Medical Image Analysis and Processing

Medical Image Registration

Introduction

Emad Fatemizadeh

Distance/online Course: Session 25

Date: 25 May 2021, 4th Khordad 1400

Contents

› Mathematical Definition

› Categorization

› Registration Implementation

› Rigid body transform

› Procrustean transform

› Affine transform

› Projective transform

› Bilinear transform

Mathematical Definition

› An optimization task:

› F: Fixed image

› M: Moving image

› Search for a optimal spatial transform T:

$$T^* = arg\max_T similarity(F, M(T))$$

or

$$T^* = arg\max_T similarity(F, T(M))$$

Mathematical Definition

- Spatial Transform:

$$T = (T_x, T_y): \mathbb{R}^2 \longrightarrow \mathbb{R}^2$$

- Goal of optimization:

$$T^* = arg \max_T similarity\left(F(x,y), M\left(T_x(x,y), T_y(x,y)\right)\right)$$

- Example:

$$T^* = arg \min_T \sum_{x,y} \left\| F(x,y) - M\left(T_x(x,y), T_y(x,y)\right) \right\|^2$$

Classification of Registration Methods

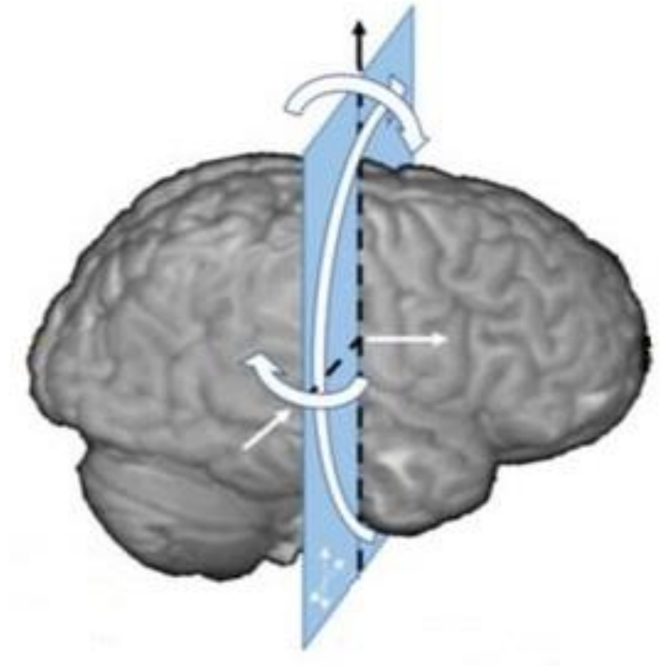
- Dimensionality
- Nature of Registration basis
- Nature of transformation
- Domain of transformation
- Interaction
- Optimization procedure
- Modalities involved
- Subject
- Object

Dimensionality

- Spatial dimension:
  - 2D-2D
  - 3D-3D
  - 2D-3D
- Temporal with spatial dimension:
  - 2D-2D
  - 3D-3D
  - 2D-3D
- 2D-3D: The position of one or more 2D slices are to be established relative to a 3D volume.

Dimensionality

› 2D-3D registration



Nature of Registration Basis

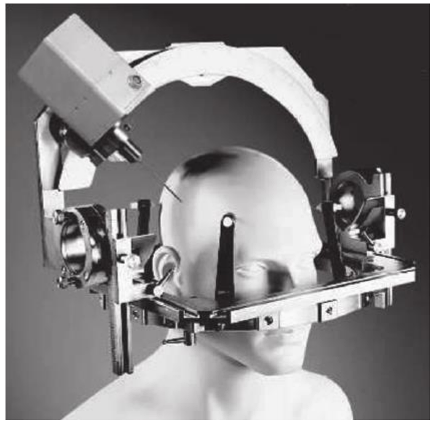
- Extrinsic:

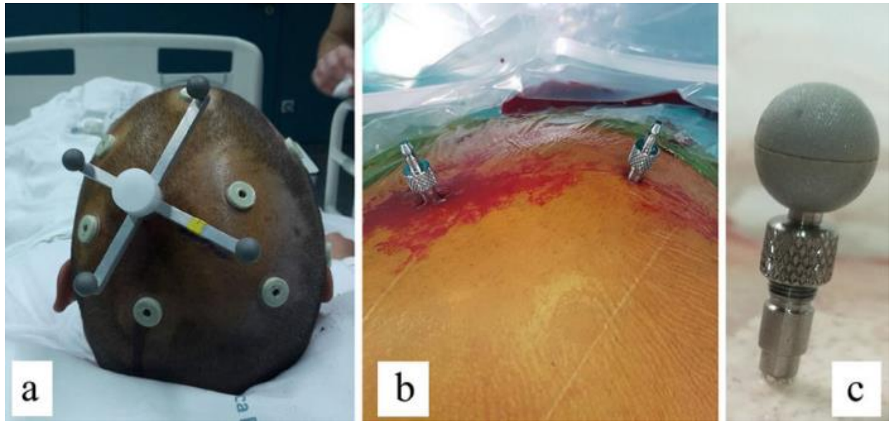
  “Based on foreign objects introduced into the imaged space”
- Intrinsic:

