Globally Constrained Deformable Models for 3D Object Reconstruction

Johan Montagnat, Hervé Delingette

I.N.R.I.A., BP 93, 06902 Sophia Antipolis Cedex, France

{Johan.Montagnat, Herve.Delingette}@sophia.inria.fr

Abstract

To achieve geometric reconstruction from 3D datasets two complementary approaches have been widely used. On one hand the deformable model framework locally applies forces to fit the data. On the other hand, the non-rigid registration framework computes a global transformation minimizing the distance between a template and the data. We first show that applying a global transformation on a surface template, is equivalent to applying certain global forces on a deformable model. Second we propose a scheme which combines the registration and free-form deformation. This globally constrained deformation model allows us to control the amount of deformation from the reference shape with a single parameter. Finally, we propose a general algorithm for performing model-based reconstruction in a robust and accurate manner. Examples on both range data and medical images are used to illustrate and validate the globally constrained deformation framework.

Key words: Registration; Free-form deformation; Deformable models; Segmentation; 3D Reconstruction; Medical images

1 Introduction

In this paper we are interested in model-based reconstruction of data and the segmentation of images. Model-based approaches proved to be powerful for many reconstruction tasks by providing an *a priori* shape knowledge of the data to recover. We intend to build a robust reconstruction algorithm for the modeling of 2D and 3D data using respectively deformable contours, and deformable surface meshes.

The model-based reconstruction problem can be summarized in the following manner: let $\mathbf{M}$ be a contour or surface model of a given object consisting of a set of vertices $\{V_i \in \mathbb{R}^d\}$, $(i = 1, \ldots, n)$. Let $\mathbf{D}$ be a 3D dataset. The problem

is to find a geometric transformation T such that $T[\mathbf{M}]$ is an appropriate representation of the object in $\mathbf{D}$. Two main model-based approaches can be found in the literature for the computation of T: *free-form deformation* and *registration*.

1.1 Free-form Deformation

With free-form deformation (FFD), the transformation T is only defined at the model vertices. There is no *a priori* restriction on the transformation which can be seen as a set of independent displacement vectors: $\forall V_i \in \mathbf{M} \longrightarrow T[V_i] \in \mathbb{R}^d$.

The *active contours* originally proposed by Kass et al. [9] derive the transformation T from *external* forces representing data attachment and *internal*, or regularizing, forces. The external forces are devised to deform the model towards the data. The internal forces are based on the intrinsic geometric properties of the model used. The FFD scheme is illustrated in figure 1.

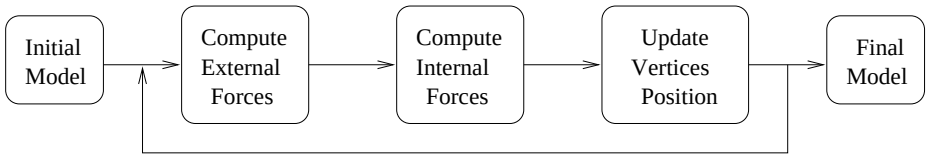


Fig. 1. The iterative scheme for free-form deformation.

Each model vertex V_i is thus submitted to a weighted sum of internal, f_i^{int}, and external, f_i^{ext}, forces. We use a Newtonian law of motion to update each vertex, V_i, position: $m\frac{d^2V_i}{dt^2} = -\delta\frac{dV_i}{dt} + \alpha f_i^{int} + \beta f_i^{ext}$, where α and β are weights and δ is a damping coefficient. By discretizing time and space with finite differences we get an explicit scheme:

$$V_i^{t+1} = V_i^t + (1-\delta)(V_i^t - V_i^{t-1}) + \alpha f_i^{int} + \beta f_i^{ext} \tag{1}$$

where V_i^t is the location of vertex V_i at time t. In this equation, the time step Δt does not appear. It is part of the coefficients δ, α, and β. FFD is very flexible since the only regularizing constraints are internal forces computed locally at each vertex. It is also quite sensitive to noise and outliers. Section 4 shows FFD a example and associated problems.

1.2 The Registration Framework

Registration of a template consists of finding a global transformation T minimizing the distance between the transformed template and the dataset. T is

constrained to belong to a transformation space $\mathbf{T}_{reg}$, usually a group for the composition operator: $T \in \mathbf{T}_{reg}, \forall P \in \mathbb{R}^d \longrightarrow T[P] \in \mathbb{R}^d$.

Typical registration methods iteratively compute the best transformation through least squares estimation until some convergence criterion is met. Besl [1] and Zhang [22] proposed an Iterative Closest Point algorithm (or ICP). The general registration scheme is illustrated in figure 2. As will be shown in section 5, registration provides a robust framework for shape recovery by restricting the set of allowed transformations. Yet, it suffers from limited shape variability or computation instability problems.

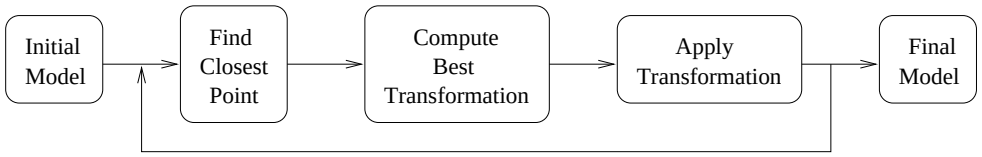


Fig. 2. The iterative scheme for template registration.

