Laboratoire Bordelais de Recherche en Informatique
UMR 5800 - Université Bordeaux I, 351, cours de la Libération,
33405 Talence CEDEX, France

Research Report `RR-1357-05`

Reconstruction of topology valid boundary of discrete object from $3D$ range images

by Stefka GUEORGUIEVA and Pascal DESBARATS

May, 2005

Reconstruction of topology valid boundary of discrete object from $3D$ range images

Stefka GUEORGUIEVA and Pascal DESBARATS

Abstract

Isosurface extraction algorithms are widely used to generate meshes from range data. In order to describe objects in detail, huge meshes are used and often user interactiveness and fast rendering are privileged over description precision and validity. However, in a great number of applications as in immersion systems for surgery simulation for example, validity of the resulting surface avoiding crack pathing and triangle degeneracies is primordial.
In the present paper, a topology based approach is proposed in order to ensure surface reconstruction mesh integrity. The geometric embedment of the underlying data structure is evaluated over finite precision geometric computations and different surface reconstruction and representation resolution.

Keywords
discrete objects, boundary surface reconstruction, controlled topology simplifying, isosurface extraction, marching cube, triangular meshes, multiresolution

Contents

1 State of the art

We are interested in boundary reconstruction of discrete objects defined by $3D$ range images. In contrast with approximation methods as [9], initial data are produced by range scanning systems and correspond typically to rectangular grids of resolution (cell grid size) sensitive to the object being scanned. A popular technique is to reconstruct object boundary using boundary surface sampled values in the acquisition vertex grid. Simple and efficient methods as the marching cube and the marching tetrahedra, are reported and evaluated with respect to the disambiguation and the reparation of artifacts in the reconstructed boundary surface [13, 4, 19, 5, 11, 12]. In a more general context, a great number of methods for polygonal meshes reparation and simplification use voxel based approach, see for example [10, 2, 1, 15]. Often, objectives of the boundary surface reconstruction design are application dependant and the graphical visual expression is privileged. However, incorrect topology and artifacts become easily perceived and lead to erroneous interpretations under closer examination [6, 8]. A common feature to most of the disambiguation methods is the refinement and the extension of the lookup table introducing the classification and the treatment of the different cell vertex arrangements. In the present paper we propose a new approach for disambiguation of the sampled data. The elaborated method is based on the natural incorporation of the neighbourhood information, available as well in the initial step in the acquisition range data as in the computed reconstruction grid. According to acquisition data, a vertex classification in Black/White/Gray is first performed. Then a Gray vertex reclassification in Black or White is carried on following an appropriated vertex neighbourhood analysis and thus the intersection point [6] is computed. This allows on one hand preprocessing of the ambiguities and on the other hand the final topological validity is achieved within the reconstruction grid at a given resolution. This approach is particularly useful for data where re-sampling is not available and takes best advantage of initial acquisition range image data. Besides, the method is developed under the assumption of variable reconstruction resolution including non-integer cell size factor in such a way that the resulting boundary surface reconstruction is topologically correct in the terms of the chosen resolution. This property supports adaptive boundary reconstruction , see [14, 16, 18], an open issue in process of our current research and experimentation.

2 Boundary reconstruction

2.1 $2D$ boundary reconstruction algorithm

1. Acquisition range $2D$ data preprocessing: construction of an array $Ac[i][j], 0 \leq i < n, 0 \leq j < m$, each element being a pixel centered at integer coordinate points $P_{i,j}$ where $Ac[i][j] = 1$ iff $P_{i,j}$ belongs to the object, $Ac[i][j] = 0$ otherwise. The illustration is given in Fig.1

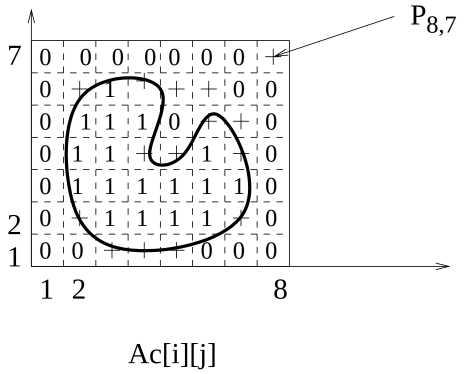
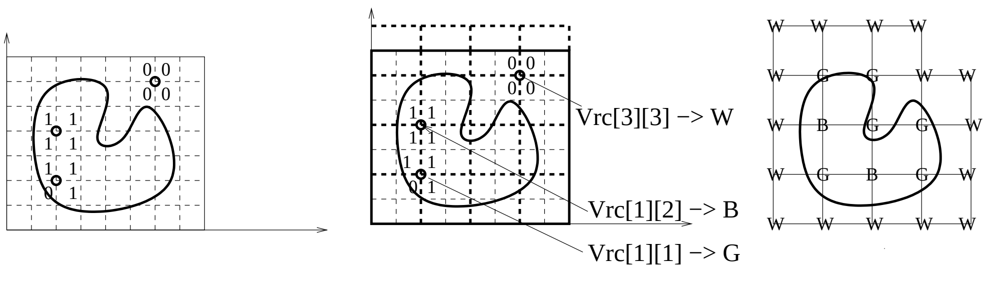


Figure 1: Acquisition range $2D$ data grid Ac.

2. Cover the acquisition range $2D$ data array Ac with an array of cells $Rc[i][j], 0 \leq i < p, 0 \leq j < q$.

   In fact, at this point, one should define the resolution (the precision) of the contour reconstruction. Each cell of Rc is covered by a set of Ac cells, centered at a given position and with a given size (not obligatory a multiple of the Ac cell size).
   As for example, the Rc cell grid in Fig.2(b) is simply a double size resampling of the Ac grid in Fig.2(a).



(a) Ac[i][j] (b) Rc[i][j] (c) Brc[i][j]

Figure 2: Reconstruction $2D$ data grid Rc.

3. Classify each vertex $Vrc[i][j]$ in Rc as black (B), white (W) or gray (G) according to the acquisition $2D$ data Ac. There are three possibilities:

   (a) If $Vrc[i][j]$ is in the interior of an acquisition $2D$ data grid cell $Ac[k][l+1]$ then $Vrc[i][j]$ is classified as B if $Ac[k][l+1] = 1$ and $Vrc[i][j]$ is classified as W otherwise ($Ac[k][l+1] = 0$), see Fig.3(a).

(b) If $Vrc[i][j]$ is on an edge common to two cells $Ac[k][l]$ and $Ac[k][l+1]$, see Fig.3(b), then $Vrc[i][j]$ is classified as B if $Ac[k][l] = Ac[k][l+1] = 1$, $Vrc[i][j]$ is classified as W if $Ac[k][l] = Ac[k][l+1] = 0$, and otherwise $Vrc[i][j]$ is classified as G.

(c) If $Vrc[i][j]$ is on a corner, i.e. common to four cells as in Fig.3(c). The vertex $Vrc[i][j]$ is classified as B if $Ac[k][l] = Ac[k][l+1] = Ac[k+1][l] = Ac[k+1][l+1] = 1$. The vertex $Vrc[i][j]$ is classified as W if $Ac[k][l] = Ac[k][l+1] = Ac[k+1][l] = Ac[k+1][l+1] = 0$. Otherwise, $Vrc[i][j]$ is classified as G.

