Level set methods and the Stereo Problem

Olivier Faugeras[1*] and Renaud Keriven[2]

[1] INRIA, Sophia-Antipolis, France and MIT AI-Lab, USA
[2] CERMICS - ENPC, France

Abstract. We present a novel geometric approach for solving the stereo problem for an arbitrary number of images (greater than or equal to 2). It is based upon the definition of a variational principle that must be satisfied by the surfaces of the objects in the scene and their images. The Euler-Lagrange equations which are deduced from the variational principle provide a set of PDE's which are used to deform an initial set of surfaces which then move towards the objects to be detected. The level set implementation of these PDE's potentially provides an efficient and robust way of achieving the surface evolution and to deal automatically with changes in the surface topology during the deformation, i.e. to deal with multiple objects. Results of a two dimensional implementation of our theory are presented on synthetic and real images.

1 Introduction and preliminaries

The idea that is put forward in this paper is that the methods of curve and surface evolutions which have been developed in computer vision under the name of snakes [15] and then reformulated by Caselles, Kimmel and Sapiro [2] and Kichenassamy et al. [16] in the context of PDE driven evolving curves can be used effectively for solving 3D vision problems such as stereo and motion analysis.

As a first step in this direction we present a mathematical analysis of the stereo problem in this context as well as a partial implementation.

The problem of curve evolution driven by a PDE has been recently studied both from the theoretical standpoint [9,10,20] and from the viewpoint of implementation [17,22,23] with the development of level set methods that can efficiently and robustly solve those PDE's. A nice recent exposition of the level set methods and of many of their applications can be found in [21].

The problem of surface evolution has been less touched upon even though some preliminary results have been obtained [23,3].

Stereo is a problem that has received considerable attention for decades in the psychophysical, neurophysiological and, more recently, in the computer vision literatures. It is impossible to cite all the published work here, we will simply refer the reader to some basic books on the subject [14,11–13,6]. To explain the problem of stereo from the computational standpoint, we

* This work was partially supported by the Esprit grant Improofs, Nr. 23515, LTR domain.

will refer the reader to Fig. 1a. Two, may be more, images of the world are taken simultaneously. The problem is, given those images, to recover the geometry of the scene. Given the fact that the relative positions and orientations and the internal parameters of the cameras are known which we will assume in this article (the cameras are then said to be calibrated [6]), the problem is essentially (but not only) one of establishing correspondences between the views: one talks about the matching problem. The matching problem is usually solved by setting up a matching functional for which one then tries to find extrema. Once a pixel in view i has been identified as being the image of the same scene point as another pixel in view j, the 3D point can then be reconstructed by intersecting the corresponding optical rays (see Fig. 1a again).

In order to go any further, we need to be a little more specific about the process of image formation. We will assume here that the cameras perform a perpective projection of the 3D world on the retinal plane as shown in Fig. 1b The optical center, noted C in the figure, is the center of projection and the image of the 3D point M is the pixel m at the intersection of the optical ray $\langle C, m \rangle$ and the retinal plane $\mathcal{R}$. As described in many recent papers in computer vision, this operation can be conveniently described in projective geometry by a matrix operation. The projective coordinates of the pixel m (a 3×1 vector) are obtained by applying a 3×4 matrix $\mathbf{P}_1$ to the projective coordinates of the 3D point M (a 4×1 vector). This matrix is called the perspective projection matrix. If we express the matrix $\mathbf{P}_1$ in the coordinate system (C, x, y, z) shown in the Fig. 1b, it then takes a very simple form:

$$\mathbf{P}_1 = [\mathbf{I}_3 \, \mathbf{0}]$$

where $\mathbf{I}_3$ is the 3×3 identity matrix. If we now move the camera by applying to it a rigid transformation described by the rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$, the expression of the matrix $\mathbf{P}$ changes accordingly and becomes:

$$\mathbf{P}_2 = [\mathbf{R}^T \; -\mathbf{R}^T\mathbf{t}]$$

Let us decide on some definitions and notations. Images are denoted by I_k, k taking some integer values which indicate the camera with which the image has been acquired. They are considered as smooth (i.e. C^2, twice continuously differentiable) functions of pixels m_k whose coordinates are defined in some orthonormal image coordinate systems (x_k, y_k) which are assumed to be known. We note $I_k(m_k)$ or $I_k(x_k, y_k)$ the intensity value in image k at pixel m_k.

