Object Models From Contour Sequences

Edmond Boyer

HAL Id: inria-00590062
https://inria.hal.science/inria-00590062

Submitted on 3 May 2011

Object Models From Contour Sequences

Edmond Boyer

Crin–Cnrs / Inria Lorraine
54506 Vandœuvre les Nancy Cedex, France.
e-mail: boyere@loria.fr

Abstract. We address the problem of building 3D object models from
image sequences obtained with known camera motion. An approach
based on a local reconstruction method is presented. Recovered sur-
faces are described as polygonal meshes. To this purpose, reconstructed
points are triangulated and surface areas which are not covered by rims
are detected since they may lead to false reconstructed points. Result-
ing meshes are then regularised in order to correct noise perturbations
which affect the reconstruction. Experimental results on real data are
presented.

1   Introduction

Recovering and representing three-dimensional object shape is an important task
in computer vision. The derived models can be used for recognition, localisation
and design automation. In the case of curved objects, rich and robust information
on the shape are provided by image contours which are called *occluding contours*.
The corresponding contours on the surface, the *rims*, are viewpoint dependent
and defined by the fact that viewing directions at their points are tangential to
the surface. In addition, it has been shown that local shape recovery from three
or more occluding contours is possible given a known camera motion. Several
algorithms [Cip 90, Vai 92, Sze 93] allow such a local reconstruction under the
assumption of a linear camera motion. In previous works [Boy 95], we proposed
an explicit solution for rim point reconstruction which is correct for any camera
motion.

The approaches mentioned above are concerned with local shape estimation.
However, a complete surface description is needed to build an object model.
Seales and Faugeras [Sea 95] generate a polygonal surface mesh using a spline-
based slicing technique. The input is a set of surface points which are recov-
ered using the local approach of Vaillant and Faugeras [Vai 92]. Zhao and Mohr
[Zha 94] attempt to recover the global surface description in a single stage by
use of B-spline patches. This approach introduces a direct regularisation of the
reconstructed surface, but it requires a complete *a priori* parametrisation of the
surface which is usually not available. Zheng [Zhe 94] presents a global method
in the case of plane camera rotations. In this work, it is shown that regions of
the surface unexposed to contours are related to non-smoothness of contour dis-
tribution in the image. A detection algorithm based on this fact is proposed for

plane camera rotations.

In this paper, we present a global approach for shape estimation which extends previous results on surface reconstruction from occluding contours [Boy 95] and yields robust surface model. First, we study the case of surface regions where local approximations of the reconstruction method are not valid. This the case, for example, at planar, concave parts of the surface or at surface discontinuities. This corresponds to surface areas which are unexposed to rims and where reconstruction yields 3D points which are not on the surface. We present an algorithm to detect them, the interest is to remove such *false* points from the final surface mesh. Our approach extends the result of Zheng [Zhe 94] for planar camera rotations to any camera motion.

Secondly, we propose a surface description based on a triangulation of the recovered rim points. Such a description preserves the coherence with data since no approximation functions which introduce a bias are needed to describe the surface, and it does not require any *a priori* knowledge on the surface parametrisation.

We then present an original method to regularise the reconstructed surface. The idea is to optimise point positions by minimising an energy function which takes into account the surface area.

Finally, significant results on a real object are shown which prove the reliability of the method.

2   Preliminaries

In this section, we summarise some elementary notions to be used along this paper.

We assume that the imaging system is based on the pinhole model (i.e. perspective projection). Therefore, the vector position of a point P on the object surface can be written

$$r = C + \lambda T, \tag{1}$$

where C is the camera centre position, T the unit viewing direction and λ the depth of the point P along the viewing direction. For a given camera position there is a locus of points on the surface $\mathcal{S}$ where the normal N is perpendicular to the viewing direction. This set of points is called the *rim* and its projection onto the image plane is called the *occluding contour*.

