Go Digital, Go Fuzzy

Jayaram K. Udupa
Medical Image Processing Group
Department of Radiology University of Pennsylvania
423 Guardian Drive 4th Floor Blockley Hall
Philadelphia, PA 19104-6021
jay@mipg.upenn.edu

ABSTRACT

In many application areas of imaging sciences, object information captured in multi-dimensional images needs to be extracted, visualized, manipulated, and analyzed. These four groups of operations have been (and are being) intensively investigated, developed, and applied in a variety of applications. In this paper, after giving a brief overview of the four groups of operations, we put forth two main arguments: (1) Computers are digital, and most image acquisition and communication efforts at present are toward digital approaches. In the same vein, there are considerable advantages to taking an inherently digital approach to the above four groups of operations rather than using concepts based on continuous approximations. (2) Considering the fact that images are inherently fuzzy, to handle uncertainties and heterogeneity of object properties realistically, approaches based on fuzzy sets should be taken to the above four groups of operations. We give two examples in support of these arguments.

1. INTRODUCTION

In imaging sciences, particularly medical, there are many sources of multidimensional images [1].

2D: digital/digitized radiographic images, tomographic slices.

3D: a time sequence of 2D images of a dynamic object, a stack of slice images of a static object.

4D: a time sequence of 2D images of a dynamic object for a range of imaging parametric values (e.g., MR spectroscopic images of a heart), a time sequence of 3D images of a dynamic object.

5D: a time sequence of 3D images of a dynamic object for a range of parametric values.

Three- and higher-dimensional images may also be generated by a computational step. For example, a 5D binary image is produced if we apply a "lifting" operation [2] to a 4D gray-level image to represent the 4D image intensity distribution as a surface in a 5D space. Columns of volume elements in this higher dimensional space represent the height of the surface.

For ease of further reference, we will refer to a multidimensional image simply as a *scene* and represent it by a pair $C = (C, f)$, where C, the *scene domain*, is a multidimensional rectangular array of spatial elements (*spels* for short), and *f*, the *scene intensity,* is a function that assigns to every spel a vector whose component elements are from a set of integers.

There is usually an *object* of study for which the scene is generated. This may be a physical object such as an organ/tissue component, a tumor/fracture/lesion/an abnormality, a prosthetic device, or a phantom. It may also be a conceptual object such as an isodose surface in a radiation treatment plan, an activity region highlighted by PET or functional MRI, or a mathematical phantom. The objects may be rigid, deformable, static, or dynamic. For example, the

bones at a moving joint represent a dynamic, rigid object assembly. The heart muscle and the blood pool in the various chambers constitute a dynamic, deformable object assembly. We will refer to any operation, which, for a given set of multimodality scenes of an object of study, produces information about the object of study, as *3D imaging*. The information may be qualitative - for example, a 3D rendition of a static object, an animation sequence of a dynamic object or an animation sequence of a static object corresponding to different viewpoints. It may be quantitative - for example, the volume of a static object, and the rate of change of volume of a dynamic, deformable object, the extent of motion of a rigid dynamic object. The purpose of 3D imaging is, therefore, given multiple scenes as input, to output qualitative/quantitative information about the object of study.

In this exposition, we will give a quick overview of the various 3D imaging operations that are commonly used and identify the challenges currently faced. We argue with strong evidence that, since the scenes are inherently digital, we should take entirely digital approaches to realize all 3D imaging operations. We also suggest that, since object information in scenes is always fuzzy, we should take approaches that are based on fuzzy set concepts. The need to fulfill these strategies opens numerous problems that are inherently digital in both hard and fuzzy settings. The reader is referred to [2] for a detailed account of 3D imaging operations and their medical applications.

Although the examples considered in this exposition are all medical, the principles and the arguments are also applicable to other application areas.

2. AN OVERVIEW OF 3D IMAGING OPERATIONS

3D imaging operations may be classified into four groups: *preprocessing, visualization, manipulation*, and *analysis*. All preprocessing operations aim at improving or extracting object information from the given scenes. The purpose of visualization operations is to assist humans in perceiving and comprehending the object structure/characteristics/function in two, three and higher dimensions. The manipulation operations allow humans to interactively alter the object structures (mimic surgery, for example). The analysis operations enable us to quantify object structural/morphological/functional information.

