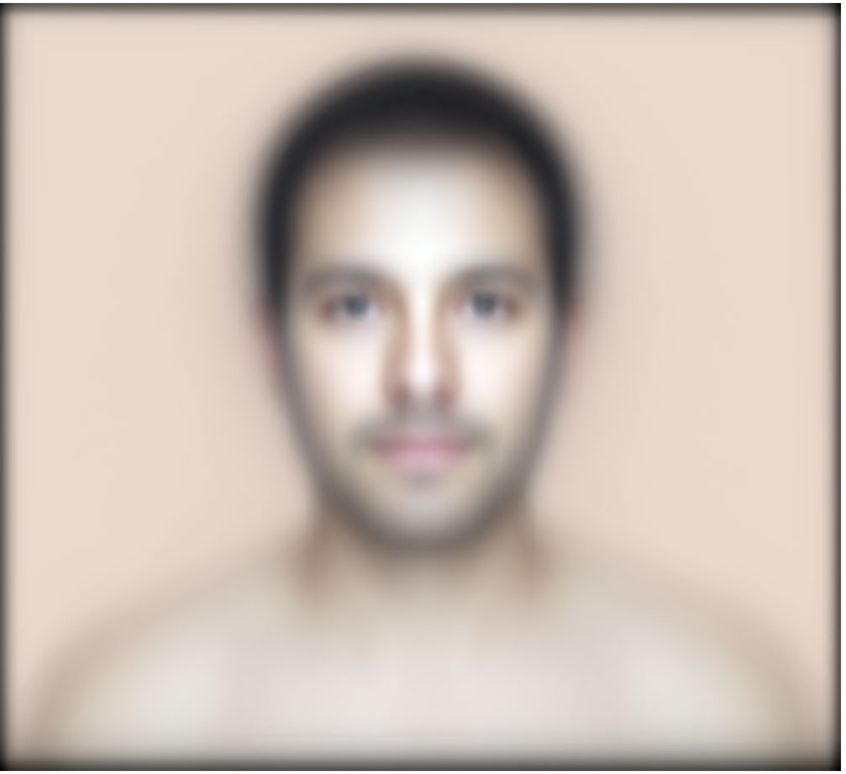
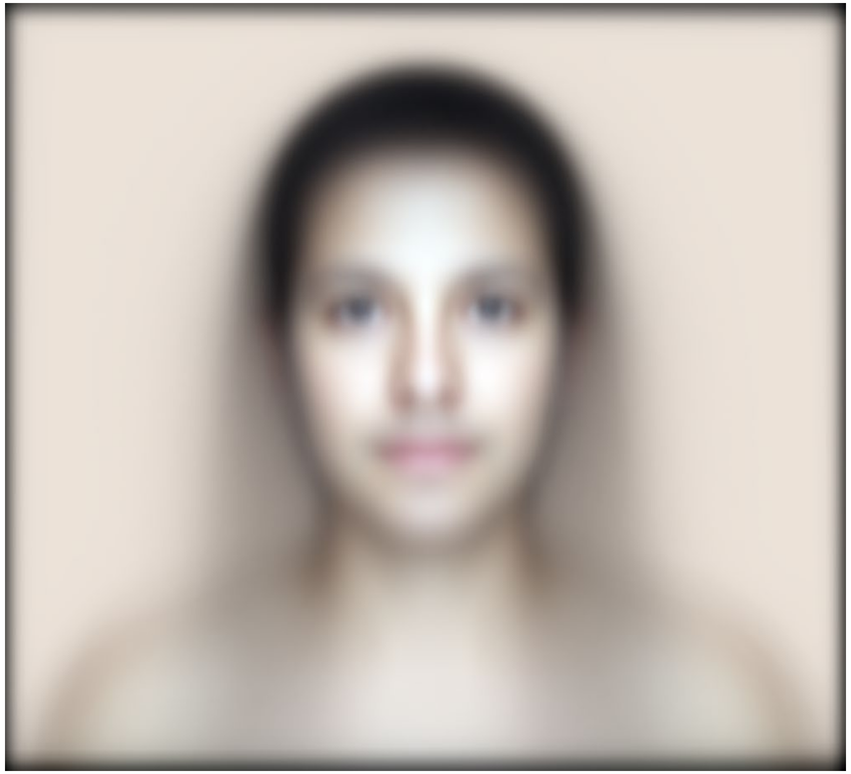
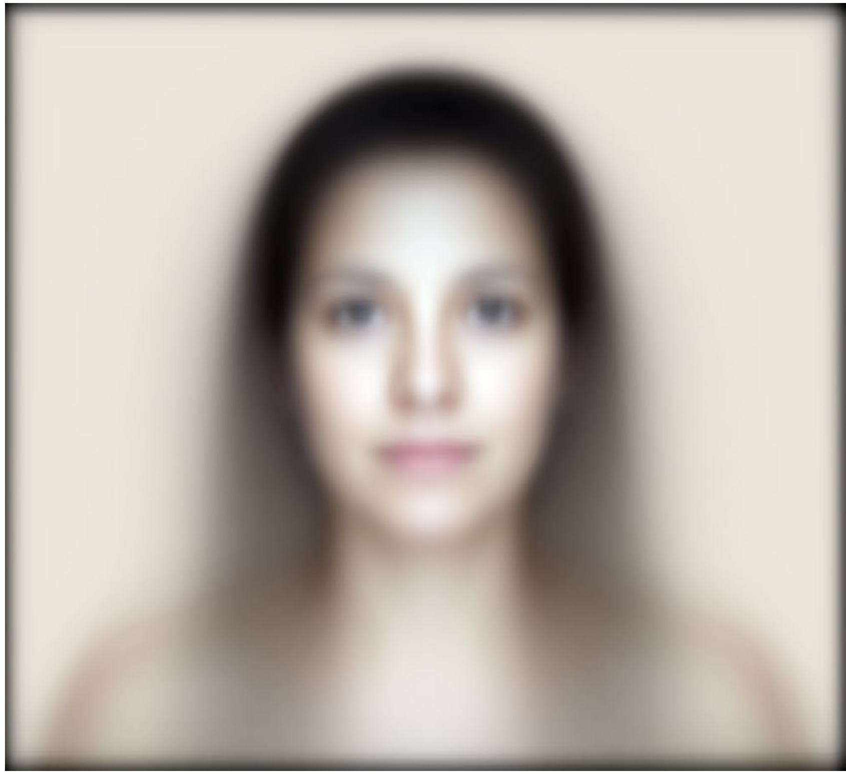
RIGID Registration

- ROTATION
  - an angular shift around each axis
- 2D images (3 df)
  - Rotation (XY): rotation within the 2D plane
- 3D volumes (6 df)
  - Rotation (XY, YZ, XZ): Roll, Pitch, Yaw



