CONTINUOUS SKELETONS FROM DISCRETE OBJECTS

Levent Gursoz, Fritz B. Prinz, Atul Sudhalkar

EDRC 24-87-92

Continuous Skeletons from Discrete Objects

Levent Gursoz* Fritz Prinz Atul Sudhalkar†

April 1992

Technical Report CMU EDRC 24-87-92
Engineering Design Research Center[1],
Carnegie Mellon University,
Pittsburgh, PA 15213.

Abstract

The Medial Axis Transform (MAT) was defined by Blum in the sixties as an alternate description of the shape of an object. Since then, its potential applicability in a wide range of engineering domains has been acknowledged. However, this potential has never quite been realized, except recently in two dimensions. One reason is the difficulty in defining algorithms for finding the MAT, especially in three dimensions. Another reason is the lack of incentive for modeling designs directly in MAT's.

Given this impasse, some lateral thinking appears to be in order. Perhaps the MAT *per se* is not the only skeleton which can be used. Are there other, more easily derived skeletons, which share those properties of the MAT which are of interest in engineering design?

In this work, we identify a set of properties of the MAT which, we argue, are of primary interest. Briefly, these properties are dimensional reduction (in the sense of having no interior), topological equivalence, and invertibility. For the restricted class of *discrete* objects, we define an algorithm for identifying a point set, generically called a skeleton, which shares these properties with the MAT.

The algorithm will be defined for two dimensions, and a proof will be outlined. The true focus of our work, however, lies in the extension of this to three dimensions. We present ideas on how the 2D algorithm can be extended to 3D objects, and also present a line of argument which should extend the proofs of the 2D algorithm to three dimensions.

---

*In alphabetical order

†e-mail: atul@edrc.cmu.edu, phone (412) 268-5214

[1]An NSF Engineering Research Center at Carnegie Mellon University

1 Introduction

For the purposes of modeling, an object will be regarded as the closure of a bounded open set in space. An object can then be represented as the union of the closure of some finite number of open sets, or (since it is bounded) by representing its boundary. Most geometric modelers conform to this. However, there is an equivalent way of representing the closure of bounded open sets in space, called the Medial Axis Transform (MAT), defined first in [2]. By equivalent we mean that the closure of any bounded open set in space can be represented using the MAT.

For a two-dimensional (2D) object, the medial axes are the locus of the centers of maximal discs which can be fit into the object, where a disc is considered maximal if it is not a subset of any other such disc. Figure 1 shows how to determine medial axes by fitting maximal discs into a polygon. A radius function is defined at each point on the medial axes, giving the minimal distance from that point to the original object boundary. Figure 2 shows a polygon and its medial axes.

The MAT definition is similarly extended to three-dimensional objects. The MAT then consists of a set of medial surfaces (instead of axes) and a radius function. The medial surfaces are identified from the boundary of the solid using a set of maximal balls (instead of discs). Figure 3 shows a rectangular plate, and it's MAT, which consists of four triangles, eight trapezia, and a rectangle.

The MAT of an object facilitates analysis in several domains. In particular, such a representation is very useful in automating manufacturability analysis for net-shape manufacturing of thin-walled parts, as in injection molding and die-casting [7]. Briefly, many numerical analysis packages for such processes take advantage of the plate-like nature of such parts, and simplify the analysis by assuming laminar flow. With this assumption, it is now possible to approximate the part by a collection of so-called $2\frac{1}{2}$D patches, with associated thickness information (see figure 4). It is possible to use portions of the MAT as the elements of such a $2\frac{1}{2}$D representation. The MAT can also play a very important role in extracting shape features useful in a knowledge-based approach to moldability analysis (see [7], [3].)

Despite this, it is neither convenient nor easy to use the MAT as the primary representation of a design. Firstly, any form-function relationships which designers use to create the design are likely to be in terms of the spatial properties of the point-set to be represented. In general, it is not easy to visualize this from the MAT. Secondly, in manufacturing processes where the surface of the design must be created by shaping or removing material, a representation which directly yields this information must surely be preferred.

2 Voronoi Diagrams and the MAT

Given a finite set of points in space, the *Voronoi diagram* is defined to be the set of points which are equidistant from two or more of the given points (see, e.g., [17]). As Wolter points out in his recent paper [22], both the MAT and Voronoi diagrams are special cases of a more general concept, which he terms the *cut locus* of a set. More precisely, the cut locus of a closed set A is defined to be the set of points which have at least two shortest paths to the set A. If A is the boundary of some open set (solid), then the (closure of) parts of the cut locus properly contained in the solid are the medial axes. If the set A is finite, then the cut locus is precisely the Voronoi diagram.

2.1 Finding the MAT

The relationship between the MAT and Voronoi diagrams in the plane has been exploited in earlier work by Lee ([13], [12]), who describes efficient algorithms to find the medial axes of planar objects as subsets of their generalized Voronoi diagrams. However, the extension of this work to three dimensions has proved formidable.

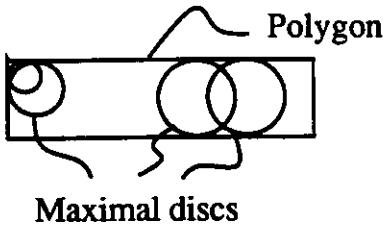


Figure 1: Definition of 2D MAT

Many people have investigated the "direct" problem, i.e., that of finding the MAT given a description of the boundary. Although success has been reported for two-dimensional objects [5] the three-dimensional case has proved much harder. Dutta and Hoffman have reported some preliminary work in this area [4] [9], but their focus has been on getting a solution to the problem of finding point sets which are equidistant to a given pair of boundary elements. While this would be an essential component of any MAT-finding algorithm, a major difficulty they do *not* address is the identification of so-called critical points (edges) in the MAT. A *critical point* can be defined as a point at which more than two branches of the medial surfaces (axes) meet. The identification of such critical points (which are equidistant to more than two components of the boundary) is an essential step in finding MAT's.

2.2 Approximate MAT's via Voronoi Diagrams

The concept of a Voronoi diagram can be used to find an *approximation* to the MAT of a planar object, as reported by Tam et al [20] and also by Yu et al [23]. Since the medial axes of a planar figure are, given a sufficiently abstract view of the problem, a generalization of the Voronoi diagram, it is natural to ask if the Voronoi diagram of a *finite* subset of the boundary of a planar object can be useful in finding the MAT. As Tam et al report, this is indeed so. The Voronoi diagram of a sufficiently well-chosen "representative" set of points from the boundary of a planar object is found to be a good approximation for several engineering purposes, including automatic mesh generation [19].

Suppose a finite point-set A is chosen from the boundaries of some object O. Tam et al proceed by first finding the *Delaunay triangulation* of this set. For a good treatment of this subject, see [17]. In two dimensions, this consists of polygons having the property that the circumscribing circle of any of the polygons does not enclose any points of the set A. It is usually assumed that the polygons are all triangles, but this need not be so in general. In any case, the Voronoi diagram of these points is the straight-line dual of the Delaunay triangulation. Since the circumscribing circles can be considered to be approximations to maximal disks, the Voronoi diagram could be considered an approximation to the MAT.

Since the Delaunay tessellation of a set of points is defined without reference to any solid, it can be seen that this can be defined only for the convex hull of the set A. Since engineering objects are usually far from convex, it is necessary to ensure that the edges marking the original boundary of the object be a subset of the Delaunay edges. If the set of points A is sufficiently well-chosen, then this can be assured. Unfortunately, "well-chosen" is difficult to define *a priori*. Tam et al get around the problem by introducing additional points into A from the missing elements of the boundary, if any. The Delaunay tessellation can be recomputed in the neighborhood of such additional points.

In investigating this approach, however, it was found that the problem of extending the ideas to 3D is quite formidable. The necessity of handling higher order polyhedra, as also ensuring that the set A is well-chosen, are both much harder in 3D than in 2D.

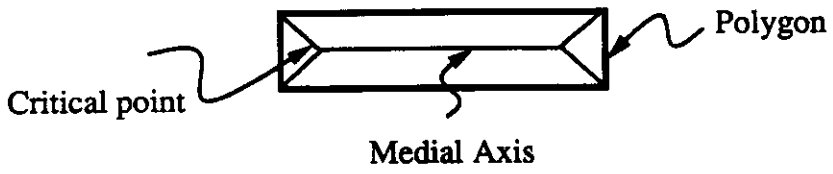


Figure 2: Example of MAT for a simple polygon

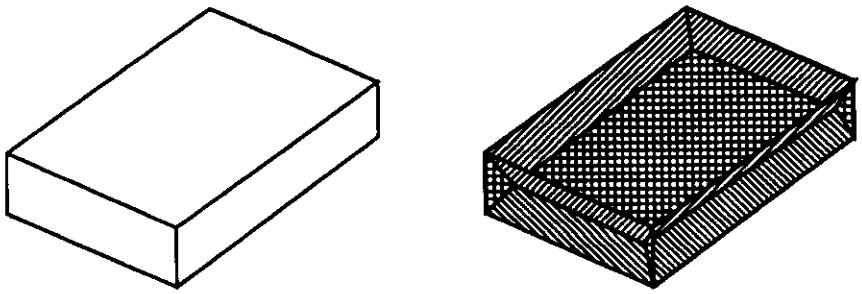


Figure 3: A rectangular plate and the 3D MAT

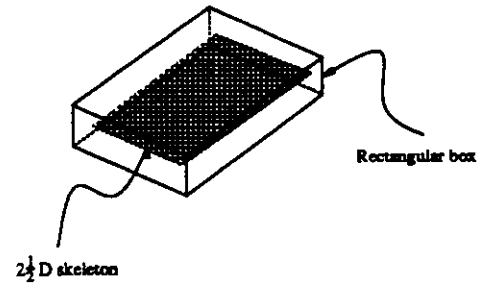


Figure 4: A rectangular plate and the $2\frac{1}{2}Dskeleton$

3 Digital Thinning

Quite independent of research in engineering design, there has evolved an area of research in computer vision called *digital thinning*. The literature in this area is vast, covering problems posed in both two- and three-dimensions. Instead of describing individually the different approaches taken by various authors, we refer instead to the work of Kong and Rosenfeld [11], in which a unifying view of the field is presented. The problem, as stated therein, is as follows:

Objects (images) are binary arrays (i.e., have value 0 or 1) in two or three dimensions. Each element is identified with an integer tuple (i.e., an element of Z^n). They can thus be referred to as lattice points, or points. Each point is considered to be adjacent to several others, depending on the type of adjacency chosen. If two points differ by unity in *at most* one of their coordinates, they are said to be 4-adjacent in 2D and 6-adjacent in 3D. If two points differ by at most unity in *any* (or all) of their coordinates, they are said to be 8-adjacent in 2D, and 26-adjacent in 3D. Obviously, the latter type of adjacency includes the former. Note that these two types of adjacencies are related to the discrete analogues of the L_1 and the L_∞ norms, respectively. The *immediate neighborhood* of a point p, denoted by $N(p)$, is defined to be the set of all points adjacent to it. Note that $N(p)$ does not include p.

In order to get a consistent topology on the image, it is necessary to require the object and its complement to have different types of connectivity. Thus, in 3D, the compatible pairs of connectivity type for the object and for the complement are (26, 6), (6, 26). A digital picture is then defined as $\mathbf{P} = (Z^3, m, n, B)$, where $(m,n) = (6,26)$ or $(m,n) = (26,6)$, and denotes the connectivity type of the object and the complement, respectively. B denotes the set of points which define the object. Using this definition, it is not difficult to define standard notions of topology such as connectedness, components, paths, etc.

The topology of the object is concerned with the number of components of the object, whether and where components of the object (complement) are enclosed (surrounded) by the complement (object), and, for three-dimensional images, the number and location of holes (also called handles, or tunnels). A thinning process should clearly preserve all these. An *Euler characteristic*, denoted by χ, is a means of capturing the topological properties of the object.

In order to get skeletal or thinned versions of the object, a thinning process is then defined. The process deletes points which are called *simple*, defined below:
A point is simple *only* if

1. p is adjacent to just one component of $N(p) \cap B$.
2. p is adjacent to just one component of $N(p) \setminus B$.
3. $\chi(Z^3, m, n, B \cap N(p)) = \chi(Z^3, m, n, B \cap N(p)$.

The above can be said to be the topological properties of simple points; these specify *necessary* conditions for "simplicity". Topologically, however, a billiard ball is equivalent to a baseball bat. Hence it is necessary to impose other "geometric" constraints. Such constraints attempt to identify so-called edge points, i.e., points which lie at the extremes of skeletal components of the thin image, and require such points to be non-simple. Tsao and Fu [21], and Morgenthaler [15] present different methods to achieve these same ends. Hafford and Preston (see [6]) take an approach which allows them to unify all these criteria (or their equivalent) into a uniform framework; as it turns out, however, their approach also leads to the same goal: a discrete skeleton.

Thus, the objective of the thinning process has traditionally been set at reducing the discrete object to a set of points (pixels/voxels) which together define an image which has thickness as close to unity as possible, while preserving "lengths" of components of the skeleton, so that they do not get shortened. Clearly, this is related to the MAT of an object. Unfortunately, from the design perspective at least, there is one fundamental

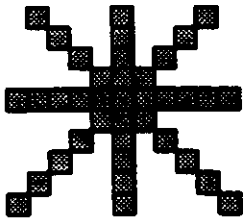


Figure 5: An example of a 2D thinned object which remains too dense to allow a continuous skeleton to be fit to it.

shortcoming in such algorithms. These algorithms consider the job to be complete when all points are either a single layer thick (so to speak), or no points can be deleted without changing the topological properties of the object. Note, however, that these are still points in a "discrete" space. If these points are regarded as being the centers of unit cells,[2] there is no reduction in dimension; and if they are regarded as points on a lattice (as, for instance, in [11]), there is no continuous skeleton to be inferred in their distribution.

The separate problem of inferring a continuous line/surface on the borders of discrete objects has been addressed at some length in the literature (see [10], [15]). However, these papers describe methods for inferring continuous surfaces onto boundaries which are assumed to be 1-manifold and 2-manifold (appropriately mapping these "continuous" topology concepts into the discrete topology), for 2D and 3D objects respectively. In general, a skeleton will not satisfy these assumptions. Furthermore, specific instances of objects can be found where no point can be removed while thinning (since that would change the topology of the object,) yet the points are so densely packed that no skeletal interpretation (in the continuous sense just mentioned) can be found. Figure 5 shows a 2D example of just such a case: no point is "simple", yet there is a 3×3 block in the center which defies any skeletal interpretation.

It seems to us that the research in digital thinning, motivated as it is by the needs of computer-based imaging, is always concerned with *digital-*, or discrete images. Hence, no attempt appears to have been made to *define*, and then search for, what might be termed a skeletal point. Instead, simple ($\equiv$ non-skeletal) points are identified, and removed. It might appear at first sight that, since points can be one of simple or skeletal, defining simple points automatically defines skeletal points as well. While this is true, it leaves unanswered the question of what, precisely, the characteristics of the skeleton are. Seemingly, the only skeletal characteristics of interest in image analysis are those which can described as "non-simple. Unfortunately, the only thing which can be said about non-simple points *vis-a-vis* the continuous skeleton is that the skeleton must pass through these points. Where the neighborhood of these points is sufficiently sparse, adjacency information could be used to construct the continuous skeleton, but, as figure 5 shows, this cannot be relied upon.

4 How is the MAT useful?

We have suggested that the MAT is potentially a very useful tool in design/manufacturing. However, given the undesirability (and difficulty) of using the MAT as the primary representation of a design, as also the difficulty of deriving the MAT (exact or approximate) of a given object, we appear to have reached an impasse. It is then worthwhile to take a closer look at why the MAT appears to be so useful. To this end, we examine one application: numerical analysis of injection molding.

As mentioned before, numerical analysis packages for injection molding represent the geometry of the part as a set of $2\frac{1}{2}$D facets, with thickness information stored at the vertices. Clearly, such a representation

[2] i.e., as pixels (unit squares) in 2D and voxels (unit cubes) in 3D, which is a common way of visualizing the object in R^n

should be topologically equivalent to the original object. Where the skeleton divides into plate-like components between junction points/edges, the proportions between these components should reflect the shape of the object. Surface properties of the part, such as notches or flutings, appear to affect the mold-filling analysis minimally, and should influence the skeleton also minimally. This suggests that any acceptable skeleton must meet the following specifications:

1. The skeleton should be a set with no interior.
2. The skeleton should be topologically equivalent to the original object.
3. The shape of the skeleton should be strongly influenced by that of the object. As we shall see, this can be tied to the *reconstructibility* of the object from the skeleton.

5 Terminology and Notation

Before proceeding further with the presentation, it is useful to define the terminology and notation to be used. The following terms are understood to have their usual meanings from mathematics, and are not defined here (see a text such as [16]):

open set, closed set, closure, interior, neighborhood, norm, L_1 (diamond) norm, L_∞ (box) norm, flat span (affine hull)

R is the set of real numbers

$P = \{x \mid x \geq 0\}, P^\times = P \setminus \{0\}$.

$\mathcal{E}$ is the modeling (flat) space, one of R^2 or R^3. We shall sometimes refer to R^n, but it should be understood in such cases that $n \in \{2,3\}$.

If $f : A \to B$ denotes a mapping, and if $A_1 \subset A$ and $B_1 \subset B$, then $f_>(A_1)$ denotes the *image* of A_1 under f, and $f^<(B_1)$ denotes the *pre-image* or *inverse image* of B_1 under f.

The symbol **Clo** is used to denote the closure, and **Int** to denote the interior of a set.

An *object* is the closure of some open set in space; we usually denote objects by O, perhaps with a subscript. Thus, $O = \mathbf{Clo}(\mathbf{Int}(O))$. In later sections, the term *object* shall be used to refer to a more restricted class of *discrete* objects.

Given a norm $d : E \times E \to P^\times$, a *norming cell* based on that norm is the set $\{x \mid d(0,x) \leq 1\}$.

A *cell* is defined as the set $C(d,c,r) = \{x \mid d(x,c) \leq r\}$, where d is a given norm, c is the center of the cell, and $r \in P^\times$, generically called the *radius*, is specified somehow.

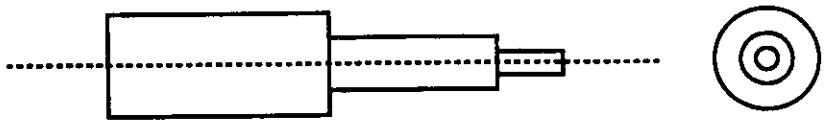
A cell *contained* in some object O is a cell which is a subset of O. A *maximal cell* contained in O is any cell, contained in O, which is not a subset of any other cell also contained in O.

The terms *disk* and *ball* are reserved for cells defined using the euclidean norm, in 2D and 3D respectively. Similarly, the term *box* is reserved for use with the box norm (L_∞ norm).

