Using Force Fields Derived from 3D Distance Maps for Inferring the Attitude of a 3D Rigid Object

Lionel Brunie[1] *and Stéphane Lavallée*[1] *and Richard Szeliski*[2]

[1] TIMC - IMAG, Faculté de Médecine de Grenoble
38 700 La Tronche, France, lionel@timb.imag.fr
[2] Digital Equipment Corporation, Cambridge Research Lab
One Kendall Square, Bldg. 700, Cambridge, MA 02139, szeliski@crl.dec.com

Abstract. This paper presents a new method for evaluating the spatial attitude (position-orientation) of a 3D object by matching a 3D static model of this object with sensorial data describing the scene (2D projections or 3D sparse coordinates). This method is based on the pre-computation of a *force field derived from 3D distance maps* designed to *attract any 3D point toward the surface* of the model. The attitude of the object is infered by *minimizing the energy necessary to bring all of the 3D points (or projection lines) in contact with the surface* (geometric configuration of the scene). To quickly and accurately compute the 3D distance maps, a precomputed distance map is represented using an *octree spline* whose resolution increases near the surface.

1 Introduction

One of the most basic ability of any human or artificial intelligence is the inference of knowledge by matching various pieces of information [1]. When only a few data are available, one can introduce *a priori* knowledge to compensate for the lack of information and match it with the data. In this latter frame, one of the most classical problematics is the *inference of the attitude of a 3D object from sensorial data* (2D projections or sparse 3D coordinates).

This problem can be formulated as follows: assume that we know a 3D description (model) or some features of an object in a first 3D attitude (location and orientation). We acquire various sensorial data describing this object in another (unknown) attitude, and we then attempt to estimate, from the model of the object and this new data, this unknown attitude. This generally implies the determination of 6 parameters: three components of translation (location) and three components of rotation(orientation).

In this paper, we will suppose the segmentation of the sensorial data achieved and focus on the interpretation of the scene described by the segmented images.

In spite of a considerable amount of litterature (see [2] for a review of related works), no general algorithm has been published yet. This paper presents a new complex object-oriented geometric method based on the pre-computation of a *force field derived from 3D distance maps*. Experimental results, in the field of computer-assisted surgery, are proposed.

2 Problem formulation : an energetic paradigm

To be independant from any 3D object representation and in order to have as wide an application field as possible, the start point of our matching process will be therefore a

* The research described in this paper is supported by DEC and Safir-Groupe Sem companies

set of 3D points distributed on the surface of the object and defining our model of the object. Such a model can be extracted from any 3D initial representation.

The problem is to estimate the transformation **T** between $\text{Ref}_{\text{sensor}}$ (the reference system of the sensorial data) and Ref_{3D} (reference system in which the 3D model of the object is defined). After the sensor calibration (*N-planes spline method* ([3]), in 3D/2D matching every pixel $\mathcal{P}_i$ of each projection is associated with a 3-D line, L_i, called *matching line*, whose representation is known in $\text{Ref}_{\text{sensor}}$.

In 3D/2D matching, when the 3D object is in its final attitude, **T**, every line L_i is tangent to the surface S. In the same way, when matching the 3D model with a set of sparse 3D control points, these latter are in contact with S. For sufficiently complex objects (i.e. without strong symmetries), **T** is the only attitude leading to such a geometric configuration. Our algorithm is based on this observation :

1. We first define the 3-D *unsigned distance* between a point **r** and the surface S, $d_E(\mathbf{r}, S)$, as the minimum Euclidean distance between **r** and all the points of S. We use this distance function to define a force field in any point of the 3D space. Every point **r** is associated with a force vector $\mathbf{F}(\mathbf{r}) = \mathbf{w} - \mathbf{r}$ where **w** is the point of S the closest to **r**. We therefore have:
$$|\mathbf{F}(\mathbf{r})| = d_E(\mathbf{r}, S) \tag{1}$$
2. In 3D/2D matching, an attraction force $\mathbf{FL}(L_i)$ is associated to any matching line L_i by:
   (a) if L_i does not cross S , $\mathbf{FL}(L_i) = \mathbf{F}(M_i)$ where M_i is the point of L_i the closest to S;
   (b) else $\mathbf{FL}(L_i) = \mathbf{F}(N_i)$ where N_i is the point of L_i inside the surface S the farthest from S (see fig. 1).