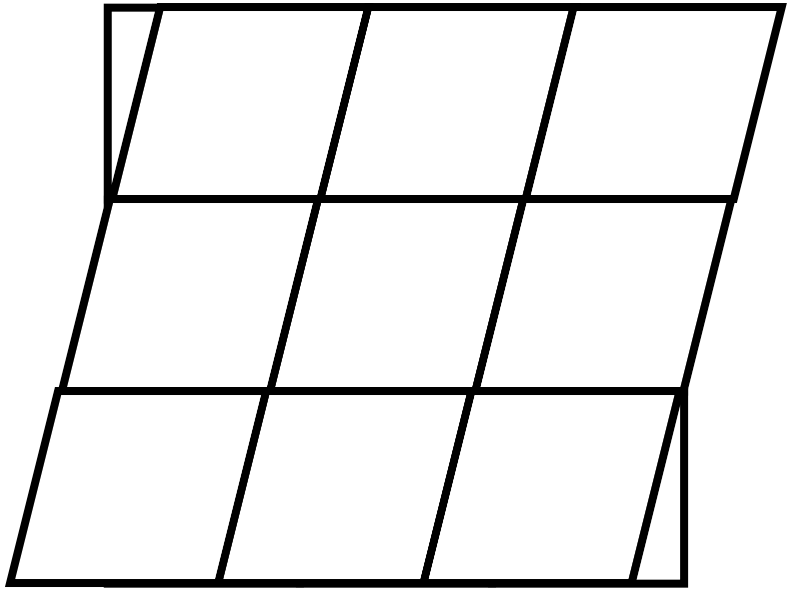
Drones are RIGID Registration Machines

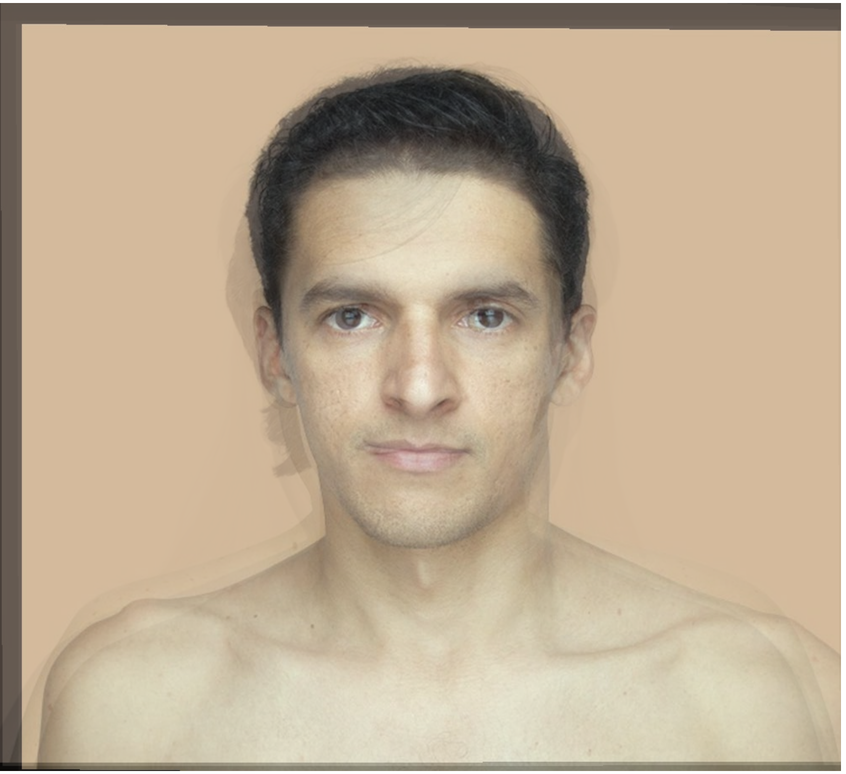


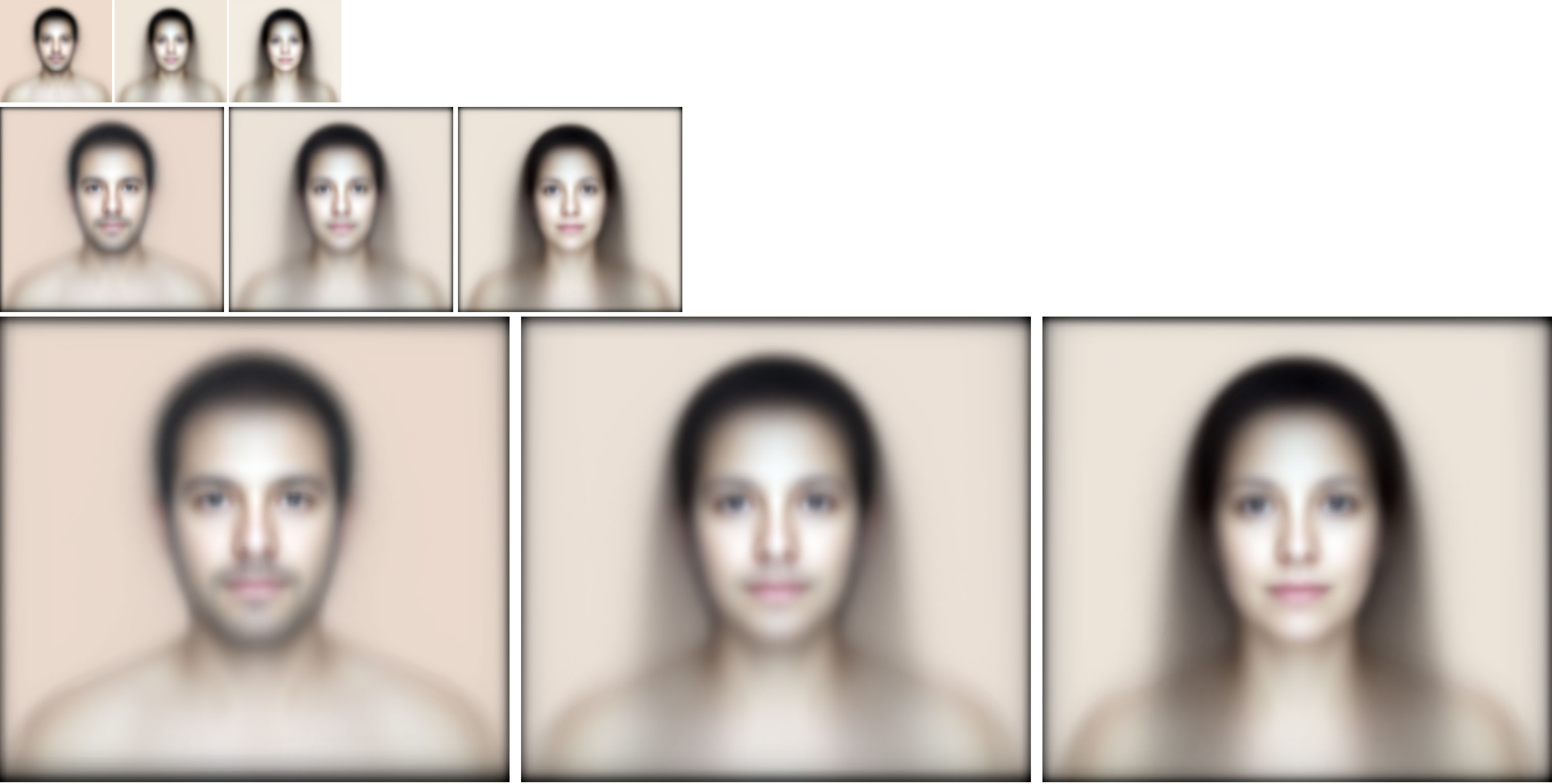







AFFINE Registration

- includes rigid components of translation and rotation, plus changes that alter the shape of the image
- ALL PIXELS AND VOXELS MOVE THE SAME
- SHEARING
  - points are displaced in a given direction proportional to their distance from a line parallel to that direction that runs through the origin
  - area/volume is *unchanged*
- SCALING
  - points are displaced in all directions proportional to the distance of that point from the origin.
  - things become bigger or smaller
  - area/volume is *changed*