The pixels in the images are considered as functions of the 3D geometry of the scene, i.e. of some 3D point M on the surface of an object in the scene, and of the unit normal vector $\mathbf{N}$ to this surface.

Our approach is an extension of previous work by Robert et al., Robert and Deriche, [19,18], and Deriche et al. [4]. For a comparison with the work of those authors see [7].

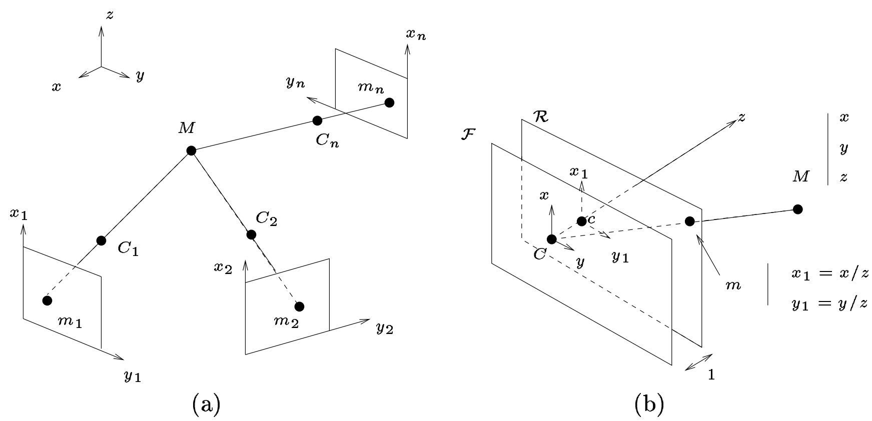

Fig. 1. (a) The multicamera stereo vision problem is, given a pixel m_1 in image 1, to find the corresponding pixel m_2 in image 2, ... , the corresponding pixel m_n in image n, i.e. the ones which are the images of the same 3D point M. Once such a correspondence has been established, the point M can be reconstructed by intersecting the optical rays $\langle m_i, C_i \rangle$, $i = 1, \cdots, n$. (b) The focal plane (x, y) is parallel to the retinal plane (x_1, y_1) and at a distance of 1 from it.

2 The object and matching models

In this section we present the way we model the objects in the scene and the way we match their images.

Objects are modelled as the zero level set of a function $\hat{u} : \mathbf{R}^3 \rightarrow \mathbf{R}$ which we assume to be smooth, i.e. C^2. The coordinates (x, y, z) of the points in the scene which are on the surface of the objects present are thus defined by the equation $\hat{u}(x, y, z) = 0$. This approach has at least two advantages. First, it potentially allows us to use an arbitrary number of cameras to analyze the scene and second, it leads very naturally to an implementation of a surface evolution scheme through the level set method as follows.

Let us consider a family of smooth surfaces $S : (v, w, t) \rightarrow \mathbf{S}(v, w, t)$ where (v, w) parametrize the surface and t is the time. It is in general not possible to find a single mapping S from $\mathbf{R}^2$ to $\mathbf{R}^3$ that describes the entire surface of the objects (think of the sphere for example where we need at least two) but we do not have to worry about this since our results will in fact be independent of the parametrization we choose. The objects in the scene correspond to a surface $\hat{\mathbf{S}}(v, w)$ and our goal is, starting from an initial surface $\mathbf{S}_0(v, w)$, to derive a partial differential equation

$$\mathbf{S}_t = \beta \mathbf{N}, \tag{1}$$

where $\mathbf{N}$ is the unit normal to the surface, which, when solved with initial conditions $\mathbf{S}(v, w, 0) = \mathbf{S}_0(v, w)$, will yield a solution that closely approximates $\hat{\mathbf{S}}(v, w)$. The function β is determined by the matching functional that we minimize in order to solve the stereo problem. We define such a functional in the next paragraph. An interesting point is that the evolution equation (1) can be solved using the level set method which has the advantage of coping automatically with several objects in the scene. In detail, the surfaces $\mathbf{S}$ are at each time instant the zero level sets of a function $u : \mathbf{R}^4 \rightarrow \mathbf{R}$:

$$u(\mathbf{S}, t) = 0$$

Taking derivatives with respect to u, v, t, noticing that $\mathbf{N}$ can be chosen such that $\mathbf{N} = -\frac{\nabla u}{|\nabla u|}$, where ∇ is the gradient operator for the first three coordinates of u, one finds easily that the evolution equation for u is:

