

**Model-Based Penetration Path Calculations
For The Diagnosis Of Multiple Trauma**

**MS-CIS-90-91
LINC LAB 188**

Leonard J. Karpf

**Department of Computer and Information Science
School of Engineering and Applied Science
University of Pennsylvania
Philadelphia, PA 19104-6389**

December 1990

UNIVERSITY OF PENNSYLVANIA
THE MOORE SCHOOL OF ELECTRICAL ENGINEERING
SCHOOL OF ENGINEERING AND APPLIED SCIENCE

MODEL-BASED PENETRATION PATH CALCULATIONS
FOR THE DIAGNOSIS OF MULTIPLE TRAUMA

Leonard J. Karpf

Philadelphia, Pennsylvania

December 1990

A thesis presented to the Faculty of Engineering and Applied Science of the University of Pennsylvania in partial fulfillment of the requirements for the degree of Master of Science in Engineering for graduate work in Computer and Information Science.

Bonnie Lynn Webber
(Advisor)

Mitch Marcus
(Graduate Group Chair)

Abstract

In order to enhance TraumAID, a system that provides decision support in the initial definitive management of multiple trauma, TSARR (TraumAID System for Anatomical Representation and Reasoning) has been designed to address the need for greater depth in TraumAID's understanding of anatomical reasoning. TSARR provides a framework for representing a three-dimensional model of relevant parts of the body; utilizing this model, TSARR is able to calculate three-dimensional representations of paths of injury, generated from wound locations input to the system. Using these paths, the system hypothesizes which anatomical structures in the patient might have been injured due to their location along a possible path of an injury. In the future, TraumAID will be able to utilize this information to focus its attention more accurately on specific areas of the body that have sustained injury.

This work has been done in conjunction with the TraumAID project being conducted by Professor Bonnie Webber of the University of Pennsylvania and by Dr. John R. Clarke, MD, of the Medical College of Pennsylvania.

Acknowledgements

There are several people I would like to thank for their assistance. Without their help, this project would have been far more difficult, if not impossible.

I would like to thank Dr. Gordon Banks of the University of Pittsburgh and Drs. John R. Clarke and Laurie Loiacino of the Medical College of Pennsylvania for their help in resolving medical issues which arose during the development of TSARR. I would also like to thank Professor Norman Badler, Carey Phillips, and Jiahe Lu of the Graphics Laboratory of the University of Pennsylvania as well as Stephen Harrison of Digital Equipment Corporation for their patience and guidance with graphics problems.

I would like to extend my special thanks to Professor Bonnie Webber for her support and guidance throughout this project and also for giving me the opportunity to work on such an interesting and challenging thesis topic.

Contents

Abstract	ii
Acknowledgements	iii
1 Introduction	1
1.1 TraumAID	1
1.2 TSARR	2
2 The Anatomical Representation	3
2.1 Spatial Representations	3
2.1.1 Octree-based Models	3
2.1.2 Surface Models	4
2.2 Requirements for the Anatomical Representation	5
2.2.1 Body types	5
2.2.2 Internal Organ Variances	5
2.2.3 Knife wounds	6
2.2.4 Gunshot wounds	7
2.3 Suitability of Octree and Surface Model Representations	7
2.3.1 Octree Implementation	7
2.3.2 Surface Model Implementation	9
2.4 Initial Implementation	10

3	Implementation of TSARR	13
3.1	Simplifying Assumptions	13
3.1.1	Body Types	13
3.1.2	Gender	13
3.1.3	Movement	14
3.1.4	Wound Locations	14
3.1.5	Knife wounds	14
3.1.6	Gunshot wounds	14
3.1.7	Data Quality	14
3.2	Representations	15
3.2.1	Coordinate System	15
3.2.2	The “Poly” Data Structure	16
3.2.3	Organs	18
3.2.4	Stab Wounds	18
3.2.5	Gunshot Wounds	20
3.3	System Parameters	21
3.3.1	Knife Length	21
3.3.2	Number of Facets	21
3.3.3	Poly Proximities	22
3.3.4	Outer Shell	22
3.4	Wound Path Reasoning	22
3.4.1	Generating Possible Paths	22
3.4.2	Path Feasibility	22
3.4.3	Eliminating Extraneous Paths	23
3.4.4	User Feedback	24
3.5	System Input	24
3.5.1	Entry of Anatomical Structures	24
3.5.2	Entry of Medical Findings	25
3.6	System Output	26

3.6.1	System Conclusions	26
3.6.2	Soliciting Feedback from the User	27
4	Graphic System Compatibility	28
5	Further Work	29
5.1	Improvements to the Initial Implementation of TSARR	29
5.1.1	Body Types	29
5.1.2	Scaling	30
5.1.3	Blunt Trauma	30
5.1.4	Z Coordinate Generation	30
5.1.5	Movement	31
5.1.6	Probabilistic Reasoning	31
5.2	Facilitating Interaction with TSARR	31
5.2.1	User Interface	31
5.2.2	Display	32
5.3	Extensions to TraumAID to Support TSARR	32
5.3.1	Anatomical Reasoning	32
5.3.2	Probabilistic Reasoning	32
6	Summary and Conclusions	33
A	A Sample Run	34
B	Sample PSURF/Jack File	36
	Bibliography	37

Chapter 1

Introduction

In the United States, more years of human life are lost due to injury than any single disease [9]. It has been shown that thirty to forty percent of trauma deaths occurring within the first hours of injury could be prevented if expert care were administered to the patient in a timely manner [34].

When a patient arrives at an emergency ward with penetrating injuries, an attending physician must first determine the extent of those injuries and, in particular, determine which structures of the body have been damaged so that diagnosis and treatment can begin. The physician and attending medical staff must draw on a considerable body of anatomical knowledge in order to correctly diagnose the extent of injury. This anatomical knowledge is utilized along with beliefs about the direction of a single wound or, in the case of multiple wounds, which wounds might be connected (i.e., identifying entrance and exit wound pairs caused by multiple bullets) in order for the medical staff to determine which anatomical structures may have been damaged.

1.1 TraumAID

TraumAID [32, 33] is a knowledge-based system designed to diagnose and manage patients (after they have been resuscitated and stabilized) who arrive at a trauma center presenting penetrating trauma (i.e., gunshot and/or stab wounds). Because of the potential severity of

their wounds, these patients must receive immediate treatment, or they might not survive.

Currently, TraumAID has a rather superficial knowledge of the anatomy of the human body. The body is represented as being partitioned into relatively large sections, and TraumAID utilizes a whole/part hierarchy to represent the section of the body in which an injury has occurred.

In its present version, TraumAID performs only rudimentary forms of anatomical and geometric reasoning. This limits its usefulness since this knowledge has proven insufficient in some cases, leading the system to incorrect conclusions.