1.3 Known Problems and Previous Work

For many shape-based reconstruction problems, the registration approach does not have enough degrees of freedom (DOF) for the recovery of complex and high variable shapes, contrary to FFD which may not enforce the regularity of the deformation to make the model robust to noise and data outliers. Internal forces, even computed at a large scale over the surface, are not equivalent to restricting the number of DOF in a model.

Several approaches have been proposed in computer vision for controlling the amount of allowed deformation of a model. On one hand, researchers have introduced more general deformations in the registration framework either by applying global transformations locally [8] or by regularizing deformation fields [20].

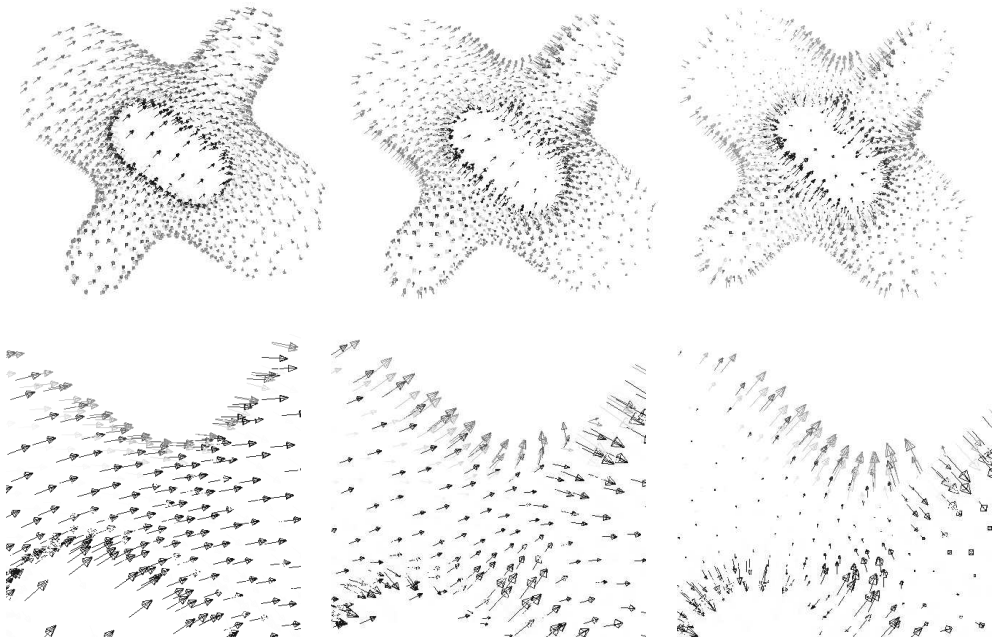
On the other hand, several methods for improving the global control of FFD have been devised. A first approach consists of decomposing a deformation field into a set of hierarchical deformation modes. For instance, in [11, 19] Metaxas and Terzopoulos superimpose a deformable superquadrics over a surface spline function. Modal analysis [13, 17], Fourier domain analysis [16], or wavelet basis [21] provide a set of deformation modes with a decreasing scale of extent. Principal component analysis [3] introduces meaningful deformation modes from statistical study of shape variation.

A second approach consists of including global parameters in the regularizing functionals. In [18], translation and rotation parameters are included in the internal energy to take, explicitly, into account rigid motions. In [5], Delingette

introduces a rigidity parameter that controls the extent of the regularizing forces.

2 Globally constrained deformation

In this section we provide a link between computing a global transformation and applying an external force field on a model $\mathbf{M}$. We then propose a unified framework encompassing both FFD and registration.



(a) rigid ($\lambda = 0$) (b) GCD ($\lambda = 0.2$) (c) FFD ($\lambda = 1$)

Fig. 3. Different displacement fields on a cross-shaped surface.

2.1 Registration as a Displacement Field

Let V_i^t be the vertex position of $\mathbf{M}$ at time t. In the registration framework, we first find the best transformation T^t and finally apply the transformation at each vertex: $V_i^{t+1} = T^t[V_i^t]$. The displacement field applied between two iterations is therefore $T^t - I$ where I is the identity transformation: $D = T^t - I = \{T^t[V_i^t] - V_i^t\}$. To illustrate we 3. apply some external forces on a cross-shaped surface model and we plot the displacement vector at each model vertex. Figure 3(a) shows the displacement field resulting from a rigid transformation while figure 3(c) shows the displacement field of FFD.

In equation 1, we have seen that the displacement field applied in the FFD is equal to: $\{V_i^{t+1} - V_i^t\} = \{(1-\delta)(V_i^t - V_i^{t-1}) + \alpha f_i^{int} + \beta f_i^{ext}\}$. We can therefore consider that the registration method based on the ICP algorithm is equivalent to having a deformable model without inertial and internal forces ($\delta = 1$, $\alpha = 0$) and submitted to the *global* force $f_i^{global} = T^t[V_i^t] - V_i^t$.

