Overview:
ITK Registration Methods

Lydia Ng
Insightful Corp.

Outline

- Introduction
- ITK registration framework
  - Component overview
- Examples
- GUI communication
- Displaying registered images
- Deformable registration techniques

Introduction to Medical Image Registration

What is Image Registration?

- Process of finding the spatial transform that maps points from one image to the homologous points in another image
- Medical image registration has many clinical and research applications
  – Intra- and inter- subject registration
  – Single and multi- modality registration

Intra-subject Registration

- Repeated image scans of the same subject is used to capture the effect of disease development, treatment progress, contrast bolus propagation etc.
- Registration can be used to compensate for differences in patient placement
- Subtraction of registered images used for visualization and quantification

Inter-subject Registration

- Creation of mean images and atlases
  – Useful for objective and quantitative assessment of abnormalities
  – Computed deformation field used to encode pattern of anatomic variability within a population
- Atlas based segmentation
  – Automatic segmentation by mapping to a labeled atlas

Multi-modality Registration

- Correlate information obtained from different imaging modalities
  - MR
    - Soft tissue discrimination; lesion identification
  - CT
    - Bone localization; surgical guidance
  - PET/SPECT
    - Functional information; tumor localization

Image Registration Classification

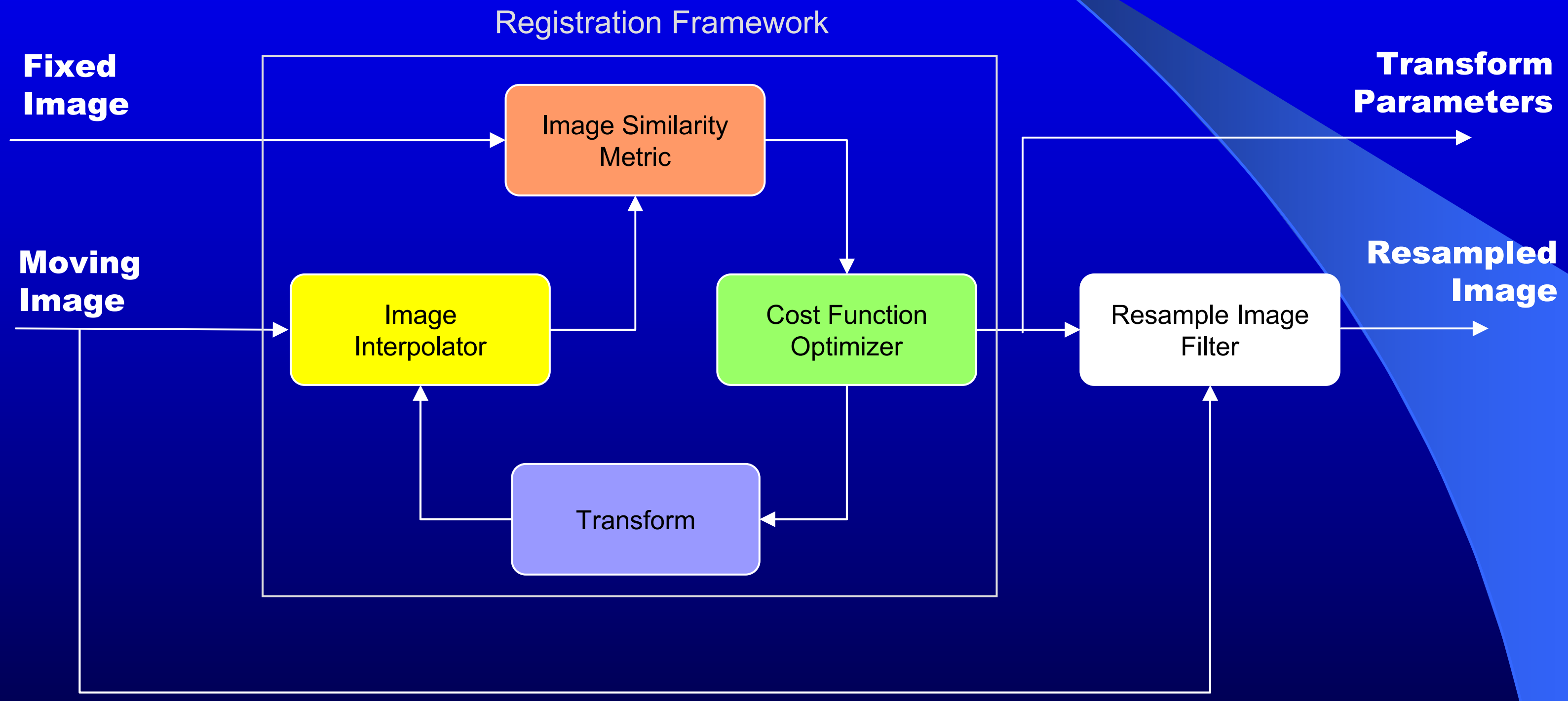
- Registration criteria
  - Quantitative measure of a “good match”
  - Focus on intensity based measures
- Spatial transform type
  - Allowable mapping from one image to another
- Optimization algorithm used
  - Optimize transform parameters w.r.t to measure
- Image interpolation method
  - Value of image at non-grid position

ITK Registration Framework

ITK Registration Framework

- Generic framework for building intensity based registration algorithms
- Each functionality encapsulated as components
- Components are inter-changeable allowing a combinatorial variety of registration methods
- Components are generic
  - Can be used outside the registration framework

Registration Framework Components



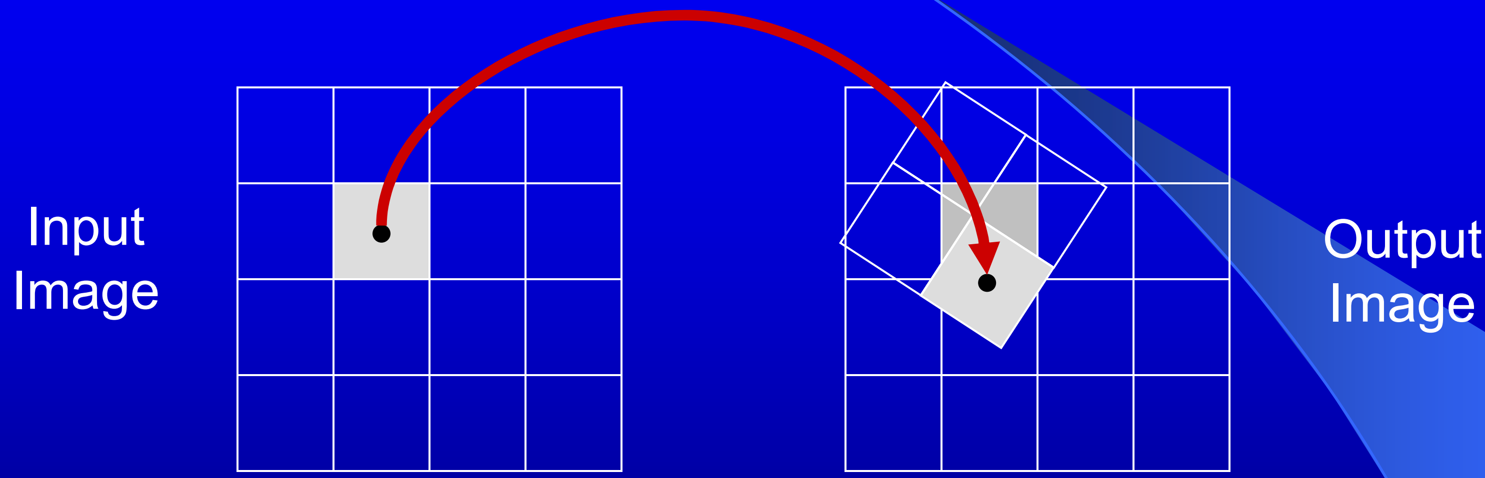
itk::Transfrom

- Encapsulates the mapping of points and vectors from an “input” space to an “output” space
- ITK provides a variety of transforms from simple translation, rotation and scaling to general affine and kernel transforms
- Forward versus inverse mapping
- Parameters and Jacobians
- Examples:
  - Translation, rotation, affine and B-spline deformble

Forward and Inverse Mappings

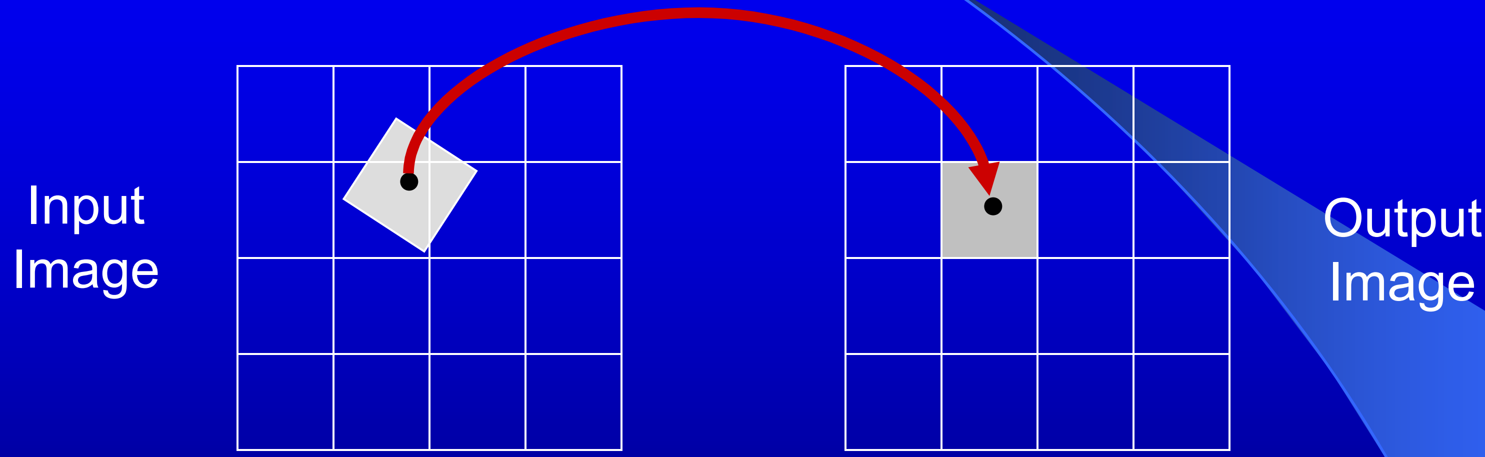
- Relationship between points of two images can be expressed in two ways:
  - Forward
    - Pixel of input image mapped onto the output image
  - Inverse
    - Output pixels are mapped back onto the input image

Forward Mapping



- Input image pixel is mapped onto the output image
- Output pixels with more than one hit: overlap
  - Value must be accumulated from overlapping pixels
- Output pixels with no hits: hole

Inverse Mapping



- Output pixels are mapped back onto the input image
- Output pixel value must be interpolated from a neighborhood in the input image
- Scheme avoids any holes and overlaps in the output image because all pixels are scanned sequentially

Inverse Mapping Contd.

- The registration framework uses **inverse** mapping
- The transform component maps points from the **fixed** image space to the **moving** image space

$$\mathbf{x}' = \mathbf{T}(\mathbf{x}|\mathbf{p})$$

Point in moving image space

Point in fixed image space

Transform Parameters

- Each transform class is defined by a set of parameters
  – Translation, angle of rotation etc
- Represented as a flat array of doubles to facilitate communication with itk::Optimizers
- Transform parameters define the search space for the optimizer
- Goal of registration
  – Find the set of transform parameters that result in the best value of an image similarity metric

Transform Jacobian

- Some metrics require the knowledge of the "transform Jacobian" in order to compute metric derivatives
- The "transform Jacobian" is a matrix whose elements are the partial derivatives of the output point with respect to the transform parameters

$$J_{ij} = \frac{\partial x_i'}{\partial p_j}$$

itk::IdentityTransform

- Maps every point back onto itself
- No parameters

itk::TranslationTransform

- Maps all points by adding a constant vector:

$$\mathbf{x}' = \mathbf{x} + \mathbf{t}$$

- Parameters: $\mathbf{p} = \mathbf{t}$
  - i-th parameter represent the translation in the i-th dimension
- Jacobian in 2D:

$$\mathbf{J} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

itk::Euler2DTranform

- Represents a rotation and translation in 2D

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

- Parameters:

$$\mathbf{p} = \{\theta, t_x, t_y\}$$

- Jacobian:

$$\mathbf{J} = \begin{bmatrix} -(\sin\theta)x - (\cos\theta)y & 1 & 0 \\ +(\cos\theta)x - (\sin\theta)y & 0 & 1 \end{bmatrix}$$

Note on Center of Transformation/Rotation

- Transformation/Rotation is about the origin {0,0}
  - Can represent rotation at arbitrary rotation center

$$\mathbf{x}' - \mathbf{c}_m = \mathbf{M}(\mathbf{x} - \mathbf{c}_f) + \mathbf{t}$$

$$\mathbf{x}' = \mathbf{M}\mathbf{x} + \underbrace{(\mathbf{t} + \mathbf{c}_m - \mathbf{M}\mathbf{c}_f)}_{\mathbf{t}'}$$

  - But at the expense of optimization performance, especially for large auxiliary translation
- Alternatives:
  - “Centered” versions of transform
  - Move the image origins

