INTERACTIVE RECONSTRUCTION OF 3-D OBJECTS FROM SILHOUETTES

Bottino A., Cavallero L. and Laurentini A.

Dipartimento di Automatica ed Informatica
Politecnico di Torino, Corso Duca Degli Abruzzi 24
10129 Torino
Italy
{bottino,laurentini}@polito.it www.polito.it

ABSTRACT

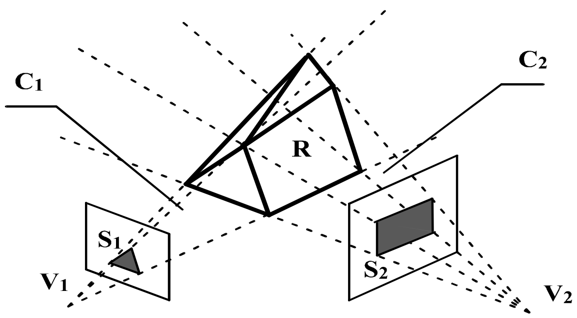
Many techniques for reconstructing 3-D shapes from 2-D images use silhouette data. A problem with this approach is that, if no a priori information about the 3-D shape is available, we do not know neither the accuracy of the reconstruction, nor where it is better to locate new viewpoints for improving the accuracy. We present and demonstrate a new general approach to interactive, object-specific shape-from-silhouette algorithms. The approach holds for completely unknown shapes. It is based on a necessary condition for the reconstruction to have been performed with the best possible accuracy. From this condition, we derive: 1) a quantitative measure of reconstruction accuracy; 2) rules for finding new viewpoints if the accuracy is not satisfactory. The algorithm has been implemented for polyhedra, and demonstrated in a virtual environment.

Keywords: computer vision, shape from silhouette, visual hull, volume intersection, interactive algorithms, geometric probing

1. INTRODUCTION

Reconstructing 3-D shapes from 2-D images is a basic research area in computer vision. Many algorithms are based on occluding contours or silhouettes (see for instance [Astro99], [Vaill92], [Zheng94], [Sulli98]). In the latter case, only the contours which occlude the background are considered. Reconstruction from silhouettes is as reconstruction from shadows cast from the object with point light sources located at the viewpoints.

The information provided by set of silhouettes and viewpoints is summarized by the volume **R** shared by the solid regions of space $\mathbf{C}_i$ within which each silhouette S_i constrains the unknown object **O** to lie (Fig.1). **R** approximates **O** more or less closely, depending on the viewpoints and the shape of **O** itself. Finding this volume is a popular reconstruction technique ([Matus00], [Ahuia89], [Nobor88], [Potem87], [Chian89]) called volume intersection (VI).



The Volume Intersection approach.
Figure 1

Silhouette-based reconstruction algorithms must face an important problem. If no *a priori* information about the 3-D shape is available, we have no idea of the accuracy of the reconstruction obtained, and consequently we do not know whether halting or not the reconstruction process. It is also clear that, for a given object, each new VI operation can refine the reconstruction to different degrees, depending on the viewpoint chosen. So another problem is where to locate new viewpoints. If we were given the shape of the object, in principle we could construct some object specific algorithm for finding the next best viewpoint, but the shape is the very information we are looking for. So in general we are reduced to a simple "the more silhouettes the better" strategy.

Object specific heuristics for finding the next viewpoint have been suggested in [Shanm91] and [Lavak89]. However, they do not face the basic problem of understanding when a satisfactory reconstruction accuracy has been obtained. Some results have been obtained in 2D in the area known as *geometric probing* (see [Skiena92] for a comprehensive survey). Optimal strategies for finding the next viewpoint have been found for determining convex polygons from their 2D silhouettes.

The purpose of this paper is to present and demonstrate a *quantitative* approach able to solve, at least in part, this apparently under-constrained problem. This approach is based on a *necessary condition* for the reconstruction obtained to be optimal. This condition can be verified considering the reconstructed object **R** only, without any *a priori* knowledge about **O**. We will also show that, when this necessary condition is not satisfied, we can compute a measure of the current reconstruction accuracy, and derive suggestions for locating a new viewpoint.

Although the necessary condition holds for any object, verifying if the condition is satisfied, and, if not, computing the measure of reconstruction accuracy, are not trivial tasks and depend on the object's category.