   A simple way to compute **FL** is to consider a *signed distance*, $\tilde{d}$, of same module than d_E, but negative inside S and to choose the point of L_i of minimum *signed* module. same module than

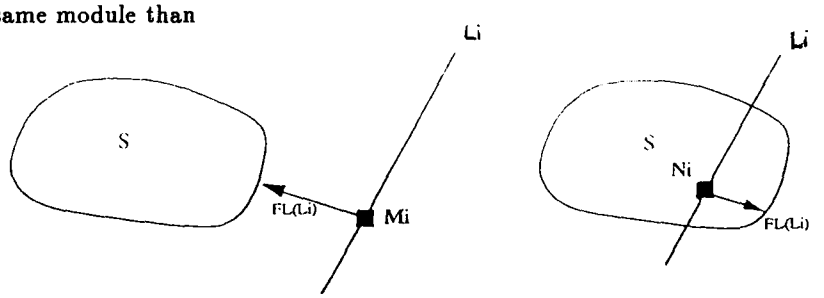

Fig. 1. Force vector associated to a matching line

3. **lemma** (not proved here) : The potential energy of the force field F at a point **r** with respect to the surface S i.e. the energy necessary to bring **r** in contact with S is
$$PE(\mathbf{r}) = \frac{1}{2}F(r)^2 + o(F(r)) \tag{2}$$
   For a set of N_q 3D control points, r_i, to take into account the reliability of the data, we introduce the variance of the noise of the measurement $d_E(r_i, S)$, σ_i^2, (see section 4) to weight the energy of a control point and consider the energy E :
$$E(\mathbf{p}) = \sum_{i=1}^{N_q} \frac{1}{\sigma_i^2}[d_E(r_i, S)]^2. \tag{3}$$

4. In the same way, the potential energy of a matching line L_i, i.e. the work necessary to bring the line into contact with S is equal to the potential energy of the point where the attraction force is applied (M_i or N_i). As previously, to take into account the reliability of the data on the matching lines, we weight the potential energy of each matching line by the variance of the noise of the measurement $\tilde{d}(l_i(\mathbf{p}), S)$, σ_i^2, and consider the energy E:

$$E(\mathbf{p}) = \sum_{i=1}^{M_P} \frac{1}{\sigma_i^2}[\mathbf{FL}(L_i)]^2 = \sum_{i=1}^{M_P} \frac{1}{\sigma_i^2}[\tilde{d}_l(l_i(\mathbf{p}), S)]^2. \quad (4)$$

5. As shown above, when the object is in its final attitude, every line (every control point) is in contact with S and the energy of the attitude is therefore zero, the lowest possible energy. If the object is sufficiently complex the minimum of the energy function is reached only once, in the final attitude, and the energy function is convex in a large neighborhood of this attitude. A minimization procedure of convex function can therefore be performed (see section 4).

3 Fast force field computation and octree splines distance maps

The method described in the previous section relies on the fast computation of the distances d_E and $\tilde{d}$. If the surface S is discretized in n^2 points, the computation of the distance d_E is a $O(n^2)$ process. Similarly, if a line $l_i(\mathbf{p})$ is discretized in m points, the computation of the distance $\tilde{d}$ is a $O(mn^2)$ process. To speed up this process, we precompute a 3-D *distance map*, which is a function that gives the signed minimum distance to S from any point $\mathbf{q}$ inside a bounding volume V that encloses S.