Note on Parameter Scaling

- Different type of parameters (translation vs. angle vs. scaling) have different dynamic ranges
  - E.g. a unit change in rotation has a much larger impact than a unit change in translation
- Differences in scale appears as long narrow valleys in the search space making optimization difficult
- Rescaling the parameters can help fix this problem

itk::Euler3DTranform

- Represents 3D rotation and translation
  - Rotation about each coordinate axis

$$\mathbf{x}' = \mathbf{R}_Z \mathbf{R}_Y \mathbf{R}_X \mathbf{x} + \mathbf{t}$$

or

$$\mathbf{x}' = \mathbf{R}_Z \mathbf{R}_X \mathbf{R}_Y \mathbf{x} + \mathbf{t}$$

- Parameters: $\mathbf{p} = \{\theta_x, \theta_y, \theta_z, t_x, t_y, t_z\}$

Alternative 3D Rigid Transforms

- itk::QuaternionRigidTransform
  - 3D rotation represented by a quaternion
  - Represented by 4 numbers
  - Similar to axis/angle representation
  - Unit quaternion is equivalent to pure rigid
  - Does not suffer from “Gimbal lock”
- itk::VersorRigidTransform
  - Strictly the rotational part of a quaternion (always rigid)
  - Represented by 3 numbers
- Both quaternion and versor components do not form vector spaces
  - Specialized optimizers required

Note on Image Spacing and Origin

- Medical image volume are typically anisotropic
  - In-plane pixel size smaller than inter-slice spacing
- A transform is rigid only with respect to physical coordinates and not pixel coordinates
  - PhysCoord = PixelCoord x ImageSpacing + ImageOrigin
- The registration is always with respect to physical coordinates
- Make sure spacing and origin information is set correctly in the images!

itk::AffineTransform

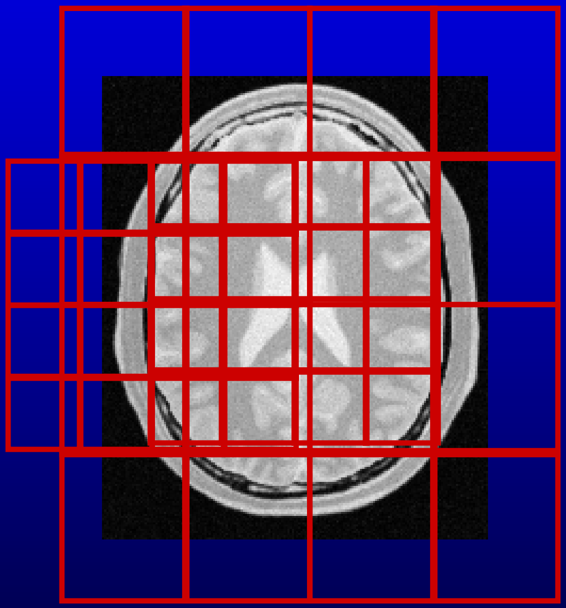
- General affine transform can represent rotation, scaling, shearing and translation
- Parameters: matrix coefficient + translation (6 in 2D, 12 in 3D)
- Parallel lines are preserved

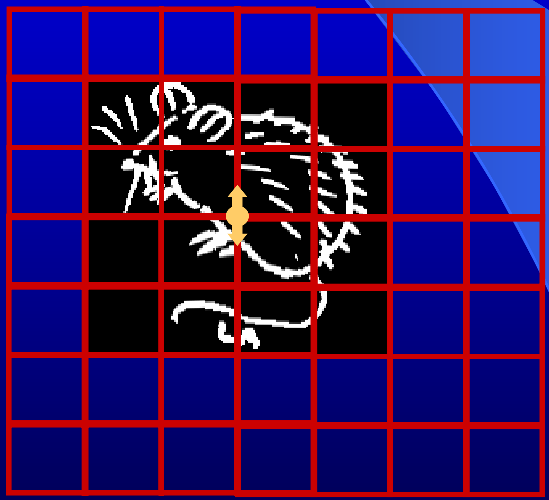
itk::BSplineDeformableTransform

- Represent a deformable warp
- Deformation field represented by B-splines coefficient on a regular grid
- Parameters: B-spline coefficient for each dimension

B-Spline Grid Placement

- Grid defined by origin, spacing, size





SplineOrder = 3

itk::InterpolateImageFunction

- When a point is mapped from one image space to another image space, it will generally be mapped to a non-grid position
- Interpolation is needed to compute the intensity value at the mapped position

Choice of Interpolation Method

- Interpolation affects smoothness of metric space
- Interpolations computed 1000's of times in a single optimization cycle
- Trade-off efficiency with ease of optimization

Interpolation Schemes

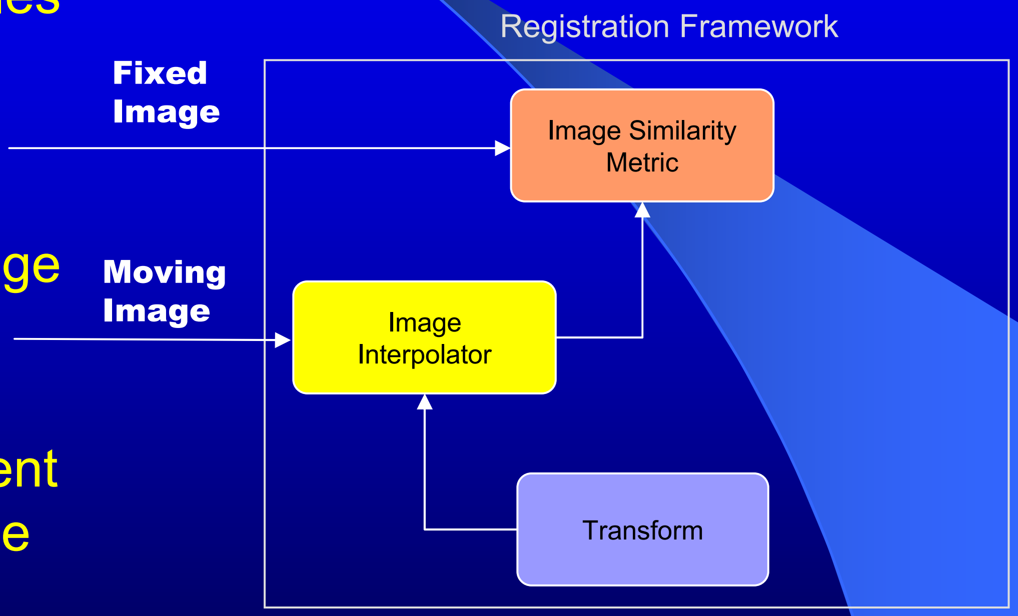
- itk::NearestNeighborInterpolateFunction
  - Assumes image is piecewise constant
- itk::LinearInterpolateFunction
  - Assumes image is piecewise linear
- itk::BSplineInterpolateFunction
  - Underlying image represented using B-spline basis functions
  - On connection, image of B-spline coefficients is computed

itk::ImageToImageMetric

- Measures how well a transformed moving image “matches” the fixed image
- The most critical component
- Scalar function of the transform parameters
  - For given, fixed image, moving image, transformation type and interpolation type

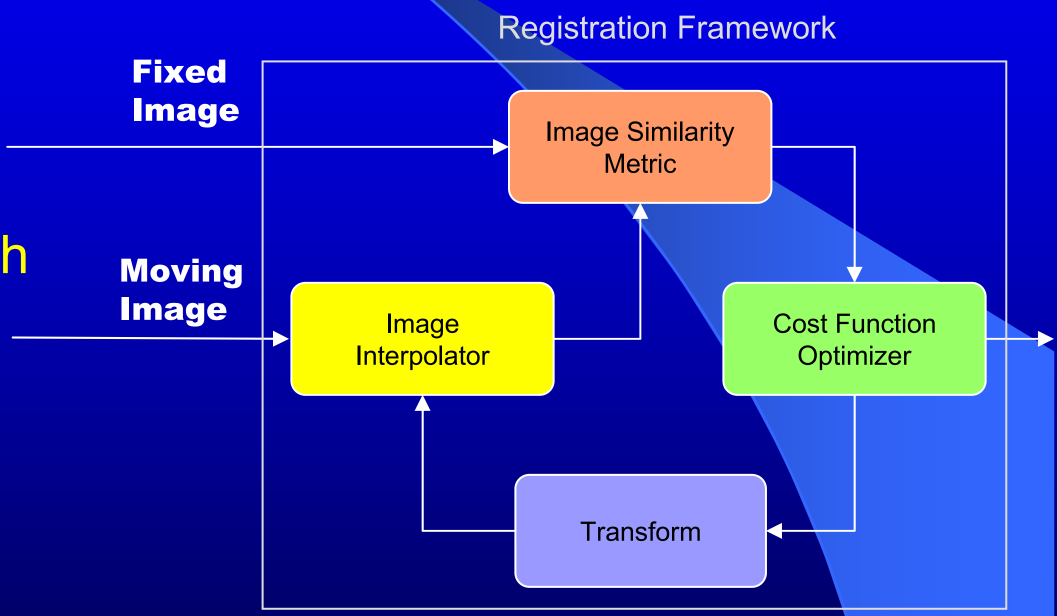
Metrics, Transforms and Interpolators

- The **metric** typically samples points within a defined region of the fixed image
- For each point, the corresponding moving image point is obtained using the **transform** component
- The **interpolator** component is then used to compute the moving image intensity at the mapped position



Metrics and Optimizers

- The **metric** is used by the **optimizer** to evaluate the quantitative criterion at various positions in the transform parameter search space
- For gradient-based optimizer, the metric must also provide the metric derivatives w.r.t each transform parameter
  - Use chain rule with moving image derivatives and transform Jacobian



Choice of Metric

- Highly dependent on the registration problem to be solved
  - Single-modality
  - Multi-modality
  - Large capture range
  - Requires close initialization
- Examples:
  - Mean squares, normalized correlation and mutual information

Mean Squares Metric

Interpolate the moving image

$$S(\mathbf{p}|F,M,\mathbf{T}) = \frac{1}{N}\sum_{i}^{N}\left(F(\mathbf{x}_i) - M(\mathbf{x}_i')\right)^2$$

where

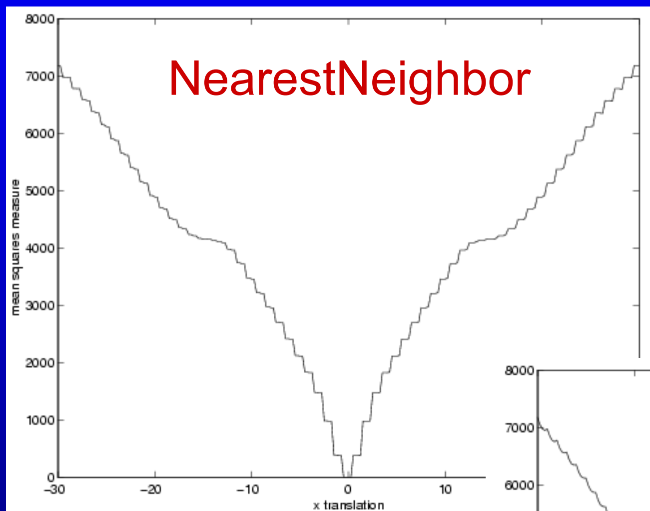
$$\mathbf{x}_i' = \mathbf{T}(\mathbf{x}_i, \mathbf{p})$$

Over a user specified fixed image region

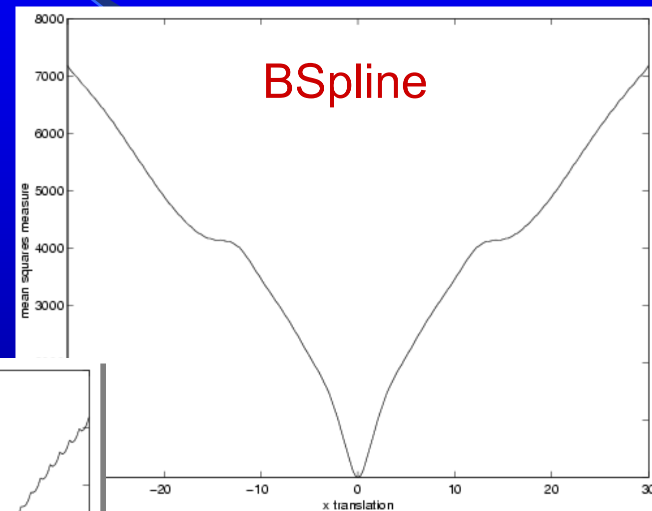
Transform from fixed image point to moving image point

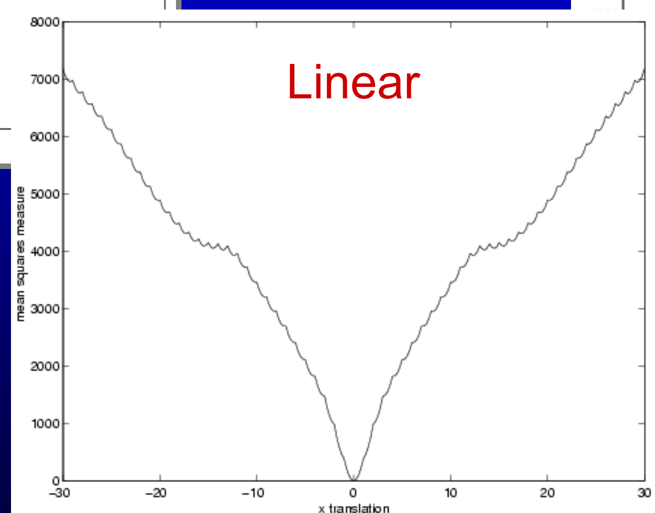
- Simple to compute
- Large capture range
- Restricted to mono-modality applications
- Linear differences in intensity results in poor similarity measure