1.2 TSARR

TSARR (TraumaAID System for Anatomical Representation and Reasoning) is a system that is designed to address the need for greater depth in TraumAID's understanding of anatomical reasoning. TSARR provides a framework for representing a three-dimensional model of relevant parts of the body; utilizing this model, TSARR is able to calculate three-dimensional representations of paths of injury, generated from wound locations input to the system. Using these paths, the system hypothesizes which anatomical structures in the patient might have been injured due to their location along a possible path of an injury. In the future, TraumAID will be able to utilize this information to focus its attention more accurately on specific areas of the body that have sustained injury.

TSARR can also be used independently of TraumAID. In this stand-alone mode, physicians can enter clinical findings either to confirm or refute the suspicions that TSARR reports regarding whether or not various organs have sustained injury. Based on this additional information, the system will then revise its list of hypothesized wound paths and, therefore, its list of organs suspected of injury.

Chapter 2

The Anatomical Representation

2.1 Spatial Representations

Two different spatial models were considered in which to represent structures in TSARR: octree-based models and surface models.

2.1.1 Octree-based Models

Octrees are decomposition-based representations of three-dimensional objects [5, 14, 15, 19]. The major features of an octree representation is that it is a hierarchical data structure, objects are kept in a spatially pre-sorted order at all times, and it has spatial addressability. Octrees are created by recursively subdividing the *object space* into eight smaller cells of equal size until each cell is uniform. The octree, representing the location of cells in space, can be easily traversed and pre-sorted to facilitate operations such as the detection of intersections between objects [4].

Objects are represented as $2^n \times 2^n \times 2^n$ arrays of unit cubes. Each unit cube is labeled according to some salient feature and the object array is divided into *octants*. Each octant is further subdivided into smaller octants until all of the unit cubes within an octant have the same label [14].

The object array is represented by an octree whose nodes are either leaves or have

eight children. The root node represents the entire object array, and each of its children corresponds to octants. The children of octants are other, smaller octants until the leaves are reached (i.e., the labels of each of the unit cubes in an octant are the same).

2.1.2 Surface Models

An alternative to octree models is surface models. Two popular ways in which surface models can be represented are *polygon mesh models* and *parametric models*. Although polygon mesh models are computationally less complex than parametric models, they may also be less accurate. While the two surface models may be considered to be alternatives to one another, parametric models are not applicable in all situations.

The polygon mesh model is the most popular method for representing an object in computer graphics [31]. Objects are represented by a series of three-dimensional coordinates or *vertices* and a set of straight lines or *edges* that connect the vertices to one another to form *polygons*. The polygons are structured in such a way that they form a complete object, known as a *polyhedron*.

Depending upon the object being represented, the polygon mesh model representation can be exact (as in the case of a cube), or only an approximation (as in the case of a cylinder). The number of polygons used to approximate an object determines how precisely that object is represented. The degree of accuracy by which a polyhedron is represented determines the amount of storage, modelling cost, and computational complexity of the object.

A parametric representation of three-dimensional surfaces allows closed and multiple-valued functions to be easily defined [8]. Coordinates of points on a curved surface are defined by three equations (one each for the X, Y, and Z coordinates) called *parametric bicubic patches*. The boundaries of the patch are *parametric cubic curves* (equations in which X, Y, and Z are each represented as a third-order polynomial of some parameter t). Relatively few bicubic patches are required to represent a curved surface to a given accuracy when compared to the polygon mesh representation. However, the algorithms dealing with bicubics are more complex.

2.2 Requirements for the Anatomical Representation

2.2.1 Body types

While people vary greatly as to their shape and size, physicians in today's trauma centers are provided with only a paper, two-dimensional drawing of an idealized model of a male on which to denote the locations of injuries sustained by a patient. Therefore, the physician must not only extrapolate a three-dimensional object onto a two-dimensional view, but he or she also must map the injuries onto an idealized body. This injury mapping is accomplished by using body landmarks (e.g., the nipple line, the umbilicus) as guides. However, in the case of someone with a protruding and pendulous belly, for example, the umbilicus may not be in the same position as in the case of a thin person.

One problem with the current anatomical representation in TraumAID is that it is not sensitive to varying body types. While a stab wound in the abdominal area of a "thin" person might be very damaging, it might be less so in a grossly obese stabbing victim. Therefore, some provision must be made to cope with different body types.

Despite the variations in body types, it seems that physicians are capable of mapping an injury from a patient onto a standard body type. While this limited mapping is not an ideal situation, for the initial implementation of TSARR only the "standard" body type will be represented. Future extensions to TSARR could allow physicians to map injury locations onto a model with a body type and size corresponding to that of the patient.

2.2.2 Internal Organ Variances

The size and location of internal organs also vary from one person to another without respect to body type. Therefore, any anatomical representation must be able to incorporate such variations. In this project, the creation of "fuzzy" boundaries for anatomical structures are used to accommodate anatomical structures of different sizes. Thus, an organ is represented as having a *normal size* and an area surrounding it that is designed to account for individual variances. At some point in the future, a measure of uncertainty may be associated with various sizes of organs to represent the probability of injury to a specific structure, given the

location of an injury. The exact location of each organ also varies. For example, the location of the gall bladder inside of a person with a relatively large liver may differ from that of a person with an average-sized liver, since the placement of the gall bladder is relative to the size of the liver. Thus, the size of one organ may dictate the placement of another.

Not only are there great anatomical variations from person to person, but also in the same person over time as well. For example, the size of a person's bladder is dependent upon when that person last urinated. Therefore, any static representation for the size of internal organs can only be an approximation. However, it is expected that the representation discussed in Section 3.2 will be able to cope with individual variances by overlapping the positions of anatomical structures.

The example of the liver also has another interesting property. Differences in the position of the liver cannot extend in all directions. Since one side of the liver is situated against the rib cage, the liver cannot extend in that direction. However, it can extend in other directions. Therefore, in order to create a model of an anatomical structure that may vary in size, it is not sufficient to create a model whose size increases in all directions.

2.2.3 Knife wounds

A knife wound cannot be represented by a simple straight line representing the path of the knife. Since knives come in various widths and lengths, there is no "standard" knife wound. Furthermore, the assailant may twist or move a knife while it is in the victim. When representing the areas penetrated by a stab wound, one must also take into account the direction of the penetration. If the direction of the stab wound is known, the space in which the knife may have caused damage is lessened.

It is necessary to estimate a standard knife length so that some model of the path of such a wound can be created. Knife wounds with no known direction of penetration will have to be represented by a path encompassing all of the areas that could be reached by the knife blade, given a point of penetration. If the direction in which the knife penetrated the victim is known, then only those areas reachable by a knife thrust in that direction need be examined.

