

Analysis and Visualization of Multidimensional Medical Images

The work presented in this thesis has been carried out at:

- Department of Computer Systems and Telematics at the Norwegian University of Science and Technology.
- National Center for Fetal Medicine, dept. of Gynecology and Obstetrics at Trondheim University Hospital.
- Vingmed Sound, Horten

Abstract

This thesis is a contribution to the field of medical imaging. The first part of the thesis relates to analysis and display of medical ultrasonic images. The second part of the thesis relates to analysis and display of Nuclear Magnetic Resonance images. The third part relates more generally to analysis and display of multispectral and/or multidimensional medical images.

We develop an algorithm for boundary detection in ultrasonic imaging which integrates anisotropic diffusion with a signal model for displayed ultrasonic images. The usefulness of the algorithm is demonstrated for applications in volume rendering and automatic contour detection.

Problems related to rendering of three-dimensional (3D) ultrasonic data are discussed. A rendering method for ultrasonic data which incorporates our boundary detection scheme is proposed. This and several other volume rendering techniques are evaluated for display of fetal organs. With the proposed rendering method we have produced rendered images with very fine details which were difficult or impossible to interpret from the original 2D images.

The problem of segmenting and visualizing brain lesions in 3D magnetic resonance images is studied. A probabilistic classification scheme which combines a contour detection algorithm with Haslet's contextual classification method is integrated with Drebins volume rendering algorithm. The combination of the methods are evaluated for display of soft tissue structures in the human brain. Some enhancements to the classification strategy are discussed.

Two new algorithms for visualization of multi attribute medical images are introduced. The aim of the algorithms is to provide as much information as possible from the multi attribute image in one gray scale or color image without making any rigid classification into different tissue categories. One algorithm is based on Sammons nonlinear projection, the other algorithm projects the multi attribute data onto a curve defined by an ordered set of reference vectors.

A general framework for fast visualization of multispectral volume data is presented. Dedicated hardware with a non-numeric co processor is utilized in the first step of the rendering pipeline to process the volume data and extract voxels according to feature characteristics. The selected voxels are generated in a front-to-back (or back-to-front) order and projected to the view plane where a 3D rendering is accumulated with an adaptation of the shell rendering technique proposed by Udupa and Odhner.

An efficient algorithmic framework for fuzzy object segmentation is presented. The framework is targeted at volume rendering methods which are fuzzy display techniques. In many volume rendering methods, opacity values are assigned to each voxel on the basis of local characteristics. With the proposed framework we incorporate a measure of spatial connectedness into the opacity function. A weight is introduced for each connection between neighboring voxels. Dijkstras algorithm is utilized for computation of a minimum weight path from a user defined set of seed voxels to all other voxels. A simple and very efficient implementation of Dijkstras algorithm which exploits some specific properties of our application has been developed. The presented algorithm is demonstrated for visualization of tumor and vessel geometry using three dimensional ultrasonic images.

Contents

- 1** Organization
- 2** Summary of the thesis
- 3** Acknowledgements
- 4** List of papers
- 5** A brief overview of important medical imaging modalities
- A.** “Scale-space and boundary detection in ultrasonic imaging using nonlinear signal-adaptive anisotropic diffusion”
- B.** “Volume Rendering of 3D Medical Ultrasound Data using Direct Feature Mapping”
- C.** “Segmentation and Visualization of Brain Lesions in Multispectral Magnetic Resonance Images”
- D.** “Visualization of multi attribute medical images”
- E.** “Shell rendering with hardware supported data extraction”
- F.** “An efficient algorithmic framework for fuzzy object segmentation”

1 Organization

This thesis is mainly a result of work that has been published in [133, 134, 129, 130, 131, 128, 101, 66, 106, 107]. References to these papers are also given in a separate list in section 4. The thesis contains 4 published papers [129, 130, 128, 66]. In addition, there is one paper which has not been published yet + one paper which summarizes work from two papers [106, 107]. The first paper [129] has been slightly extended since it was published. Because each paper is meant to be self-contained, some material is repeated, especially in the introductions. The following section is a summary where each paper is given a brief presentation. Each presentation includes the title of the paper, reference, problem description and main contributions. Acknowledgments are given in section 3. The next section contains a common reference list for all the papers and the introductory part of the thesis. A brief overview of different imaging modalities is given in section 5. Since most of the papers in this thesis covers applications in multidimensional ultrasound imaging we give a brief introduction to conventional and multidimensional ultrasound imaging in a separate section. The papers are given in the subsequent chapters.