Mean Squares Metric

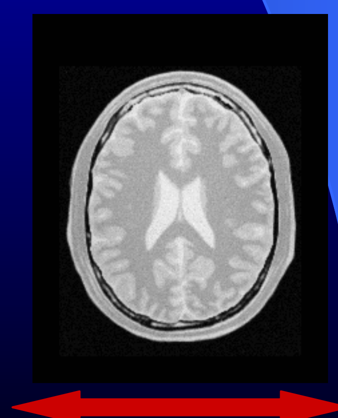


Optimal Value at Zero





Metric range: image dependent



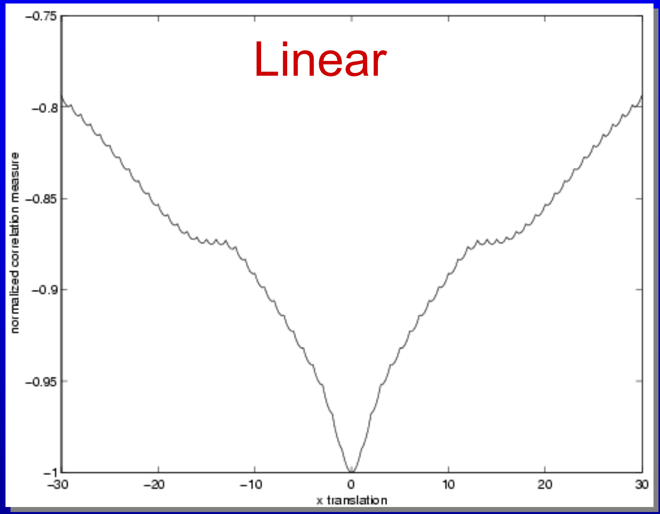
Translations

Normalized Correlation Metric

$$S(\mathbf{p}|F,M,\mathbf{T}) = -1 \times \frac{\sum_i^N F(\mathbf{x}_i)M(\mathbf{x}_i')}{\sqrt{\sum_i^N F^2(\mathbf{x}_i)\sum_i^N M^2(\mathbf{x}_i')}}$$

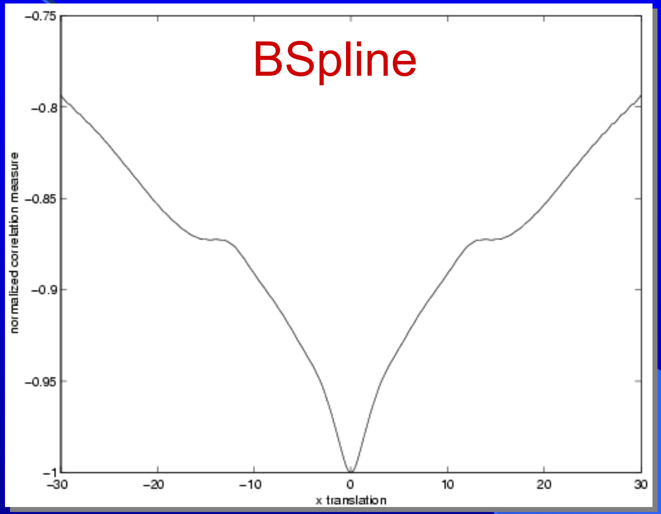
- Pixel-wise cross-correlation between fixed and moving image intensity
- Factor -1 added to work with minimum seeking generic optimizers
- Insensitive to multiplicative difference

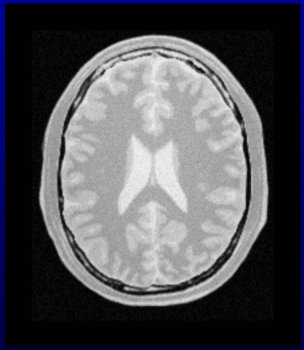
Normalized Correlation Metric



Optimal Value at
-1

Metric range:
1 to -1





Translations

Mutual Information Metric

- Information theoretic entity that qualitatively measures how much information is gained about one RV (intensity in one image) by the knowledge of another RV (intensity in another image)

Mutual Information Metric

- Introduced in the context of multi-modality registration by two different groups: Viola and Wells (1997) and Collignon et al. (1995)
- MI well suited since actual form of dependency between the two RVs does not have to be specified
- MI defined in terms of entropy
  - Measure of information contained in a piece of data

Entropy

- Two RVs: A and B
- Marginal entropies:

$$H(A) = -\int p_A(a) \log p_A(a)\, da$$

$$H(B) = -\int p_B(b) \log p_B(b)\, db$$

- Joint entropy:

$$H(A,B) = -\int p_{AB}(a,b) \log p_{AB}(a,b)\, da\, db$$

Mutual Information

- If A and B independent:

$$H(A,B) = H(A) + H(B)$$

- If A and B not independent:

$$H(A,B) < H(A) + H(B)$$

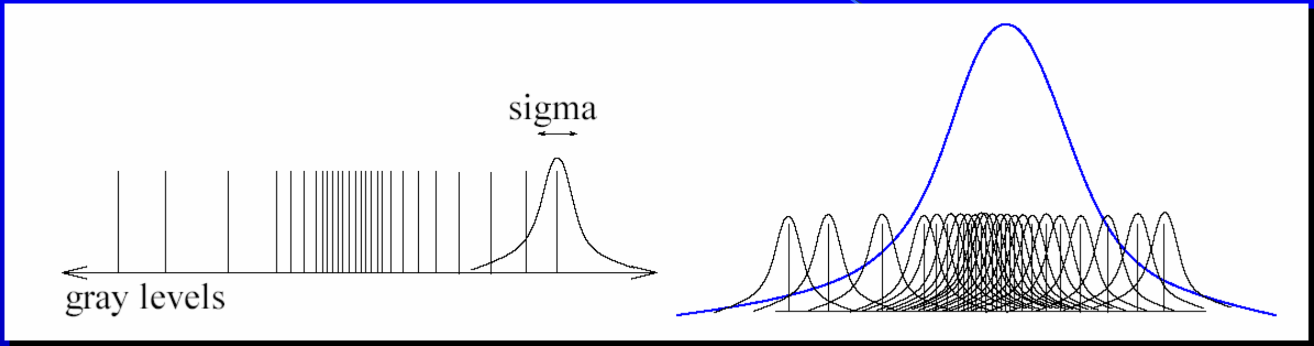
- The difference is MI:

$$I(A,B) = H(A) + H(B) - H(A,B)$$

Estimating the Probabilistic Models

- Entropy is defined in context of a probabilistic model of the data source
  - Typically there is no direct access to these models
- Marginal and joint densities has to be estimated from the image data
  - Parzen windowing
  - Kernel density estimate
  - Histogram binning

Parzen Windowing



- Density function is constructed by superimposing kernel functions centered on the intensity samples obtained from the image
- Kernel type
  - Gaussian, boxcar, B-Spline
- Kernel width (crucial)
  - Depend on dynamic range of data
  - Normalize data
- Number of samples

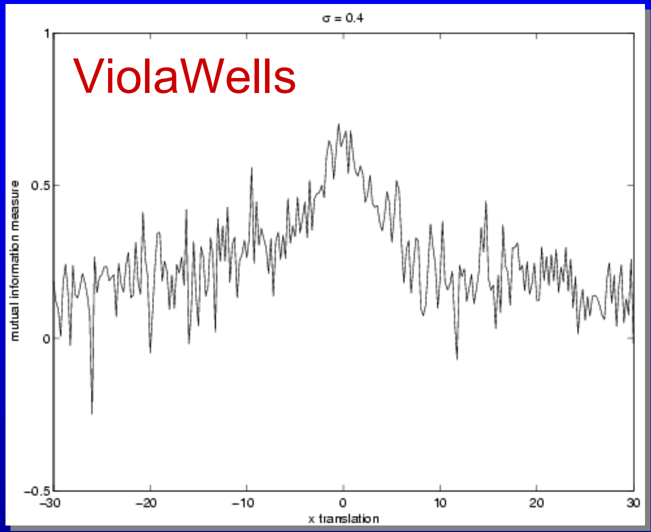
Estimating Entropy

- Using the density estimate, approximate entropy integral by a sum
  - Evaluate at discrete positions/bins uniformly spread within dynamic range
  - Computing a sampled mean using another set of intensity samples randomly drawn from image

Flavors of Mutual Information

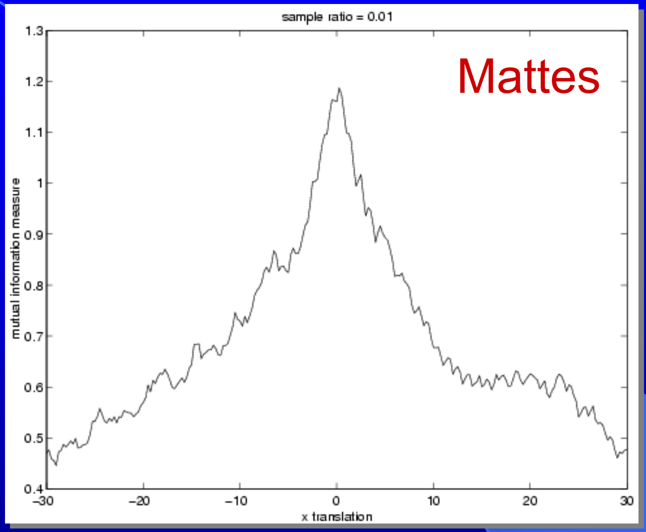
- itk::MutualInformationImageToImageMetric
  - Viola and Wells
  - New samples (50-100) each iteration
- itk::MattesMutualInformationImageToImageMetric
  - Mattes et al, Thevenaz and Unser
  - One set of samples (5-10% of image)
- itk::MutualInformationHistogramImageToImageMetric
  - Use all pixels