An essential property of an occluding contour is that the normal to the surface on the corresponding rim can be computed from image information T and τ:

$$N = \frac{T \wedge \tau}{|T \wedge \tau|}, \tag{2}$$

where τ is the tangent to the occluding contour. This property leads to the following expression for the depth at a rim point [Cip 90]:

$$\lambda = \frac{C_t.N}{T_t.N},\tag{3}$$

where t denotes time (suffix denoting derivative). This formula is defined for continuous observations of the object surface. In the case of discrete information, depth can be computed only by approximation using successive contours. This can be done by using at least three occluding contours and locally approximating the surface up to order two. Such an approximation leads to a linear estimation of rim point depth. For further information about the local reconstruction method, the reader is referred to [Boy 95].

3   Detection of regions unexposed to contours

In the process of reconstruction from contours, parts of the observed surface [1] may not be covered by rims, depending on the local form of the object surface and the camera motion. When the camera turns around such a region, rims *jump* from one extremity to the other and a non-smoothness appears in the spatio-temporal surface. These unexposed regions correspond to part of the observed surface which are concave, parabolic, planar [Zhe 94] or self-occluded and they can not be recovered using only contour information. Moreover, the local model used to compute surface point positions is not valid in these areas. Therefore, they should be detected in order to build a surface model. In this section, we propose a method to detect them.

3.1   Continuous observations

In the case of continuous observations, we will say that a surface point P is exposed to contours if there exists a camera centre position $C(t)$ in the sequence for which the viewing line at P does not pierce the observed surface but is tangent to it.

A particular case corresponds to points which belong to exposed parts of the surface and where the viewing line is tangent to the surface at more than one point (including itself). Such *critical* points constitute the limit of surface areas exposed to contours.

From a quantitative point of view, we can see that at critical points P, the viewing direction T and the normal to the surface N are continuous functions[2] of t but not the depth λ; indeed the observed surface point jump from the first

---

[1] From now on, we will suppose that the term *observed surface* means the union of all object surfaces which are present in the scene.

[2] t usually refers to the time, however, a more rigourous definition is that t parametrises the camera motion

point of tangency P^- to the second P^+. If we denote by λ^- and λ^+ depths at P^- and P^+, and by using the depth formula 3 we have:

$$\lambda^- = \frac{C_t^-.N}{T_t^-.N}, \quad \lambda^+ = \frac{C_t^+.N}{T_t^+.N},$$

where C_t^-, T_t^- denote the left derivatives according to t and C_t^+, T_t^+ denote the right derivatives. If we suppose that t parametrises the camera motion $C(t)$, then C_t is defined at P_c and:

$$T_t^+.N = \frac{C_t.N}{\lambda^+}, \quad T_t^-.N = \frac{C_t.N}{\lambda^-}.$$

This shows that the viewing direction T is not differentiable according to the camera motion parameter t at critical points. Therefore, regions unexposed to contours correspond to non-smoothness of the spatio-temporal surface [Gib 94, Fau 93] defined by $T(s,t)$, where s parametrises the occluding contours. Thus, they may be detected by considering the continuity of T_t along epipolar curves. This result generalises the one obtained in [Zhe 94] for planar camera motions to the case of perspective projections and any camera motions.

3.2 Discrete observations

In the case of discrete observations, the depth at a rim point is computed by a second order approximation [Boy 95]. Such a reconstruction can lead to *false* surface points if the viewing lines which are used jump over a surface area unexposed to contours. To detect these *false* points, we can estimate λ^- and λ^+. However, this requires more information than the three occluding contours used in the second order approximation. Another approach consists in estimating T_t^- and T_t^+ at the reconstructed point. Such values should be equal for a point exposed to contours. Thus, we can detect false points by computing:

$$\sigma = \frac{T_t^+.N}{T_t^-.N}, \tag{4}$$

This can be done by first order approximation and, consequently, three successive occluding contours.

We denote by $C(t_1)$, $C(t_2)$ and $C(t_3)$ three successive camera positions at time t_1, t_2 and t_3 and by $T(t_1)$, $T(t_2)$ and $T(t_3)$ three viewing directions which are epipolar correspondents ($T(t_1)$ with $T(t_2)$, and $T(t_2)$ with $T(t_3)$). At the rim point $P(t_2)$ on the viewing line define by $C(t_2)$ and $T(t_2)$, we have at order 1:

$$T_t^- = \frac{(T(t_2) - T(t_1))}{(t_2 - t_1)}, \quad T_t^+ = \frac{(T(t_3) - T(t_2))}{(t_3 - t_2)},$$