In the rest of this section, we will cursorily examine these four classes of operations. The references given here are only samples from a literature, which has over a thousand publications excluding pure application–directed papers.

2.1 Terminology

Body region: Region of the space, for example, of the human body, that is imaged.

Pixel: Image element in a 2D scene.

Spel: Image element in a 2D, 3D, 4D, . . . , scene.

Scalar scene: A scene wherein each spel has one value assigned to it.

Vector scene: A scene wherein each spel has two or more values assigned to it. For example, T2 and proton density value in an MR scene.

Scene intensity: The value(s) assigned to the spel.

Scene domain: The set of all spels in a scene.

Pixel size: The size of the spel within the natural slice of the scene.

Slice spacing: The distance between the centers of two successive slices. In case of isotropic resolution, the spels have equal length in all dimensions.

Scene coordinate system: A coordinate system affixed to the scene.

Object: An object of study in the body region such as an organ or a pathology such as a tumor.

Object system: A collection of objects.

Structure: A computer representation of an object. It may be hard (crisp or binary) or fuzzy. In the hard case, a spel in the scene domain is considered to be either in the object or not in the object. In the fuzzy case, each spel has a value between 0 and 100 that indicates the degree of objectness assigned to the spel.

Structure system: A collection of structures.

Structure coordinate system: A coordinate system affixed to the structure/structure system.

Rendition: A 2D scene created by the computer that depicts some aspects of the object information contained in a scene. For example, the display of a slice of the scene, a shaded-surface display of a structure computed from the scene.

Display coordinate system: A coordinate system affixed to the display device.

Imaging device coordinate system: A coordinate system affixed to the imaging device. Information about the various coordinate systems is useful in registering scenes and structures.

2.1 Preprocessing Operations

The input to these operations is a set of scenes of a body region and the output is either a set of scenes or a structure system. Typically, the structure system contains only one structure corresponding to the object of interest.

Volume of Interest (VOI) [3]:

The input scene $C_i = (C_i, f_i)$ is converted to another smaller scene $C_o = (C_o, f_o)$ such that $C_o \subset C_i$ and f_o is a restriction of f_i to C_o. C_o is specified, usually interactively, to be the set of spels in a rectangular box, a sphere, or an ellipsoid (and their equivalents in higher dimensions). The purpose here is to make C_o just contain the objects of interest and to reduce the amount of data to be handled in other expensive operations. In large, routine applications, to automate this operation is a challenge. For this, clearly, some object knowledge is essential.

Filtering:

In these operations, the output scene C_o is obtained from the input scene C_i by modifying the intensity values in such a way as to either *enhance* object information [4] or to *suppress* unwanted non-object information such as noise [5] in C_i. Usually for these operations $C_o = C_i$. Various types of filters are available under these two categories. Unfortunately, usually some non-object information is also enhanced by the former methods and object information is also suppressed by the latter methods. The challenge in filtering is how to incorporate object knowledge into the filter design so as to achieve optimum performance.

Interpolation:

The purpose of interpolation is to change the level and orientation of discretization of C_i to a desired entity. For these operations, usually $C_o \neq C_i$. These operations are needed for converting a non-isotropically sampled scene into a scene with isotropic resolution, or for obtaining scenes at desired level and orientation of discretization. Two classes of operations are available: scene-based and object-based. In *scene-based* methods [6], $f_o(c)$ is estimated from $f_i(c')$ for spels c' lying in the close vicinity of c. In *object-based* methods [7], some object information is used in guiding interpolation. Object-based methods have been shown repeatedly in the literature to produce better results than scene-based methods. The challenge here is how to incorporate object knowledge specific to the application into the interpolation process.

Registration:

The purpose of registration is to represent structures obtained from multiple scenes in a common coordinate system. These are needed for combining object information obtained from multiple modalities (such as MRI, PET and CT) for the same body region, and for determining change, growth, motion, and displacement of objects over time. Again two classes of methods are available: scene-based and object-based. In *scene-based* methods [8], scene intensity patterns are matched, while in *object-based* methods [9] structures derived from scenes are matched. Both rigid and deformable methods of registration under both categories are available. As in other operations, the challenge here is to incorporate specific object knowledge into the registration process in both object-based and scene-based methods.

