Reconstructing 3D Objects from Silhouettes with Unknown Viewpoints: The Case of Planar Orthographic Views

Andrea Bottino[1], Luc Jaulin[2], and Aldo Laurentini[1]

[1] Dipartimento di Automatica ed Informatica, Politecnico di Torino,
Corso Duca degli Abruzzi, 24 – 10129 Torino, Italy
{andrea.bottino,aldo.laurentini}@polito.it
[2] LISA
Avenue Notre Dame du Lac, 62, 49 000 Angers, France
jaulin@univ-angers.fr

Abstract. 3D shapes can be reconstructed from 2D silhouettes by back-projecting them from the corresponding viewpoints and intersecting the resulting solid cones. This requires knowing the position of the viewpoints with respect to the object. But what can we say when this information is not available? This paper provides a first insight into the problem, introducing the problem of understanding 3D shapes from silhouettes when the relative positions of the viewpoints are unknown. In particular, the case of orthographic silhouettes with viewing directions parallel to the same plane is thoroughly discussed. Also we introduce sets of inequalities, which describe all the possible solution sets and show how to calculate the feasible solution space of each set.

1 Introduction

A central problem in computer vision is understanding the shape of 3D objects from various image features. Many algorithms are based on occluding contours or silhouettes. The main approach is volumetric, and consists in building the volume **R** shared by the regions $\mathbf{C}_i$ (see Fig. 1) obtained by back-projecting each silhouette S_i from the corresponding viewpoint. This simple reconstruction technique is called *Volume Intersection* (VI) (see [1], [4], [8], [9], [10]). It requires the 3D positions of silhouettes and viewpoints. However, in several practical situations this information is not available and therefore VI cannot be performed. Even if this simple reconstruction technique is not possible, we would like to get the best of the available information.

Before entering the problem, we briefly review some definitions relevant to our problem. First, the concept of *visual hull* of an object [6], which is the object that can be obtained by VI using all the viewpoints that belong to a viewing region completely enclosing the original object without entering its convex hull. It is also the largest object that produces the same silhouettes as the given object. A point of the surface of the reconstructed object **R** is an *hard point* [6] if it belongs to any object that produces the same silhouettes from the same viewpoints. The concept of hard point allows

A. Sanfeliu and J. Ruiz-Shulcloper (Eds.): CIARP 2003, LNCS 2905, pp. 153–162, 2003.

stating a necessary condition for the reconstruction to be optimal, and is at the basis of interactive VI algorithms [3].

In the following, for brevity, we will use the expression “set of silhouettes” to specify a set of silhouettes together with the position of the corresponding viewpoint with respect to each silhouette. These data, allow constructing a solid cone for each silhouette, but not positioning the cones in the 3D space. To understand how the 3D shape is related to such a set of silhouettes, two main questions can be considered. The first question is: given a set of silhouettes, does an object exist able to produce them? We will call *compatible* a set of silhouettes if the same object can generate them. An object able to produce a compatible set of silhouettes will be said to be compatible with the set. The second question is the main practical issue: how can we find one or more compatible objects given a compatible set of silhouettes, as that produced by a real object? We will present a set of results that provide a first insight into the problem.

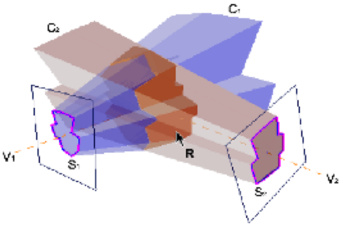

Fig. 1. The volume intersection technique

2 Compatibility of Orthographic Silhouettes of 3D Objects

In the rest of this paper we will restrict ourselves to consider simply connected 3D objects and their orthographic projections. This approximates the practical case of objects small with respect to their distance from the camera. The reader is referred to [12] for a proof of the statements of this section.

First, we will investigate the compatibility of two silhouettes. Let S be a 2D orthographic silhouette of a 3D object. Let us project orthographically S along a direction in the plane of S. The 1D silhouette obtained depends on the angle α that the chosen direction makes with the *x* axis of a coordinate system fixed with respect to S (Fig. 2). Let $L(S,\alpha)$ be the length of the 1D silhouette of S. The following statement holds.

