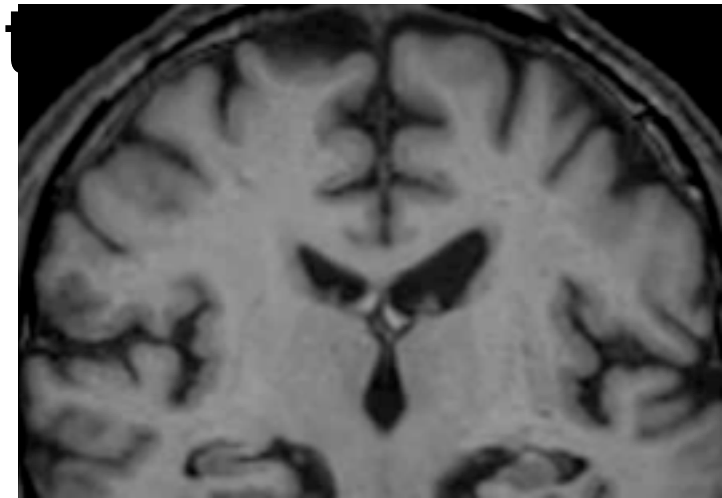
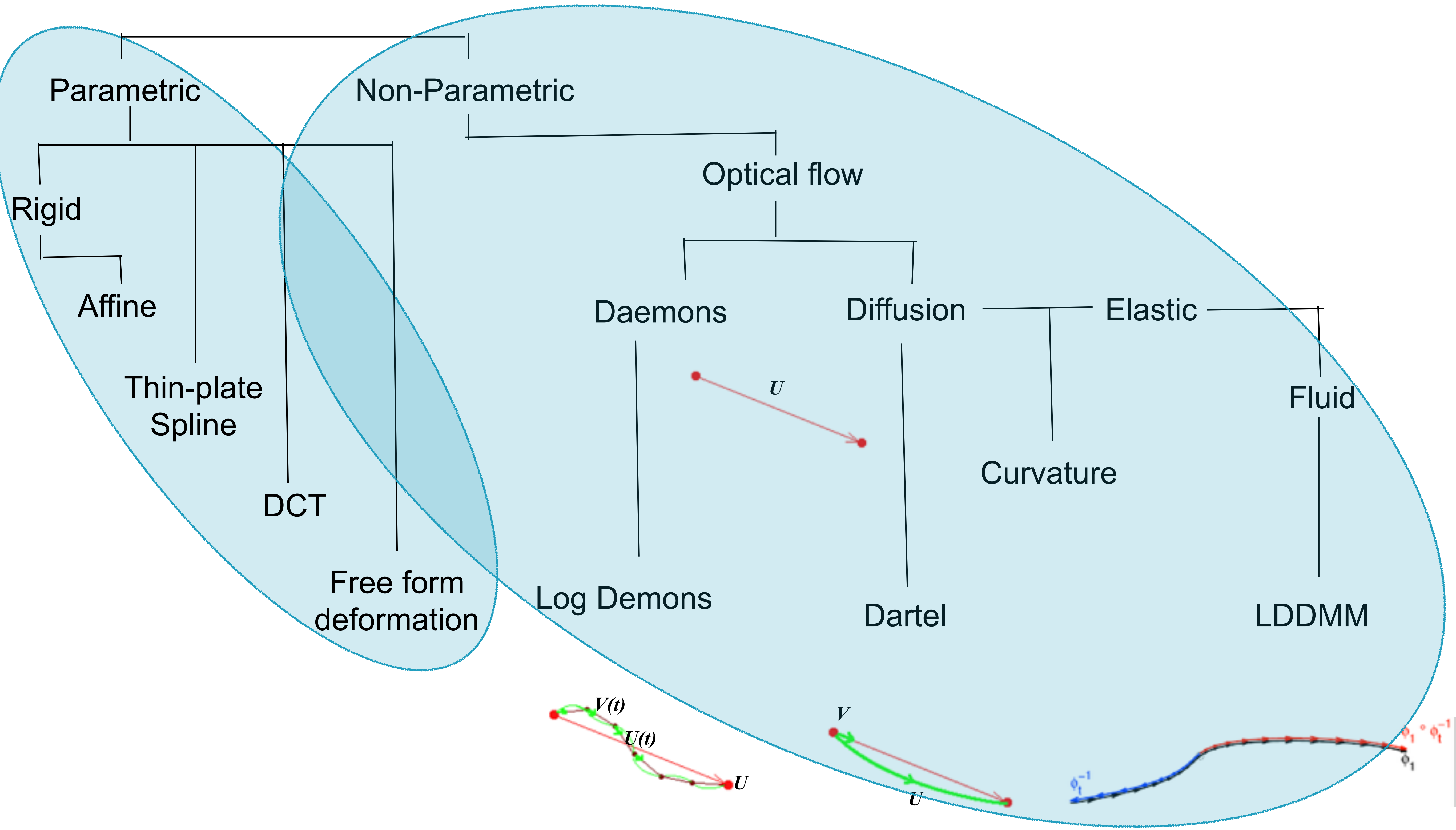
Medical Image Registration
Free Form Deformation

Marc Modat

Centre for Medical Image Comput
Dementia Research Centre
University College London



A taxonomy of reg. algorithms



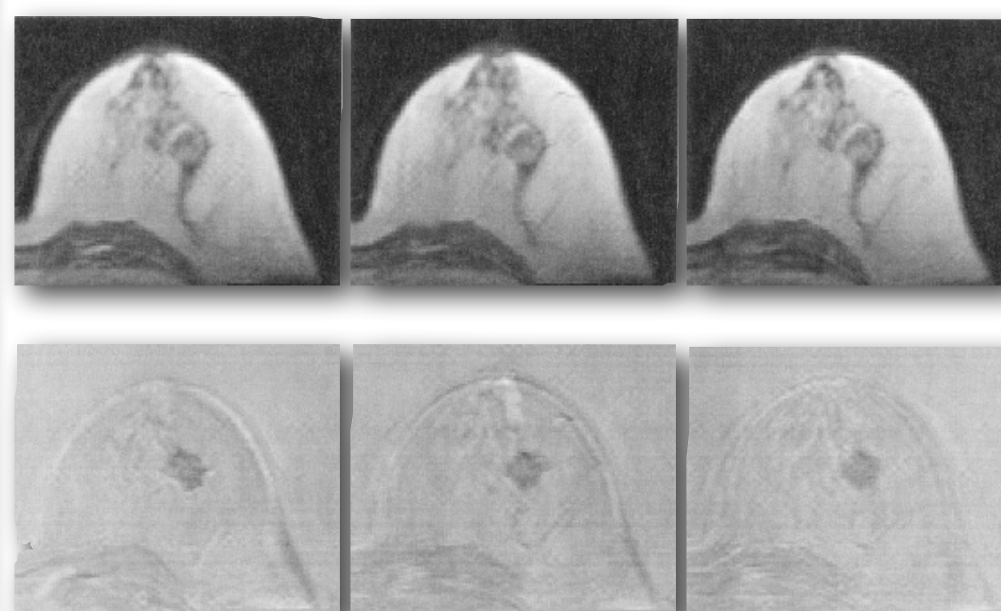
The Free-Form Deformation algorithm

IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. 18, NO. 8, AUGUST 1999

Nonrigid Registration Using Free-Form Deformations: Application to Breast MR Images

D. Rueckert,* L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes

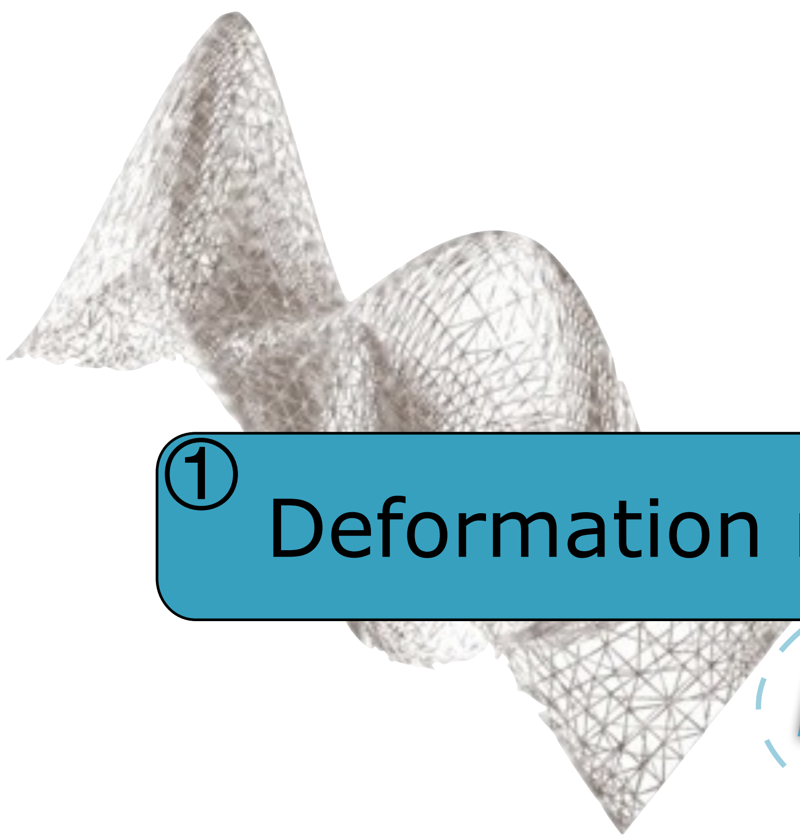
Abstract— In this paper we present a new approach for the nonrigid registration of contrast-enhanced breast MRI. A hierarchical transformation model of the motion of the breast has been developed. The global motion of the breast is modeled by an affine transformation while the local breast motion is described by a free-form deformation (FFD) based on B-splines. Normalized mutual information is used as a voxel-based similarity measure which is insensitive to intensity changes as a result of the contrast enhancement. Registration is achieved by minimizing a cost function, which represents a combination of the cost associated with the smoothness of the transformation and the cost associated with the image similarity. The algorithm has been applied to the fully automated registration of three-dimensional (3-D) breast MRI in volunteers and patients. In particular, we have compared the results of the proposed nonrigid registration algorithm to those obtained using rigid and affine registration techniques. The results clearly indicate that the nonrigid registration algorithm is much better able to recover the motion and deformation of the breast than rigid or affine registration algorithms.