In this paper we will demonstrate our active approach for polyhedral objects in a virtual environment. In Section 2 we summarize some relevant theoretical points. In Section 3 we state the necessary condition for a reconstruction to be optimal, and outline the interactive reconstruction approach. In Section 4 we describe the algorithm implemented for reconstructing convex and concave polyhedra and report the experimental results obtained.

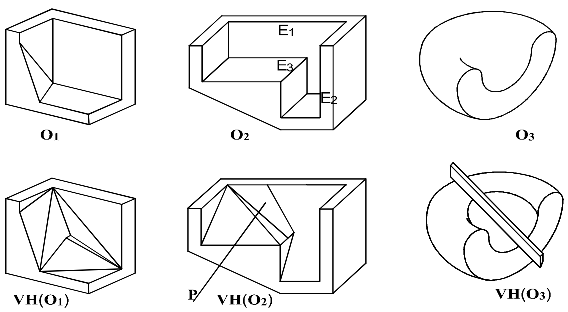
2. VISUAL HULL, HARD AND SOFT POINTS

2.1 The Visual Hull

The *visual hull* **VH**(**O**,**V**) of an object **O** relative to a viewing region **V** is the closest approximation of **O** which can be obtained by VI with viewpoints belonging to **V**. It is also the largest object that produces the same silhouettes as **O** from viewpoints belonging to **V.** All the visual hulls relative to viewing regions which: 1) completely enclose **O**; 2) do not share any point with the *convex hull* of **O**; are equal. This is called the *external visual hull* of O, or simply the visual hull **VH**(**O**).

Three objects and their visual hulls are shown in Fig.2. In the figure, P is a curved ruled patch due to lines tangent at the edges E_1, E_2, E_3. An intuitive physical construction of the visual hull is shown for the third object. Suppose filling the concavity with soft material. The visual hull can be obtained by scraping off the excess material with a ruler grazing the hard surface of the object in all possible ways.

The reader is referred to [Laure94] for any further details.



Some objects and their visual hulls
Figure 2

Algorithms for computing the visual hulls of polygons, polyhedra, solids of revolution and smooth surface objects have been presented in [Laure94], [Petit98], [Laure99a], [Laure99b].

2.2 Inferring the shape of an object from its silhouettes: hard and soft points

The problem of inferring the shape of **O** from the boundary volume **R** has been discussed in [Laure95]. Here we summarize the relevant matter of this paper. The points of the *surface* of **R** can be divided into: *hard points*, which belong to any possible object originating **R**; *soft points*, which could belong or not to **O**. In other words, hard points are guaranteed to belong to (the surface of)**O**.

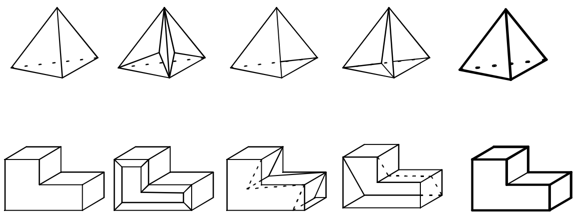
The problem of computing the hard points can be considered in two reconstruction cases.

Case 1 - In this case we assume that the *optimal* reconstruction has been performed, i.e. the visual hull of **O** has been obtained. This is the inverse problem of computing the visual hull of a given object. Let a hard point of a visual hull be a *VH-hard* point. A necessary and sufficient condition for a point to be VH-hard is the following:

Prop. 1. A point P is VH-hard iff there exists one straight line sharing only P with **VH(O)** .

This is a point condition, which requires further work for finding the hard parts of general surfaces. For the case of polyhedral visual hulls however an algorithm has been presented [Laure95], which allows to find in polynomial time the hard edges of the object (the internal points of planar faces are always soft).

Examples of VH-hard points are shown in Fig.3. The four leftmost objects of each row have the same visual hull (the first object of the row), and share its hard edges, marked thick in the rightmost objects.



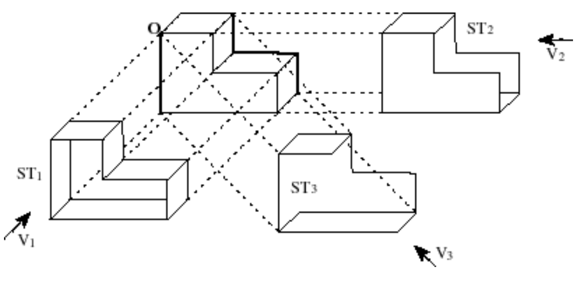
Examples of VH-hard edges.
Figure 3

Case 2 - Let us assume that we do not know whether the reconstruction is optimal or not. Obviously, this is the situation to face usually. Let call *R-hard* a hard point in this case.