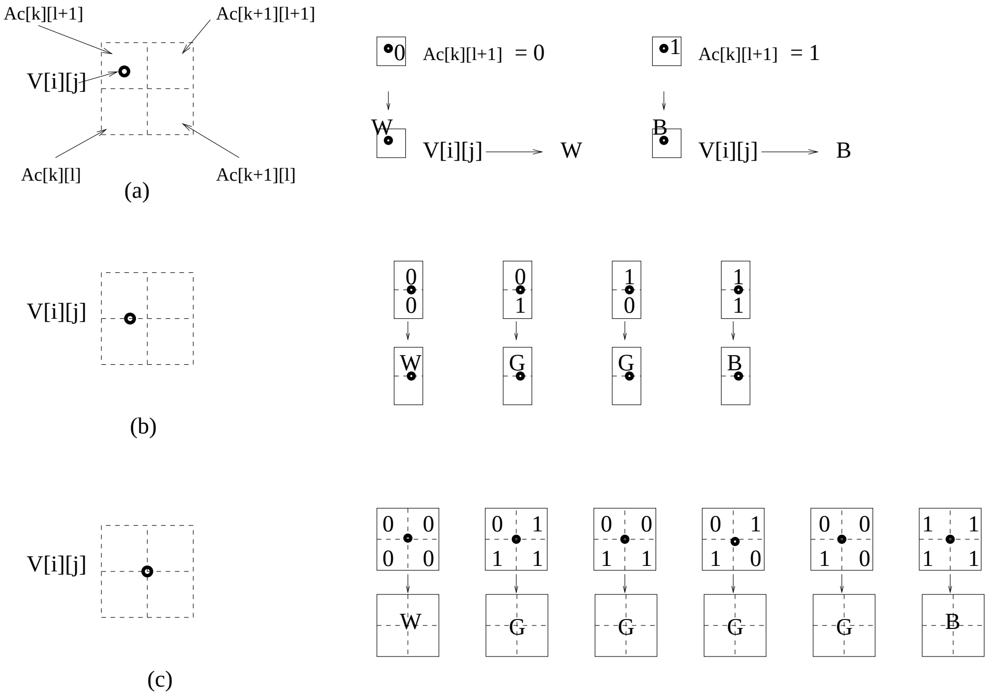
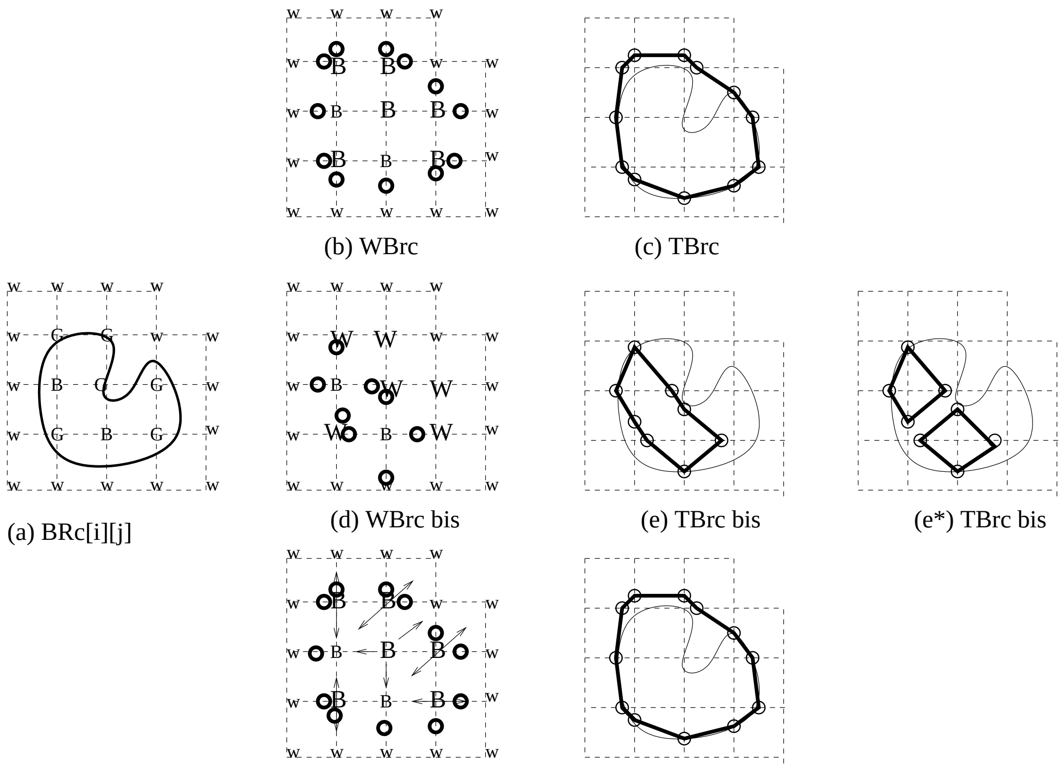


Figure 3: Classification of the reconstruction $2D$ data grid Rc vertices according to the acquisition $2D$ data Ac.

4. Extract cells $BRc[i][j]$ in Rc that include heterogeneously classified vertices $Vrc[i][j]$. These cells contain the object boundary.
   For the above example, the illustration is given in Fig.2(c).

5. Construct $WBrc$ including all vertices from $BRc[i][j]$ with gray vertices in $BRc[i][j]$ reclassified as B or W in $WBrc$. See the example in Fig.4. The reclassification preserve consistent boundary orientation (see[17]).

   Gray vertex reclassification is inspired from Tilove's classification function [17] and is based on the vertex local neighbourhood analysis.



(b) WBrc (c) TBrc

(a) BRc[i][j] (d) WBrc bis (e) TBrc bis (e*) TBrc bis

(f) WBrc ter (g) TBrc ter

Figure 4: Reconstruction $2D$ data gray vertices reclassification.

Look at the example in Fig.5 where the intersection calculation of the ray r and the oriented polygon p is illustrated. Suppose that the oriented distance from points, lying in the half-plane pointed by the normal vector n, to r, is positive, while for points in the opposite half-plane, is negative. Points with positive oriented distance are classified as B, as vertices 7 and 9. Points with negative oriented distance are classified as W, as vertices 1 and 11. Finally, points collinear to the ray r are classified as G as the vertices $2, 3, 5, 6$ and 8 . In an intersection point, the sign of the oriented distance of points belonging to the polygon and lying in different half-planes, should change. Thus the vertices $2, 3$ and 8 are rejected. Vertex 10 is retained as an intersection point. For the vertices 5 and 6, as long as the entire edge $(5, 6)$ is on r, just one intersection is enumerated. For a total of two intersection points, one can consider that the ray once enters into p and once leaves it. If we approximate the polygon with a max[/min] bounding box the intersection are resp. in P and Q, and $P*$ and $Q*$. In this way we can simplify the oscillation of p while preserving the number of intersection points.
For our purpose, the classification swap, B to W (resp. W to B) is used as a criterion for object boundary crossing and thus for boundary point detection and reconstruction.

This classification method is applied for gray vertices reclassification in BRc. For each G vertex $Vrc[i][j]$, a neighbourhood $N_{Vrc}[i][j]$ compound by its $8-$*connected* neighbours $\{Vrc[i-1][j-1], Vrc[i][j-1], Vrc[i+1][j-1], Vrc[i-1][j], Vrc[i+1][j], Vrc[i-1][j+1], Vrc[i][j+1], Vrc[i+1][j+1]\}$ is defined as illustrated in Fig.6. One can consider that $N_{Vrc}[i][j]$ includes on one hand the closest neighbours of $Vrc[i][j]$, the $4-$ *connected* neighbours, and on the other hand, the most distant neighbours, the extremities of the bounding box including the four cells adjacent to $Vrc[i][j]$. Thus if boundary crossing occurs, then it should be detected through a classification swap of corresponding neighbours from the closest and/or the remotest "envelope" of $Vrc[i][j]$ in $N_{Vrc}[i][j]$.

The reclassification algorithm follows:
For each gray vertex $Vrc[i][j]$ in BRc examine the neighbourhood:

(a) $Vrc[i][j]$ is reclassified as B if all of its neighbourhood vertices are B. Indeed, no change in the sign occurs (see Fig.6(a)).

(b) By analogy, $Vrc[i][j]$ is reclassified as W if all of its neighbourhood vertices are W (see Fig.6(b)).

(c) $Vrc[i][j]$ is reclassified as B if there exists at least one couple of neighbourhood vertices: $\{(Vrc[i-1][j], Vrc[i+1][j]), (Vrc[i][j-1], Vrc[i][j+1]), (Vrc[i-1][j+1], Vrc[i+1][j-1]), (Vrc[i-1][j-1], Vrc[i+1][j+1])\}$ such that the members of the couple have B and W (resp. W and B) classification (see Fig.6(c)). In other words, a boundary intersection is supposed to occur in $Vrc[i][j]$.

