A Snake Based Tracking Tool For 3D Reconstruction: Snake Strategies For Tracking The Right Contour

Marie-Odile Berger
INRIA Lorraine/CRIN – CNRS
Campus scientifique, B.P. 239
54506 Vandœuvre-les-Nancy CEDEX, France
email: berger@loria.fr

Abstract: *This paper addresses the problem of rigid and non-polyhedral object tracking with a view to reconstructing 3D objects, without a priori knowledge on objects and on the camera motion. The outlines of our algorithm have been previously described in [4]. It is made up of a prediction step and of an identification step based on active contour models. In this paper we focus on the identification stage and we propose new snake strategies to be sure that the snake has reached the homologous contour.*

1 Introduction

Our interest lies in modeling of non polyhedral scenes from observations taken by a camera whose motion is unknown. In order to recover the camera motion and to recover the 3D structure of the objects from occluding contours, an efficient tracking tool capable of tracking rigid curves as well as occluding contours is therefore essential. This task must be performed without any knowledge on the motion, nor on the 3D object shape. The tracking tool we have designed is based on the active contour models and we will firstly justify this choice.

Most tracking algorithms are made up of two parts: a prediction step first allows the velocity field to be computed more or less precisely. Then, the homologous contour is searched for in the vicinity of the prediction on the basis of shape similarity. This implies that the less the contour has salient characteristics, the more accurate the motion estimation must be computed.

Concerning our application, since the contours to be tracked generally do not have salient characteristics, the use of active contour models appears as an interesting tracking way since their deformation capabilities allow them to converge towards the homologous contour even if the initialization is far from the contour. Thus, a rough estimation of the velocity field is only required and this makes the prediction step easier and quicker.

1.1 Specific snake based tracking problems

The idea of tracking objects in a sequence using snakes has been originally proposed in [7]. Since then, numerous works have demonstrated that snakes are well suited for tracking rigid or non rigid objects [10, 8, 6, 4, 1, 9]. Among these works, we must distinguish those which operate in a simple context [8, 6] (tracking a strong or isolated contour on a dark background in order to make easier the feature ground detection) from those operating in complex environments [4, 1].

In order tracking can be performed successfully in complex environments, two important problems must be solved during the identification stage: mismatching problems and accuracy problems.

Mismatching

The main difficulty with snake based tracking is to avoid **mismatching**: this means that we have to develop means to ensure that the snake will converge towards the right contour. This difficulty stems from the fact that the snake evolution is in some sense blind since it converges towards the nearest edge. In fact, the initialization contains all a priori information available on the contour. Mismatching not only arises if the prediction is too far from the contour to be tracked but also if the image is complex: if there exists another contour close to the contour to be tracked, the snake may be attracted by this spurious contour even if the prediction is quite acceptable.

Solutions generally used consist in reducing the deformation capabilities of the snake so that it converges towards a curve whose shape is as similar as possible to the initialization, and which is probably the homologous contour [10, 6, 4, 1]. Nevertheless, the only use of rigidification cannot solve all mismatching problems, especially when there exists a strong contour close to the contour to be tracked (Fig. 2.b). Thus, if some properties of the searched contour (such as the gradient profile) are not available, such failure cannot be detected and, of course, not corrected.

However, in the tracking context, the snake can be assessed to a large extent because the intensity profile does not vary too much from frame to frame. One of the contribution of this paper is to show that such an information can be used to detect parts of the snake where tracking fails and that we can build a more appropriate initialization to reach the right contour. Our strategy is based on the growing method presented in [2].

Accuracy

Besides mismatching problems, the **snake accuracy** must be improved because 3D reconstruction requires well localized contours. To be convinced of this, the reader may observe the quality of the 3D reconstruction on the glass sequence when accurate (Fig. 4.d) or inaccurate (Fig. 4.b) snakes are used. The snake accuracy must be especially improved in two typical situations:

1. to detect curves with corner points or high curvature points. Indeed, it is well known that classical snakes only produce smooth curves.

2. to accurately detect contour outlining regions whose intensities are very different. In such a case (Fig. 3.a), the strong contour locally attracts the weak contour and this gives rise to a localization error.