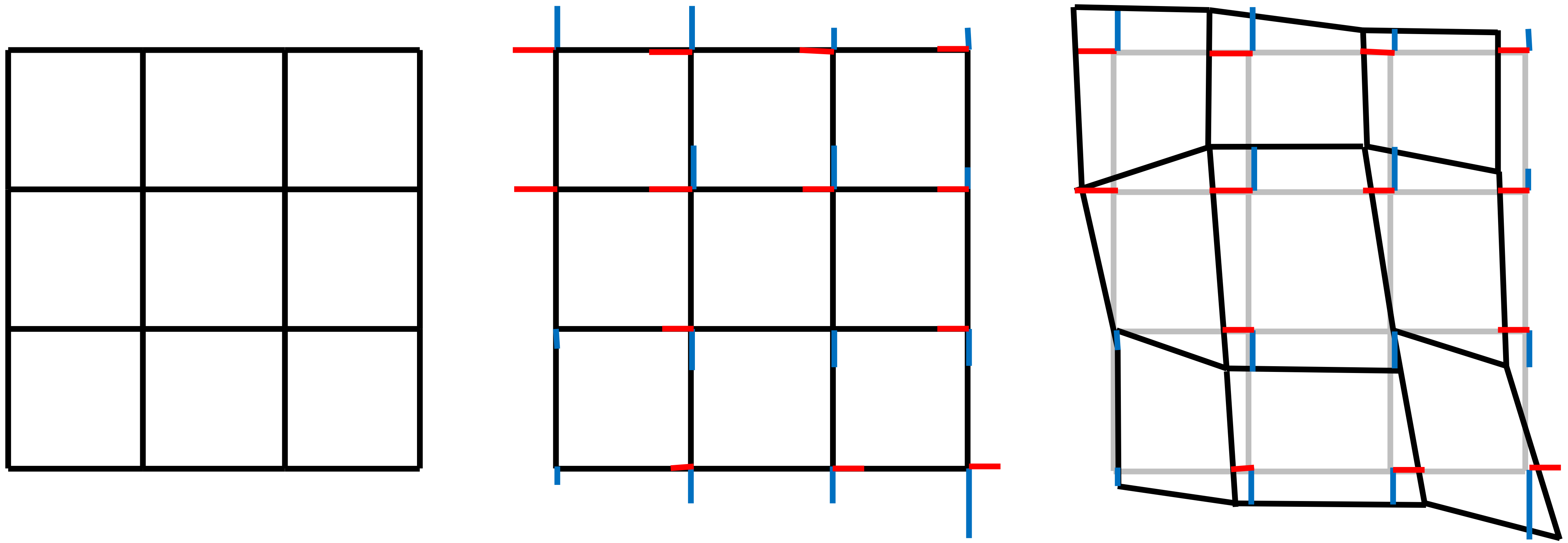






DEFORMABLE Registration

- each point moves independently, each point gets a displacement vector toward the nearest best-matching feature
- shape distortion is the goal, to match moving shape to the fixed shape.
- manifold:
  - a representation of space such that neighbors in the representation are neighbors in space
- diffeomorphism:
  - a mapping between manifolds (or different representations of space)
  - has some specific mathematical properties, beyond our scope (differentiable and invertible)



The Command Line and the Bourne Again Shell

aka a digression into more Linux and BASH, hooray.

- A *Command Line* is an interface that is used to enter text commands, an important command line is the *Terminal*
  - a *Console* is a physical terminal directly communicating with the OS… but we're not prehistoric savages.
- The terminal runs a *shell*
- A *shell* is a program that processes commands and outputs results
- *Commands* is a sequence of characters that provides instructions for the shell to do something
- *Commands* often have parameters you can set, called *arguments*
- *Commands* with multiple *arguments* use *flags* to indicate which parameter is being set

```
sh>> miscCommand
sh>> miscCommand –i “input”
sh>> miscCommand ––input “input”
sh>> miscCommand –i “input” –o “output”
sh>> miscCommand –I “input” \
-o “output”
```

Anatomy of an ANTs Registration

- Advanced Normalization Tools (ANTs) are the ***de facto*** standard tools for coregistration in neuroimaging today
  - other tools have incorporated ANTs into themselves or the methods
- Linux-based command line tools for coregistration
- ANTs implements many transform types, and allows implementing these in sequence for high-quality registrations

```
# INPUTS:
FIXED_T1=${DIR_TEMPLATE}/template_T1w.nii.gz
FIXED_T2=${DIR_TEMPLATE}/template_T2w.nii.gz
FIXED_MASK${DIR_TEMPLATE}/template_mask-brain.nii.gz

MOVING_T1=${DIR_PROJECT_ANAT}/native/sub-123_ses-20211108_T1w.nii.gz
MOVING_T2=${DIR_PROJECT_ANAT}/native/sub-123_ses-20211108_T2w.nii.gz
MOVING_MASK=${DIR_PROJECT_ANAT}/mask/sub-123_ses-20211108_mask-brain.nii.gz
```

- The critical inputs for ANTs registration are images in a moving set that correspond to images in the target fixed set.

- masks are optional, but can help focus the registration to a region like the brain for better registration within that region.

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

antsRegistration

- the main function call for an ANTs registration
- for help and additional input flags
  - > antsRegistration --help

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--float

- use 32bit floating point numbers instead of 64bit double precision

--verbose (0)/1

- logical toggle verbose output while registering, default off

--random-seed

- a numeric value to start random processes, for reproducibility

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform BsplineSyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox
```

--dimensionality 3

- the spatial dimensions of the item being registered, typically 2 or 3 for 2D aimages and 3D volumes

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--collapse-output-transforms 1

- logical specifying whether or not to combine sequential transforms into a single file
  - i.e., combine rigid and affine into a single affine representation or combine all deformations into a single warp and inverse warp file

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

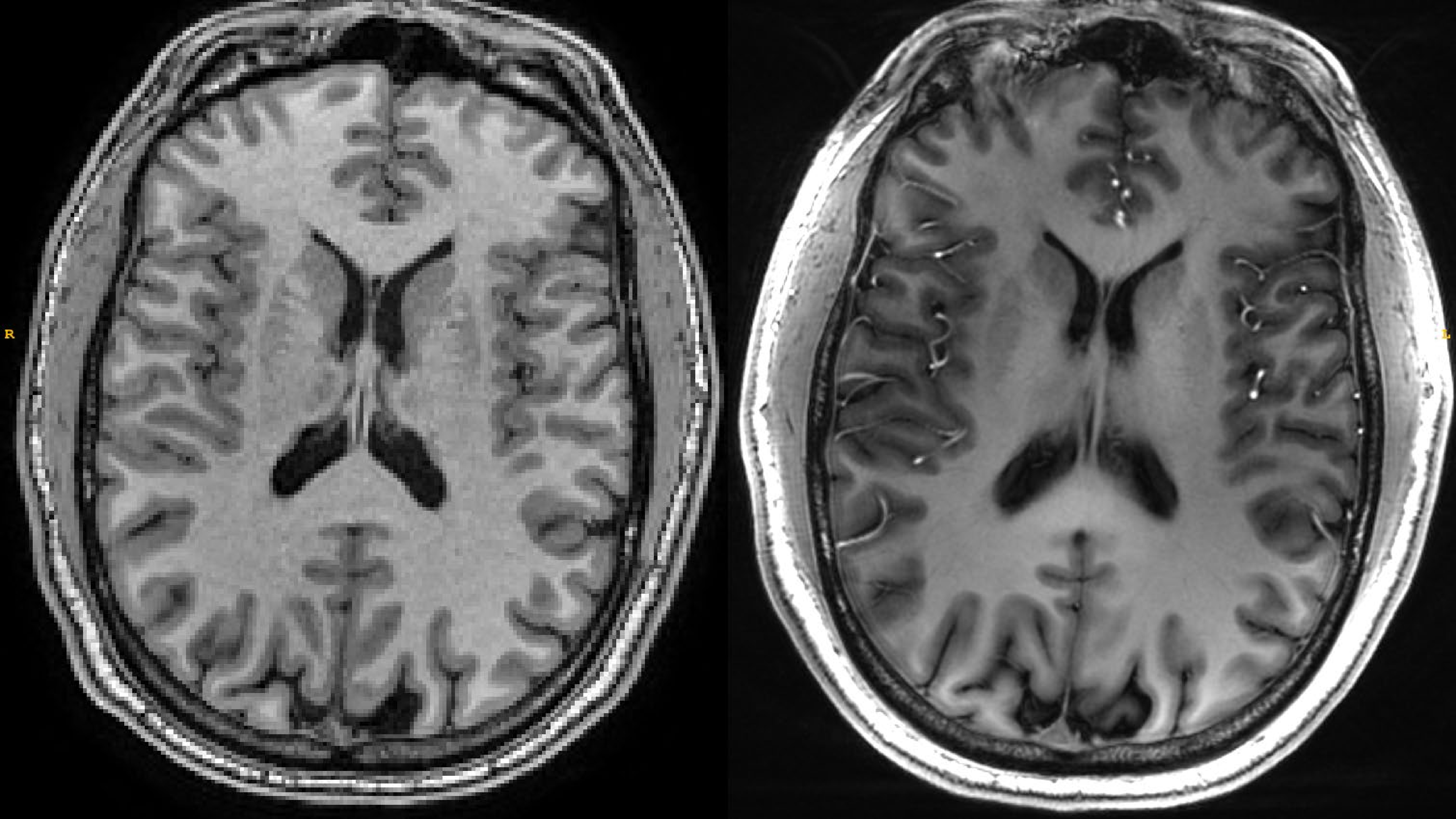
--initialize-transforms-per-stage 0

- logical indicating whether to use the previous stage as a starting point for the following stage
- default is to do this (1)
- results tend to be better with masked regions when each stage is unbiased

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--use-histogram-matching

- similar intensity values in fixed and moving images can facilitate better registrations
- only works within-modalities, e.g., T1w to T1w
- registering between modalities, e.g., T2w to T1w, matching intensities would result in poorer registration
  - due to non-matching tissue contrasts


R
L

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--use-estimate-learning-rate-once 0

- may help with secondary rounds of registration, generally leave it off.

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--winsorize-image-intensities [0.005,0.995]

- "winsorize" intensities in the image at the specified quantiles
- winsorizing "clamps" the values at the specified quantiles (0.5% and 99.5%)
  - <0.5% = 0.5%
  - >99.5% = 99.5%
- helps prevent intensity spikes in the image distorting the range of the calculations used to match images

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--initial-moving-transform [${FIXED},${MOVING},1]

- perform a very simple initial alignment of the images to kick start registration, the better the starting point the better the registration.
  - option 0: use the geometric center of the images
  - option 1: use the image intensities
  - option 2: use the image origin points
- can use an existing transform instead (or an inverse transform)
- multiple transforms can be used by repeating this input

--initial-fixed-transform [${FIXED},${MOVING},1]

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,2,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--transform WHICHXFM[gradientStep]

- which transformation to perform, specified in sequence
- available options:
  - Translation[gradientStep]
  - Rigid [gradientStep]
  - Affine [gradientStep]
  - SyN[gradientStep,FieldVariance,TotalVariance]
  - BsplineSyN[gradientStep,FieldMeshSize,TotalMeshSize,splineOrder]
- gradientStep is how much each point can move during each step
- for deformable registrations (SyN and BSplineSyN), additional parameters limit how much each voxel can move independently of its neighbors
  - the first value limits changes per iteration
  - the second value limits overall changes across iterations
  - effect how fluid or elastic deformable registrations are

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--metric METRIC[${FIXED},${MOVING},PARAMETERS]

- the metric to use to compare images at each step of the registration
- many options are available, most commonly used are Mattes/MI and CC
- Mattes/MI: mutual information
  - preferred across modalities
  - based on location of intensity gradient similarity*
  - MI[${FIXED},${MOVING},weight,bins,sampling,sample]
    - weight – how much to weight this metric in this stage
    - bins – number of bins for calculating histogram for metric, higher = finer detail
    - sampling procedure: None (use all voxels), Regular (sample voxels regularly throughout the image), Random (Random sample of voxels)
- CC: cross-correlation
  - preferred for deformable registration
  - based on similarity in local intensity*
  - CC[${FIXED},${MOVING},weight,radius,sampling,sample]
    - radius – distance (in voxels) over which to calculate local cross-correlation
- Multiple modalities / moving images get their own metric input
- all fixed images must be coregistered first

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--masks [${FIXED_MASK},${MOVING_MASK}]

- a binary mask, where 1s represent the region to be registered and 0s to be ignored
- can be specified once to apply to all levels, or for each level independently
- masks at each level could be different ROIs
- to have no mask at a certain level use NULL
- allows focusing the sampling and computation on a smaller region which improves local registration within the mask
- be careful with edge features
  - edges provide powerful information for registration
  - removing them may hinder registration
  - dilating masks (making them slightly bigger) to include desired edges can improve performance

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
  --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
     --convergence [2000x2000x2000x2000x2000,1e-6,10] \
     --smoothing-sigmas 4x3x2x1x0vox \
     --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
     --convergence [2000x2000x2000x2000x2000,1e-6,10] \
     --smoothing-sigmas 4x3x2x1x0vox \
     --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
     --convergence [2000x2000x2000x2000x2000,1e-6,10] \
     --smoothing-sigmas 4x3x2x1x0vox \
     --shrink-factors 8x8x4x2x1 \
  --transform BsplineSyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
     --convergence [100x70x50x20,1e-6,10] \
     --smoothing-sigmas 3x2x1x0vox \
     --shrink-factors 8x4x2x1
```