A necessary and sufficient condition exists for a point to be R-hard. Let the *strip* ST(**O,**V_i) relative to the viewpoint V_i and the object **O** be the part of surface (let it be a *visual cone*) of the solid cone $\mathbf{C}_i$ starting at V which bounds **R**, or, in other words the part of the visual cone which is left after the VI operations due to the other silhouettes. Also let the *width* of a strip ST(**O,**V_i) at a point P be the length of the segment of the half-line starting at V_i and passing through P which lies on the strip. The following statement holds:

Prop. 2. A point P is R- hard iff it belongs to a strip of width zero at P.

This is a constructive condition. The R-hard points can be obtained as a by-product of the VI algorithm. In practice, we can find R-hard points or, at most, R-hard lines (R-hard surface would require an infinite number of silhouettes). The idea of R-hard points is demonstrated in Fig.4. Two ideal viewing directions, V_1 and V_2, shown together with the corresponding strips, are insufficient to produce hard points, since both strips have everywhere non-zero width. One additional co-planar viewing direction V_3 supplies a couple of R-hard segments.



Examples of R-hard edges.
Figure 4

3. THE INTERACTIVE RECONSTRUCTION APPROACH

In this Section we state a necessary condition for the reconstruction to be optimal, and show that from this condition we can derive: i) a *measure* of reconstruction accuracy and ii) hints for selecting new viewpoints when the accuracy is not satisfactory.

3.1 A necessary condition for the reconstruction to be terminated

Suppose that we are given a set of viewpoints and silhouettes of an unknown object **O**, which produce the object **R**. Let VH(**R**) be the set of VH-hard points, and R(**R**) be the set of R-hard points.

In general it is VH(**R**)>R(**R**) (see for instance Fig.3 and Fig.4). In fact, by adding new silhouettes we cannot delete old hard points, but only add new hard points, and the visual hull is the reconstruction made with all possible silhouettes. If the best possible reconstruction has been obtained, that is **R**=**VH**(**O**), no more hard points can be found. It follows that:

Prop 3. A necessary condition for the reconstruction to be optimal is: VH(**R**)= R(**R**)

Let us define *compatible* a reconstruction which satisfies this condition, that is, such that all the points of **R** which *could be* hard have been found to be *actually* hard.

The condition is not sufficient for the reconstruction to be optimal, and in some cases small details of the object could remain undetected. However, the necessary condition stated is fruitful. In fact, not only it can tell us when to halt the VI process, but it also allows, if not satisfied, to understand how far we are from a compatible reconstruction. In addition, if we are not satisfied with the accuracy already obtained, it suggests the location of the next viewpoint.

3.2 A general approach to interactive reconstruction

Let us outline a general interactive reconstruction approach, which holds for any kind of object. First, let us introduce some figure of merit *f(*HPS,**R***)* depending on a set HPS of hard points and the surface of **R**. This figure should be defined in relation with the category of objects considered. For instance, for smooth-surface object, where only hard points, and not hard lines, can be found, *f* could be the number of hard points per square inch, possibly corrected according to the local curvature.

For polyhedra, where both hard points and hard lines can be found, *f* could be defined as the total length of the hard lines, plus the number of hard points multiplied by some constant weight. Let the *relative reconstruction accuracy RA* be defined as:

$RA = f(\,R(\mathbf{R}),\mathbf{R}) / f(\,VH(\mathbf{R}),\mathbf{R})$

Clearly *RA* is one iff the reconstruction is compatible. An interactive reconstruction algorithm can be constructed as follows

a) compute **R**, VH(**R**), R(**R**) and *RA*

b) if $(1-RA) \le \varepsilon$, where ε is some predefined error , stop;

c) otherwise, refine the reconstruction by testing if the *potentially* hard points VH(**R**)-R(**R**) are actually hard or not. This can be done for a subset of the potentially hard points by selecting a viewpoint such that this subset is projected on the boundary of the silhouette of **R**.

Although this approach is quite general, to implement a general algorithm is not trivial. One reason is that it requires algorithms for computing VH(**R**) for any kind of object. Since one algorithm is available for polyhedra, we have implemented the interactive algorithm for these solids.