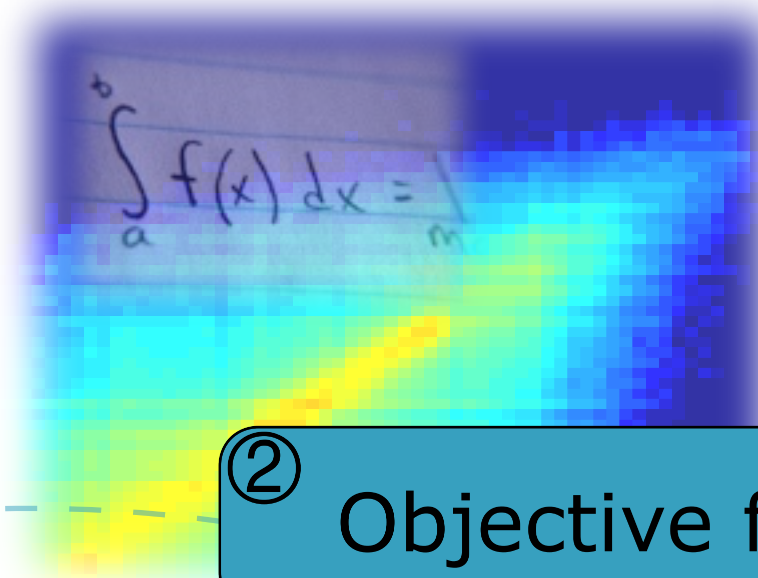


Cited **3363** times
Google citation
15 Jan 2015

Cited **2318** times
Scopus
15 Jan 2015

The original framework - Overview





① Deformation model

② Objective function

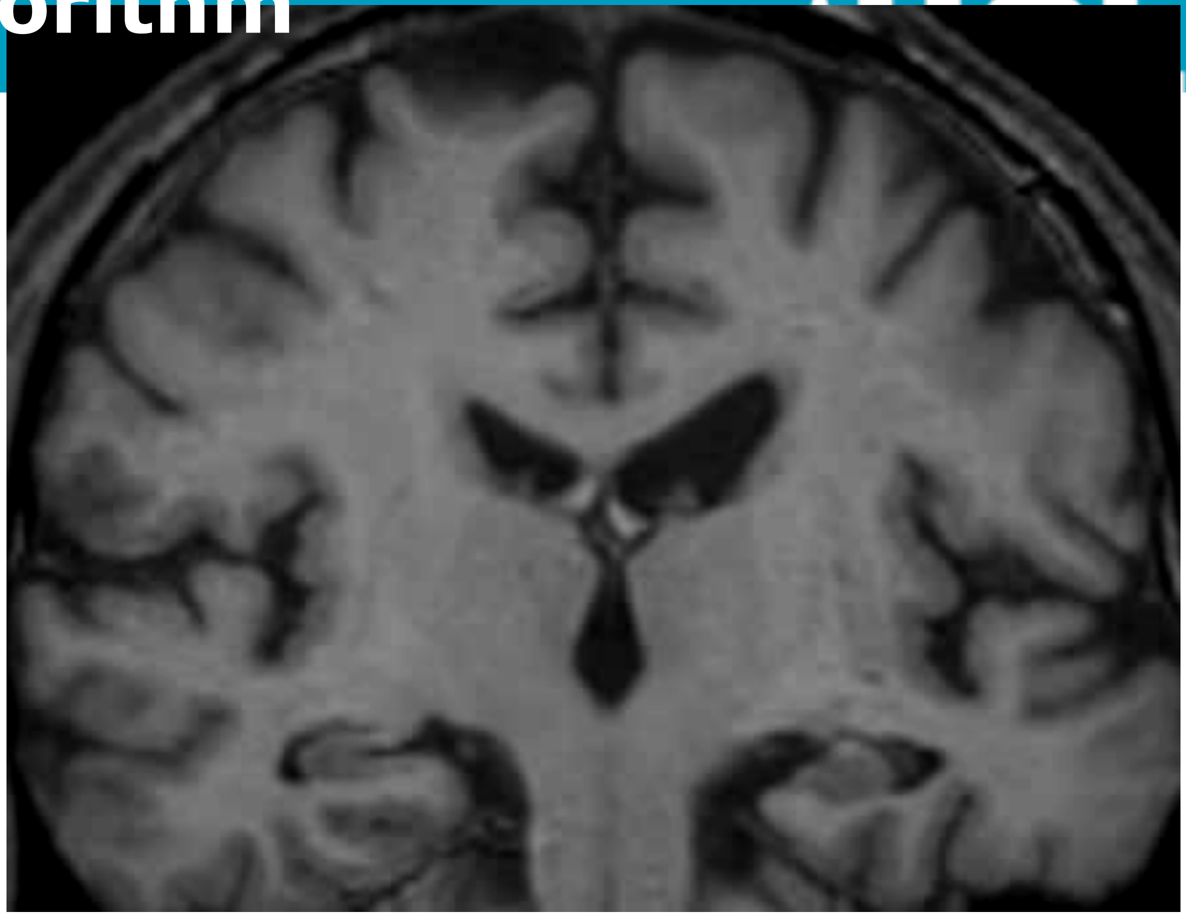
Free Form Deformation

③ Optimisation

① Cubic B-splines transformation model

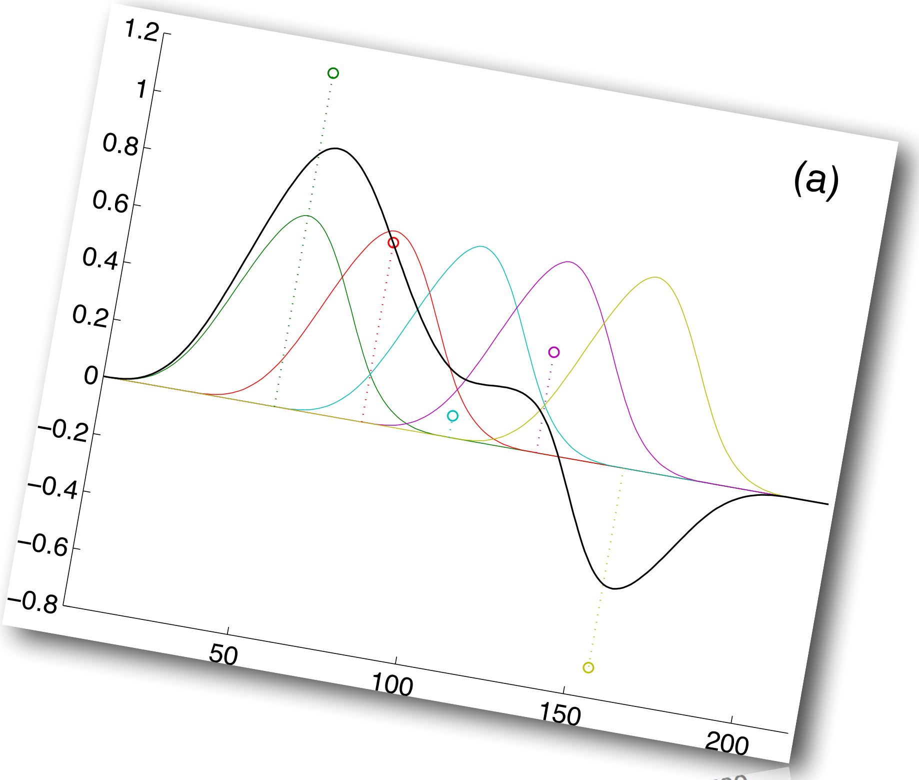
② Normalised Mutual Information
Bending Energy