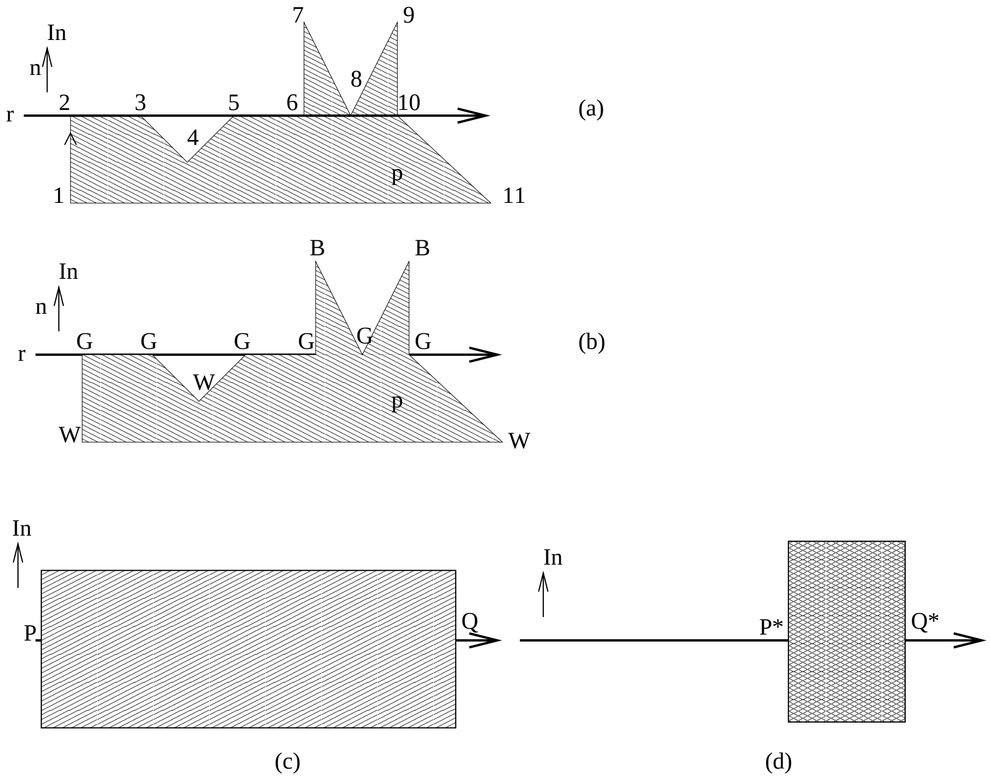


Figure 5: Boundary oscillation simplification.

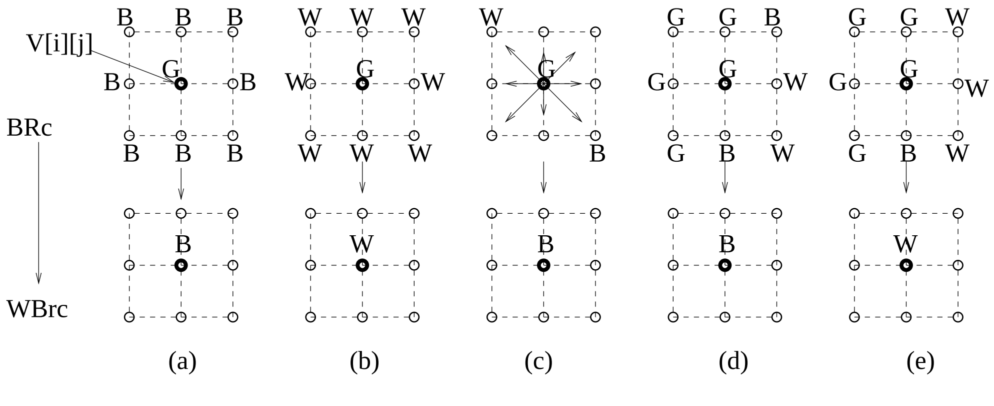


Figure 6: Reclassification of the reconstruction $2D$ data grid vertices according to the reconstruction $2D$ data grid BRc.

(d) $Vrc[i][j]$ is reclassified B if the number of B vertices in its neighbourhood is superior or equal to the number of W vertices (see Fig.6(d)).

(e) $Vrc[i][j]$ is reclassified W if the number of W vertices in its neighbourhood is superior to the number of W vertices (see Fig.6(e)).

It should be noted that vertex reclassification according to local reconstruction information is a primordial feature:
Indeed, object topology and characteristics could variate: non-manifold versus manifold, simply connected or not, etc.
Two paradigms interfere:

(a) The variation of the resolution of the reconstruction. In Fig.7 there is an one-to-one correspondence i.e. the reconstruction cell size is equal to the acquisition cell size. While, in Fig.8, the reconstruction cell size is equal to four times the acquisition cell size. In both cases, acquisition grid and reconstruction grid have the same origin.

With a constant gray vertex reclassification strategy, when all G vertices are reclassified in B vertices, the first case gives an object homomorphic to an annulus (see Fig.7(e,f)) while in the second case (see Fig.8(d)) the reconstructed object is homomorphic to a disc.

(b) The vertex reclassification strategy.
With a constant resolution (see Fig.7) different vertex reclassification strategies are illustrated:
-for the example in Fig.7(e,f), all G vertices are reclassified as B;
-for the example in Fig.7(g,h), all G vertices are reclassified as W;
-for the example in Fig.7(i,j), all G vertices are reclassified according to neighbourhood analysis (Fig.6);
In the first and last reconstructions, the final object is homomorphic to an annulus while in the second one, the object is compound by two parts each one homomorphic to a disc.

6. Proceed all cells $WBrc$ associating to each one $WBrc[i][j]$ of them a set of line segments $TBrc[i][j]$ that approximate the part of the object boundary that is contained in this cell according to the marching square look-up table illustrated in Fig.9.

   An illustration for the boundary reconstruction of the example of Fig.2 is given in Fig. 10.

7. Traverse $TBrc[i][j]$ to check up the validity of the reconstructed object.
   Validity is evaluated in terms of the number of boundary connected components. Several boundary components could occur either when there are holes through the reconstructed object or when the object is a multi-shell one. In some cases these should be corrected through a reevaluation of the ambiguous cases in the marching square algorithm and replacement of the corresponding $TBrc[i][j]$ with the alternative choice.
   For the $2D$ marching square method, ambiguities arise as illustrated in