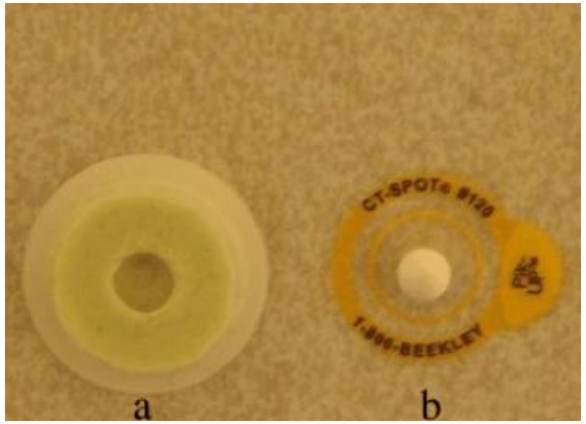
  “Based on the image information as generated by the patient”

Nature of Registration Basis - Extrinsic

- Stereo tactic frame screwed rigidly to the patient's outer skull table
- Screw-mounted markers
- Markers glued to the skin





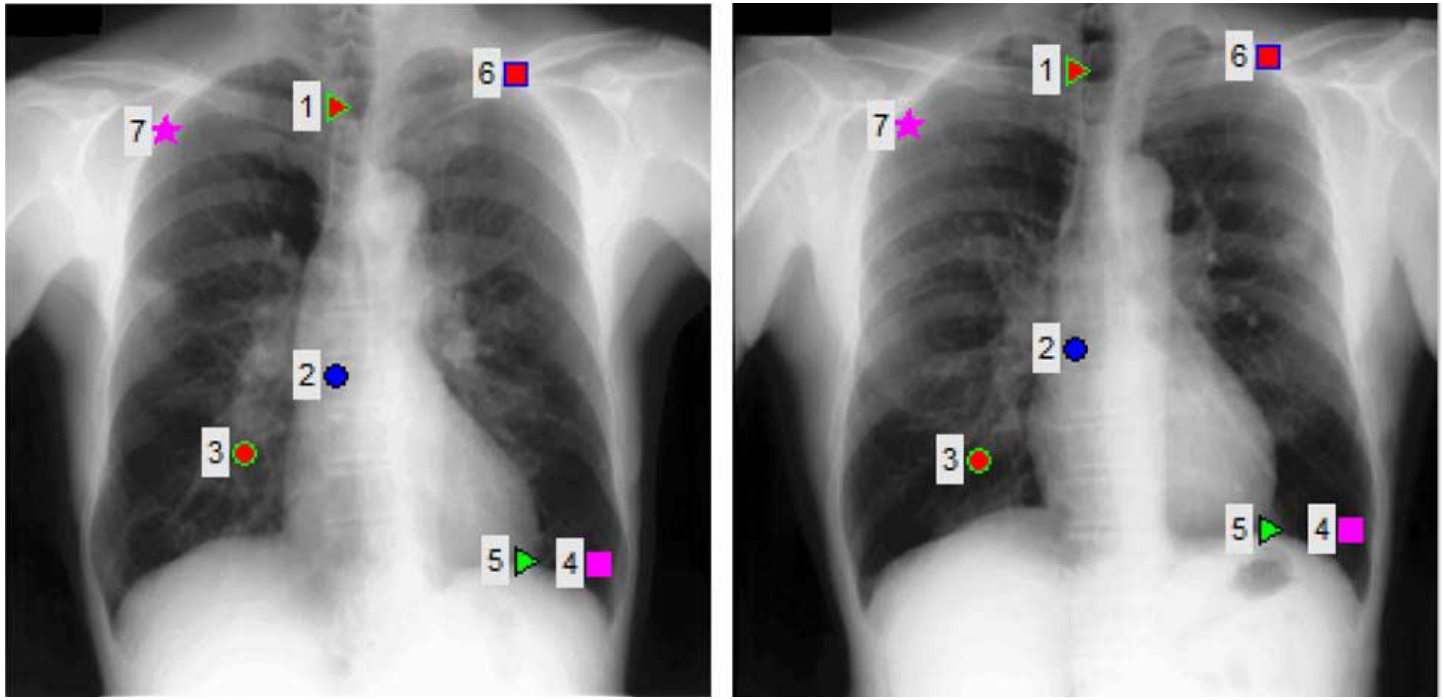


Nature of Registration Basis - Intrinsic

- Landmark based:
  - Match corresponding geometric or anatomical landmarks
- Segmentation based:
  - Surfaces extracted from both the fixed and float image are used as input for the registration process
- Voxel property based:
  - A measure (mathematical or statistical) of intensity similarity is used for registration
- Hybrid methods