Proposition 1. *A necessary and sufficient condition for two orthographic silhouettes* S_1 *and* S_2 *to be compatible is that two angles* α_1 *and* α_2 *exist such that* $L(S_1,\alpha_1)= L(S_2,\alpha_2)$.

What happens when we have to deal with more silhouettes? That is, how can we find if three or more silhouettes are compatible? Clearly, we have that:

Proposition 2. *A necessary condition for a set of silhouettes to be compatible is that all pairs of silhouettes of the set are compatible.*

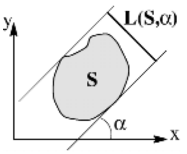

Fig. 2. The 1D silhouette L(S,α)

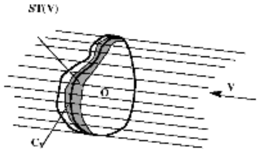

Fig. 3. The strip ST(V) and the curve C_v

However, in general, to be compatible in pairs is not sufficient for a set of silhouettes to be compatible (see [12]). A necessary and sufficient condition for the compatibility of more than two silhouettes can be found considering a property of the reconstructed object **R**. Let us consider one of the silhouettes involved in the process, the corresponding viewing direction V and the cylinder circumscribed to the object **O** made of lines parallel to this direction (Fig. 3). Each line of this cylindrical surface must share with the surface of **O** at least one point. These points form a curve C_V. This curve belongs to an annular surface, a strip ST(V) of variable width, which is what is left of the original circumscribed cylinder after the various intersections. During the reconstruction process, this annular strip cannot be interrupted; at most it can reduce to a curve with zero width. In this case, the curve consists of hard points. Therefore we can formulate the following condition for the VI algorithm to be feasible:

Proposition 3. *A necessary and sufficient condition for a set of silhouettes to be compatible is that it be possible to find viewpoints such that no annular strip of the reconstructed object is interrupted.*

In the next sections this condition will be used for constructing algorithms both for verifying the compatibility of a set of silhouettes and reconstructing compatible 3D objects.

3 Silhouettes with Viewing Directions Parallel to a Plane

In this section we deal with a particular case of the general problem, where all viewing directions are parallel to the same plane (Fig. 4). Clearly, all silhouettes have the same height and the same plane must support all cylinders obtained by back-projection.

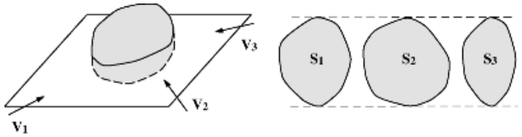

Fig. 4. Viewing directions parallel to the same plane

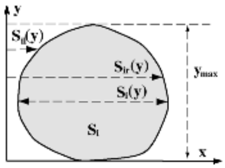

Fig. 5. Notations used for a silhouette.

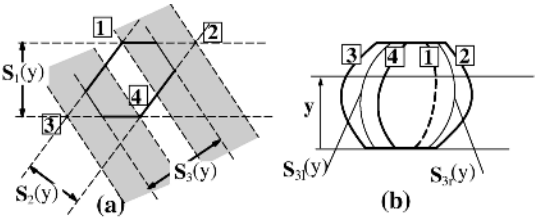

Fig. 6. (a) A case where S_3(y) is compatible with S1(y) and S_2(y) in a horizontal plane. (b) The condition for the compatibility of the whole silhouettes.

We consider first the compatibility of three silhouettes S_1, S_2 and S_3. Each planar silhouette S_i is defined, for $0 \le y \le y_{max}$ by two curves $S_{il}(y)$ and $S_{ir}(y)$ (see Fig. 5). For simplicity, let us consider mono-valued functions. Also let $S_i(y)=S_{ir}(y)-S_{il}(y)$. Let us consider a horizontal plane corresponding to a value of y between 0 and y_{max}, and its intersection with the three cylinders obtained by back-projecting the silhouettes. Let us consider in this plane the arrangement of the 2D silhouettes $S_1(y)$, $S_2(y)$, $S_3(y)$ and of the viewpoints V_1,V_2,V_3 shown in Fig. 6(a). It is not difficult to see that proposition 3 requires that the two lines projecting the endpoints of S_3(y) along the direction V_3 must lie inside the two areas highlighted in Fig. 6(a). For the whole silhouettes to be compatible, this must hold for all y. For the reconstruction to be possible, $S_{3l}(y)$ must lie between the two leftmost curves, in this case the projections of the vertices **3** and **4**, and $S_{3r}(y)$ must lie between the two rightmost curves, the projections of the vertices **1** and **2**.