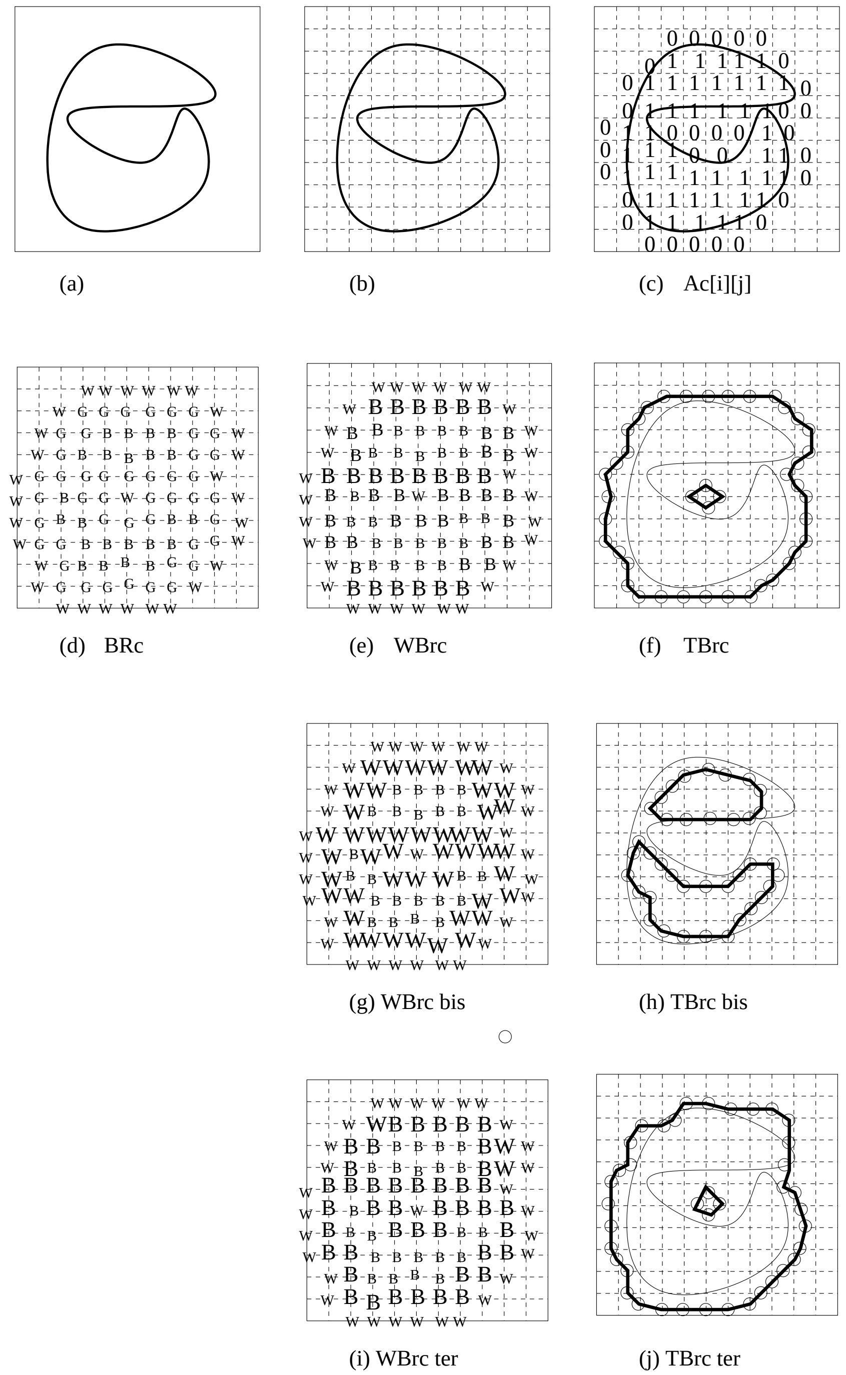


Figure 7: Boundary reconstruction with acquisition range $2D$ data precision.

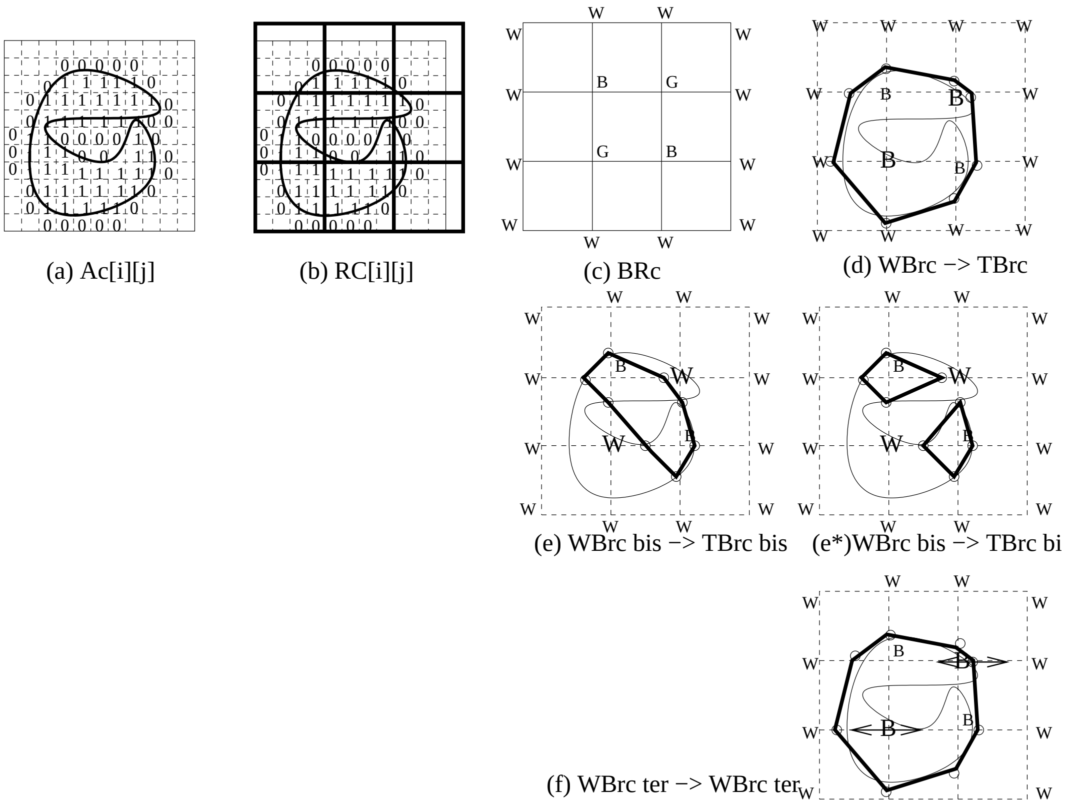


Figure 8: Boundary reconstruction with precision equal to four times the acquisition range $2D$ data grid.

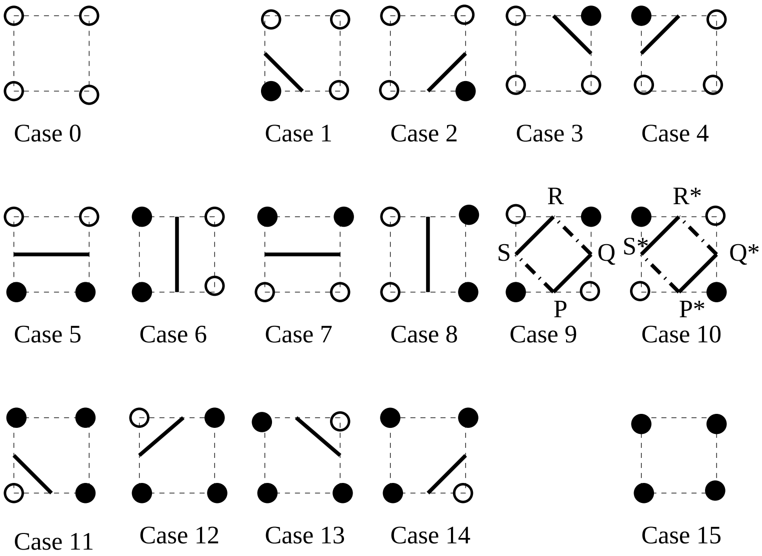
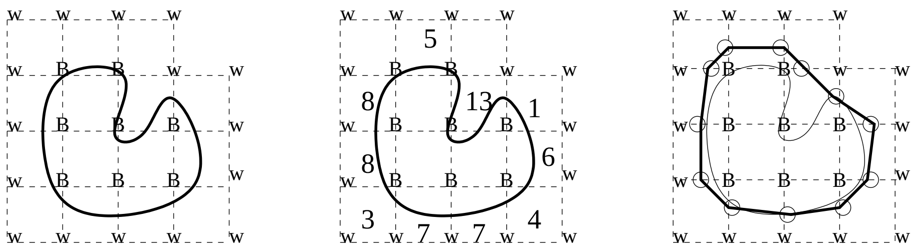


Figure 9: Marching square case analysis.



(a) WBrc[i][j]

(b) Marching square case cell classification

(c) TBrc[i][j]

Figure 10: Marching square contour reconstruction.

cases 9 and 10 in Fig.9. The first possible boundary reconstruction for the case 9 is PQ and RS shown with continuous lines. The second one is QR and SP drawn with dashed lines.
Different object reconstructions, due to these ambiguities, are given in Fig.4(e,e*) and Fig.8(e,e*). Additional constraints on the reconstructed object should be provided in order to invalidate the unworkable shape.

8. Output final $TBrc[i][j]$ list.

2.2 $3D$ boundary reconstruction algorithm

1. Acquisition range $3D$ data preprocessing: construction of an array $Ac[i][j][k], 0 \leq i < p$, $0 \leq j < q$, $0 \leq k < r$ each element being a voxel centered at integer coordinate points $P_{i,j,k}$ where $Ac[i][j][k] = 1$ iff $P_{i,j,k}$ belongs to the object, $Ac[i][j][k] = 0$ otherwise. The illustration is given in Fig.11

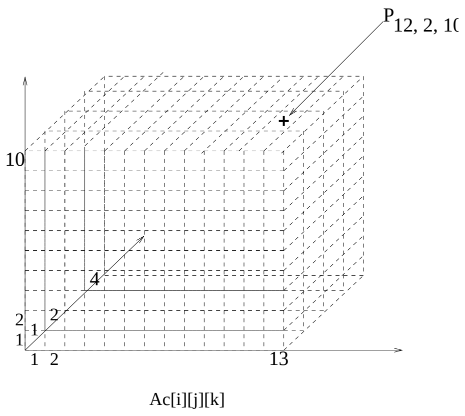


Figure 11: Acquisition range $3D$ data grid Ac.

2. Cover the acquisition range $3D$ data array Ac with an array of cells $Rc[i][j][k], 0 \leq i < p, 0 \leq j < q, 0 \leq k < r$ (see Fig.12) with a specified resolution of the reconstruction.