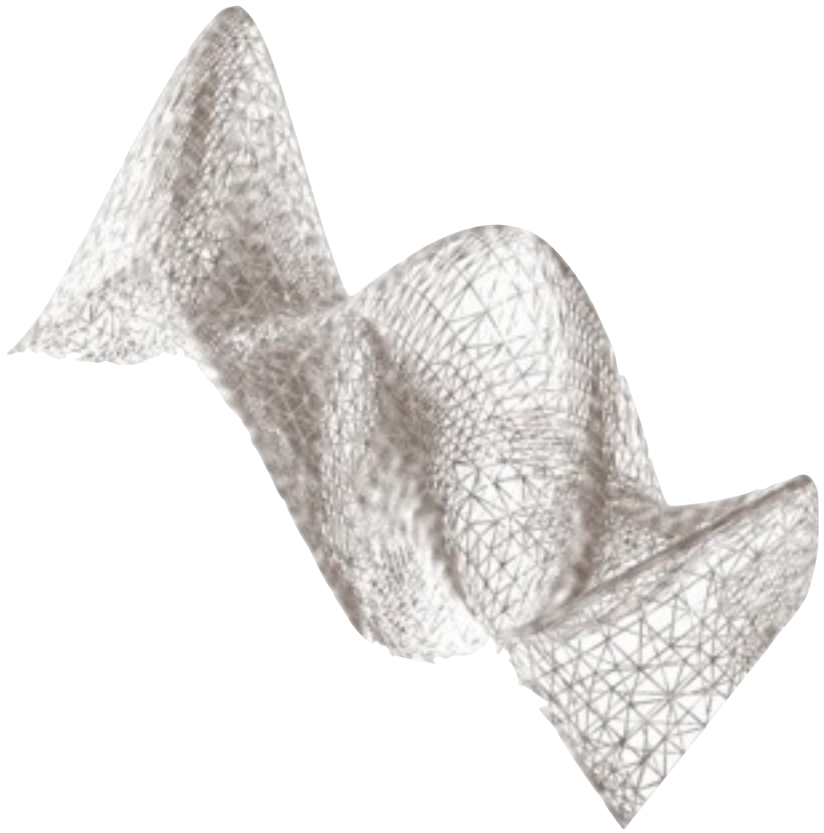
③ Gradient descent



Deformation Model

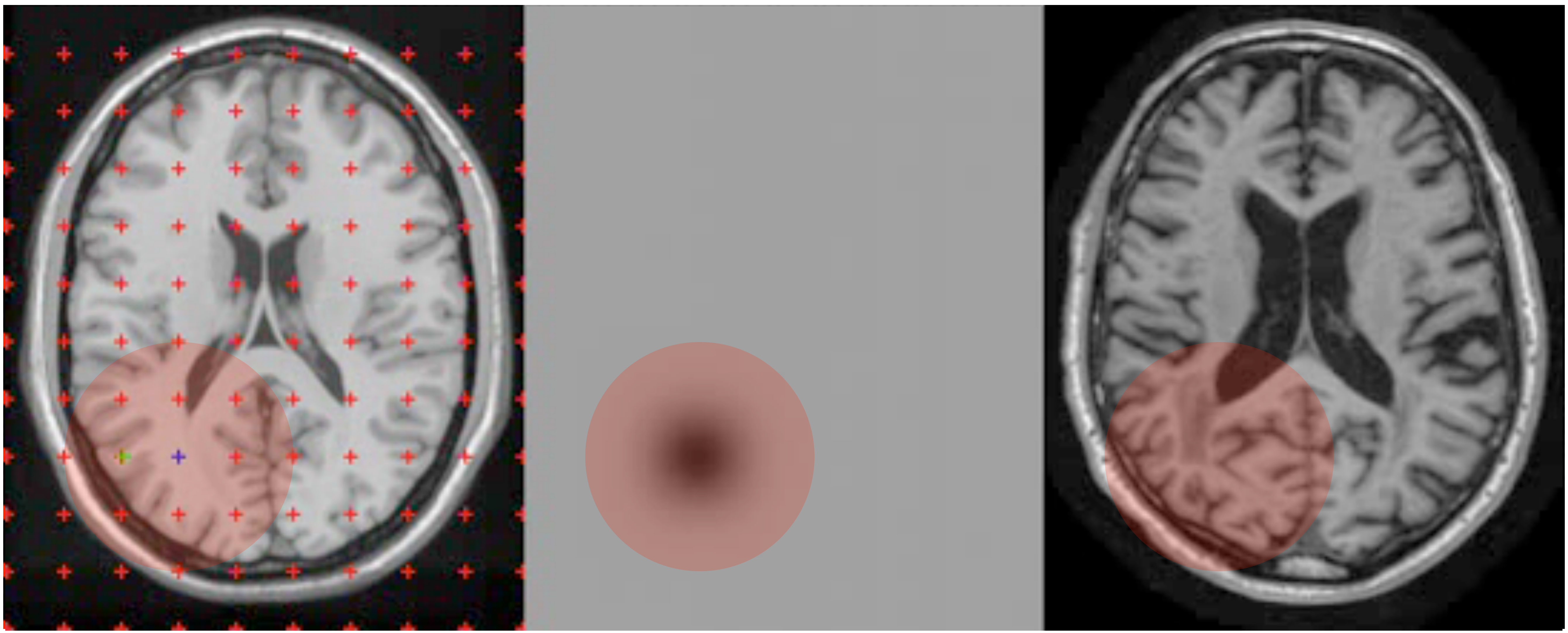
Cubic B-Splines





FFD - Deformation model

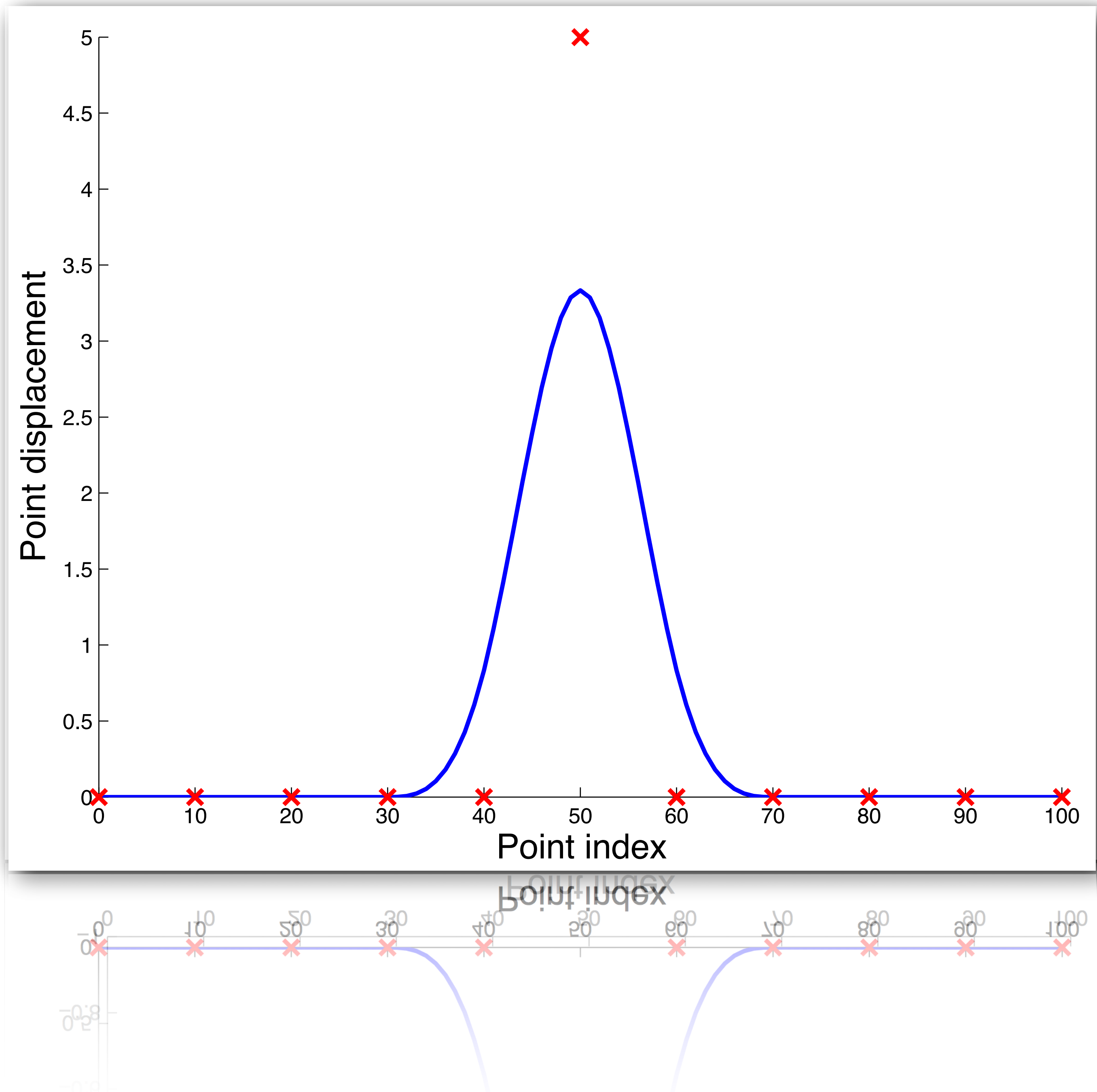
- Aim: to generate a deformation field



Overlaid lattice x-axis displacement Deformed image

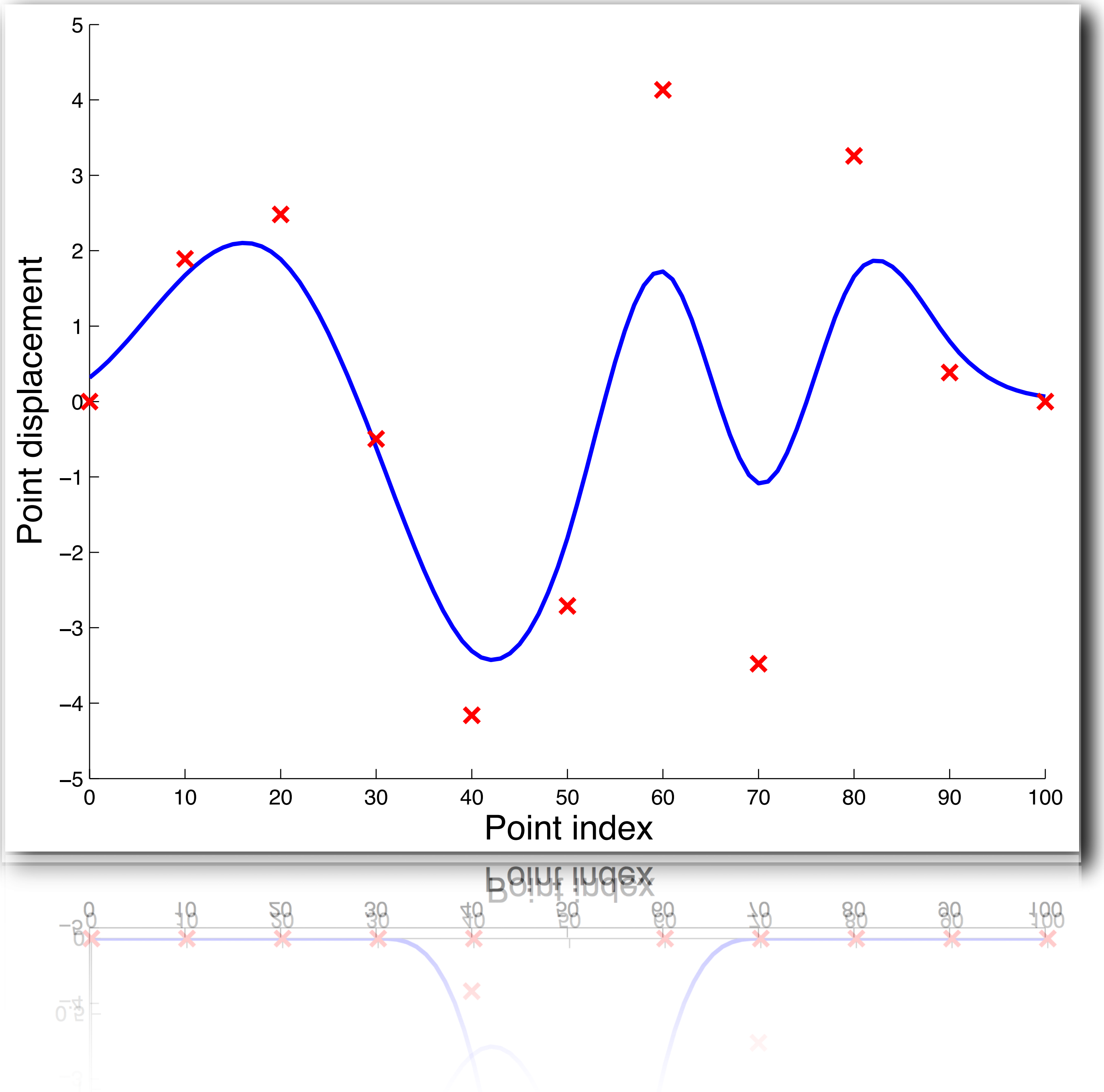
FFD - Deformation Model

- “One” dimensional example



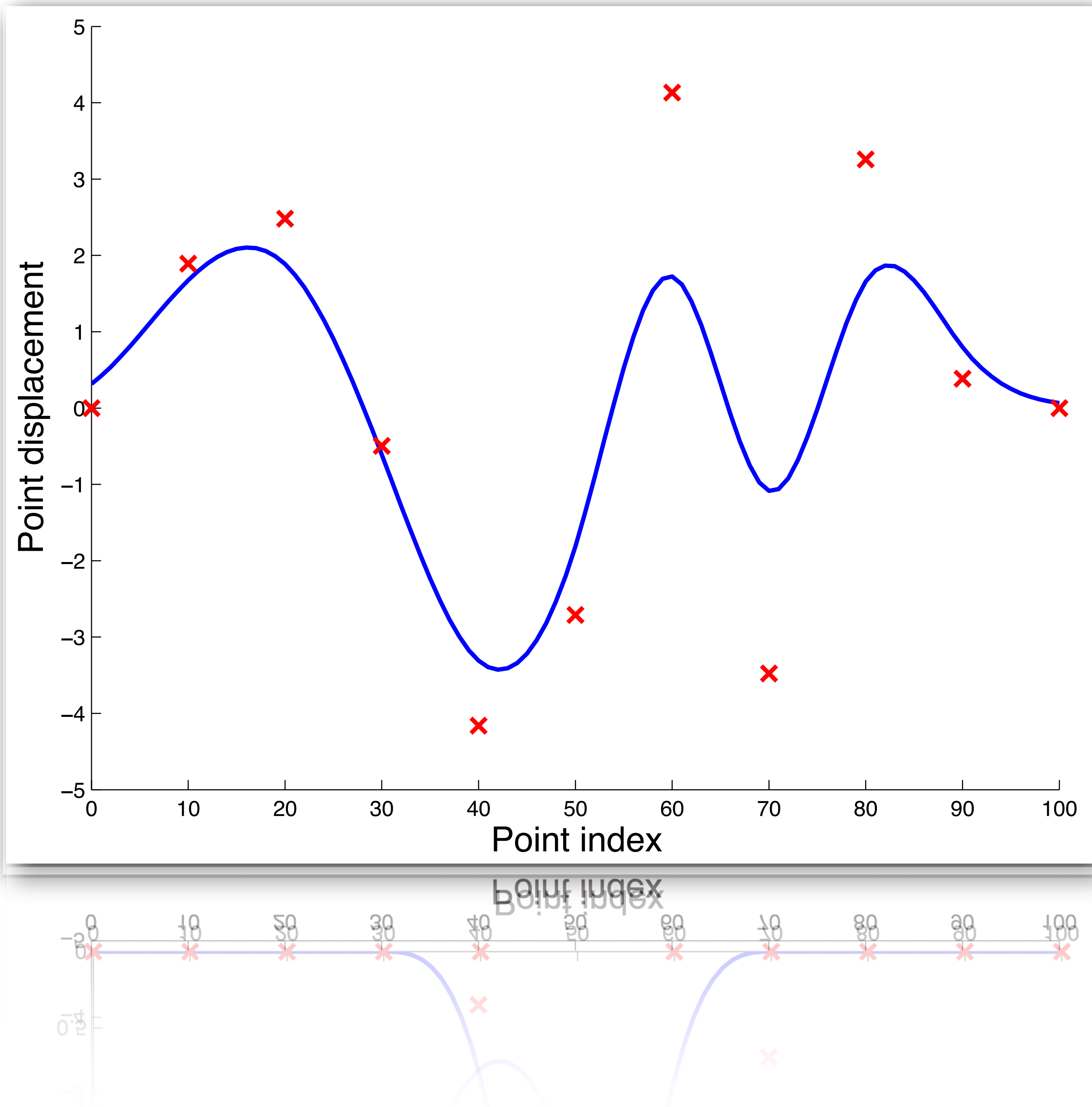
FFD - Deformation Model

- “One” dimensional example



FFD - Deformation Model

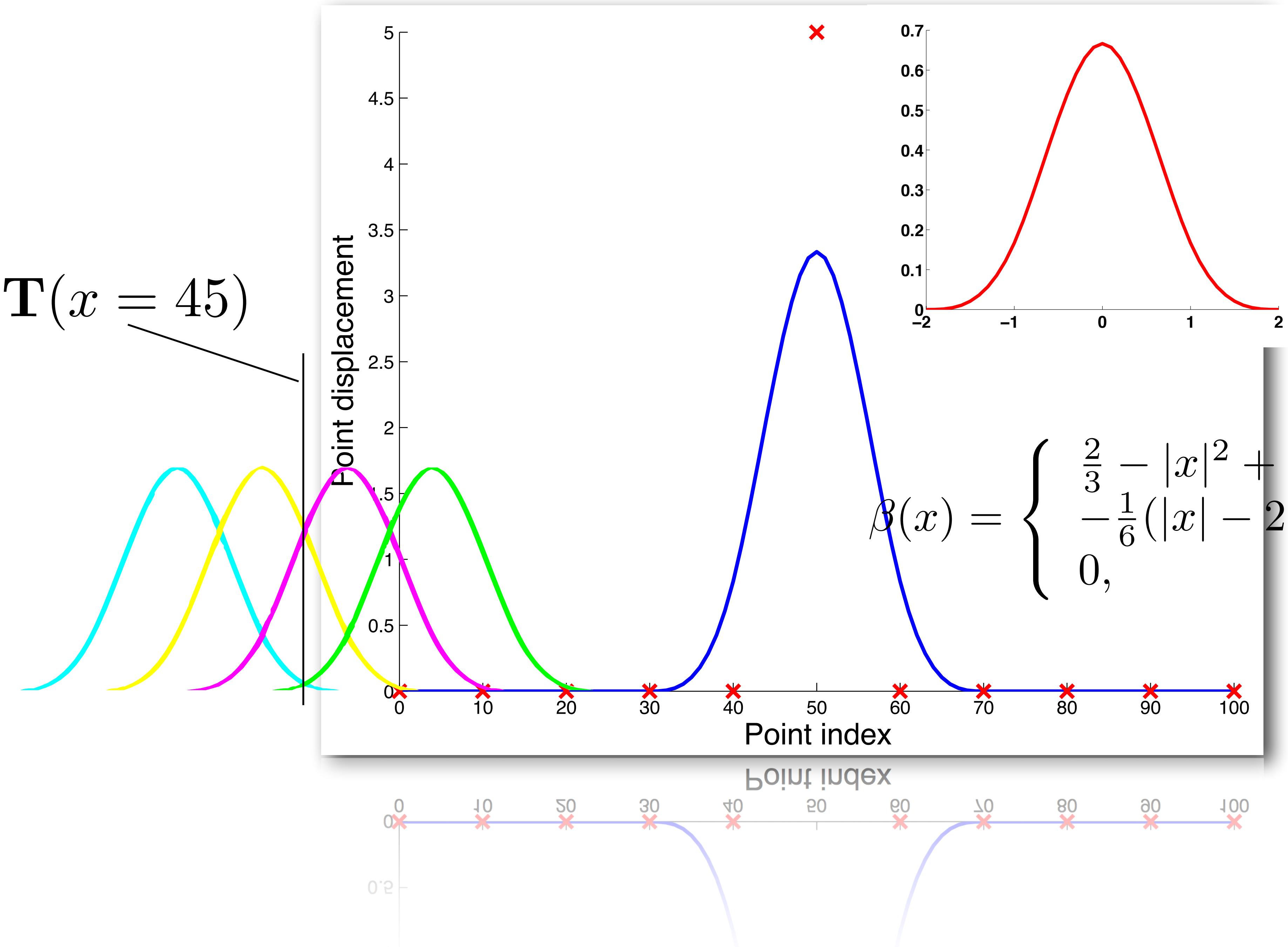
- “One” dimensional example



FFD - Deformation Model

- “One” dimensional example

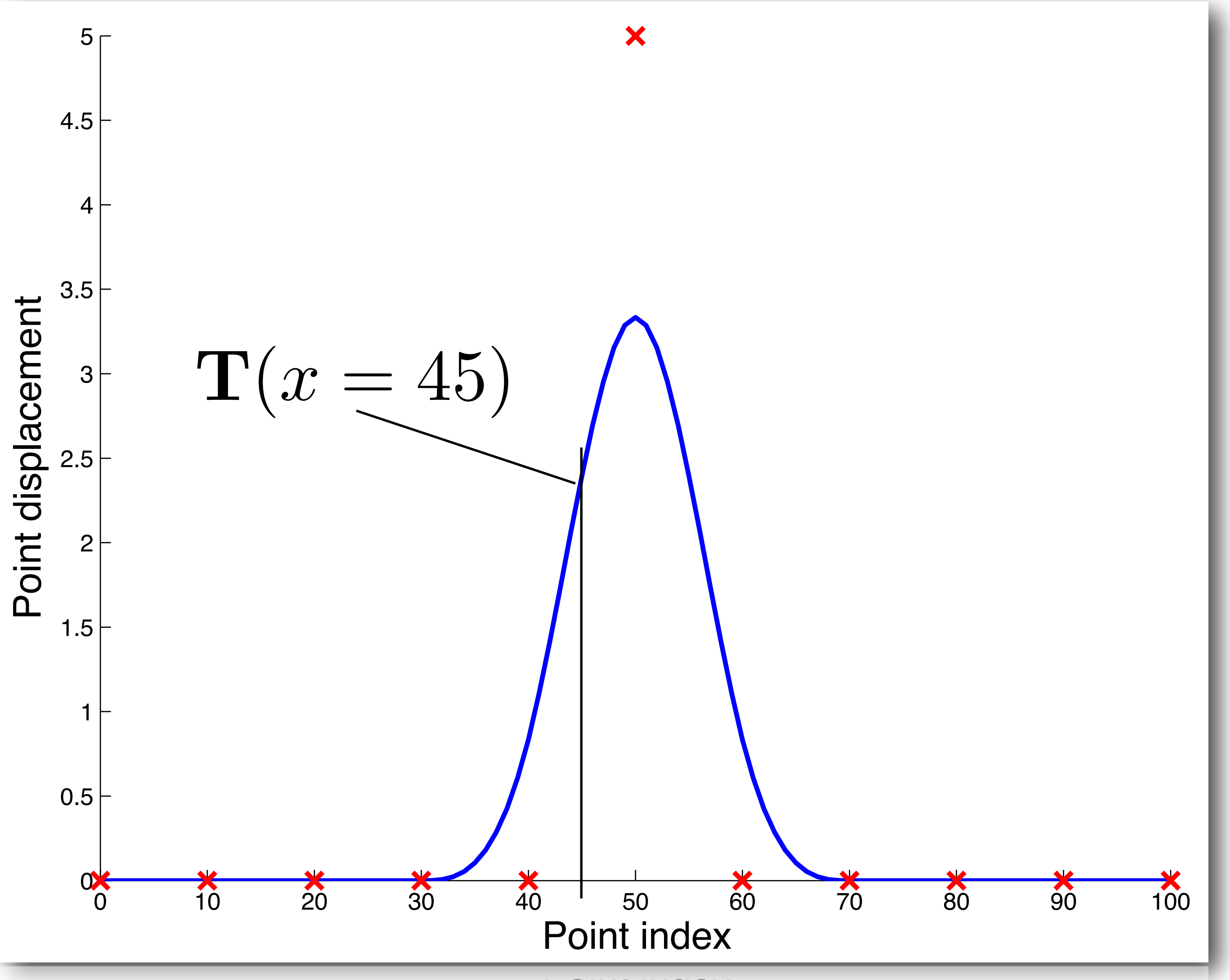
$$\mathbf{T}(x = 45)$$



$$\beta(x) = \begin{cases} \frac{2}{3} - |x|^2 + \frac{1}{2}|x|^3, & |x| < 1 \\ -\frac{1}{6}(|x| - 2)^3, & 1 \leqslant |x| < 2 \\ 0, & |x| \geqslant 2 \end{cases}$$