$$u_t = \beta \mid \nabla u \mid \tag{2}$$

We now define our matching criterion. It is basically a modified version of the standard cross-correlation criterion which takes into account the orientation of the tangent plane to the object. It is known [7,6] that a plane induces a collineation between two retinal planes. The correlation window in image 2 is therefore the image by the collineation induced by the tangent plane of the rectangular window in image 1 (see Fig. 2). This collineation is a function of the point M and of the normal to the object at M. It is therefore a function of S and $\mathbf{N}$ that we denote by K. It satisfies the condition $K(m_1) = m_2$. We can define an inner product of the two images I_1 and I_2 as follows:

$$\langle I_1, I_2 \rangle(S, \mathbf{N}, x, y) = \frac{1}{4pq} \int_{-p}^{+p} \int_{-q}^{+q} (I_1(m_1 + m) - \overline{I_1}(m_1))(I_2(K(m_1 + m)) - \overline{I_2}(m_2))dm, \tag{3}$$

Note that the definition of $\langle I_1, I_2 \rangle$ is not symmetric, because of K. It is possible to make it symmetric [7]. We note $\mid I \mid^2$ the quantity $\langle I, I \rangle$. $\overline{I_1}$ and $\overline{I_2}$ are the means of I_1 and I_2 which are classically defined as:

$$\overline{I_k}(m_k) = \frac{1}{4pq} \int_{-p}^{+p} \int_{-q}^{+q} I_k(m_k + m')dm' \quad k = 1, 2 \tag{4}$$

Note that the definition (4) of $\overline{I_2}$ should be modified to take K into account. We do not do it here for lack of space, but see [7], and because it does not modify the fundamental ideas exposed in this paper but makes the computations significantly more complex.

We can now define the following error measure:

$$C(\mathbf{S}, \mathbf{N}) = - \sum_{i,j=1, i \neq j}^{n} \int \int \frac{1}{\mid I_i \mid \cdot \mid I_j \mid} \langle I_i, I_j \rangle d\sigma = \int \int \Phi(\mathbf{S}, \mathbf{N}, v, w) d\sigma \tag{5}$$

In this equation, the indexes i and j range from 1 to n, the number of views. In practice it is often not necessary to consider all possible pairs but it does not change our analysis of the problem. In (5), the integration is carried over with respect to the area element $d\sigma$ on the surface S rather than wit respect to the image coordinates. With the previous notations, we have

$$d\sigma = \mid \mathbf{S}_v \times \mathbf{S}_w \mid dvdw$$

The presence of the term $\mid \mathbf{S}_v \times \mathbf{S}_w \mid$ has two dramatic consequences:

1. It automatically regularizes the variational problem like in the geodesic snakes approach [2], and
2. it makes the problem intrinsic, i.e. independent of the parametrization of the objects in the scene.

Note also that each integral that appears in (5) is only computed for those points of the surface S which are visible in the two concerned images. Thus, *visibility and occlusion* are modelled in this approach (see Sect. 3 for more details).

The rest of the derivation proceeds as usual, we write the Euler-Lagrange equations of the variational problem (5), consider their component β along the normal to the surface, set up a surface evolution equation (1) and implement it by a level-set method. This is all pretty straightforward except for the announced result that the resulting value of β is *intrinsic* and does not depend upon the parametrization of the surface S.

We in fact prove in [7] a more general result. Let $\Phi : \mathbf{R}^3 \times \mathbf{R}^3 \longrightarrow \mathbf{R}$ be a smooth function of class at least C^2 defined on the surface S and depending upon the point $\mathbf{S}(v, w)$ and the unit normal $\mathbf{N}(v, w)$ at this point, which we denote by $\Phi(\mathbf{X}, \mathbf{N})$. Let us now consider the following error measure:

$$C(\mathbf{S}, \mathbf{S}_v, \mathbf{S}_w) = \int\int \Phi(\mathbf{S}(v, w),\ \mathbf{N}(v, w))h(v, w)dv\,dw \tag{6}$$

where the integral is taken over the surface S and $h(v, w) = \mid \mathbf{S}_v \times \mathbf{S}_w \mid$.