and if we suppose that $\|C_t\|$ is constant between the three camera positions, which is not a constraint in the discrete case, we can write:

$$(t_2 - t_1) = \frac{\|(C(t_2) - C(t_1))\|}{\|C_t\|}, \quad (t_3 - t_2) = \frac{\|(C(t_3) - C(t_2))\|}{\|C_t\|},$$

where $\| \ \|$ denotes the euclidian norm. Finally:

$$\sigma = \left(\frac{T(t_3).N(t_2)}{\|(C(t_2) - C(t_1))\|} \Big/ \frac{-T(t_1).N(t_2)}{\|(C(t_3) - C(t_2))\|} \right), \tag{5}$$

where $N(t_2)$ is the normal to the surface at $P(t_2)$. Due to the approximation σ will not necessarily be equal to one at a point exposed to contour but close to this value. Therefore, surface areas which correspond to false reconstructed points will be detected by use of a threshold. Hence, reconstructed points which verify:

$$|1 - \sigma| < thresh, \tag{6}$$

where $|\ |$ denotes the absolute value, will be considered as surface points exposed to contours and points which do not satisfy this condition will be considered as false points occluding an area unexposed to contour. Note that in the case of discrete observations (i.e., occluding contours at discrete times) the fact that the reconstructed points are false or not depend on *a priori* information on the surface. Indeed, *tresh* corresponds to the limit below which a planar surface region will be considered as a concavity. This value should therefore be set according to the application.

4 Triangulation

The result of the reconstruction is a set of 3D contour points. A parametric description of this set of points is required for a global surface representation. Such a description is used to build functions which approximate or model the object surface. This can be either a parametrisation of the surface or first, a triangulation of the rim points. In this section, we briefly discuss such a description and we present our approach.

If a parametrisation of the surface is available then an approximation can be computed using, for example, spline functions [dB 78]. However, without any *a priori* information on the object surface, it might be difficult to compute such a parametrisation. Another approach consists in triangulating the reconstructed points. The object surface can then be represented as a set of connected polygonal facets. The advantage is that only topological information are required: the neighbours of each vertex of the triangulation.

An *optimal* triangulation in the 2D case is the Delaunay triangulation which maximises the minimum angle of the resulting mesh. The generalisation to the 3D case leads to the tetrahedrisation of the set of points which is a volume. This method is therefore not well adapted to the case of rim points; indeed rims describe the object surface or only part of it and thus, they do not necessarily define a volume. In addition, the 3D points are organised in contours and hence, the resulting triangulation should conserve this information.

Consequently, our approach consists in constructing a triangular mesh which respect the adjacency of successive rims:

– two 3D points which are not on the same rim can be connected if and only if they are on two consecutive rims.

Thus, the problem of triangulating the set of rim points is reduced to one of triangulating each pair of consecutive rims in the sequence. This leads to the following condition for a triangular facet:

– A facet is defined by two points on one rim and one point on the next or the previous rim.

This condition is not sufficient to define a unique triangulation of a rim pair and an additional criterion must be used to isolate one set of facets. The triangulation of a rim pair corresponds therefore to the problem of finding a minimum cost path in a directed graph, in which the vertices correspond to the set of all possible connections between the points of the rim pair [Fuc 77]. A solution can be found in n^2 operations where n is the number of points on a rim.

The additional criteria that can be used are: the sum of facet areas, the sum of edge lengths or the sum of angles. We choose the sum of facet areas, thus the resulting surface minimises the total surface area.

The triangulation algorithm is applied to the whole set of reconstructed points including false points. Then, all the triangular facets which contains one false point (according to 6) are removed from the surface mesh.

5 Regularisation

The resulting triangular mesh approximates the part of the surface which was covered by the observed rims. However, since the 3D reconstruction process is very sensitive, the reconstructed surface may present perturbations such as folds. This is due to different reasons including:

– the noise which is present in the acquisition system,
– camera calibration errors,
– contour tracking errors.

In order to correct these defaults, positions of mesh vertices are optimised by minimising a functional E:

$$E = E_{dist} + \alpha \ E_{reg},$$

where E_{dist} controls the fitness to the data and E_{reg} the smoothness of the reconstructed surface. In this section, we precise the original energy functions that are used.