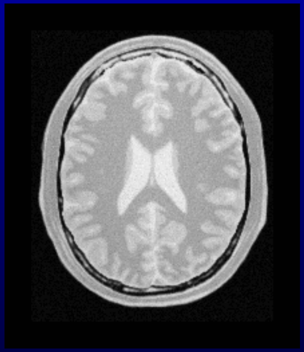
Mutual Information Metric

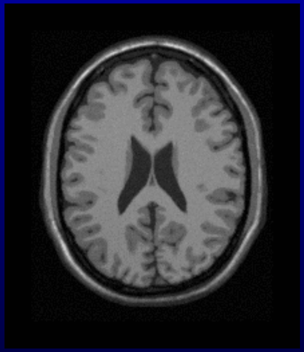


Optimal Value at maximum

Metric range: Image dependent



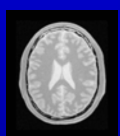


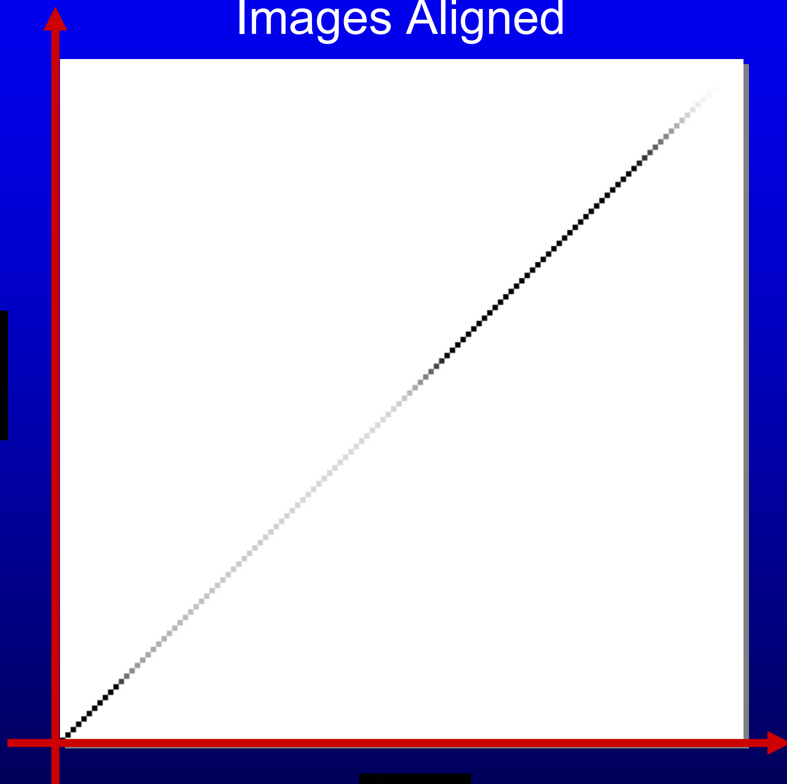


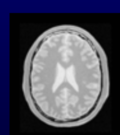
Translations

Joint Histograms: Mono-modality

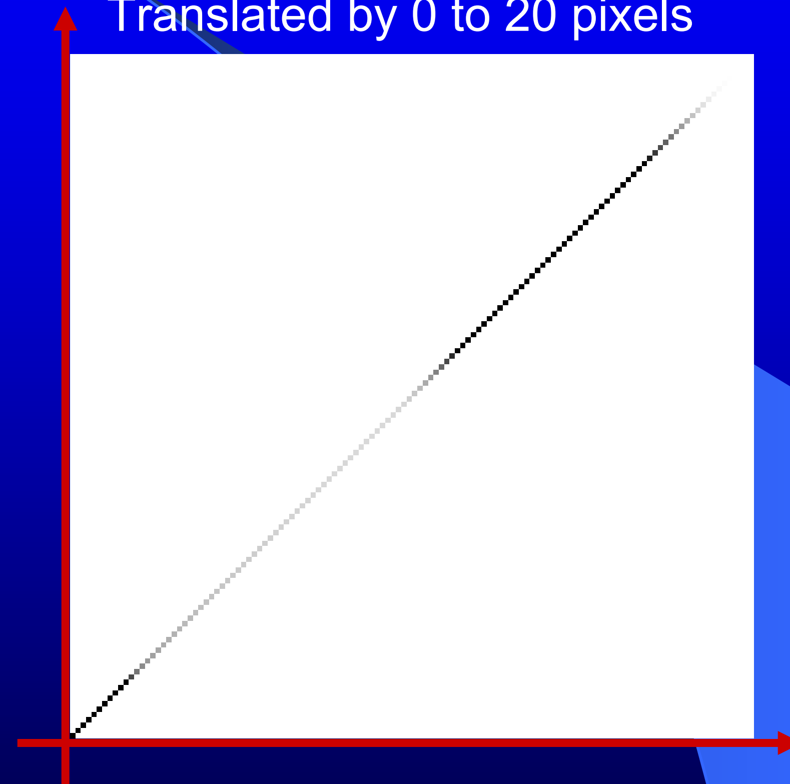
Images Aligned







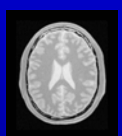
Translated by 0 to 20 pixels



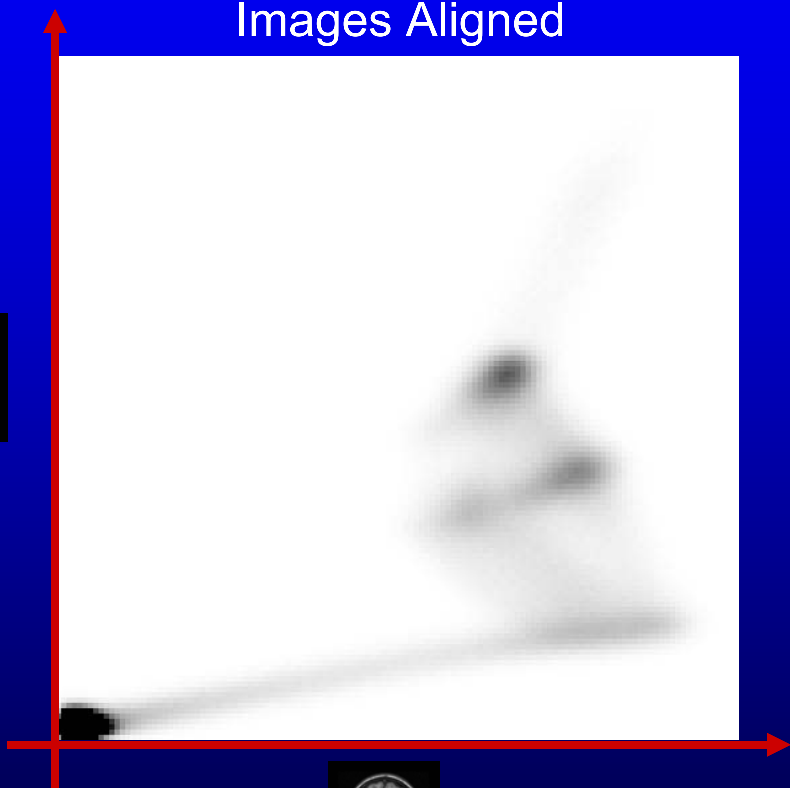
White = zero value
Black = highest value

Misalignment causes dispersion

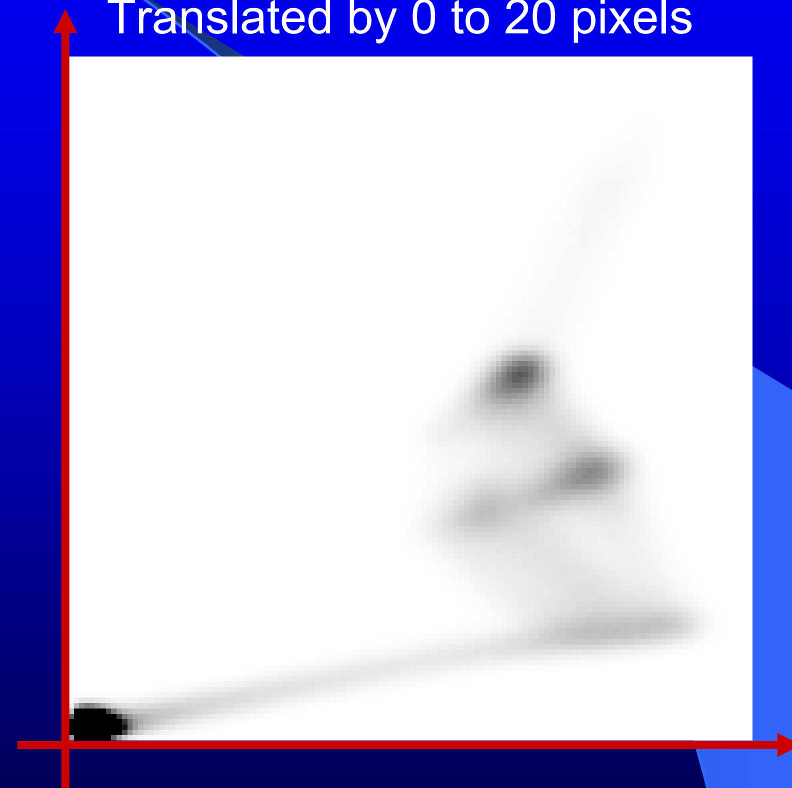
Joint Histograms: Multi-modality



Images Aligned



Translated by 0 to 20 pixels



White = zero value
Black = highest value

Misalignment causes dispersion

Joint Histograms and Registration

- Seek transform that produces a small number of high value bins and as many zero bins as possible == minimizing joint entropy
- Joint entropy metric favors transforms which causes the images to be far apart as possible (i.e. minimum overlap)
- MI overcomes this problem by also trying to maximize the information contributed by each image in the overlap region

itk::ImageRegistrationMethod

- Simple helper driver class that connects up the components and starts the optimizer
- Examples in Software Guide

Examples/Registration

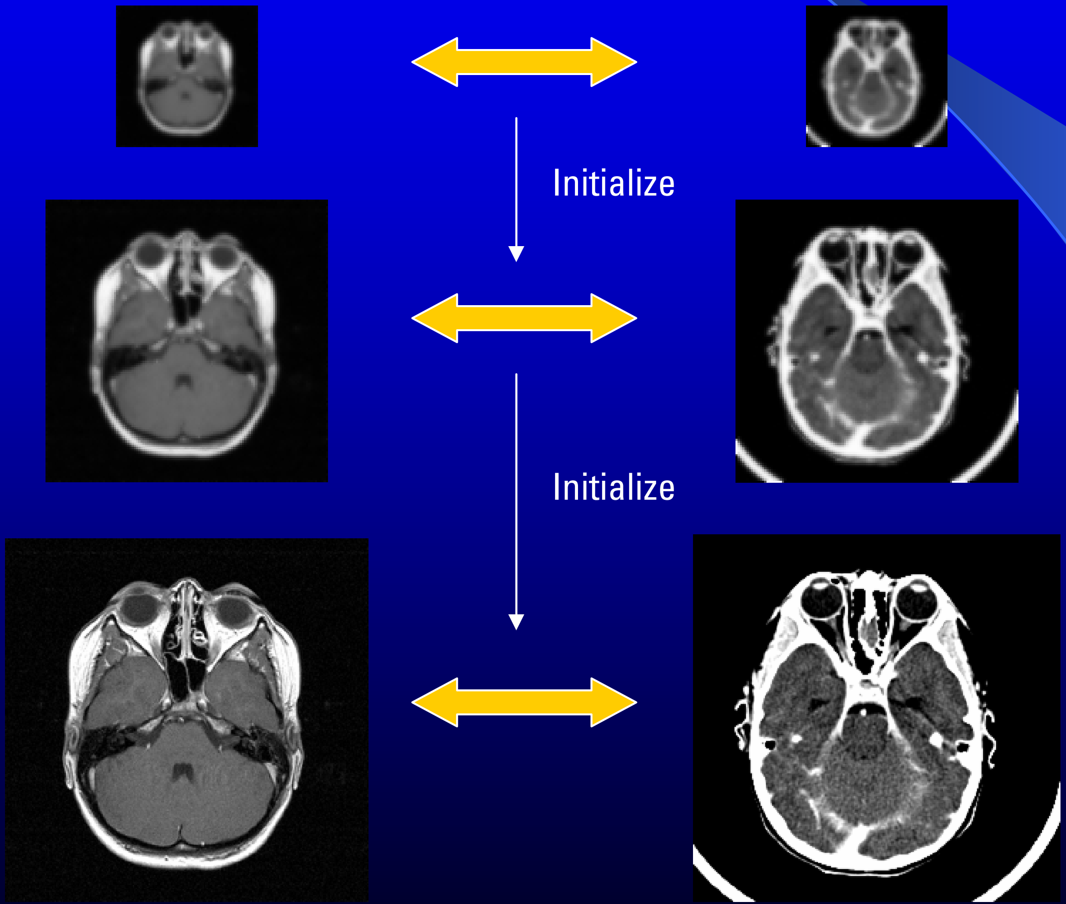
- Declare types
- Instantiate objects via New()
- Connect components and images to the driver using Set methods
- Set initial transform parameters
  – Don't forget!
- Setup each component
  – optimization parameters: step length, convergence …
  – MI parameters: number of samples,
- Connect up observers
- StartRegistration()
  – Should do this inside try/catch block
- Get the last transform parameters
- Create registered image using ResampledImageFilter

Registration Strategies

- Naive application of registration may not work
- Coarse to fine strategy
  - Improves computational speed and robustness
- Low to high dimensional transform
  - Translation to rigid to deformable
  - Sparse to fine grid

itk::MultiResolutionImageRegistrationMethod

- Helper driver class to perform multi-resolution registration

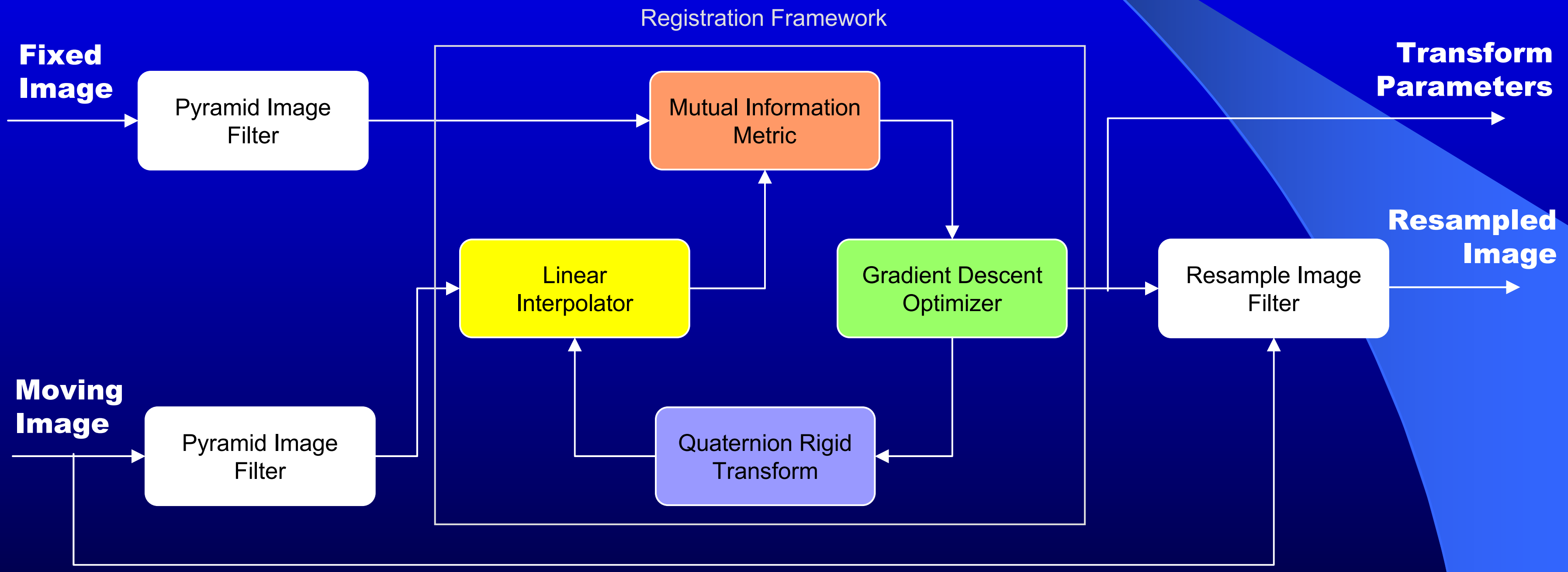


Examples

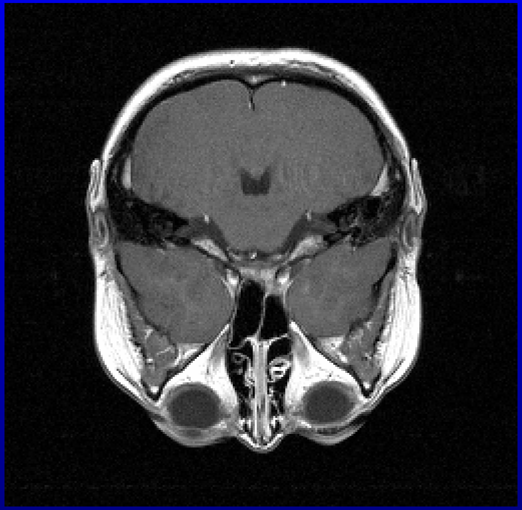
Rigid Registration of Multi-modality Images of the Brain