Nature of Registration Basis - Intrinsic

› Landmark based:



Nature of Transformation

- Rigid (Rotation and Translation)
- Affine (arbitrary linear transform and translation)
- (Non-rigid) Projective
- (Non-rigid) Curved

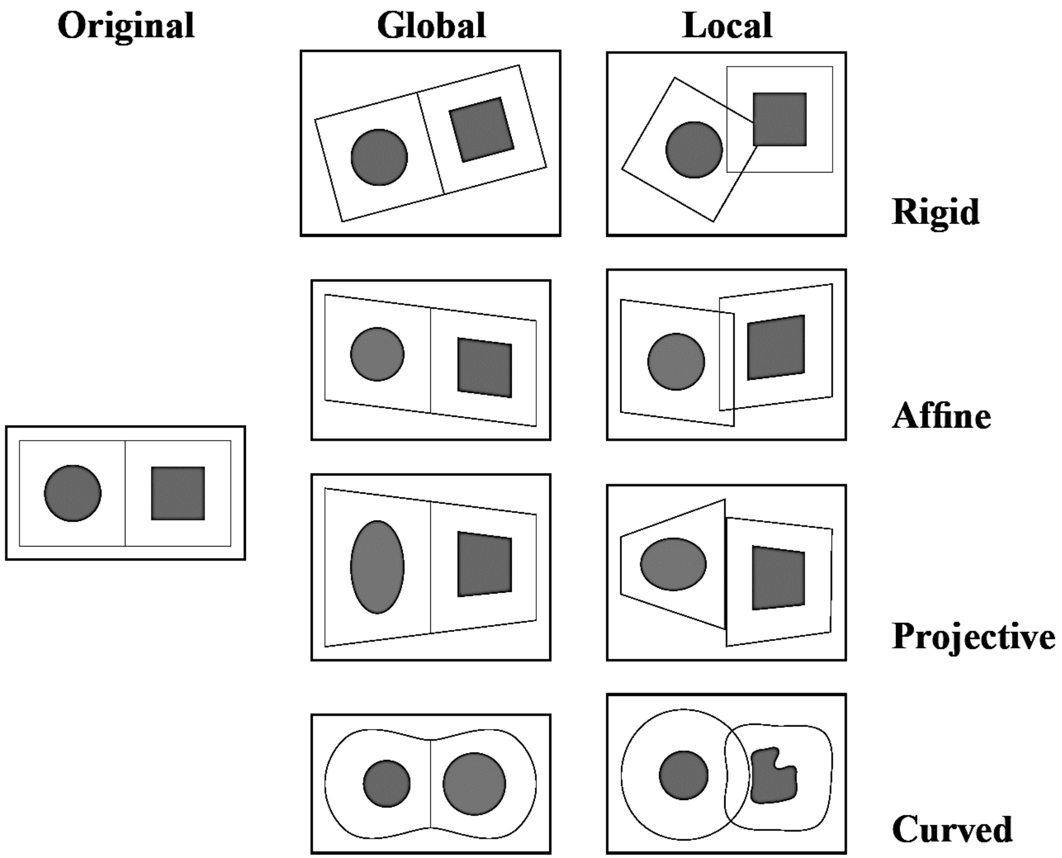
Domain of Transformation

› Global

› Local

Nature and Domain of Transform

› Illustration:

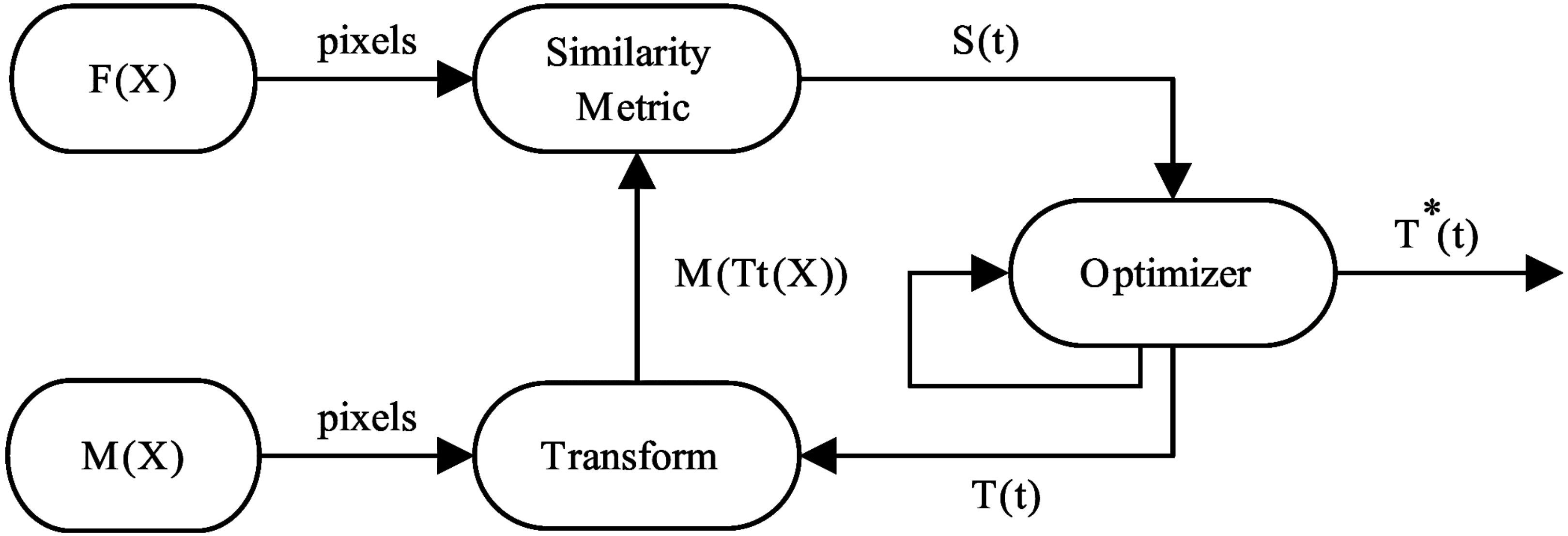


User Interaction

- Extrinsic methods:
  - Automated
  - Semi-automatic
- Intrinsic methods:
  - Semi-automatic
  - Anatomical landmark
  - Segmentation based
- Automated:
  - Geometrical landmark
  - Voxel property based

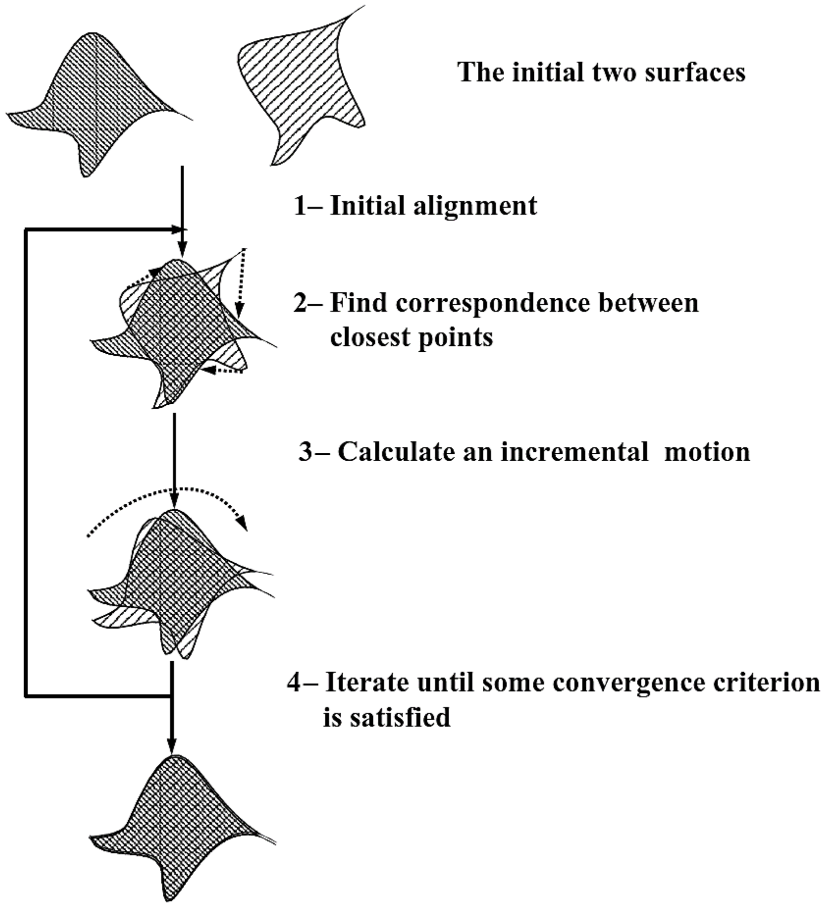
Optimization Procedure

- Optimize the cost function:
  - Parameters are computed Analytically,
  - Search to find best solution (may be local optimum)



Optimization Procedure

› Search to find best solution:



Modality involved

- Monomodal:
  - One patient, One modality (MRI-MRI, CT-CT, …), temporal registration
- Multimodal:
  - One patient, Several modalities (CT-MRI, CR-PET, US-MRI, …)
- Modality to model:
  - One modality-One model (mathematical models or atlas)
- Patient to modality:
  - Patients positioning in intra-operative task (pre and post images registration)

Subjects

- Intra-subject (data from one patient)
- Inter-subject (data from multiple patients)
- Atlas (one image from patient and others from Atlas)

Objects

- Several organs are studied:
  - Brain
  - Heart
  - Spine
  - Others.