More precisely, let G a regular grid of N^3 points bounding V. We first compute and store the distance $\tilde{d}$ for each point $\mathbf{q}$ of G. Then $\tilde{d}(\mathbf{q}, S)$ can be computed for any point $\mathbf{q}$ using a trilinear interpolation of the 8 corner values $\tilde{d}_{ijk}$ of the cube that contains the point $\mathbf{q}$. If $(u, v, w) \in [0, 1] \times [0, 1] \times [0, 1])$ are the normalized coordinates of $\mathbf{q}$ in the cube,

$$\tilde{d}(\mathbf{q}, S) = \sum_{i=0}^{1}\sum_{j=0}^{1}\sum_{k=0}^{1} b_i(u)b_j(v)b_k(w)\tilde{d}_{ijk} \quad \text{with} \quad b_l(t) = \delta_l t + (1 - \delta_l)(1 - t). \quad (5)$$

We can compute the gradient $\nabla\tilde{d}(\mathbf{q}, S)$ of the signed distance function by simply differentiating (5) with respect to u, v, and w. Because $\tilde{d}$ is only C^0, $\nabla\tilde{d}(\mathbf{q}, S)$ is discontinuous on cube faces. However, these gradient discontinuities are relatively small and do not seem to affect the convergence of our iterative minimization algorithm.

In looking for an improved trade-off between memory space, accuracy, speed of computation, and speed of construction, we have developed a new kind of distance map which we call the *octree spline*. The intuitive idea behind this geometrical representation is to have more detailed information (i.e., more accuracy) near the surface than far away from it. We start with the classical octree representation associated with the surface S and then extend it to represent a continuous 3-D function that approximates the signed Euclidean distance to the surface. This representation combines advantages of adaptive spline functions and hierarchical data structures. For more details on the concept of octree-splines, see [2].

4 Least Squares Minimization

This section describes the nonlinear least squares minimization of the energy or error function $E(\mathbf{p})$ defined in eq. 4 and eq. 3.

Least squares techniques work well when we have many uncorrelated noisy measurements with a normal (Gaussian) distribution[3]. To begin with, we will make this assumption, even though noise actually comes from calibration errors, 2-D and 3-D segmentation errors, the approximation of the Euclidean distance by octree spline distance maps, and non-rigid displacement of the surface between Ref_{3D} and $\text{Ref}_{\text{sensor}}$.

To perform the nonlinear least squares minimization, we use the Levenberg-Marquardt algorithm because of its good convergence properties [4]. An important point of this method is that in both equations 4 and 3 $E(\mathbf{p})$ can be easily differentiated which allows to exhibit *simple analytical forms for the gradient and Hessian of* $E(\mathbf{p})$, used in the minimization algorithm.

At the end of the iterative minimization process, we compute a robust estimate of the parameter $\mathbf{p}$ by throwing out the measurements where $e_i^2(\mathbf{p}) \gg \sigma_i^2$ and performing some more iterations [5]. This process removes the influence of *outliers* which are likely to occur in the automatic 2-D and 3-D segmentation processes (for instance, a partially superimposed object on X-ray projections can lead to false contours).

Using a gradient descent technique such as Levenberg-Marquardt we might expect that the minimization would fail because of local minima in the 6-dimensional parameter space. However, for the experiments we have conducted, false local minima were few and always far away from the solution. So, with a correct initial estimate of the parameters, these other minima are unlikely to be reached.

Finally, at the end of the iterative minimization procedure, we estimate the uncertainty in the parameters (covariance matrix) to compute the distribution of errors after minimization in order to check that it is Gaussian.

5 Experimental results

We have performed tests on both real anatomical surfaces and on simulated surfaces. In 3D/2D matching, the projection curves of these surfaces were obtained by simulation in order to know the parameters $\mathbf{p}^*$ for which the correct pose is reached. Figures 2 and 3 show an example of convergence for an anatomical surface (VIM of the brain ; surface S_1) in 3D/2D matching. The state of the iterative minimization algorithm is displayed after 0, 2, and 6 iterations. Figure 2 shows the relative positions of the projections lines and the surface seen from a general viewpoint. Figure 3 shows the same state seen from the viewpoints of the two cameras (computation times expressed below are given for a DECstation 5000/200). Experiments have also been conducted to test this method for 3D/3D matching by simulating a complex transformation on a vertebra (surface S_2) (see fig. 4 for the convergence).