Segmentation:

This operation, the most crucial among all 3D imaging operations, outputs a structure system from a given set of scenes. It consists of two related tasks - recognition and delineation. *Recognition* is the process of determining roughly the whereabouts of the objects in the scene, and *delineation* is the process of determining precisely the spatial extent and composition of the objects in the scene. Knowledgeable humans usually outperform computer algorithms in the high-level task of recognition, while computer algorithms can outperform humans in the precise, accurate, and efficient delineation of objects.

There are two classes of approaches to recognition – *automatic* (knowledge-based/model-based) [10] and *human-assisted*. By far the latter is the most commonly used in practice. At the outset, two classes of approaches to delineation may be identified - *boundary-based* [11] and *region-based* [12]. In the former, the output is a set of boundaries of the object, and in the latter, it is a set of object regions. Each of these two strategies can be further divided into subgroups - *hard* [11] and *fuzzy* [12] - depending on whether the output is a hard or a fuzzy set. Combined with the two strategies of recognition, therefore, we may identify eight classes of approaches to segmentation.

As seen from the outline of other preprocessing operations, object knowledge usually facilitates all 3D imaging operations, including segmentation. This implies that segmentation is helpful/needed for all 3D imaging operations, ironically also for segmentation itself. To devise generic segmentation methods with high precision, accuracy, and efficiency, that can be quickly adapted to a given application, is indeed the greatest challenge in 3D imaging and in image analysis.

2.2 Visualization

The input to these operations is a set of scenes or a structure system and the output is a rendition.

Two classes of methods are available: *scene-based* wherein renditions are created directly from given scenes, and *object-based*, wherein renditions are created from (hard or fuzzy) structures extracted from the scene. Scene-based methods may be further divided into two groups: slice mode and volume mode. In the former, 2D "slices" of different orientations - natural, orthogonal, oblique and curved - are extracted from the scene and displayed as a montage or in a roam through fashion using gray scale, color or overlay [2]. In the latter, surfaces, interfaces and intensity distributions are displayed with a variety of 3D cues using *surface rendering* and *volume rendering* techniques [13, 14]. In object-

based methods, the hard or fuzzy structures extracted from the scene are displayed with 3D cues using surface rendering and volume rendering techniques [15, 16]. The challenges in visualization are the realistic display of objects including color, texture and surface properties, speeding up volume rendering, and objective evaluation of the large number of rendering schemes that are available.

2.3 Manipulation

These operations are needed for editing structures (for simulating surgery in medical applications) for unobscured visualization, and for developing aids for interventional procedures. They take as input a structure system and output another structure system by altering structures or their relationship. Two classes of operations are being developed: rigid [15, 17] with operations to cut, separate, add, subtract, move and mirror structures and their components, and deformable [18]. The main challenge in manipulation is to realize the manipulative operations realistically utilizing object material properties.

2.4 Analysis

The purpose of these operations is, given a set of scenes or a structure system, to generate a quantitative description of the morphology or function of the objects in the body region captured in the scene.

Two classes of operations are available: *scene-based*, wherein quantities based on scene intensities such as intensity statistics within object regions, tissue density, activity, perfusion, flow are estimated; and *object-based*, wherein quantities measured from segmented structures and how they change with time are estimated including distance, length, width, curvature, area, volume, kinematics and mechanics. The main challenge here is the assessment of the accuracy of the estimated measures.

3. GO DIGITAL, GO FUZZY

Any scene of any body region exhibits the two following important characteristics of the objects contained in the body region.

Graded Composition: The spels in the same object region exhibit heterogeneity of scene intensity due to the heterogeneity of the object material, and noise, blurring, and background variation introduced by the imaging device. Even if a perfectly homogeneous object were to be imaged, its scene will exhibit graded composition.

Hanging-togetherness (Gestalt): In spite of the graded composition, knowledgeable humans do not have any difficulty in perceiving object regions as a whole (Gestalt). This is a fuzzy phenomenon and should be captured through a proper theoretical and computational framework.

There are no binary objects or acquired binary scenes. Measured data always have uncertainties. Additionally, scenes are inherently digital. As seen from the previous section, no matter what 3D imaging operation we consider, we cannot ignore these two fundamental facts – fuzziness in data and their digital nature.