FFD - Deformation Model

- “One” dimensional example



$$\beta(x) = \begin{cases} \frac{2}{3} - |x|^2 + \frac{1}{2}|x|^3, & |x| < 1 \\ -\frac{1}{6}(|x| - 2)^3, & 1 \leqslant |x| < 2 \\ 0, & |x| \geqslant 2 \end{cases}$$

For each control point i, we have a coordinate c(i) and a position μ(i)

We compute the relative distance d(i) of x to each control point μ:

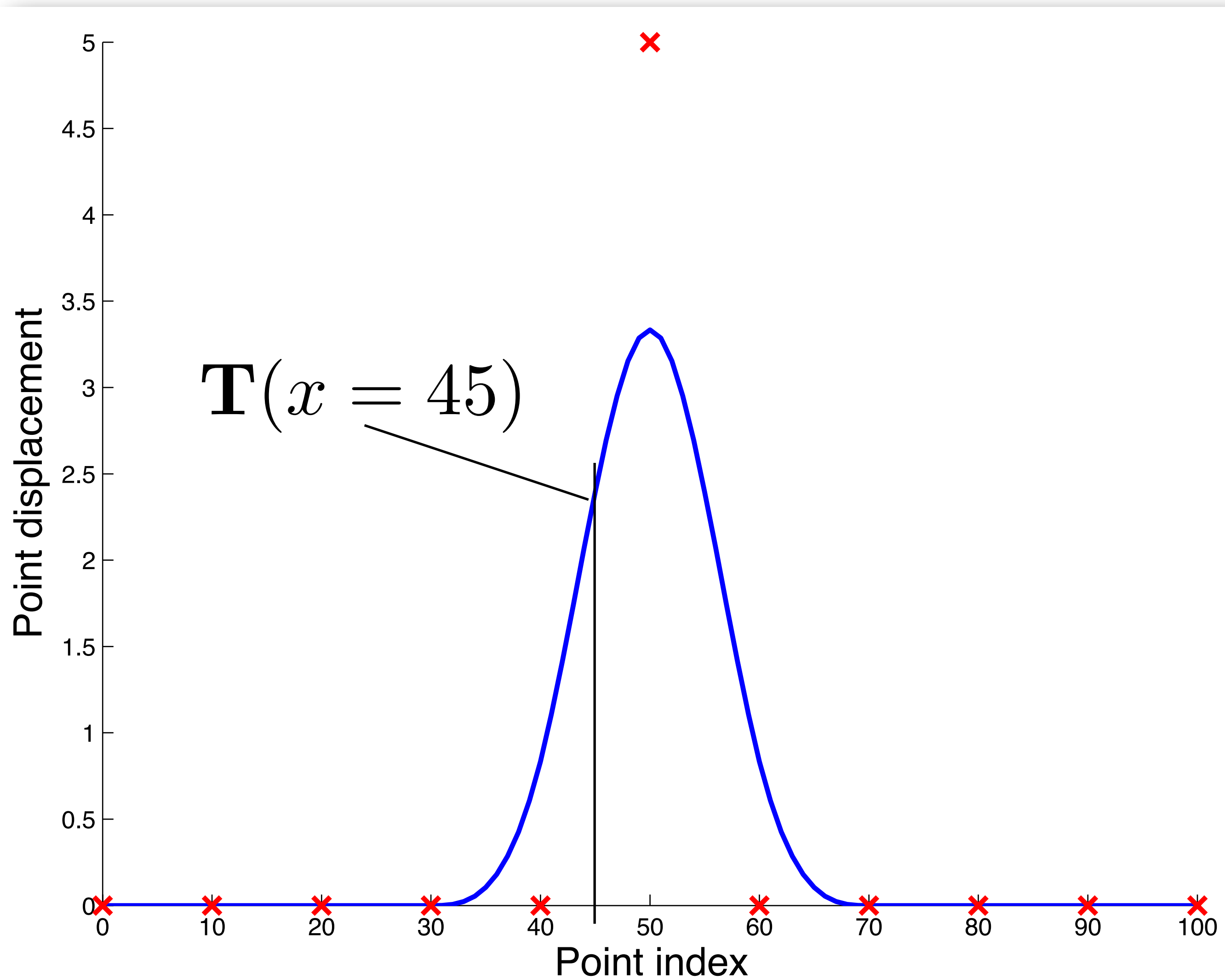
$$d(i) = \frac{(x - c(i))}{\delta}$$

We can then compute the T(x), as:

$$\mathbf{T}(x) = \sum_i \beta\left(d\left(i\right)\right) \times \mu(i)$$

FFD - Deformation Model

- “One” dimensional example



Point index

Control point array

$$\mathbf{T}(x) = \sum_{l=0}^{3} \beta_l(u)\mu_{i+l}$$

$$\text{where} \quad i = \lfloor \frac{x}{\delta} \rfloor - 1$$

$$u = \frac{x}{\delta} - \lfloor \frac{x}{\delta} \rfloor$$

Basis values

Spacing

$x = 45$

$\mu_3 = 0,\ \mu_4 = 0,\ \mu_5 = 5 \text{ and } \mu_6 = 0.$

Basis functions

$$\begin{aligned} B_0(u) &= (1-u)^3/6 \\ B_1(u) &= (3u^3 - 6u^2 + 4)/6 \\ B_2(u) &= (-3u^3 + 3u^2 + 3u + 1)/6 \\ B_3(u) &= u^3/6 \end{aligned}$$

FFD - Deformation Model

- "One" dimensional example

$$u = \frac{45}{10} - \lfloor \frac{45}{10} \rfloor$$

$$\begin{aligned} B_0(0.5) &= 0.0208 \\ B_1(0.5) &= 0.4792 \\ B_2(0.5) &= 0.4792 \\ B_3(0.5) &= 0.0208 \end{aligned}$$

$$i = \lfloor \frac{45}{10} \rfloor - 1$$

$$\begin{aligned} \mathbf{T}(45) &= 0.0208 \times 0 + 0.4792 \times 0 \\ &+ 0.4792 \times 5 + 0.0208 \times 0 \\ &= 2.3960 \end{aligned}$$

$$x = 45$$

$\mu_3 = 0,\ \mu_4 = 0,\ \mu_5 = 5$ and $\mu_6 = 0.$

Point index

Control point array

$$\mathbf{T}(x) = \sum_{l=0}^{3} \beta_l(u)\mu_{i+l}$$

Basis values

where $i = \lfloor \frac{x}{\delta} \rfloor - 1$

Spacing

$$u = \frac{x}{\delta} - \lfloor \frac{x}{\delta} \rfloor$$

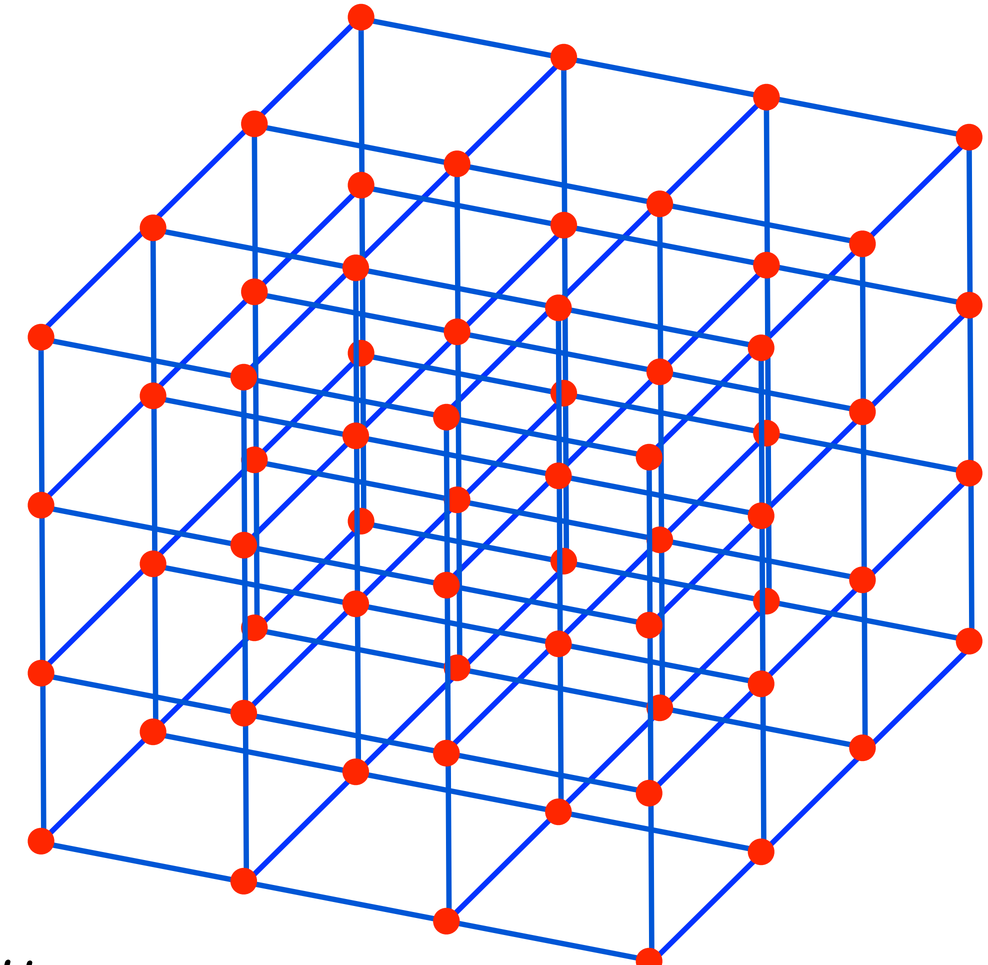
Basis functions

$$\begin{aligned} B_0(u) &= (1-u)^3/6 \\ B_1(u) &= (3u^3 - 6u^2 + 4)/6 \\ B_2(u) &= (-3u^3 + 3u^2 + 3u + 1)/6 \\ B_3(u) &= u^3/6 \end{aligned}$$

FFD - Deformation Model

- From 1 to 3 dimensions

3D: 64 control points per voxel



- $$\mathbf{T}(x) = \sum_{l=0}^{3} \beta_l(u)\mu_{i+l}$$

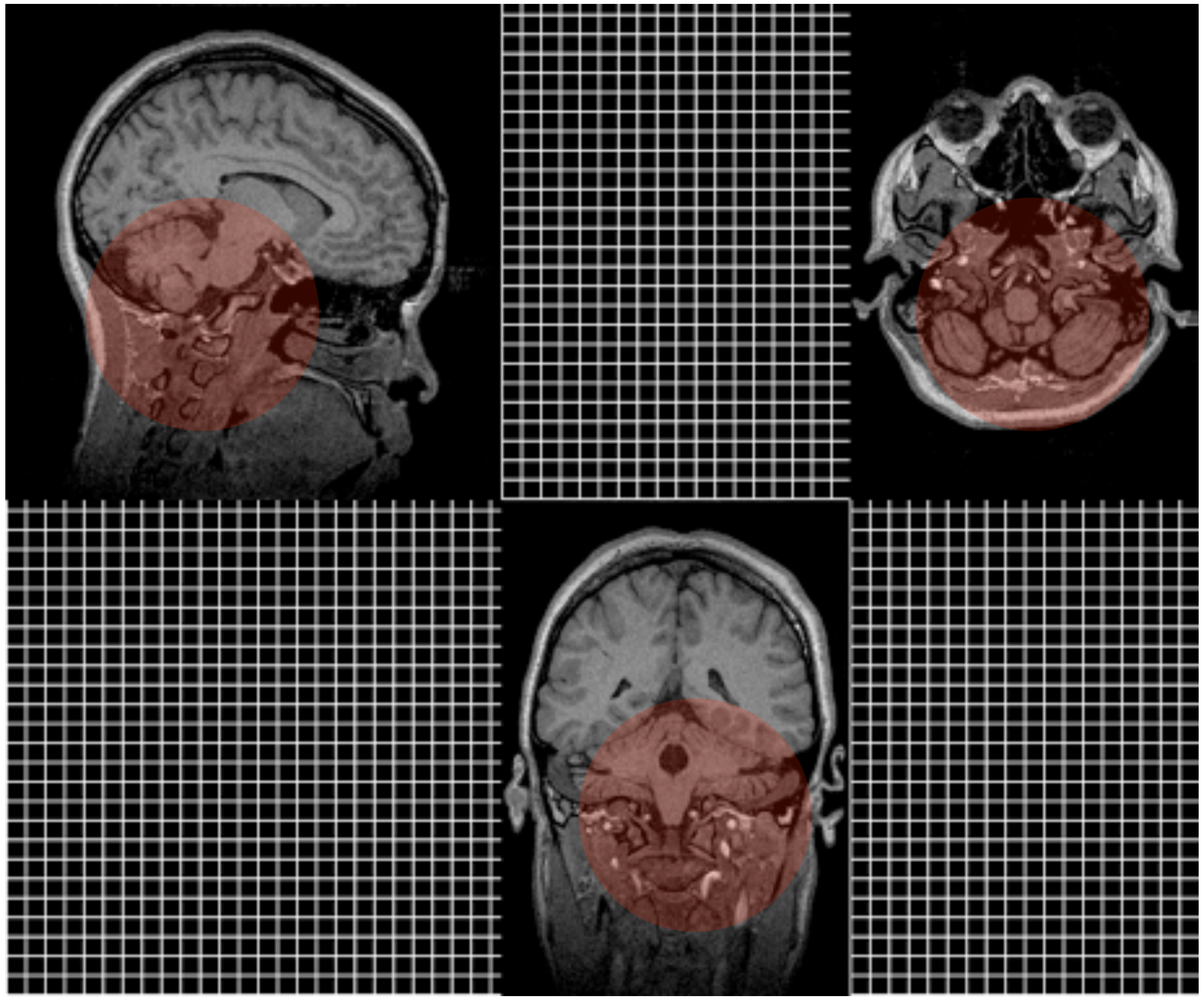
$$\mathbf{T}(x,y) = \sum_{l=0}^{3}\sum_{m=0}^{3} \beta_l(u)\beta_m(v)\mu_{i+l,j+m}$$

$$\mathbf{T}(x,y,z) = \sum_{l=0}^{3}\sum_{m=0}^{3}\sum_{n=0}^{3} \beta_l(u)\beta_m(v)\beta_n(w)\mu_{i+l,j+m,k+n}$$