We prove in [7] the following theorem:

Theorem 1. *Under the assumptions of smoothness that have been made for the function Φ and the surface S, the component of the Euler-Lagrange equations for criterion (6) along the normal to the surface is the product of h with an intrinsic factor, i.e. which does not depend upon the parametrization (v, w). Furthermore, this component is equal to*

$$h(\Phi_{\mathbf{X}}\mathbf{N} - 2H(\Phi - \Phi_{\mathbf{N}}\mathbf{N}) + Trace((\Phi_{\mathbf{XN}})_{T_S} + d\mathbf{N} \circ (\Phi_{\mathbf{NN}})_{T_S})) \tag{7}$$

where all quantities are evaluated at the point $\mathbf{S}$ of normal $\mathbf{N}$ of the surface, T_S is the tangent plane to the surface at the point $\mathbf{S}$. $d\mathbf{N}$ is the differential of the Gauss map of the surface, H is its mean curvature, $\Phi_{\mathbf{XN}}$ and $\Phi_{\mathbf{NN}}$ are the second order derivatives of Φ, $(\Phi_{\mathbf{XN}})_{T_S}$ and $(\Phi_{\mathbf{NN}})_{T_S}$ their restrictions to the tangent plane T_S of the surface at the point S.

Note that the error criterion (5) is of the form (6) if we define Φ to be

$$-\sum_{i,j=1,i\neq j}^{n} \frac{1}{\mid I_i \mid \cdot \mid I_j \mid} \langle I_i, I_j \rangle$$

According to the theorem 1, in order to compute the velocity β along the normal in the evolution equations (1) or (2) we only need to compute $\Phi_{\mathbf{S}}$, $\Phi_{\mathbf{N}}$, $\Phi_{\mathbf{SN}}$ and $\Phi_{\mathbf{NN}}$ as well as the second order intrinsic differential properties of the surface S. Using the fact that the function Φ is a sum of functions $\Phi_{ij} = -\frac{1}{|I_i|\cdot|I_j|}\langle I_i, I_j\rangle$, the problem is broken down into the problem of computing the corresponding derivatives of the Φ_{ij}'s. The results and proofs can be found in [7].

In terms of the level set implementation, we ought to make a few remarks. The first is to explain how we compute β in (2) at each point (x, y, z) rather than on the surface S. It should be clear that we do not have any problem for computing $\mathbf{N} = -\frac{\nabla u}{|\nabla u|}$ and $2H = div(\frac{\nabla u}{|\nabla u|})$ and $d\mathbf{N}$ which is the differential of the Gauss map of the level set surface going through the point (x, y, z). The vectors $\Phi_{\mathbf{X}}$, $\Phi_{\mathbf{N}}$, the matrices $\Phi_{\mathbf{XN}}$, $\Phi_{\mathbf{NN}}$ are computed as explained in [7].

The second remark is that we can now write (2) as follows:

$$\begin{aligned} u_t =\mid \nabla u \mid div(\Phi \frac{\nabla u}{\mid \nabla u \mid}) - \Phi_{\mathbf{N}}(D\mathbf{N} + trace(D\mathbf{N})\mathbf{I}_3)\nabla u \\ - Trace((\Phi_{\mathbf{XN}})_{T_S} + d\mathbf{N} \circ (\Phi_{\mathbf{NN}})_{T_S}) \mid \nabla u \mid \end{aligned} \tag{8}$$

where $D\mathbf{N}$ is the 3×3 matrix of the derivatives of the normal with respect to the space coordinates, $\mathbf{I}_3$ the identity matrix, and at each point (x, y, z) the tangent plane T_S is that of the level set surface $u = constant$ going through that point. Note that $trace(D\mathbf{N}) = -div(\frac{\nabla u}{|\nabla u|})$.

The first term $\mid \nabla u \mid div(\Phi \frac{\nabla u}{|\nabla u|})$ is identical to the one in the work of Caselles, Kimmel, Sapiro and Sbert [3] on the use of minimal surfaces or geodesic snakes to segment volumetric images. Our other terms come from the particular process that we are modelling, i.e. stereo. We believe and hope that we can prove in the near future that, under some reasonable assumptions, (8) is well-posed. As a first step in that direction, we report on some important implementation details:

1. Near the solution, Φ is close to -1 and the term $\mid \nabla u \mid div(\Phi \frac{\nabla u}{|\nabla u|})$ becomes anti-diffusive! As a consequence, we used $\Phi' = \Phi + 1$ instead of Φ,which takes values between 0 and $+2$, as a new error measure. This is equivalent to introducing in the criterion a term that tends to minimize the total area of the objects.
2. Regarding the image smoothness assumptions, a Gaussian modulated correlation could be used [8], although we did not have to use it in our 2D implementation (see Sect. 3).

3. Concerning the problems of visibility and occlusion, the total error measure C_4 assumes the choice of certain camera pairs. Due to lack of place, we will not go into the details and just say that our 2D implementation handles this problem such that C_4 is at least continuous.

