

Optimal Step Nonrigid Iterative Closest Point Algorithms for Surface Registration

Brian Amberg, Sami Romdhani, Thomas Vetter

CVPR 2007

Iterative Closest Point rigide : algorithme



Soit une *source* $\mathcal{S}$ et une *cible* $\mathcal{T}$, on recherche la déformation rigide $\mathbf{X} \in \mathbb{R}^{3 \times 4}$ (rotation et translation) qui minimise :

$$E(\mathbf{X}) := \sum_{\mathbf{s}_i \in \mathcal{S}} w_i \text{dist}(\mathcal{T}, \mathbf{X}\mathbf{s}_i)^2 = \sum_{\mathbf{s}_i \in \mathcal{S}} w_i \|\mathbf{X}\mathbf{s}_i - \mathbf{t}_i\|^2$$

Input: $\mathcal{S}$: *source*, $\mathcal{T}$: *cible*

Output: $\mathbf{X}$: déformation rigide

initialise $\mathbf{X}$

while $E(\mathbf{X}) > \varepsilon$ **do**

 /* Etape d'assignation: construction des $\mathbf{t}_i$ */

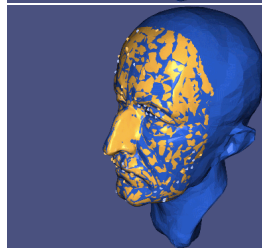
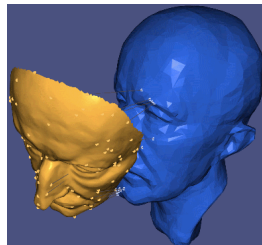
 projette $\mathbf{X}\mathcal{S}$ sur $\mathcal{T}$

 /* Etape de déformation: calcul de $\mathbf{X}$ */

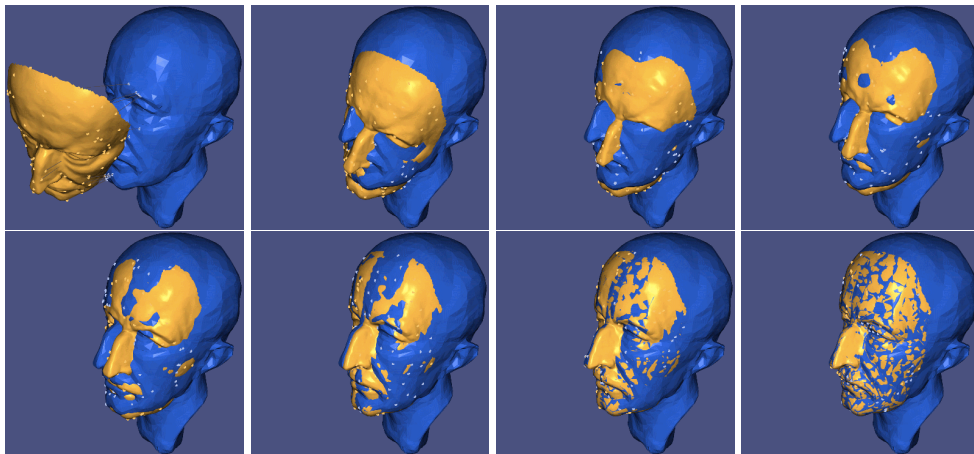
 minimise $E(\mathbf{X})$ selon les paires $\{(\mathbf{s}_i, \mathbf{t}_i)\}$

end

return $\mathbf{X}$



Iterative Closest Point rigide : exemple



Iterative Closest Point non rigide : le problème



Soit une *source* $\mathcal{S}$ et une *cible* $\mathcal{T}$, on recherche la déformation non rigide $\mathbf{X} = \{\mathbf{X}_i \in \mathbb{R}^{3 \times 4}\}$ (étirement, rotation et translation) qui aligne au mieux $\mathcal{S}$ sur $\mathcal{T}$.