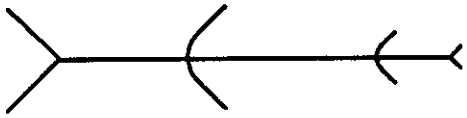
6 Skeletal Abstraction

The MAT is defined in terms of maximal euclidean cells (disks or balls) contained in the object. This is a covering of the object by elemental balls, abstractly represented by the set of centers of these cells. The word *elemental* is used to capture our sense of the simplicity of the shape of the disk or ball — it is sufficiently simple that a point located at its center is a good abstraction of its shape. It is worthwhile to note that it is in this sense that the MAT represents the shape of an object.

It is conceivable that under appropriate circumstances another shape would be considered simple. For instance, consider Figure 6. If the object shown in part (a) is to be turned on a lathe, its MAT (part (b) of



(a) A cylindrical part



(b) The Medial Axis of the part

(c) The generalized cylinder axes of the part

Figure 6: (a) A cylindrical part, (b) Its MAT, (c) The "generalized cylinders" skeleton

the figure) is not of much help in planning the tool path. On the other hand, if one applies the concept of *generalized cylinders* [1], shown in part (c) of the figure, then the axis of the cylinders, along with radius information, can be seen to be immediately useful. This suggests that alternate skeletons can be defined, based upon different notions of "simplicity" of shape, and reinforces our statement above that the MAT abstracts the shape of an object only to the extent that the euclidean ball is well-abstracted by its center.

The euclidean cell is thus seen to be just one of a class of elemental cells upon which to define a skeleton. We suggest that other skeletons could be defined upon the norming cells of other norms, such as the box norm (L_∞ norm), or the diamond norm (L_1 norm).[3] Accordingly, we term the MAT the *euclidean skeleton.*

Definition 1: Given an object and a norm, the closure of the set of centers of maximal cells contained in the object is defined to be the *skeleton* of the object, denoted by S. Associated with this skeleton is a function $\mathrm{r} : S \rightarrow P^{\times}$, called the *radius function*, which gives the norm of the maximal cell at each point in the skeleton.

The term *shape abstraction* is sometimes used by authors in describing the usefulness and properties of the euclidean skeleton. It is also used to describe the concept of shape features [7]. It is worthwhile to note that shape is a very fuzzy concept, and shape features are sometimes simply defined as geometric configurations of interest. If the claim is made that the euclidean skeleton abstracts the shape of the object, on what is this claim based? It is our contention that shape abstraction in the skeleton is a property of the

[3]This was also suggested by Lee [13]

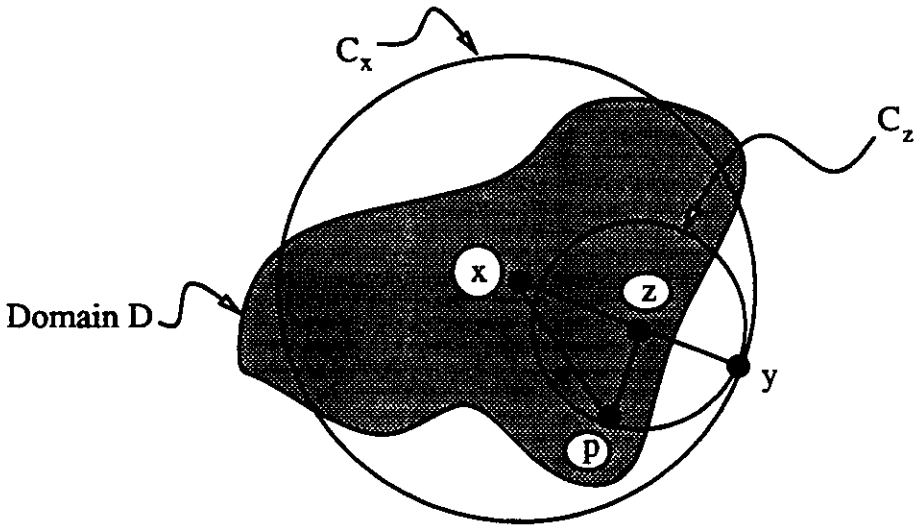


Figure 7: Figure for Theorem 1.

norming cell used for defining the skeleton. A euclidean skeleton is a "good" abstraction only if the ball is the simplest possible shape for the application at hand, as illustrated in figure 6.

Given an elemental cell, what can be inferred about the connectivity properties of the skeleton defined thereby? This appears to be a rather difficult question. Familiarity with the euclidean skeleton (MAT) leads one to suppose that the euclidean skeleton has the same homotopy type as the original object. However, this has been shown rigorously only very recently by Wolter [22], and that too only for objects with C^2 smooth boundaries. Certainly, this is not true for a skeleton defined on, say, the box norm, as Figure 8 shows. We speculate that, in general, the connectivity properties of objects are shared by the skeleton if and only if the cell boundaries are not flat anywhere.

Another property widely believed to be held by the euclidean skeleton, and recently shown by Wolter, is what we call dimensional reduction. More properly, this is the property of having no interior. It remains to be shown that this is true of *all* skeletons, and we present this as our first result:

Theorem 1: A skeleton has no interior.

We need the following lemma:

Lemma 1: A maximal cell touches the boundary of its containing object.

Outline of Proof of Lemma: Assume there is some maximal cell, $C(d, c, r)$ which does not touch the boundary of its containing object, O. Then all points in C are in $\mathbf{Int}(O)$, and $\forall x \in \mathbf{Bdy}(C(d, c, r))$ we can find $\epsilon \in P^{\times}$ such that $C(d, x, \epsilon) \subset O$. Let ρ be the infimum of all such ϵ. Then, clearly, the cell $C(d, c, (r + \rho)) \subset O$. This contradicts the maximality of cell $C(d, c, r)$.

Outline of Proof of Theorem 1: This can be shown by contradiction. Assume, then, that the skeleton S of some object O does, in fact, have a non-empty interior. Choose some component of this interior, say D, and a point $x \in D$. Let C_x be the maximal cell at x, with radius $\mathbf{r}(x)$. Refer to figure 7.

By lemma 1, $\mathbf{Bdy}(C_x) \cap \mathbf{Bdy}(O)$ is not empty. Choose some $y \in \mathbf{Bdy}(C_x) \cap \mathbf{Bdy}(O)$. If $\overline{(x, y)}$ denotes the line segment between x and y, then clearly $\overline{(x, y)} \cap D \neq \phi$. Choose some $z \in \overline{(x, y)} \cap D$. Since $z \in S$, there must be a maximal cell centered at z. The value of the radius function at z, $\mathbf{r}(z)$, must be less than the distance to the boundary along $\overline{(x, y)}$, $\mathbf{d}(y, z)$, since otherwise y would not be on the boundary of O. That is, $\mathbf{r}(z) \leq \mathbf{d}(y, z)$.

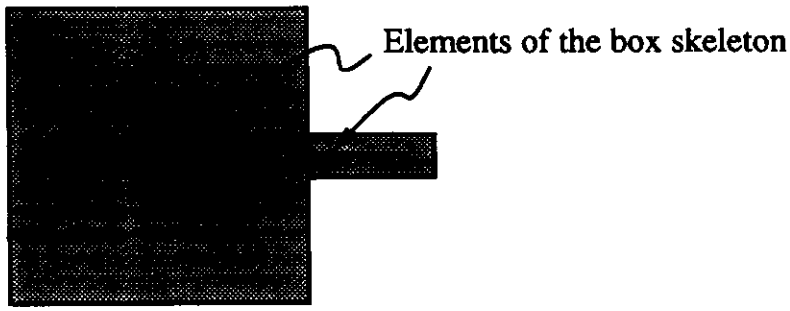


Figure 8: A simple object and its Box-skeleton. Note that the skeleton is disconnected, although the object is not.

Let C_z denote the maximal cell at z, i.e., $C_z = C(d, z, \mathbf{r}(z))$. For any $p \in C_z$, the following holds by the triangle inequality:

$$d(x,p) \leq d(x,z) + d(z,p)$$

Since $d(z,p) \leq \mathbf{r}(z) \leq d(y,z)$, this becomes

$$d(x,p) \leq d(x,z) + d(y,z)$$

Since $z \in \overline{(x,y)}$, clearly $d(x,z) + d(y,z) = d(x,y) = \mathbf{r}(x)$, and hence

$$d(x,p) \leq \mathbf{r}(x)$$

Since $p \in C_z$ was arbitrary, we conclude that $C_z \subset C_x$, which contradicts the assumption that C_z is maximal.

We next present an algorithm to derive a skeleton. We shall show that this is the box-skeleton. Since the box-skeleton need not be connected, and we want a skeletal set which is preserves the topology of the object, we present a procedure for augmenting the box skeleton with additional elements, thus yielding a set which

1. has no interior,
2. has the same homotopy type as the original object,
3. is a superset of the box skeleton which "fills-in" the gaps in the box skeleton,

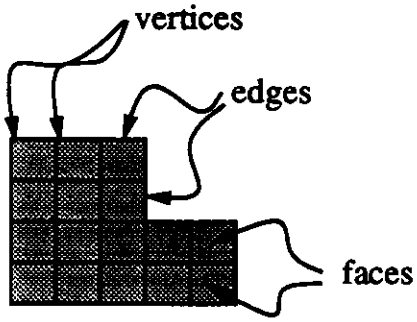
and thus is a good abstraction of the shape of the object, at least for some applications.

7 Dual-based Thinning

In order to describe our thinning process, we first define some additional terms.

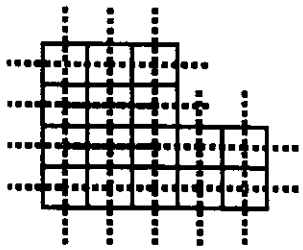
A *unit box* is a pixel if $\mathcal{E}$ is 2D, and a voxel if it is 3D.