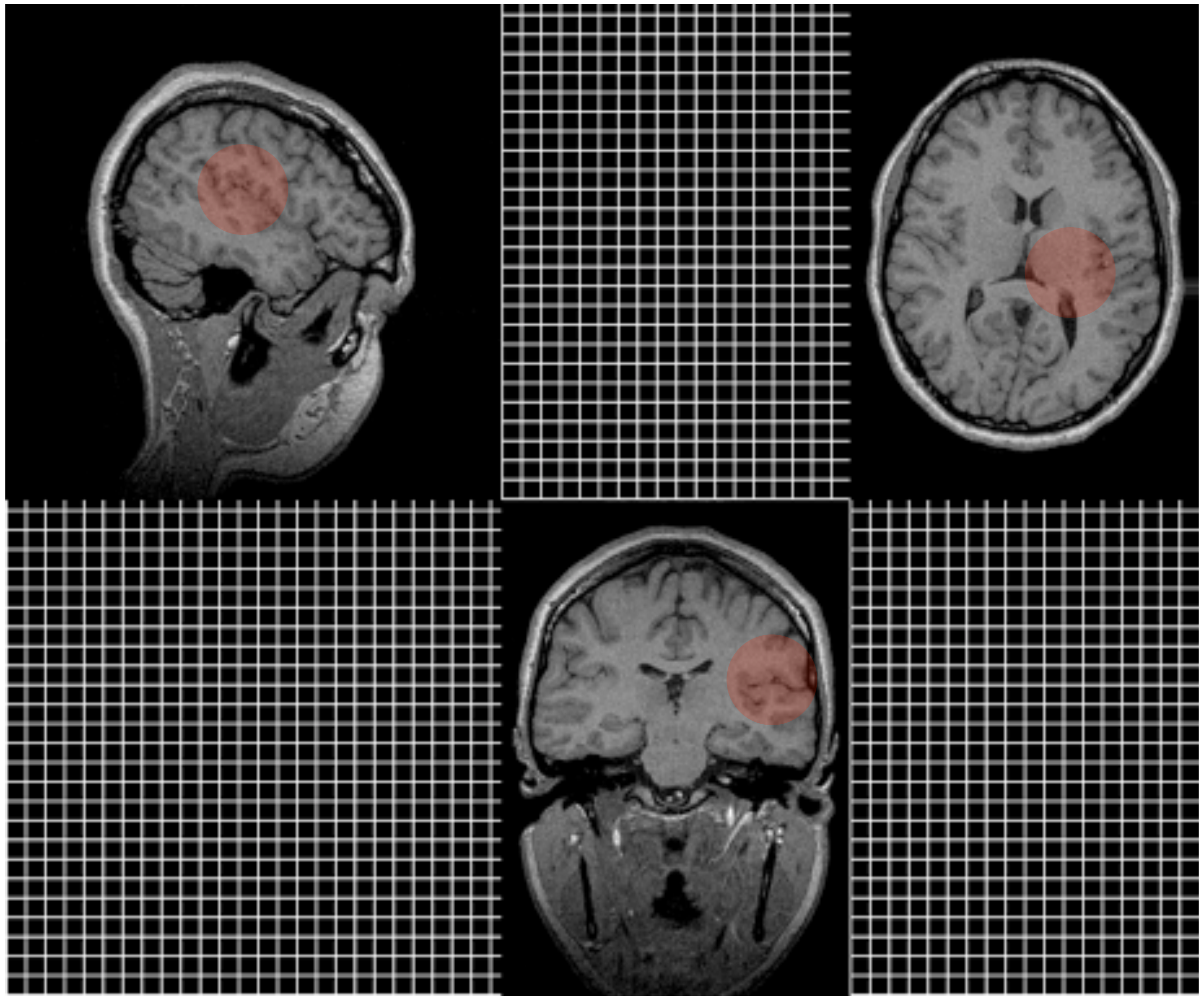
$$u = \frac{x}{\delta_x} - \lfloor\frac{x}{\delta_x}\rfloor \quad w = \frac{z}{\delta_z} - \lfloor\frac{z}{\delta_z}\rfloor \quad v = \frac{y}{\delta_y} - \lfloor\frac{y}{\delta_y}\rfloor \qquad i = \lfloor\frac{x}{\delta_x}\rfloor - 1 \quad j = \lfloor\frac{y}{\delta_y}\rfloor - 1 \quad k = \lfloor\frac{z}{\delta_z}\rfloor - 1$$