--convergence [iterations,threshold,window]

--smoothing-sigmas #x#x#vox

--shrink-factors #x#x#

- each transformation is done in a series of multi-resolution steps
- smoothing-sigmas at each step indicate how much to smooth the image for each step
  - avoids blockiness in the interpolation which would lead to false edges
  - values are SD of Gaussian kernel
- shrink-factors will shrink the image by the specified factor in each step
  - done for speed, get rough low resolution registration, then iteratively improve at higher resolutions
  - applied after smoothing
- convergence sets the parameters for each step, including:
  - the maximum number of iterations per step
  - threshold and window indicate to stop the stage if the metric has not improved by the threshold amount in the last number of iterations

```
antsRegistration \
  --float 1 --verbose 0 --random-seed 32300298 \
  --dimensionality 3 \
   --output ${DIR_SAVE}/xfm_ \
  --collapse-output-transforms 1 \
  --initialize-transforms-per-stage 0 \
  --use-histogram-matching 1 \
  --use-estimate-learning-rate-once 0 \
  --winsorize-image-intensities [0.005,0.995] \
  --initial-moving-transform [${FIXED_T1},${MOVING_T1},1] \
  --transform Rigid[0.2] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.5] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,32,Regular,0.25] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,32,Regular,0.25] \
    --masks[NULL,NULL] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform Affine[0.1] \
    --metric Mattes[${FIXED_T1},${MOVING_T1},1,64,Regular,0.30] \
    --metric Mattes[${FIXED_T2},${MOVING_T2},1,64,Regular,0.30] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [2000x2000x2000x2000x2000,1e-6,10] \
    --smoothing-sigmas 4x3x2x1x0vox \
    --shrink-factors 8x8x4x2x1 \
  --transform SyN[0.1,3,0] \
    --metric CC[${FIXED_T1},${MOVING_T1},1,4] \
    --metric CC[${FIXED_T2},${MOVING_T2},1,4] \
    --masks[${FIXED_MASK},${MOVING_MASK}] \
    --convergence [100x70x50x20,1e-6,10] \
    --smoothing-sigmas 3x2x1x0vox \
    --shrink-factors 8x4x2x1
```

--output [xfm,transformedImage,inverseWarped]

- names for output
- first output is the transforms:
  - set the prefix to be appended to each transform
  - ${DIR_SAVE}/xfm_0GenericAffine.mat
  - ${DIR_SAVE}/xfm_1Warp.nii.gz
  - ${DIR_SAVE}/xfm_1InverseWarp.nii.gz
- second output is the transformed image, MOVING in FIXED space
  - generally, a separate call to antsApplyTransforms gives better control
- third output is the FIXED image in MOVING space

```
antsApplyTransforms \
  --dimensionality 3 \
  --input-image-type 0 \
  --input ${MOVING_T1} \
  --output ${DIR_SAVE}/MOVING_TO_FIXED.nii.gz \
  --interpolation Bspline[3] \
  --transform ${DIR_SAVE}/xfm_1Warp.nii.gz \
  --transform ${DIR_SAVE}/xfm_0GenericAffine.mat \
  --reference-image ${FIXED_T1}
```

antsApplyTransforms

- a function to apply transforms generated by antsRegistration to images

```
antsApplyTransforms \
  --dimensionality 3 \
  --input-image-type 0 \
  --input ${MOVING_T1} \
  --output ${DIR_SAVE}/MOVING_TO_FIXED.nii.gz \
  --interpolation Bspline[3] \
  --transform ${DIR_SAVE}/xfm_1Warp.nii.gz \
  --transform ${DIR_SAVE}/xfm_0GenericAffine.mat \
  --reference-image ${FIXED_T1}
```

--dimensionality 3

- spatial dimensions in image
- pictures are 2D
- neuroimages are generally 3D volumes
- time-series, are 3D spatial volumes in this sense --- NOT 4D-space

--input-image-type 0

- what type of image to apply the transform to
- 0 is default, for 3D volumes and pictures
- time-series, you need to apply the 3D transform over a series of 3D volumes, set this input to 3

```
antsApplyTransforms \
  --dimensionality 3 \
  --input-image-type 0 \
  --input ${MOVING_T1} \
  --output ${DIR_SAVE}/MOVING_TO_FIXED.nii.gz \
  --interpolation Bspline[3] \
  --transform ${DIR_SAVE}/xfm_1Warp.nii.gz \
  --transform ${DIR_SAVE}/xfm_0GenericAffine.mat \
  --reference-image ${FIXED_T1}
```

--input ${MOVING}

- image to be transformed

--output ${DIR_SAVE}/MOVING_TO_FIXED.nii.gz

- output warped image
- OR output merged transform file
  - --output [${DIR_SAVE}/combinedDeformation.nii.gz,1]
- OR combine affine matrices
  - --output Linear[${DIR_SAVE}/combinedDeformation.nii.gz,0]
  - logical indicates to calculate inverse