1.2 Contributions

We especially tackle the mismatching problem which has received little attention up to now. The outlines of our algorithm are presented in section 2. The main causes of mismatching problems are studied in section 3 and we explain the criterion allowing such situations to be detected. The local strategies to overcome the mismatching problems and to improve the snake accuracy are described in section 4.

2 Overview of the tracking algorithm

We summarize in this section the two main steps of our algorithm. The interested reader can find technical developments as well as numerous examples in [4].

Initialization: the contour to be tracked is detected in the first frame

1. **step 1** A prediction of the curve location in the next frame is computed iteratively from the normal optical flow. Since we are interested in rigid objects, the contour shape does not vary too quickly from frame to frame. In order to avoid the divergence trend often encountered with iterative flow field computation, we therefore resort to an explicit 2D model and approximate the velocity field with a 2D rigid displacement. This gives rise to a robust estimation.

2. **step 2** Then the identification step is based on the active contour model and uses the predicted curve as initialization. In order to reduce mismatching, we use a rigid snake model so that the snake converges towards the contour whose shape is the most similar.

3. **step 3** Go to step 1 for the next image.

Statistical tests to avoid divergence of the prediction steps have been added and the prediction is now robust. We now focus on the way to control that the snake has reached the right contour.

3 Typology and detection of mismatching errors

We now describe successively the main causes of mismatching errors and the criteria we have developed to detect them.

3.1 Origins of mismatching errors

1. **Bad prediction**
   The obtained prediction is badly localized, due to a strong perspective effect between two frames for instance, making the 2D rigid displacement hypothesis transgressed.

2. **Scene complexity**
   Even with a correct initialization, problems may occur if there exists a strong contour very close to the contour to be tracked (Fig. 2). Indeed, if there are holes in the gradient profile of the contour to be tracked, the strong contour attracts the weak contour because of the small distance between them (Fig.

2.b). Moreover, the snake can also be attracted by a very close contour because the numerical scheme used to control snake evolution create small oscillations. Consequently, the snake may jump from one contour to another.

3. **Contour profile**
If the contour to be tracked outlines regions with very different intensity (Fig. 3), some localization errors occur in the neighborhood of a junction of a weak contour with a strong one.

3.2 Detecting the mismatching errors

We now focus on the way to detect the contour parts on which tracking fails. Local strategies are then applied on these parts to recover the whole contour. At first, it must be noticed that accurate assessment criteria are not required. Indeed, if a piece of contour is labeled as erroneous whereas it is not, the local strategies applied afterwards will restore the initial curve. We have therefore defined three criteria to detect erroneous parts:

<u>Criterion 1: the gradient profile is not preserved from frame to frame</u>
Given the parameterization induced by the active contours (if M_i^t is the i^{th} curve point in frame t, M_i^{t+1} denotes the position reached by this point after both prediction and convergence step), the points verifying:

$$|\nabla I^t(M_i^t) - \nabla I^{t+1}(M_i^{t+1})|/|\nabla I^t(M_i^t)| > threshold\ (in\ practice\ 30\%)$$

are considered as possible mismatched points.

<u>Criterion 2: Analysis of shape variations between two consecutive contours</u>
The contour shape may vary a lot between two frames without mismatching occurs (for occluding contours and large motion for instance). Thus, only strong curvature variations are significant because they often indicate that the snake either locks on a small detail (for instance on the circle used for camera calibration (Fig. f3)) or starts to converge towards the wrong contour (Fig. 2.b). If m and σ are respectively the average and the standard deviation of the curvature variation $diff_{curv}$ between two consecutive contours, the points such that $diff_{curv}(i) > m + 3\sigma$ are considered as problematic.

<u>Criterion 3: Detection of localization errors</u>
In order to detect the third mismatching case, contour points where the gradient modulus varies a lot are searched for.

Error detection

The first criterion delivers parts of the contour whose gradient profile varies a lot between two frames. Therefore it suggests either mismatching problems or real change in the contour profile (when a pattern appears on an occluding contour for instance). The second criterion is used to distinguish these two cases. When the snake converges towards an erroneous contour, a high curvature variation appears at the junction between the right and the wrong contour (Fig. 2.b). Therefore, strong variations of the gradient profile on a sufficiently large interval lead us to suppose mismatching whereas variation of the gradient profile on a small interval without curvature variation will be considered as a natural profile variation.