2.2.4 Gunshot wounds

The path of a bullet as it passes through a body cannot be represented as a straight line that has the width of the bullet. This is because the trajectory of a bullet may be altered by the density of the tissue through which it passes. Bullets passing through the body also have a percussive effect, resulting in the tearing of tissue on either side of the path of the bullet. Furthermore, a bullet may hit a very dense structure, such as a large bone, and ricochet. Thus, a simple straight line model of the path of a bullet is insufficient.

One of the tasks a physician faces in the assessment of a patient's injuries is to determine which wounds are connected (in the case of multiple gunshot wounds). Any attempt to do so requires that the examiner match entrance wounds with exit wounds. However, in many instances the medical staff cannot determine the orientation of the bullet that caused a particular injury. Furthermore, bullets do not always exit the body; thus, each entrance wound may not have a corresponding exit wound. In such cases, the locations of bullets trapped in the body can be revealed through x-rays, however.

In the case of a patient with multiple gunshot wounds, many path options could exist for each individual bullet. Some of these paths might be mutually exclusive, which would reduce the number of possible paths - but only after a suspected path has been proven or disproven by clinical tests. The number of possible paths also may be reduced in those instances where the direction of the injury is known. Such cases are relatively rare, however.

2.3 Suitability of Octree and Surface Model Representations

2.3.1 Octree Implementation

Advantages

There are several attractive features of octree-based representations. First of all, an octree-based model mandates the use of a single primitive shape: the cube. Any object may be represented using this shape to the precision of the smallest cube. The use of this single

spatial primitive involves only a single set of manipulation and analysis tools, thereby simplifying coding. Furthermore, techniques have been developed to efficiently index data associated with points in space.

From a practical standpoint, there are other issues that make octrees desirable. The fact that something analogous had been done with another part of the body [1], indicates that octree-based models have been proven sufficient to solve the problem. The fact that Dr. Banks, the developer of such a model, was also willing to answer questions and make available the editor that he used to construct his model provided additional advantages. Furthermore, the anatomical structures that can be found in a specific cube in the model can be specified purely symbolically, without having to indicate continuous shape.

Disadvantages

There are a number of technical problems associated with octree-based models. If it were desirable to change the selected level of granularity at a later date, a great deal of effort could be required. While increasing the size of each cube would be simple since the representation is designed to generalize, decreasing the size of each cube could potentially require that the entire mapping process be re-done if voxels with multiple values existed.

The number of voxels that would be required to represent those areas of the body with which TraumAID is concerned is daunting. This is due in part to the fact that the desired granularity of the representation is 0.5 centimeters. Thus, it is evident that the number of 0.5 cm cubes required to represent the thorax alone would be quite substantial.

While this project is not concerned with displaying the anatomical model, future extensions based upon this particular work would most likely desire such a capability. Unfortunately, the octree-based model might not be a good choice in this regard. The fact that the size and placement of anatomical structures might vary significantly from person to person necessitates that some “fuzziness” be built into the representation. The best way to include such variation seems to be by overlapping structures [2], thereby causing individual cubes to contain conjunctive or possibly disjunctive values. It is unclear how cubes containing multiple values (which represent the model at the lowest level of granularity) might be

displayed, although fuzziness must be incorporated into the model of the human body.

An additional drawback is the computations performed with the octree modeling technique are generally considered to be slow because the decomposition of space consumes so much space that tree traversals can be quite expensive. In an effort to overcome the perceived slowness of the technique, octree-based models require relatively large amounts of physical memory.

2.3.2 Surface Model Implementation

As discussed above, there are two common alternatives to representing objects using a surface model: using a polygon mesh model or using a parametric model. While the parametric form provides a very compact representation and is easy to define, it is only useful for those objects that may be defined by an equation.

A polygon mesh representation was selected because not all of the objects that need to be modeled in TSARR are definable by equation. While those shapes representing wound paths can be described parametrically, those shapes representing organs cannot. The use of polyhedrons to represent objects allows a single representation to be used for both anatomical structures and wound paths, as well as a single set of geometric procedures to be employed.

Advantages

The surface model is certainly well understood. Numerous articles may be found in the computer graphics literature as to how three-dimensional objects might be represented [13, 17, 22, 23]. Naturally, such three-dimensional computer graphics techniques lend themselves quite well to being displayed since that is their purpose.

The fact that the University of Pennsylvania has a well respected graphics laboratory was advantageous. Researchers in that lab could provide local support with computer graphics issues and problems.

Disadvantages

Internal organs must be represented as concave objects. Intersections of such objects are difficult to deal with since a single object may be intersected multiple times by another object. It is also a “geometrically hard” problem to cover all of the possible ways that concave objects can intersect [11].

2.4 Initial Implementation

An octree-based approach seemed the most promising for solving the problem.

Work began by mapping a grid over horizontal cross sections of a human body in an atlas of human anatomy [16]. Figure 1 shows a horizontal cross section of an abdomen taken from the atlas. Each square on the grid was marked to indicate which anatomical structures were at that location, and then this square was to be extrapolated to a cube. There was no apparent way of automating this process. Even if one had access to a digitizer, each of the voxels must still be labeled as to its content - a very labor-intensive task.

As previously stated, the level of detail desired by TraumAID’s medical expert was at a granularity of 0.5 cm per cubic voxel. Using an overlay placed over cross-sectional views of the body at that level of granularity (and scaled to the photograph), this meant approximately 6,300 cubes could have to be labeled per overlay (if it were not possible to label larger areas with labels having the same values). Each photographed section of the body was, in actuality, 2.5 cm thick. Some license had been planned to extrapolate on each voxel, using five identical voxels stacked on top of each other to represent the thickness of the slice. Thus, approximately 31,500 could be required to represent each photograph. Furthermore, a number of cross-sections were necessary to comprise an area, since each organ occupied several cross-sections. The thorax, for example, was divided into fifteen slices. Therefore, to represent the thorax at the desired level of granularity, over 472,500 voxels could be needed (in a worst case scenario).

The photographs of the cross-sectional anatomical slices used in the selected atlas of



Figure 1: An Example of a Photograph of a Horizontal Cross Section of the Abdomen

human anatomy presented another problem. While the photographs attempted to maintain a consistent scale, they maintained no consistent standard of reference. Therefore, maintaining a mapping of the body which relied heavily on positional information became quite difficult. The grids representing each photograph were label independently, with the intent of being able to align the grids using an anatomical structure that runs the length of the area of interest - the spinal column. Unfortunately, the spinal column has several curves in it, thereby making it difficult to use as a point of reference.

Chapter 3

Implementation of TSARR

3.1 Simplifying Assumptions

There are several simplifying assumptions that must be made in order to solve the problem of calculating possible penetration paths through the human body. Since the physical dimensions of most of the objects (i.e., organs, wound paths, and the instruments that cause injury) that TSARR is designed to address can vary from patient to patient, some assumptions will be made as to their size. The fact that people can move, twist, and bend will cause other assumptions to be made.