--reference-image ${FIXED}

- image to be used to define the spacing, origin, size, and direction of the output

```
antsApplyTransforms \
  --dimensionality 3 \
  --input-image-type 0 \
  --input ${MOVING_T1} \
  --output ${DIR_SAVE}/MOVING_TO_FIXED.nii.gz \
  --interpolation Bspline[3] \
  --transform ${DIR_SAVE}/xfm_1Warp.nii.gz \
  --transform ${DIR_SAVE}/xfm_0GenericAffine.mat \
  --reference-image ${FIXED_T1}
```

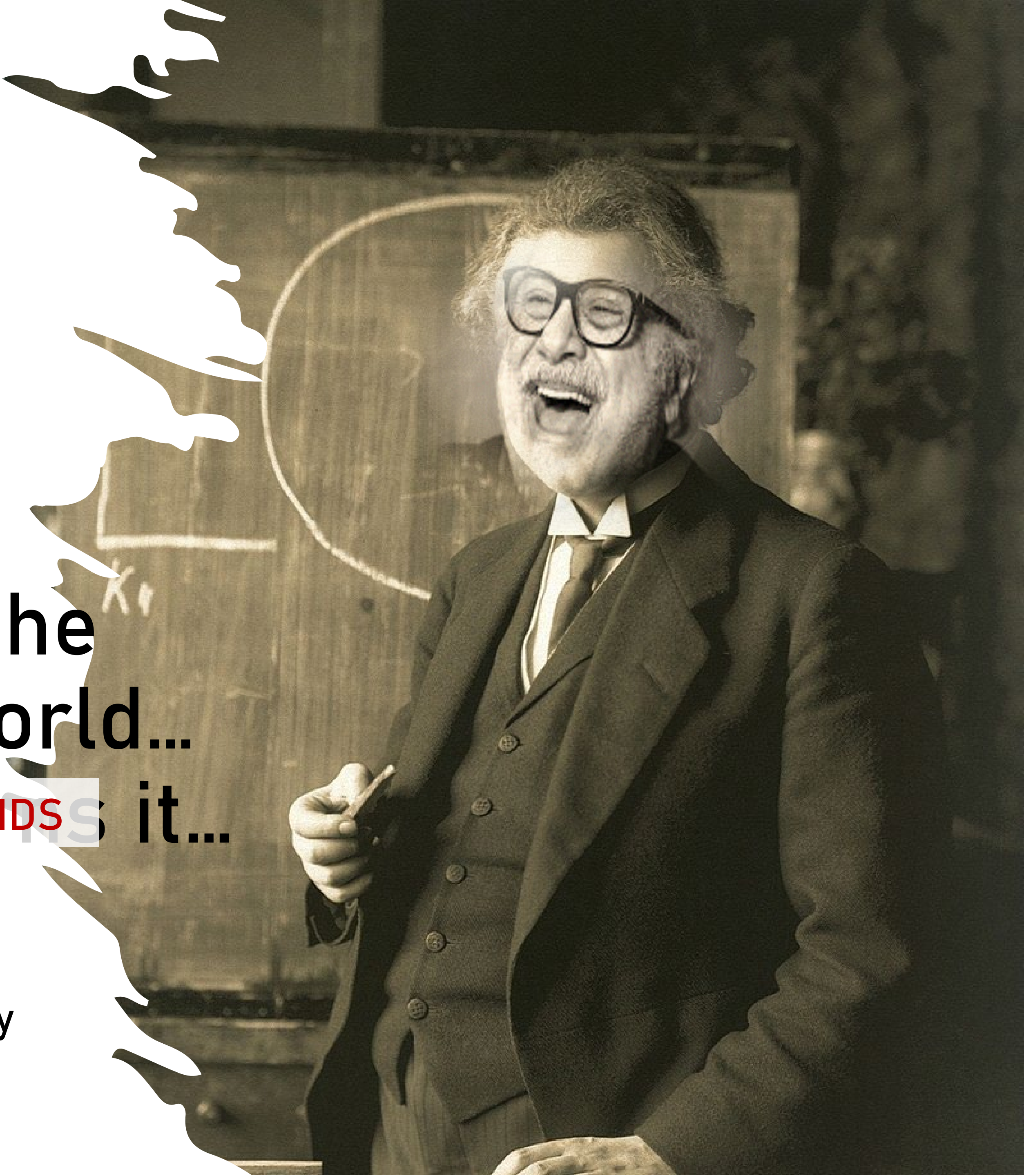
--interpolation TYPE

- the type of interpolation to use when applying transforms
- Good for images with continuous distributions of intensity:
  - Linear
  - Bspline[3] (cubic interpolation)
- For images with discrete values like masks and label sets
  - GenericLabel (binary)
  - MultiLabel (categorical)
  - NearestNeighbor (categorical, blocky)

Repeated Resampling and Interpolation Error

"Compound ~~interest~~ INTERPOLATION is the eighth ~~wonder~~ BLUNDER of the world… who understands it, ~~earns~~ AVOIDS it… who doesn't, ~~pays~~ MAKES it"