The different cases are summarized below, and we also indicate corresponding strategies used to override these difficulties, which we will present in the next section.

	Detection criterion	Diagnosis and Method
Case 1	Similarity of the gradient profile No curvature variation	acceptable tracking
Case 2	Similarity of the gradient profiles except on small intervals No curvature variation	Quite usual variations $\rightarrow$ acceptable tracking
Case 3	Strong gradient profile variations between two frames on a large interval	Mismatching on this interval $\rightarrow$ growing snake method
Case 4	No gradient profile variation (or only on a small interval) between two frames. strong curvature variation	localization error due to a detail $\rightarrow$ Continuity extension
Case 5	Strong gradient profile variations along the first contour, often strong curvature variation between the two contours	localization error due to a junction $\rightarrow$ improve the accuracy

It must be noticed that several situations locally give rise to curvature variations: for instance when a part of the snake is attracted towards another contour (case 3), the extremities of these parts often present curvature variations which must not be processed using case 5. On the same way, points where the gradient profile varies a lot (case 4) may involve curvature variations which must not be processed using case 5. Thus, the curvature variations considered in case 5 are points which are concerned by neither case 3 nor by case 4.

4 Growing strategies to recover the homologous contour

Once the mismatching errors have been detected, we use local strategies based on the growing method to recover the homologous contour. The aim is to supply the snake process with a more appropriate initialization than the predicted curve. The growing strategy presented in [2] allows the whole contour to be recovered from a small part. We briefly recall the principles of our method and then we present an improved growing method which works fine in complex environments.

4.1 The growing strategy

Fig. 1 summarizes the main steps of the method. First, the assessment criteria (see section 3) allows the mismatchings to be detected (a). Then the snake is lengthened at each extremity in the tangent direction (b) and we let the snake converge (c). The preceding stages are then iterated until the whole contour has been detected.

4.2 The improved growing strategy

An improved strategy is needed in complex environments. For instance in (Fig. 1.b), the snake may converge to the other object if the growth is too large. A weaker growth is therefore needed to avoid that the snake may be influenced by the wrong contour. Nevertheless, because of the shrinking effect due to the energy term $\int |v'(s)|^2$ of the active models, the snake length does not increase as soon as the

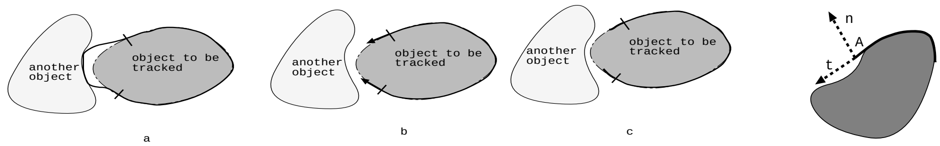

Figure 1: The growing method (a,b,c) and the boundary condition

growth is too small. We therefore use special boundary conditions to override this problem: we impose that the snake extremity belongs to the line passing through the snake extremity and perpendicular to the contour. Such a boundary condition ensures that the snake does not retract even with small growth, and the method therefore converges even in intricate environments.

On a technical point of view such conditions are called *transversality condition* in the Euler formalism and can be dealt with by the use of an additional equation. Nevertheless, this simplified case can be solved straightforwardly; if n is normal to the snake at the extremity A (Fig. 1), in the frame (A, t, n), the coordinates $(X(i), Y(i))$ of the snake satisfy the conditions: $X(0) = 0$; the other snake extremities $Y(0)$ and $(X(n), Y(n))$ are free. We therefore impose the further conditions $Y''(0) = Y'''(0) = X''(n) = X'''(n) = Y''(n) = Y'''(n) = 0$. Hence the discretization matrix associated to the X and Y coordinates can be computed (more technical details on the implementation of the snake evolution can be found in [3]).

Nevertheless, the snake must be assessed after each growth to be sure that it has not been attracted by the strong contour again. A simple way to do that is to cut the snake before the locus where mismatching has been detected. After convergence of the mismatched parts, the strong contour is then naturally detected. It is therefore easy to decide after each growth whether the snake extremities belong to this spurious contour or not. Significant results are shown in (Fig. 2). The contour to be tracked is the owl eye (Fig. 2.a) but the tracking fails in the next frame because the strong occluding contour attracts the snake (Fig. 2.b). The detected mismatchings are shown in (Fig. 2.c) and the result of snake growing as well as an intermediary step are shown in (Fig. 2.e and d).