4. AN INTERACTIVE ALGORITHM FOR POLYHEDRA

In many cases, the visual hull of a polyhedron is a polyhedron in his turn. Obviously, this happens when the object is exactly reconstructable (**O**=**VH**(**O**)). Also if this is not the case, the surface of **VH**(**O**) could consist of planar patches only. In all these cases, we state as goal of the interactive algorithm to find a compatible reconstruction, where VH(**R**)=R(**R**) so that the reconstruction accuracy is one.

In some cases however, the surface of **VH**(**O**) includes curved ruled patches, as shown in Fig. 2. For simplicity, we will not deal with this cases. The algorithm, implemented for virtual polyhedra, consists of four parts:

1) ALGVI, which performs the volume intersection;
2) ALGVH, the algorithm for computing the hard points VH(**R**);
3) ALGR, the algorithm for computing the hard points R(**R**);
4) NEXT, which compares VH(**R**) and R(**R**) and, if the sets are not equal, computes the next viewpoint;

We will not describe here in details the first three parts. ALGVI, the VI algorithms, works for any kind of polyedra and both parallel and perspective projections. ALGR is a rather straightforward consequence of the volume intersection, and ALGVH is described in [Laure95], to which the reader is referred.

In the following we will present NEXT, which uses two different sets of rules for convex and non convex polyhedra, and the experimental results obtained.

4.1 The convex case

In this case running ALGVH is not necessary, since any reconstructed object **R** is convex, and thus all the edges are VH-hard. Therefore, NEXT computes new viewpoints until all the edges of **R** are R-hard.

As far as the authors know, reconstructing unknown convex polyhedra from silhouettes has been only studied by Dobkin, Edelsbrunner and Yap [Dobki86]. They proposed a silhouette probing strategy, dual of a finger probing strategy, which determines the polyhedron with a bounded number of silhouettes N_s :

$V/2 \le N_s \le V+5F$

where V and F are the numbers of vertices and faces respectively. The strategy and the bounds are for ideal viewpoints only.

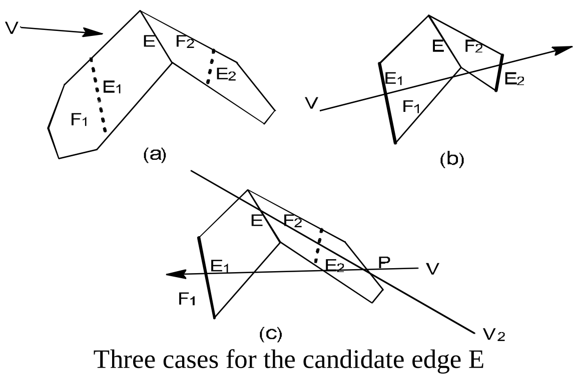
In our case, it is not difficult to define finite strategies for any kind of viewpoints based on the hard points and hard lines already found. Let us recall first that one edge E is R-hard if it belongs to a strip with zero width. This requires *three* surfaces of visual cones to make contact with **R** at E. In addition, each face must be verified with a viewpoint lying in the plane of the face.

Let an edge of **R** be a *candidate* edge if it is not R-hard. The following general strategy requires at most

$N_s \le F+3E$

silhouettes, since at each step it verifies a new face or adds a surface making contact with an undetected or candidate edge .

Strategy. Start with three random viewpoints. Until there are candidate edges, chose a new viewpoint such that at least one candidate edge E is on the boundary of the silhouette of the current **R**. According to the type of the faces F_1 and F_2 of **R** converging into E, it could be necessary for the viewpoint to satisfy some further condition.