A *space graph* G^* is a partition of the space, $\mathcal{E}$. The subsets under this partition, called the elements of G^*, are all connected components, and furthermore are restricted to being open sets in their flat spans in $\mathcal{E}$. This means that the elements of G^* are points, straight line segments, planar polygons, and (for 3D) polyhedra. The dimension of a subset is understood to be the dimension of its *flat span*. Then, the 0D



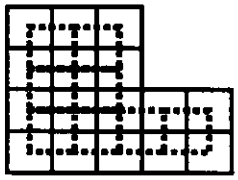
A 2D discrete object as a graph

(a)



Part of graph (bold) and part of its dual (dotted)

(b)



The graph (bold) and the restricted dual (dotted)

(c)

Figure 9: Graphs and duals.

subsets are called *vertices*, the 1D subsets are called *edges*, 2D ones are *faces*, and 3D ones are *solids*. Thus, $\mathcal{E} = \bigcup_{A \in G^*} A$.

Two elements A and B of G^* are said to be *adjacent* if $\mathbf{Clo}(A) \cap \mathbf{Clo}(B) \neq \phi$.

$\mathcal{P}(A)$ is the set of all subsets of A.

$\mathcal{P}_G(\mathcal{E})$ is the set of all space graphs of $\mathcal{E}$.

The *dual mapping* $\mathcal{D}^* : G^* \rightarrow G^{**}$, where $G^{**} \in \mathcal{P}_G(\mathcal{E})$, is an adjacency- preserving mapping. The image of G^* under this mapping, $\mathcal{D}^*{}_>(G^*)$, is also a graph. For the case where $\mathcal{E}$ is 2D, the mapping, and hence G^{**}, is specified as follows (see figure 9):

1. The image of every face is a dual vertex contained in the face.
2. The image of every edge is a dual edge crossing the given edge. The end-points of the dual edge are the dual images of the two faces adjacent to the given edge.
3. The image of every vertex is a dual face in which the vertex lies. The corners of this dual face are the dual images of faces adjacent the given vertex.

This can be extended to 3D as below:

1. The image of every solid is a vertex contained in the solid,
2. The image of every face is a dual edge through the face. The ends of this dual are the images of the two solids on either side of the face.
3. The image of every edge is a dual face through which it passes. The corners of this dual face are the images of the solids adjacent to the edge.
4. The image of every vertex is a dual solid, bounded by the images of the edges adjacent to the vertex.

That $\mathcal{D}^*$ is adjacency preserving can be seen by constructing an *adjacency graph*, as follows. The elements of the adjacency graph are termed *nodes* and *links*, to avoid confusion. Each element of G^* has a corresponding (unique) node in the adjacency graph. The links of the adjacency graph join two nodes whose corresponding elements in G^* are adjacent. Then it is easy to see that the adjacency graph of any G^* and that of $\mathcal{D}^*{}_>(G^*)$ are isomorphic. It is thus apparent that the dual graph of any G^* is an abstract representation of the adjacencies of elements in G^*.

Given any object O, the *restriction* of any $G^* \in \mathcal{P}_G(\mathcal{E})$ to the object O is defined to be the subgraph of G^* whose vertices are in O, and denote it by $G^* \mid_O$. We use the notation $\mathcal{P}_G \mid_O$ to denote the set of all such restrictions.

If a restriction to O, $G = G^* \mid_O$, is such that $O = \bigcup_{A \in G} A$, then G^* is said to be *compatible* with O, and the restriction G is called the *object graph* of O.

Let O be an object, let $G^* \in \mathcal{P}_G(\mathcal{E})$, and $G = G^* \mid_O$. Then it is of interest to examine the restriction of the dual to O, $\mathcal{D}^*(G^*) \mid_O$. Since this will be of great use later, a new mapping is defined from $\mathcal{P}_G \mid_O$ to itself,

$$\mathcal{D} : \mathcal{P}_G \mid_O \rightarrow \mathcal{P}_G \mid_O$$

which associates each element $G \in \mathcal{P}_G \mid_O$ with an image, also in $\mathcal{P}_G \mid_O$, such that this image is the restriction of $\mathcal{D}^*(G^*)$ to O.

7.1 Duals of Discrete Objects

As mentioned in the section on digital thinning (see section 3), a discrete object can be described using a "lattice" of integer tuples. It can also be interpreted as an object in R^n, by considering each point to be a unit box centered at the coordinates given by the integer tuple. It is immediately apparent that under the definition of a graph given above, the set of vertices, edges, faces, etc. of this object is also an object graph, say G. If the centers of these boxes are regarded as being at the integer tuple, then the corners are at non-integral coordinates. For simplicity, we assume instead that the corners of these boxes are at integer coordinates. It is then clear that all edges of the graph so defined are of unit length, all faces are unit squares, and all solids are unit cubes.

The dual mapping $\mathcal{D}^*$ defined above for any space graph is enhanced to be geometrically precise when the objects are discrete. This can be achieved by simply requiring the image under D^* of any pixel/voxel to be a vertex *located at its center*. Note that this achieves geometric precision only for *restricted* space graphs, since the description of a discrete object would typically not define any structure on the components of the background.

A very important point is that if the space $\mathcal{E}$ is assumed to be the union of closed unit boxes (pixels/voxels), then the interior of these boxes defines the n-dimensional cells of some space graph G^*, and their boundary elements define the lower-dimensional cells. Then the dual of every such $G^* \in \mathcal{P}_G(\mathcal{E})$ *is also composed of closed unit boxes.* That is, duals of discrete objects are also discrete objects in the same sense, although "shifted" in space by one-half length. Therefore, $\mathcal{D}^*(\mathcal{D}^*(G^*)) = G^*$. The thinning procedure (to be described below) uses this property implicitly. From here on in this document, we assume this to be the case.

(a) Vertex neighbors (b) Dual consists of two vertices

Figure 10: Vertex neighbors and their dual.

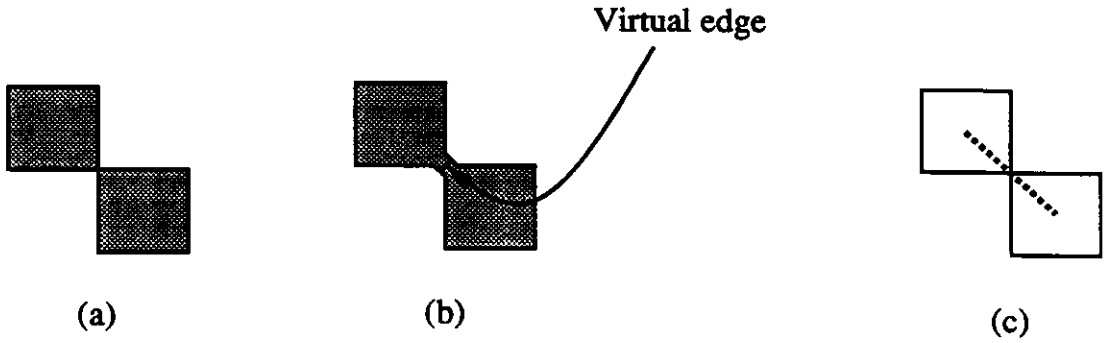


Figure 11: Virtual edge and its dual.

One other issue needs to be addressed. In such a discretized object, it is not necessarily true that $\mathbf{Int}(O)$ is connected. If $\mathbf{Int}(O)$ is not connected, but O is, as can be seen in figure 10, then the question arises whether we wish to regard the object as being a single component or not. This is precisely the problem of what connectivity to impose on discrete objects (see section 3), and the results of digital topology can be used here. As has been noted by Rosenfeld, it is necessary to have different types of connectivity for the object and the complement, in order to avoid inconsistencies (see [11]). Although many choices exist, for our purposes we find that one is superior to the rest: once and for all, we choose the most extensive connectivity for the object, and the most restrictive one for the complement. Thus, in 2D the connectivity pairs of the object and the complement are (8, 4), and in 3D they are (26, 6).

This needs to be reflected in the dual. As defined, the dual of the graph in figure 10 would be two unconnected vertices. This does not reflect our understanding that the two are, in fact, connected. To get around this problem, define the *immediate neighborhood* of a point to be some cell of size smaller than the resolution of the discretization. Define the immediate neighborhood of an edge to be the immediate neighborhood of its midpoint. Denote the immediate neighborhood by N. Examine the immediate neighborhood of the edges (for 3D) and vertices (both 2D and 3D) of the graph. Let n be the number of connected components in $N \cap O$, and n' be the number of connected components in $N \setminus O$. If either $n > 1$ or $n' > 1$, then we capture the requirement of connectivity by inferring a "virtual" face at the edge or vertex in the 3D case, and a virtual edge at the vertex in 2D. This is illustrated for 2D in figure 11. The purpose of so "promoting" some of the lower-dimensional entities of the graph of the object is to enable the dual to correctly reflect the connectivity properties required of the object. That this works in two dimensions is immediately obvious from the figure.