The global force is an external force which leads to a deformation field with fewer DOF than the usual external forces since it is constrained by the nature of the transformation T.

2.2 Globally Constrained Deformation

We have demonstrated the equivalence between registration-based deformation and the application of a global force to a deformable model. We now propose a constrained deformation scheme, where a deformable model is submitted to global, external, and internal forces. The purpose of this scheme is to have a computer-efficient deformable model with an easy control on the number of DOF. Our approach is to weight with a single parameter λ the influence of the global forces versus the local forces (see figure 4):

$$V_i^{t+1} = V_i^t + (1-\delta)(V_i^t - V_i^{t-1}) + \lambda\left(\alpha f_i^{int} + \beta f_i^{ext}\right) + (1-\lambda) f_i^{global} \quad (2)$$

We call λ, the *locality* parameter. It controls the number of DOF of the deformable model. With $\lambda = 0$, the model is under the influence of global forces and internal forces, and therefore has very few DOF. With $\lambda = 1$, the model is under the influence of internal forces and external forces and therefore has a maximum number of DOF. With intermediate values of λ, we can control the shape variation allowed during the deformation. Figure 3(b) shows a globally constrained deformation (GCD) field with $\lambda = 0.2$.

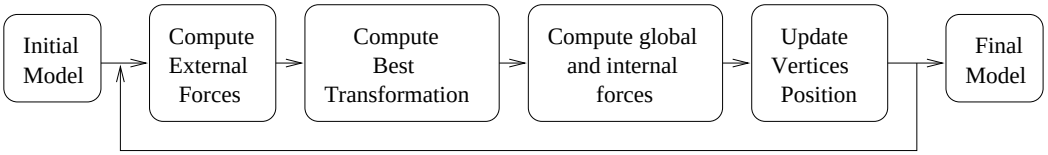


Fig. 4. The iterative scheme for globally constrained deformations.

3 Implementation

To motivate the introduction of the GCD framework, we perform experiments using our deformable model implementation based on *simplex meshes*. This is

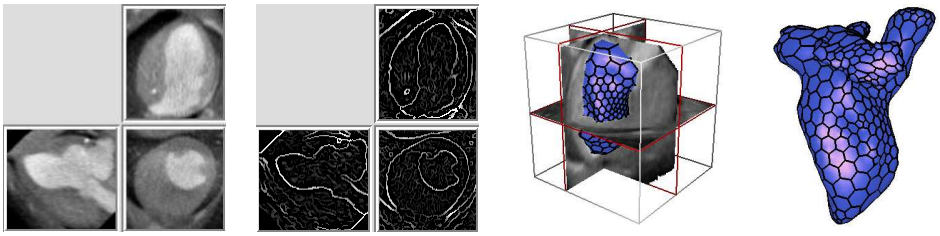
a discrete representation of models whose formalism applies in 2D (deformable contours or snakes) as well as 3D (deformable surface meshes).

Simplex meshes [6] are meshes with constant vertex adjacency. They allow a local description (at each vertex) of the shape they represent. They may model objects with almost any topology without dealing with surface parameterization problems. 2-simplex meshes are a natural extension of snakes in 3D and they provide a powerful framework to express regularizing constraints. The simplex mesh framework is computationally very efficient since local force computation does not require a minimization step.

For internal forces, we are using a *shape* constraint that enforces local shape around a vertex up to a similarity. It provides an interesting behavior to models in parts where no dataset is available: they tend to keep locally the reference shape. Registration and FFD require the model vertices to match data points. To enforce robustness against outliers, we search for a data point to match vertex V_i only within a limited distance of V_i. We compute the external force f_i^{ext} as a vector directed along the normal direction proportional to the distance from V_i to the dataset.

4 Free-Form Deformation Example

Figure 5 shows an example of FFD used to segment a left ventricle from an MR image. Figure 5(a) shows three orthogonal slices of the volumetric image; (b) is the gradient image for the corresponding slices; (c) shows the model inside the volumetric image; initialized as a sphere it deforms toward image gradient points; (d) is the resulting model after a refinement stage.



(a) image (b) gradient (c) deformation (d) result

Fig. 5. Free-form deformation example.

FFD allows the model to deform with few restrictions. This enables the reconstruction of complex shapes such as the ventricle shown in figure 5(d) starting from a very different template (here a 800 vertices sphere). However, in case of noisy or sparse data, FFD leads to an under-constrained system and the

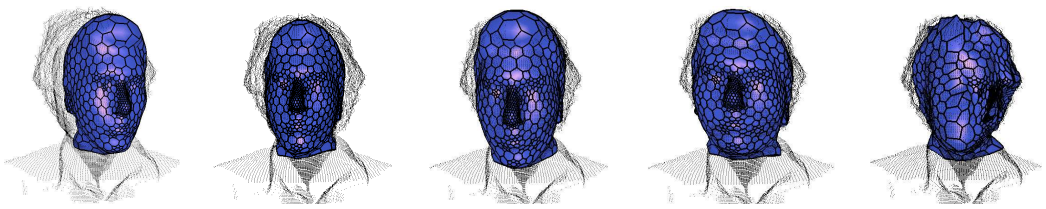
model is likely to lock on unwanted local energy minima. An under-constrained model is also likely to intersect itself.