We have to deal with essentially two types of data - scenes and structures. We argue that, instead of imposing some sort of a continuous model on the scene or the structure in a hard fashion, taking an inherently digital approach, preferably in a fuzzy setting, for all 3D imaging operations can lead to effective, efficient and practically viable methods. Taking such an approach would require, in almost all cases, the development of the necessary mathematical theories and algorithms from scratch. We need appropriate theories and algorithms for topology, geometry and morphology, all in a fuzzy, digital setting, for dealing with the concept of a "structure" in scenes. Note that almost all the challenges we raised in the previous section relate to some form of object definition in scenes. Therefore, they all need the above mathematical and algorithmic developments. Additionally, since we deal with deformable objects (all soft-tissue organs, and often even bone, as in distraction osteogenesis – the process of enlarging or compressing bones through load applied

over a long period of time), we also need fuzzy digital mechanics theories and algorithms to handle structure data realistically for the operations relating to registration, segmentation, manipulation and analysis. Because of the digital setting, almost all challenges raised previously lead to discrete problems. Since we are dealing with n-dimensional scenes, the mathematics and algorithms need to be developed for hard and fuzzy sets of spels defined in n-dimensions.

In the rest of this section, we shall give two examples of digital/fuzzy approaches that motivated us to the above argument. The first relates to modeling an object as a 3D surface and visualizing it. The second relates to the fuzzy topological concept of connectedness and its use in segmentation.

3.1 Polygonal versus Digital Surfaces

In hard, boundary-based 3D scene segmentation (e.g., thresholding approaches), the output structure is often a 3D surface. The surface is represented usually either as a set of polygons (most commonly triangles [19]), or in a digital form [20] as a set of cubes or as a set of oriented faces of cubes. We shall describe in some detail how both the representation and rendering of such surfaces is vastly simpler and more efficient using digital approaches than using polygonal or other continuous approximations for them.

Let us consider the representation issues first. We shall subsequently consider the rendering issues. It is very reasonable to expect these surfaces to satisfy the following three properties since surfaces of real objects possess these properties.

(1) The surface is connected.

(2) The surface is oriented. This means that it has a well defined inside and outside.

(3) The surface is closed; that is, it constitutes a Jordan boundary. The latter implies that the surface partitions the 3D space into an "interior" set and an "exterior set such that any path leading from a point in the interior to a point in the exterior meets the surface.

The definitions of connectedness, orientedness, and Jordan property are much simpler, more natural and elegant in the digital setting using faces of cubes (or cubes) than using any continuous approximations such as representations via triangles. These global concepts can be arrived at using simple local concepts for digital surfaces [21, 22]. For example, orientedness can be defined by thinking of the faces to be oriented. That is, a face with a normal vector pointing from inside of the surface to its outside in the $-x$ direction is distinguished from a face at the same location with a face normal in exactly the opposite direction. Usually the surface normals at various points p on these surfaces (in both the polygonal and the digital case) are estimated independent of the geometry of the surface elements, for example, by the gradient at p of the intensity function f of the original scene from which the surface was segmented [23]. The gradient may also be estimated from other derived scenes such as a Gaussian smoothed version of the segmented binary scene [24]. Since the surface normals dictate the shading in the rendering process, the surface elements are used here only as a geometric guide rather than as detailed shape descriptors of the surface. Therefore the digital nature of the geometry introducing "staircase" effects in renditions can be more or less completely eliminated. Since the surface elements in the digital case are simple *and* all of identical size and shape, they can be stored using clever data structures that are typically an order of magnitude more compact than their polygonal counterparts [25]. Finally, there is a well-developed body of literature (e.g., [26], Chapters 6, 7) describing the theory of digital surfaces that naturally generalizes to n-dimensions for any n. Such a generalization is very difficult for polygonal representations.

