The Visual Computer

Original Articles

Three-dimensional reconstruction of geoscientific objects from serial sections

Malcolm J. Herbert[1,*], Christopher B. Jones[2], Douglas S. Tudhope[2]

[1] Department of Infomatics, University of Wales, Lampeter, Dyfed, Wales SA48 7ED, United Kingdom
[2] Department of Computer Studies, University of Glamorgan, Pontypridd, Mid Glamorgan, Wales, CF37 1DL, United Kingdom

Serial section reconstruction is widely used for visualising complex three-dimensional objects, but little research has been applied to modelling geoscientific applications. We review previous work and highlight the correspondence problem, particularly important in reconstructions from geoscientific data. We propose an automatic solution to the correspondence problem, based on a minimum-spanning-tree algorithm. The improved results stem from the use of topological information to help decide which edges appear in the final correspondence graph. We then reconstruct some invertebrate fossil samples, before outlining future possibilities in deriving solutions for complex samples, using richer information for each specimen.

Keywords: Serial sections – Three-dimensional modelling – Palaeontological reconstruction – Contour matching

*e-mail: malc@lampeter.ac.uk
Correspondence to: M.J. Herbert

1 Introduction

A three-dimensional object is commonly sampled by a set of parallel planar sections, in which closed contour loops define the intersection between these sections and both the internal and external features of the object. Examples of section data can be provided by computer-assisted tomography (CAT) in medical imaging (Marko et al. 1990; Rhodes 1990), geophysical surveys in subsurface exploration (Dabek et al. 1988) and serial grinding techniques in palaeontology (Ager 1965).

Although much work has been done in developing computer-based reconstruction techniques for medical imaging [for example, Gershon (1991) and Stytz and Freider (1991)], only a limited number of specialised applications in the geosciences have been created (for example, Johnson and Moore 1993; Karonen 1985; Ohasi 1990; Tipper 1977). The main reason for reconstruction of serial sections is to enable computer-based three-dimensional visualisation that includes views of the internal features. This is important when dealing with geoscientific information, as frequently such images cannot be provided by any other means. There is without doubt potential for further work in geoscientific fields, such as palaeontology, sedimentology and orebody/reservoir modelling. These latter disciplines require modifications to existing reconstruction techniques, mainly due to the nature of the data. The work presented in this paper is primarily concerned with the reconstruction of invertebrate fossils, specifically Brachiopoda. However, it is intended to be applicable to other types of three-dimensional geoscientific data, and a set of contour data from a sand-shingle coastal spit is processed to demonstrate the versatility of the method.

The section drawings of palaeontological samples are vector in nature, lending themselves to surface-based reconstruction methods, such as the ones proposed in this paper. These types of techniques have seen a large amount of development in the last 20 years, but their usage in other application areas, especially in medical imaging, is now limited, with current trends towards volume-based approaches. Whilst data from medical imaging is generally better suited for reconstruction using this type of method, the sparse sampling of most geoscientific data renders the volume-based techniques less relevant.

The structure of the paper is as follows. In Sect. 2 we give an overview of serial section reconstruction,

The Visual Computer (1995) 11:343–359
© Springer-Verlag 1995

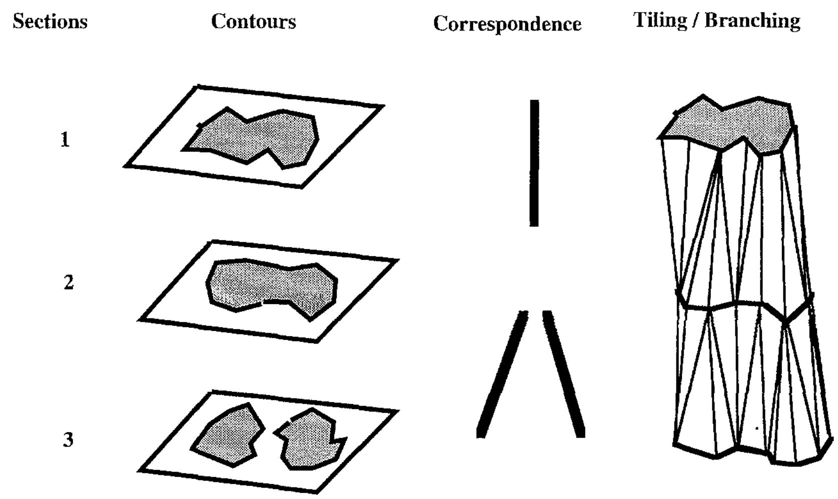


Fig. 1. Serial section reconstruction subproblems

and briefly review previous work. Section 3 discusses geoscientific reconstruction and outlines a new approach, highlighting a solution to the correspondence problem, one that uses a variety of spatial information. Other stages of the process are also discussed, including an automatic method to resolve complex branching situations. Results are presented in Sect. 4, and Sect. 5 is then used to discuss some of the problems encountered and highlight the directions for future research.

2 Review of serial section reconstruction techniques

Overview

The process of serial section reconstruction can be broken down into a number of subproblems, as outlined by Meyers et al. (1992) (Fig. 1).

- the *correspondence problem* is concerned with specifying the connectivity between contours in adjacent sections and therefore the underlying topology of the whole object. A solution to this subproblem can then be used to direct the other parts of the reconstruction process.
- the *tilting problem* consists of finding the best set of triangulated facets to define the surface between a pair of connected contours in adjacent sections.
- the *branching problem* occurs when the numbers of contours on a pair of adjacent sections are not equal. This causes problems for standard tiling algorithms. One solution is to decompose the problem into a one-to-one connection so that the tiling algorithm can be used. The surface then needs to be completed with further triangulations.
- the *surface fitting problem* is concerned with fitting a smooth surface to the resulting tiling triangulation and therefore facilitating visual inspection of the reconstructed object.

2.2 Correspondence solutions

Automated solutions to the correspondence problem have so far received limited attention. This may be due to the fact that the problem arises largely as a consequence of inadequate data sampling, and is therefore of less significance in areas such as medical imaging. High sampling rates tend to result in relatively small changes to size, shape and position between contours in adjacent sections. Therefore the topology of the object has been relatively easy to derive, either manually by the user or automatically by using simple metrics (Ekoule et al. 1991).

A number of graph-based methods have been used to identify the correspondence between neighbouring contours. These graphs have been generated in a number of ways, ranging from user definition (Giertsen et al. 1990) to the use of a minimum-spanning tree (MST) (Meyers et al. 1992) or a Reeb graph (Shinagawa and Kunii

1991). A rule-based system using generalised cones has also been proposed (Soroka 1981; Meyers et al. 1992), which creates cylinders from the set of contours to ascertain the correspondence.

These methods all provide automated solutions to the correspondence problem, but there are limitations that restrict their use. They tend to look for an object consisting of only a single component, one that progresses at right angles through the set of sections. Most algorithms make connectivity decisions based upon the information provided by just a single pair of sections. Such methods have difficulty in dealing with cyclic components such as toruses and spirals and usually require user assistance.

To achieve better results, the automatic approach that we propose increases the range of information that can be used. In particular we employ information based on topological relationships, shape difference and positional change between contours in adjacent sections. The technique is presented in detail in Sect. 3.

2.3 Tiling algorithms

There have been a large number of proposals for the resolution of the tiling problem. Initial graph solutions Fuchs et al. (1977) and Keppel (1975) used an optimal approach, based on graph theory, and gave the triangulation that best met a given criterion. For example, the algorithm of Fuchs et al. (1977) attempts to minimise the surface area of a polygon it is reconstructing. Refinements to this approach, to improve upon performance, have been suggested by Sloan and Painter (1988). These methods do give good results, but have a computational overhead in comparison with heuristic approaches. The 'shortest span' method (Christiansen and Sederberg 1978) is one such heuristic, which looks for the shortest of two possible edges between vertices in adjacent contours to create a triangular mesh. Other heuristic algorithms have been proposed by Cook et al. (1981) and Ganapathy and Dennehy (1982), the latter being based on the relative position of vertices on a contour rather than the actual distance. The heuristic algorithms require pairs of contours that are aligned and similar in shape to work successfully.

Alternative tiling methods include three-dimensional volume triangulation (Boissonnat 1988), which, rather than decomposing the problem to the contour level, processes the whole section simultaneously. This method provides a general tiling solution for both one-to-one and one-to-many branching situations, but is not very flexible, as it cannot easily be combined with a graph-based solution to the correspondence problem. Recent improvements to the original algorithm (Geiger 1993) have increased the range of complex samples that can be reconstructed.