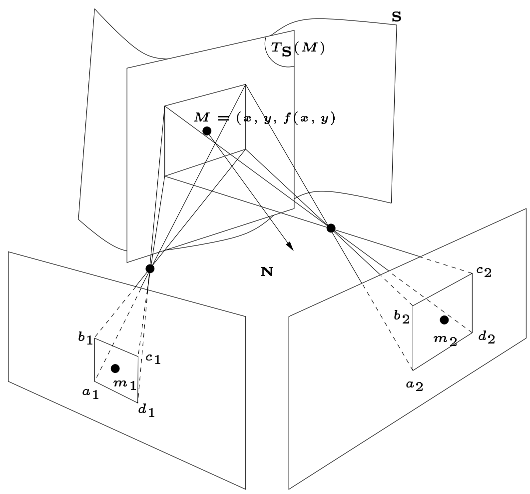

Fig. 2. The square window (a_1, b_1, c_1, d_1) in the first image is back projected onto the tangent plane to the object S at point M and reprojected in the retinal plane of the second camera where it is generally not square. The observation is that the distortion between (a_1, b_1, c_1, d_1) and (a_2, b_2, c_2, d_2) can be described by a collineation which is function of M and the normal $\mathbf{N}$ to the surface of the object.

3 Two-dimensional implementation

We now present the results of an implementation in two dimensions of the theory that has been described in the previous section. The situation is described in Fig. 3 which depicts a planar object S with two connected components. This object is observed by the five cameras numbered from 1 to 5 and the level set corresponding to some value k has been drawn to illustrate how the visibility condition is taken into account when solving (8): the point M on the level set is not seen by the cameras 3 to 5 but only by 1 and 2.

Planar objects are viewed by linear cameras. Pixels and images are functions of only one variable.

The inner product (3) is:

$$\langle I_i, I_j\rangle(\mathbf{S}, \mathbf{N}) = \frac{1}{2p}\int_{-p}^{p}(I_i(m_i+m) - \overline{I_i}(m_i))(I_j(K_{ij}(m_i+m)) - \overline{I_j}(m_j))dm$$

We use the affine approximation: $K_{ij}(m_i + m) = m_j + \alpha_{ij} m$. The surface $\mathbf{S}(v, w)$ is a plane curve $\mathbf{S}(v)$ and the error measure:

$$\int \Phi(\mathbf{S}(v), \mathbf{N}(v)) h(v) dv$$

where $h(v) = |\mathbf{S}_v|$ and $\Phi = \sum_{(i,j) \in S_{\mathrm{cam}}(v)} \lambda_{ij}(v)(1 - \frac{1}{|I_i| \cdot |I_j|} \langle I_i, I_j \rangle)$ ($S_{\mathrm{cam}}(v)$ is the set of camera pairs used at v and $\lambda_{ij}(v)$ scale factors choosen such that Φ is at least continuous). Let σ be the arc-length parameter of $\mathbf{S}$ ($d\sigma = h(v)dv$), $\mathbf{T}$ the unit tangent, and κ its curvature, the normal component β of the Euler-Lagrange equation simplifies to the intrinsic quantity:

$$\beta = \kappa \Phi + [\Phi_{\mathbf{S}} + \kappa(\mathbf{T}\mathbf{T}^T - \mathbf{N}\mathbf{N}^T)\Phi_{\mathbf{N}}^T] \cdot \mathbf{N} + \mathbf{T}^T(\Phi_{\mathbf{SN}} + \kappa \Phi_{\mathbf{NN}})\mathbf{T}$$

In order to implement the evolution equation (2) of $u(x, y, t)$, the following steps are required to get β at point $\mathbf{M} = (x, y)$ at time t:

- Considering the level set $\mathbf{S}(v)$ of u passing through point $\mathbf{M}$, determine from which cameras is $\mathbf{M}$ visible, let us call them the S-cameras. They are determined by assuming that the level curve going through the point is opaque.
- Compute the normal $\mathbf{N}$ and the curvature κ of $\mathbf{S}$ at $\mathbf{M}$.
- For each S-camera, compute m_i, $\frac{\partial m_i}{\partial \mathbf{S}}$ and $\frac{d}{d\sigma} m_i$.
- For each pair of S-cameras, compute $\alpha_{ij}, \alpha_{ij\mathbf{S}}, \alpha_{ij\mathbf{N}}, \frac{d}{d\sigma}\alpha_{ij}$, and $\frac{d}{d\sigma}\alpha_{ij\mathbf{N}}$.
- Compute $\langle I_i, I_j \rangle, \langle I_i, I_j \rangle_{\mathbf{S}}, \langle I_i, I_j \rangle_{\mathbf{N}}, \frac{d}{d\sigma}\langle I_i, I_j \rangle$ and $\frac{d}{d\sigma}\langle I_i, I_j \rangle_{\mathbf{N}}$
- Compute $\Phi, \Phi_{\mathbf{S}}, \Phi_{\mathbf{N}}$ and $\frac{d}{d\sigma}\Phi_{\mathbf{N}}$ hence β.