Three cases for the candidate edge E

Figure 5

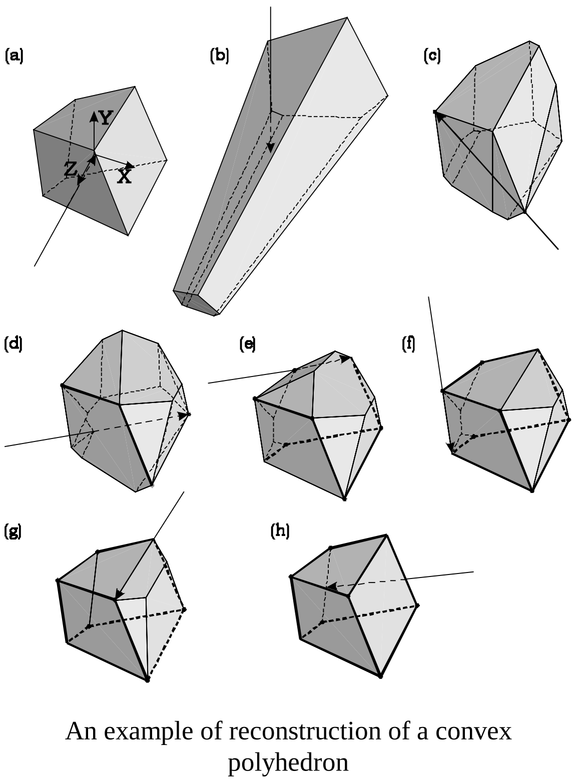
Case 1 - There are no hard edges lying on F_1 and F_2. In other words, the edges of **O** which produce F_1 and F_2 have not yet been detected (see Fig. 5a, where the undetected edges are shown as dotted lines). In this case, any viewpoint seeing E on the boundary of the silhouette is fit. In fact, it will either verify E as a hard edge, or produce a new face of **R** making contact with at least one edge yet undiscovered, inside the solid angle formed by F_1 and F_2.

Case 2 - There is one R-hard edge on both F_1 and F_2(see Fig. 5b, where the hard edges are thick segments). A viewpoint lying on a line passing through one point of both the hard edges will either verify a possible undetected face, or produce one or more faces making contact with undetected edges.

Case 3 - There is one R-hard edge on one face, let it be F_1 (see Fig. 5c). Consider a line starting at V_2 (the viewpoint which produces F_2) and lying on F_2. This line will intersect F_2 at two points. Let P by the point closer to V_2. Chose a new viewpoint on a line passing trough P and one point of the R-hard edge on F_1. It is clear from the figure that it will produce a new face making contact with one or more undetected edges (possibly E_2).

A number of rules can be applied for speeding up the algorithm. For instance, if two R-hard vertices joined by a candidate edge have been found, the edge can be classified as hard, even if it does not belongs to a strip of zero width .

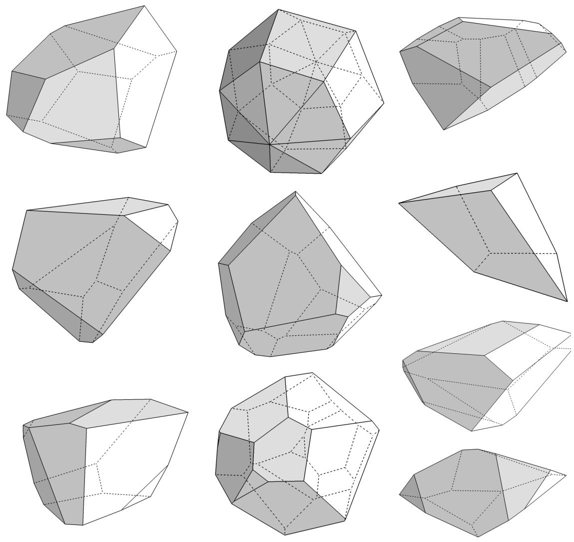
It is also clear that the general strategy described allows to position the viewpoints in several different positions. To restrict the choices, some heuristics can be used.



An example of reconstruction of a convex polyhedron
Figure 6

The algorithm NEXT actually implemented for convex polyhedra uses several heuristics which we will omit for brevity. An example of the steps performed by the algorithm is shown in Fig.6, where the arrows indicate the viewpoints and the hard edges are thicker.

We have experimentally evaluated the performance of the algorithm with respect to our upper bound F+3E, and the bounds of the algorithm of Dobkin, Edelsbrunner and Yap. For this purpose, the algorithm has been applied to fifty randomly generated convex polyhedra. Some of them are shown in Fig.7.



A subset of the fifty convex polyhedra reconstructed
Figure 7

The average numbers of vertices, edges and faces of the fifty polyhedra are 17.8, 26.7 and 10.9 respectively. We have found that the average number of silhouettes required for the reconstruction is 13. This number should be compared with 91, our average upper bound, with 99.9, the average upper bound of Dobkin, Edelsbrunner and Yap and with 8.9, their average lower bound. The last comparison shows that the average behaviour of our algorithm is close to optimality.