3.1.1 Body Types

The initial implementation of TSARR will only concern itself with a single, “standard” body type.

3.1.2 Gender

The sex of the patient will not only determine which organs a patient has but also their placement. For example, the positions of the livers of two patients of the same body type and same height may differ due to physical gender differences. This project will not deal with these differences.

3.1.3 Movement

Movement further complicates any representation of the human body. A victim - in attempting to protect himself or herself from an assault - may twist or bend to deflect a projectile or knife with another part of his or her body. Therefore, the path of an injury might be dependent upon the position of the victim during the time of attack. While such issues will not be dealt with in this work, the capability of supporting such extensions in the future is considered in Section 5.1.5.

3.1.4 Wound Locations

It is assumed that only one actual wound path can emanate from any given wound location, although multiple hypothetical wound paths may be generated from a single wound location. Thus, the existence of multiple gunshot or knife wounds cannot be specified by a single wound location.

3.1.5 Knife wounds

Knives will be assumed to be of a "standard" size. This standard size may be set with a parameter. The size specified for the average knife should err on the side of being larger than the perceived standard to account for indentations made when a knife is thrust into the abdomen.

3.1.6 Gunshot wounds

A bullet can hit a bone and ricochet in an unpredictable direction; such cases will not be addressed in this work. Nor will bullets that fragment inside of the body.

3.1.7 Data Quality

It is assumed that the points comprising each face or layer of a polyhedron will be co-planar.

3.2 Representations

TSARR organ and wound path models are represented using a polygon mesh model. This representation was chosen because of its simplicity and the fact that any object can be approximated using this technique. Objects will be defined in terms of their name, location and the objects by which they are intersected.

In order to determine which organs a wound might have damaged, TSARR must create a representation of the path of injury. All possible paths of the bullet or knife which inflicted a particular wound must be calculated, along with an estimation of the shape of the path.

3.2.1 Coordinate System

TSARR employs a cartesian coordinate system to define the points that comprise structures. The X axis will be the horizontal axis, while the Y axis will be the vertical axis. The Z axis will indicate depth. If one is facing the front of the body model, the +Z axis will extend *towards* the viewer.

Z Coordinate Generation

If no Z coordinate is specified when entering the coordinates of a wound, TSARR will supply the Z coordinate. This is accomplished by determining where the user-supplied X and Y coordinates lie on the body, and returning the maximal Z value on that face. The Z coordinate of the location of a bullet lodged in the body must be entered by the attending medical staff, however.

UV Positions

In order to facilitate geometric tests, three-dimensional polygons and points are sometimes collapsed down into a two-dimensional space. To do so, a pair of numbers will be used to determine which of the X, Y, and Z coordinates are the *non-dominant* coordinates. The *dominant* coordinate is the coordinate in the plane equation of the largest magnitude. For example, if the plane equation gives a result of [4 -6 2], where 4, -6, and 2 represent the

X, Y, and Z coordinates, respectively, then the dominant coordinate is the Y coordinate. Thus, the *uv positions* will be (0 2). By using the uv positions, a polygon (i.e., a face) can be projected onto a plane to facilitate testing, without having to perform expensive matrix operations [10].

3.2.2 The “Poly” Data Structure

One Lisp data structure will be used to represent both organs and wound paths. It will be called a “poly.” The structure of a poly is given as follows:

```
(defstruct poly          ;;; structure to rep wound polys and organs
  name                  ;;; poly name
  represents             ;;; organ or type of wound
  points                 ;;; coordinates of each poly vertex
  faces                  ;;; how the vertices are connected
  intersections          ;;; polygons intersected by
  existential_status)    ;;; status of polyhedron
```

Name

This is the name given to the “poly” (an abbreviation for “polyhedron”) structure. The list of polys representing anatomical structures will have names assigned to them by the person inputting the data for those structures and will be established before TSARR is executed. Polys representing wound paths will have names assigned to them. The names will start with one of the following symbols:

- “dstab-” (a stab wound about which the direction is known),
- “ustab-” (a stab wound about which the direction is unknown), and
- “gunshot-” (paths representing gunshot wounds).

A computer-generated number is used as a suffix for each of these types. This number will be unique for each poly represented in TSARR and will be used to distinguish multiple wounds of the same type.

Represents

This field will be used to indicate what type of structure a particular poly represents. The possibilities are: "organ," "gunshot," or "stab."

Points

This field will specify the list of vertices that comprise the polyhedron. This will be a list of lists, with each sublist containing an X, Y, and Z coordinate.

Faces

A list of lists will be used to indicate which points comprise which faces of the polyhedron. Each face will consist of a list of indices to those points of the poly that comprise that particular face. It will be assumed that the last point specified will always be connected to the first point in that list of face points.

Intersections

A list will be maintained of all objects that a particular polyhedron is believed to have intersected. Thus, a poly representing an organ may have a list of those wound paths that intersect it, whereas a wound path would maintain a list of those organs it intersects.

Existential Status

This status is used to indicate the level of belief that one has in the existence of that poly. The possible values for this status are:

- unexamined
- suspected
- ruled_out
- confirmed

Only organs can have an *unexamined* status. This is the default for organs and indicates that the organ has not been considered as an organ that might have been involved in an injury.

A *suspected* existential status indicates that an organ or wound poly might have been intersected.

Ruled_out indicates that a suspected poly has been eliminated from consideration by the system. For polys representing organs, this means that the attending medical staff has ascertained that the organ has not been injured. In the case of polys representing wound paths, a ruled out status will indicate that this possible path has been disproven.

An existential status of *confirmed* indicates that the suspicion of involvement of an organ or wound has been validated by a physician.

3.2.3 Organs

Organs will be represented as a series of connected coordinates which comprise the three-dimensional model of the structure. The names of individual organs will be assigned when the list of organ polys is constructed, in advance of the execution of TSARR. As previously stated, the initial existential status of an organ is "nil."

3.2.4 Stab Wounds

Undirected

More often than not, the direction of a knife thrust responsible for a particular stab wound is not known [7]. Therefore, all organs surrounding the area of the stab wound - and within reach of the knife blade - must be suspected of having sustained injury.