In previous work (Herbert 1993), a change was made to the algorithm, so that the user could guide the reconstruction process with a correspondence graph. The implementation of this approach found that the graph could only limit the polyhedra constructed by the algorithm between neighbouring contours, and was not able to force contours to be connected.

2.4 Branching algorithms

Most reconstruction algorithms require a special solution to build triangulated surfaces when there is not a one-to-one connection between adjacent contours. A solution for simple cases was proposed by inserting intermediate vertices at a level half-way between the two sections and making it a single contour (Christiansen and Sederberg 1978). More complex situations demand that a polygon or 'canyon' patch be inserted between the multiple contours to complete the surface (Ekoule et al. 1991, Meyers et al. 1992). Other solutions include those suggested by Shantz (1981) and Zyda et al. (1987), which interpolate new contours to decompose the problem to a series of one-to-one correspondences. Many-to-many branching situations usually require additional data sampling to simplify the problem, with insertion of new sections.

The approach adopted for geoscientific reconstruction is similar to the method of Meyers et al. (1992), which is able to deal with complex branching situations. The improvements to the algorithm that have been proposed mean that, for a large number of cases, the process is completely automated. The algorithm is presented in more detail in Sect. 3.

2.5 Surface fitting

The facets produced by the tiling algorithm provide the basis for fitting a smooth surface to the reconstruction. This can be done by fitting a spline surface, based on the triangulation obtained from the tiling algorithm or using computer graphic shading techniques, such as Gouraud shading, to simulate a smoother surface. These techniques result in more acceptable images for visualisation. Where the sampling data has been sparse or nonuniform, the surface-fitting stage is vital to produce an image that is comprehensible to the user.

3 Modelling with geoscientific information

3.1 Introduction

When applying reconstruction techniques to geoscientific information, a number of factors, not necessarily encountered in other disciplines, have to be considered:

- For geoscientific information, the data sampling process tends to be more complex and time consuming, as well as being specifically designed for a single data set. As a result, the data collection costs can make up a large part of the total modelling costs.
- The sampling process is not necessarily exhaustive or complete, resulting in data that is sparse and for which the underlying topology is therefore difficult to understand.
- There can also be a lack of knowledge about the detailed form of the three-dimensional object prior to reconstruction, with only limited preconceptions about shape or form, so that little human interactive assistance can be provided to the modelling process at a low level.

Programs developed for use with geoscientific information must place a greater emphasis on an automated solution to the correspondence problem. This is not only because of the time involved in any user definition of contour correspondence, but also because of the complexity of the problem. The sampling rate is the major factor, as geoscientific objects can be difficult to comprehend from their two-dimensional sections. The other stages in the reconstruction process also have to be adapted, to cope with the more complex sections, but to a lesser extent. The tiling solution must be able to handle extremely concave contours, whilst the branching solution must handle a variety of one-to-many contour connections and possibly many-to-many connections.

3.2 GeoSect methods and implementation

3.2.1 Introduction

GeoSect is composed of a number of modules, each of which addresses one of the reconstruction subproblems. One module is the prototype of an automatic correspondence solution that employs a variety of spatial and geometric information, while the other modules cover the tiling, branching and surface-fitting algorithms. By building GeoSect in this way, it has been possible to develop the components incrementally and to experiment with various methods for each subproblem. The aim of GeoSect is to apply current techniques for serial section reconstruction to geoscientific data sets, with a number of enhancements to enable their reconstruction.
The modules have been developed in C for a UNIX workstation, and have used PHIGS/PHIGS+ graphics routines (Gaskins 1991; Hopgood et al. 1992) for visualisation purposes.

3.2.2 The correspondence solution

Introduction. This module is being used to develop an automatic solution to the correspondence problem. The current solution consists of three stages.

1. *Pre-processing.* A candidate graph is set up and edges are removed from it if they are regarded as invalid.
2. *Processing.* The graph is then reduced by a minimum spanning tree (MST) algorithm to provide a set of suitable edges for the correspondence solution.
3. *Post-processing.* Edges are removed from the MST to divide the graph into the individual components that make up the object.

The advantage of this approach over previous methods is the use of additional spatial information to derive a solution. The correspondence solution proposed uses three forms of spatial information.

1. *Relative Position.* The location of the pair of contours on adjacent sections, i and j, given by the contour centroids, (x_i, y_i) and (x_j, y_j).
2. *Shape.* This information is provided by the major and minor axes for each of the contours A_i and B_i, A_j and B_j.
3. *Topological Relations.* The relationship between a contour and the other contours on the same section is used to validate the existence of possible surfaces between contours on neighbouring sections. There are three types of relation, *Inside*, *Contains* and *Disjoint*, that can exist between a pair of contours on the same section. Inside and Contains are simply the converse of each other.

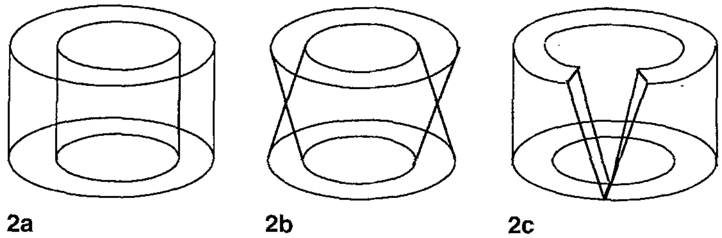
Position and shape information are represented in the edge metric, with the topological information being calculated for each section separately and used in the preprocessing stage.
The aim is to find a correspondence solution for a naturally occurring object and to do this by moving away from results derived by essentially geometric algorithms. The proposed algorithm builds upon the work of Meyers et al. (1992), but is concerned with making use of the topological information to derive the correspondence of objects, which may consist of more than one component.

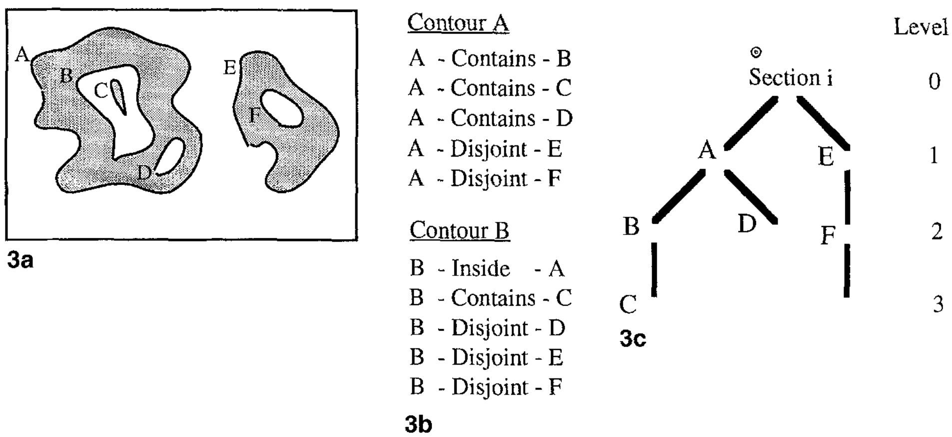
3.2.2.1 Preprocessing

The first step is to build a candidate undirected graph $G = (V, E)$, where V is the set of contours and E is the set of possible surfaces between contours in adjacent sections. For each edge e in set E, a weight is determined by the following metric:

$$e(i,j) = (x_i - x_j)^2 + (y_i - y_j)^2 + (A_i - A_j)^2 + (B_i - B_j)^2$$

where (x, y, A, B) are the centroid coordinates and the length of the major and minor axes (Tough 1988) for the pair of contours (i,j) in adjacent section (the lower the value of the weight, the more suitable the edge). The metric used combines a comparison of both the position and shape information for each pair of neighbouring contours (Meyers et al. 1992). This metric is suitable for objects with components that experience only gradual change in shape and that tend to progress at right angles to the sectioning. It therefore has its limitations when dealing with features like spires and concentric components. The low rates of sampling in many geoscientific objects will have a detrimental effect on the metric values for the edges, and the performance of the correspondence algorithm will be affected.
In the candidate correspondence graph there are no edges between contours in the same section. It is assumed that a cyclic component in an object is not intersected twice or more in a section without being sampled at least once in both of the neighbouring sections.
This stage of the correspondence process is different from previous graph-based solutions, as it uses topological relationships to eliminate edges that are invalid (Fig. 2). Haig et al. (1991) used this method to deduce connectivity solutions coupled with an overlap metric, rather than an MST.
With our proposed method, each contour is defined in relation to other contours in the section, using the three types of relation, *Inside*, *Contains* and *Disjoint* (Fig. 3b). For example, when contour B lies completely within another contour A, an Inside relationship is said to exist from B to A and a *Contains* relationship from A to B. A *Disjoint* relationship exists when there is no overlap between two contours in the same section. From the set of relationships for a section, a relationship tree can be built. (Fig. 3c). The first level of the tree includes all those contours that are not contained in another contour. This means that they are also disjoint from other contours on this level of the tree.
The relationship trees of adjacent sections are then used to check the edges in the candidate correspondence graph and to remove those that are invalid. A valid edge exists between pairs of adjacent contours when both of the contours are at the same level in their respective relationship trees (Fig. 2a). In most cases, an edge in the graph is invalid when the contours are at different levels in their relationship tree (Fig. 2b). A valid edge is possible between contours at different