In 3D, this is less obvious, but equally effective. Even in 3D connectivity could still be a matter of having just an edge between two edge-neighboring or vertex-neighboring voxels. In the first case, the common edge gets promoted to a virtual face, and its endpoints to virtual edges. This is illustrated in figure 12. In the latter case, the vertex gets promoted directly to a face. In both cases, it is trivial to see that the dual edge

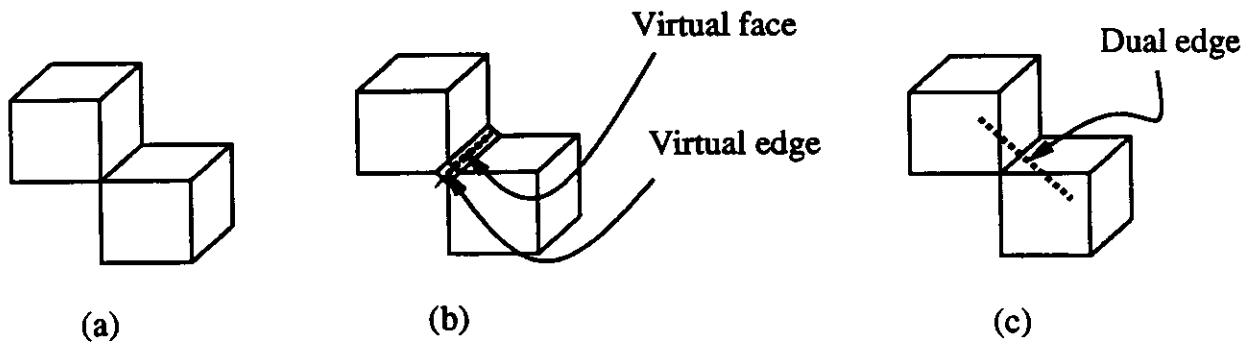


Figure 12: One example of virtual entities in 3D.

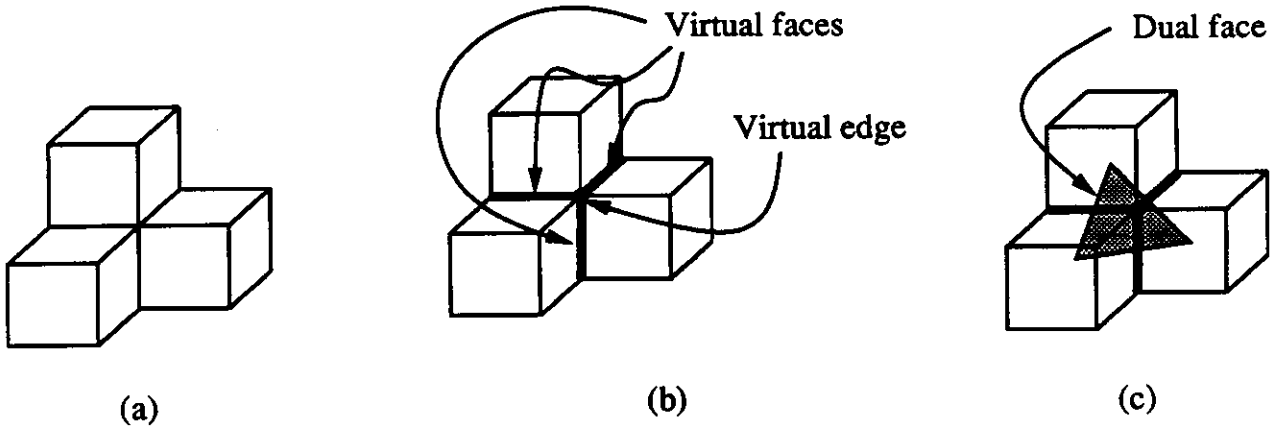


Figure 13: A dual face inferred by identifying a circuit of solids around a virtual edge.

does get constructed. For the more complex 3D situations where the construction of a dual *face* is necessary, note that there must be a circuit of solids around some vertex in the graph. One example of this is shown in figure 13. This common vertex gets promoted to a virtual edge when one or more of its neighboring edges gets promoted to a face. Then this virtual edge is surrounded by the object, and a circuit of solids can be found around it; this circuit then defines a dual face.

7.2 Dual-based Thinning Procedure T

The thinning properties of the dual mapping are first illustrated in figure 14 for a 2D example. The sequence of figures there shows the results of the repeated application of $\mathcal{D}$, starting with an object graph G. Define

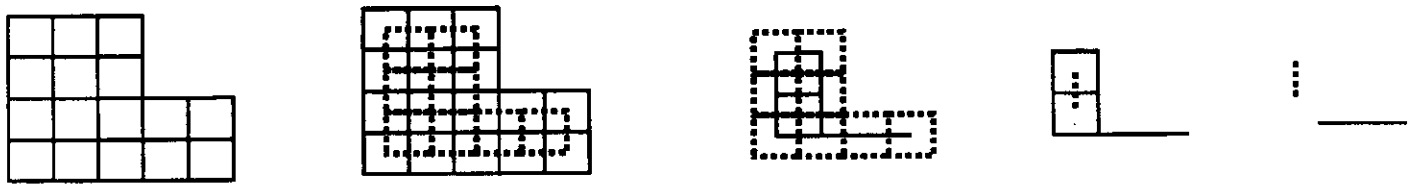


Figure 14: A sequence of duals, showing the object getting thinner.

$O_1 = \mathbf{Clo}(\mathbf{Int}(\bigcup \mathcal{D}(G)))$. We denote the set $\bigcup \mathcal{D}(G) \setminus O_1$ by S_1. Clearly, O_1 is also composed of pixels, and thus has an associated object graph, say G_1. The process can be repeated for G_1, and yields a family of sets S_i, $i \in N$. The process terminates when (and if), for some $i \in N$, $O_i = \phi$.

Proposition 2 (Shrinking Proposition): If an n-dimensional object O and its object graph G are given, then $\mathcal{D}(G)$ consists of strictly fewer n-dimensional cells than G.
Discussion: Let G^* be some space graph compatible with O. Then $G \subset G^*$. The n-dimensional cells of $\mathcal{D}(G)$ are images under $\mathcal{D}^*$ of the vertices of G^*. Since $\mathcal{D}(G)$ is the subgraph of $\mathcal{D}^*(G^*)$ whose vertices are properly contained in O, it is clear that the vertices of G which are on the boundary of O have no corresponding n-dimensional cell as their image in $\mathcal{D}(G)$. Thus, the number of n-dimensional cells in the dual is strictly less than the number of vertices of G which are in the interior of O. It remains to be shown that the number of such vertices is strictly less than the number of n-dimensional cells of G. We expect to show this by using, perhaps, the work of Morgenthaler reported in [14].

This suggests the definition of **thinning procedure** $\mathcal{T}$, as follows: the procedure starts with some object O, and an associated object graph G. Let G^* denote the space graph of which G is a restriction. We define $O_0 = O$, $G_0 = G$, $S_0 = \phi$, and i = 0. Then the following steps are applied:

1. If $O_i \neq \phi$, identify $\mathcal{D}(G_i)$.
2. Let $O_{i+1} = \mathbf{Clo}(\mathbf{Int}(\bigcup \mathcal{D}(G_i)))$,
3. $S_{i+i} = \mathbf{Clo}(\bigcup \mathcal{D}(G_i) \setminus O_{i+1})$.
4. If i is even, $G_{i+1} = G^* \mid_{O_{i+1}}$. If i is odd, $G_{i+1} = \mathcal{D}^*(G^*) \mid_{O_{i+1}}$.
5. If $O_{i+1} \neq \phi$, increment i by 1, and repeat from step 1.

By Proposition 2, procedure $\mathcal{T}$ terminates. Then define $S_\infty = \bigcup S_i$, for all $i \in N$ for which S_i is defined. We contend that S_∞ is a skeleton, specifically,

Proposition 3: Thinning procedure $\mathcal{T}$ yields the set S_∞ which is the box-skeleton of the object.
Discussion: By definition of each S_i (above), S_∞ has no interior. We will show that all the points of S_∞ are the centers of maximal boxes which can be fit in the object.

It can be seen that each step of T is invertible. That is, given some $\mathcal{D}(G_i)$, it is possible to reconstruct the graph G_i by simply replacing all vertices in $\mathcal{D}(G_i)$ with unit boxes centered at those vertices. Moreover, it is clear that if each point in $\bigcup \mathcal{D}(G_i)$ is replaced by a unit box, then the union of all these is precisely O_i. Then, if at every stage i of procedure $\mathcal{T}$, we associate with each point of S_{i+1} the integer i, it should be equally clear that the original object O can be recovered by replacing each point of S_∞ by a unit box of size specified by the integer associated with that point.

Proposition 4: Under procedure $\mathcal{T}$, $\mathcal{D}(G_i)$ is a deformation retract of G_i, for each $i \in N$ for which $O_i \neq \phi$.
Discussion: We expect to show this by *constructing* a *retract* mapping, and a homotopy. This closely follows the spirit of Wolter's work in exploring the properties of the euclidean skeleton [22].

The retract

$$R_t : O_i \rightarrow \bigcup \mathcal{D}(G_i)$$

is a continuous map, such that $R_t(x) = x, \forall x \in \bigcup \mathcal{D}(G_i)$.

The homotopy

$$h(x,t) : O_i \times I \rightarrow \bigcup \mathcal{D}(G_i)$$

where $I = [0, 1]$, is such that

$$h(x, 0) = x, \text{ and } h(x, 1) = R_t(x), \forall x \in O_i$$

and

$$h(x, t) = x, \forall x \in \bigcup \mathcal{D}(G_i)$$

Figure 15 depicts, in 2D, the retraction mapping in two regions of the digital asterisk. The arrows depicting the retract mapping in figure 15(c) can themselves be regarded as the homotopic "traces", as the parameter t varies from 0 to 1. We expect to show that there are really only three types of geometries (locally speaking) in the dual-based thinning of 2D discrete objects: endpoints of skeletal arcs, junctions of skeletal arcs and either some object component or other skeletal arcs, and lastly an object component inside another object component. By enumerating such situations, we expect to show that proposition 4 is true for 2D. It is also expected that a similar classification of the types of geometries can be done for 3D.