For these purposes, a parameter can be specified to the system as to the length of the *standard* knife blade, and to the precision of the model of the knife wound. The model of a stab wound is then represented as a hemisphere-shaped structure, having a diameter equal to twice the length of the knife length parameter, and a height equal to the knife length. This hemispheric shape is designed to encompass all of the points that the knife blade could

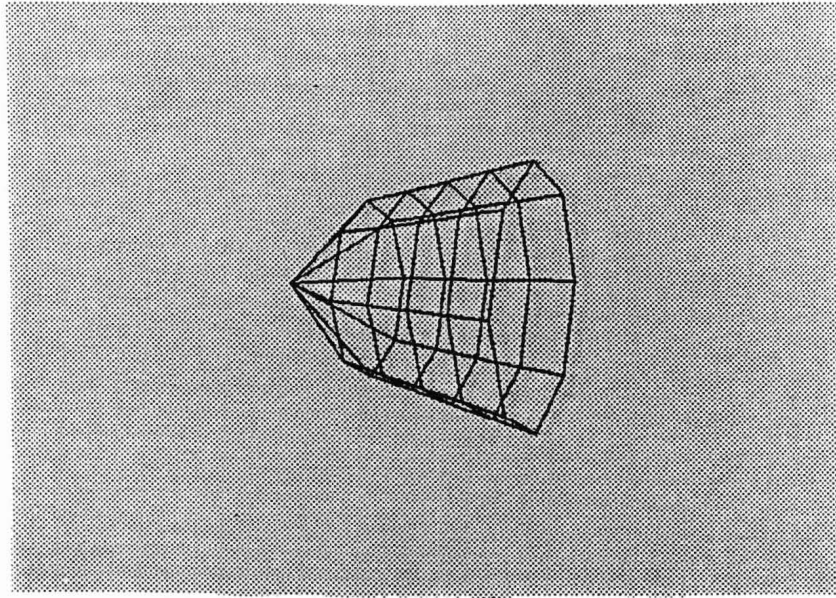


Figure 2: Polyhedron Representing an Undirected Knife Wound Path

potentially reach (under "normal" circumstances).

Directed

When the direction of a stab wound is known, the search space of points which may have been reached by the knife blade can be pared accordingly. Furthermore, the angle of the wound helps to limit the search space as well.

In those instances where the direction of the wound is known, the attending physicians may specify the direction of injury using the following scheme:

```

123
456
789
0 = direction unknown

```

Thus, if the knife is known to have travelled straight into the victim, the attending medical staff would enter "5" as the direction of injury. If the knife thrust travelled straight down from the point of entry, an "8" would be entered. A "1" would be entered if the wound extended to the upper left of the point of entry, and so on.

Directed stab wounds will be represented by a conic shape. The diameter and the height

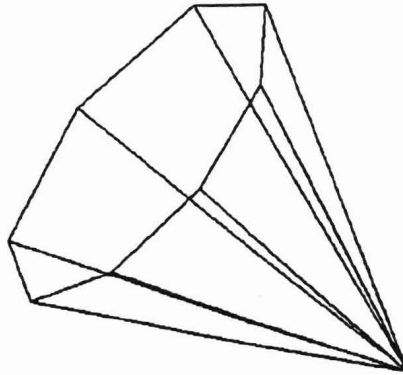


Figure 3: Polyhedron Representing an Directed Knife Wound Path

of the cone will be equal to the parameter that specifies the length of the knife, so that the solid angle of the cone will equal sixty degrees. The angle and center point of the bottom of the cone will be determined by the direction of the wound.

3.2.5 Gunshot Wounds

Gunshot wounds will be specified by entering the X, Y, and Z coordinates of the wound, as well as the direction and type of wound. The wound direction may be specified in the same manner as knife wounds. The heuristic used to calculate the width of the wound path at its widest point is the distance between the two end points divided by eight.

Gunshot Wound Classifications

The attending medical staff can classify gunshot wounds for the system as being of one of three types: entrance wounds (i.e., those wounds known to be caused by a bullet penetrating the body), exit wounds (i.e., those wounds known to be caused by bullets exiting the body), and unspecified wounds (i.e., holes in the body caused by a projectile, although whether the bullet was entering or exiting the body is unknown). The medical staff may also report to the system the location of any projectiles still lodged in the body as shown by x-ray or

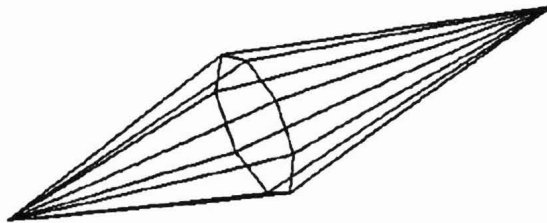


Figure 4: Polyhedron Representing a Gunshot Wound Path

other diagnostic techniques.

3.3 System Parameters

System parameters exist to enable TSARR to be tailored to different levels of precision, as well as to specify a “standard” size of a knife blade.

3.3.1 Knife Length

The `*knife_length*` parameter allows the specification of what is considered to the length of a typical knife blade, relative to the coordinates of the body model. This number is used in calculating the dimensions of a stab wound path.

3.3.2 Number of Facets

TSARR uses the `*num_facets*` parameter when calculating the number of points that are used to represent a circle (used in the creation of polyhedrons representing wounds). For example, if `*num_facets* = 8`, then the circle will actually be an octagon. If `*num_facets* = 360`, then a very precise circle will be created.

The larger the number of facets that is specified, the more processing the system will

have to perform for each calculation. The volume of the wound polyhedron will vary directly with the number of facets specified, however.

3.3.3 Poly Proximities

The **poly_proximity** parameter is a number, relative to the coordinates of the body model, used to eliminate the examination of organ polys that are not in the proximity of a wound path. For example, if a patient is shot in the side of the head, there is no need to examine the patient's stomach for injury. The parameter specifies how near to a wound path an organ must be in order to be examined for injury.

3.3.4 Outer Shell

A name is specified to indicate to TSARR which of the polys representing the body is the **outer_shell**. The outer shell poly is used to calculate the Z coordinates of wound locations, if necessary.

3.4 Wound Path Reasoning

3.4.1 Generating Possible Paths

Polyhedrons representing bullets paths are generated as follows: entrance wounds are connected with exit wounds, unspecified wounds, and bullets lodged in the body; unspecified wounds are connected with other unspecified wounds, entrance wounds, exit wounds, and bullets lodged in the body.

3.4.2 Path Feasibility

Although several different paths might exist which connect a collection of penetrating wounds, not all of the paths may be possible. Paths may be eliminated from consideration if they do not follow the laws of geometry or if they contradict common sense.

Before the system creates a path to connect two wounds, it first attempts to determine

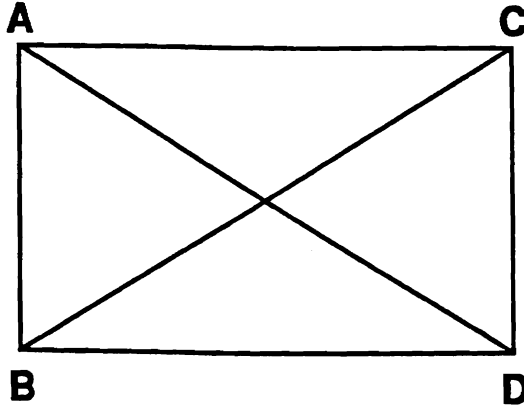


Figure 5: Paths connecting 4 gunshot wounds

whether that path is feasible. In the current implementation, this check is rudimentary and only affects those wounds with a specified direction of injury.