5.1 Distance energy

In the context of a reconstruction from occluding contours, data consist of image positions $\{p_{i,j}\}$ of mesh vertices, where i is the image number and j is the point number on the occluding contour. Thus the fidelity to the data can be characterised by the distance between the image point and the projection of the corresponding mesh vertex $P_{i,j}$ onto the image plane. Hence:

$$E_{dist}(P_{i,j}) = \sum_{i,j} |M_i P_{i,j} - p_{i,j}|^2,$$

where $\{M_i\}$ are the calibration matrices (i.e., perspective projection matrices) of the different image planes. This expression is consistent with the fact that original data are viewing lines and not 3D reconstructed points. In the optimisation procedure, surface point displacements are therefore not limited to a closed neighbourhood of the reconstructed point, but to a closed neighbourhood of the corresponding viewing line.

5.2 Regularising energy

In order to optimise surface point positions and thus, smooth the reconstructed surface we introduce a regularising energy. Classically, such energies are based on curvatures or, equivalently, second derivatives of surface point position function [Pog 85, Ter 86]. To this aim, derivatives can be approximated by finite differences [Wel 94] or discrete curvatures can be computed [Hen 92]. However, in the case of a triangular mesh, resulting functionals may be strongly non-linear and thus, difficult to minimise. Furthermore, errors such as surface folds may not be corrected by considering discrete curvatures.

We therefore introduce a term which is based on the triangle areas. Hence, the regularising energy is given by:

$$E_{reg}(\Gamma_{k \in \{1, N_t\}}) = \sum_{k=1}^{N_t} S(\Gamma_k)^2,$$

where N_t is the number of triangles and $S(\Gamma_k)$ the area of a triangle. This energy, and its derivatives, are easy to compute. Consequently, it can be minimised using a classical optimisation method.

Finally, the total energy can be written:

$$E = \sum_{i,j} |M_i P_{i,j} - p_{i,j}|^2 + \alpha \sum_{k=1}^{N_t} S(\Gamma_k)^2. \tag{7}$$

Only positions of points which do not belong to the surface boundaries are optimised. The parameter α controls the trade-off between fidelity to the data and variation of the sum of squared triangle areas and should therefore be set by the user.

6   Experimental results

We present here results for a real image sequence of a jug (see figure 1(a)(b)).
This sequence was taken using a rotating turntable and the occluding contours
were tracked using snakes [Ber 94, Kas 88]. The result of the reconstruction is
shown in figures 1(c)(d). In these figures, the triangular mesh corresponds to
the hull of the object surface defined by the observed contours. Note that due
to the contour tracking, the flower which appears on the jug yields a fold in the
triangular mesh (see figure 1(c)). In figure 2(a) the triangular facets which con-
tains one or more false point (according to the detection algorithm) are shown.
These facets were removed from the final surface as shown in figures 2(b). Note
that the fold corresponding to the flower has been corrected (see figure 2(b)). In
figures 2(c)(d) the final surface was rendered using a ray tracer and projected
in two images of the sequence. This is done by use of the perspective projec-
tion matrices computed during a preliminary calibration step. It shows that the
resulting surface is coherent with the original one.

(a)   (b)