FFD - Deformation Model

- Coarse-to-fine approach



20 mm spacing

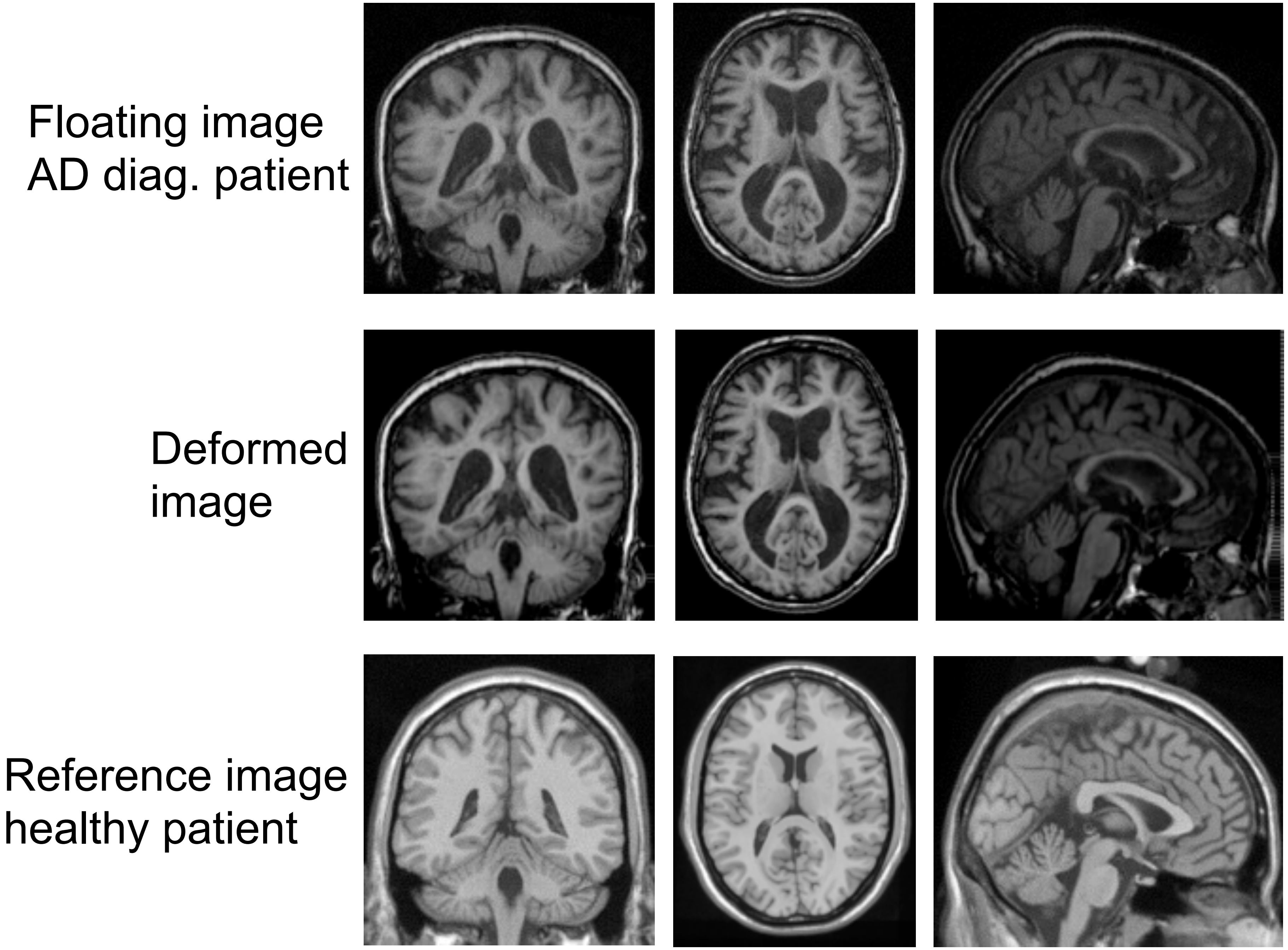


5 mm spacing

Displacement of 1 control point along 1 axis

FFD - Deformation Model

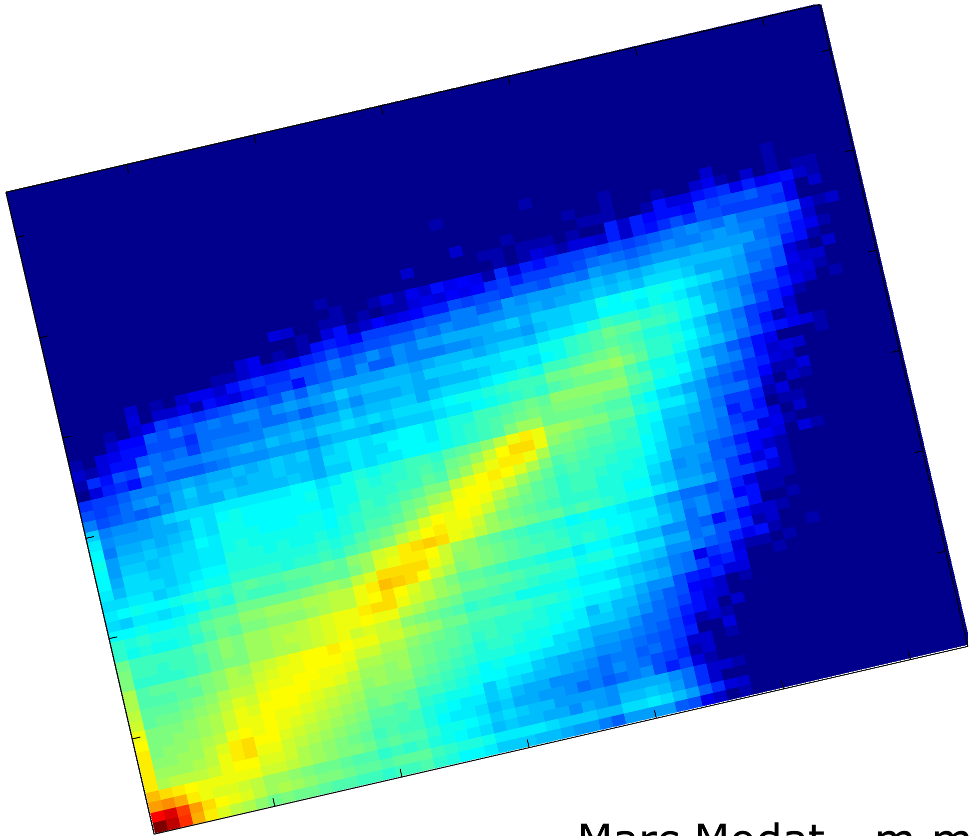
- Application to medical image

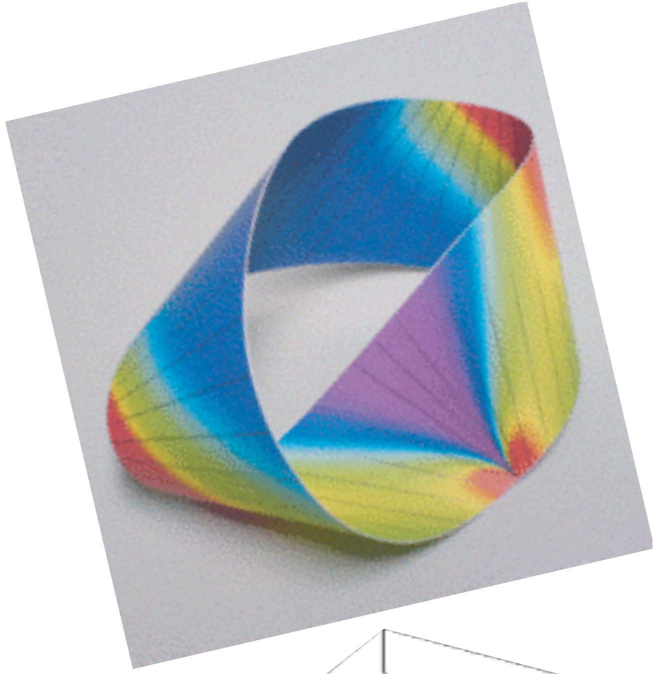


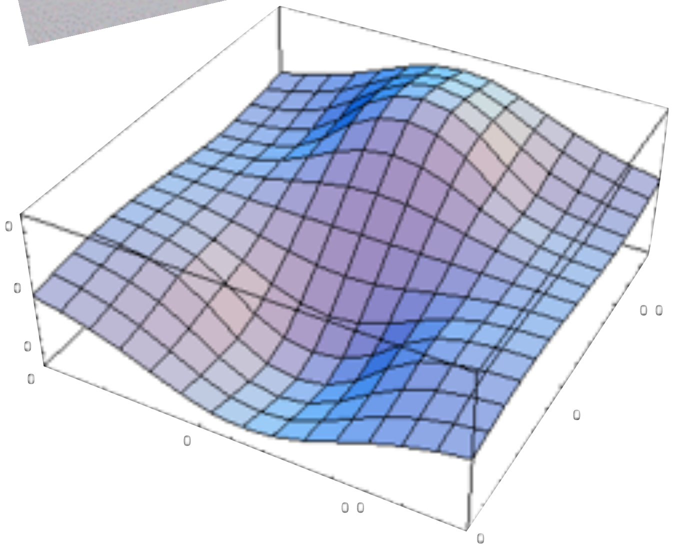
Objective function

- Normalised Mutual Information

- Bending-energy

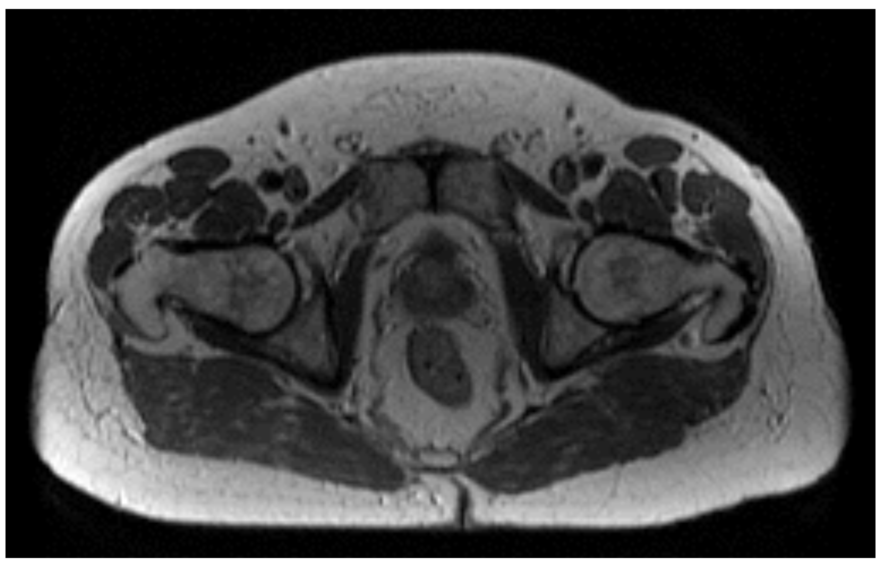




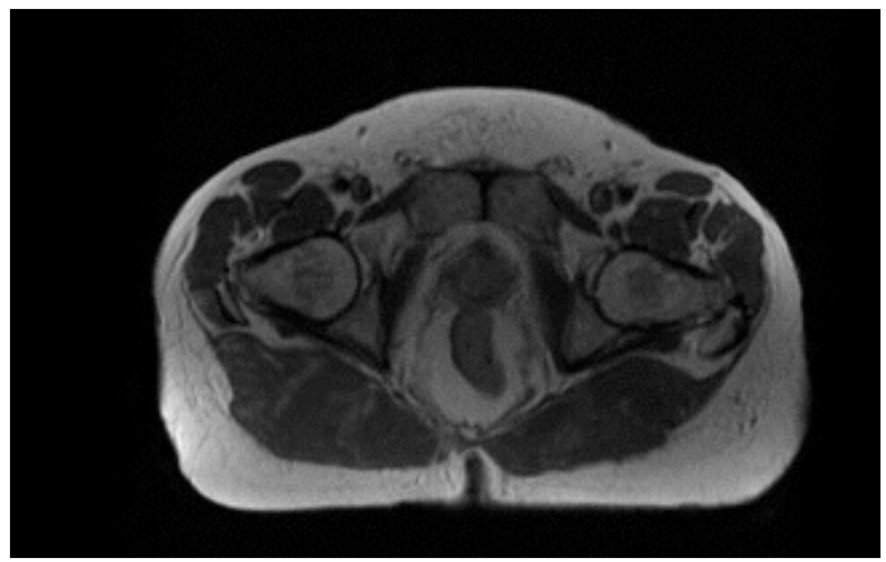


FFD - Objective function - NMI

- Normalised Mutual Information
  - based on theory of information
  - allows multi-model registration



Floating image
MRI



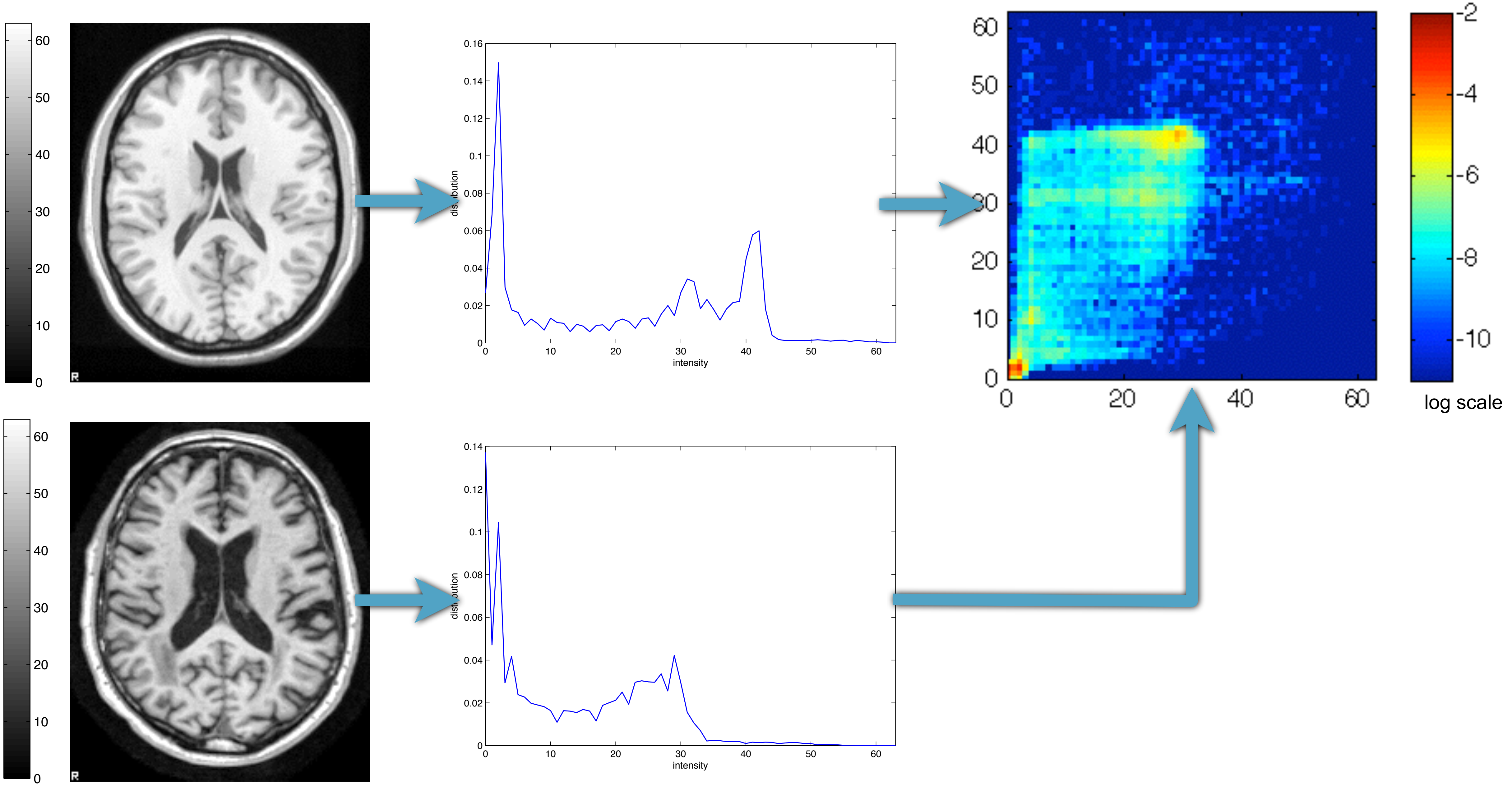
Non-rigid result

Reference image
CT

  - need to build a joint histogram

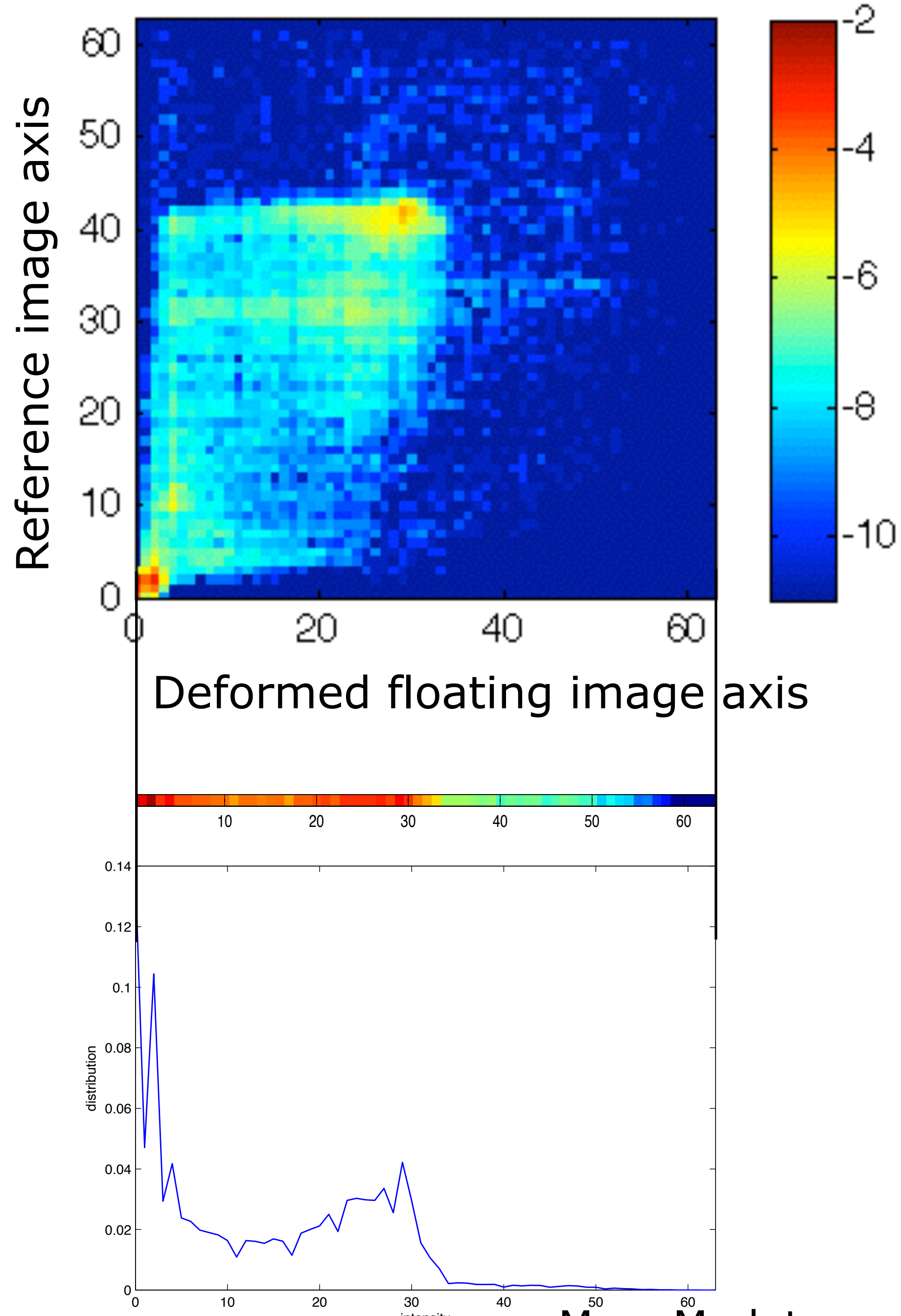
FFD - Objective function - NMI

- Joint histogram
  - intensity paired-distribution of two images



FFD - Objective function - NMI

- Marginal and joint entropies



Joint entropy

$$H(R,F(\mathbf{T})) = -\sum_{r=0}^{bin-1}\sum_{f=0}^{bin-1} p(r,f) \times log(p(r,f))$$

Reference image entropy

$$H(R) = -\sum_{r=0}^{bin-1} p(r) \times log(p(r))$$

Deformed floating image entropy

$$H(F(\mathbf{T})) = -\sum_{f=0}^{bin-1} p(f) \times log(p(f))$$

FFD - Measure - NMI

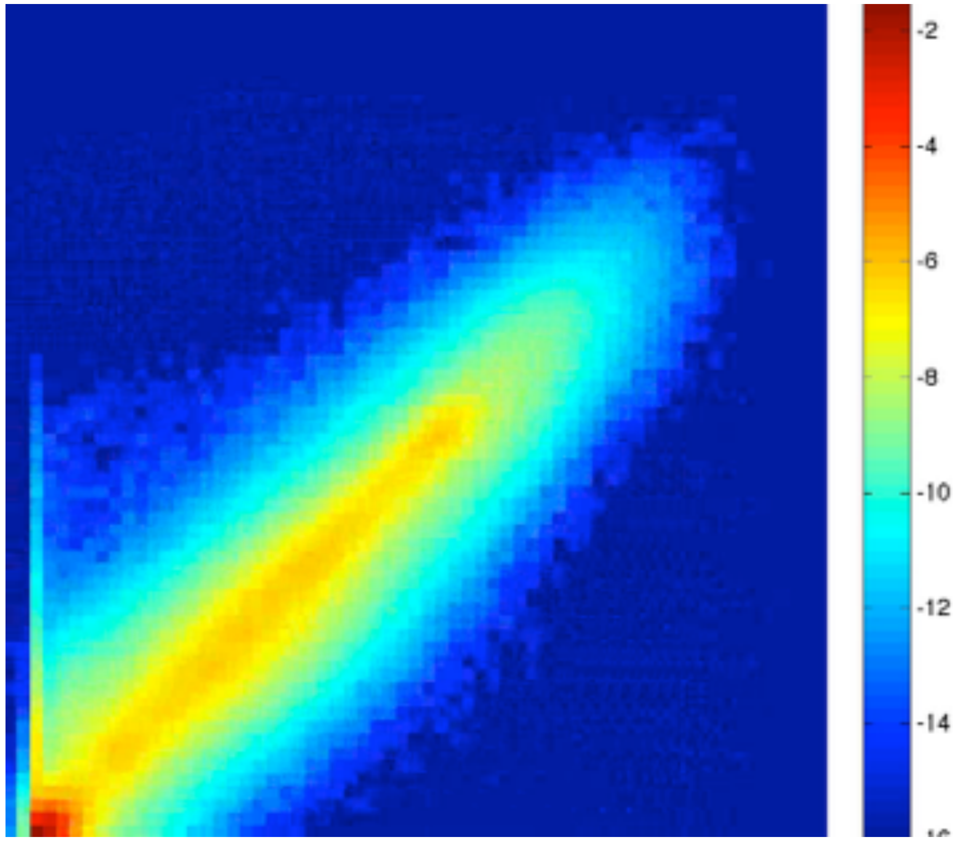
- Normalise Mutual Information principle
  - Maximise the information one image has about the other

$$\mathrm{NMI} = \frac{H(R) + H(F(\mathbf{T}))}{H(R, F(\mathbf{T}))}$$

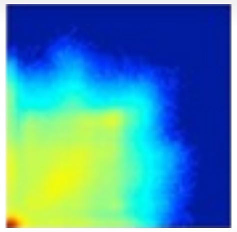
$$H(R) = -\sum_{r=0}^{bin-1} p(r) \times log(p(r))$$