5 Registration Examples

We implemented rigid transformations (6 DOF), similarities (7 DOF), and affine transformations (12 DOF) using the closed-form expression proposed in [14]. We also investigated a cubic B-spline transformation evaluation (many DOF depending on the number of knots) based on a gradient descent [4].

5.1 Influence of the Degrees of Freedom

Figure 6 shows a face reconstruction example using the four transformation classes presented above. The dataset is a cloud of 3D points acquired with a Cyberware scanner. The template (rendered surface) represents another face. The role of the template is to provide an a priori information on the shape to recover. Thus it is geometrically adapted (vertices are concentrated in high curvature areas).



(a) initial (b) rigid (c) similar (d) affine (e) spline

Fig. 6. Face registration with different transformations.

It appears that increasing the number of DOF improves accuracy of the fit. We measure the goodness of the fit by computing the accumulated distance between each vertex and its closest data point (see figure 7). In the case of cubic B-spline deformations, the number of DOF is sufficient to get a close approximation of the data. However, the entailed deformations do not preserve the *a priori* geometry of the model (for instance, due to the initial position of the model, nose vertices migrated toward the left jaw).

Figure 7 also compares the computation time of the four transformations used. In general, increasing the number of DOF increases the computation time since it involves the convergence of a minimization stage in the ICP algorithm. However, affine transformations are slightly more efficient to compute than rigid ones (see [14] for details).

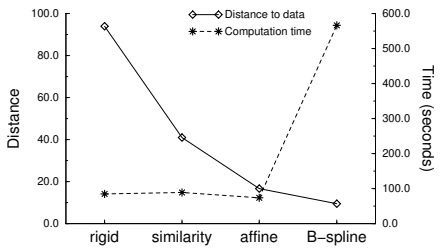


Fig. 7. Comparison between transformation classes.

5.2 Computation Cost Issues

As can be seen in figure 7, increasing the number of DOF causes an augmentation of the computation time. In particular, the B-spline transformation is far more expensive than other transformation classes. The registration computation cost is the sum of the cost for finding the closest points and the LSE algorithm: $C_{registration} = C_{closest}(N) + C_{LSE}(D)$ where N is the number of model vertices and D the number of transformation DOF. $C_{closest}(N)$ is only dependent on the number of vertices. $C_{LSE}(D)$ depends on the number of DOF and the LSE algorithm used. Thus $C_{closest}(N)$ is constant for different transformations, and $C_{registration}$ is almost constant as long as $C_{LSE}(D)$ remains negligible compared to $C_{closest}(N)$; this is the case for rigid, similarities and affine transformations. For the B-spline transformation, the LSE algorithm uses a gradient descent method (see [4] for details) which is much more costly than the matching algorithm.

5.3 Numerical Stability Issues

Another issue is the stability of the method used. The number of equations needed is proportional to the number of DOF of the transformation. In the case of an affine transformation, only 12 parameters are evaluated. With more DOF, a sufficient number of equations, i.e. a minimum number of matches between model vertices and data points, must be found to stabilize the equation system. We encountered numerical stability problems with B-splines when data is noisy and no enough matches can be found.

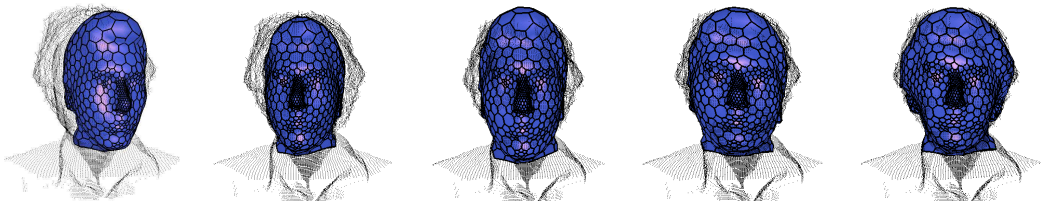
6 A General Algorithm for Model-Based Segmentation

To overcome problems encountered with the FFD and the registration framework we proposed a GCD scheme that provides an intermediate behavior between these two ends. In fact, we integrate the GCD in a "coarse to fine"

algorithm that gradually increases the amount of deformation allowed during the deformation process.

6.1 Hierarchical Registration

In figure 6(e) we show a face reconstruction example using B-spline. The deformed surface is close to data points but the high number of B-spline DOF combined with a rough initialization lead to the destruction of the geometrical correspondences during the deformation process. To achieve an accurate reconstruction with proper geometrical correspondences, we propose to gradually increase the number of DOF. By starting the transformation with few DOF, we can enforce the robustness of the reconstruction. Figure 8 shows the face registration result, when first using rigid transformations, then similarity, then affine, and finally B-spline transformations.