Registration Implementation

- Recall: F and M are fixed and moving image and $\boldsymbol{T}(x, y)$ is 2D spatial transform.
- How to warp moving image to get registered image?
  - Forward warping (mapping)
  - Backward warping (mapping)

Forward Mapping

› For each position (x, y) of the template image, the corresponding intensity value is stored in the new image at the location $\boldsymbol{T}(x, y)$:

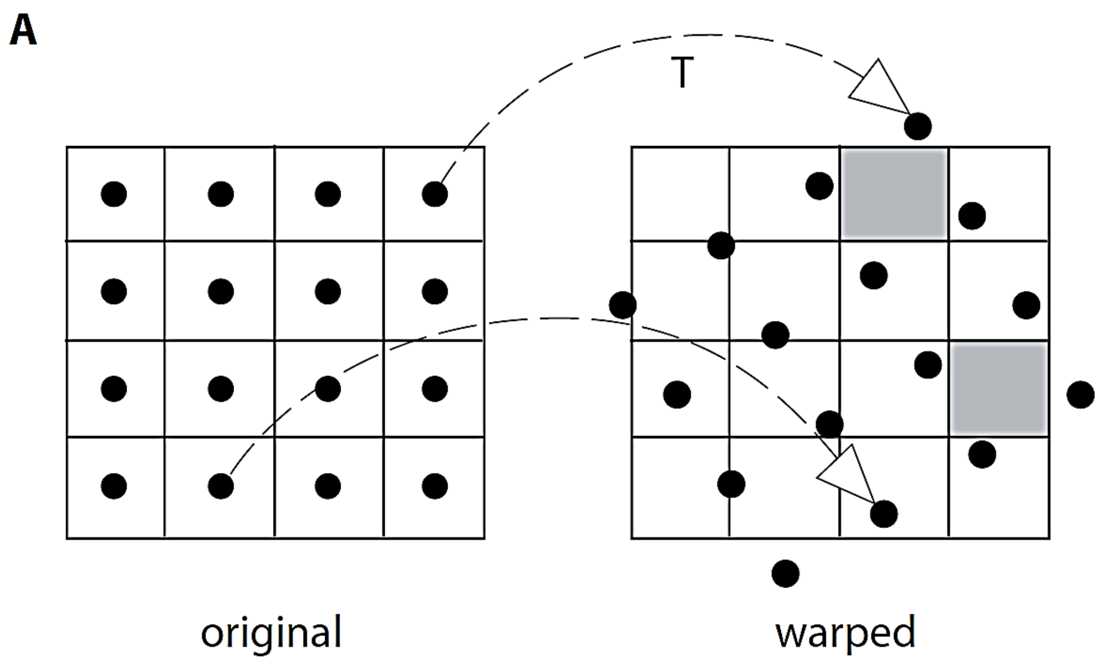
$$M_{warped}\big(\boldsymbol{T}(x, y)\big) \leftarrow M(x, y)$$

› The problem with this intuitive approach is that the transformation function is generally neither *injective* nor *surjective,* due to the discrete nature of pixel images, non-integer values of the transformation function have to be rounded.

› As a result, *not* every pixel in the new image will be necessarily *assigned* a value and some pixels can be *assigned several* times.

Forward Mapping

› Illustration:



› We need scattered data interpolation

Backward Mapping

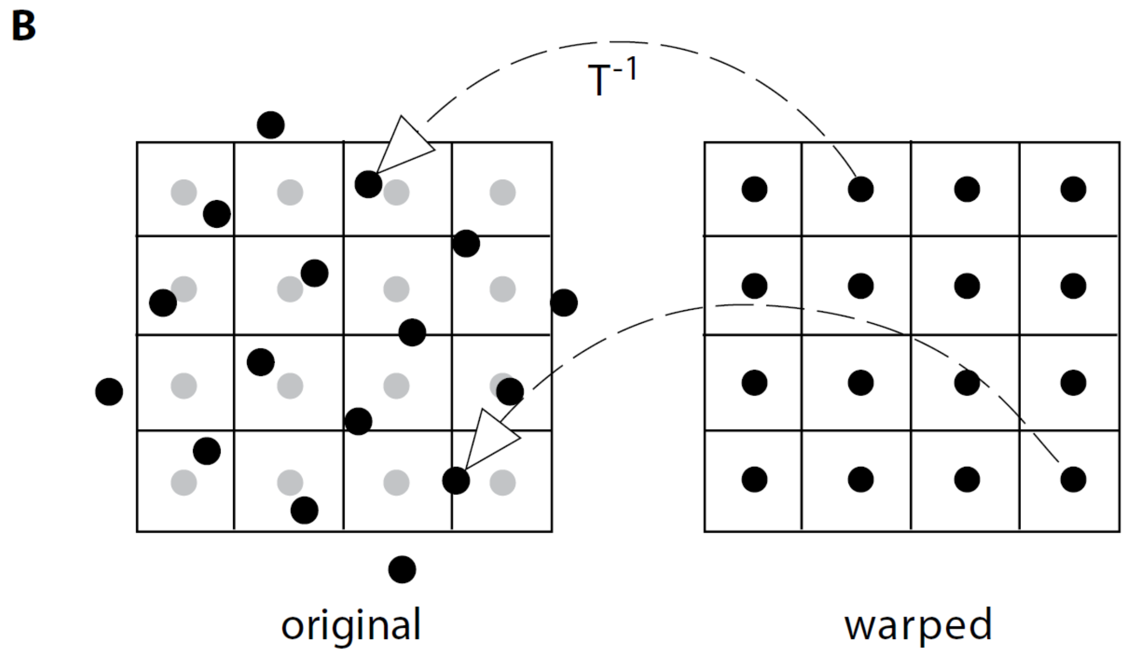
- The main difference is that now for every pixel of the new image a coordinate in the original image is computed, where its intensity value originates from:

$$M_{warped}(x, y) \leftarrow M\left(\boldsymbol{T}^{-1}(x, y)\right)$$

- In analogy to forward warping, it is possible that $\boldsymbol{T}^{-1}(x, y)$ yields a non-integer value. However, in the case of backward warping an interpolation scheme on the original image can be used to obtain intensity values at coordinates between pixels.
- Bilinear or trilinear interpolation (for 2D and 3D) are generally reasonable choices. *Unfortunately* the inverse of the transformation function is often not trivial to obtain.

Backward Mapping

› Illustration:



› We need grid data interpolation

Backward Mapping

- For small deformation,

$$\boldsymbol{T}(x, y) = (x, y) + \Delta\boldsymbol{T}(x, y)$$

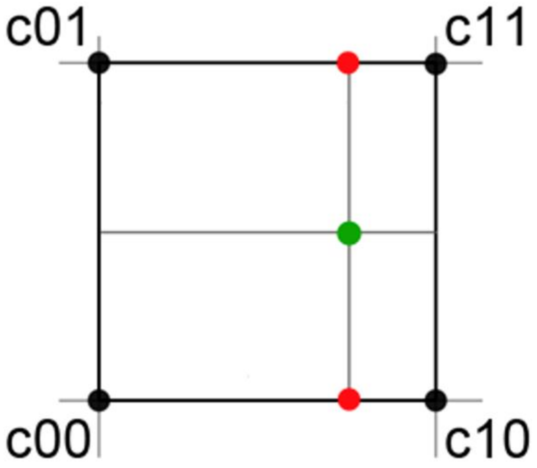
- The inverse transformation approximated via:

$$\boldsymbol{T}^{-1}(x, y) \approx (x, y) - \Delta\boldsymbol{T}(x, y)$$

Bilinear interpolation (for gray level)

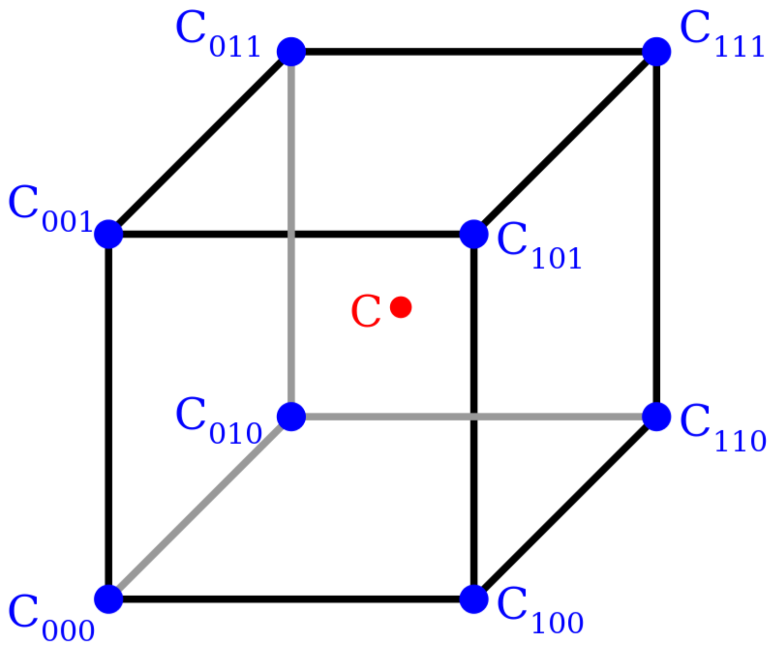
- Mathematical Formulation:
- $f(x, y) \approx a_0 + a_1 x + a_2 y + a_3 xy$
- For unit square (after shift to origin):

$$f(x, y) \approx [1 - x, x] \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1 - y \\ y \end{bmatrix}$$