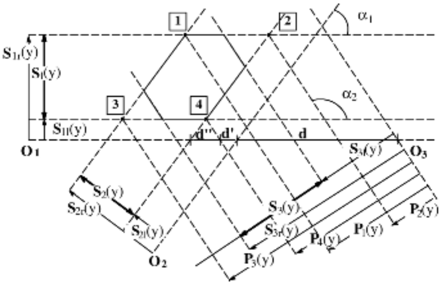

Fig. 7. The intersections in a horizontal plane

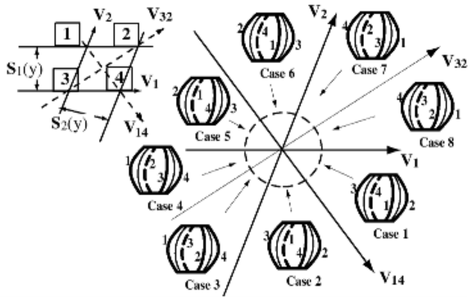

Fig. 8. The eight intersection cases

To derive the set of inequalities that define for this case feasible intersection parameters, let us inspect in more detail the intersection in a horizontal plane (Fig. 7). Let O_1, O_2, O_3 be the intersections of the axes y of the coordinate system of each silhouette with this plane. Intersecting $S_1(y)$ and $S_2(y)$ requires to fix an angle, let it be α_1. Intersecting also $S_3(y)$ requires choosing two more parameters: the angle α_2 and a distance, let it be d (see Fig. 8). d is the distance between two points lying on the line projecting O_1 along the direction V_1. The first is the intersection of this line with the line projecting O_2 along V_2, and the second is the intersection with the line projecting O_3 along V_3. Thus, to find feasible solutions we must search the 3-dimensional space $[\alpha_1, \alpha_2, d]$. Let $P_1(y)$, $P_2(y)$, $P_3(y)$ and $P_4(y)$ be the distances from O_3 of the orthographic projections of the vertices of the parallelogram onto the line supporting $S_3(y)$. The compatibility condition for the three silhouettes is expressed by the following inequalities:

$$\begin{array}{l} S_{3r}(y) \geq P_4(y), \quad S_{3r}(y) \leq P_3(y), \quad S_{3l}(y) \geq P_2(y) \\ S_{3l}(y) \leq P_1(y), \quad P_4(y) \geq P_1(y) \end{array} \qquad (1)$$

In (1), the purpose of the fifth inequality is to characterize the case just analysed, let it be Case 1. Seven other cases, determined by the direction of V_3 with respect to V_1, V_2, and the directions of the diagonals V_{14} and V_{32} of the parallelogram, are possible, each producing different sets of inequalities (see Fig. 8). For each case, a possible orthographic projection onto the plane of S_3 of the edges of the object produced by the first intersection is shown with thick lines. The boundaries of S_3 are the thin lines.

Four Silhouettes

Let us consider Case 1 and add a fourth silhouette S_4. In each horizontal plane $S_1(y)$, $S_2(y)$ and $S_3(y)$ produce a polygon with six vertices and three pairs of parallel edges (Fig. 9). The new intersection is defined by two more parameters, the angle α_3 between V_1 and V_4 and the distance d_1, measured, as d, along the line that projects O_1 from V_1. Satisfying the condition of Proposition 3 requires, in each horizontal plane, to cut away two opposite vertices, without eliminating completely the edges that meet at these vertices. By orthographically projecting the six vertices onto the plane of S_4 we obtain six curves. For the new intersection to be feasible, the boundaries $S_{4l}(y)$ and $S_{4r}(y)$ of S_4 must lie in the areas bounded by the two leftmost and the two rightmost curves respectively.