This scheme has been tested with synthetic noisy images of 2D objects viewed from a number of cameras located around them and on a number of real images. We first present our results on three synthetic objects, each one being meant to demonstrate one feature of the algorithm. All those results have been obtained with 18 cameras observing the scene and are presented in an homogeneous manner: on the left hand side of the figure we show the views from some of the cameras, on the right hand side we show the convergence of the zero level set.

We start with Fig. 4 which demonstrates that the algorithm works for non convex objects. Notice that the background is dark. Figure 5 shows the stereovision process for two circles viewed on a random background. This example shows that our algorithm can deal with multiple objects (note the change in topology at time $t2$) and can cope with a textured background without being fooled.

Figure 6 shows the results for two squares. This example shows that our modelling can cope somewhat with non smooth objects.

Figure 7 shows a real example in the case of two cameras. The stereo pair of a human face is shown on the left hand side of the figure, the trace of the epipolar plane (vertical in this case) being shown. On the right hand side of

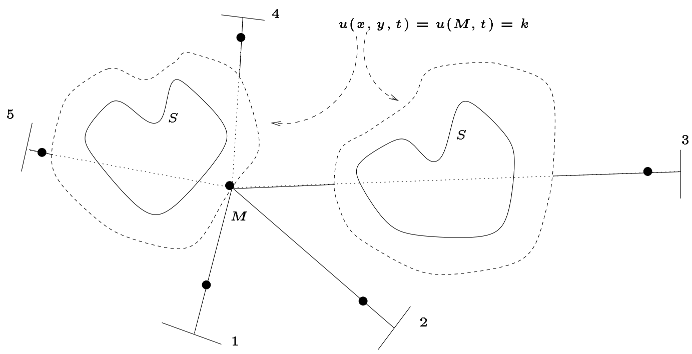

Fig. 3. The 2D implementation of the algorithm described in the previous section.

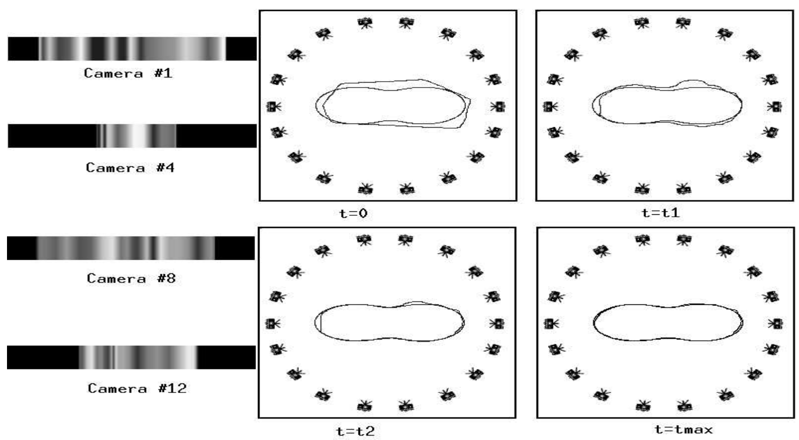

Fig. 4. 2D case – Detection of a non convex object.

the figure we show the evolution of the zero level set at four time instants. One curve is the result of the correlation algorithm described in [5] while the other one is the level set. It can be seen that the convergence is satisfactory.