4.3 Improving the accuracy: avoiding localization errors

Given a contour, the points where gradient varies a lot can be easily detected. Let $M(i_0)$ be such a jump location. The main idea is identical as the one developed for snake growing assessment: the snake is broken so that the two contours (the strong one and the weak one) may be detected and does not influence each other during the snake process. The whole contour can therefore be recovered from the two others in the following manner: the snake is cut on the side where the gradient is the weakest. let m_0 be the gradient average for points M_i such that $i_0 - d < i < i_0$ and let m_1 be the gradient average for points M_i such that $i_0 + d > i > i_0$ (in practice $d = 20$ *pixels*) . If $m_0 > m_1$ the snake is split into the two parts $M_{i_0+\epsilon-d} ... M_{i_0+\epsilon}$ and $M_{i_0+\epsilon} .. M_{i_0+\epsilon+d}$; otherwise in the two parts $M_{i_0-\epsilon-d} ... M_{i_0-\epsilon}$ and $M_{i_0-\epsilon} .. M_{i_0-\epsilon+d}$ (Fig. 3.a). The contour is therefore broken and gives rise to the two natural contours (Fig. 3.b.) and (Fig. 3.c) because the first part is attracted by the strong contour whereas the second part,which has been split under the junction

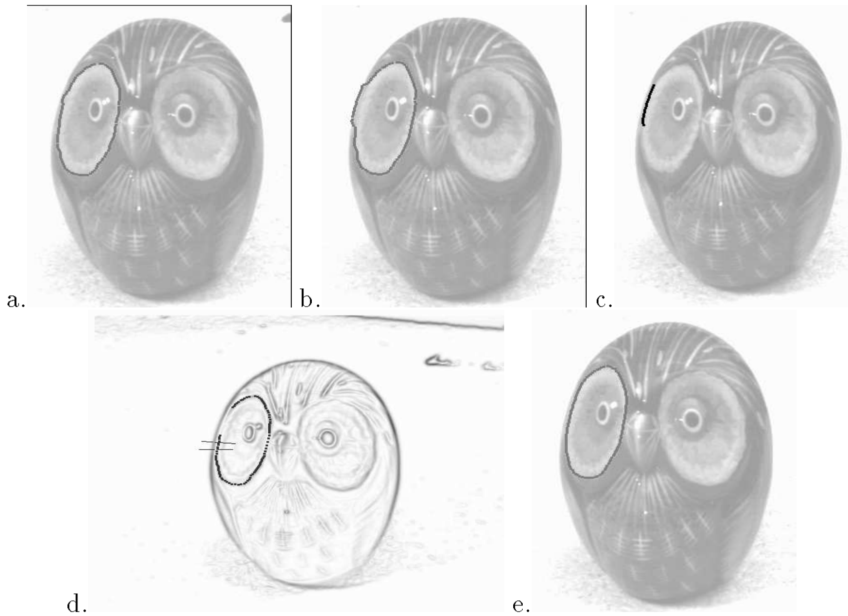

Figure 2: Tracking the owl eye with the improved growing method

converges towards the weakest contour without being attracted by the other contour. The snake boundary condition is the same as for growing assessment in order to avoid retraction. The whole contour must then be recovered : the weakest contour is then extended by a line until the strong contour has been reached. If this line does not intersect the strong contour, the connexion between the two contours is the point such that the distance between the line and the strong contour is the smallest (Fig. 3.d). This strategy allows delocalization errors to be overcome as shown in (Fig. 3.e). Significant results on the whole sequence are shown in (Fig. 4). The first figure (Fig. 4.a) exhibits the tracking results when the classical method is used whereas the use of the described method above leads to significant improvements of the tracking (Fig. 4.b) especially on the upper glass corners. This allows 3D reconstruction [5] to be noteworthy improved (Fig. 4.d).