3. Classify each vertex $Vrc[i][j][k]$ in Rc as black (B), white (W) or gray (G) according to the acquisition data Ac. There are four possibilities:

   (a) If $Vrc[i][j][k]$ is in the interior of an acquisition data grid cell then $Vrc[i][j][k]$ is classified as B if this cell belongs to the object and $Vrc[i][j][k]$ is classified as W otherwise see Fig.13(a).

   (b) If $Vrc[i][j][k]$ is on a face common to two acquisition data grid cells (see Fig.13(b)) then $Vrc[i][j][k]$ is classified as B if both cells belong to the object. The vertex $Vrc[i][j][k]$ is classified as W if both cells are outside the object, otherwise $Vrc[i][j][k]$ is classified as G.

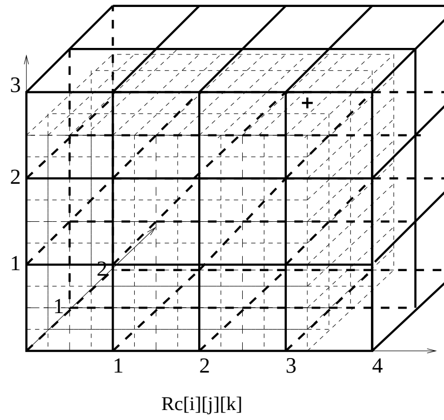


Figure 12: Reconstruction 3D data grid Rc.

(c) If $Vrc[i][j][k]$ is on an edge common to four acquisition data grid cells (see Fig.13(c)) then if all adjacent cells belong to the object, $Vrc[i][j][k]$ is classified as B. If all adjacent cells do not belong to the object, $Vrc[i][j][k]$ is classified as W, otherwise $Vrc[i][j][k]$ is classified as G.

(d) If $Vrc[i][j][k]$ is on a corner, i.e. common to eight cells as in Fig.13(d). By analogy, if all of them are part of the object, $Vrc[i][j][k]$ is classified as B, if all of them are outside the object, $Vrc[i][j][k]$ is classified as W. Otherwise, $Vrc[i][j][k]$ is classified as G.

4. Extract cells $BRc[i][j][k]$ in Rc that include heterogeneously classified vertices $Vrc[i][j][k]$.

5. Construct $WBrc$ including all vertices from $BRc[i][j][k]$ with gray vertices in $BRc[i][j][k]$ reclassified as B or W in $WBrc$. For each gray vertex $Vrc[i][j][k]$ there are 24 neighbour vertices in BRc as illustrated in Fig.14(a). A neighbourhood $N_{Vrc}[i][j][k]$ of the vertex $Vrc[i][j][k]$ is defined as shown in Fig.14(b), compound by the set of the $6-\textit{connected}$ neighbours of $Vrc[i][j][k]$, see Fig.14(c), and the extremities of the rectangular bounding box including the four cells adjacent to $Vrc[i][j][k]$, Fig.14(d). The intuition, as in the $2D$ case, is to incorporate in $N_{Vrc}[i][j][k]$ on one hand an inner boundary shell level (the closest to $Vrc[i][j][k]$ neighbours, the $6-\textit{connected}$ neighbours) and an outer level (the most distant neighbours), and to examine the object behaviour in between: if there exists a piece of boundary in the vicinity of $Vrc[i][j][k]$ then a classification swap in opposite vertices in $N_{Vrc}[i][j][k]$ should occur. Otherwise, one should correct possible range data errors or flatten the boundary surface oscillation. The reclassification algorithm of the $3D$ data grid BRc vertices follows:

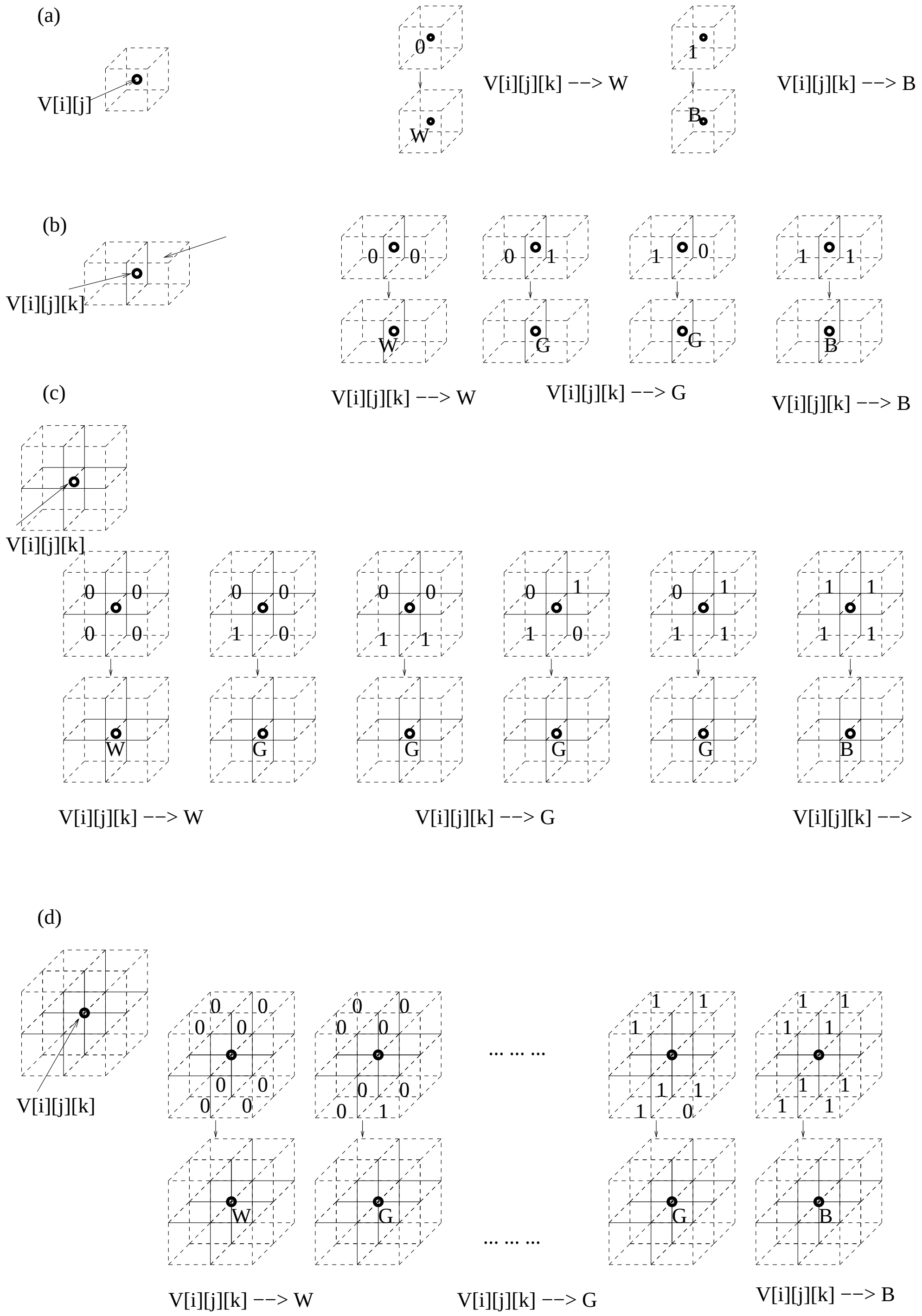


Figure 13: Classification of the reconstruction $3D$ data grid Rc vertices according to the acquisition $3D$ data Ac.

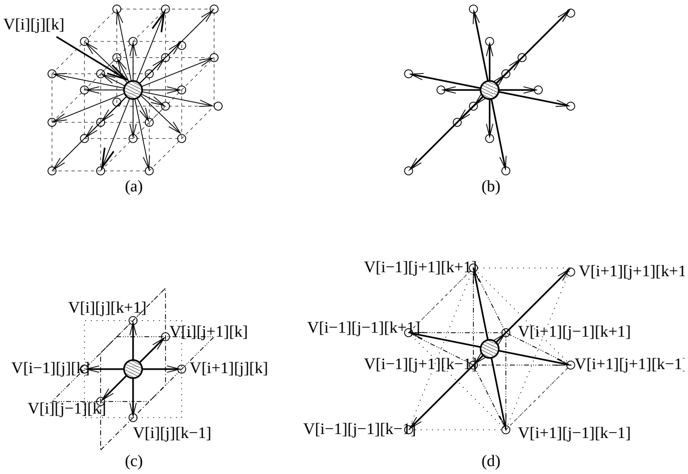


Figure 14: Vertex neighbourhood definition in the $3D$ data grid BRc.