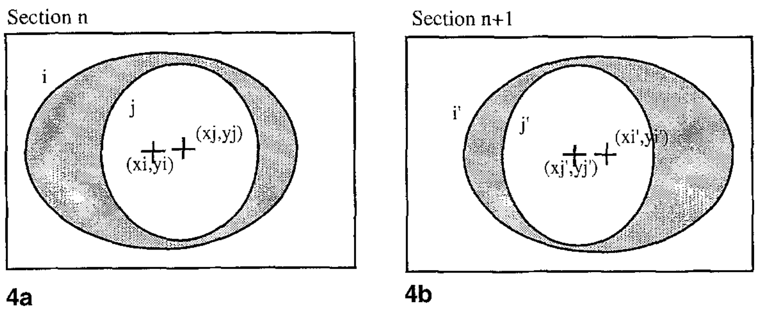


Fig. 2a–c. Topologically valid and invalid edges: **a** valid surfaces; **b** invalid surfaces; **c** valid surfaces

Fig. 3a–c. Topological information for a section: **a** section i; **b** topological relationships (for contours A and B only); **c** relationship tree

Fig. 4a, b. Component with an internal component

relationship levels when there is a branching situation (Fig. 2c), and such an edge is not marked as invalid by the preprocessor.

Removing invalid edges before calculating the MST is important. Although, in some cases a true MST of the original graph may now not be calculated from the remaining candidate graph edges, it means that the MST process cannot now include invalid edges in the graph, which previously may have meant the exclusion of valid edges that had higher weight values. It also allows more than one component to exist in the object, where they are contained by other components.

Figure 4 demonstrates the problems that can occur when topologically invalid edges are not eliminated prior to using the other forms of spatial information to calculate the graph. The distance between adjacent centroids (x_i, y_j) and $(x_{j'}, y_{j'})$, and centroids (x_j, y_j) and $(x_{i'}, y_{i'})$ is less than that of the correct contour matching i to i' and j to j'.

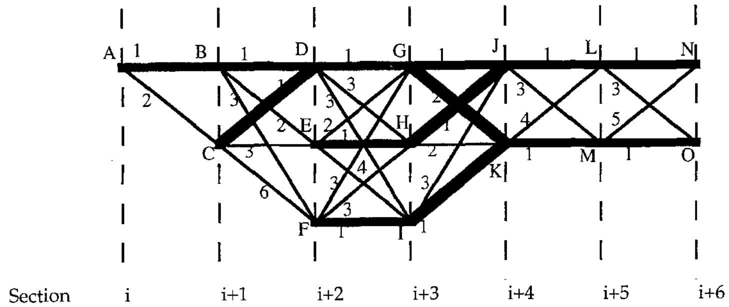


Fig. 5. MST (*thicker lines*) from a candidate correspondence graph

The difference in size and shape of the contours also affects the metric, but possibly not enough to prevent the lower weight value being given to a topologically incorrect edge, *i* to *j'* or *j* to *i'*.

3.2.2.2 Calculate minimum spanning tree (MST)

The MST (Fig. 5) of the candidate graph is computed using Kruskal's method (Corman et al. 1989). An MST of a graph gives a basic geometric structure, which has only one component and is acyclic in nature.

This method can be seen as global, as it examines edges from all over the model, not just from a local situation between a single pair of sections. It constructs the edges that have a greater certainty of inclusion in the component first, before trying to add the less likely ones. This approach does not concentrate on building one component of the object at a time, but processes the weaker edges with the higher weights last, so that the edge that completes the MST is the most likely to be removed by the postprocessing stage.

This implementation, due to the topological preprocessing, does not necessarily result in a single tree, but in a set of disjoint trees, each representing separate components. The complete list of sorted edges can be traversed by the algorithm with the MST remaining incomplete.

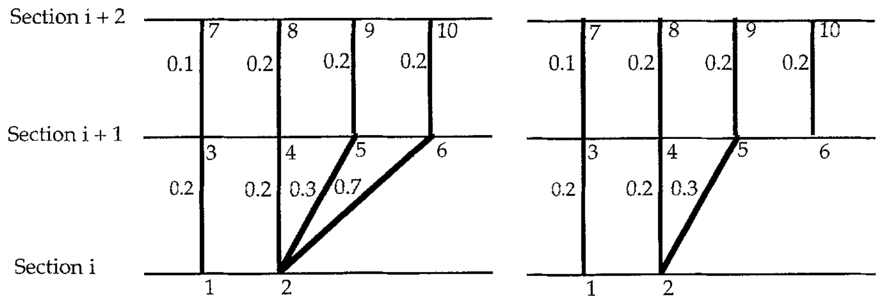
3.2.2.3 Postprocessing stage

The earlier stage has separated those components that are topologically inside other components, or ones that contain other components. For an object with more than one component, the MST calculation always makes erroneous connections between topologically disjoint components. The postprocessing stage presented here isolates each separate component by removing edges from the current graph to give a final correspondence graph.

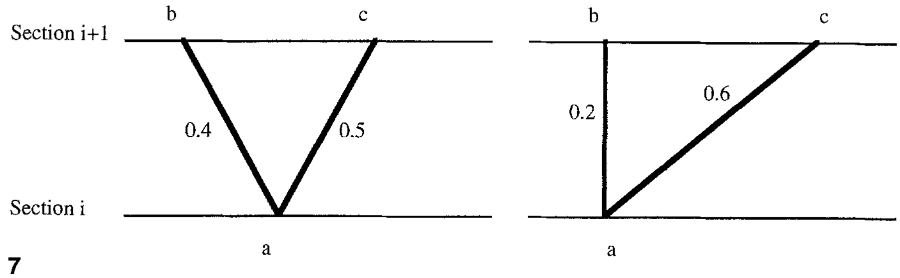
To carry this out (Fig. 6), each branching situation in the current graph is examined. It is at these points that incorrect edges have been added by the MST process. To eliminate these edges, the postprocessor checks the edges between each pair of sections. If the number of edges is greater than half the number of contours in the two adjacent sections, then an edge included in error may exist. The next step is to examine each of the edges between the two sections. The edges are sorted in order of ascending weight and are then examined, starting with the edge with the lowest weight. The next edge in the list is compared to the current edge and the difference in weight values calculated. If this difference is greater than a threshold value, then all edges with weights greater than the current edge are removed.

This method of reducing the edges to locate separate components from the single MST is only valid in a limited number of situations. Where the branch forks evenly, in a V shape, it is valid, as both edges have similar weights. If it is an uneven branch, with one edge continuing as the main component and the other edge branching at a large angle, there is a noticeable difference in weight and one will be removed by the postprocessing due to its higher value (Fig. 7).