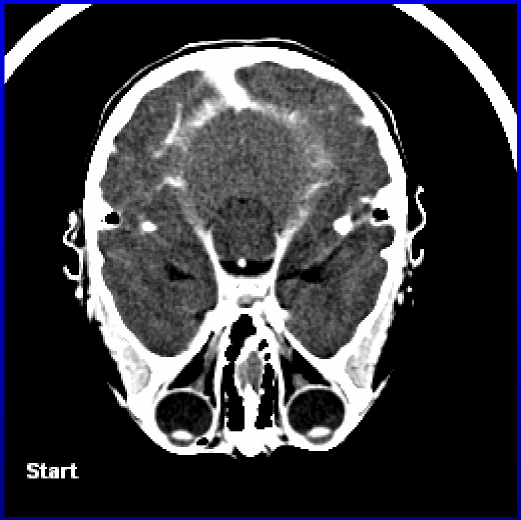
- “Retrospective Image Registration Project” (PI: Dr Fitzpatrick, Vanderbilt University)
- Multi-modality images of patient undergoing neurosurgery
- Fidicual markers attached to skull before imaging
- Airbrushed out from images to allow blind valuation of retrospective registration methods
- http://www.vuse.vanderbilt.edu/~image/registration/

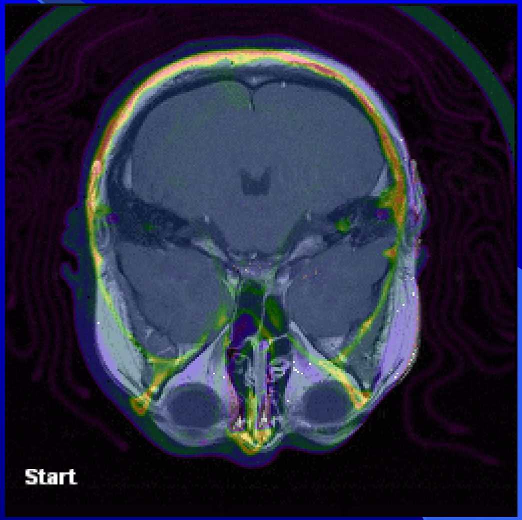
Multi-resolution Multi-modality Rigid Registration Process



3D CT to MR-T1 Rigid Registration



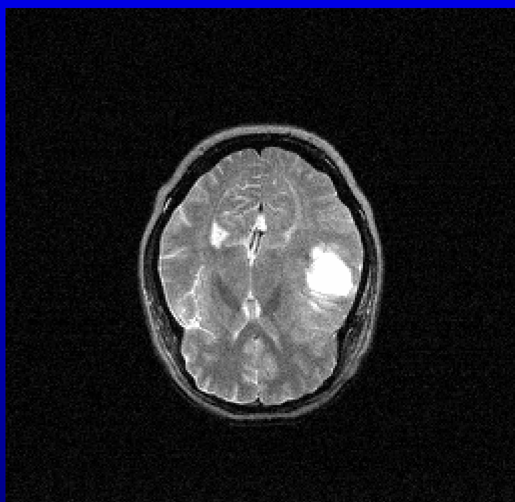


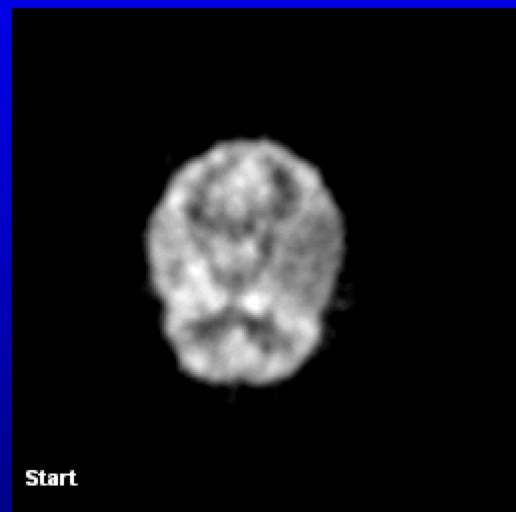


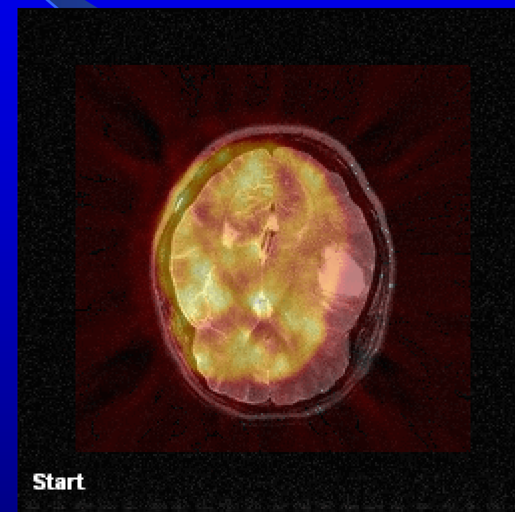
Fixed Image: MR-T1, 256 x 256 x 52 pixels, 0.78 x 0.78 x 3.00 mm
Moving Image: CT, 512 x 512 x 44, 0.41 x 0.41 x 3.00 mm
Registration: 4 levels, MI, gradient descent, quaternion rigid

Images provided as part of the project: “Retrospective Image Registration Evaluation”,
NIH, Project No. 8R01EB002124-03, Principal Investigator, J. Michael Fitzpatrick, Vanderbilt University, Nashville, TN.

3D PET to MR-T2 Rigid Registration







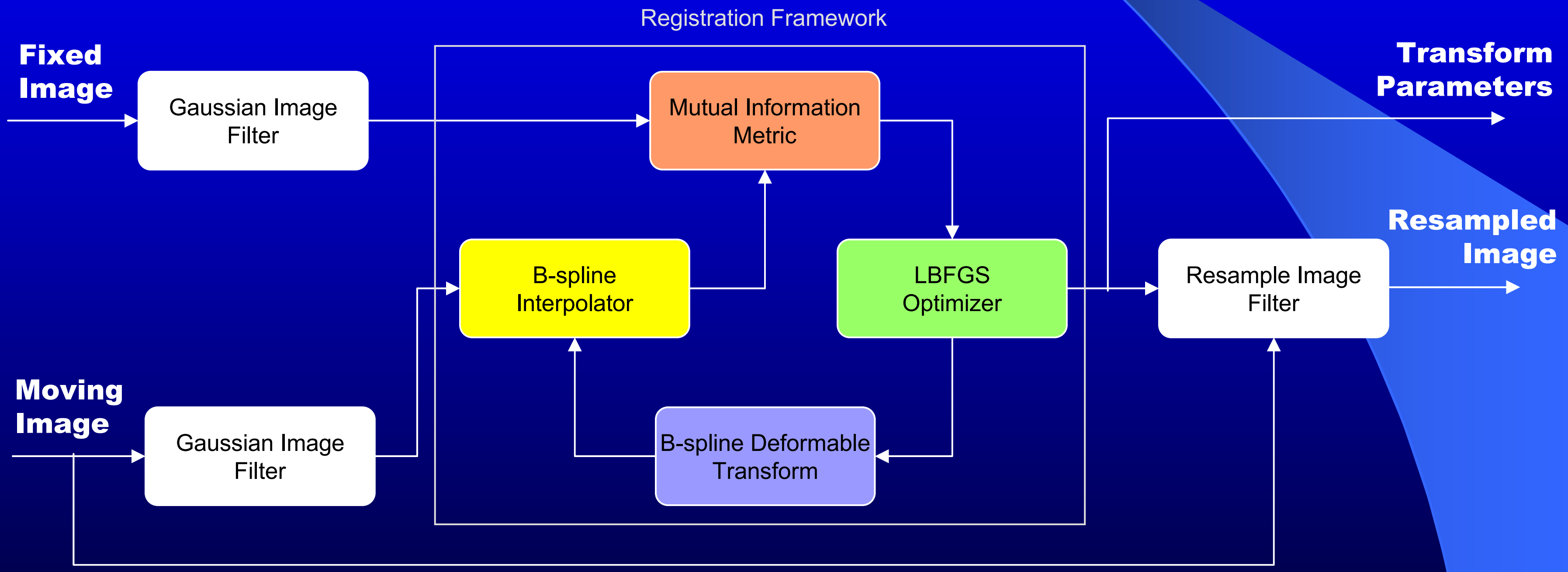
Fixed Image: MR-T2, 256 x 256 x 26 pixels, 1.25 x 1.25 x 4.00 mm
Moving Image: PET, 128 x 128 x 15, 1.94 x 1.94 x 8.00 mm
Registration: 3 levels, MI, gradient descent, quaternion rigid

Images provided as part of the project: “Retrospective Image Registration Evaluation”,
NIH, Project No. 8R01EB002124-03, Principal Investigator, J. Michael Fitzpatrick, Vanderbilt University, Nashville, TN.

Time Series Registration of Dynamic Contrast Uptake Breast MRI Images

- Valuable tool for early detection of breast cancer
- Contrast agent injected via a catheter
- Scan acquired before injection and followed by a series of scans post-injection
- Total exam time approx. 15 mins
- Inspect the contrast uptake curves
  - Discriminate between malignant and benign lesions
- Remove effects due to patient movement

Multi-resolution Mutual Information Deformable Registration Process



3D Breast MR Contrast Uptake Deformable Registration

Image data courtesy of University of Washington.

Unregistered:

Registered:

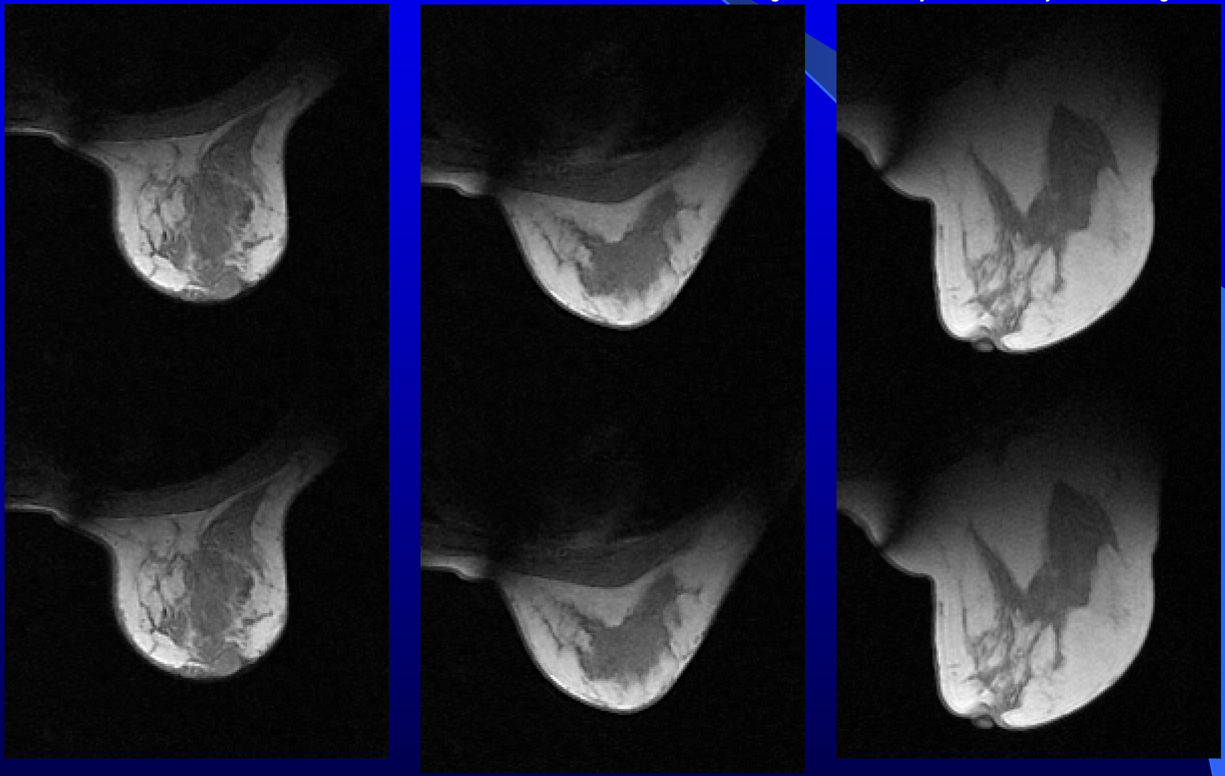


Image Data: 192 x 192 x 13 pixels, 0.94 x 0.94 x 8.00 mm
Registration: 2 levels, MI, LBFGS, B-spline deformation

Additional Examples

- ITK Software Guide
  - 9 examples
- InsightApplications
  - ImageRegistration
  - MRIRegistration
  - MultiResMIRegistration
  - MutualInformationEuler2DRegistration
- Landmark Initialized 3D Registration Tool
  - CADD Lab, UNC