For each wound with a specified direction of penetration, the system will not establish a path that is geometrically unlikely. For example, if an injury is known to have travelled to the left of the wound, then all wounds/bullets to the right of the wound site will not be considered as an endpoint for a path from the wound. While this test is quite simplistic at present, it may easily be enhanced to filter out more complex situations.

3.4.3 Eliminating Extraneous Paths

Given any n bullet holes or bullet locations, there are $n(n - 1)/2$ possible paths that may be created to connect those entities. However, under normal circumstances, only $n/2$ of the possible paths will exist. Therefore, extraneous paths must be eliminated so that the anatomical structures which they intersect need not be included in the list of possibly injured structures.

Consider Figure 5 above. If it is established that the path between points A and C is valid (i.e., it has been proven that an organ intersecting that path has been injured), then other paths which share the same endpoints may be eliminated.

Conversely, if no evidence exists that the path between points A and C is correct (i.e.,

all of the organs that are intersected by that path have been found to be uninjured), then that path may be eliminated. Thus, all of the organs that were suspected of injury but that did not intersect with any other path of injury may be removed from suspicion. No other paths may be eliminated in this case, however.

3.4.4 User Feedback

The attending medical staff has the ability to confirm, refute, or state that no new evidence exists about an organ that TSARR suspects might have been injured. The program may then be re-executed, utilizing this new information.

If the physician states that there are negative findings regarding damage to a particular organ, then that organ will be eliminated from further consideration by the system on subsequent runs. Furthermore, any organs that were not suspected of injury in a preceding run of TSARR will not be examined by the system in subsequent runs since wound paths will not change. Thus, TSARR will further discriminate among organs about which there have been medical findings each time the system is run.

If injury to a particular organ is confirmed and that organ is only intersected by a single wound path, then alternative wound paths may be eliminated. The wound path that intersected the confirmed organ will be marked as confirmed, and all alternative wound paths to that path will be ruled out.

3.5 System Input

3.5.1 Entry of Anatomical Structures

The entry of anatomical structures may be input into the system through the use of a digitizer. Eventually, three-dimensional structures may be built up from a collection of structures digitized from two-dimensional photographs.

All of the photographs of cross-sectional anatomy representing a particular structure of the body must be used to build up a three-dimensional representation of that structure. By using a digitizer equipped with a puck, the user can trace the outline of each anatomical

structure of interest on each cross-sectional photograph, selecting some number of points sufficient to outline a structure.

After a structure has been "outlined" on a cross-section, those points comprising the two-dimensional outline of that structure will form a polygon. The sum of all polygons representing a particular organ may then be layered on top of one another using a process called *tiling* so that they form a single polyhedron. The resulting polyhedron will be a three-dimensional representation of the structure.

In using the digitizer, it is important to maintain a consistent way of tracing each structure (i.e., clockwise or counter-clockwise). It is also desirable, at least initially, to use a small number of points to comprise each polygon. More precise representations (i.e., those with a greater number of points) may always be entered at a later date.

3.5.2 Entry of Medical Findings

Although a graphical interface for entering medical findings about a patient is envisioned, it is beyond the scope of this project. Thus in the initial implementation, the user is required to input the X, Y, and Z coordinates of each gunshot wound, knife wound, as well as the coordinates for the locations of any bullets lodged inside of the victim.

The following is an example of how a user could interact with the system in its present implementation. (It is envisioned that TSARR will only directly interact with TraumAID in future implementations, however.)

Enter gunshot wound x coordinate (no input to terminate):

Enter bullet x coordinate (no input to terminate):

Enter stab wound x coordinate (no input to terminate): 50

Enter stab wound y coordinate: 50

Enter stab wound z coordinate (no input for computer-generated coord): 100

Enter wound direction [0 = unknown (default)]: 3

Enter stab wound x coordinate (no input to terminate): 55
Enter stab wound y coordinate: 55
Enter stab wound z coordinate (no input for computer-generated coord):
 z coordinate = 100.0
Enter wound direction [0 = unknown (default)]:

Enter stab wound x coordinate (no input to terminate):

The interaction of the TraumAID reasoner with TSARR is simulated by requiring user interaction as follows (note: only the anatomical reasoner would actually assert a conclusion with regards to the existential status of an organ):

Enter findings for each organ suspected [0 = none, 1 = negative, 2= positive]:
 STOMACH? 2
 SPLEEN? 1
 PANCREAS? 2
 LEFT_KIDNEY? 0

Re-run program on new information? (Y or N): n

The following organs are CONFIRMED to have been injured:
 STOMACH
 PANCREAS

The following organs are SUSPECTED to have been injured:
 LEFT_KIDNEY

3.6 System Output

3.6.1 System Conclusions

After performing its reasoning functions, TSARR will return a list of the names of organs that have been confirmed to have been injured as well as a list of those organs that *may* have been injured.

For each organ, the system will maintain a list of those possible injury paths that may

have intersected it. A record of each organ intersected by the structure that represents an injury path will be maintained as well.

3.6.2 Soliciting Feedback from the User

After the system has reached its conclusions, it prompts the user (or the TraumAID anatomical reasoner in future versions) to either confirm or refute its findings. For each organ that the system suspects might have been injured, the user is asked if there is any conclusive evidence to either support or refute that that organ has, in fact, been affected. If there is no clinical evidence to either confirm or deny that a particular organ has been damaged, the system continues to suspect that organ. The same is true for organs that have been confirmed to have been injured. When evidence exists to disprove the suspicion of injury to a particular organ, that organ will be removed from suspicion.

A refutation also has greater consequences. If an organ lies directly in a possible injury path, it must have been intersected if the path is to be confirmed. Therefore, if such an organ is proven to be unscathed, then the wound path can be concluded to be incorrect. Therefore, all organs that are suspected of injury by the errant path and the errant path alone may be removed from suspicion.

Chapter 4

Graphic System Compatibility

The anatomical reasoning system is designed to be compatible with the Jack graphics system developed at the University of Pennsylvania [21, 20]. Jack is a program that is designed to display and manipulate articulated geometric figures. It uses files that are in “psurf” format. In this representation, the last point specified for a face of a polyhedron will be connected to the first point of that face. An example of a the format of a psurf file is illustrated in Appendix B.

TSARR functions exist so that figures represented in the Jack representation may be read into the TSARR system. Other functions are provided to output any TSARR anatomical or path representation to Jack format.

Chapter 5

Further Work

As this system is designed to be an initial prototype to allow TraumAID to reason about anatomy, it is far from having the complete functionality that the TraumAID researchers desire. There are many potential extensions to this body of work, classified as being of one of the following types: improvements to the initial implementation of TSARR, programs to facilitate interaction with TSARR, and extensions in TraumAID to support TSARR.