Let us now come to the rendering issues. Rendering of digital surfaces is considerably simpler and more efficient than that of polygonal surfaces, the main reason being the simplicity of the surface elements and of their spatial arrangement in the former case [15, 27]. There are mainly two computational steps in any rendering algorithm: hidden surface removal and shading. Both these steps can be considerably simplified exploiting the special geometry of the surface elements, reducing most expensive computations to table lookup operations. For example, the faces in a digital surface can be classified into six groups based on their face normals (corresponding to the directions ($+x$, $+y$, $+z$, -x, -y, -z). For any viewpoint, all faces from at least three of these groups are not visible and hence can be discarded without

doing any computation per face [28]. There are other properties that allow the rapid projection of discrete surfaces in a back-to-front or front-to-back order by simply accessing the surface elements in some combination of column, row, and slice order from a rectangular array. Triangular and other continuous approximations to surfaces simply do not possess such computationally attractive properties. It was shown in [25] based on 10 objects of various sizes and shapes that, digital surface rendering entirely in software on a 300 MHz Pentium PC can be done about 4-17 times faster than the rendering of the same objects, whose surfaces are represented by triangles, on a Silicon Graphics Reality II hardware rendering engine for about the same quality of renditions. Note that the Reality II workstation is vastly more expensive than the Pentium PC.

The design of rendering engines such as Reality II was motivated by the need to visualize, manipulate, and analyze structure systems representing human-made objects such as automobiles and aircrafts. Early efforts in modeling in computer graphics took, for whatever reason, a continuous approach. Digital approaches to modeling have originated mostly in medical imaging [29] and have recently been applied also to the modeling of human-made objects [30] with equal effectiveness. Digital approaches are more appropriate for modeling natural objects (such as internal human organs) than continuous approaches. Natural objects tend to be more complex in shape and morphology than human-made objects. The applications of digital approaches to human-made objects [30] have demonstrated that the latter are equally appropriate even for human-made objects.

3.2 Fuzzy Connected Object Definition

Although much work has been done in digital geometry and topology based on hard sets, analogous work based on fuzzy sets is rare [12, 31]. As pointed out earlier, the uncertainties about objects inherent in scenes must be retained as realistically as possible in all operations instead of making arbitrary hard decisions. Although many fuzzy strategies have been proposed particularly for image segmentation, none of them has considered the spatio-topological relationship among spels in a fuzzy setting to formulate the concept of hanging-togetherness. (We note that the notion of hard connectedness has been used extensively in the literature [31], Chapter 1. But hard connectedness already assumes segmentation and removes the flexibility of utilizing the strength of connectedness in segmentation itself.) This is a vital piece of information that can greatly improve the immunity of object definition (segmentation) methods to noise, blurring and background variation.

Given the fuzzy nature of object information in scenes, the frameworks to handle fuzziness in scenes should address questions of the following form: How are objects to be mathematically defined in a fuzzy, digital setting taking into account the graded composition and hanging-togetherness of spels? How are fuzzy boundaries to be defined satisfying a Jordan boundary property? What are the appropriate algorithms to extract these entities from scenes in such a way as to satisfy the definitions? These questions are largely open. We will give one example below of the type of approaches that can be taken. This relates to fuzzy connected object definition [12]. This framework and the algorithms have now been applied extensively on 1000's of scenes in several routine clinical applications [32-34] attesting to their strength, practical viability, and effectiveness.

We define a fuzzy *adjacency* relation α on spels independent of the scene intensities. The strength of this relation is in [0, 1] and is greater when the spels are spatially closer. The purpose of α is to capture the blurring property of the imaging device.

We define another fuzzy relation κ, called *affinity*, on spels. The strength of this relation between any two spels c and d lies in [0, 1] and depends on α as well as on how similar are the scene intensities and other properties derived from scene intensities in the vicinity of c and d. Affinity is a local fuzzy relation. If c and d are far apart, their affinity is 0.

Fuzzy *connectedness* is yet another fuzzy relation on spels, defined as follows. For any two spels c and d in the scene, consider all possible connecting paths between them. (A path is simply a sequence of spels such that the successive spels in the sequence are "nearby".) Every such path has a strength which is the smallest affinity of successive

pairwise spels along the path. The strength of fuzzy connectedness between c and d is the largest of the strength of all paths between c and d.

A *fuzzy connected object* of a certain strength θ is a pool O of spels together with an objectness value assigned to each spel in O. O is such that for any spels c and d in O, their strength of connectedness is at least θ, and for any spels e in O and g not in O, their strength is less than θ.