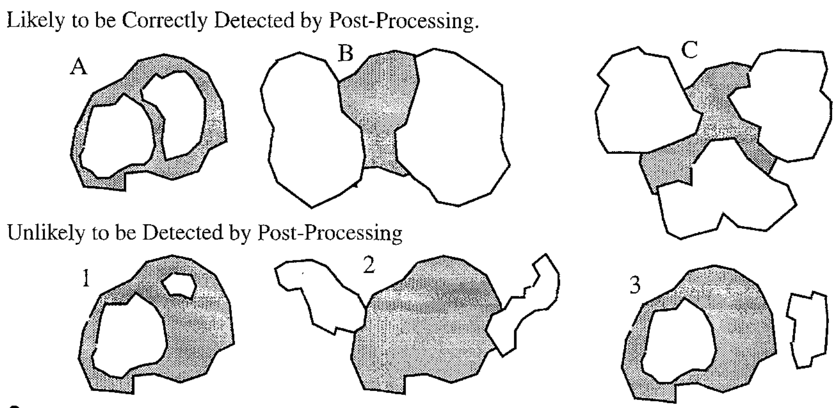
Using this edge weight metric, the postprocessing also favours branching, instead of edge removal, when the change in shape of the contours is small (Fig. 8). With geoscientific data sets, which may include ambiguous situations due to the undersampling, branching is unlikely to be always in



6a **6b**



7



8

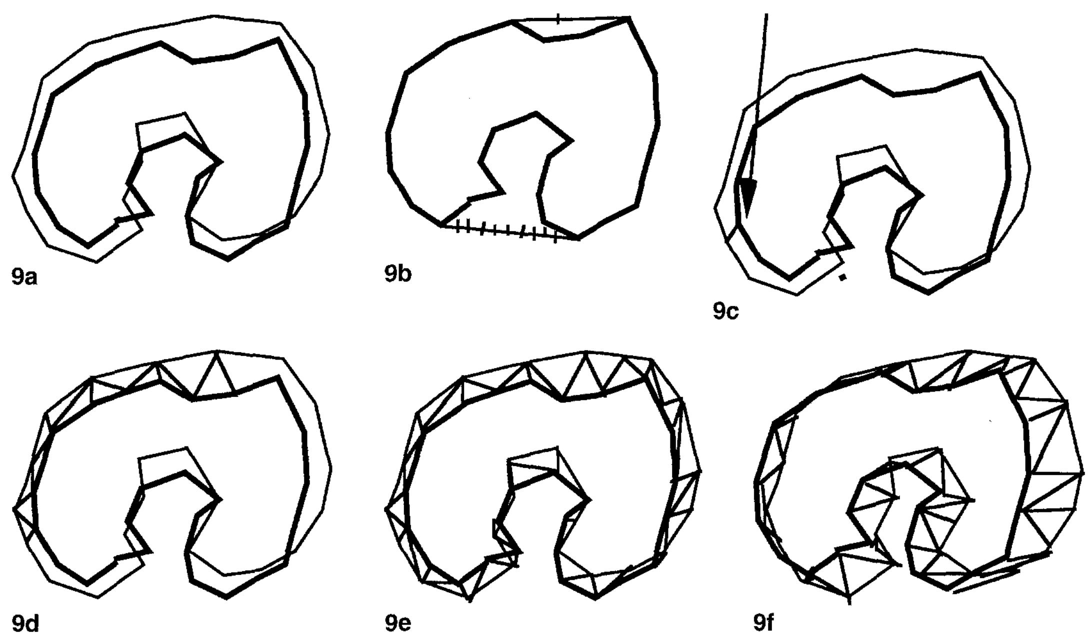
Fig. 6a, b. Removing edges during postprocessing: **a** before postprocessing – examine connections between section i and section i + 1; **b** remove edge 2–6, as increase in weight from edge 2–5 is too great

Fig. 7. V shape and uneven shape branching

Fig. 8. Detecting different branching situations

a format that the postprocessor can understand easily.

Discussion of correspondence techniques. By being able to deduce the correspondence of an object that may contain a number of components, the algorithm that has been presented here improves upon previous methods. This has been achieved by combining spatial information, especially topological, with geometric methods, such as the MST algorithm, to derive the final solution.
However, the MST algorithm is currently unable to find cycles or loops in the correspondence graph. The solution to this lies in either improving the pre- and postprocessing stages to make them more exhaustive and capable of adding edges to the MST or using a method other than the MST to find the basic graph and hence the major components. Leaving the detection of branching situations completely to a postprocessing stage is unsatisfactory. At least part of this process should be carried out during the main processing and should use the information that is available at that time. Whilst the algorithm has attempted to take into account topological information, further development is required. This is discussed in more detail in Sect. 5.



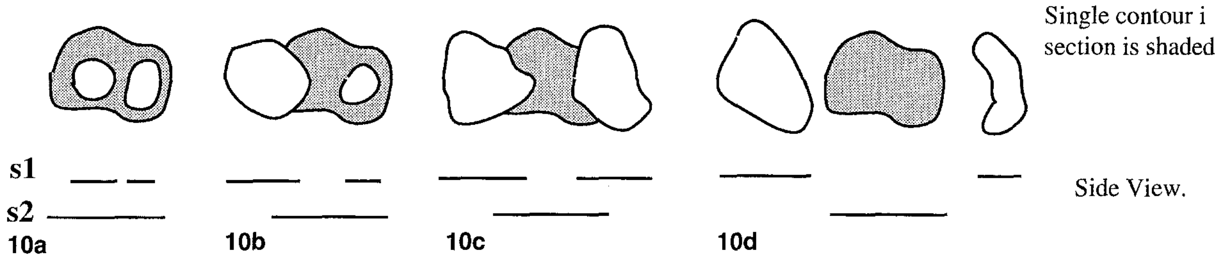


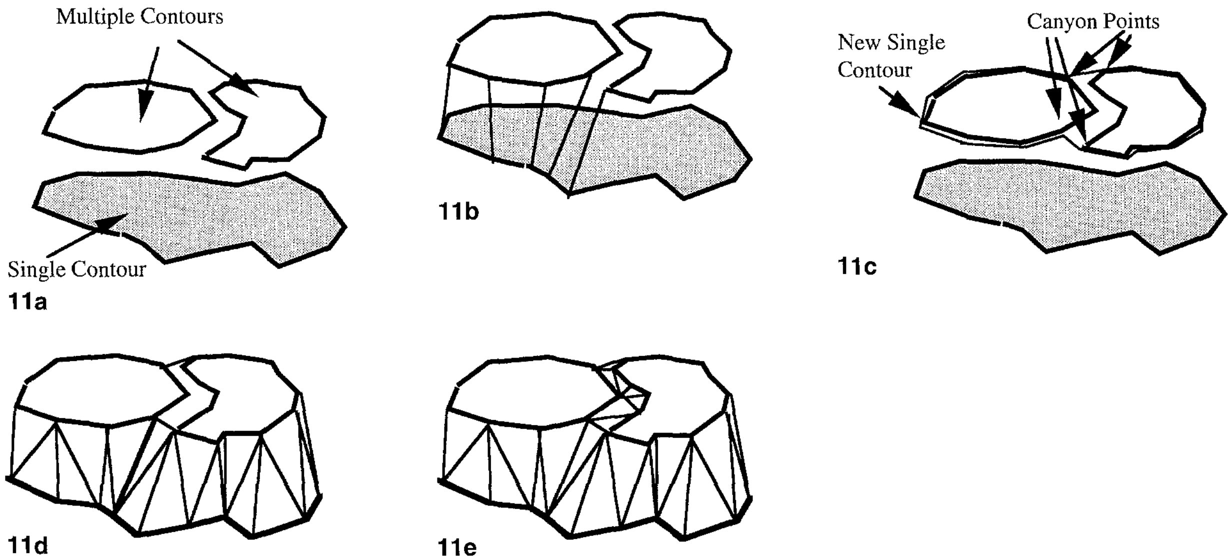
Fig. 9a–f. Tiling algorithm: **a** align the pair of contours; **b** construct the convex hull and project points onto the hull; **c** find the start pair for the triangulation; **d** using the heuristic, begin surface triangulation; **e** complete the triangulation; **f** move the contours to their original positions

Fig. 10a–d. Classification of branching situations: **a** enclosed; **b** mixed; **c** overlap; **d** disjoint. A single contour in the section is *shaded*

3.2.3 The tiling module

The tiling module can be used to build a three-dimensional triangulation in all situations identified in the correspondence graph in which a contour is connected to a single contour in the adjoining section. The algorithm is based on the 'shortest span' heuristic proposed by Christiansen and Sederberg (1978) and has been developed to be able to handle extremely concave contours. It is also fairly robust in that it can cope with contours that are somewhat dissimilar in shape, position and size.