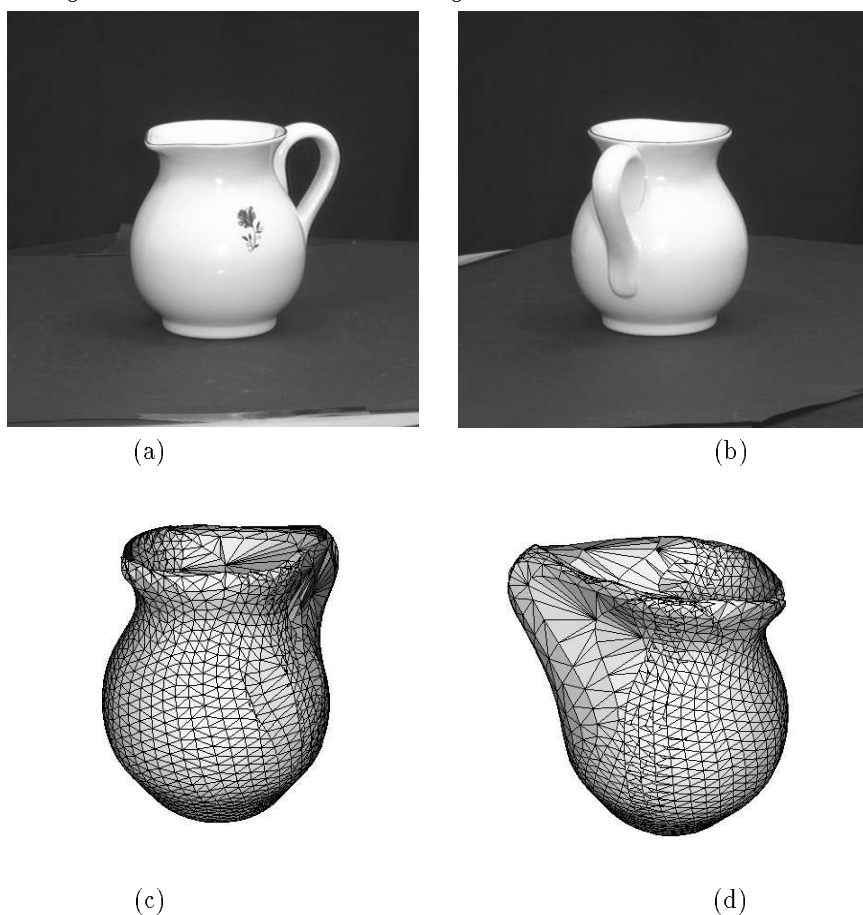(c)   (d)

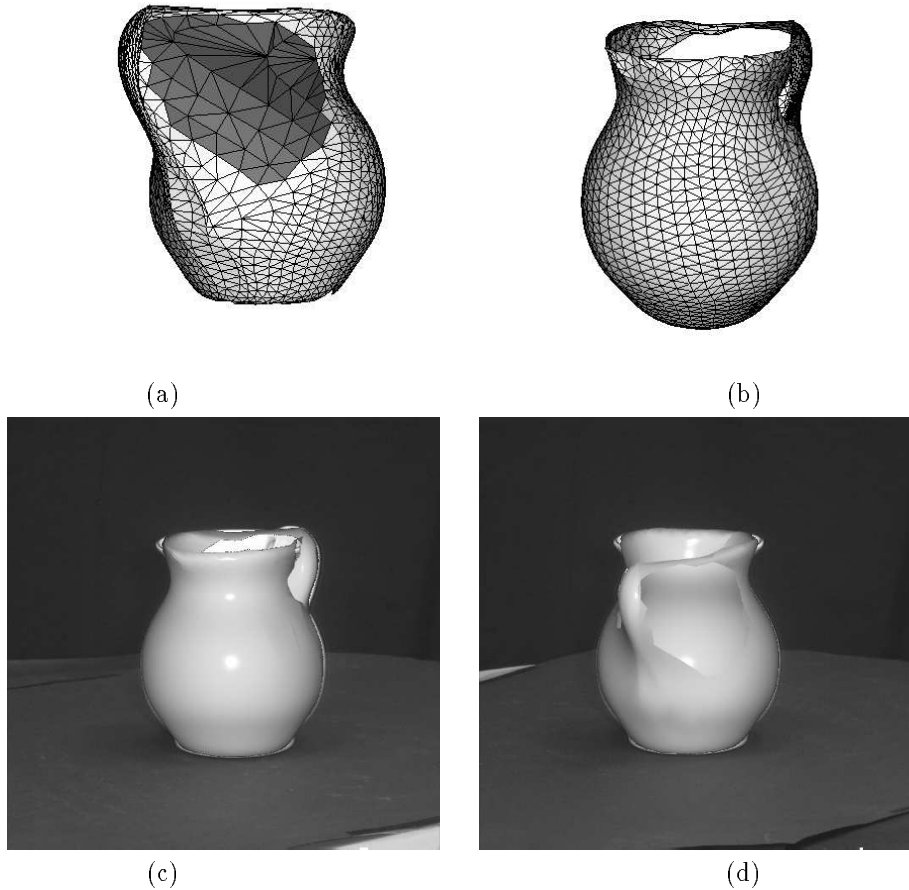Figure 1: (a)(b) two sequence images, (c)(d) triangulated rim points.