GUI Communication

Registration To GUI Communication

- Drive progress bar
- Observe registration progress
  - Interrogate metric values
  - Interrogate current parameters
  - Display current results
    - Resample and display ROI
- Change component parameters
- Terminate registration

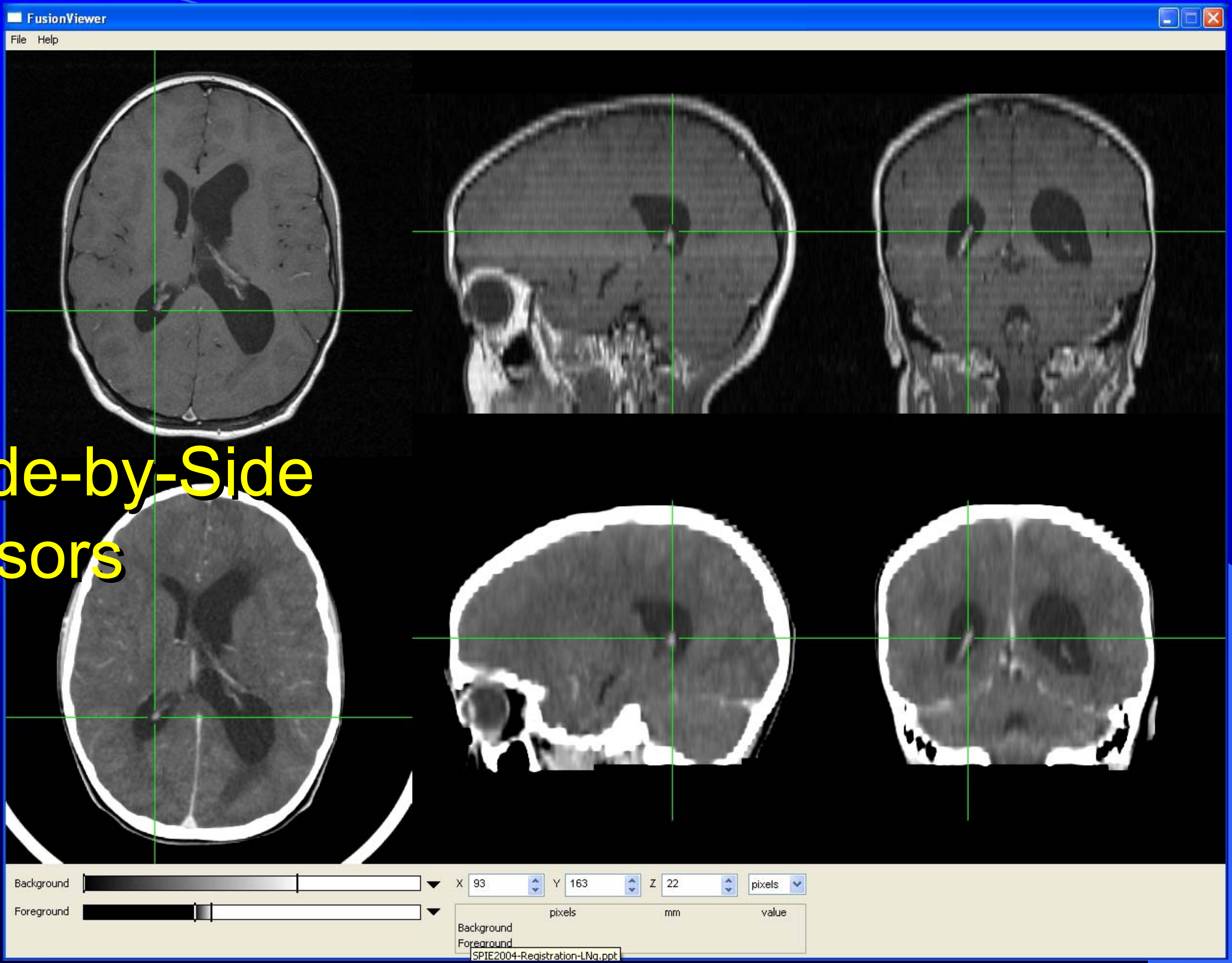
ITK Observers/Commands

- itk::Object can invoke events
  – Start, End, Progress, Iteration
- itk::Object maintains a linked list of event observers
- Observers register themselves to an object and declare the type of events they are interested in

Observing Registration

- ITK Optimizers typically execute an iterative process
- Most Optimizers invoke an IterationEvent at the end of each iteration
- Observing IterationEvents provide periodic communication facilitating progress feedback and opportunity to control the registration
- itk::MultiResolutionImageRegistrationMethod also invokes an IterationEvent between levels to give the GUI an opportunity to change components and/or parameters
- Examples in Software Guide

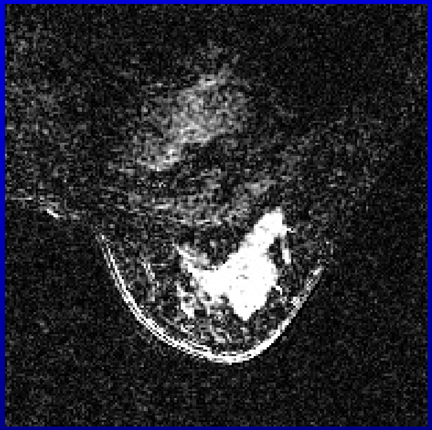
Displaying Registered Images



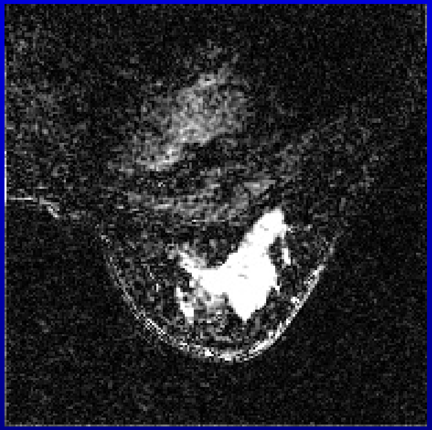
View Side-by-Side
Link cursors

(Absolute) Image Difference

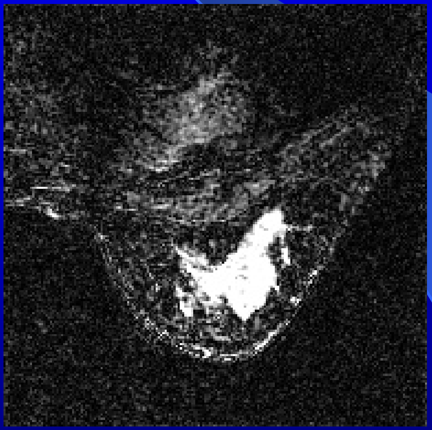
3D Dynamic MRI Images of the breast: before and after contrast injection



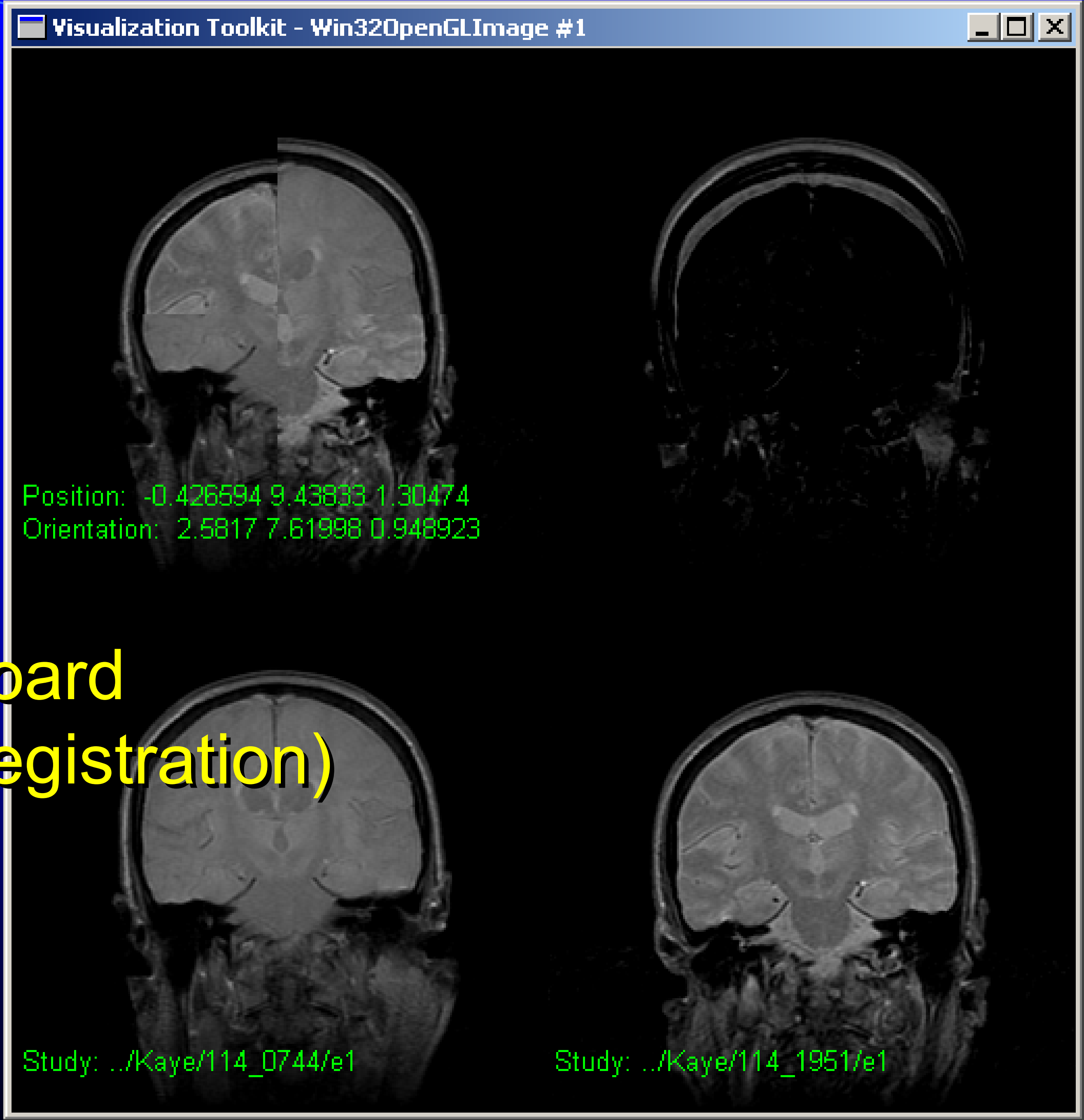
Unregistered



Rigid Registration



B-Spline Deformable Registration



Checkerboard (Before Registration)

Image Courtesy of GE Research

Checkerboard (After Registration)