6 Discussion

In comparison with existing methods, the experiments we ran showed the method presented in this paper had five main advantages.

[3] Under these assumptions, the least squares criterion is equivalent to maximum likelihood estimation.

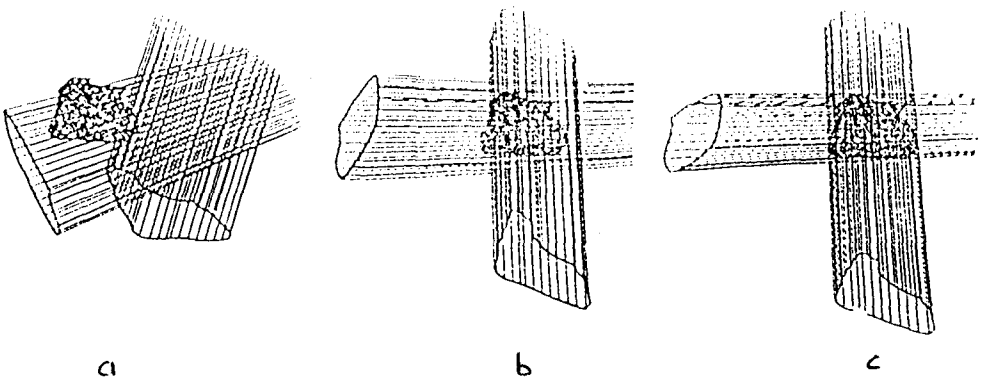

Fig. 2. Convergence of algorithm observed from a general viewpoint (surface S_1 is represented by a set of points). Two sets of projection lines evolve in the 3D potential field associated with the surface until each line is tangent to S_1: (a) initial configuration, (b) after 2 iterations, (c) after 6 iterations. For this case, the matching is performed in 1.8 s using 77 projection lines, in 0.9 s using 40 projection lines.

First, the matching process works for *any free-form smooth surface.* Second, we achieve *the best accuracy possible* for the estimation of the 6 parameters in p, because the octree spline representation we use approximates the true 3-D Euclidean distance with an error smaller than the segmentation errors in the input data. Third, we provide an *estimate of the uncertainties* of the 6 parameters. Fourth, we perform the matching process *very rapidly.* Fifth, in our method, *only a few pixels on the contours are needed.* This allows to estimate the attitude of the object even if it is *partially occluded.* Moreover, reliability factors can be introduced to weight the contribution of uncertain data (for instance, the variance of the segmentation can be taken into account).

This method could also be used for *recognition* problems, where the purpose is to match some contour projections with a finite set of 3-D objects $\{O_i\}$.

Researches are presently underway to adapt this algorithm to *non-segmented gray-levels images* by selecting potential matching lines, then assign credibility factors to them and maximize a matching energy.

References

1. A. Wackenheim. *Perception, commentaire et interprétation de l'image par les intelligences naturelle et artificielle.* Springer Verlag, 1987.
2. S. Lavallee and L. Szeliski, R. Brunie. Matching 3d smooth surfaces with their 2d projections using 3d distance maps. In *SPIE Geometric methods in CV*, San Diego, CA, July 1991.
3. G. Champleboux. *Utilisation des fonctions splines a la mise au point d'un capteur tridimensionnel sans contact : application a la ponction assistee par ordinateur.* PhD thesis, Grenoble University, July 1991.
4. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press, Cambridge, England, 1986.
5. P. J. Huber. *Robust Statistics.* John Wiley & Sons, New York, New York, 1981.