Trilinear interpolation (for gray level)

- Mathematical Formulation
- $f(x, y, z) \approx a_0 + a_1x + a_2y + a_3z + a_4xy + a_5xz + a_6yz + a_7xyz$



Global Geometric Mapping

- Rigid body transform
- Procrustean transform
- Affine transform
- Projective transform
- Bilinear transform
- Curved transform

Rigid Body Transform

- Rotation and Translation
- 3-6 freedom degree in 2D-3D cases.
- Application: Hard organs: Radiology, Spinal cord, Hip, skull, and femur.

Rigid Body Transform

- Formulation for N–to-N correspondence ($N >> D$)
- Data: $\{\mathbf{x}_i\}_{i=1}^{N} \in \mathbb{R}^D, \{\mathbf{y}_i\}_{i=1}^{N} \in \mathbb{R}^D, \mathbf{x}_i \rightsquigarrow \mathbf{y}_i$
- Model: $\boldsymbol{y} = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}, \boldsymbol{R} \in \mathbb{R}^{D\times D}, \boldsymbol{t} \in \mathbb{R}^D$
- Goal:

$$\min_{\boldsymbol{R},\boldsymbol{t}} \left\{ \sum_{i=1}^{N} \|\boldsymbol{R}\boldsymbol{x}_i + \boldsymbol{t} - \boldsymbol{y}_i\|^2 \right\}, \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}_{D\times D}$$

Rigid Body Transform

- Reformulation:

$$\left.\begin{aligned} \boldsymbol{X} &= [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N] \in \Re^{D\times N} \\ \boldsymbol{Y} &= [\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_N] \in \Re^{D\times N} \\ \boldsymbol{T} &= [\boldsymbol{t}, \boldsymbol{t}, \dots, \boldsymbol{t}] \in \Re^{D\times N} \end{aligned}\right\} \Rightarrow \min_{\boldsymbol{R},\boldsymbol{t}} \left\{ \|\boldsymbol{RX} + \boldsymbol{T} - \boldsymbol{Y}\|_F^2 \right\}, \boldsymbol{RR}^T = \boldsymbol{I}_{D\times D}$$

- Shift to Center: $\widetilde{\boldsymbol{X}} = \boldsymbol{X} - \overline{\boldsymbol{X}}, \widetilde{\boldsymbol{Y}} = \boldsymbol{Y} - \overline{\boldsymbol{Y}}$

$$\left\|\widetilde{\boldsymbol{Y}} - \boldsymbol{R}\widetilde{\boldsymbol{X}}\right\|_F^2 = tr\left\{\widetilde{\boldsymbol{X}}^T\widetilde{\boldsymbol{X}}\right\} + tr\left\{\widetilde{\boldsymbol{Y}}^T\widetilde{\boldsymbol{Y}}\right\} - 2tr\left\{\widetilde{\boldsymbol{Y}}\widetilde{\boldsymbol{X}}^T\boldsymbol{R}^T\right\}$$

- We need maximize:

$$tr\left\{\widetilde{\boldsymbol{Y}}\widetilde{\boldsymbol{X}}^T\boldsymbol{R}^T\right\}$$

Rigid Body Transform

- We need maximize:

$$tr\{\widetilde{\boldsymbol{Y}}\widetilde{\boldsymbol{X}}^T\boldsymbol{R}^T\}$$

- Solution:

$$\widetilde{\mathbf{Y}}\widetilde{\mathbf{X}}^T \underbrace{=}_{SVD} \mathbf{W}\Sigma\mathbf{V}^T \rightarrow \begin{cases} \boldsymbol{R} = \boldsymbol{W}\mathbf{V}^T \\ \boldsymbol{t} = \overline{\boldsymbol{Y}} - \boldsymbol{R}\overline{\boldsymbol{X}} \end{cases}$$

Procrustean Transform

- Rotation, Scaling and Translation:
- $\mathbf{X}_d = \boldsymbol{R}\mathbf{X}_s + \boldsymbol{t}$
- $R = \begin{bmatrix} k\cos(\theta) & k\sin(\theta) \\ -k\sin(\theta) & k\cos(\theta) \end{bmatrix}, t = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$
- $R = \begin{bmatrix} k_x\cos(\theta) & k_y\sin(\theta) \\ -k_x\sin(\theta) & k_y\cos(\theta) \end{bmatrix}, t = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$