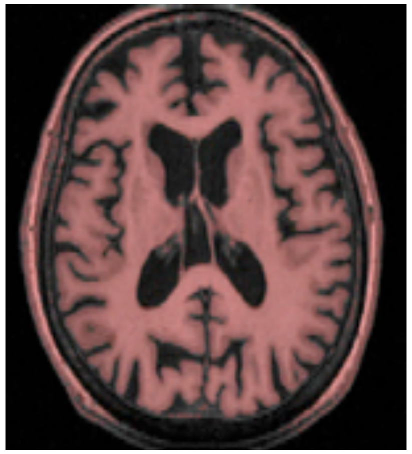
$$H(F(\mathbf{T})) = -\sum_{f=0}^{bin-1} p(f) \times log(p(f))$$

$$H(R, F(\mathbf{T})) = -\sum_{r=0}^{bin-1}\sum_{f=0}^{bin-1} p(r, f) \times log(p(r, f))$$



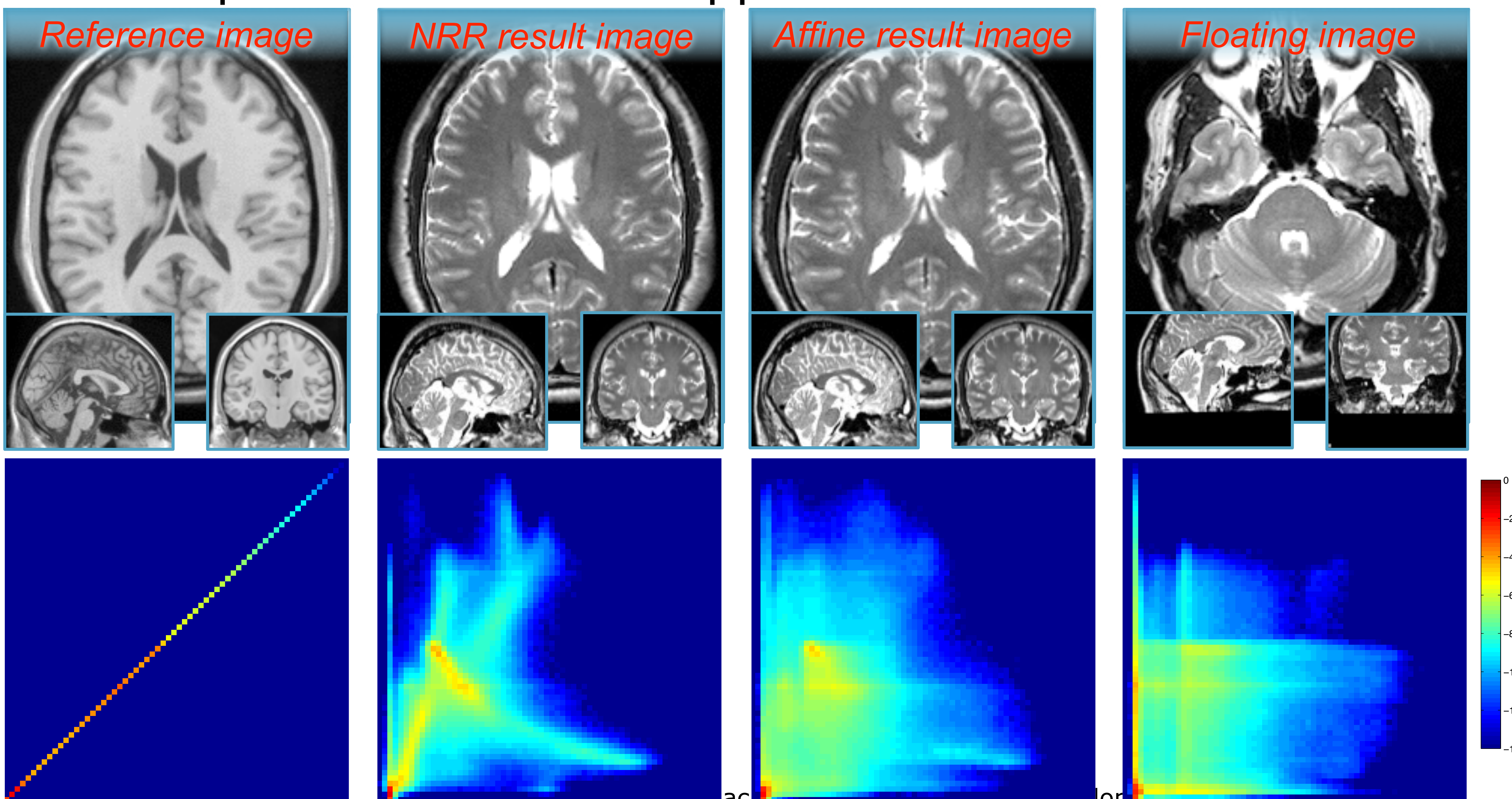
Initial joint histogram





FFD - Measure - NMI

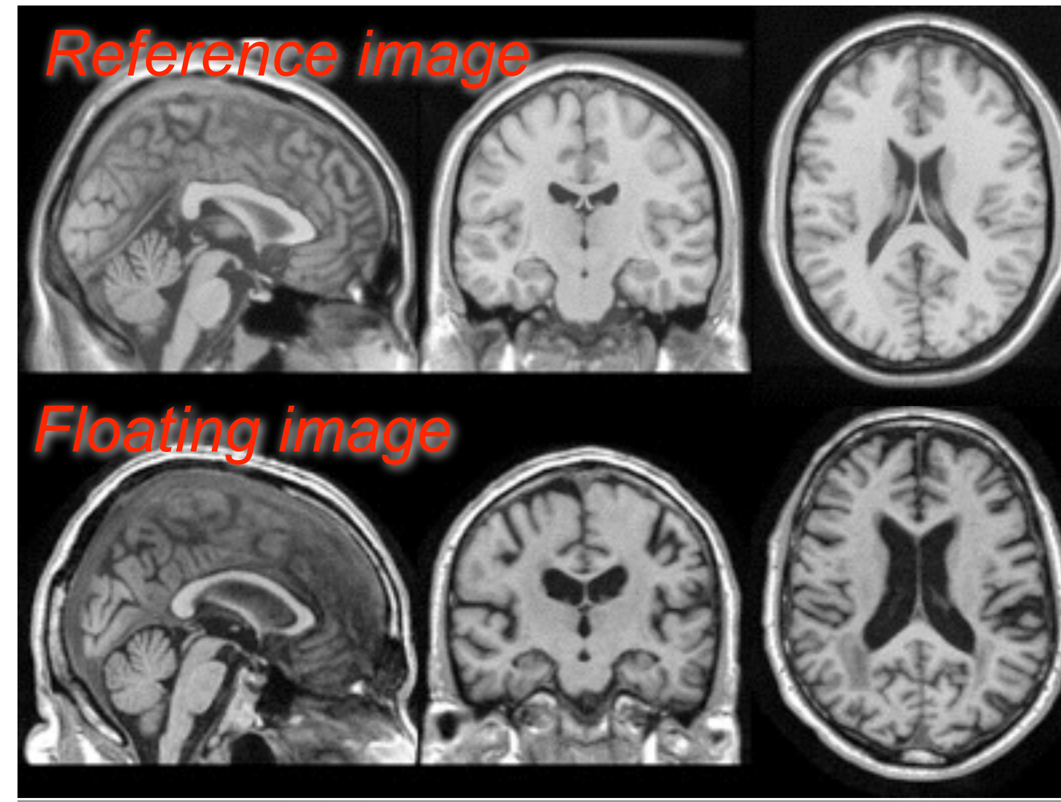
- Example of multi-modal application

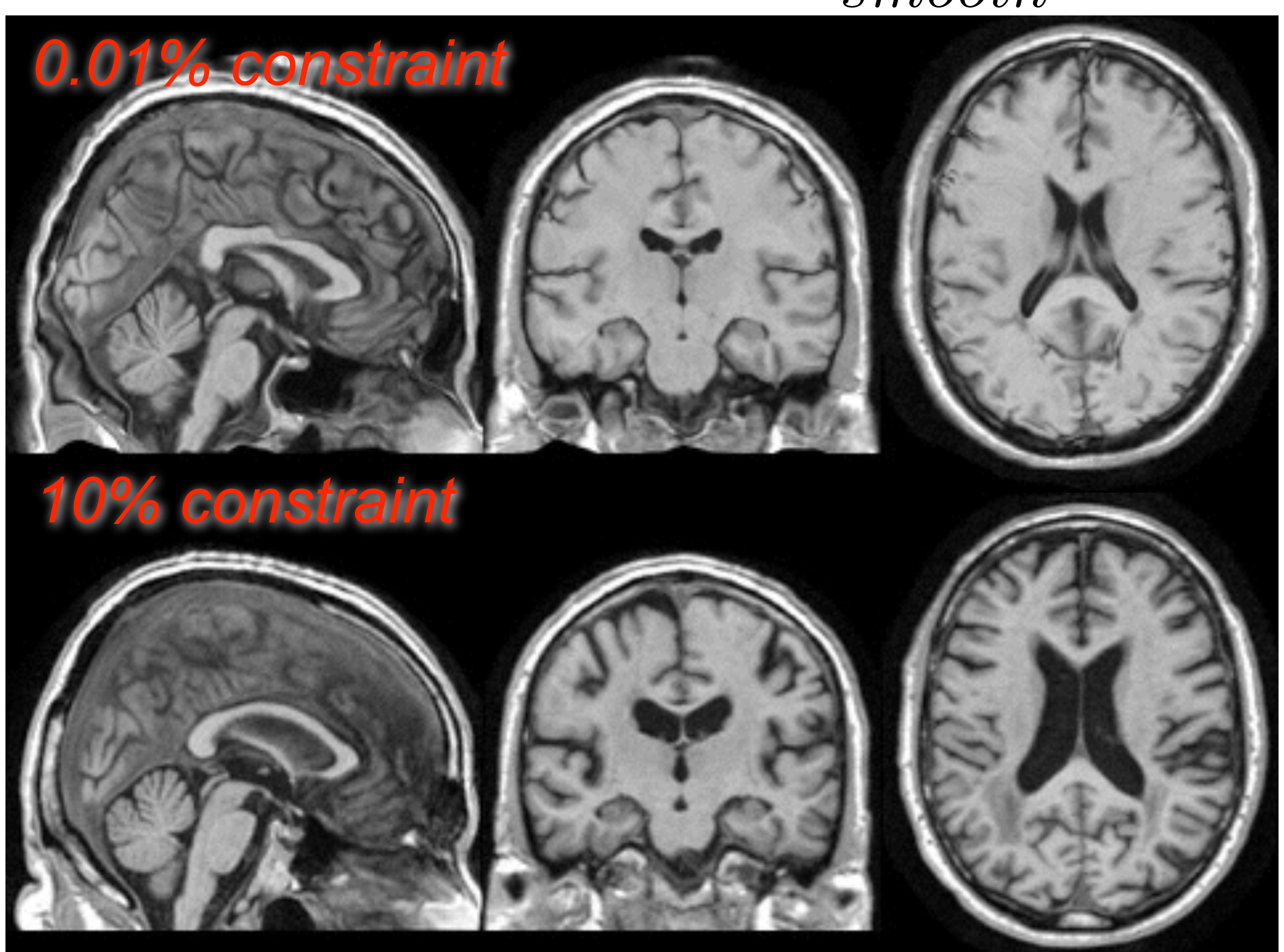


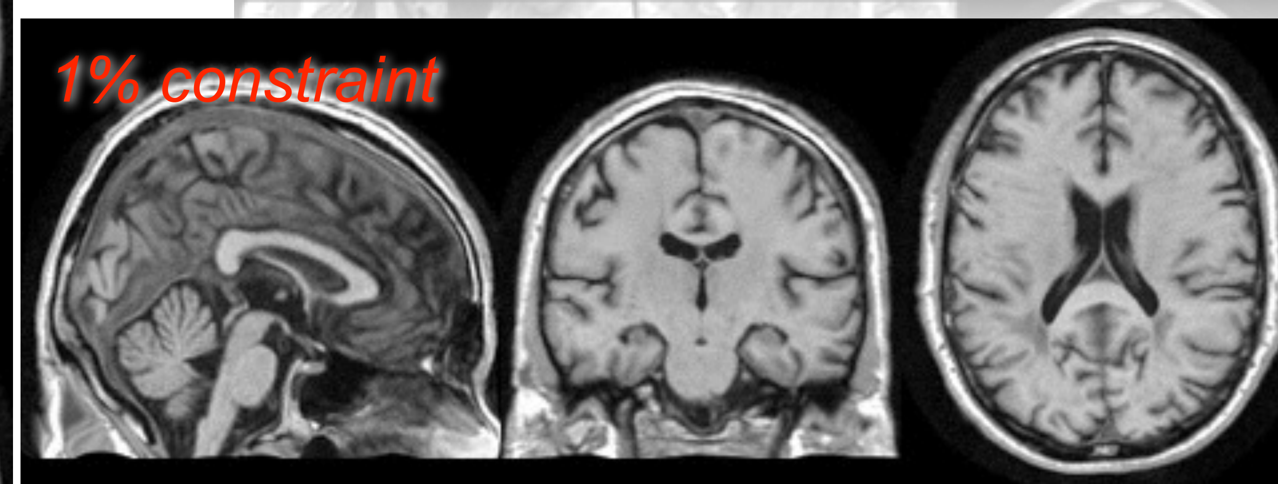
FFD - Regularisation - Bending Energy

- Bending Energy
  - penalize large constraint
  - favors "smooth" deformation

$$\begin{aligned}\text{Obj Fct} &= (1-\alpha)\times \text{NMI}\\ &- \alpha \times C_{smooth}\end{aligned}$$







FFD - Regularisation - Bending Energy

- Bending Energy

$$C_{smooth} = \int_{\mathbf{\Omega}} \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial x^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial y^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial z^2}\right)^2 + 2 \times \left[\left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xy}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial yz}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xz}\right)^2\right] d\vec{x}$$

  – How to compute the second derivative of the trans.?