The method (Fig. 9) starts by aligning the contours, using translations based on the contour centroids. For each of the pair of contours, a convex hull is calculated and all vertices of the contour not lying on the hull, are projected onto it (Ekoule et al. 1991). For the heuristic calculations, the convex hull vertices are used, but the original concave contour points are connected and are used as the basis for the surface. The algorithm starts by finding the closest pair of vertices and uses them as a start point from which to 'stitch' together the contours. The algorithm then moves round the contour pair, connecting the vertices together by using a 'shortest span' heuristic until the starting vertices are reached. This produces a series of triangular facets, to give a tiled surface. The process finishes by translating the contours



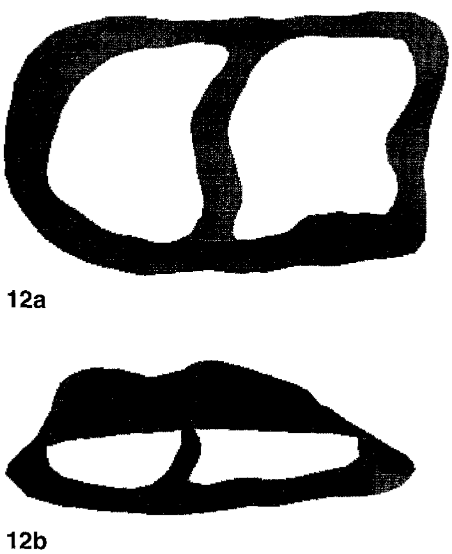


Fig. 11a–e. Branching algorithm: **a** identification of a branching situation from the correspondence graph; **b** for each vertex on the single contour, find the closest vertex on a multiple contour. The canyon points are found where the multiple contour changes; **c** mark the canyonpoints and use them to construct the new single contour around the multiple contours; **d** tile the surface between the two single contours; **e** complete the surface by triangulating the canyon area

Fig. 12a, b. Samples of various branching scenarios

back to their original positions. The algorithm is not a new solution to the problem and although a satisfactory surface is provided for the majority of cases, it has served to highlight problems that can be encountered (see Sect. 5.2).

3.2.4 The branching module

Branching scenarios are automatically detected from the final correspondence graph. The algorithm aims at reducing the problem to a one-to-one-contour situation, thus allowing the tiling algorithm to produce a triangulation. The simplest case occurs when a single contour connects to two contours in the subsequent section. There are a number of ways in which this situation can occur, some of which can be complex due to the change in position of the contours on the neighbouring sections. These are classified according to the position of the matching contours on their sections in Fig. 10. In the most straightforward case, the perimeter of the single contour, contains the multiple contours and this is classed as *Enclosed.* Where there is some overlap between the contours, this is either an *Overlap* or *Mixed*

situation. The *Disjoint* situation covers those matching contours that have a large change in contour position between the neighbouring sections and no overlap. Any solution to the branching problem must provide a satisfactory result for as many of these situations as possible.
The aim of the proposed solution is to make a new single contour that encompasses the multiple contours. This 'concave hull' can then be tiled to the adjacent single contour (Fig. 11). The algorithm commences by matching each contour vertex with the nearest vertex of the multiple contours in the adjacent section. When the multiple contour that contains the nearest vertex changes, the last vertex and the current vertex are marked as canyon start and end points. For example, when there are two multiple contours (a bifurcation), there are two canyon start and end points, one of each type on a contour. Using these points, new single contour is constructed around the multiple contours. A surface between this contour and the single contour in the adjacent section is tiled.
The canyon patch (the area in the new single contour that is not enclosed by one of the original multiple contours) is then tiled with the use of a constrained Delaunay triangulation (CDT) (De Floriani and Puppo 1988), to complete the surface. This algorithm is a dynamic, two-phase method for producing a CDT. These first stage constructs a conventional, unconstrained Delaunay triangulation from the canyon vertices, with the second stage iteratively inserting the constraining features, in this case, the canyon edges. After each insertion, a new CDT is produced.
The approach used with this algorithm is not new (Meyers et al. 1992), but it does contain improvements that mean, in the majority of cases, no user intervention is required to mark the start and end points of what can be complex canyon regions. The method proposed provides good results for some one-to-many situations, but does not, however, provide surfaces for many-to-many scenarios. Referring to the various types of one-to-two branching in Fig. 10, the algorithm can handle all Enclosed (Fig. 10a) situations along with most Mixed and Overlap situations (Figs. 10b and 10c). The Disjoint (Fig. 10d) case is most likely to fail during the detection of the canyon start and end points due to the large possible canyon area lying between the multiple contours. It then becomes necessary for the user to identify the points. Figure 12 shows results from the algorithm for a number of test samples, all constructed without user intervention.

3.2.5 The surface fitting module

No surface fitting is currently carried out with GeoSect, but the visualisation is enhanced by using Gouraud shading when displaying the surface triangulations. Conventional lambertian shading may however be more appropriate when examining specific parts of the reconstruction, since Gouraud shading may obscure the structure of the triangulation.

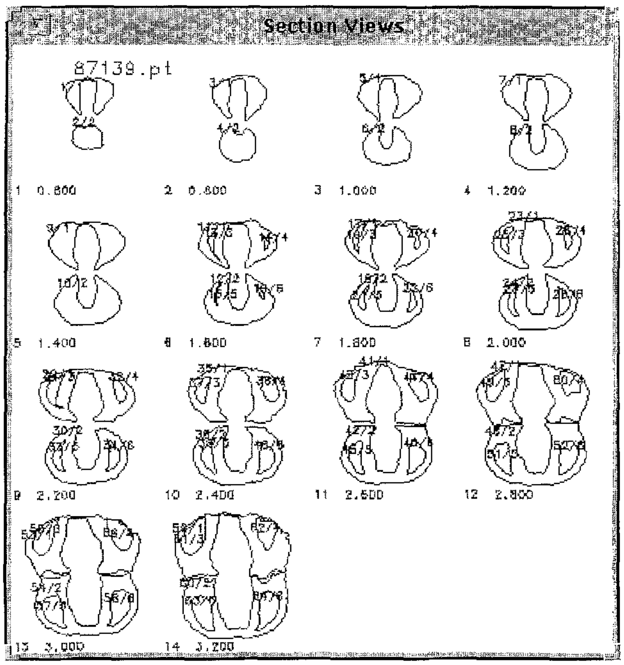
4 Reconstruction of geoscientific samples

4.1 Introduction

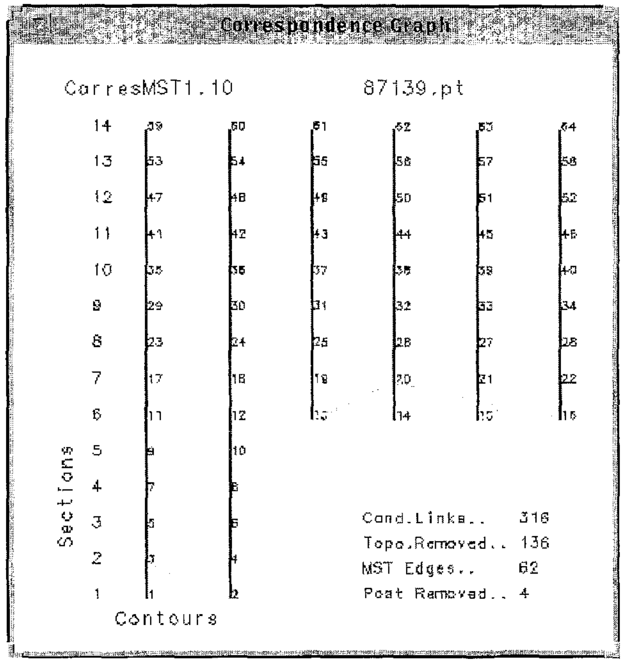
Four samples are presented, each having different characteristics. They were provided by the National Museum of Wales, Cardiff; the Natural History Museum, London and the Countryside Council for Wales (CCW), Bangor. The brachiopod samples (samples 1, 2 and 3), were obtained with serial grinding techniques. Cellulose peels were taken at various depths, from which the section drawings were made. These were then digitised to provide the data for the reconstruction. The sand-shingle spit (sample 4) was digitised from surveys made by the CCW.