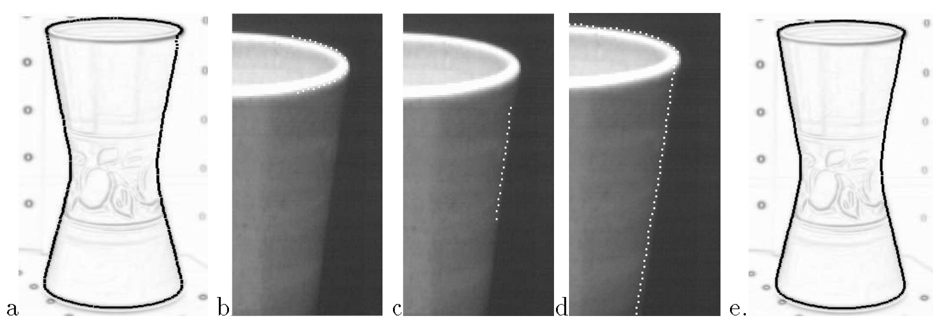

Figure 3: Avoiding the delocalization errors.

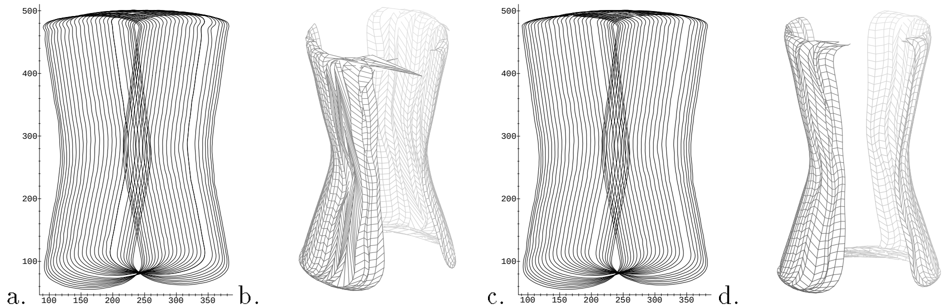

Figure 4: Tracking on the whole sequence without (a) and with (c) correction of the localization errors and the associated reconstruction

5 Conclusion

The main contribution of this article is to provide control strategies on the identification step when no a priori knowledge is available. This point has been little dealt with in works involving snake based tracking. We have shown in this paper that local strategies based on the growing method allow most mismatching problems to be overcome in the case of rigid objects or not very deformable objects. In fact, local strategies we have developed aim to supply the snake with a more appropriate initialization than the one given by the prediction stage.

References

[1] B. Bascles, P. Bouthemy, R. Deriche, and F. Meyer. Tracking Complex Primitives in an Image Sequence. In *ICPR 1994, Jerusalem (Israel).*

[2] M.-O. Berger. Snake Growing. In *ECCV 90, Antibes (France)*, pages 570–572. Springer-Verlag, April 1990.

[3] M.O. Berger. *Les contours actifs: modélisation, comportement et convergence.* Thèse de doctorat, Institut National Polytechnique de Lorraine, Vandœuvre-lès-Nancy, février 1991.

[4] M.O. Berger. How to Track Efficiently Piecewise Curved Contours with a View to Reconstructing 3D Objects. In *ICPR 1994, Jerusalem (Israel)*

[5] E. Boyer and M.O. Berger. 3D Surface Reconstruction From Occluding Contours. *Technical report, Centre de Recherche en Informatique de Nancy.*

[6] R. Curwen, A. Blake, and A. Zisserman. Real time Visual Tracking for surveillance and Path Planning. In *ECCV 92, Santa Margherita Ligure (Italy)*, pages 879–883.

[7] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active Contour Models. *International Journal of Computer Vision*, 1: 321–331, 1988.

[8] F. Leymarie and M. Levine. Tracking Deformable Objects in the Plane Using an Active Contour Model. *IEEE Transactions on PAMI*, 15(6): 617–634, june 1993.

[9] D. Terzopoulos and R. Szeliski. Tracking with Kalman Snakes. *Active Vision, chapter 1, MIT Press, 1992*, pages 3–20, 1992.

[10] N. Ueda and K. Mase. Tracking Moving Contours using Energy Minimizing Elastic Contour Models. In *ECCV 92, Santa Margherita Ligure (Italy)*, pages 453–457.