1D cubic B-Spline reminder

$$\mathbf{T}(x) = \sum_{l=0}^{3} \beta_l(u)\mu_{i+l}$$

Basis functions

$$\begin{aligned} B_0(u) &= (1-u)^3/6 \\ B_1(u) &= (3u^3 - 6u^2 + 4)/6 \\ B_2(u) &= (-3u^3 + 3u^2 + 3u + 1)/6 \\ B_3(u) &= u^3/6 \end{aligned}$$

1D cubic B-Spline first derivative

$$\frac{\partial \mathbf{T}(x)}{\partial x} = \sum_{l=0}^{3} \frac{d\beta_l(u)}{dx}\mu_{i+l}$$

BF 1^{st} derivative

$$\begin{aligned} dB_0(u)/du &= (-u^2 + 2u - 1)/2 \\ dB_1(u)/du &= (3u^2 - 4u)/2 \\ dB_2(u)/du &= (-3u^2 + 2u + 1)/2 \\ dB_3(u)/du &= u^2/2 \end{aligned}$$

FFD - Regularisation - Bending Energy

- Bending Energy

$$C_{smooth} = \int_{\mathbf{\Omega}} \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial x^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial y^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial z^2}\right)^2 + 2 \times \left[\left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xy}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial yz}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xz}\right)^2\right] d\vec{x}$$

  - How to compute the second derivative of the trans.?

1D cubic B-Spline reminder

$$\mathbf{T}(x) = \sum_{l=0}^{3} \beta_l(u)\mu_{i+l}$$

Basis functions

$$\begin{aligned} B_0(u) &= (1-u)^3/6 \\ B_1(u) &= (3u^3 - 6u^2 + 4)/6 \\ B_2(u) &= (-3u^3 + 3u^2 + 3u + 1)/6 \\ B_3(u) &= u^3/6 \end{aligned}$$

1D cubic B-Spline second derivative

$$\frac{\partial^2 \mathbf{T}(x)}{\partial x^2} = \sum_{l=0}^{3} \frac{d^2\beta_l(u)}{dx^2}\mu_{i+l}$$

BF 2^{nd} derivative

$$\begin{aligned} d^2B_0(u)/du^2 &= -u + 1 \\ d^2B_1(u)/du^2 &= 3u - 4 \\ d^2B_2(u)/du^2 &= -3u + 1 \\ d^2B_3(u)/du^2 &= u \end{aligned}$$

FFD - Regularisation - Bending Energy

- Bending Energy

$$C_{smooth} = \int_{\mathbf{\Omega}} \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial x^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial y^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial z^2}\right)^2 + 2 \times \left[\left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xy}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial yz}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xz}\right)^2\right] d\vec{x}$$

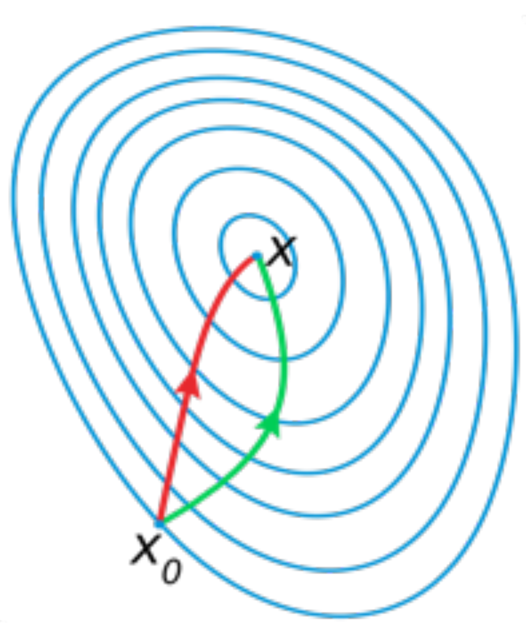
  - How to compute the second derivative of the trans.?

$$\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial x^2} = \sum_{l=0}^{3} \sum_{m=0}^{3} \sum_{n=0}^{3} \frac{d^2 \beta_l(u)}{dx^2} \beta_m(v) \beta_n(w) \mu_{i+l,j+m,k+n}$$

$$\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xy} = \sum_{l=0}^{3} \sum_{m=0}^{3} \sum_{n=0}^{3} \frac{d\beta_l(u)}{dx} \frac{d\beta_m(v)}{dy} \beta_n(w) \mu_{i+l,j+m,k+n}$$

Optimisation

- (Steepest / conjugate / …)Gradient Descent



Analytical derivative of the NMI

$$\mathrm{NMI} = \frac{H(R) + H(F(\mathbf{T}))}{H(R, F(\mathbf{T}))}$$

$$\frac{\partial \mathrm{NMI}}{\partial \mu^{\xi}_{ijk}} = \frac{\frac{\partial H(R)}{\partial \mu^{\xi}_{ijk}} + \frac{\partial H(F(\mathbf{T}))}{\partial \mu^{\xi}_{ijk}} - \mathrm{NMI} \times \frac{\partial H(R,F(\mathbf{T}))}{\partial \mu^{\xi}_{ijk}}}{H(R, F(\mathbf{T}))}$$

$$H(R) = -\sum_{r=0}^{bin-1} p(r) \times log(p(r))$$

$$H(F(\mathbf{T})) = -\sum_{f=0}^{bin-1} p(f) \times log(p(f))$$

$$H(R, F(\mathbf{T})) = -\sum_{r=0}^{bin-1} \sum_{f=0}^{bin-1} p(r, f) \times log(p(r, f))$$

$$\frac{\partial H(R)}{\partial \mu^{\xi}_{ijk}} = -\sum_{r=0}^{bin-1} \frac{\partial p(r)}{\partial \mu^{\xi}_{ijk}} \times \log(p(r) + 1)$$

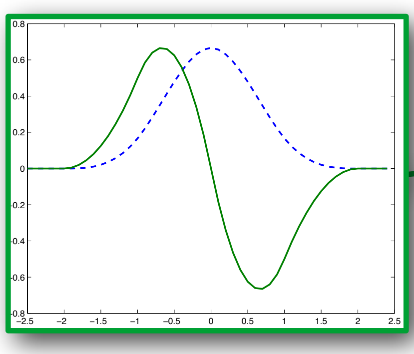
$$\frac{\partial H(F(\mathbf{T}))}{\partial \mu^{\xi}_{ijk}} = -\sum_{f=0}^{bin-1} \frac{\partial p(f)}{\partial \mu^{\xi}_{ijk}} \times \log(p(f) + 1)$$

$$\frac{\partial H(R, F(\mathbf{T}))}{\partial \mu^{\xi}_{ijk}} = -\sum_{r=0}^{bin-1} \sum_{f=0}^{bin-1} \frac{\partial p(r, f)}{\partial \mu^{\xi}_{ijk}} \times \log(p(r, f) + 1)$$

Analytical derivative of the NMI

- Derivative of the distribution (joint histogram)

$$\frac{\partial \mathbf{H}(r,f)}{\partial \mu^{\xi}_{ijk}} = \sum_{\vec{x}\in\Omega} \beta^3_r\left(R(\vec{x});r\right) \left.\frac{\partial \beta^3_f(i,f)}{\partial i}\right|_{i=F(\mathbf{T}(\vec{x}))} \left.\frac{\partial F(p)}{\partial p}\right|_{p=\mathbf{T}(\vec{x})} \frac{\partial \mathbf{T}(\vec{x})}{\partial \mu^{\xi}_{ijk}}$$



$$\frac{\partial T(\vec{x})}{\partial \mu^{\xi}_{ijk}} = B_l(u)\, B_m(v)\, B_n(w)$$

Analytical derivative of the BE

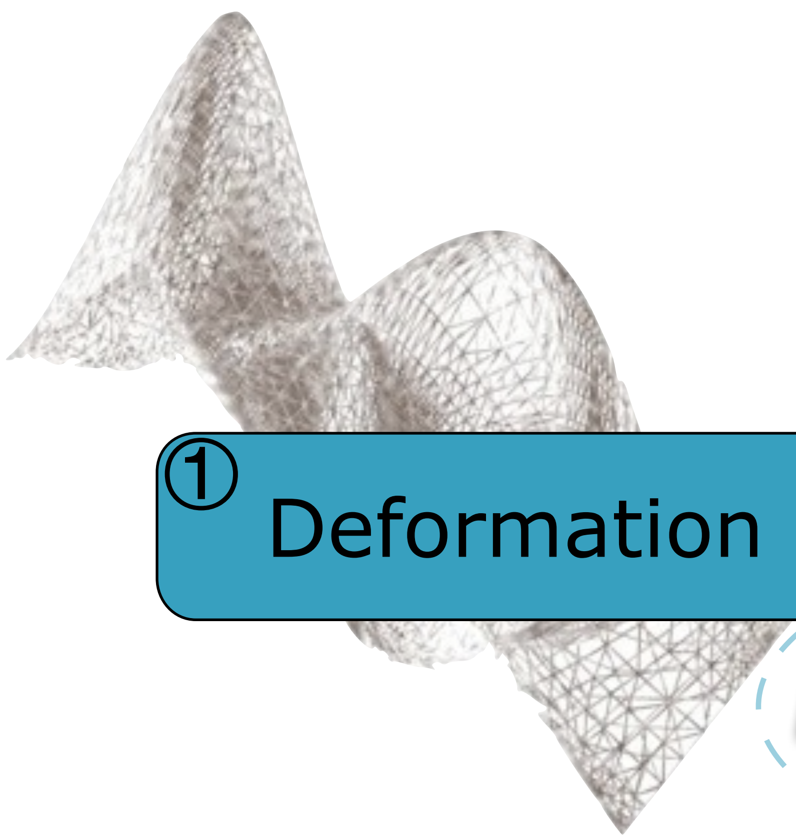
$$\text{Obj Fct} = (1-\alpha) \times \text{NMI} - \alpha \times C_{smooth}$$

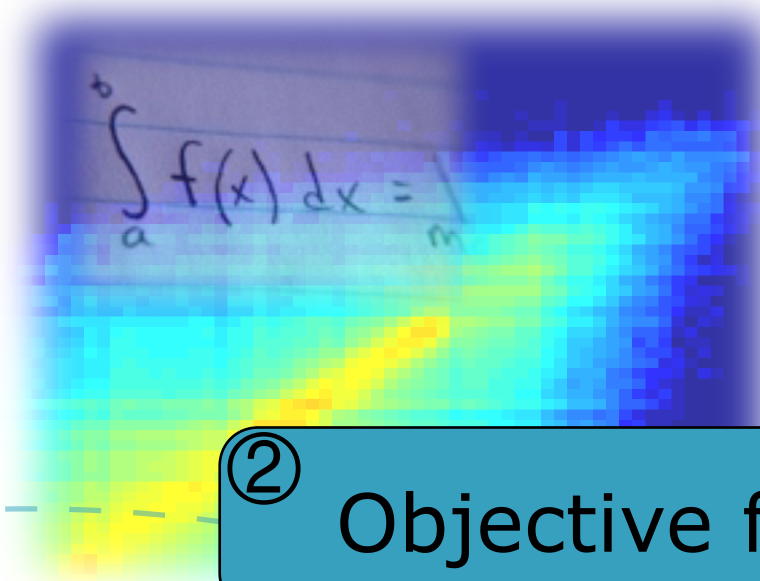
$$\frac{\partial \text{Obj Fct}}{\partial \mu_{ijk}^{\xi}} = (1-\alpha) \times \frac{\partial \text{NMI}}{\partial \mu_{ijk}^{\xi}} - \alpha \times \frac{\partial C_{smooth}}{\partial \mu_{ijk}^{\xi}}$$

$$C_{smooth} = \int_{\mathbf{\Omega}} \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial x^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial y^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial z^2}\right)^2 + 2 \times \left[\left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xy}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial yz}\right)^2 + \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial xz}\right)^2\right] d\vec{x}$$

$$\frac{\partial \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial x^2}\right)^2}{\partial \mu_{ijk}^{\xi}} = \frac{2 \times \frac{\partial^2 \mathbf{T}(\vec{x})}{\partial x^2} . \partial \left(\frac{\partial^2 \mathbf{T}(\vec{x})}{\partial x^2}\right)}{\partial \mu_{ijk}^{\xi}}$$

The original framework - Recap.





① Deformation model

② Objective function

Free Form Deformation

③ Optimisation

① Cubic B-splines transformation model

② Normalised Mutual Information
Bending Energy

③ Gradient descent