7.3 Shrink Wrap Procedure $\mathcal{W}$

From figure 8, it is clear that the box skeleton does not share the topological properties of the original object. Although we expect to show that the dual of a graph is a deformation retract of the graph, procedure $\mathcal{T}$ does not operate with such duals. There is a step whereby a new object is identified by taking the closure of the interior of the point-set represented by the dual (step 2 of procedure $\mathcal{T}$), and this step leads to a disconnection between S_i and S_{i+1}.

We now define a procedure, called the **shrink wrap procedure** $\mathcal{W}$, which guarantees that the retract and the homotopy defined above, are maintained across the stages of procedure $\mathcal{T}$. These mappings under $\mathcal{W}$ are identical to their counterparts under $\mathcal{T}$, except in the vicinity of points of $S_i \cap O_i$, which are the points where connection is lost between S_i and $\bigcup \mathcal{D}(G_i)$. Informally, the procedure operates like a plastic wrap around the object. As the object thins, the wrapping "shrinks", but does not tear.

The retract mapping under procedure $\mathcal{W}$ is not different from the one shown in figure 15 if $S_i \cap O_i = \phi$. If the set is not empty, then $\mathcal{W}$ redefines the mapping for points in O_i *in the vicinity of* points in $S_i \cap O_i$, as shown in figure 16. This is further illustrated in figure 17.

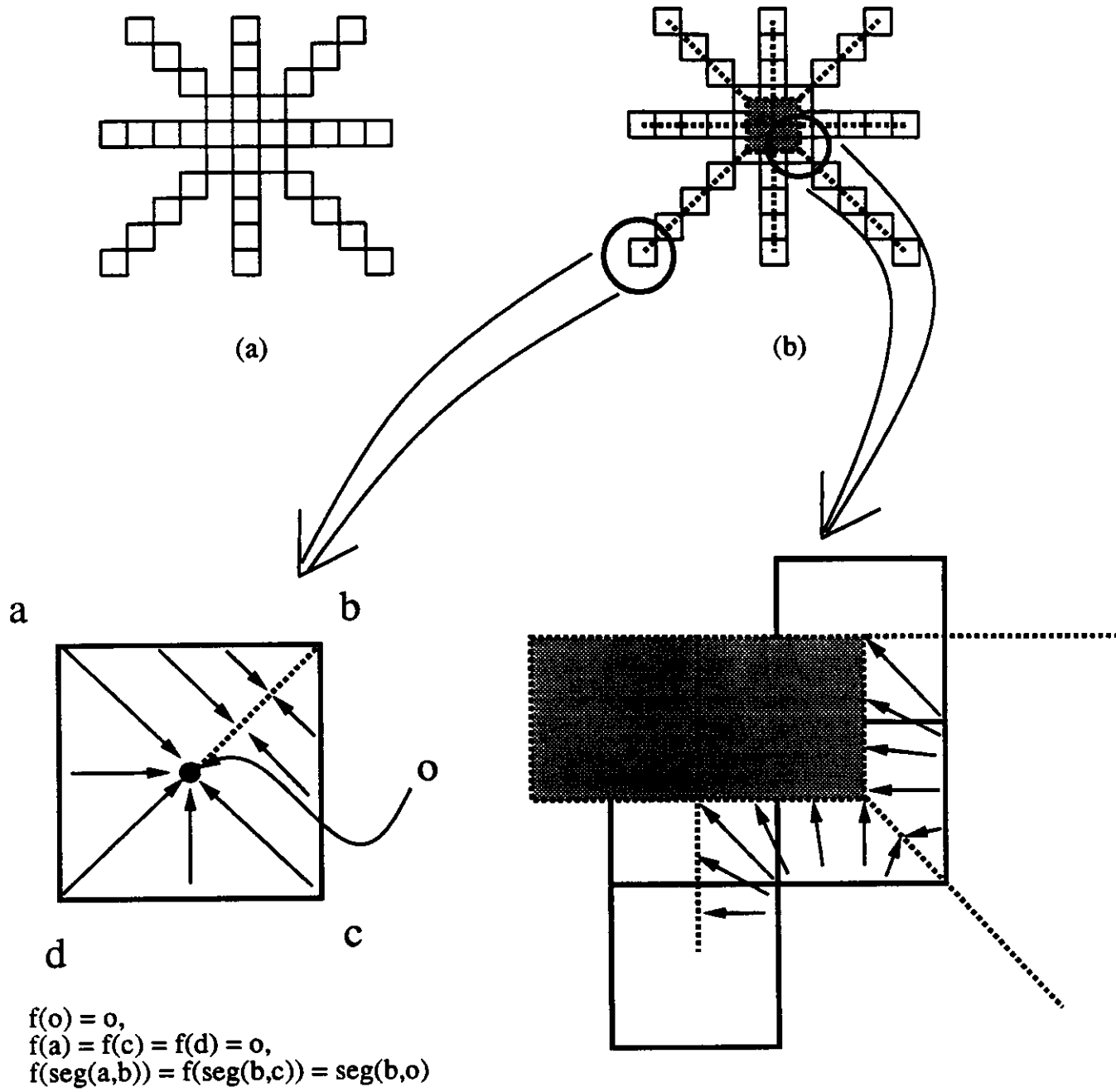
The general procedure for constructing the retraction mapping R_w under the shrink-wrap procedure $\mathcal{W}$ is given next. As mentioned before, this mapping is identical to R_t, except in the vicinity of points in $S_i \cap O_i$. By *vicinity* is meant a neighborhood smaller than our resolution, which is unity. Once again, we first explain the 2D case, and the reader is referred to figure 16. Consider the immediate neighborhood of some point $x \in S_i \cap O_i$ to be a box of size less than unity, centered at x. Denote this neighborhood by $N^*(x)$. Recalling from the previous section that $h : O_i \times I \rightarrow \bigcup \mathcal{D}(G_i)$ is the homotopy between points of O_i and points of $\bigcup \mathcal{D}(G_i)$, the mapping R_w for points in $\mathbf{Bdy}(O_i) \cap N^*(x)$ is as follows:

1. Augment $\mathcal{D}(G_i)$ with the set $C_{i+1} = \{y \in O_i \mid y = h(x, t), t \in [0, 1]\}, \forall x \in S_i \cap O_i$.
2. $\forall p \in \mathbf{Bdy}(O_i) \cap \mathbf{Bdy}(N^*(x)), R_w(p) = R_t(x)$.
3. Points of $\mathbf{Bdy}(O_i)$ between x and each point in $\mathbf{Bdy}(O_i) \cap \mathbf{Bdy}(N^*(x))$, map to *an appropriate point* in C_{i+1}. That is, $\mathbf{Bdy}(O_i) \cap (N^*(x)$ is a 1-manifold segment containing x; then