5.1 Improvements to the Initial Implementation of TSARR

There are several ways in which the initial implementation of TSARR may be improved upon.

5.1.1 Body Types

The ability to deal with various body types is clearly desirable. The system must have the ability to reason about all people, no matter what their size. Therefore, an anatomist should be consulted to determine how to model the percentage of fat on people, as well as to model how the fat may be distributed on the human body.

5.1.2 Scaling

To be effective, TraumAID must “understand” the proportions of a patient. For example, a six-inch knife might reach more anatomic structures in a 4’10” person than in a 6’10” person. Since the surface model lends itself quite well to being scaled, some mechanism should be devised to allow a physician to observe medical findings on a body model that is scaled to the size of the patient.

The TraumAID system should be able to deal with the scaled model, since the reasoning relies on the size and placement of the organs. If the organs are scaled to the proper dimensions before any reasoning takes place, TraumAID should require few modifications, if any.

5.1.3 Blunt Trauma

While the current version of this system deals only with penetrating trauma, cases in which blunt trauma has occurred is an envisioned extension to TraumAID and therefore, must be an extension of the anatomical reasoner as well.

Some method of describing the shape and dimensions of a blunt injury must be formulated. This shape could then be represented in the same way as gunshot and stab wounds, and could be reasoned about in a similar manner as well.

5.1.4 Z Coordinate Generation

A more complex procedure than described in Section 3.2.1 could be used to generate Z coordinates in instances where they are not supplied by the user. This method would involve the use of *barycentric coordinates* (coordinates used in computing the point at which a ray intersects a triangle [27]) and would provide TSARR with more precise Z coordinates based upon the given X and Y coordinates and the faces of the body model.

5.1.5 Movement

A quite challenging aspect to the nature of the problem of determining wound paths is that people generally do not hold still for their attacker. Victims may twist or bend at the time of attack. They may also attempt to protect themselves, for example, by raising their arms to deflect or absorb a blow. Therefore, a complete model of how an injury might be sustained cannot be achieved without modeling the various ways in which human beings can bend and move.

5.1.6 Probabilistic Reasoning

Future versions of TSARR should possess the ability to assign probabilities to the likelihood that a particular organ was injured. For example, an organ whose center has been intersected by a path of injury is more likely to have been injured than one whose periphery has been intersected.

5.2 Facilitating Interaction with TSARR

The implementation of programs designed to interact with TSARR would make the system more user-friendly as well as more useful.

5.2.1 User Interface

Rather than having the attending medical staff enter discrete coordinates, the user should be able to enter the wound locations by mouse clicking on the locations on an image of a human body on a computer screen. The computer display should include six views of the body (front, rear, left side, right side, top, and bottom) on which wound locations could be entered. Menus could be provided to allow the user to specify the instrument of injury (i.e., knife or gun), the type of wound (i.e., entrance, exit, or unknown), and its direction (i.e., a value between 0 and 9 or something analogous).

5.2.2 Display

While the current system is not concerned with the display of anatomical structures, future versions of TraumAID might want to possess such a capability. Such an extension should be relatively trivial since the surface model is designed for display purposes. The fact that the anatomical structures are represented in such a way that they are compatible with a graphics system available at the University of Pennsylvania should also make this a simple endeavor.

5.3 Extensions to TraumAID to Support TSARR

The current implementation of TraumAID is not designed to interact with a program such as TSARR because TraumAID assumes that wound location information is entered symbolically (e.g., "Chest_Below_Diaphragm_Level"). Therefore, some enhancements must be made to TraumAID before it can utilize the information provided by TSARR.

5.3.1 Anatomical Reasoning

The current version of TraumAid does not reason about a specific list of organs suspected of injury. Its rules are designed to reason about areas of involvement rather than specific organs of involvement. Therefore, to handle more specific information, a number of TraumAID's rules would have to be re-written.

5.3.2 Probabilistic Reasoning

The TraumAID system would have to be modified to reason with probabilistic information if a similar extension to TSARR is to be of use. While this extension has much merit, a great deal of effort would be required to modify both TraumAID and its anatomical reasoning system to reason probabilistically.

Chapter 6

Summary and Conclusions

Because many hospitals in the United States offer little or no expert trauma care, a demonstrated need exists for assistance to medical personnel in providing expert care to victims of life-threatening injuries.

TraumAID was designed to provide physicians with decision support for the initial definitive management of victims of penetrating trauma. It does have some limitations, however. To overcome one such limitation, TSARR has been designed to provide TraumAID with the ability to represent human anatomy and to reason about injuries in terms of the locations of wounds and anatomical structures.

Knowledge of the location of organs in the human body is essential to diagnose the extent of injury sustained by a victim of penetrating trauma. Without this knowledge of spatial relationships, the capabilities of TraumAid are limited. TSARR, by providing a three-dimensional representation of human anatomy, gives TraumAID the ability to more accurately assess which organs have been damaged and how severely.

Appendix A

A Sample Run

```
> (tsarr poly_list)
Enter gunshot wound x coordinate (no input to terminate): 44
Enter gunshot wound y coordinate: 44
Enter gunshot wound z coordinate (no input for computer-generated coord): 101
Enter type of wound if known [default = nonspecific].
(0 = nonspecific, 1 = entrance, 2 = exit):
Enter wound direction [0 = unknown (default)]:

Enter gunshot wound x coordinate (no input to terminate): 47
Enter gunshot wound y coordinate: 51
Enter gunshot wound z coordinate (no input for computer-generated coord): 0
Enter type of wound if known [default = nonspecific].
(0 = nonspecific, 1 = entrance, 2 = exit):
Enter wound direction [0 = unknown (default)]:

Enter gunshot wound x coordinate (no input to terminate): 72
Enter gunshot wound y coordinate: 77
Enter gunshot wound z coordinate (no input for computer-generated coord): 100
Enter type of wound if known [default = nonspecific].
(0 = nonspecific, 1 = entrance, 2 = exit):
Enter wound direction [0 = unknown (default)]:

Enter gunshot wound x coordinate (no input to terminate): 66
Enter gunshot wound y coordinate: 58
Enter gunshot wound z coordinate (no input for computer-generated coord): 0
Enter type of wound if known [default = nonspecific].
```

(0 = nonspecific, 1 = entrance, 2 = exit):
Enter wound direction [0 = unknown (default)]:

Enter gunshot wound x coordinate (no input to terminate):

Enter bullet x coordinate (no input to terminate):

Enter stab wound x coordinate (no input to terminate):

Enter findings for each organ suspected [0 = none, 1 = negative, 2 = positive]:
DIAPHRAGM? 2
ESOPHAGUS? 2
HEART? 1
LEFT_LUNG? 0
Re-run program on new information? (Y or N): y