Figure 2: (a) facets which are detected as unexposed to contours, (b) final mesh surface, (c)(d) projections of the final surface in the original image.

7 Conclusion

We have described a reconstruction procedure that produces a smooth surface from image sequences. Resulting polygonal meshes are obtained by triangulating reconstructed rim points. Such approach is well adapted to the shape from contour problem since it does not require any *a priori* informations on the observed surface. Thus, it allows partial as well as complete surface descriptions. In addition, mesh point positions may be optimised by considering the regularity of the surface. We have proposed a regularising term which is based on triangle areas. This term as well as its derivatives are easy to compute and allows reconstruction perturbations such as folds to be corrected. This make it possible to build object models.

In this work, we have also studied surface areas unexposed to contours. In the case of discrete observations, such regions may leads to false points estimation. We extended previous results and we proposed a criterion to detect these false points.

The interest of detecting these areas is also related to improvements of the global recovering procedure. Indeed, another reconstruction method may be applied to these regions once they are clearly determined in 3D as well as in the images. Our current work is concern with such integration of different reconstruction methods.

References

[Ber 94] M.-O. Berger. How to Track Efficiently Piecewise Curved Contours with a View to Reconstructing 3D Objects. In *ICPR'94, Jerusalem (Israel)*, 1994.

[Boy 95] E. Boyer and M.-O. Berger. 3D Surface Reconstruction Using Occluding Contours. In *CAIP'95, Prague (Czech Republic)*, September 1995. LNCS, volume 970.

[Cip 90] R. Cipolla and A. Blake. The Dynamic Analysis of Apparent Contours. In IEEE, editor, *ICCV'90, Osaka (Japan)*, December 1990.

[dB 78] C. de Boor. *A Practical Guide to Splines*. Springer-Verlag, 1978.

[Fau 93] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. Artificial Intelligence. MIT Press, 1993.

[Fuc 77] H. Fuchs, Z.M. Kedem, and S.P. Uselton. Optimal Surface Reconstruction from Planar Contours. *Communications of the ACM*, 20(10), 1977.

[Gib 94] P.J. Giblin and R.S. Weiss. Epipolar Fields on Surfaces. In *ECCV'94, (Stockholm, Sweden)*, May 1994. LNCS, volume 801.

[Hen 92] Henry P. Moreton and Carlo H. Séquin. Functional Optimization for Fair Surface Design. In *Computer Graphics (Proceedings Siggraph)*, July 1992.

[Kas 88] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active Contour Models. *International Journal of Computer Vision*, 1: 321–331, 1988.

[Pog 85] T. Poggio, V. Torre, and C. Koch. Computational Vision and Regularization theory. *Nature*, pages 314–319, 1985.

[Sea 95] W.B. Seales and O.D. Faugeras. Building Three-Dimensional Object Models From Image Sequences. *CVIU*, 61(3): 308–324, 1995.

[Sze 93] R. Szeliski and R. Weiss. Robust Shape Recovery from Occluding Contours Using a Linear Smoother. In *CVPR'93, New York (USA)*, 1993.

[Ter 86] D. Terzopoulos. Regularizarion of Inverse Visual Problems Involving Discontinuities. *IEEE Transactions on PAMI*, 8: 413–424, 1986.

[Vai 92] R. Vaillant and O. Faugeras. Using Extremal Boundaries for 3-D Object Modeling. *IEEE Transactions on PAMI*, 14(2): 157–173, February 1992.

[Wel 94] William Welch and Andrew Witkin. Free-Form Shape Design Using Triangulated Surfaces. In *Computer Graphics (Proceedings Siggraph)*, July 1994.

[Zha 94] C. Zhao and R. Mohr. Relative 3D Regularized B-Spline Surface Reconstruction Through Image Sequences. In *ECCV'94, (Stockholm, Sweden)*, May 1994. LNCS, volume 801.

[Zhe 94] Jiang Yu Zheng. Acquiring 3-D Models from Sequences of Contours. *IEEE Transactions on PAMI*, 16(2): 163–178, 1994.

This article was processed using the LaTeX macro package with ECCV'96 style