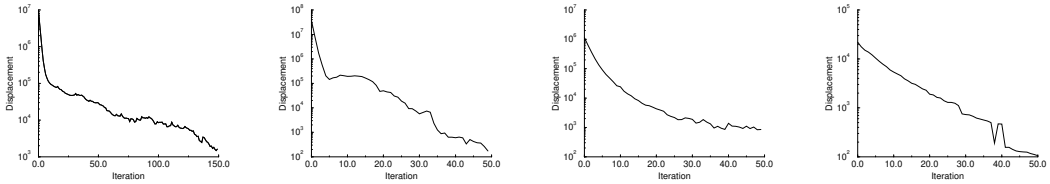
(a) initial (b) rigid (c) similar (d) affine (e) spline

Fig. 8. Registration with increasing number of DOF.

The remaining problem is to determine when to change the transformation class and therefore increase the number of DOF. Consider figure 9 which represents for each registration stage the total displacement of all model vertices between two successive iterations of the ICP algorithm. Due to the fast convergence of the ICP algorithm (notice the log-scale on the Y axis), the displacement is sharply decreasing. When the displacement is low enough the model does not evolve significantly anymore. Therefore we can set a low threshold to stop a deformation stage and increase the number of DOF by changing the transformation. This leads to good results with preservation of the model geometry as can be seen in figure 8(e).

We propose two strategies to set the low threshold. An *absolute threshold* value can be provided by the user as a percentage of the initial mesh size. A *relative threshold* is computed as a percentage of the initial displacement. After evolving a few iterations (three for instance), we compute the mean displacement and we set the threshold as a fraction of the obtained distance.

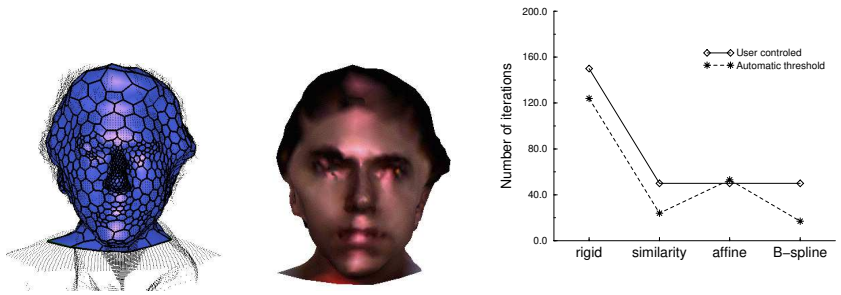
We made the face registration experiment with automatic thresholding: each deformation stage ends when the model displacement is lower than 0.1% of



(a) rigid (b) similarity (c) affine (d) B-spline

Fig. 9. Displacement of model.

initial displacement. Figure 10(a, b) shows the resulting mesh. Figure 10(c) compares the manual and automatic threshold strategies. This reveals a significant diminution in the total computation time. More convergence results are given in section 7.2.3.



(a) result (b) texture (c) iterations

Fig. 10. Registration with automatic thresholding.

6.2 From Rigid Registration to Free-form Deformation

A natural idea is to extend the previous incremental process using GCD. To optimize computational cost and numerical stability we use transformations with few DOF such as rigid, similarity, or affine transformations. The locality parameter allows us to tune the number of DOF. GCD extends the registration framework in a computationally efficient manner since the search for the closest points has to be performed only once for the computation of the global and external forces. The resulting deformation scheme is illustrated in figure 11. We first increase the number of DOF by making the global transformation evolve. We then gradually increase λ using the same displacement criterion. We can therefore have an automatic model-based reconstruction algorithm.

It is difficult to estimate the maximal value that the locality parameter should have (do we go up to FFD or do we stop for a maximal λ value?). With high

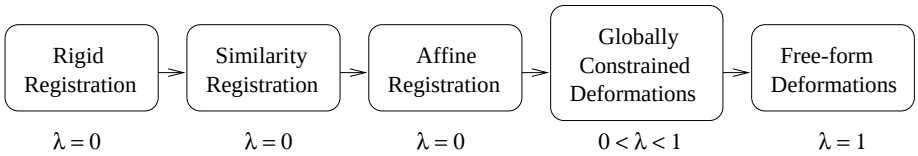


Fig. 11. General framework.

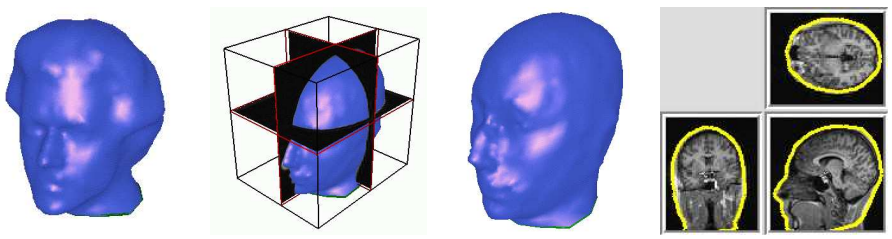
λ values the model is more likely to fit the data but it is also more sensitive to noise. Thus the λ maximal value choice is a trade-off between reconstruction accuracy and data quality. We let the user decide.

7 Results

In this section, we present reconstruction and segmentation examples to validate our algorithm.