4.2 Sample 1

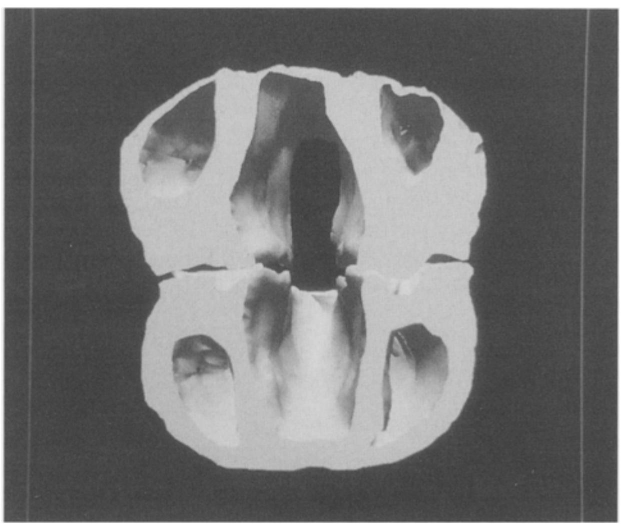
This is a fairly complex brachiopod sample (Fig. 13), from which the first 14 sections were digitised from drawings that contained no alignment marks. The user aligned the sections visually, using a computer-based tool. The correspondence graph (Fig. 14) found six separate components, the pedicle and brachial valves and the two holes that appear in each of them. The triangulation was produced with the use of the tiling algorithm only, as there were no branching situations (Fig. 15). The postprocessing stage of the correspondence module successfully removed the illegal branching edges left from the MST stage.



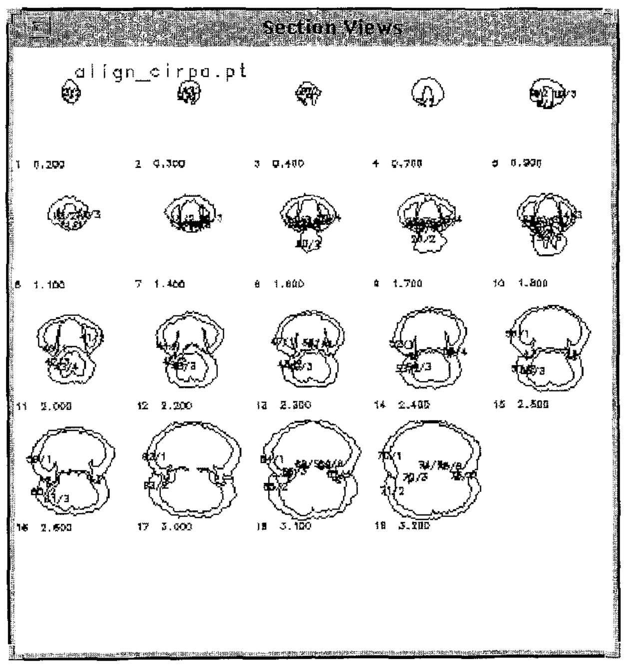
13



14



15



16

Fig. 13. Sections of sample 1

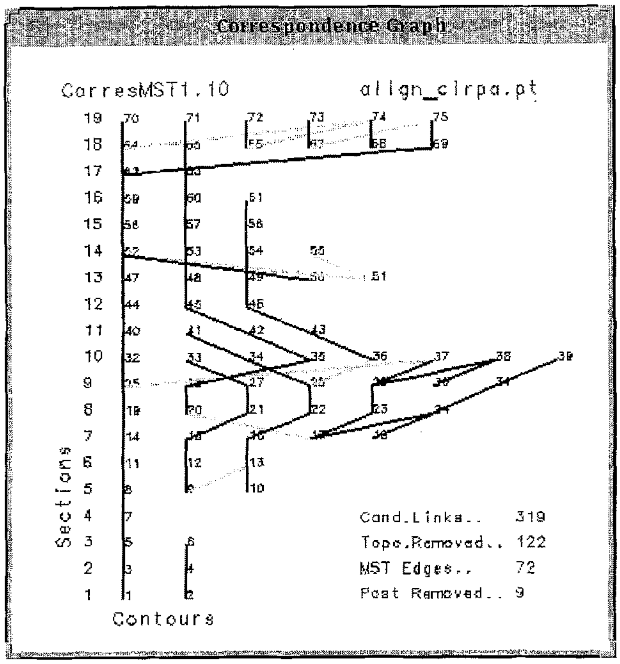
Fig. 14. Correspondence graph for sample 1

Fig. 15. Reconstruction of sample 1

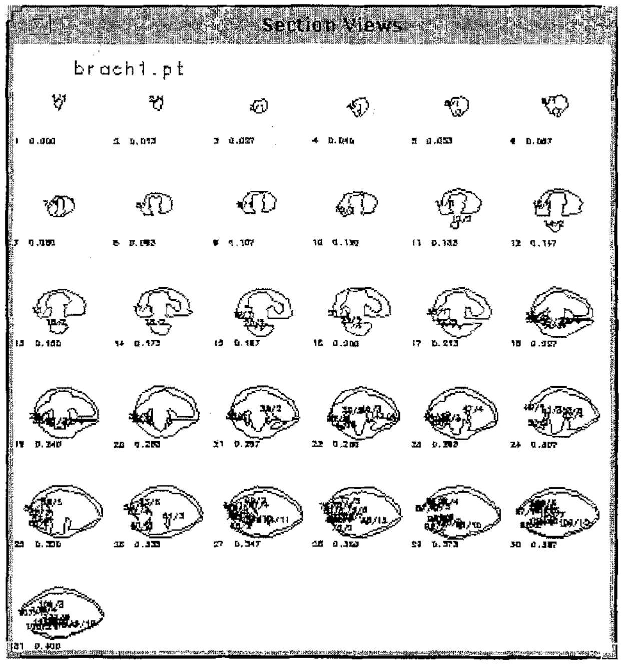
Fig. 16. Sections of sample 2

4.3 Sample 2

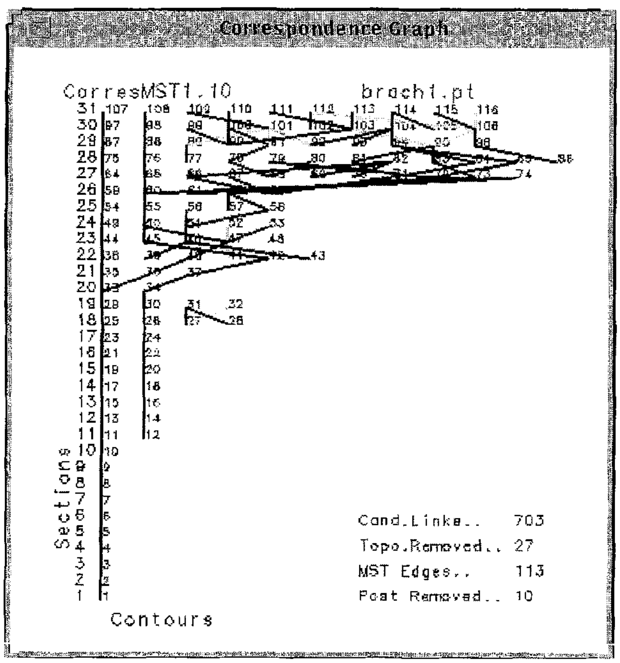
The 19 sections of this brachiopod sample were digitised from existing drawings (Fig. 16). The preprocessing eliminated 122 edges that were topologically invalid from the candidate graph (Fig. 17). The final correspondence graph has successfully located both main valves of the brachiopod, as well as the main internal features. Whilst correct branching situations were located



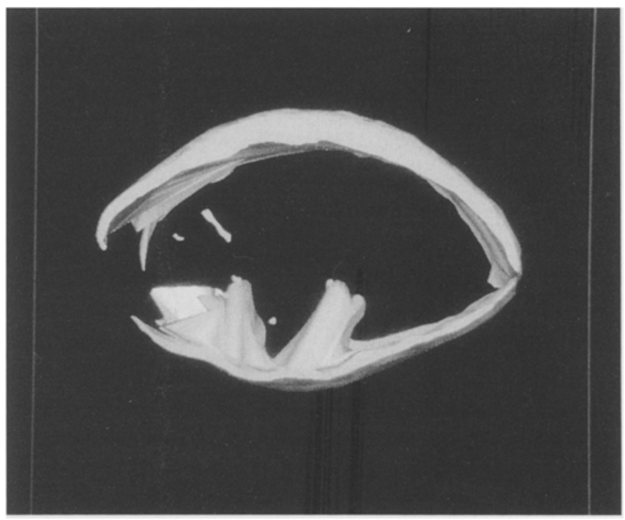
17



18



19



20

Fig. 17. Correspondence graph for sample 2

Fig. 18. Sections of sample 2

Fig. 19. Correspondence graph for sample 3

Fig. 20. Reconstruction of sample 3

for the sample, some incorrect branching edges were not removed by the postprocessing.