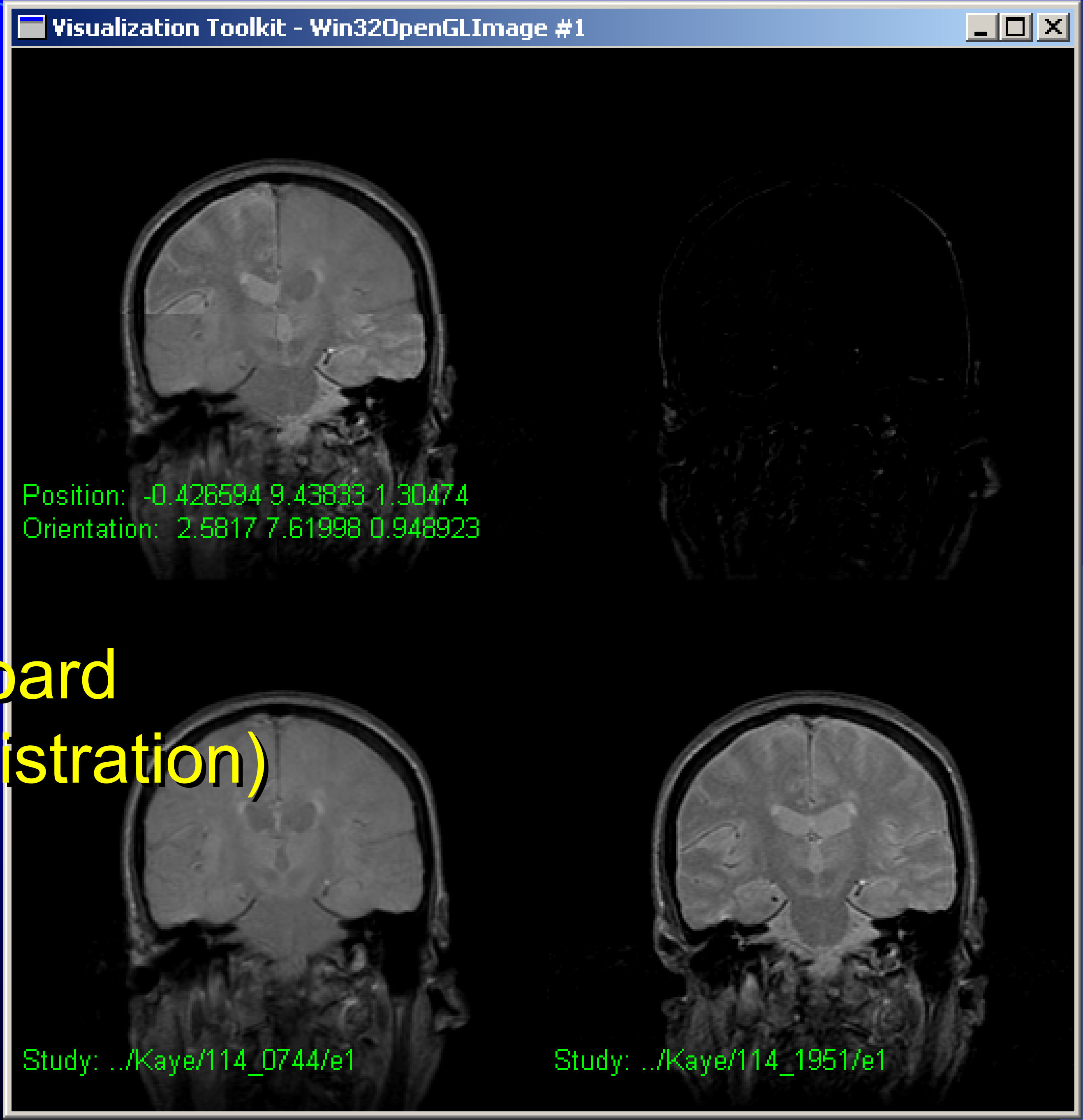
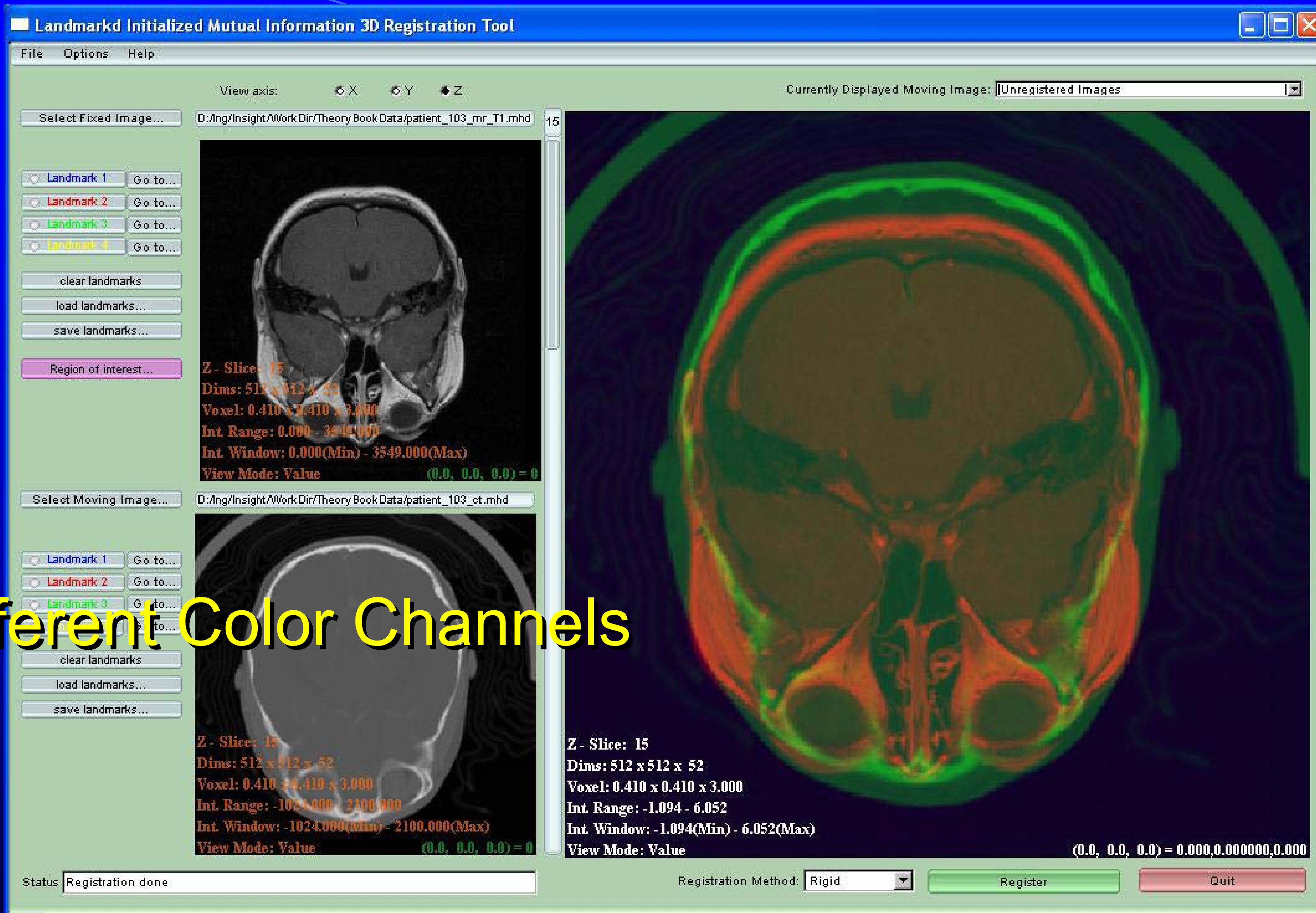
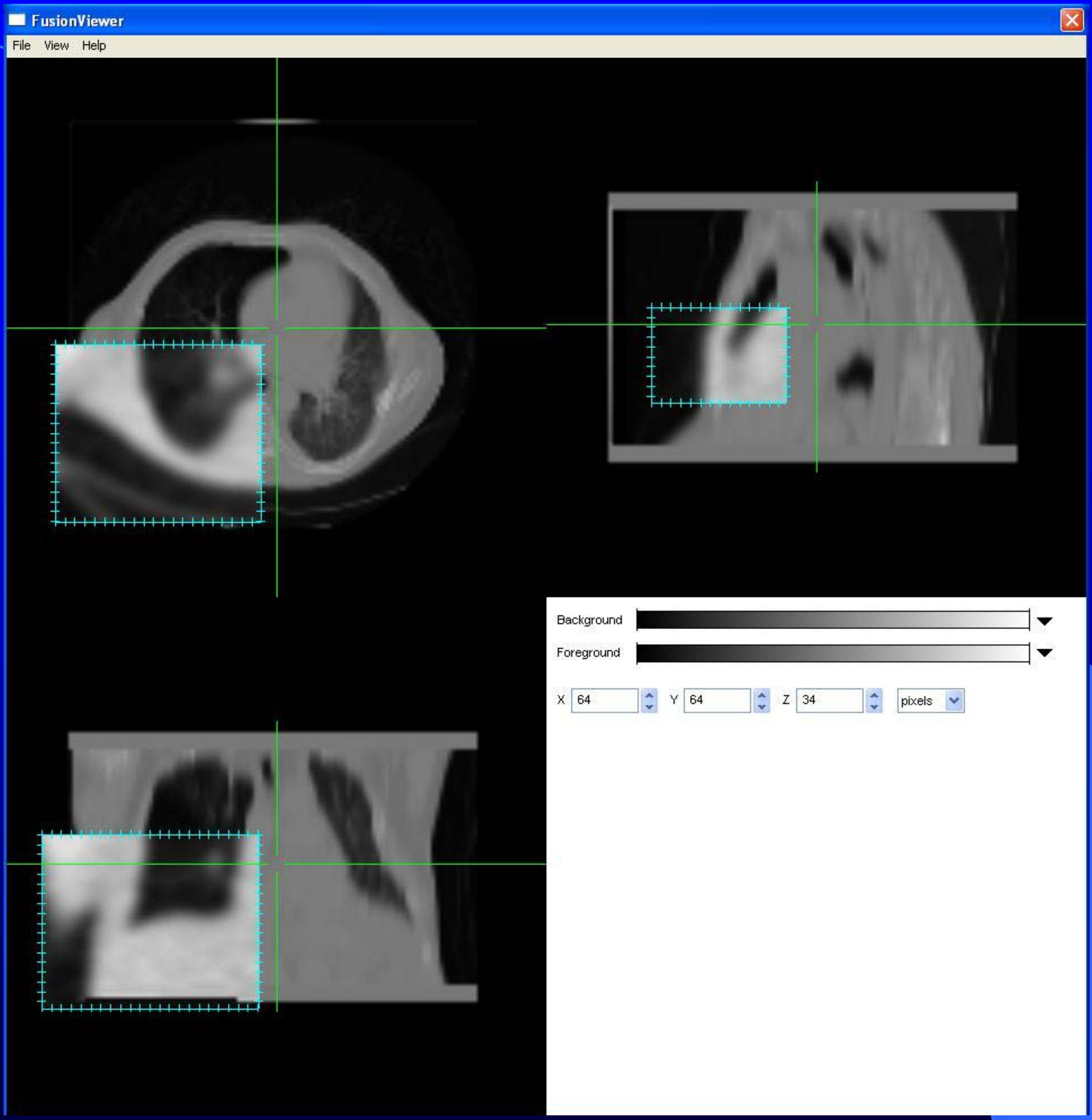


Image Courtesy of GE Research

Different Color Channels

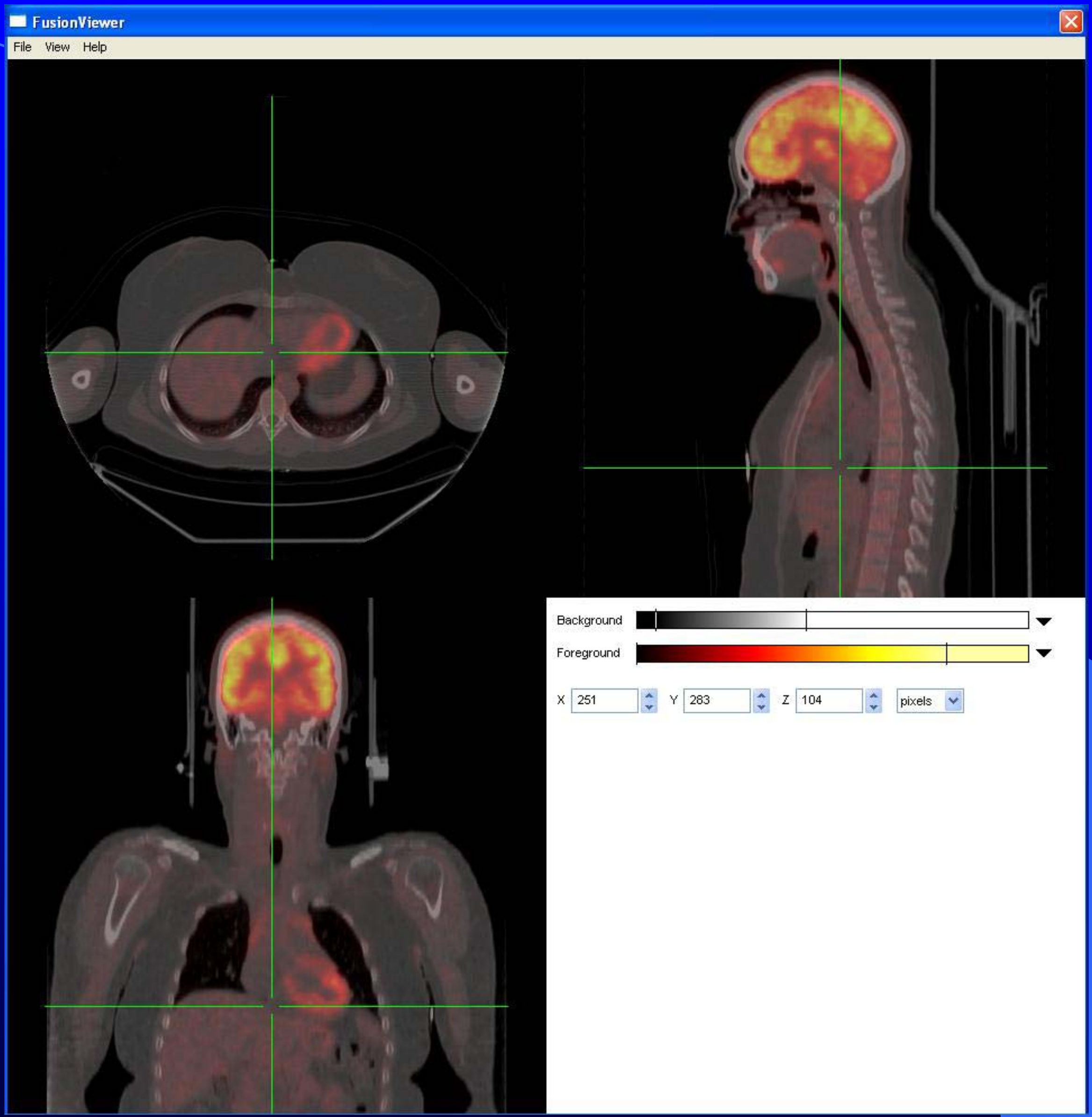


Split-View: Movable and adjustable window



Copyright Insightful Corp.

Alpha-blended overlay: image intensity mapped to a colorscale



Copyright Insightful Corp.