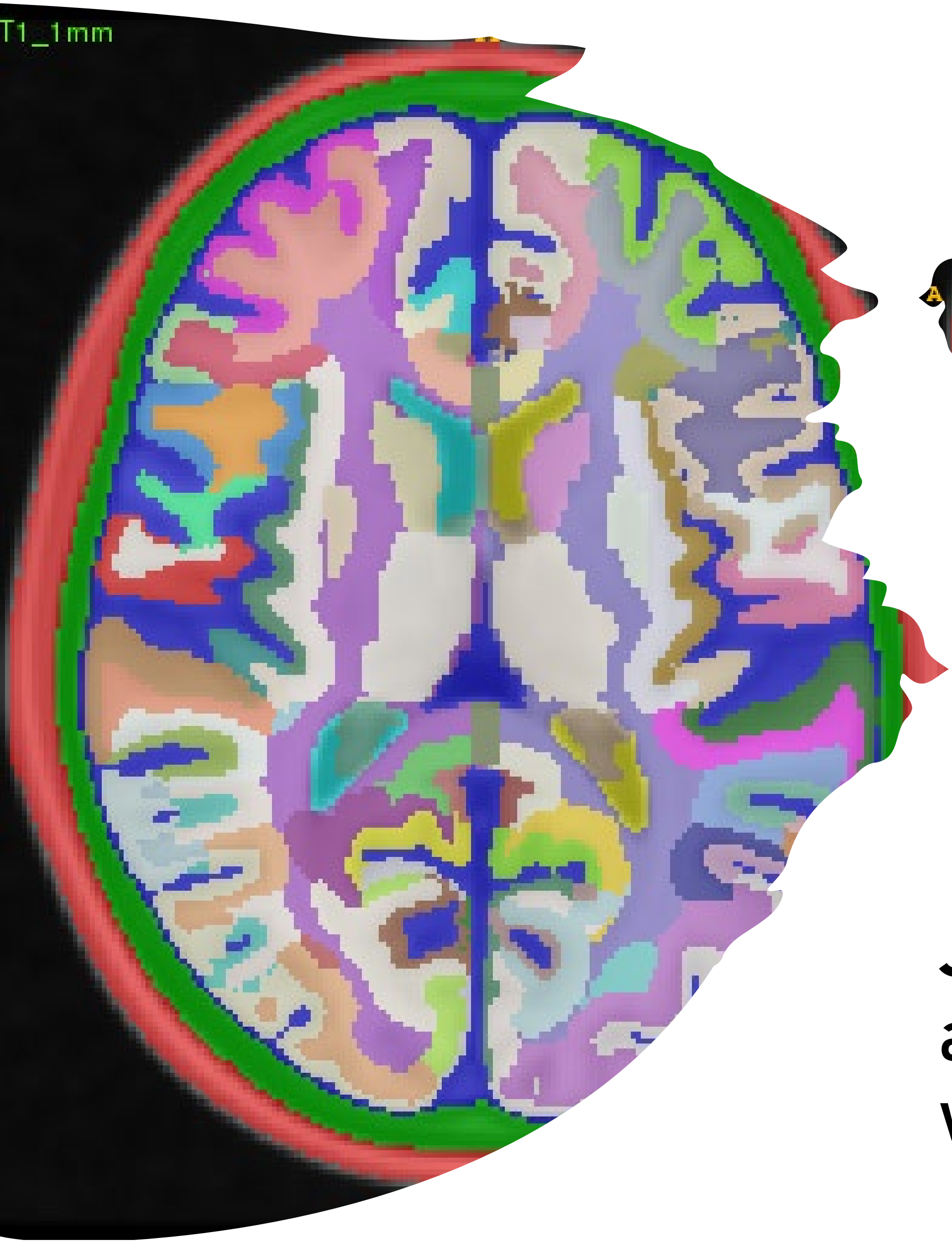
-Einstein's Statistician Probably

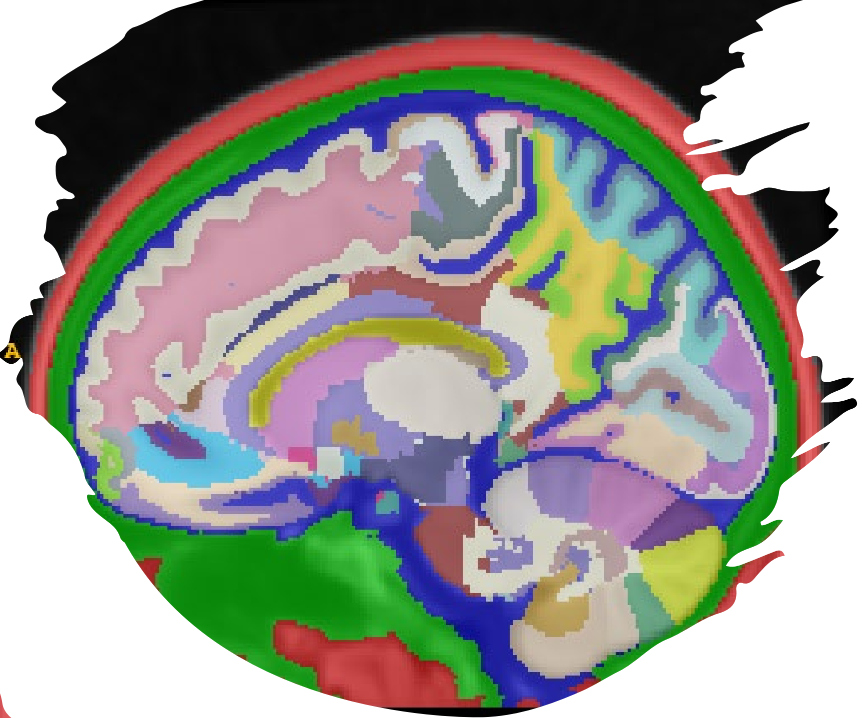


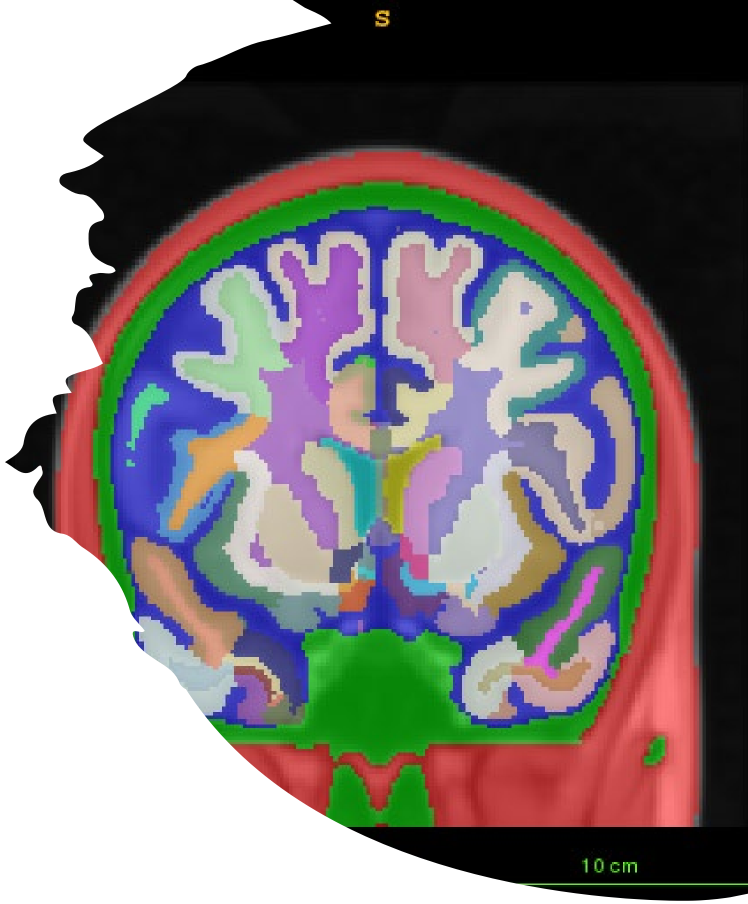
```
antsApplyTransforms \
  --dimensionality 3 \
  --input-image-type 0 \
  --input ${MOVING_T1} \
  --output ${DIR_SAVE}/MOVING_TO_FIXED.nii.gz \
  --interpolation Bspline[3] \
  --transform ${DIR_SAVE}/xfm_1Warp.nii.gz \
  --transform ${DIR_SAVE}/xfm_0GenericAffine.mat \
  --reference-image ${FIXED_T1}
```

--transform <XFM_FILE>

- transforms to apply to the image
- applied in ascending order, i.e., bottom up, last in order is first to be applied
- affine transforms can be applied as inverse
  - --transform [${DIR_SAVE}/xfm_0GenericAffine.mat,1]
- an identity transform is always applied last
  - NIfTI format has transforms specified in the header, these are always applied to the data first