$$R_{w>}(\mathbf{Bdy}(O_i) \cap N^*(x)) \subset C_{i+1}$$

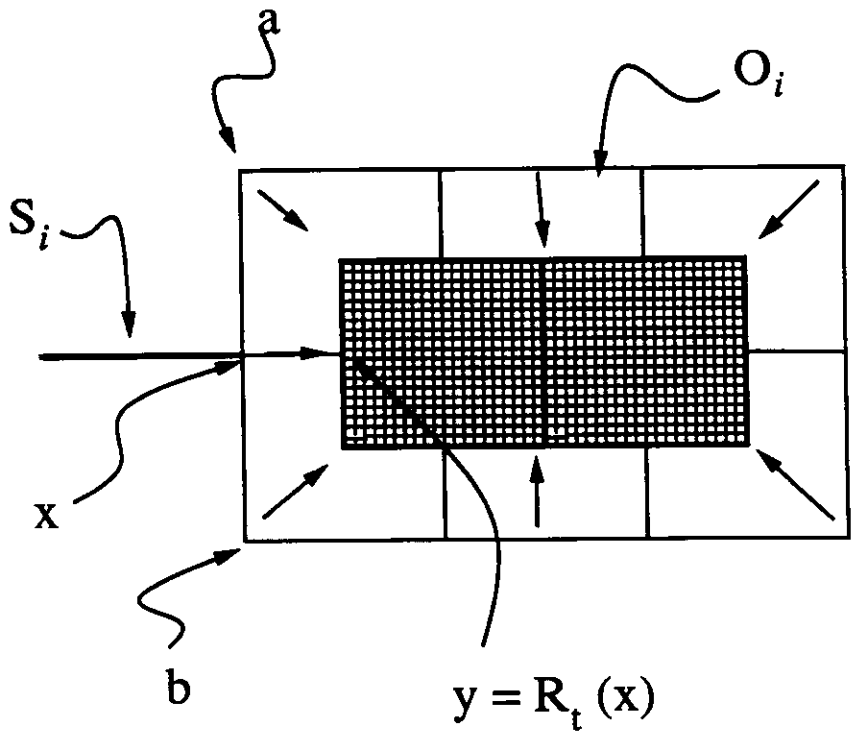
and

$$R_w(x) = x, \forall x \in S_i \cap O_i$$

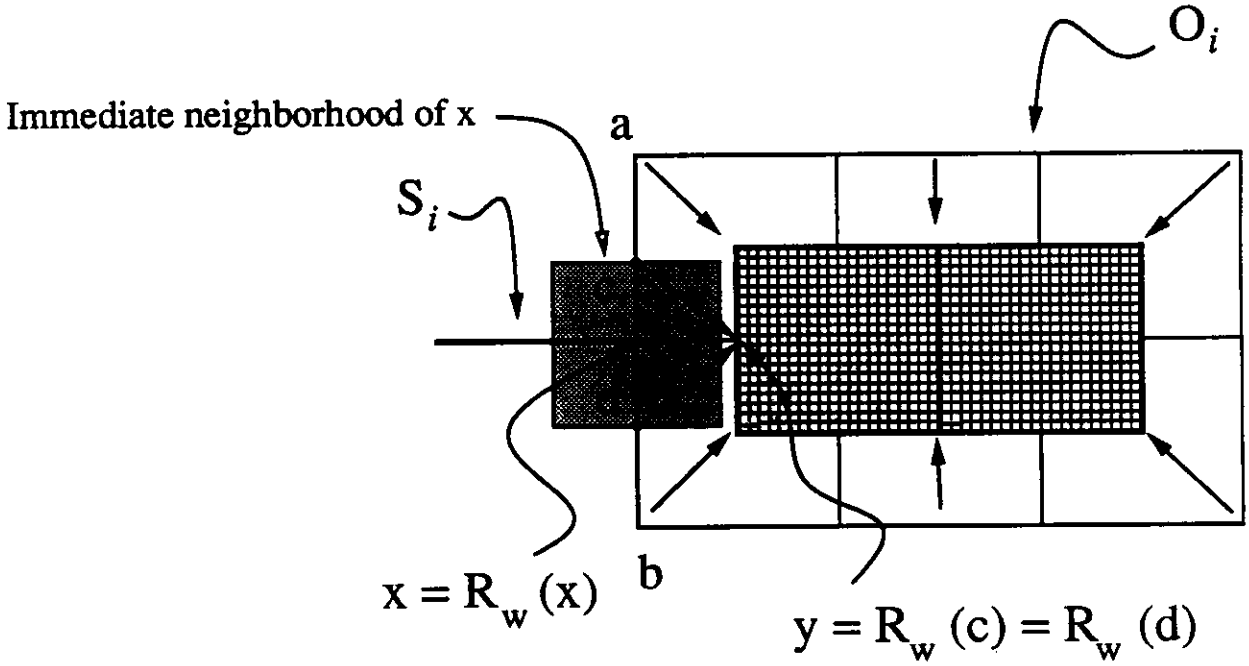


(c) The retraction mapping in two regions

Figure 15: Illustration of retract mapping for 2D.



(a) Retract mapping under procedure T



(b) Retract mapping under W

Figure 16: The construction of the retract mapping, under the shrink-wrap procedure.

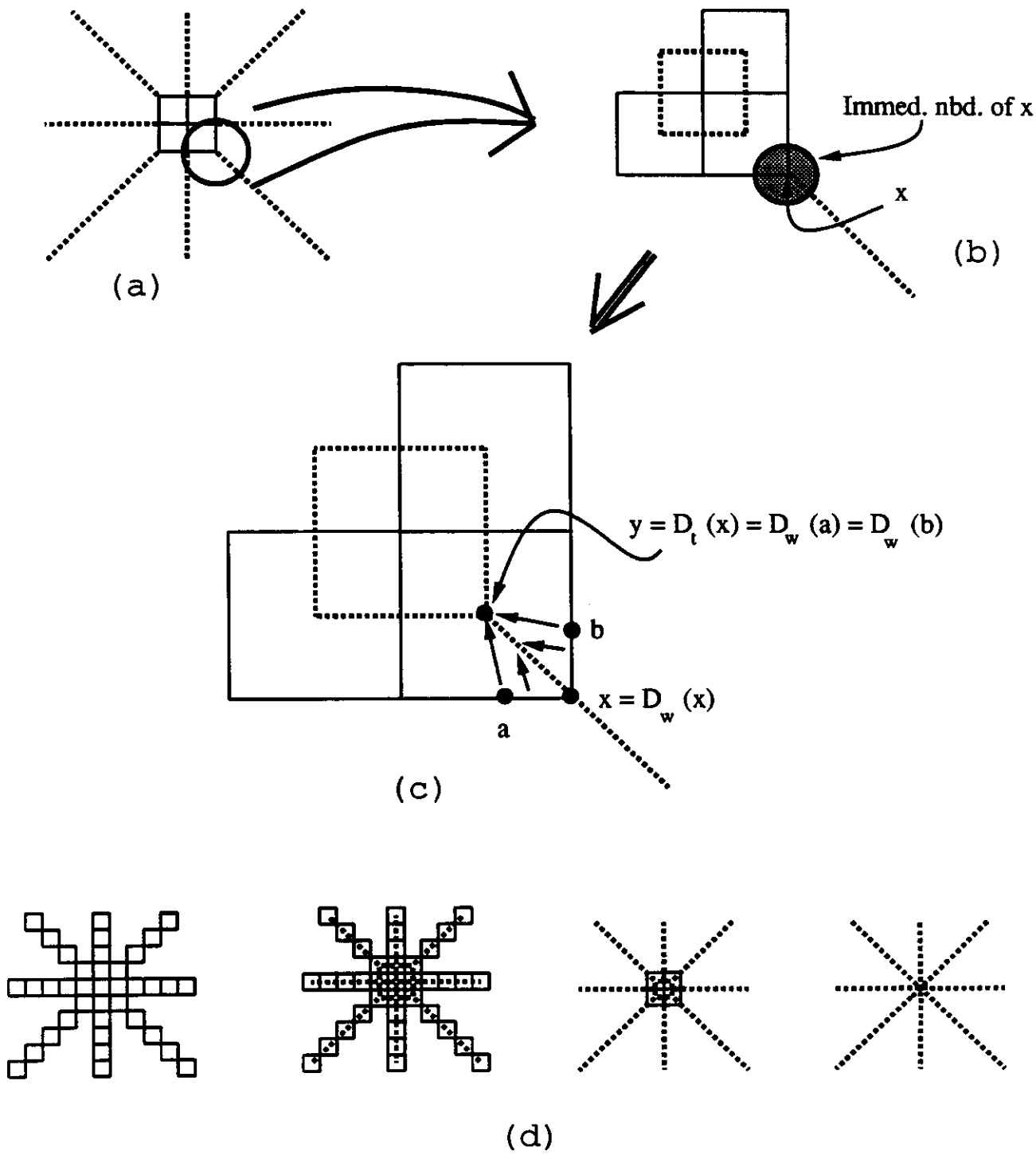


Figure 17: Illustration of shrink-wrap procedure, and skeleton of the asterisk.

In figure 16, $y = R_t(x)$, and $\mathbf{Bdy}(O_i) \cap \mathbf{Bdy}(N^*(x)) = \{c, d\}$. Hence, c and d map under R_w to y, which was the image of x under the earlier map R_t. The point $x \in S_i \cap O_i$ maps to itself under R_w. The straight line segment $\overline{(x,y)}$ is the the path under homotopy (in procedure T) of x, and hence the segments $\overline{(x,c)}$ and $\overline{(x,d)}$ map to this segment, and $R_w(x) = x$.

This extends to 3D cases also. The difference is that $S_i \cap O_i$ need not be isolated points, but could also be edges (segments). In this case, we identify the immediate neighborhood for each point on the segment, as before, but consider the union of all such neighborhoods together. Consider figure 18. Part (a) shows a simple 3D figure and its dual, which happens to be skeletal. Part (b) is the same 3D figure, except with a skeletal component adjacent to it. Part (b) shows the construction of the composite immediate neighborhood of edge segment $\overline{(a,b)}$. If we use the notation $N_{seg}(a,b) = \bigcup_{x \in \overline{(a,b)}} N^*(x)$, then $N_{seg}(a,b) \cap \mathbf{Bdy}(O_i)$ is the shaded region on the faces of the voxels in part (b) of the figure. All points in $\mathbf{Bdy}(N_{seg}(a,b)) \cap \mathbf{Bdy}(O_i)$ map to $R_{t>}(\overline{(a,b)})$. Part (d) of the figure shows the set $C_{i+1} = \{y \in O_i \mid h(x,t) = y, t \in [0,1]\}$, for each $x \in S_i \cap O_i$ Thus, $N_{seg}(a,b) \bigcap O_i$ (shown shaded in part (b) of the figure), maps to the figure marked C_{i+1} in part (d). Part (d) shows the skeletal components from the two iterations, S_i and C_{i+1}.

Proposition 5: Procedure $\mathcal{W}$ yields a set S_∞ which has the following properties:

1. S_∞ has no interior.
2. The box skeleton of O is a subset of S_∞.
3. S_∞ has the same homotopy type as the original object.

8 Applications

In this section, some possible applications of the box-skeleton are presented. These applications are, broadly,

1. Numerical analysis of injection molding.
2. Feature recognition in injection molding design.
3. Automatic mesh generation.
4. Initial solution to finding critical points in tracing the euclidean skeleton.
5. Shrinkage analysis.

The first two have been explored in some detail using the results of our early attempts at skeletonization. Some of the ideas in this work are reported in [7], and also in [8]. An early implementation of our ideas was carried out, and showed great promise in handling complex injection molding parts from industry.

The use of the euclidean skeleton in automatic mesh generation has been well-explored by Gursoy [5], Tam et al [18], and others. It is the extension of their methods to three dimensions that we plan to explore.

Hoffman and others are interested in extracting the euclidean skeleton of 3D objects. To this end, they have carried out substantial work in finding the cut locus of two disjoint components of the boundary of 3D obects. As mentioned earlier, one important aspect of the problem of finding the euclidean skeleton is the identification of the so-called critical points, which they do not address. We expect that the critical points of the skeleton we identify could be used as good initial guesses for the critical points of the euclidean skeleton.