Although the computation of a fuzzy connected object in a given scene for a given κ and θ appears to be combinatorially explosive, the theory leads to solutions for this problem based on dynamic programming. In fact, fuzzy connected objects in 3D scenes ($256 \times 256 \times 60$) can be extracted at interactive speeds (a few seconds) on modern PCs such as a 400 MHz Pentium PC.

4. CONCLUDING REMARKS

In this article, we have first given an overview of the operations available for 3D imaging - a discipline wherein, given a set of multidimensional scenes, the aim is to extract, visualize, manipulate, and analyze object information captured in the scenes. We have also raised numerous challenges that are encountered in real applications. Computers are digital. Current attempts in image acquisition, storage, and communication are completely digital or are proceeding in that direction. We have presented an argument with evidences that there are considerable advantages in taking an inherently digital approach to realizing all 3D imaging operations. We have also argued that, since object information in scenes is fuzzy, the digital approaches should be developed in a fuzzy framework to handle the uncertainties realistically. This calls for the development of topology, geometry, morphology and mechanics, all in a fuzzy and digital setting, all of which are likely to have a significant impact on imaging sciences such as medical imaging and their applications.

ACKNOWLEDGMENTS

REFERENCES

[1]   Z.H. Cho, J.P. Jones, and M. Singh: *Foundations of Medical Imaging*, New York, New York: John Wiley & Sons, Inc., 1993.

[2]   J. Udupa and G. Herman (eds.): *3D Imaging in Medicine*, 2nd Edition, Boca Raton, Florida: CRC Press, 1999.

[3]   J.K. Udupa: "Interactive Segmentation and Boundary Surface Formation for 3-D Digital Images," *Computer Graphics and Image Processing*, **18**, pp. 213-235, 1982.

[4]   P. Perona and J. Malik: "Scale Space and Edge Detection Using Anisotropic Diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligent* **12**, pp. 629-639, 1983.

[5]   R.T. Chin and C.L. Yeh: "Quantitative Evaluation of Some Edge-Preserving Noise Smoothing Techniques," *Computer Graphics and Image Processing* **23**, pp. 67-91, 1983.

[6]   G.T. Herman, S.W. Rowland, and M.-M. Yau: "A Comparative Study of the Use of Linear and Modified Cubic Spline Interpolation for Image Reconstruction," *IEEE Transactions on Nuclear Science* **NS-26(2)**, pp. 2879-2894, 1979.

[7]   S.P. Raya and J.K. Udupa: "Shape-Based Interpolation of Multidimensional Objects," *IEEE Transactions on Medical Imaging* **9(1)**, pp. 32-42, 1990.

[8]   R.P. Woods, S.R. Cherry, and J.C. Mazziotta: "Rapid Automated Algorithm for aligning and Reslicing PET Images," *Journal of Computer Assisted Tomography* **17(4)**, pp. 620-633, 1992.

[9]   C.A. Pelizzari, G.T.Y. Chen, D.R. Spelbring, R.R. Weichselbaum, and C.-T. Chen: "Accurate Three-Dimensional Registration of CT, PET, and/or MR Images of the Brain," *Journal of Computer Assisted Tomography* **13**, pp. 20-26, 1989.

[10] L. Gong and C. Kulikowski: "Composition of Image Analysis Processes Through Object-Centered Hierarchical Planning," *IEEE Transactions on Pattern Analysis and Medicine Intelligence* **17(10)**, pp. 997-1008, 1995.

[11] T. McInerney and D. Terzopoulos: "A Dynamic Finite Element Surface Model for Segmentation and Tracking in Multidimensional Medical Images With Application to Cardiac 4D Image Analysis," *Computerized Medical Imaging and Graphics* **19(1)**, pp. 69-83, 1995.

[12] J.K. Udupa and S. Samarasekera: "Fuzzy Connectedness and Object Definition: Theory, Algorithms, and Applications in Image Segmentation," *Graphical Models and Image Processing* **58(3)**, pp. 246-261, 1996.

[13] S. Goldwasser and R. Reynolds: "Real-time Display and Manipulation of 3-D Medical Objects: The Voxel Machine Architecture," *Computer Vision, Graphics, and Image Processing* **39(4)**, pp. 1-27, 1987.