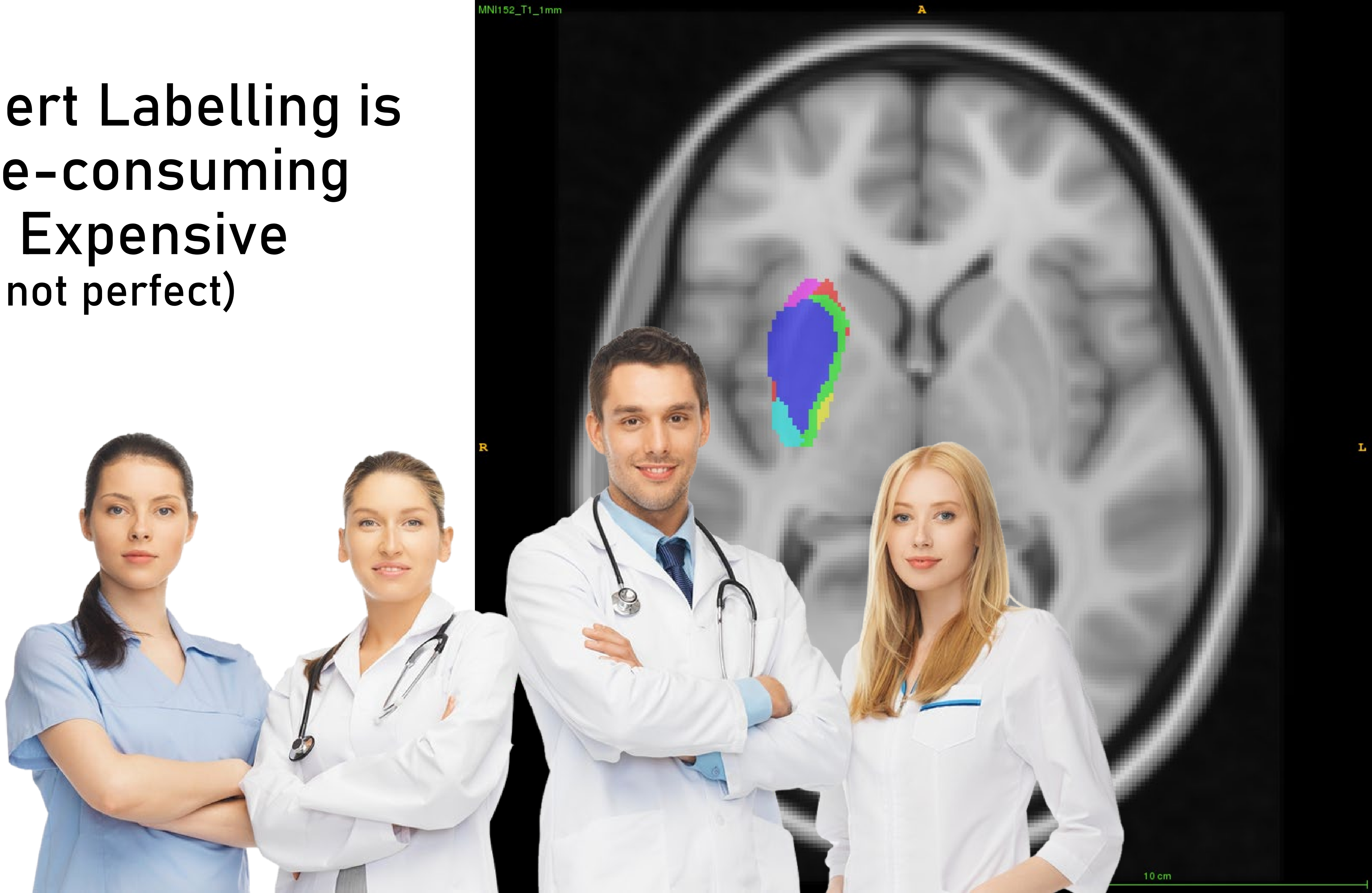


Joint Label Fusion

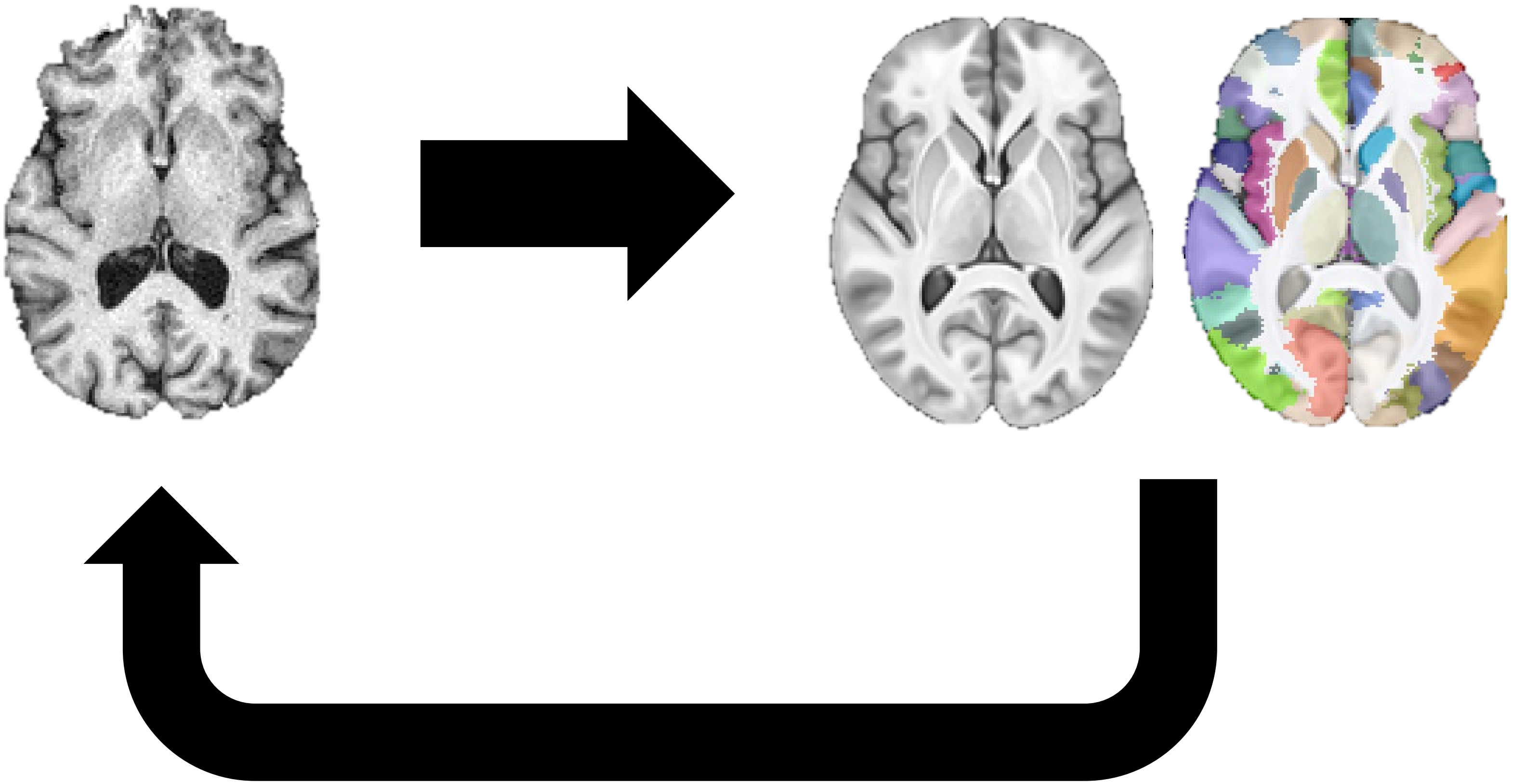
aka I want my brains labeled and I don't want to try or pay for it.

Expert Labelling is Time-consuming and Expensive

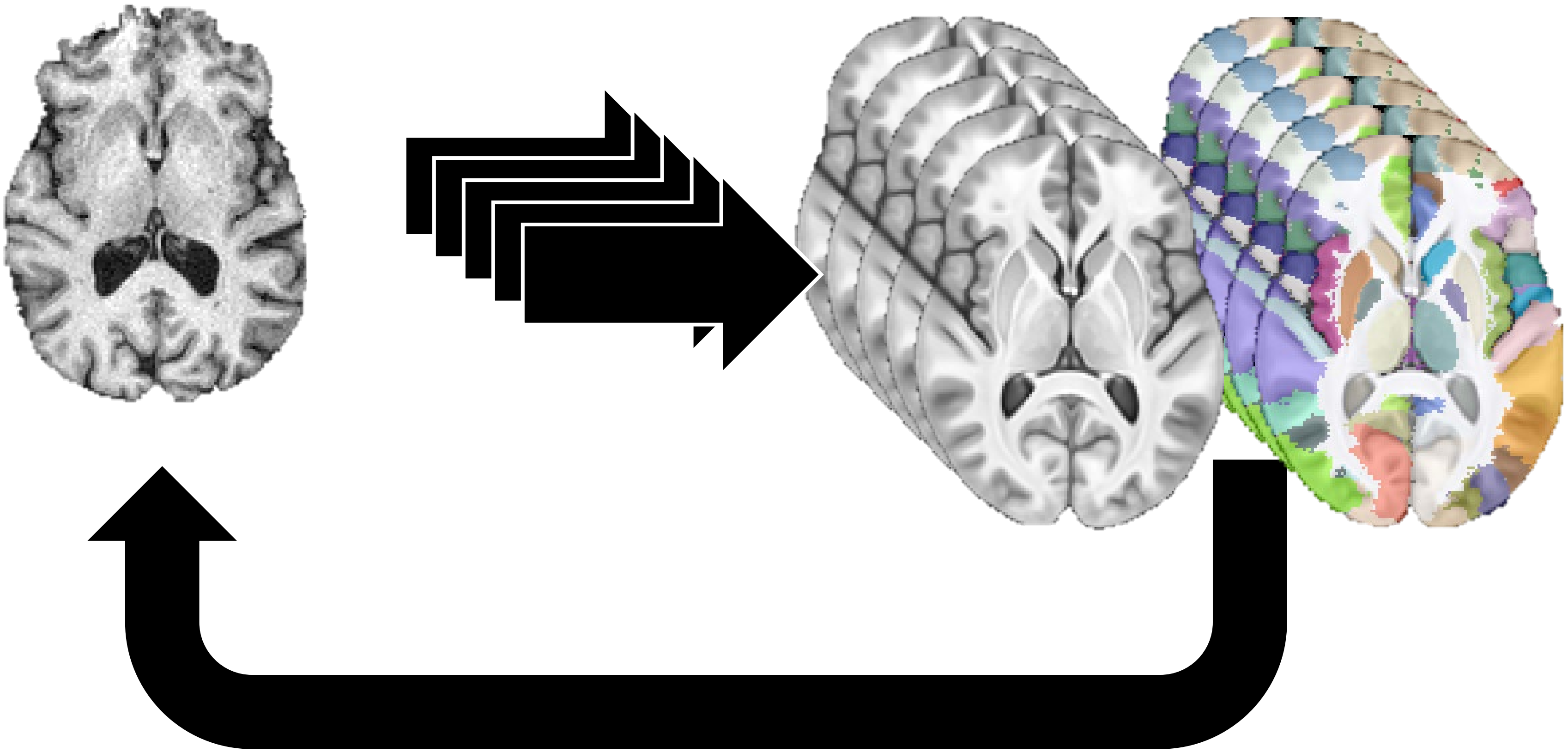
(and not perfect)



Coregistration can help!