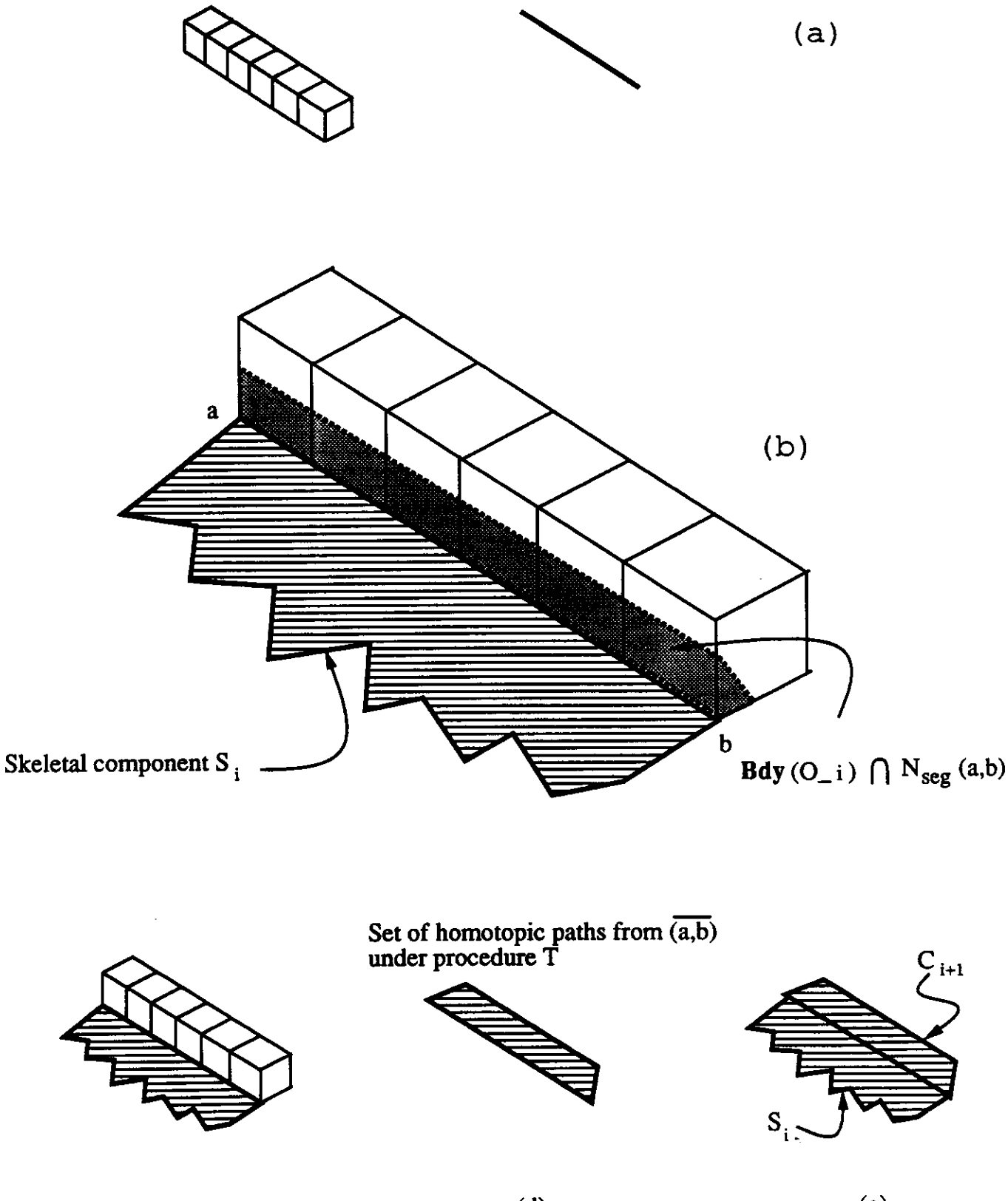


Figure 18: Shrink-wrap procedure in 3D.

9 Summary

We have indicated that the euclidean skeleton (MAT) is a useful alternate description of the geometry of physical objects. Nevertheless, it has proved very difficult to use the euclidean skeleton in engineering design, since it is very hard to define algorithms for finding the euclidean skeleton of 3D objects. We have indicated that although the 3D euclidean skeleton remains elusive, other skeletons can be defined which share some of the properties of the euclidean skeleton.

We have described a procedure for finding one such alternate skeleton, and have indicated that this skeleton is closely related to the box skeleton. Our procedure yields a skeleton which, we contend, shares the following properties with the euclidean skeleton:

1. dimensional reduction (i.e., no interior)
2. topological equivalence (i.e., the skeleton is a deformation retract of the object)
3. shape abstraction (i.e., the skeleton is based upon a fundamental shape which is simple, just as the euclidean ball is simple)

It is the basic contention in this work that the skeleton so defined will prove useful in a wide variety of engineering design problems, and also in tracing the euclidean skeleton of objects.

Readers will note that the technique presented here has a strong resemblance to the Voronoi-based approximation techniques of Tam [20] and Yu [23]. That is to say, if the initial object is thin (in the sense that all the vertices in the object graph are on the boundary of the object), then the skeleton we obtain is very similar to the Voronoi-based approximate skeletons. In this sense, this work is a generalization of the concept, since our technique can handle the case of internal vertices as well. Since we can only operate with discrete (pixel/voxel based) objects, our technique does not converge to the euclidean skeleton. However, as discussed, this does not deter the applicability of our work in engineering design.

The proofs of various propositions put forth in this document is the focus of current work.

References

[1] T.O. Binford.
Visual perception by computer.
In *IEEE Systems Science and Cybernetics Conference, Miami*. IEEE, Dec 1971.

[2] H. Blum.
A transformation for extracting new descriptors of shape.
In W. Whaten-Dunn, editor, *Models for Perception of Speech and Visual Form*, Cambridge, MA, 1967. MIT Press.

[3] J. Corney and D.E.R. Clark.
A feature recognition algorithm for multiply connected depressions and protrusions in 2.5d objects.
In Jarek Rossignac and Joshua Turner, editors, *Proceedings of Symposium on Solid Modeling Foundations and CAD/CAM Applications*. 1991.

[4] Debashish Dutta and Christoph Hoffman.
A geometric investigation of the skeleton of csg objects.
In *Proceedings of ASME Conference on Design Automation, Chicago*. ASME, 1990.

[5] Halit Nebi Gursoy.
Shape Interrogation by Medial Axis Transform for Automated Analysis.
PhD thesis, MIT Department of Ocean Engineering, 1990.

[6] Kimberly Jyl Hafford and Kendall Preston, Jr.
Three-dimensional skeletonization of elongated solids.
Computer Vision, Graphics and Image Processing, 27:78–91, 1984.

[7] Mark Hall, Rajit Gadh, Atul Sudhalkar, Levent Gursoz, and Fritz Prinz.
Feature abstraction in a knowledge-based critique of designs.
In *Proceedings of ASME Winter Annual Meeting, Chicago 1990*. ASME, 1990.

[8] Helgi Hjálmarsson.
Master's thesis (in preparation).
Master's thesis, Mechanical Engineering, Carnegie Mellon University, 1992.

[9] Christoph M. Hoffman.
Computer vision, descriptive geometry, and classical mechanics.
Technical Report CSD-TR-91-073, Computer Sciences Department, Purdue University, Oct 1991.

[10] T.Y. Kong and A.W. Roscoe.
Continuous analogs of axiomatized digital surfaces.
Computer Vision, Graphics and Image Processing, 29:60–86, 1985.

[11] T.Y. Kong and Azriel Rosenfeld.
Digital topology: Introduction and survey.
Computer Vision, Graphics and Image Processing, 48:357–393, 1989.

[12] D.T. Lee.
Medial axis transformation of a planar shape.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 4(2):363–369, July 1982.

[13] D.T. Lee and R.L. Drysdale.
Generalization of voronoi diagrams in the plane.
SIAM Journal on Computing, 10(1):73–87, Feb 1981.

[14] David G. Morgenthaler.
Three-dimensional digital topology: The genus.
Technical Report TR-980, University of Maryland Computer Vision Laboratory, College Park, MD 20742, November 1980.

[15] David G. Morgenthaler.
Three-dimensional simple points: Serial erosion, parallel thinning, and skeletonization.
Technical report, University of Maryland Computer Science Center TR-1005, 1981.

[16] Walter Noll.
Finite Dimensional Spaces.
Martinus Nijhoff Publishers, 1987.

[17] F.P. Preparata and M.I. Shamos.
Computational geometry : an introduction.
Springer-Verlag, New York, 1985.

[18] Azriel Rosenfeld.
Three dimensional digital topology.
Information and Control, 50:119–127, 1981.

[19] T.K.H. Tam and C.G. Armstrong.
2d finite element mesh generation by medial axis subdivision.
Advances in Engineering Software and Workstations, 1992.

[20] T.K.H. Tam, M.A. Price, C.G. Armstrong, and R.M. McKeag.
Computing the critical points on the medial axis of a planar object using a delaunay point triangulation algorithm.
submitted to IEEE Pattern Analysis and Machine Intelligence, 1992.

[21] Y.F. Tsao and K.S. Fu.
A parallel thinning algorithm for 3-d pictures.
Computer Graphics and Image Processing, 17:315–331, 1981.

[22] Franz-Erich Wolter.
Cut locus and medial axis in global shape interrogation and representation.
Technical Report DL Memo 92-2, Design Laboratory, Department of Ocean Engineering, MIT, Cambridge, MA 02139, Jan 1992.

[23] Xinhua Yu, John Goldak, and Lingxian Dong.
Constructing 3d discrete medial axis.
In Jarek Rossignac and Joshua Turner, editors, *Proceedings of Symposium on Solid Modeling Foundations and CAD/CAM Applications*, 1991.