Procrustean Transform

- Formulation for N–to-N correspondence (N>>2)

$$\left.\begin{array}{l} \boldsymbol{K} = diag(k_1, k_2, \cdots, k_D) \\ \min\limits_{\boldsymbol{R},\boldsymbol{K}} \left\{ \left\| \boldsymbol{R}\widetilde{\boldsymbol{X}} - \widetilde{\boldsymbol{Y}} \right\|_F^2 \right\}, \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{K}^2 \\ \boldsymbol{R} = \widehat{\boldsymbol{R}} \times \boldsymbol{K} \end{array}\right\} \Rightarrow \rho = \left\| \widehat{\boldsymbol{R}}\boldsymbol{K}\widetilde{\boldsymbol{X}} - \widetilde{\boldsymbol{Y}} \right\|_F^2, \qquad \widehat{\boldsymbol{R}}\widehat{\boldsymbol{R}}^T = \boldsymbol{I}_{D\times D}$$

- We need two steps optimization

Procrustean Transform

- Iterative Solution:
- Initialization: $\boldsymbol{K}(0) = diag(1,1,\cdots,1)$
- Solve for rotation:

$$\min_{\widehat{\boldsymbol{R}}}\left\{\left\|\widehat{\boldsymbol{R}}\boldsymbol{K}\widetilde{\boldsymbol{X}} - \widetilde{\boldsymbol{Y}}\right\|_F^2\right\}, \widehat{\boldsymbol{R}}\widehat{\boldsymbol{R}}^T = \boldsymbol{I}_{D\times D}$$

- Update $\boldsymbol{K}$ using:

$$\frac{\partial\rho}{\partial k_i} = -2\sum_{j=1}^{D}\left[\widetilde{\boldsymbol{Y}}\widetilde{\boldsymbol{X}}^T\right]_{ji}\widehat{\boldsymbol{R}}_{ji} + 2k_i\sum_{j=1}^{N}\left[\widetilde{\boldsymbol{X}}\right]_{ij}^2 = 0$$

- check for convergence

Affine Transform

› Affine Transform:

› 6-12 free parameters (2D-3D):

$$\mathbf{R} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \mathbf{t} = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

› Solution:

$$\min_{\boldsymbol{R}} \left\{ \left\| \widetilde{\boldsymbol{Y}} - \boldsymbol{R}\widetilde{\boldsymbol{X}} \right\|_F^2 \right\} \Rightarrow \begin{cases} \boldsymbol{R} = \left(\widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{X}}^{\boldsymbol{T}} \right)^{-1} \widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{Y}}^{\boldsymbol{T}} \\ \boldsymbol{T} = \overline{\boldsymbol{Y}} - \boldsymbol{R}\overline{\boldsymbol{X}} \end{cases}$$

Projective Transform

- Projective-Perspective Transform:
- 8 free parameters (2D):

$$x_d = \frac{a_{11}x_s + a_{12}y_s + a_{10}}{b_1x_s + b_2y_s + 1}, \; y_d = \frac{a_{21}x_s + a_{22}y_s + a_{20}}{b_1x_s + b_2y_s + 1}$$

- Solution:

$$\min_{\boldsymbol{a},\boldsymbol{b}} \sum_{i=1}^{N} \left(x_d^{(i)}(b_1x_s^{(i)} + b_2y_s^{(i)} + 1) - (a_{11}x_s^{(i)} + a_{12}y_s^{(i)} + a_{10})\right)^2 +$$

$$\sum_{i=1}^{N} \left(y_d^{(i)}(b_1x_s^{(i)} + b_2y_s^{(i)} + 1) - (a_{21}x_s^{(i)} + a_{22}y_s^{(i)} + a_{20})\right)^2$$

Bilinear Transform

- Bilinear Transform:
- 8 free parameters (2D):

$$x_d = a_{00} + a_{10}x_s + a_{01}y_s + a_{11}x_s y_s$$
$$y_d = b_{00} + b_{10}x_s + b_{01}y_s + b_{11}x_s y_s$$

- Solution: similar to perspective transform

Polynomial Transform

- Polynomial Transform:

$$x_d = a_{00} + a_{10}x_s + a_{01}y_s + a_{11}x_s y_s + a_{20}x_s^2 + a_{02}y_s^2 + \ldots$$
$$y_d = b_{00} + b_{10}x_s + b_{01}y_s + b_{11}x_s y_s + b_{20}x_s^2 + b_{02}y_s^2 + \ldots$$

- Solution: similar to perspective transform

The End

› AnY QuEsTiOn?