For each G vertex $Vrc[i][j][k]$ in BRc examine its neighbourhood $N_{Vrc}[i][j][k]$:

(a) $Vrc[i][j][k]$ is reclassified as B if all vertices in $N_{Vrc}[i][j][k]$ are B. Indeed, as long as no change in the sign occurs (see Fig.15(a)), no boundary detection is supposed.

(b) By analogy, $Vrc[i][j][k]$ is reclassified as W if all vertices in $N_{Vrc}[i][j][k]$ are W (see Fig.15(b)).

(c) $Vrc[i][j][k]$ is reclassified as B if there exists at least one couple of opposite vertices in $N_{Vrc}[i][j][k]$, shown in Fig.14(a) and Fig.14(d) such that the members of the couple have B and W (resp. W and B) classifications (see Fig.15(c)). In other words, a boundary cross is encountered and the intersection point is approximated with $Vrc[i][j][k]$.

(d) $Vrc[i][j][k]$ is reclassified as B if the number of B vertices in $N_{Vrc}[i][j][k]$ is superior or equal to the number of W vertices (see Fig.15(d)). This operation supports a kind of boundary smoothing in vicinity of $Vrc[i][j][k]$

(e) $Vrc[i][j][k]$ is reclassified as W if the number of W vertices in $N_{Vrc}[i][j][k]$ is superior to the number of W vertices (see Fig.15(e)).

By analogy with the $2D$ case, depending on the precision of the reconstruction grid and the vertex reclassification strategy, variations in the reconstructed boundary could occur.

Moreover, we have experienced several reclassification strategies based on:

(a) Projection of the neighbourhood $N_{Vrc}[i][j][k]$ on the X axis, denoted as $G \rightarrow_X B/W$.

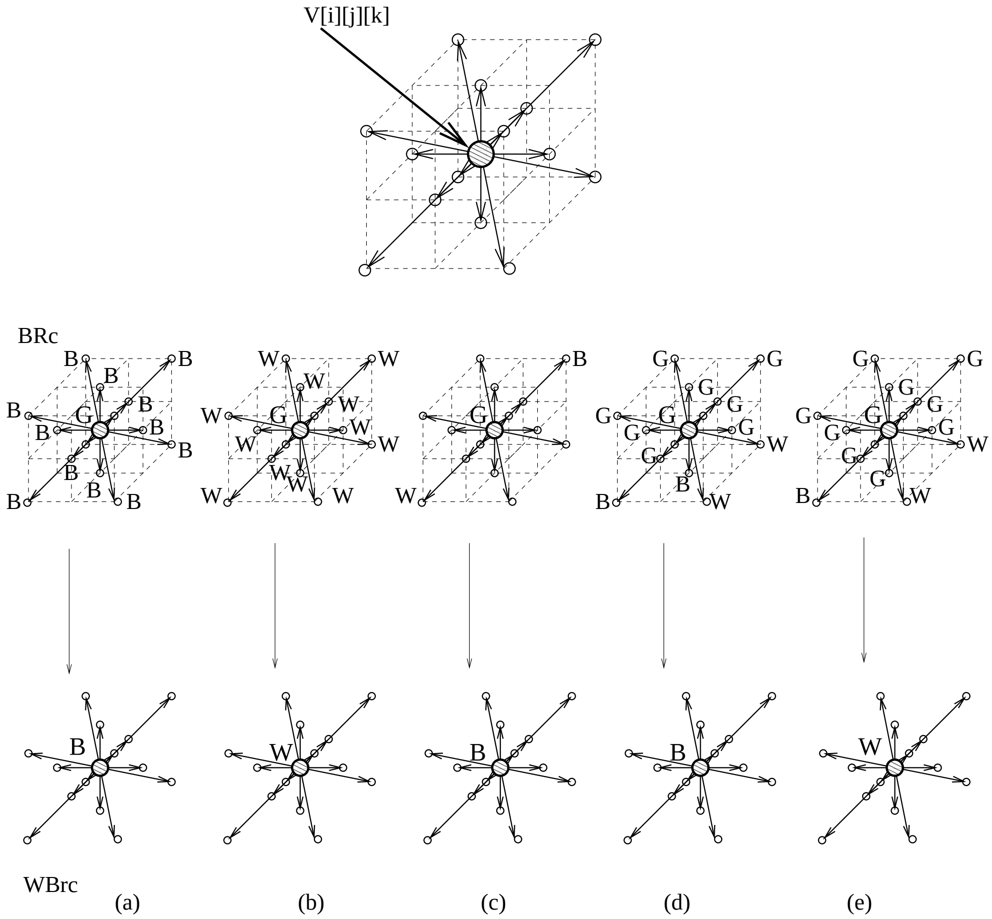


Figure 15: Reclassification of the reconstruction $3D$ data grid vertices according to the vertex neighbourhood.

(b) Projection of the neighbourhood $N_{Vrc}[i][j][k]$ on the Y axis, denoted as $G \rightarrow_Y B/W$.

(c) Projection of the neighbourhood $N_{Vrc}[i][j][k]$ on the Z axis, denoted as $G \rightarrow_Z B/W$.

(d) Classification of all G as B, denoted as $G \rightarrow B$.

(e) Classification of all G as W, denoted as $G \rightarrow W$.

(f) According to the above analysis, denoted as $G \rightarrow N_{Vrc}\ B/W$.

Results for these different strategies are given in Table 1 and Fig. 18.

6. Proceed all cells $WBrc$ associating to each one $WBrc[i][j][k]$ of them a set of triangles $TBrc[i][j][k]$ that approximates the part of the object boundary that is contained in this cell according to the marching cube look-up table illustrated in Fig.16.

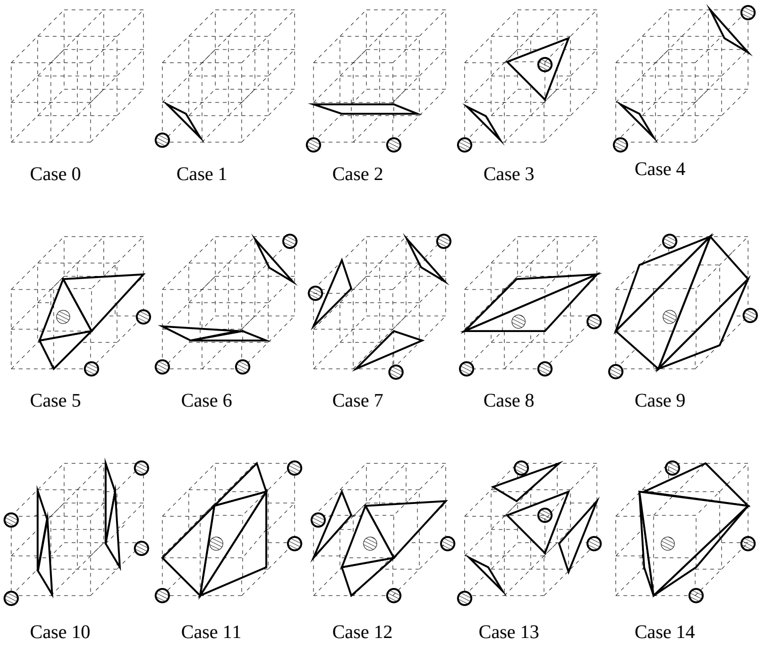


Figure 16: Marching cube case analysis.

7. Traverse $TBrc[i][j][k]$ to check up the validity of the reconstructed object. As in the $2D$ case, reconstruction validity is evaluated in terms of the number of boundary connected components. Several boundary components could be encountered when for example the object is a multi-shell one. In some cases, boundary reconstruction should be corrected through a reevaluation of the ambiguous cases in the marching cube algorithm

and a replacement of the corresponding $TBrc[i][j][k]$ with the alternative choices. An illustration is given in Fig.17 where depending on the interpretation of the cases 6, Fig.17(a), or $6bis$, Fig.17(d) and in combination of the case 3, one could reconstruct a boundary surface piece, Fig.17(c), or invalidate the reconstruction, Fig.17(f): there is no correspondence along the common boundary: edge $P*Q*$ has no matching edge i.e. $P*$ (resp. $Q*$) corresponds to P (resp. Q) and there is no edge between P and Q. Moreover, identification along RS and $R*S*$ creates a non-manifold condition along this edge.