Various sets of inequalities result, depending on the direction of V_4. First, let us distinguish two cases (case (**a**) and (**b**) in the left of Fig. 9) related to the directions which determines the leftmost and rightmost vertices (**5** and **7** for case (**a**) and **7** and **5** for (**b**)). In each case we have four sub-cases for the leftmost and rightmost strips where S_{4l} and S_{4r} must lie (see right part of Fig. 9). The inequalities corresponding to each sub-case are easily written. For instance, for the sub-case $\mathbf{a}_1$ it is:

$P_5(y) \leq S_{3l}(y)$ $S_{3l}(y) \leq P_4(y)$

$P_1(y) \leq S_{3r}(y)$ $S_{3r}(y) \leq P_7(y)$

$P_4(y) \leq P_6(y)$ $P_8(y) \leq P_1(y)$

where $P_i(y)$ are the projections of the points $\mathbf{i}(y)$ onto the plane of S_4. As before, the last two inequalities guarantee that the inner boundaries of these areas are actually P_6 and P_8.

Summarizing, each set of inequalities that defines feasible intersection parameters for four silhouettes contains 11 inequalities (the five inequalities related to the first three silhouettes and six new inequalities also referring to S_4). As for the number of sets of inequalities, we have 8 cases for three silhouettes, 3 pairs of opposite vertices and 8 cases for each pair, and thus 192 sets each containing 11 inequalities.

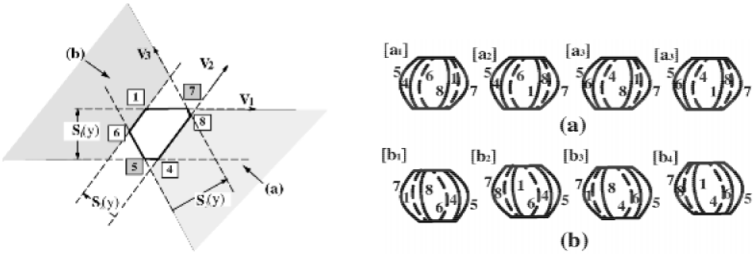

Fig. 9. Cases (a) and (b) and the 8 sub-cases

Five or More Silhouettes

The previous discussion about the fourth silhouette does hold for any further silhouette. In fact, we must always cut a pair of opposite vertices without deleting completely the edges converging at these edges. It follows that each new silhouette adds two parameters, seven inequalities for each case. Thus, for n silhouettes, the number of parameters is $2n$-3, and the number of inequalities $6(n-3)+5$ ($n \geq 3$). Each new silhouette adds 8 sub-cases for each pair of opposite vertices. For the n-th silhouette, the pair of vertices are n-1. Let $N_c(n)$ be the number of sets of inequalities for n silhouettes. For $n>3$ it is: $N_c(n)=8(n-1)N_c(n-1)$. Therefore we must face an exponential growth of the number of cases.

4 Writing the Sets of Inequalities

The inequalities discussed in the previous section allow to answer, in a particular case, both question raised in the introduction: finding objects compatible with a set of compatible silhouettes, and understanding if an (artificial) set of silhouettes is compatible. We have developed an algorithm for automatically writing the sets of inequalities, which works on the following basis. In this section we will renumber the silhouette starting from S_0, and not S_1, in order to handle easily the indices of the. The axes of the reference system are aligned with the axis of the projection of S_0 on the plane. Let's assume, without loss of generality, that V_0 is parallel to the *y* axis of the reference system and the line supporting S_0 is parallel to the *x* axis. The origin of the reference system corresponds with the intersection of the projections of $\mathbf{O}_0$ along V_0 and $\mathbf{O}_1$ along V_1 on the plane. The position of the i^th silhouette is determined by two parameters, d_i and α_i, where d_i is the distance between the projection onto the y axis of the i^th origin $\mathbf{O}_i$ along V_i and the origin (hence $d_0=d_1=0$), and α_i is the angle between V_i and V_0 ($\alpha_0=0$). We assume that the angle is positive if $V_0 \times V_i$ has the same verse of $x \times y$; it also follows that $V_i=(sen(\alpha_i),-cos(\alpha_i))$. Let C_j be the vertices of the polygon resulting from intersecting S_0 and S_1. The equations of the first 4 vertices (Fig. 10) are:

$$C_1 = (S_{0l}, S_{1l}/sen(\alpha_1) - S_{0l}/\tan(\alpha_1)), \quad C_2 = (S_{0r}, S_{1l}/sen(\alpha_1) - S_{0r}/\tan(\alpha_1))$$

$$C_3 = (S_{0r}, S_{1r}/sen(\alpha_1) - S_{0r}/\tan(\alpha_1)), \quad C_4 = (S_{0l}, S_{1r}/sen(\alpha_1) - S_{0l}/\tan(\alpha_1))$$

The sets of inequalities previously introduced can be written in terms of the distances from the origin along the y axis of the projections of the vertices of the parallelogram and of S_{il} and S_{ir} along the viewing direction of the i^th silhouette. For each projection, the lines passing through the vertices of the polygons have equations $C_j + V_i t$ and their intersections (P_{ij}) with the y axis of the reference system are given by $P_{ij} = c_{yi} + c_{xi}/\tan(\alpha_i)$. Now, let d_{il}, d_{is} be the projections on the y axis of S_{il} and S_{ir}. It follows that:

$$d_{il} = d_i + S_{il}(y)/sen(\alpha_i), \quad d_{ir} = d_i + S_{ir}(y)/sen(\alpha_i)$$

Projecting the vertices and S_i onto the y axis, the verse of the inequalities also depends on the value of the angle between the current viewing direction and V_0. For instance, in the example shown in Fig. 11, we have:

$$P_{21} \le d_{2l} \le P_{22} \le P_{24} \le d_{2r} \le P_{23}, \quad 0 < \alpha_2 < \pi$$

$$P_{21} \ge d_{2l} \ge P_{22} \ge P_{24} \ge d_{2r} \ge P_{23}, \quad \pi < \alpha_2 < 2\pi$$

In order to be able to write the inequalities in an automatic way, the general form of the inequalities can be rewritten multiplying each term by $sin(\alpha_2)$. In the previous example, the set of inequalities become:

$$\begin{cases} P_{21} sin(\alpha_2) \le d_{2l} sin(\alpha_2) \\ d_{2l} sin(\alpha_2) \le P_{22} sin(\alpha_2) \\ P_{22} sin(\alpha_2) \le P_{24} sin(\alpha_2) \\ P_{24} sin(\alpha_2) \le d_{2r} sin(\alpha_2) \\ d_{2r} sin(\alpha_2) \le P_{23} sin(\alpha_2) \end{cases} \quad \text{or} \quad \begin{cases} sin(\alpha_2)(P_{21} - d_{2l}) \le 0 \\ sin(\alpha_2)(d_{2l} - P_{22}) \le 0 \\ sin(\alpha_2)(P_{22} - P_{24}) \le 0 \\ sin(\alpha_2)(P_{24} - d_{2r}) \le 0 \\ sin(\alpha_2)(d_{2r} - P_{23}) \le 0 \end{cases}$$

In general, each term of the inequalities will be multiplied by $sin(\alpha_i)$.

Each new vertex C_j, $j>4$, is the intersection of the line every edge lies on and the specific projection line relative to V_i. All these lines are projection lines, and can be written as:

$$D_{il} + V_i r \quad or \quad D_{ir} + V_i r, \quad where:$$

$$D_{il} = (S_{0l}, 0), i = 0; (0, d_{il}), i > 0 \qquad D_{ir} = (S_{0r}, 0), i = 0; (0, d_{ir}), i > 0$$

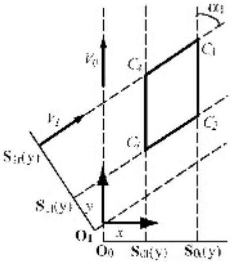

Fig. 10.

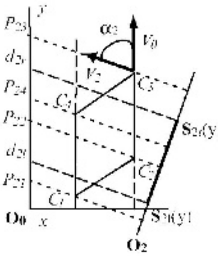

Fig. 11.