4 Conclusion

We have presented a novel geometric approach for solving the stereo problem from an arbitrary number of views. It is based upon writing a variational principle that must be satisfied by the surfaces of the objects to be detected. The design of the variational principle allows us to clearly incorporate the hypotheses we make about the objects in the scene and how we obtain correspondences between image points. The Euler-Lagrange equations which are

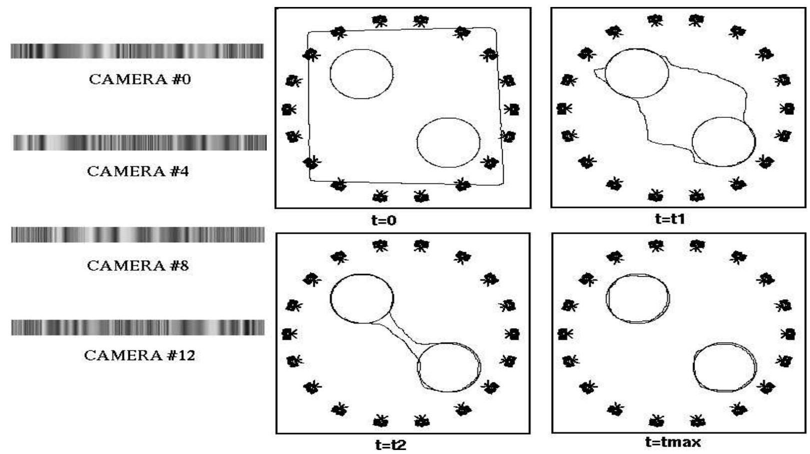

Fig. 5. 2D case – Detection of two circular objects on a random background.

deduced from the variational principle provide a set of PDE's which are used to deform an initial set of surfaces which then move towards the objects to be detected. The level set implementation of these PDE's potentially provides an efficient and robust way of achieving the surface evolution and to deal automatically with changes in the surface topology during the deformation.

Our implementation is so far only two-dimensional, i.e. it works only in epipolar planes but the results we have obtained look promising enough to lead us into thinking that the approach will also be successful in 3D.

To finish on a methodological note, we believe that it is by working in such well defined conceptual or mathematical frameworks, as the calculus of variations or the theory of PDE's, where the tools exist to prove (or disprove) the correctness of algorithms, that we will be able to bring computer vision up to the level of predictability where it can be used reliably in real applications and interfaced to other components to build complex systems.

References

1. Boston, MA, June 1995. IEEE Computer Society Press.
2. V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *Proceedings of the 5th International Conference on Computer Vision* [1], pages 694–699.
3. V. Caselles, R. Kimmel, G. Sapiro, and C. Sbert. 3d active contours. In M-O. Berger, R. Deriche, I. Herlin, J. Jaffre, and J-M. Morel, editors, *Images, Wavelets and PDEs*, volume 219 of *Lecture Notes in Control and Information Sciences*, pages 43–49. Springer, June 1996.
4. Rachid Deriche, Stéphane Bouvin, and Olivier Faugeras. A level-set approach for stereo. In *Fisrt Annual Symposium on Enabling Technologies for Law Enforcement and Security - SPIE Conference 2942 : Investigative Image Processing.*, Boston, Massachusetts USA, November 1996.

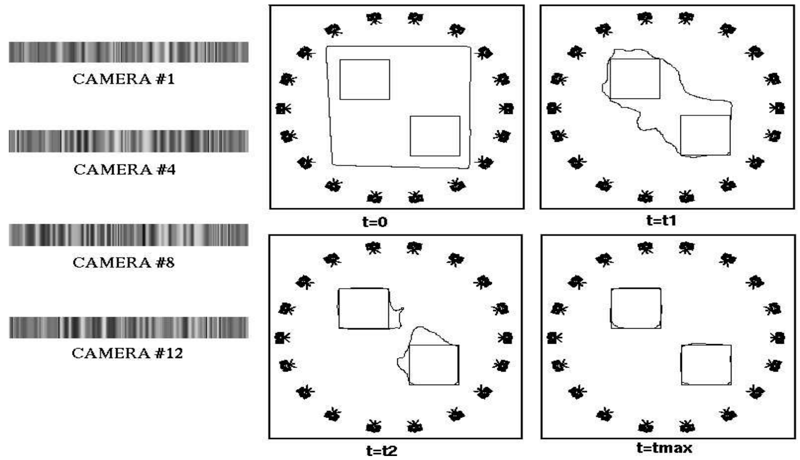

Fig. 6. 2D case – Detection of two squares.