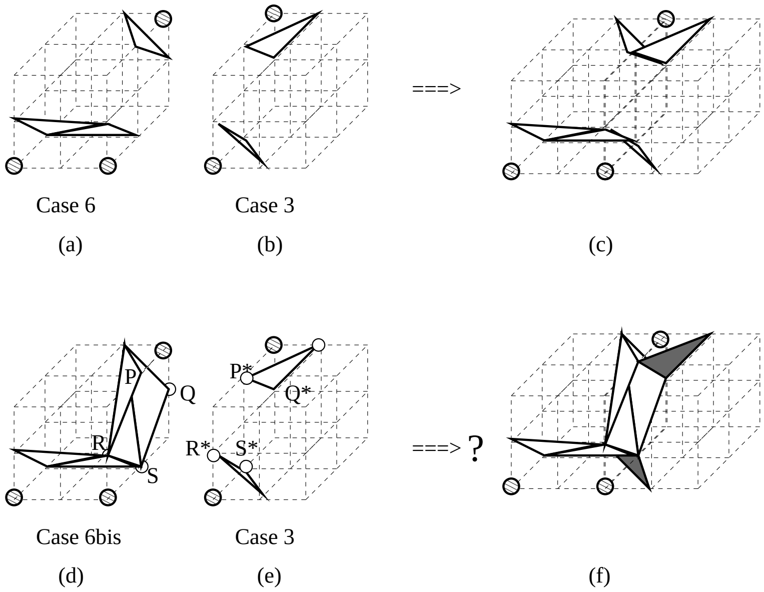


Figure 17: Marching cube ambiguities.

8. Output final $TBrc[i][j][k]$ list.

3 Implementations and Results

A prototyped implementation of the algorithm has been coded in C and evaluated varying algorithm characteristics and acquisition and reconstruction data grid resolutions. All the computation times reported have been obtained using a Pentium 2.80 GHz, 1GB RAM.

The first series of experiences is focused on the choice of the reclassification strategy, used in step 5, and the costs of the filling of holes in the reconstructed boundary surface, step 7 in the proposed algorithm. The range $3D$ image in use is a segmented brain obtained from MR images with a constant acquisition resolution grid $256 \times 256 \times 256$. The cell size of the reconstruction grid RC is

$\Re$	V	T	H	t [s]	T_{cor}	ε	t_{cor} [s]
$G \rightarrow_X B/W$	119490	238466	255	60.70	238976	0.67%	62.10
$G \rightarrow_Y B/W$	119304	237680	192	60.72	238064	0.45%	61.60
$G \rightarrow_Z B/W$	117744	235462	11	60.57	235484	0.35%	61.63
$G \rightarrow B$	121206	242396	6	60.52	242408	3.69%	61.46
$G \rightarrow W$	117784	235544	10	60.69	235564	-3.67%	61.54
$G \rightarrow N_{Vrc}\ B/W$	117784	235544	10	60.79	235564	-3.67%	61.47

Table 1: Results on different gray vertex reclassification strategies

equal to the acquisition one. Table 1 summarizes the results for a variation of the reclassification strategy. The columns sum up two versions of the boundary surface reconstruction: with and without the connected boundary component modification in order to fill the holes in the reconstructed surface (step 7). From column 2 to column 5 the first version is illustrated where the following notations are used:

V denotes the number of vertices, T is the number of triangles, H is the number of holes and t is the processing CPU time in seconds for the reconstruction model calculation without the correction. From column 6 to column 8 the second version is illustrated. The final model has a new triangle subdivision with T_{cor} triangles calculated for t_{cor} seconds. In addition, a volume error indicator ε of the reconstruction is supplied. It is defined as

$$\varepsilon = \frac{Vol_{Rec} - Vol_{Acq}}{Vol_{Acq}} \times 100$$

where Vol_{Acq} and Vol_{Rec} are respectively the volume of the acquisition data and the volume of the reconstructed object. We have drawn the following conclusions:

Boundary discontinuities and their correction are of limited impact. For the examples above in the worst case 0.10% of the faces are degenerated ie are reconstructed as belonging to separated boundary connected components and it takes 2.25% of the processing time to fill them. On the contrary, according to the gray vertex reclassification, the volume error indicator, ratio between the volume of the the reconstructed object and the acquisition one, has an amplitude of 7.36%. Thus an optimization of the reclassification strategy, could significantly improve the boundary reconstruction quality in terms of representational exactness ie correspondence with the original acquisition range $3D$ data with no additional cost for topological validity guarantee. Further more, as one can see the best reconstruction according to the error volume indicator is the one using the reclassification strategy $G \rightarrow_Z B/W$, while looking in Fig.18, the best visually smooth reconstruction corresponds to $G \rightarrow N_{Vrc}\ B/W$.

Next, in table 2, the performance of the proposed approach is compared with respect to the variation of the reconstruction data grid resolution ρ_{rec} for the same

ρ_{rec}	V	T	H	t [s]	T_{cor}	ε	t_{cor}
0.85	162978	325924	10	96.15	325944	-3.65%	97.17
1	117784	235544	10	60.69	235564	-3.67%	61.54
2	29006	57950	29	12.92	58008	-4.89%	13.37
4	7162	14296	12	6.80	14320	-4.84%	7.04

Table 2: Results on the variation of the reconstruction data grid RC

Object	aneurysm	bunny	dragon	teapot
$G \rightarrow_X B/W$	0.14%	5.88%	3.29%	0.09%
$G \rightarrow_Y B/W$	0.19%	6.43%	3.50%	0.14%
$G \rightarrow_Z B/W$	1.16%	6.68%	3.75%	0.88%
$G \rightarrow B$	5.10%	12.35%	7.73%	3.55%
$G \rightarrow W$	-5.01%	-13.28%	-9.01%	-3.49%
$G \rightarrow N_{Vrc}\ B/W$	-5.01%	-13.08%	-12.68%	-3.49%

Table 3: Results on volumetric and synthetic range data

brain image and the reconstruction strategy $G \rightarrow N_{Vrc}\ B/W$. The behaviour of the algorithm is stable under variation of reconstruction cell size. Furthermore, this experiment shows that a reduction of 95% of triangles corresponds of a reasonable augmentation of 1.19% of the volume error indicator.

Finally, table 3 summarizes volume error indicator variations depending on the nature of the acquisition data: the aneurysm and the teapot range images are raw volumetric data while the bunny and the dragon are discretized from polygonal models. Our approach has better performances on raw discrete objects. A graphical illustration of the discrete reconstructed boundaries is given in Fig.19

4 Conclusion

A topology simplification method [7] for boundary reconstruction of discrete objects from range $3D$ images is proposed. Two basic enhancements are elaborated.

First, in contrast with known methods for valid boundary reconstruction, solutions to topological problems are looked for in a preprocessing step through a reconstruction grid vertices reclassification using a vertex neighbourhood analysis. Reconstructed boundary surface topology is coherent with the chosen