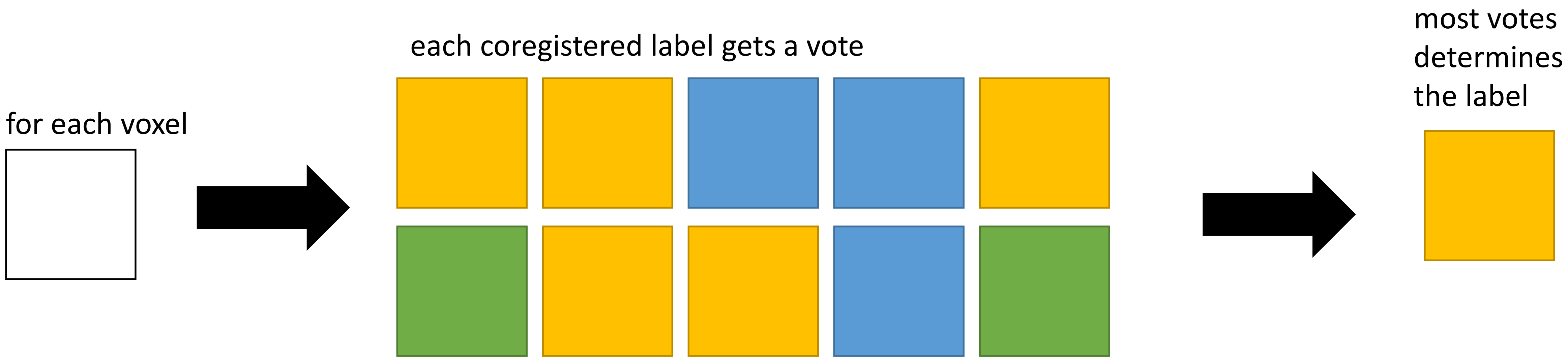
Many coregistrations can help more!



individual differences in neuroanatomy cause
“random” error in coregistration,
multiple registrations can cancel this out

Multi-Atlas Label Fusion

(a slightly simpler variant)

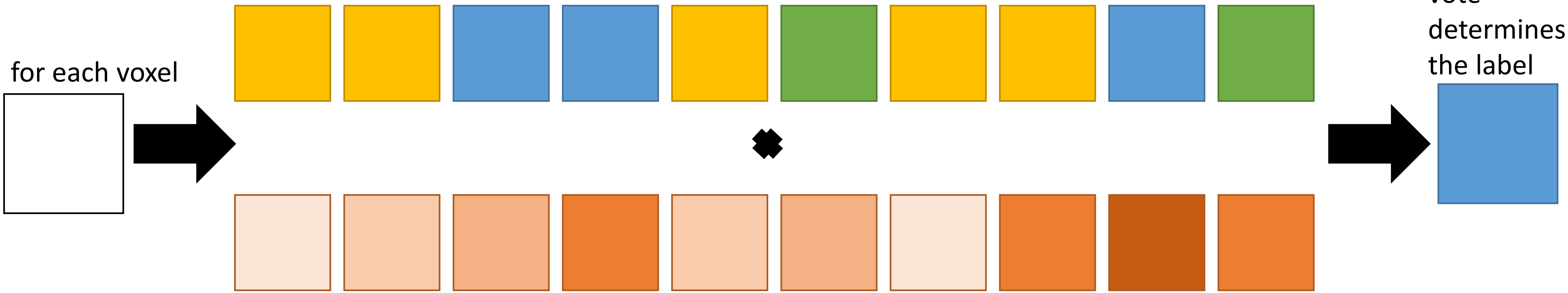


Joint Label Fusion

(registration-based error correction)

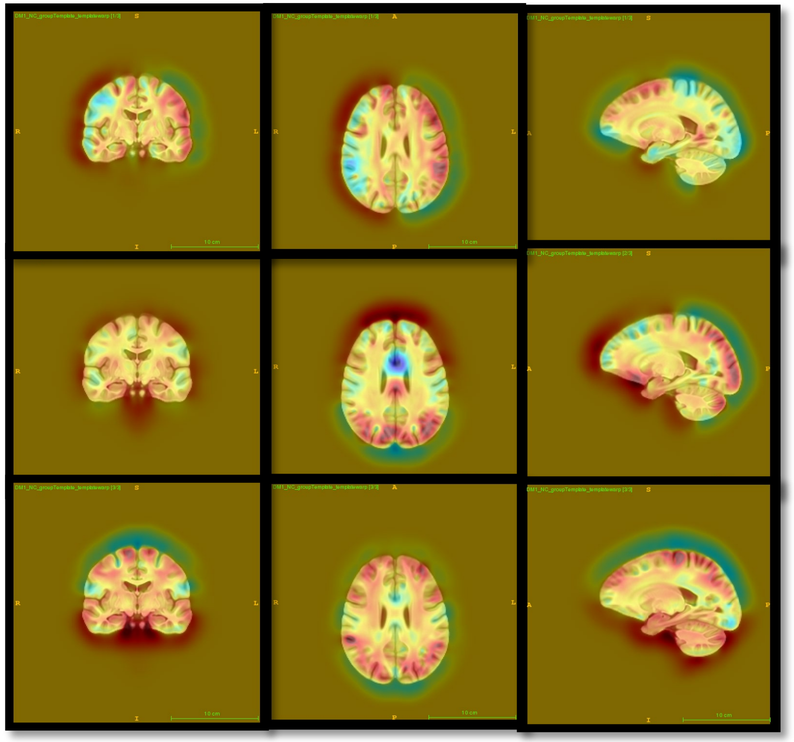
for each voxel

strongest vote determines the label

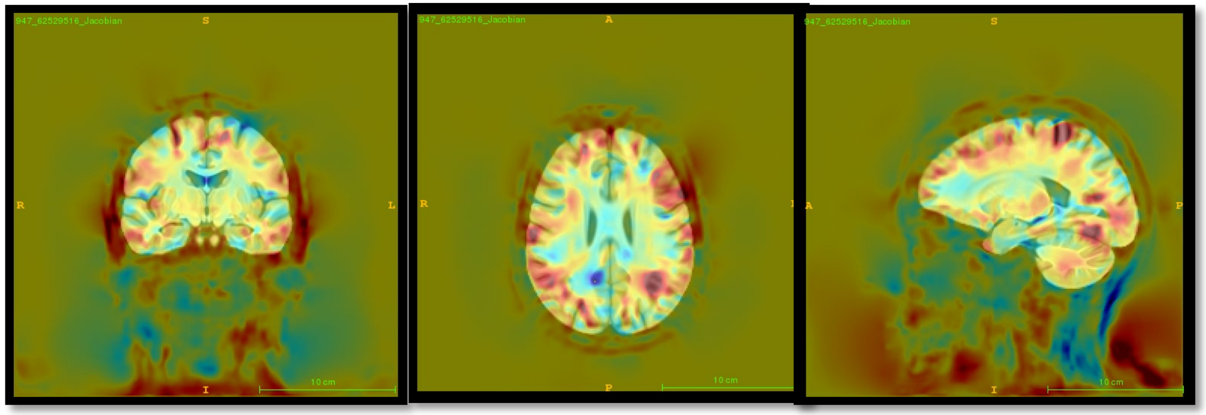


local registration quality (based on fixed and moving image similarity) weights the votes

Tensor-based Morphometry



- deformations represent the amount each voxel needs to move in a given direction to be coregistered
- the derivative of this mapping at each location represents the change at that location as a result of the deformation
  - i.e., magnitude of volumetric change at each voxel
- Jacobian determinants give the ratio of the area of the deformed voxel to the original voxel
- modelling Jacobian Determinants (or Jacobians) is tensor-based morphometry, and allows us to model voxelwise volumetric change



Summary

- Coregistration is an iterative process
- Iteratively move from one set of spatial coordinates to another, using a set of transformations
- ANTs is a powerful tool for high quality registrations that can be adapted to fit a variety of coregistration needs and situations