7.1 Qualitative Results

We consider a multi-modality face reconstruction example by fitting a face template on an MR image of the head. Figure 12(a) shows a face template. The model is embedded in the volume image (b) then registered and deformed toward the edge point using the GCD. The resulting surface is shown in (c). A control of the result (the model intersection with three orthogonal image slices) can be seen in (d).



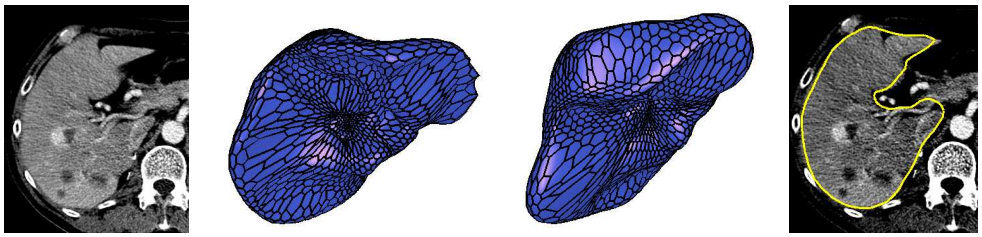
(a) model (b) deformed (c) result (d) control

Fig. 12. Face reconstruction from an MR image.

Image segmentation is a key issue in many computer vision applications. We demonstrate here the ability of the GCD to automatically segment anatomical structures. For example, the liver is a soft organ that exhibits drastic inter-patient variations. The model has to be deformable enough to take into account the shape variations. Abdominal organs have close grey-level values and they

are difficult to isolate. Therefore the model has to be constrained enough to be robust against noise and weak contours while deforming.

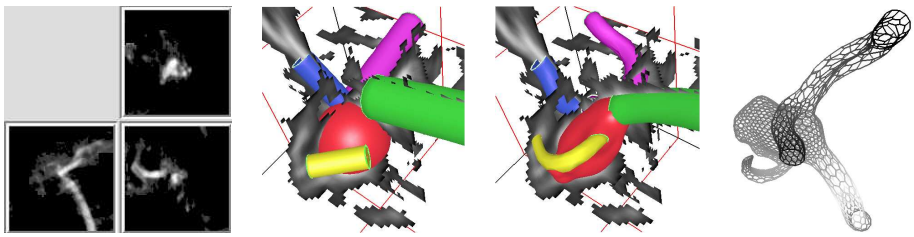
Figure 13 shows a liver segmentation example from an abdominal CT-scan. One slice of the CT-image can be seen in (a), the template (b) is first registered then iteratively deforms by gradually increasing the locality parameter up to 35% (c). The result can be seen as the model trace in one slice of the data (d).



(a) image (b) template (c) deformed (d) result

Fig. 13. Liver segmentation from contrast enhanced spiral CT-scan image.

Another application is the segmentation of blood vessels and aneurisms from angiographic images. For the segmentation of blood vessels, we use cylindrical models with a specific axial constraint. The axial constraint is a class of global transformations that makes the surface model bend along its axis. Details on the axial constraint computation can be found in [12]. Figure 14 (a) shows three orthogonal slices of an angiography. Models are initialized around the aneurism and the vessels (b). Then constrained deformations are used to fit the model (c). (d) shows the result obtained after having connected the different models through topological operations.



(a) slices (b) models (c) deform (d) result

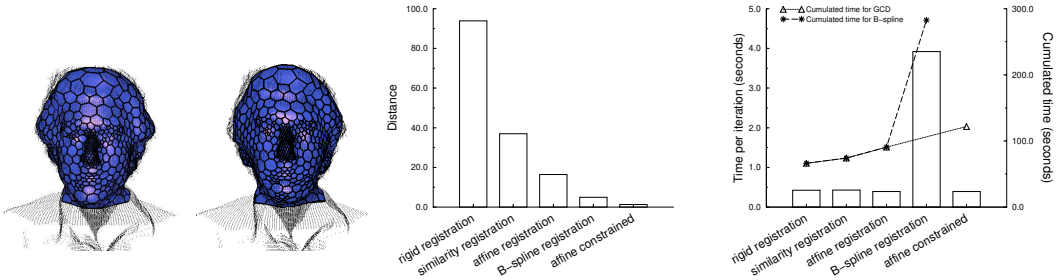
Fig. 14. Aneurism segmentation from an angiographic image.

7.2 Quantitative Results

The quantitative results given here validate the GCD accuracy for reconstruction and segmentation tasks.

7.2.1 Reconstruction accuracy

We ran the face reconstruction example proposed in section 5 using the GCD framework. The rigid, similar, and affine registration stages are identical. For high DOF fitting, we use GCD with affine constraint instead of B-spline registration. Figure 15 compares the result obtained by GCD (a) and B-spline registration (b). Since FFD has more DOF than B-spline registration, the GCD yields a more accurate fit. This can be seen in subfigure (c) showing the distance to data. Moreover, GCD has a lower computational cost due to the prohibitive cost of B-spline computation (subfigure (d)).