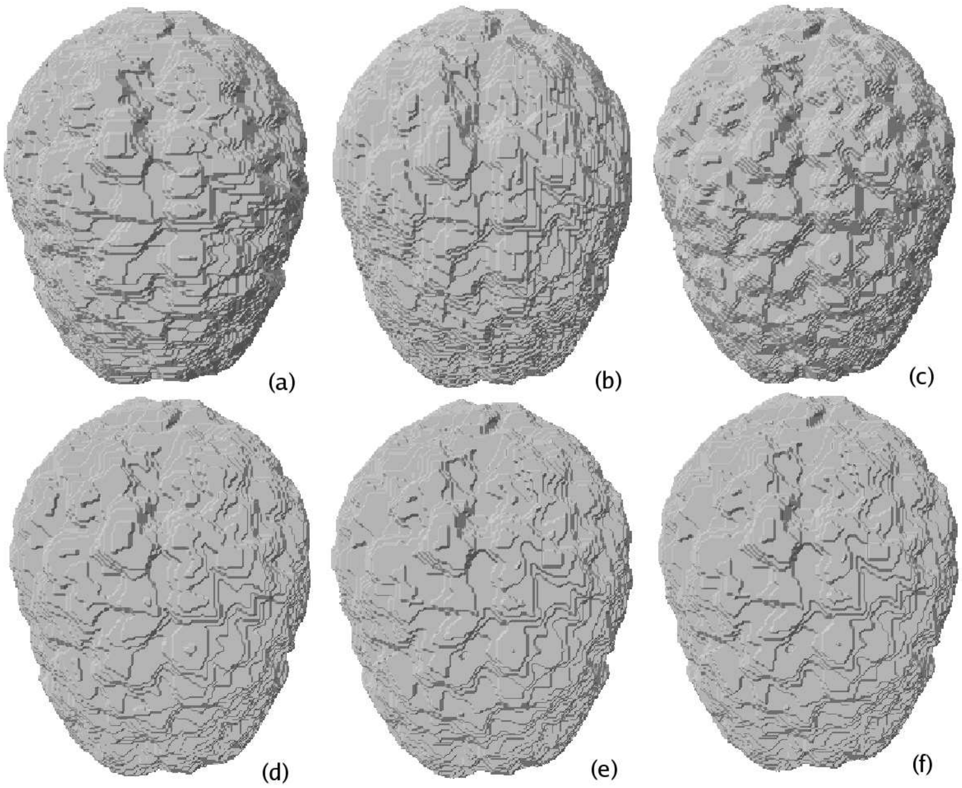


Figure 18: Visual results on different gray vertex reclassification strategies

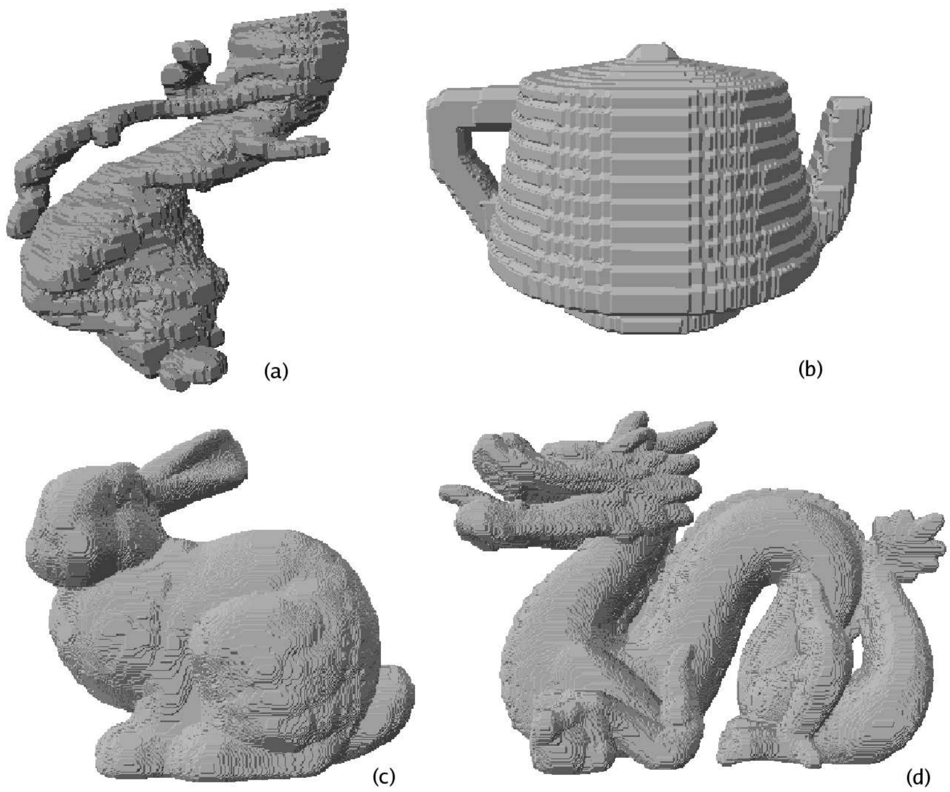


Figure 19: Visual results on volumetric and synthetic discrete objects. (a) Aneurysm ($202 \times 470 \times 210$ voxels), (b) Teapot ($200 \times 152 \times 122$ voxels), (c) Bunny ($322 \times 319 \times 252$ voxels), (d) Dragon ($420 \times 299 \times 194$ voxels).

reconstruction grid resolution and vertex reclassification strategy.
Second, the use of reconstruction surface grid with variable size including a non-integer size cell factor. This allows a great flexibility and a kind of zoom-in and zoom-out operations on the acquisition range $3D$ data.
An immediate extension of this research is the use of different free source extended Marching Cube implementations (see [12] "the Marching Cube 33" for example) in the boundary reconstruction stage of our approach for a better evaluation of the error measurement (as the quantitative and the qualitative aspects of error measurement [8]).
Further, an adaptive reconstruction grid definition will be experienced to get better control on model topology validity.

Acknowledgments
We wish to thank to Fabien Boismoreau, Alexis Derrien and Julien Fleury for their programming and helpful suggestions on the implementation of the current research [3].

References

[1] C. Andujar, P. Brunet, and D. Ayala. Topology-reducing surface simplification using a discrete solid representation. *IEEE Transactions on Graphics*, 21(2):1–18, April 2002.

[2] G. Barequet, C. Duncan, and S. Kumar. Rsvp: A geometric toolkit for controlled repair of solid models. *IEEE Transactions on Visualization and Computer Graphics*, 4(2):162–177, April-June 1998.

[3] F. Boismoreau, A. Derrien, and J. Fleury. Robustess and validity of $3d$ reconstruction. Master report, University of Bordeaux 1, 2005. MM2.

[4] E.V. Chernyaev. Marching cubes 33: Construction of topologically correct isosurfaces. In *GRAPHICON*, 1995.

[5] P. Cignoni, F. Ganovelli, C. Montani, and R. Scopigno. Reconstruction of topological correct and adaptive trilinear isosurfaces. *Computer & Graphics*, 24:399–418, 2000.

[6] A. Gelder and J. Wilhelms. Topological considerations in isosurface generation. *IEEE Transactions on Graphics*, 13(4):337–375, October 1994.

[7] T. Gerstner and R. Pajarola. Topology Preserving and Controlled Topology Simplifying Multiresolution Isosurface Extraction. In T. Ertl, B. Hamann, and A. Varshney, editors, *Proceedings IEEE Visualization 2000*, pages 259–266. IEEE Computer Society Press, 2000. (also as SFB 256 report 31, Univ. Bonn, 2000).

[8] T. Gerstner, M. Rumpf, and U. Weikard. Error indicators for multilevel visualization and computing on nested grids. *Computer & Graphics*, 24:363–373, 2000.

[9] H. Hoppe, T. DeRose, T. Duchamp, and W. MsDonald, J. andStuetzle. Surface reconstruction from unorganized points. In *Computer Graphics SIGGRAPH*, volume 26(2), pages 71–78, 1992.

[10] O. Lachaud. Topologically defined isosurfaces. Rr, Laboratoire d'Informatique du Parallélisme, ENS Lyon, 1996. 96-20.

[11] H. Lee, J. Lee, and H.S. Yang. Real-time lod: Marching-cube-and-octree-based 3d object lod modeling. Tech report, Virtual Reality Research Center, 2002. 2002-77.

[12] T. Lewiner, H. Lopes, AW. Vieira, and G. Tavares. Efficient implementation of marching cubes with topological guarantees. *Journal of Graphic Tools*, December 2003.

[13] W.E. Lorensen and H.E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH'87*, 21(4):163–170, 1987.

[14] M. Lounsbery, T. Derose, and J. Warren. Multiresolution analysis for surfaces of arbitrary topological type. *Transactions on Graphics*, 16(1):34–73, 1997.

[15] F.S. Nooruddin and G. Turk. Simplification and repair of polygonal models using volumetric techniques. *IEEE Transactions Visualization and Computer Graphics*, 9(2):191–205, April-June 2003.

[16] M. Ohlberger and M. Rumpf. Adaptive projection operators in multiresolution scientific visualisation. *IEEE Transactions on Visualization and Computer Graphics*, 4(4):343–363, October-December 1998.

[17] R. Tilove. Set membership classification: A unified approach to geometric intersection problems. *IEEE Transactions on Computers*, C-29(10):874–883, October 1980.

[18] G. Varadhan, S. Krishnan, TVN. Sriram, and D. Manocha. Topology preserving surface extraction using adaptive subdivision. In *Eurographics Symposium on Geometry Processing*, 2004.

[19] Y. Zhou, W. Chen, and Z Tang. An elaborated ambiguity detection method for constructing isosurfaces within tetrahedral meshes. *Computer & Graphics*, 19(3):355–364, 1995.