5 Solving the Inequalities

A set inversion technique ([7]) has been applied for finding the feasible solution set **S** of the set of non-linear inequalities that characterizes each sub-case. This technique performs a paving the parameter space with boxes. If the current box [**p**] is proved to be inside **S**, then the box is kept as part of the solution space. If it is proved to have an empty intersection with **S** then it is discarded. Otherwise, [**p**] is bisected except if its width is smaller than a defined threshold. The dimensionality of the initial box is equal to the number of variables involved in the set of inequalities. To prove that a given box [**p**] is inside **S**, interval computation ([13]) has been used. The technique illustrated is used to find feasible parameter sets for one value of y between 0 and y_{max}. Each feasible parameter set corresponds to a group of inequalities that can take place for the same object. If one of the parameter sets is empty, the corresponding group of inequalities can be discarded. Otherwise, we could perform an incremental computation, adding each time (or subtracting) a small Δy, related to the shape of the silhouettes, to the previous y or, in the case of polygonal silhouettes, taking as $y+\Delta y$ the height of the next horizontal strip. For each group of inequalities, the new feasible parameter set at $y+\Delta y$ must be a subset of the set at y. The cases are arranged in a tree, whose depth is the number of silhouettes. Instead of considering all the leaves at the lower level of the tree, that is all the intersections with all the silhouettes, we start the computation at higher level. If an inequality group has an empty feasible parameter set, the child cases can be discarded. Also, the initial feasible parameter set for each child is derived from the one evaluated for the father and it is not taken as the whole initial box.

In order to assess the validity of the approach described, we have experimented the algorithms in a virtual environment. An orthogonal camera rotating on a plane around the object has been used to create silhouettes of synthetic objects. The paving technique introduced has been used to find feasible parameter sets satisfying the inequality sets. For each point of the solution, a compatible object can be reconstructed using VI. We have experimented the approach with different minimal paving resolution and different number of silhouettes. Given the three silhouettes S_0, S_1 and S_2 of Fig. 10 as input, several different compatible objects, each one reconstructed from one of the eight different feasible sets, can be seen in Fig. 13. Other examples can be seen in Fig. 12 and Fig. 13. Finally, in Fig. 14, the expanded tree of the sub cases generated by four silhouettes of the chamfer box of Fig. 13 can be seen. The dark nodes correspond to the open nodes.