5. Frédéric Devernay and Olivier Faugeras. Computing differential properties of 3-D shapes from stereoscopic images without 3-D models. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 208–213, Seattle, WA, June 1994. IEEE.
6. Olivier Faugeras. *Three-Dimensional Computer Vision: a Geometric Viewpoint*. MIT Press, 1993.
7. Olivier Faugeras and Renaud Keriven. Variational principles, surface evolution, pde's, level set methods and the stereo problem. Technical Report 3021, INRIA, November 1996.
8. L.M.J. Florack. *The syntactical structure of scalar images*. PhD thesis, Utrecht University, 1993.
9. M. Gage and R.S. Hamilton. The heat equation shrinking convex plane curves. *J. of Differential Geometry*, 23:69–96, 1986.
10. M. Grayson. The heat equation shrinks embedded plane curves to round points. *J. of Differential Geometry*, 26:285–314, 1987.
11. W.E.L. Grimson. *From Images to Surfaces*. MIT Press : Cambridge, 1981.
12. H. L. F. von Helmholtz. *Treatise on Physiological Optics*. New York: Dover, 1925. Translated by J.P. Southall.
13. Berthold Klaus Paul Horn. *Robot Vision*. MIT Press, 1986.
14. Bela Julesz. *Foundations of Cyclopean perception*. The University of Chicago Press, Chicago and London, 1971.
15. M. Kass, A. Witkin, and D. Terzopoulos. SNAKES: Active contour models. *The International Journal of Computer Vision*, 1:321–332, January 1988.
16. S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi. Gradient flows and geometric active contour models. In *Proc. Fifth International Conference on Computer Vision* [1].
17. S. Osher and J. Sethian. Fronts propagating with curvature dependent speed : algorithms based on the Hamilton-Jacobi formulation. *Journal of Computational Physics*, 79:12–49, 1988.

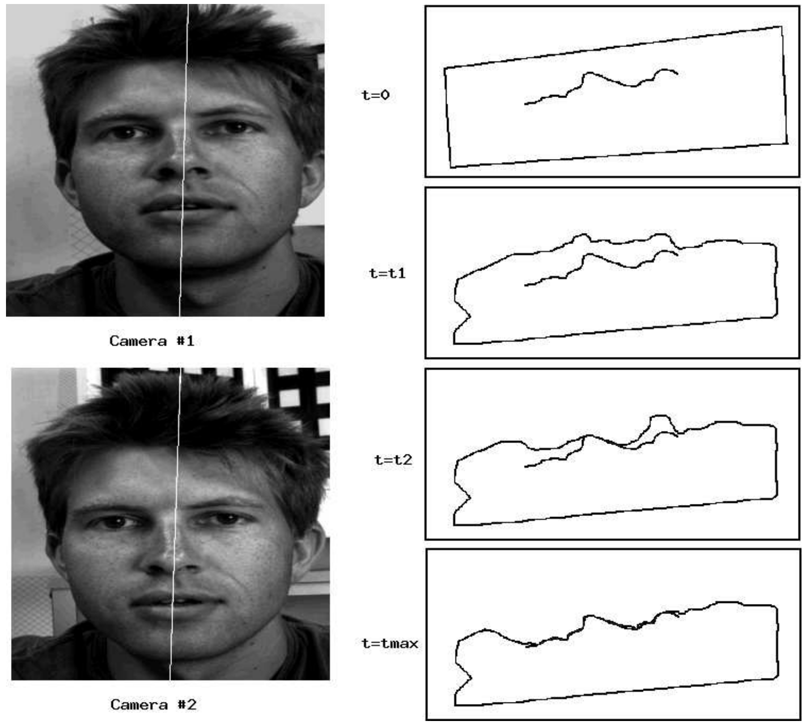

Fig. 7. 2D case – Detection of a human face in an epipolar plane.

18. L. Robert and R. Deriche. Dense depth map reconstruction: A minimization and regularization approach which preserves discontinuities. In Bernard Buxton, editor, *Proceedings of the 4th European Conference on Computer Vision*, Cambridge, UK, April 1996.
19. L. Robert, R. Deriche, and O.D. Faugeras. Dense depth recovery from stereo images. In *Proceedings of the European Conference on Artificial Intelligence*, pages 821–823, Vienna, Austria, August 1992.
20. Guillermo Sapiro and Allen Tannenbaum. Affine Invariant Scale Space. *The International Journal of Computer Vision*, 11(1):25–44, August 1993.
21. J. A. Sethian. *Level Set Methods.* Cambridge University Press, 1996.
22. J.A. Sethian. Numerical algorithms for propagating interfaces: Hamilton-jacobi equations and conservation laws. *Journal of Differential Geometry*, 31:131–161, 1990.
23. J.A. Sethian. Theory, algorithms, and applications of level set methods for propagating interfaces. Technical Report PAM-651, Center for Pure and Applied Mathematics, University of California, Berkeley, August 1995. To appear Acta Numerica.