Deformable Registration Techniques in ITK

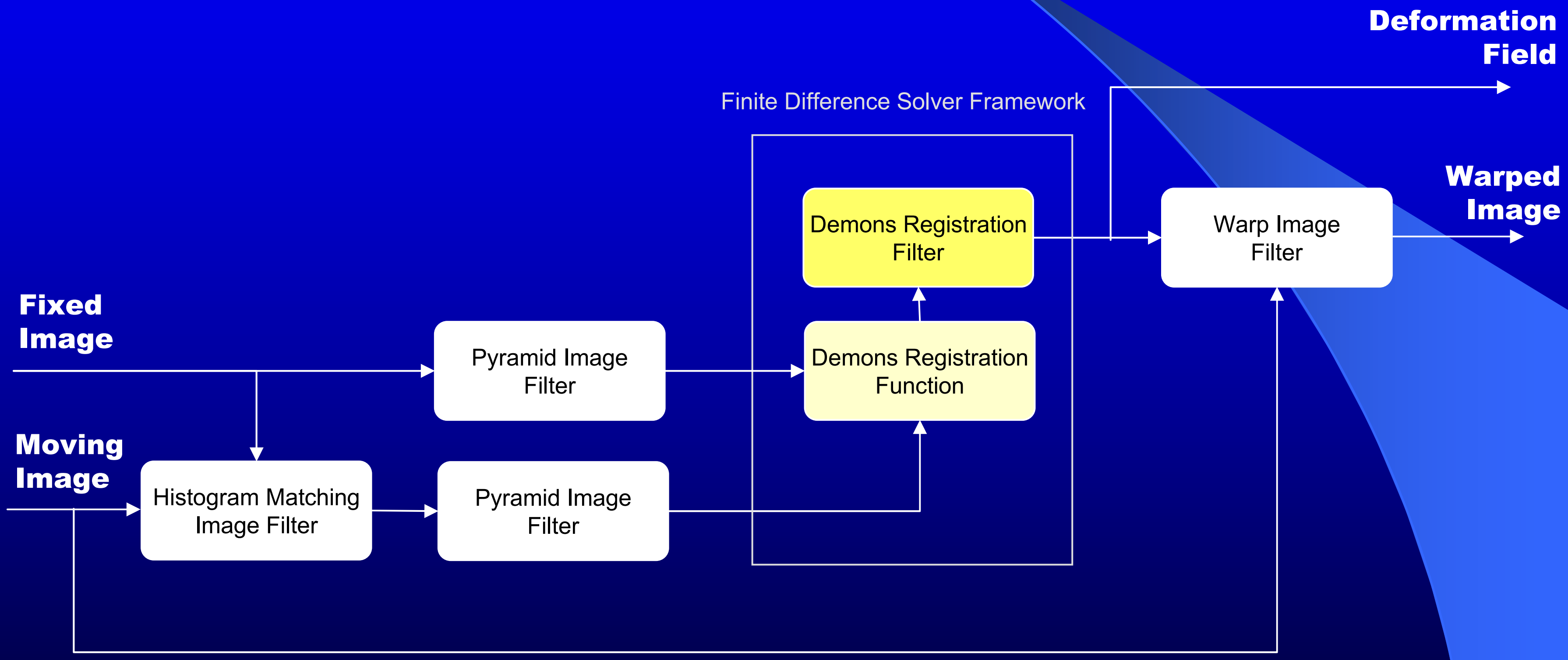
Deformable Registration in ITK

- Low dimensional transformations can be handled in the basic registration framework
- For high dimensional transformations:
  - Finite Element Methods (FEM)
  - Finite Different Methods
- Metric optimization
  - Mean squares, normalized correlation, MI
- FEM framework additionally supports regularization and landmark constraints:
  - Diffeomorphic constraints
  - Linear elastic and large deformation (fluid and transient-quadratic) models
- Simple examples in Software Guide

Inter-subject Registration and Atlas-based Segmentation

- Segmentation done once on an atlas image
- Subject image is deformable registered to the atlas
- Resulting deformation field is used to warp the atlas label to automatically generate labels for the subject image

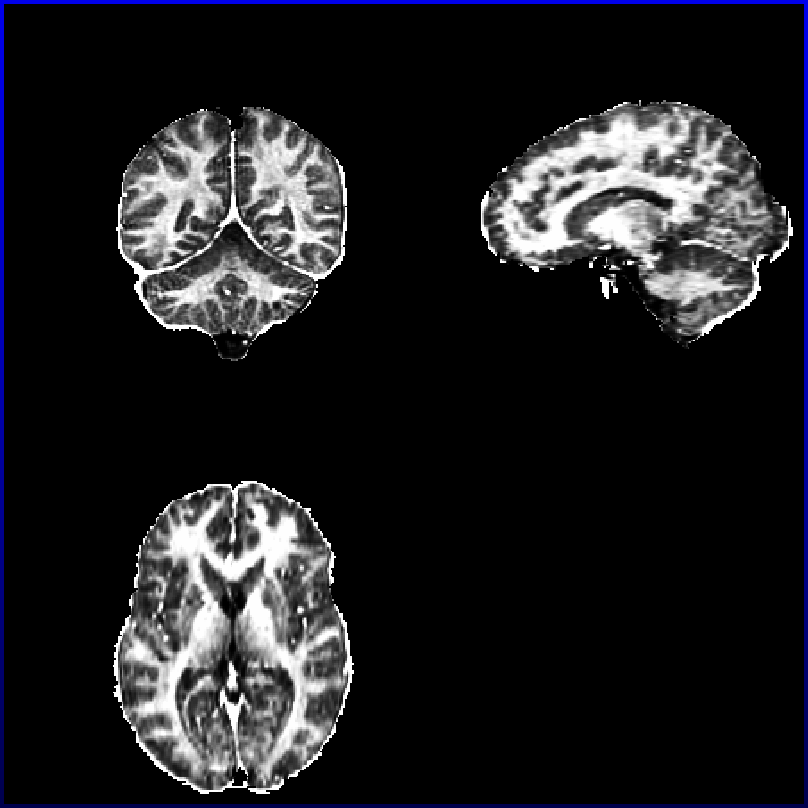
Multi-resolution Demons-based Deformable Registration Process

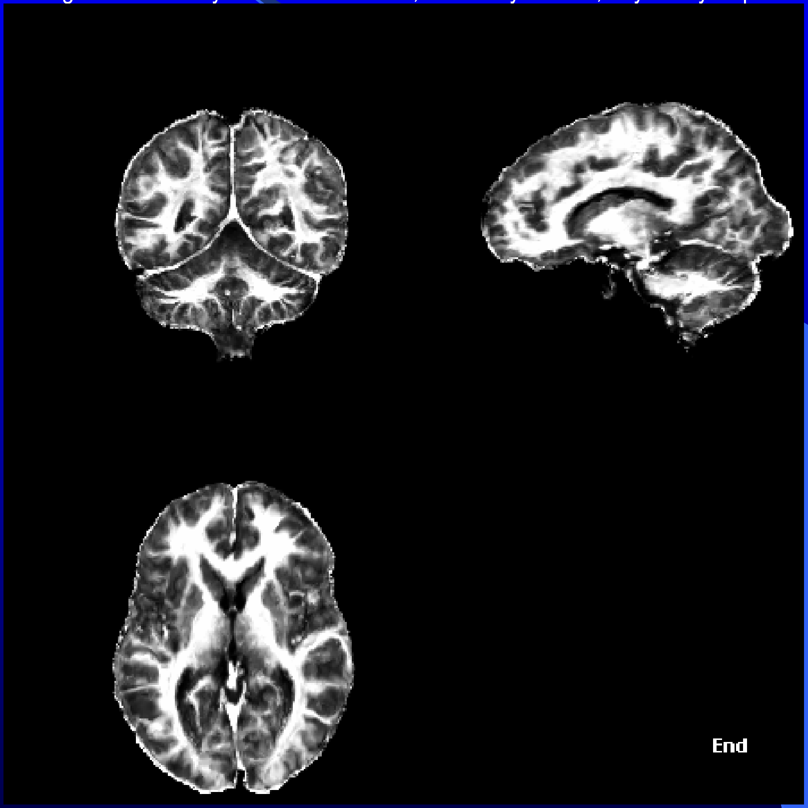


3D Inter-subject MR-PD Deformable Registration

Image data courtesy of N. C. Andreasen, University of Iowa, Psychiatry Dept.

Fixed Volume





Moving Volume

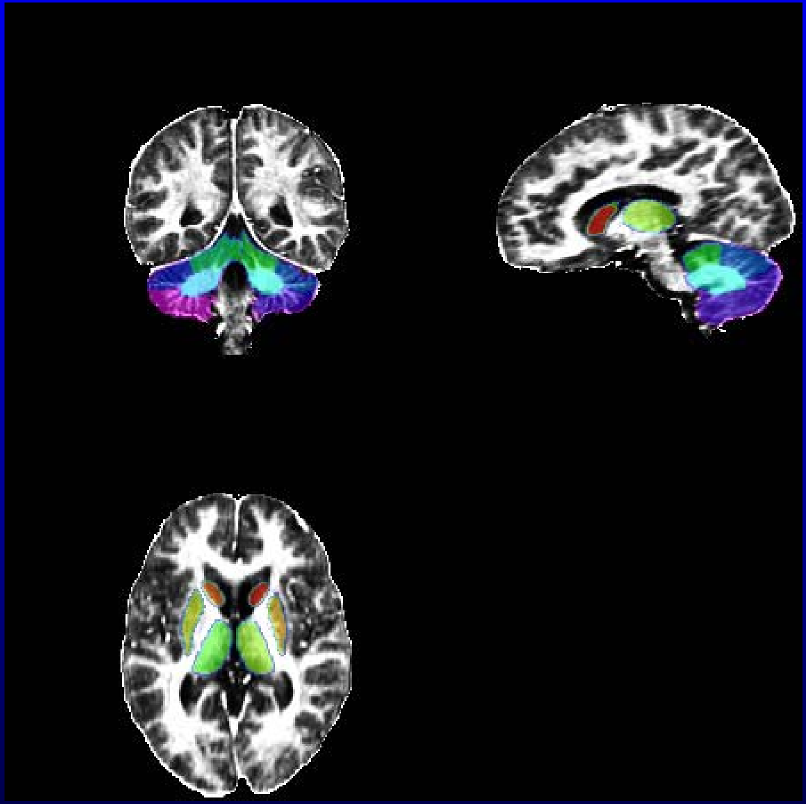
Image Data: 256 x 256 x 192 pixels, 1.02 x 1.02 x 1.02 mm

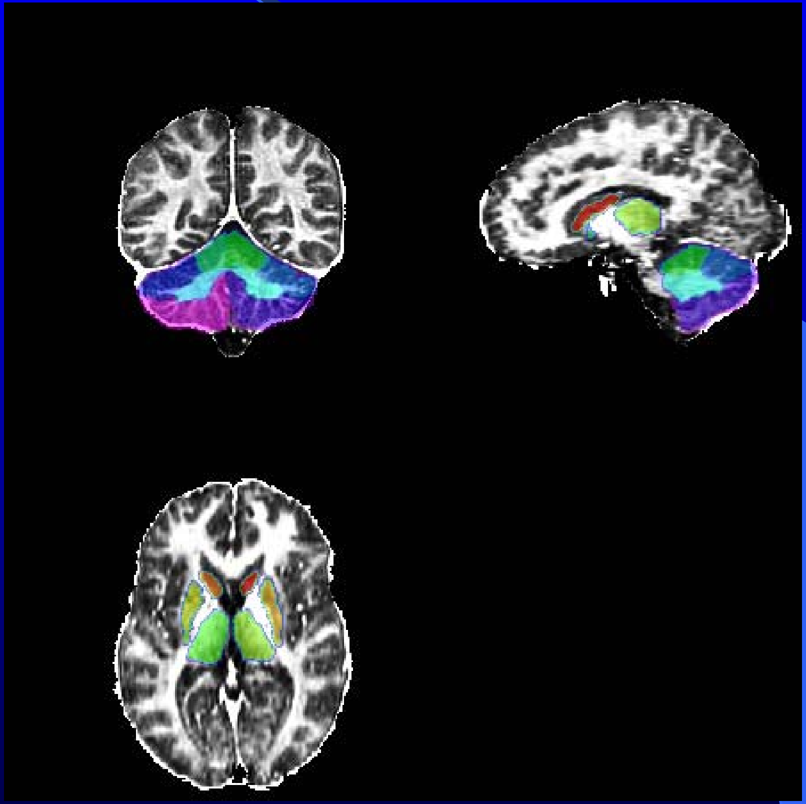
Registration: 3 levels, Demons algorithm, histogram matching

3D Atlas-based Segmentation

Image data courtesy of N. C. Andreasen, University of Iowa, Psychiatry Dept.

Atlas (Moving) Volume





Subject (Fixed) Volume

Resources

- ITK Software Guide
- Insight to Images (coming soon …)
- "A Survey of Medical Image Registration"
  - Maintz and Viergever, Medical Image Analysis, 1998
- "Handbook of Medical Imaging: Medical Image Processing and Analysis"
  - Fitzpatrick et al (eds), SPIE, 2000
- "Handbook of Medical Imaging: Processing and Analysis"
  - Bankman (ed.), Academic Press, 2000
- "Medical Image Registration"
  - Hajnal et al (eds), CRC Press, 2001
- "Mutual-Information-Based Registration of Medical Images: A Survey"
  - Pluim et al, IEEE-TMI, 22(8), 2003
- IEEE Transaction of Medical Imaging Special Issue
  - November 2003

Data for Registration Testing

- Retrospective Registration Project
  - http://www.vuse.vanderbilt.edu/~image/registration/
  - Multi-modality images of the brain
- Internet Brain Segmentation Repository
  - http://www.cma.mgh.harvard.edu/ibsr/
  - MR-TI images of brain with segmentations
- BrainWeb
  - http://www.bic.mni.mcgill.ca/brainweb/
  - Simulated brain database
- International Consortium For Brain Mapping (ICBM)
  - http://www.loni.ucla.edu/ICBM/ICBM_Databases.html