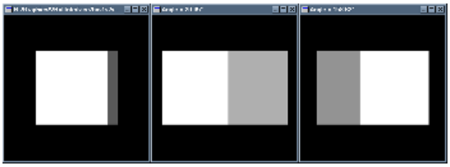

Fig. 10. The silhouettes S0, S1 and S2

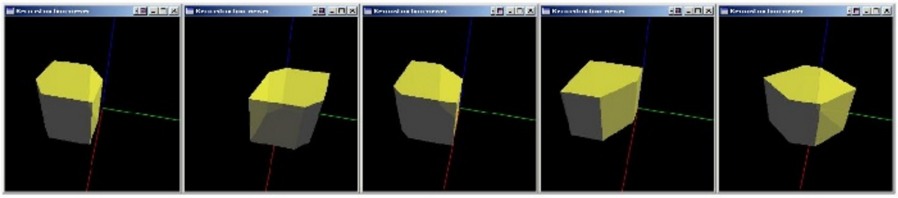

Fig. 11. Objects compatible with the silhouettes of Fig. 10

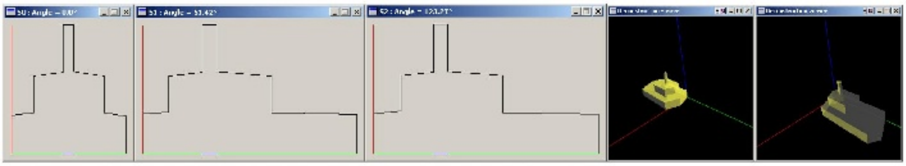

Fig. 12. A boat

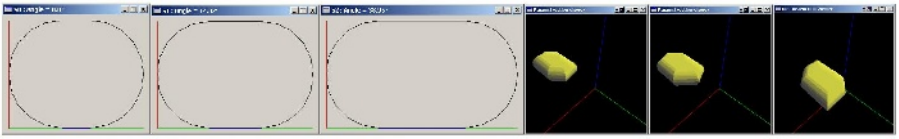

Fig. 13. A chamfer box

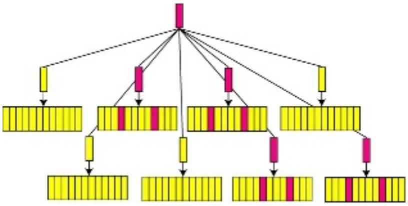

Fig. 14. Solution tree

6 Conclusions

We have introduced and explored the problem of understanding the shape of 3D objects from silhouettes when the relative position of the viewpoints is not known, which happens in several practical cases. We have presented a necessary and sufficient condition for a set of orthographic silhouettes to be compatible. This condition has been applied to the particular case of orthographic projections with viewing directions parallel to a plane. For this case, we have been able to work out sets of inequalities, involving the volume intersection parameters, which allow computing feasible solution sets. An algorithm for automatically writing the inequalities has been developed, and a technique involving the paving of the parameter space has been introduce to evaluate, if they exist, feasible parameter sets satisfying the inequalities.

Several problems are open. Among them, the case of orthographic projection with unrestricted viewing directions, and the case of perspective projections. We will also study and discuss thoroughly the case of generic silhouettes, which are not simply connected objects or having their boundaries defined by mono-valued functions. Another question is worth considering. Except for special cases, we expect that infinite compatible objects exist, specified by a region in the space of the intersection parameters. Simple ways for describing the shape of the compatible objects seem desirable.

References

[1] N. Ahuja and J. Veenstra: Generating octrees from object silhouettes in orthographic views, *IEEE Trans. on PAMI*,Vol.11,pp.137–149,1989

[2] K. Astrom and F. Kahl: Motion estimation in image sequences using the deformation of apparent contours, *IEEE Trans. on PAMI*, Vol. 21, No.2, pp.114–126, 1999

[3] Bottino, L. Cavallero, A.Laurentini, “Interactive reconstruction of 3D objects from silhouettes,” *Proc. WSCG’2001,* Vol.II, pp. 230–236, Plzen, February 5–9, 2001

[4] E. Boyer and M. O. Berger,” 3D Surface Reconstruction Using Occluding Contours,” *Int’l. J. of Computer Vision,* vol.22, no.3, pp. 219–233, 1997

[5] R. Cipolla and A. Blake,”Surface Shape from the Deformation of Apparent Contours,”*Int’l. J. of Computer Vision,* vol.9, no.2, pp. 83–112, 1992

[6] A.Laurentini : How far 3-D shapes can be understood from 2-D silhouettes," *IEEE Trans. on PAMI*, Vol.17, No.2, pp.188–195, 1995

[7] L.Jaulin and E.Walter: Guaranteed nonlinear set extimation via interval analysis, *Bounding Approaches to System Identification,* edited by M.Milanese et al, Plenum Press, New York, pp.363–381, 1996

[8] W. Matusik, et al.: Image based visual hulls, *Proc. ACM Siggraph 2000,*pp.369-374

[9] Noborio et al.:Construction of the octree approximating three-dimensional objects by using multiple views, *IEEE Trans. on PAMI*, Vol.10, pp.769–782,1988

[10] M.Potemesil: Generating octree models of 3D objects from their silhouettes in a sequence of images, *Comput.Vision Graphics Image Process.*,Vol.40,pp.1–29,1987

[11] J. Y. Zheng: Acquiring 3D models from sequences of contours, *IEEE Trans. on PAMI*, Vol. 16, no.2, pp.163–178,1994

[12] A.Bottino, A.Laurentini, Shape-from-silhouettes with unknown relative position of the viewpoints: investigating a new problem, Proc. SPPRA, 2002

[13] R.E. Moore, Methods and Applications of Interval Analysis, Proc. SIAM, 1979