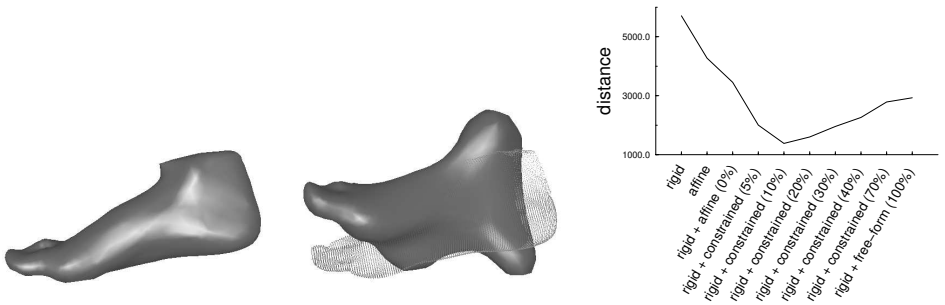


(a) (b) (c) (d)

Fig. 15. Comparison between the GCD and B-spline registration.

7.2.2 Geometric Properties

In this section, we demonstrate the relevance of the GCD scheme compared to the registration and FFD frameworks. Given some range data of a foot, we have built the model $\mathbf{M}$ of figure 16 (a). The model was manually deformed to get the model $\mathbf{M}^\star$ of figure 16 (b).



(a) template (b) initial (c) distance

Fig. 16. (a) foot model; (b) range data and deformed model; (c) distance to data.

We then fit the mesh $\mathbf{M}^\star$ on the original range data for a fixed number of iterations to get model $\mathbf{M}'$. Running the deformation process with different deformation schemes, the quality of registration is evaluated as the pointwise

distance between the reconstructed model $\mathbf{M}'$ and $\mathbf{M}$: $d = \sum_i \|V_i - V_i'\|^2$. A small value of d implies that vertices positions of $\mathbf{M}'$ are close to those of $\mathbf{M}$ and therefore the transformation is robust and accurate. Figure 16(c) shows the values of d for a rigid registration (first point), an affine registration (second point), or a GCD (with affine constraint) following a first rigid fit (other points). We get the best results at the intermediate values of the locality parameter λ whereas FFD or global (affine) transformation lead to maximum distance values. This result can be interpreted in the following manner : if the model is too constrained, it cannot deform enough to fit the dataset. On the contrary, if it has too many DOF, the surface vertices can be subject to large displacements during the deformation.

7.2.3 Convergence results

We used our algorithm to perform segmentation of the liver in images of an abdominal CT-scan database (an example of a particular image is given in figure 13). Figure 17 reproduces convergence results such as the one given in section 5 but at a larger scale. Figure 17(a) shows the number of iterations set manually (bars) for each deformation step. These thresholds have been chosen so that the deformation process properly segment any of our 30 liver images. They had to be high enough to allow convergence in any case. The solid line of figure 17(a) shows the mean number of iterations used by automatic thresholding and the minimal and maximal values. Figure 17(b) shows the convergence of the algorithm for six different models during the rigid registration step. Figure 17(c) shows the convergence of the GCD step (with $\lambda = 30\%$).

8 Conclusion

We have introduced a general reconstruction framework that encompasses both deformable models and registration approaches. GCD is an efficient and simple algorithm for controlling the amount of deformation. It leads to good conservation of shape properties (such as curvature) during the deformation.

We have shown that most accurate results are obtained by first globally registering the template. As the model converges, we increase the number of DOF by using the GCD scheme. The evolution of the λ parameter allows us to gradually increase the amount of deformation from affine transformation to free-form deformation.

Using deformable models we benefit from an *a priori* shape and geometrical information relevant throughout the deformation process. It is possible to obtain drastic shape variations of templates. By merging registration with the

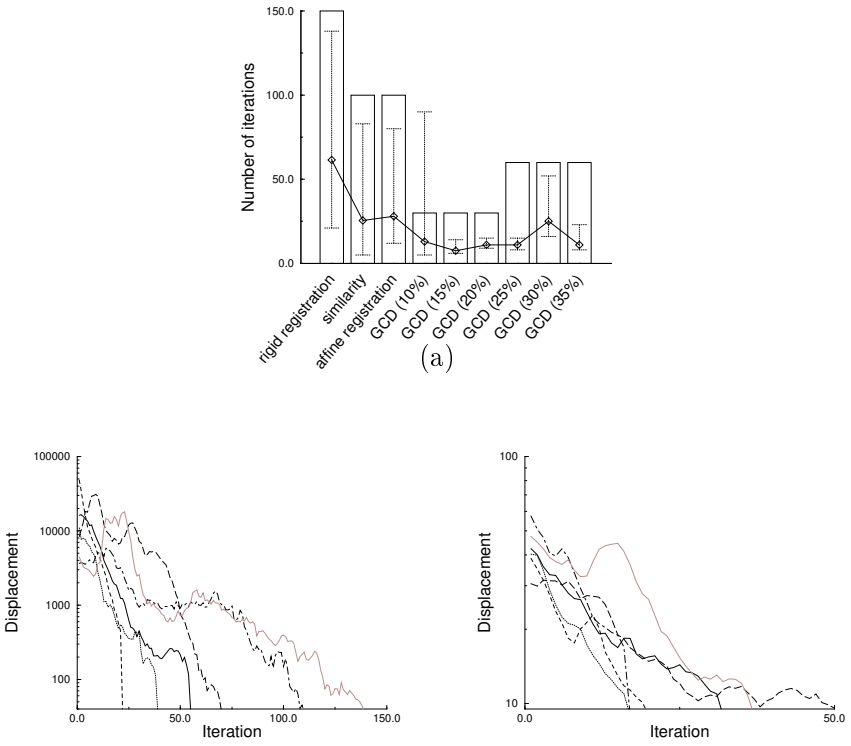


Fig. 17. Liver convergence measures.