4.2 The case of concave polyhedra

To reconstruct the visual hull of concave polyhedra is much more difficult. We have already recalled that there are curved visual hulls, which in principle require an infinite number of polygonal silhouettes. Even polyhedra with polyhedral visual hull could require an unbounded number of silhouettes for their reconstruction [Laure97].

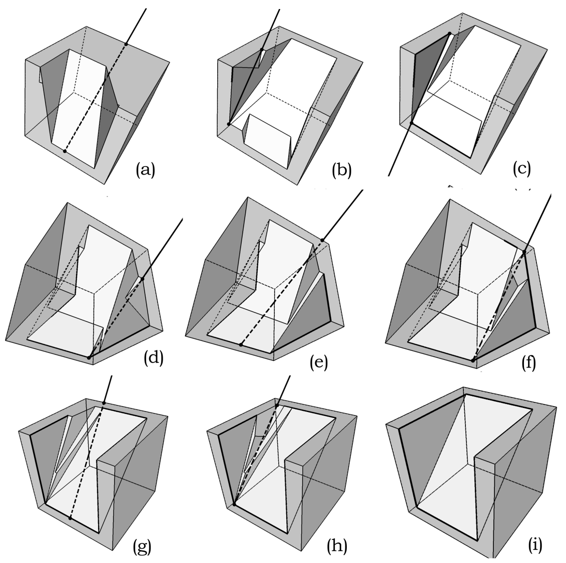
Both these cases could be approached with approximations which neglect small details. Anyway, to develop an algorithm working for any concave polygons is outside the scope of this work.

The algorithm we have implemented has been able to reconstruct concave polyhedra such that:

1) the hard edges lies all on the convex hull
2) the concave parts of the visual hull can be reconstructed with a bounded number of intersections.

As far as we know, this is the only interactive algorithm presented for reconstructing concave polyhedra from silhouettes. In more details, the algorithm first reconstructs the visual hull of the convex parts and the rim (always on the convex hull) of the concavities. In a second step it attempts to reconstruct the concavities. We omit for brevity the details of the various rules used, which can be found in [Caval00].

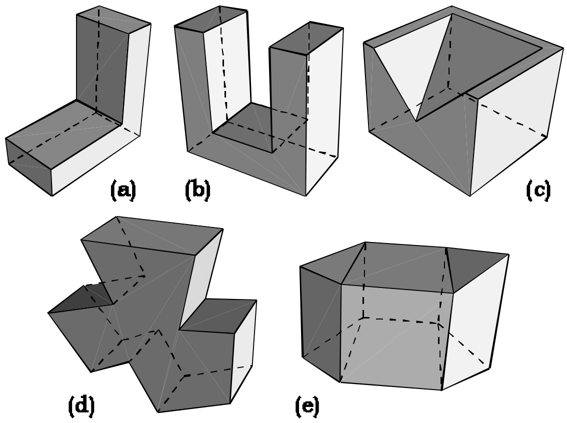
In Fig. 8 we show an example of reconstruction of a concavity. In this case, the complete reconstruction of the visual hull required 16 silhouettes.



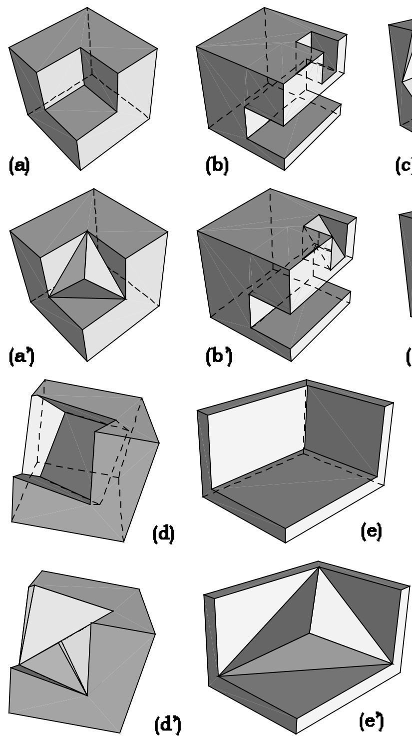
The reconstruction of a concavity
Figure 8

The algorithm has been applied for reconstructing the visual hull of several other concave polyhedra. In Fig. 9 we show some polyhedra coincident with their visual hulls and thus, in principle, exactly reconstructable. Our algorithm was able to reconstruct objects (a),(b), (d) and (e) with 16, 19, 43, 19 silhouettes respectively. The concavity of object (c) requires an infinite number of silhouettes. However, its hard edges, including the rim of the concavity, have been determined with 25 volume intersections.