4.4 Sample 3

The sections of this sample were digitised from a set of drawings (Fig. 18), each of which contained alignment marks. These were in the form of four drill holes through the resin block that held the sample. Although the correspondence solution identified the two main valves, it was unable to cope with the large change in detail that occurred in some sections (numbers 25–31) and thus

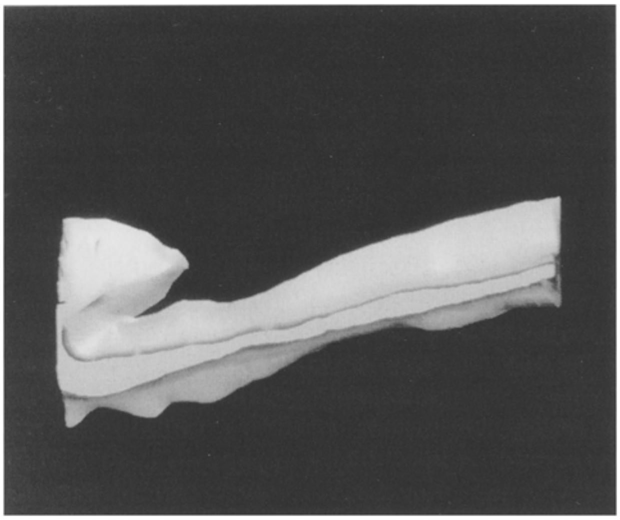


Fig. 21. Reconstruction of sample 4

was unable to pick out some branching scenarios clearly (Fig. 19). The lack of success is also due to the presence of cyclic features in these sections (the lophophore of the brachiopod). We produced the final model using both the tiling and branching modules (Fig. 20). However, as a result of the problems in deducing the underlying topology, some of the surfaces were omitted.

4.5 Sample 4

This sample, of a sand-shingle coastal spit in North Wales, has contours of a simpler nature than those encountered with the brachiopods. Here the underlying topology only consists of one-to-one connections, though one-to-many connections could exist if there were saddles and peaks. The correspondence is easy to resolve, and the tiling algorithm successfully produces a surface triangulation (Fig. 21). Surveys of this area have been made at yearly intervals since 1981. This sample was reconstructed as part of a longer term project, which is examining serial section reconstruction techniques as a way of carrying out temporal modelling for environmental monitoring.

4.6 Discussion

The samples have shown the potential for serial section reconstruction with geoscientific information. The correspondence of sample 1 was correctly identified, with each of the separate components being located. The major components of sample 2 and sample 3 were connected successfully by the Correspondence Algorithm, whilst there were no real problems with Sample 4. The tiling and branching algorithms are capable of handling most of the situations presented in these samples.
However, there are problems, especially with those objects not adequately sampled (such as in the later sections in sample 3). Although there are some difficulties in the reconstruction of these data sets with the tiling and branching algorithms, the main problem lies with the correct identification of the object components by an automatic correspondence algorithm. Thus, the problem of contour connectivity is the area most in need of improvement to build a successful geoscientific reconstruction system.

5 Further work

5.1 Deriving the correspondence graph

Algorithms that derive the correspondence graph from an MST, based only on edge weights from a simple metric, appear unlikely to cope with highly complex natural objects. These objects can contain features such as multiple branching, loops and have a variable number of components, all coupled with the problem of under sampling. Whilst the preprocessing stage of the correspondence solution proposed in this paper is able to introduce additional spatial information, it is only restricting geometric decision-making, rather than guiding it.
One way to improve the preprocessing stage of the correspondence process is to consider sets of sections, not just the two adjacent pairs. The algorithm will then be able to look at trends and changes in the connectivity, rather than just physical measurements.
Additional geometric information could be incorporated into MST calculation to improve results and reduce the likelihood of incorrect inclusion of edges. This could be carried out by having several weight values for an edge in the graph, each based on a different spatial characteristic. Examples might be measurements of

changes in parameters such as area or distance between centroids. Alternatively these values could reflect shape characteristics of the object, such as the number of times it has branched as well as its topological relationship with other components.
More significantly, the reconstruction process would be improved by using prior knowledge of the class of object being modelled, such as the number of components and the features that they may contain. This could be achieved by labelling components and defining as a set of parameters the relationships that exist between various components. This form of high-level user input would not be concerned with individual connections in the correspondence graph, but with specifying the underlying structure of the class of object being modelled with the use of either a text-based or diagrammatic notation. This high-level knowledge could then be used to guide the reconstruction process, resolving local conflicts and forming the basis of provisional hypotheses.

5.2 Tiling extremely concave contours

Some tiling metrics fail when contours change significantly between a pair of sections (Christiansen and Sederberg 1978). The solutions that have been proposed (Ekoule et al. 1991) and the algorithm proposed for GeoSect have problems with some triangulations between two highly concave contours. These tend to be skewed or twisted and do not model the original surface very well. This is because of the way the points are put onto the convex hull, based upon their distance from the last contour point that is also on the convex hull. A new solution is required.
One approach is to investigate the zonation of the contour into a series of curves, hence decomposing the triangulation on the basis of contour features. This is done in Keppel's (1975) optimal method and in other subsequent methods (Sinclair et al. 1989). This would enable parts of matching contours to be tiled together separately and, hopefully, more accurately.
In severe cases, in which the change in shape is very large, it is impossible to construct a surface between two neighbouring contours (Gitlin et al. 1994) and none of the tiling approaches mentioned will be successful. It is necessary for any solution to recognise such cases and make changes to the data, possibly by inserting extra vertices on the neighbouring contours. As has already been mentioned, with geoscientific data sets it is highly unlikely that additional intermediate contours can be sampled to help with surface reconstruction.

5.3 Classification of branching scenarios

Branching scenarios have usually been classified by the number of contours in each section: one-to-one, one-to-many and many-to-many situations. To deal with complex objects successfully, a more sophisticated classification may be necessary, which would allow the application of different branching algorithms depending on the situation (see Fig. 10). The classification could incorporate the amount of overlap or similarity of the contour sets. The type of scenario would then dictate which branching algorithm, from an available suite of solutions, would be used.

5.4 Interpretation of the section drawings

The majority of computer-based palaeontological and geoscientific reconstructions are made from existing drawings. Converting them to digital form presents some problems. Each of the drawings, if recorded in detail, requires expert knowledge to convert the often discontinuous lines into the closed contour loops, which will provide true three-dimensional surfaces on reconstruction. The way in which the drawings are interpreted can greatly affect the final three-dimensional reconstruction.

5.5 Alignment of sections

The alignment of sections presents problems, as in most cases there are no indicators or marks on each of the section drawings. When preparing new samples, there are methods by which registration marks can be put into the resin block being ground (Baker 1978), and this is vital for accurate computer reconstructions. The user can align non-marked sections visually, using a computer-based tool. Here the user matches prominent features in each section by using various transformations, such as rotation and translation. This has been carried out on some palaeontological samples (as with Sample 1) with some success. The method depends on having some expert knowledge about the sample being reconstructed.

5.6 Temporal modelling

Some applications generate a set of samples that have been recorded over a period of time. This may occur with the environmental monitoring of coastal features, for example, where surveying has taken place at regular intervals. Serial section reconstruction can be used with the contours of each section to produce a three-dimensional image, and the techniques can be extended to carry out temporal interpolation between each sample (Herbert and Kidner 1993). This would allow the creation of intermediate samples and the possible animation of changes to the model. There are possibilities of expanding on a number of the tiling and branching algorithms to produce 'surfaces' through time. Here the correspondence problem will continue to play a part in connecting time-adjacent three-dimensional surfaces.

6 Conclusion

There is little doubt that computer-based serial section reconstruction has great potential when dealing with geoscientific information. There are some occasions where existing algorithms can be directly transferred to geoscientific applications. However, in a majority of cases, refinements and enhancements need to be made to these techniques, due to the complex nature and inadequate sampling of geoscientific data.

The serial section reconstruction technique described in this paper has been applied to various geoscientific data sets with some success, and the MST appears to provide a promising basis for flexible automatic solutions to the correspondence problem. Current limitations with the algorithm stem from its restriction to purely local information when considering connectivity between two adjacent sections.

To improve the performance of correspondence procedures, future work will have to combine global information on the form of the object with local information at the section level. It is anticipated that future work will maintain a stereotypical model of the object, so that when the correspondence between adjacent sections is being considered, a wide range of prior information relevant to the specimen is available to help guide the correspondence decisions being made at the local level. Present research is following this line of approach.