Enter findings for each organ suspected [0 = none, 1 = negative, 2 = positive]:
LEFT_LUNG? 2
Re-run program on new information? (Y or N): n

The following organs are CONFIRMED to have been injured:
ESOPHAGUS
DIAPHRAGM
LEFT_LUNG

The following organs are SUSPECTED to have been injured:
NIL
>

Appendix B

Sample PSURF/Jack File

The following is a file in psurf format, readable by Jack, that will display a cube, 10 units on a side, starting at the point (70 70 70):

```
70.0  70.0  70.0
70.0  80.0  70.0
80.0  80.0  70.0
80.0  70.0  70.0
70.0  70.0  80.0
70.0  80.0  80.0
80.0  80.0  80.0
80.0  70.0  80.0
;;
1 2 3 4 ;
1 4 8 5 ;
3 7 8 4 ;
1 5 6 2 ;
5 8 7 6 ;
2 6 7 3 ;
;;
```

Bibliography

- [1] G. Banks, J. K. Vries, and S. McLinden. Radiologic automated diagnosis (RAD). *Computer Methods and Programs in Biomedicine*, 25:157–168, 1987.
- [2] G. E. Banks. University of Pittsburgh. Personal communication, June 1990.
- [3] I. Carlbom, I. Chakravarty, and D. Vanderschel. A hierarchical data structure for representing the spatial decomposition of 3d objects. *IEEE Computer Graphics and Applications*, 5(4):24–31, 1985.
- [4] H. H. Chen and T. S. Huang. Octrees: Construction, representation, and manipulation. In *Proceedings of SPIE - The International Society for Optical Engineering Vol. 579*, pages 448–458, 1985.
- [5] H. H. Chen and T. S. Huang. A survey of construction and manipulation of octrees. *Computer Vision, Graphics, and Image Processing*, 43(3):409–431, 1988.
- [6] C. Chien and J. Aggarwal. Volume/surface octrees for the representation of three-dimensional objects. *Computer Vision, Graphics, and Image Processing*, 36:100–113, 1986.
- [7] J. R. Clarke. Medical College of Pennsylvania. Personal communication, November 1990.
- [8] J. D. Foley and A. van Dam. *Fundamentals of Interactive Computer Graphics*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1982.

- [9] R. I. Glass. New prospects for epidemiologic investigations. *Science*, 234:951–959, 1986.
- [10] A. Glassner, editor. *Introduction to Ray Tracing*, pages 50–59. Academic Press, New York, 1989.
- [11] S. Harrison. Digital Equipment Corporation. Personal communication, November 1990.
- [12] B. Hayes-Roth, R. Washington, R. Hewett, M. Hewett, and A. Seiver. Intelligent real-time monitoring and control. Technical Report KSL 89-05, Knowledge Systems Laboratory, Department of Computer Science, Stanford University, January 1989.
- [13] D. Hearn and M. P. Baker. *Computer Graphics*. Prentice-Hall International Editions, 1986.
- [14] C. L. Jackins and S. L. Tanimoto. Oct-trees and their use in presenting three-dimensional objects. *Computer Graphics and Image Processing*, 14:249–270, 1980.
- [15] P. Jannin, B. Gibaud, C. Barillot, et al. Octree encoding with volume and surface information in medical imaging. In *IEEE Ninth Annual Conference of the Engineering in Medicine and Biology Society*, pages 1666–1667, 1987.
- [16] J. H. Koritke and H. Sick. *Atlas of Sectional Human Anatomy, Frontal, Sagittal, and Horizontal Planes*. Urban and Schwarzenberg, Munich; Baltimore, 1983.
- [17] M. Martti. *An Introduction to Solid Modeling*, chapter 13. Computer Science Press, Rockville, Maryland, 1988.
- [18] P. Mazumder. A new strategy for octtree representation of three-dimensional objects. In *Proceedings of CVPR '88: The Computer Society Conference on Computer Vision and Pattern Recognition*, pages 270–275, Ann Arbor, Michigan, 1988.
- [19] D. Meagher. Geometric modeling using octree encoding. *Computer Graphics and Image Processing*, 19:129–147, 1982.
- [20] C. B. Phillips. Jack user's guide. Technical report, Department of Computer and Information Science, University of Pennsylvania, August 1990. Draft.

- [21] C. B. Phillips and N. I. Badler. Jack: A toolkit for manipulating articulated figures. In *Proceedings of ACM/SIGGRAPH Symposium on User Interface Software*, Banff, Alberta, Canada, 1988.
- [22] S. P. Raya and J. K. Udupa. Shape-based interpolation of multidimensional objects. *IEEE Transactions on Medical Imaging*, 9:32–42, March 1990.
- [23] D. F. Rogers. *Procedural Elements for Computer Graphics*. McGraw-Hill, Inc., 1985.
- [24] H. Samet. Neighbor finding in images represented by octrees. *Computer Vision, Graphics, and Image Processing*, 46(3):367–386, 1989.
- [25] H. Samet and R. E. Webber. Hierarchical data structures and algorithms for computer graphics; part i: Fundamentals. *IEEE Computer Graphics Applications*, 8(3):48–68, 1988.
- [26] H. Samet and R. E. Webber. Hierarchical data structures and algorithms for computer graphics; part ii: Applications. *IEEE Computer Graphics Applications*, 8(4):59–75, 1988.
- [27] J. M. Snyder and A. H. Barr. Ray tracing complex models containing surface tessellations. *Computer Graphics*, 21(4):119–128, July 1987.
- [28] G. B. Thomas and R. L. Finney. *Calculus and Analytic Geometry*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1980.
- [29] J. K. Udupa. Computer aspects of 3d imaging in medicine: A tutorial. Technical Report MIPG153, Medical Image Processing Group, Department of Radiology, University of Pennsylvania, August 1989.
- [30] J. K. Udupa and V. G. Ajjanagadde. Boundary and object labelling in three-dimensional objects. Technical Report MIPG124, Medical Image Processing Group, Department of Radiology, University of Pennsylvania, December 1987.

- [31] A. Watt. *Fundamentals of Three-Dimensional Computer Graphics*. Addison-Wesley Publishers Ltd., 1989.
- [32] B. L. Webber, J. R. Clarke, M. Niv, R. Rymon, and M. M. Ibanez. TraumAID: Reasoning and planning in the initial definitive management of multiple injuries. Technical Report MS-CIS-90-50, Department of Computer and Information Science, University of Pennsylvania, August 1990.
- [33] B. L. Webber, J. R. Clarke, R. Rymon, and M. Niv. TraumAID: AI support for the management of multiple trauma. Technical Report MS-CIS-90-09, Department of Computer and Information Science, University of Pennsylvania, February 1990.
- [34] J. G. West, D. D. Trunkey, and R. C. Lim. Systems of trauma care: A study of two counties. *Archives of Surgery*, 114:455–460, 1979.