Iterative Closest Point non rigide : les énergies



Energie de proximité :

$$E_d(\mathbf{X}) := \sum_{\mathbf{s}_i \in \mathcal{S}} w_i \text{dist}(\mathcal{T}, \mathbf{X}_i \mathbf{s}_i)^2 = \sum_{\mathbf{s}_i \in \mathcal{S}} w_i \|\mathbf{X}_i \mathbf{s}_i - \mathbf{t}_i\|^2$$

Energie de régularisation :

$$E_s(\mathbf{X}) := \sum_{\{i,j\} \in \mathcal{S}} \|(\mathbf{X}_i - \mathbf{X}_j) \mathbf{G}\|^2, \quad \mathbf{G} := \text{diag}(1, 1, 1, \gamma)$$

Energie de « landmarks » :

$$E_l(\mathbf{X}) := \sum_{(\mathbf{s}_i, \mathbf{l}) \in \mathcal{L}} \|\mathbf{X}_i \mathbf{s}_i - \mathbf{l}\|^2$$

Energie globale avec α un paramètre de raideur :

$$E(\mathbf{X}) := E_d(\mathbf{X}) + \alpha E_s(\mathbf{X}) + \beta E_l(\mathbf{X})$$

Iterative Closest Point non rigide : l'algorithme



$$E(\mathbf{X}) := E_d(\mathbf{X}) + \alpha E_s(\mathbf{X}) + \beta E_l(\mathbf{X})$$

Input: $\mathcal{S}$: source, $\mathcal{T}$: cible

Output: $\mathbf{X}$: déformation non rigide par vertex

initialise $\mathbf{X}^0$

foreach $\alpha^i \in \{\alpha^1, \dots, \alpha^n\}$, $\alpha^i > \alpha^{i+1}$ **do**

 /* Minimisation de $E(\mathbf{X})$ pour une raideur fixe */

while $\|\mathbf{X}^j - \mathbf{X}^{j-1}\|^2 > \varepsilon$ **do**

 /* Etape d'assignation: construction des $\mathbf{t}_i$ */

 projette $\mathbf{X}^{j-1}\mathcal{S}$ sur $\mathcal{T}$

 /* Etape de déformation: calcul des $\mathbf{X}_i$ */

 /* voir papier pour le détail */

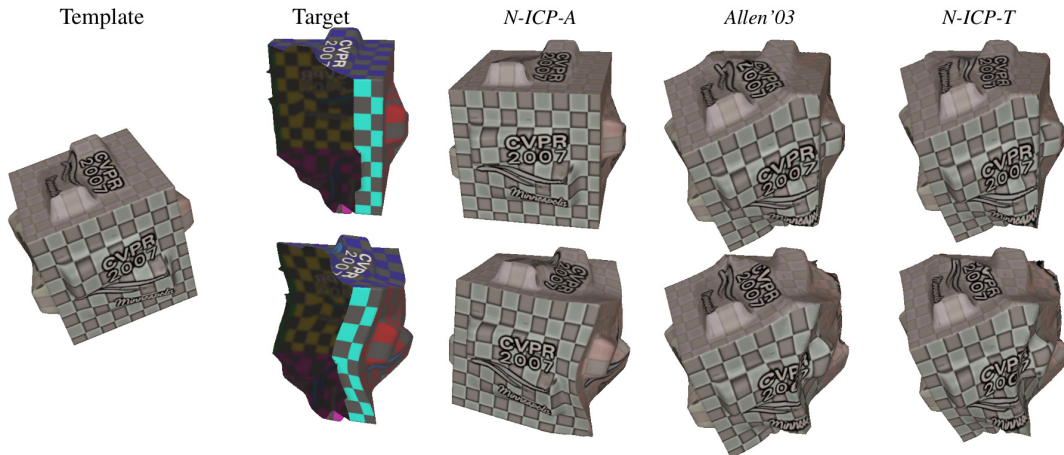
 minimise $E(\mathbf{X}^j)$ selon les paires $\{(\mathbf{s}_i, \mathbf{t}_i)\}$

end

end

return $\mathbf{X}$

Iterative Closest Point non rigide : resultats



Resources



■ papier :

https://gravis.dmi.unibas.ch/publications/2007/CVPR07_Amberg.pdf

■ code Matlab :

<https://github.com/RhythmJnh/Non-rigid-ICP>