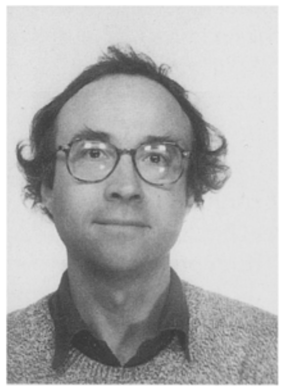
Acknowledgements. The authors thank Michael Bassett, Keeper of Geology at the National Museum of Wales, Cardiff for providing sample sections and being the main collaborator for this work. Thanks are due Howard Brunton at the National History Museum, London and Mike Hesketh at the Countryside Council for Wales, Bangor, both of whom provided samples, and also to the anonymous referees of an earlier draft of the paper, whose comments have resulted in a much improved revised version. M.J. Herbert has been supported by a Science and Engineering Research Council studentship.

References

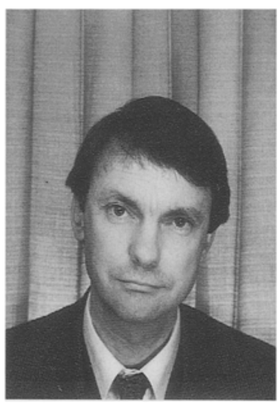
1. Ager DV (1965) Serial grinding techniques, In: Kummel B, Raup D (eds) Handbook of palaeontological techniques. W.H. Freeman, San Francisco, pp 212–224
2. Baker PG (1978) A technique for accurate reconstruction of internal structures of micromorphic fossils. Palaeontology 2:463–467
3. Boissonnat J-D (1988) Shape reconstruction from planar cross sections. Comput Vis Graph Image Processing 44:1–29
4. Christiansen HN, Sederberg TW (1978) Conversion of complex contour line definitions into polygonal element mosiacs. ACM Comput Graph 12:187–192
5. Cook LT, Cook PN, Lee KR, Batnitzky S, Wong BYS, Fritz SL, Ophir J, Dwyer SJ, Bigongiari LR, Templeton AW (1980) An algorithm for volume estimation based on polyhedral approximation. IEEE Transactions on Biomedical Engineering, Vol 27
6. Corman TH, Leisorson CE, Rivest RL (1990) Introduction to algorithms. MIT Press Cambridge, Massachusetts
7. Dabek ZK (1988) Development of advanced interactive computer modelling techniques for multicomponent three-dimensional interpretation of geophysical data. British Geological Survey Technical Report WK/88/2, Keyworth, UK
8. De Floriani L, Puppo E (1988) Constrained Delaunay triangulation for multiresolution surface description. IEEE Computer Society Reprint (Washington DC: Computer Society Press). Reprinted from the Proceedings of the 9th IEEE Conference on Pattern Recognition, Rome
9. Ekoule AB, Peyrin FC, Odet L (1991) A triangulation algorithm from arbritary shaped multiple planar contours. ACM Trans Graph 10:182–199
10. Fuchs H, Kedem ZM, Uselton SP (1977) Optimal surface reconstruction from planar contours. Commun ACM 20:693–702
11. Ganapathy S, Dennehy TG (1982) A new general triangulation method for planar contours. ACM Comput Graph 16:69–75
12. Gaskins T (1991) PHIGS Programming Manual, O'Reilly, Sebastapol, CA

13. Gershon ND (1991) Visualising 3D PET Images. IEEE Comput Graph Appl 11:11–13
14. Giertsen C, Halvorsen A, Flood PR (1990) Graph-directed modelling from serial sections. Vis Comput 6:284–290
15. Geiger B (1993) Three-dimensional modelling of human organs and its application to diagnosis and surgical planning. Technical Report 2105, INRIA BP93, 06902, Sophia Antipolis, France
16. Gitlin C, O'Rourke J, Subramanian V (1994) On reconstructing polyhedra from parallel slices. Technical report, Dept. of Computer Science, Smith College, Northampton, MA
17. Haig TD, Attikiouzel Y, Alder M (1991) border marriage: matching of contours of serial sections. IEE Proceedings I 138:371–376
18. Herbert MJ (1993) Computer Studies Technical Report CS-93-2 'Computer-based reconstruction of invertebrate fossils from serial sections'. Department of Computer Studies, University of Glamorgan, Pontypridd, Mid Glamorgan CF37 1DL, Wales
19. Herbert MJ, Kidner DB (1993) Spatial and temporal visualisation of three-dimensional surfaces for environmental management. AutoCarto 11: Proceedings of the 11th International Symposium on Computer-Assisted Cartography, Minneapolis, pp 318–327
20. Hopgood FRA, Duce DA, Johnson DJ (1992) A primer for PHIGS: C Programmers Edition. John Wiley, New York
21. Johnson SE, Moore RR (1983) Surface reconstruction from parallel serial sections using the program *Mathematica*: example and source code. Comput Geosci 19:1023–1032
22. Karonen O (1985) Geometric mine modelling: Modeling of three-dimensional objects based on Incomplete information. PhD Thesis, Acta Polytechnica Scandinavica 45:1–77
23. Keppel E (1975) Approximations of complex surfaces by triangulation of contour lines. IBM J Res Devel 19:2–11
24. Marko M, Leith A, Parsons D (1988) Three-dimensional reconstruction of cells from serial sections and Whole cell mounts using multi-level contouring of stereo micrographs. J Electron Microsc Techn 9:395–411
25. Meyers D, Skinner S, Sloan KR (1992) Surfaces from contours. ACM Trans Graph 11:228–258
26. Ohasi Y (1990) Three-dimensional reconstruction of pore geometry from serial sections – image algebraic approach. Lecture Notes in Earth Science, Comput Graph Geol 41:63–76
27. Rhodes M (1990) Computer graphics in medicine. IEEE Comput Graph Appl 10:20–23
28. Shantz M (1981) Surface definition for branching contour-defined objects. Comput Graph 15:242–270
29. Shinagawa Y, Kunii TL (1991) Constructing a Reeb graph automatically from cross sections. IEEE Comput Graph Appl 11:44–51
30. Sinclair B, Hannam AG, Lowe AA, Wood WW (1989) Complex contour organisation for surface reconstruction. Comput Graph 13:311–319
31. Sloan KR, Painter J (1988) Pessimal guesses may be optimal: a counterintuitive search result. IEEE Trans Patt Anal Mach Intell 10:949–955
32. Soroka BI (1981) Generalised cones from serial sections. Comput Graph Image Processing 15:154–166
33. Stytz MR, Frieder O (1991) Computer systems for three-dimensional diagnostic imaging: an examination of the state of the art. Crit Rev Biomed Eng 19:1–45
34. Tipper JC (1977) Three-dimensional analysis of geological forms. J Geol 85:591–611
35. Tough JG (1988) The computation of the area, Centroid, and Principal axes of a Polygon. Comput Geosci 14:715–717
36. Zyda MJ, Jones AR, Hogan PG (1987) Surface construction from planar contours. Comput Graph 11:393–408

MALCOLM HERBERT is currently a Lecturer at the University of Wales, Lampeter, based in the Department of Informatics. Prior to this, he was a research student in the Department of Computer Studies at the University of Glamorgan, Pontypridd, South Wales. His research areas include serial section reconstruction, three-dimensional modelling and application of computer techniques to the Earth Sciences. One of his main interests is examining how existing modelling techniques can be adapted for use with palaeontological data. He received his BSc in Computer Studies from the Polytechnic of Wales in 1991.



DOUGLAS TUDHOPE has been a Senior Lecturer at the Department of Computer Studies, University of Glamorgan for five years, teaching computer graphics, hypermedia and human-computer interaction. Prior to this he was a programmer/analyst on the Faint Object Spectograph Project . of the Hubble Space Telescope at the University of California, San Diego. His primary research interests are hypermedia and knowledge-based approaches to computer graphics and computer modelling. He graduated in Computer Science at the University of Edinburgh in 1976 and received a PhD in 1981 from the University of Wales.



CHRIS JONES is currently Professor of Geographical Information Systems (GIS) in the University of Glamorgan. He has had previous lecturing posts, in GIS, in the Department of Geography at the University of Cambridge, and in computer graphics in the Department of Computer Studies at the University of Glamorgan. He has also worked on geological computing applications at BP Exploration (1981–1983) and at the British Geological Survey (1976–1981). He has a degree in geology from Bristol University, and a PhD from the University of Newcastle upon Tyne, for research on periodicities in fossil growth rhythms and the history of the Earth's rotation. Current research interests include automated cartographic design; deductive, multiscale spatial databases; hypermedia and 3D geoscientific data modelling.