In Fig. 10 we show several polyhedra ((a)-(e)) not coincident with their visual hulls. In this case, the reconstruction of the visual hulls (a'), (b'), (c'), (e') required 16, 40, 15, 15 silhouettes respectively. To reconstruct completely the visual hull (d') requires infinite steps, but its hard edges and the faces lying on the convex hull took 15 intersections.



Concave polyhedra coincident with their visual hull
Figure 9



Polyhedra not coincident with their visual hulls
Figure 10

5. SUMMARY

We have presented a new a new general approach to interactive, object-specific shape-from-silhouette algorithms. This approach, which holds for completely unknown shapes, is based on a necessary condition for the reconstruction to have been performed with the best possible accuracy.

We have shown that from this condition it is possible to derive a quantitative measure of reconstruction accuracy and rules for finding new viewpoints if the accuracy is not satisfactory.

For demonstrating the effectiveness of our approach, an interactive algorithm has been implemented for polyhedra and applied in a virtual environment.

In the case of convex polyhedra, the average behaviour of the algorithm has been experimentally shown to be close to optimality. The case of concave polyhedra is more complex, and the rules for finding new viewpoints of the algorithm implemented are able to deal with a restricted class of object.

Future work will deal with extending the capabilities of the algorithm NEXT to general non-convex polyhedra, and applying the interactive approach to other categories of objects.

REFERENCES

[Ahuja89] N. Ahuja and J. Veenstra: Generating octrees from object silhouettes in orthographic views, *IEEE Trans. on PAMI*,Vol.11,pp.137-149,1989

[Astro99] 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

[Caval00] L. Cavallero: Ricostruzione interattiva di poliedri, *Master Thesis,* Politecnico di Torino, 2000

[Chian89] C. H. Chian and J. K. Aggarwal: Model reconstruction and shape recognition from occluding contours", *IEEE Trans.on PAMI*, Vol.11, pp.372-389, 1989

[Dobki86] D. Dobkin, H. Edelsbrunner, C. K. Yap: Probing Convex Polytopes, *Proc. 18th ACM Symp. Theory of Computing, 1986, pp. 424-432*

[Laure94] A. Laurentini: The visual hull concept for silhouette-based image understanding," *IEEE Trans. on PAMI*, Vol. 16, No.2,pp. 150-162, 1994

[Laure95] 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

[Laure97] A. Laurentini: How many 2D silhouettes it takes to reconstruct 3D objects, *Computer vision and Image Understanding,* Vol.67, pp.81-87, 1997

[Laure99a] A. Laurentini: Computing the visual Hull of Solids of revolution, *Pattern Recognition,* Vol.32, pp.377-388 , 1999

[Laure99b] A. Laurentini: The visual hull of curved objects, *Proc. Seventh IEEE Int. Conf. on Computer Vision,* pp.356-361, Kerkyra, Sept. 20-27, 1999

[Lavak89] Lavakusha et al.: VI algorithms with changing directions of views*, Proc. MIV-89*, Tokyo, pp. 309-314

[Matus00] W. Matusik, et al.: Image Based Visual Hulls, *Proc. ACM SIGGRAPH 2000*, pp., 2000

[Nobor88] 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

[Petit98] S.Petitjean: A computational geometric approach to visual hull, *Int. J. Of Comput. Geometry and Applications, Vol. 8, No.4, pp.407-436, 1998*

[Potem87] 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

[Shanm91] K. Shanmukh and A.K.Pujari: Volume intersection with optimal set of directions, *Pattern Recognition Letters,* Vol.12, No.3, pp. 165-170, 1991

[Skiena92] S. Skiena: Interactive reconstruction via geometric probing, *IEEE Proc.,*Vol.80, No.9, pp1364-1383, 1992

[Sulli98] S. Sullivan and J. Ponce:Automatic model construction and pose estimation from photographs using triangular splines, *IEEE Trans. on PAMI*,Vol.20,No.10,pp.1091-1097, 1998

[Vaill92] R. Vaillant and O. Faugeras: Using extremal boundaries for 3D object modeling, *IEEE Trans. on PAMI*, Vol. 14, no.2, pp.157-173,1992

[Zheng94] J. Y. Zheng: Acquiring 3D models from sequences of contours, *IEEE Trans. on PAMI*, Vol. 16, no.2, pp.163-178,1994