This article was processed using the LaTeX macro package with ECCV92 style

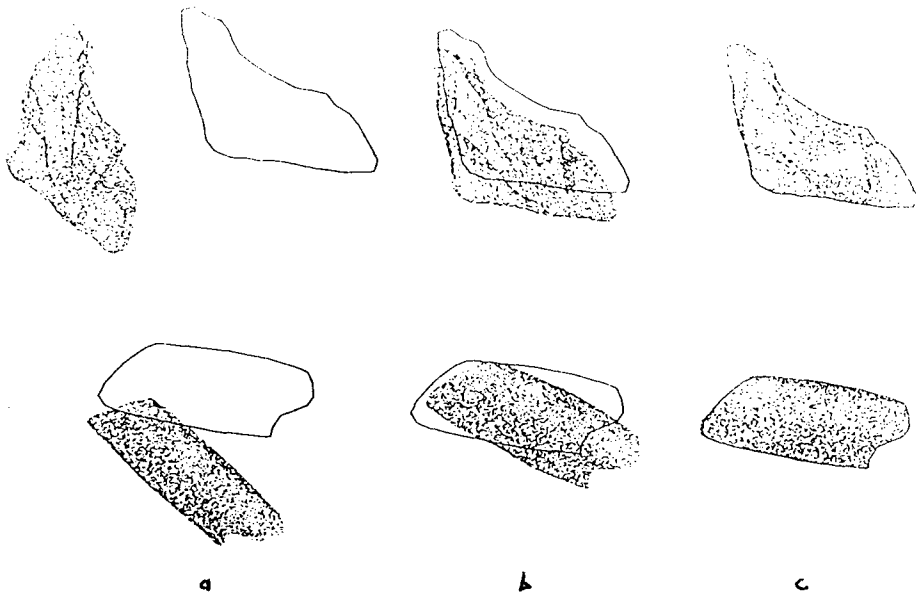

Fig. 3. Convergence of algorithm for surface S_1 observed from the 2 projection viewpoints. The external contours of the projected surface end up fitting the real contours: (a) initial configuration, (b) after 2 iterations (c) after 6 iterations.

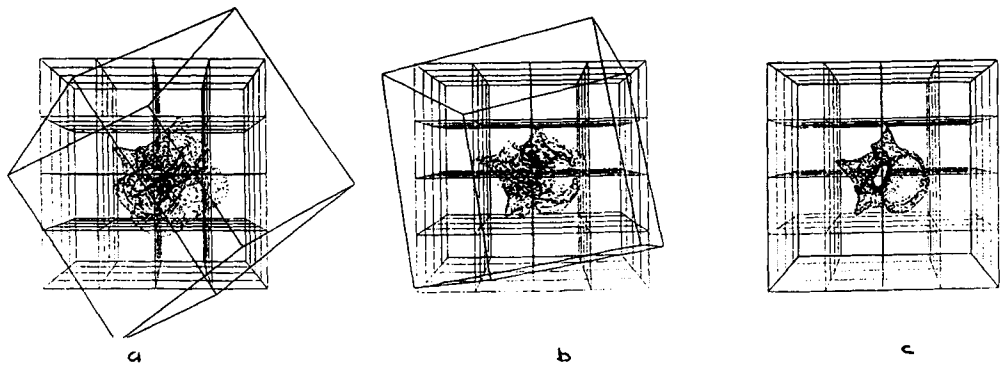

Fig. 4. Convergence of 3-D/3-D matching algorithm for surface S_3 (vertebra) segmented from a 3D CT image. For this case, the matching is performed in 2 s using 130 data points.

(a) initial configuration, $E(\mathbf{p}^{(0)})/M_P = 113.47$, $\|\Delta \mathbf{t}^{(0)}\| = 125.23$mm, $|\Delta\alpha^{(0)}| = 48.25°$,
(b) after 2 iterations, $E(\mathbf{p}^{(2)})/M_P = 38.58$, $\|\Delta \mathbf{t}^{(2)}\| = 25.97$mm, $|\Delta\alpha^{(2)}| = 20.53°$,
(c) after 6 iterations. $E(\mathbf{p}^{(6)})/M_P = 4.20$, $\|\Delta \mathbf{t}^{(6)}\| = 0.75$mm, $|\Delta\alpha^{(6)}| = 0.32°$.