free-form deformation framework, we make the deformation process robust to noise and outliers while remaining computationally efficient. However, the total amount of deformation allowed is left to the user. By constraining the model, we reduce problems that can arise with FFD such as model self intersection but we do not strictly avoid them. The whole deformation algorithm is also very dependent on the closest point computation.

In the future, we plan to incorporate some additional statistical information in the reference model, in order to introduce meaningful deformation constraints. The statistical study of medical databases should allow for the introduction of statistically relevant deformation modes.

Acknowledgment

We thank IRCAD and Dosigray for abdominal CT-scans, Brigham and Women's Hospital for brain MR images, and General Electric, Medical Systems Europe, for the 3D images of aneurisms (Courtesy of Dr. Yves Trousset).

References

[1] P. Besl and N. McKay. A method for Registration of 3D shapes. In *Pattern Analysis and Machine Intelligence*, pages 239–256, February 1992.

[2] I. Cohen, L.D. Cohen, and N. Ayache. Using deformable surfaces to segment 3-d images and infer differential structures. *Computer Vision, Graphics, and Image Processing: Image Understanding*, pages 242–263, 1992.

[3] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active shape models, their training and application. In *Computer Vision and Image Understanding*, volume 61, pages 38–59, January 1995.

[4] J. Declerck, J. Feldmar, M.L. Goris, and F. Betting. Automatic registration and alignement on a template of cardiac stress and rest spect images. In *Mathematical Methods in Biomedical Image Analysis*, pages 212–221, June 1996.

[5] H. Delingette. Simplex meshes: a general representation for 3d shape reconstruction. In *Proc. of Int. Conf. on Computer Vision and Pattern Recognition (CVPR'94)*, Seattle, USA, June 1994.

[6] H. Delingette. Simplex Meshes: a General Representation for 3D Shape Reconstruction. Technical Report 2214, INRIA, March 1994.

[7] H. Delingette. General Object Reconstruction based on Simplex Meshes. Technical Report 3111, INRIA, February 1997.

[8] J. Feldmar and N. Ayache. Locally affine registration of free-form surfaces. In *Computer Vision and Pattern Recognition*, 1994.

[9] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active Shape Models. *Intertnational Journal of Computer Vision*, 1:321–331, 1987.

[10] T. McInerney and D. Terzopoulos. A Finite Element Model for 3D Shape Reconstruction and Nonrigid Motion Tracking. In *International Conference on Computer Vision*, pages 518–523, 1993.

[11] D. Metaxas and D. Terzopoulos. Constrained Deformable Superquadrics and nonrigid Motion Tracking. In *Computer Vision and Pattern Recognition*, June 1991.

[12] J. Montagnat and H. Delingette. A Hybrid Framework for Surface Registration and Deformable Models. In *Computer Vision and Pattern Recognition, CVPR'97*, pages 1041–1046, San Juan, Puerto Rico, June 1997.

[13] C. Nastar and N. Ayache. Fast segmentation, tracking, and analysis of deformable objects. In *Proceedings of the Fourth International Conference on Computer Vision (ICCV 93)*, Berlin, May 1993. also in SPIE, Geometric Methods in Computer Vision, San-Diego, 1993.

[14] X. Pennec. *L'incertitude dans les Problèmes de Reconnaissance et de Recalage. Application en Imagerie Médicale et Biologie Moléculaire.* PhD thesis, Ecole Polytechnique, France, 1996.

[15] F. P. Preparata and M. I. Shamos. *Computational Geometry: an introduction Texts and monographs in computer science.* Springer-Verlag, 1990.

[16] G. Székely, A. Kelemen, Ch. Brechbüler, and G. Gerig. Segmentation of 2D and 3D objects from MRI volume data using constrained elastic deformations of flexible Fourier surface models. *Medical Image Analysis*, 1(1):19–34, July 1996.

[17] R. Szeliski. Fast Surface Interpolation Using Hierarchical Basis Functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 513–528, 1990.

[18] D. Terzopoulos and K. Fleisher. Deformable models. *Visual Computer*, 4:306–331, 1988.

[19] D. Terzopoulos and D. Metaxas. Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):703–714, July 1991.

[20] J-P Thirion. Non-Rigid Matching using Demons. In *Computer Vision and Pattern Recognition, CVPR'96*, San Francisco, California USA, June 1996.

[21] B.C. Vemuri and A. Radisavljevic. From Global to Local, a Continuum of Shape Models with Fractal. In *European Conference on Computer Vision*, 1993.

[22] Z. Zhang. Iterative point matching for registration of free -form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, December 1994.