[14] P.G. Lacroute: "Fast Volume Rendering Using a Shear-Warp Factorization of the Viewing Transformation," Technical Report CSL-TR-95-678, Ph.D. Thesis, Departments of Electrical Engineering and Computer Science, Stanford University, Stanford, California, September 1995.

[15] J.K. Udupa and D. Odhner: "Fast Visualization, Manipulation, and Analysis of Binary Volumetric Objects," *IEEE Computer Graphics and Applications* **11(6)**, pp. 53-62, 1991.

[16] J.K. Udupa and D. Odhner: "Shell Rendering," *IEEE Computer Graphics and Applications* **13(6)**, pp. 58-67, 1993.

[17] D. Odhner and J.K. Udupa: "Shell Manipulation: Interactive Alteration of Multiple-Material Fuzzy Structures," *SPIE Proceedings* **2431**, pp. 35-42, 1995.

[18] K. Waters: "A Muscle Model for Animating Three Dimensional Facial Expression," *Proceedings of SIGGRAPH'87* **21(3)**, pp. 17-24, 1987.

[19] W. Lorensen and H. Cline: "Marching Cubes: A High Resolution 3D Surface Construction Algorithm," *Computer Graphics* **21**, pp. 163-169, 1987.

[20] J.K. Udupa, S.N. Srihari, and G.T. Herman: "Boundary Detection in Multidimensions," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-4**, pp. 41-50, 1982.

[21] E. Artzy, G. Frieder, and G. Herman: "The Theory, Design, Implementation and Evaluation of a Three-Dimensional Surface Detection Algorithm, " *Computer Graphics and Image Processing* **15**, pp. 1-24, 1981.

[22] D. Gordon and J. Udupa: "Fast Surface Tracking in Three-Dimensional Binary Images, " *Computer Vision, Graphics, and Image Processing* **45**, pp. 196-214, 1989.

[23] K.H. Höhne and R. Bernstein: "Shading 3D Images from CT Using Gray-Level Gradients," *IEEE Transactions on Medical Imaging* **5**, pp. 45-47, 1986.

[24] J. Udupa and R. Goncalves: "Imaging Transforms for Visualizing Surfaces and Volumes," *Journal of Digital Imaging* **6**, pp. 213-236, 1993.

[25] G.J. Grevera and J.K. Udupa: "Order of Magnitude Faster Surface Rendering Via Software in a PC Than Using Dedicated Hardware," *SPIE Proceedings* **3658**, pp. 202-211, 1999.

[26] T.Y. Kong, and A. Rosenfeld (eds.): *Topological Algorithms for Digital Image Processing*, New York, New York: Elsevier Science B.V., 1996.

[27] G. Frieder, G. Gordon, and R. Reynolds: "Back-to-Front Display of Voxel-Based Objects," *IEEE Computer Graphics and Applications* **5**, pp. 52-60, 1985.

[28] E. Artzy: "Display of Three-Dimensional Information in Computed Tomography, " *Computer Graphics and Image Processing* **9**, pp. 196-198, 1979.

[29] G. Herman and H. Liu: "Three-Dimensional Display of Human Organs from Computed Tomograms, " *Computer Graphics and Image Processing* **9**, pp. 679-698, 1979.

[30] A. Kaufman: "Efficient Algorithms for 3-D Scan Conversion of Parametric Curves, Surfaces, and Volumes," *Computer Graphics* **21**, pp. 171-179, 1987.

[31] R. Rosenfeld: "Fuzzy Digital Topology," *Information and Control* **40**, pp. 76-87, 1979.

[32] J.K. Udupa, L. Wei, S. Samarasekera, Y, Miki, M.A. van Buchem, and R.I. Grossman: "Multiple Sclerosis Lesion Quantification Using Fuzzy Connectedness Principles," *IEEE Transactions on Medical Imaging* **16**, pp. 246-261, 1997.

[33] B.L. Rice and J.K. Udupa: "Clutter-Free Volume Rendering for Magnetic Resonance Angiography Using Fuzzy Connectedness," *International Journal of Imaging System and Technology* **11**, pp. 62-70, 2000.

[34] P.K. Saha, J.K. Udupa, E.F. Conant, and D.P. Chakraborty: "Near-Automatic Segmentation and Quantification of Mammographic Glandular Tissue Density," *SPIE Proceedings* **3661**, pp. 266-276, 1999.