2 Summary of the thesis

In the first paper, we present an algorithm for boundary detection in ultrasonic imaging. The paper is called “Scale-space and boundary detection in ultrasonic imaging using nonlinear signal-adaptive anisotropic diffusion” and is a slightly extended version of the paper [129] (The section on quantitative evaluation has been added later). The algorithm presented in this paper was mainly developed in order to improve the quality of gradient based visualization techniques. Several visualization algorithms involves gradient computations in order to emphasize boundaries between different tissues. Due to the coherent detection, constructive and destructive interference between neighboring scatterers give rise to multiplicative noise, speckle, which modulates the ultrasonic gray level image. Simple gradient operators are very sensitive to high frequency noise such as speckle. Speckle can be suppressed by traditional linear filtering techniques, but these techniques tend to blur the boundaries. An edge preserving smoothing algorithm which integrates anisotropic diffusion with a signal model for displayed ultrasonic images was therefore developed. Anisotropic diffusion was originally proposed by Perona and Malik [118] as a technique for boundary detection and scale space filtering. The original filtering scheme tends to produce artificially sharp edges. We have modified the filtering scheme in order to avoid this sharpening. We have also modified the scheme so that the filtering strength is adjusted according to the local signal level. The properties of the filtering strategy is demonstrated in numerical experiments both on synthetic images and real ultrasonic images. A recently proposed framework for quantitative evaluation of restoration algorithms [104] is used in the evaluation process. The usefulness of the strategy is demonstrated for applications in volume rendering and automatic contour detection. A real time temporal speckle suppression filter has later been developed on the basis of the work presented in this paper. The real time filter is now a feature on the System Five ultrasound scanner from Vingmed Sound.

In the second paper, several volume rendering techniques are evaluated for visualization of 3D medical ultrasound images. The paper is called “Volume Rendering of 3D Medical Ultrasound Data using Direct Feature Mapping” [130]. Several fundamental limitations in the imaging system, make volume rendering of ultrasonic data a difficult challenge. Phase effects (speckle and phase aberrations), acoustic noise and thermal noise make ultrasonic images hard to interpret. Another severe limitation is that the received echo signal from structures lying parallel to the radial direction can be very weak and thus difficult to render. Yet another problem is that high density material may totally absorb the transmitted ultrasound waves, leaving structures further from the transducer in the shadow. These limitations make it very difficult to develop general and reliable tissue classification schemes. In this paper we have focused on methodologies which do not rely on any initial classification into different tissue categories. Instead, features are extracted from the original 3D data-set, and projected into the view-plane. The feature extraction then serves as a low-level segmentation process which then is meant to emphasize on clinically interesting features of the data-set. In particular, we have developed a rendering pipeline which incorporates the filtering scheme which was presented in the first paper. With this rendering pipeline we produced rendered images with very fine details which were difficult or impossible to interpret from the original 2D images. The initial filtering was essential to achieve satisfactory results.

Magnetic Resonance imaging is a very important imaging modality for display of soft tissue structures. The multiparameter dependence of signals from tissue voxels give high contrast between several different soft tissue types. To improve contrast even more, contrast-enhancing agents are often given to the patient prior to image acquisition. For examinations of the human brain, several parallel cross-sectional images are usually obtained. The physician then has to do a sequential examination of the cross sectional images in order to mentally reconstruct the three dimensional brain structures. Volume rendering makes it possible to present an overview of three dimensional structures to the physician in a single image. In the third paper, we integrate a probabilistic multi parameter classification method with a rendering method developed by Drebin et. al. [22]. The paper is called "Segmentation and Visualization of Brain Lesions in Multispectral Magnetic Resonance Images" [66]. The classification method combines Hasslet's contextual classification method with a contour detection algorithm. The probability that a voxel belongs to tissue category k is computed for all possible categories $k \in \{1..K\}$. In Drebin's method, different tissue categories can be displayed simultaneously by assigning different colors to them. The color of a voxel is given as the normalized sum of tissue colors weighted by their associated probability values. In this way, many artifacts which would have been introduced by a binary classification scheme is avoided. The methods were evaluated on images of a human brain with an intracranial tumor (Metastasis). The results were very encouraging.

In the fourth paper we present new algorithms for visualization of multi attribute medical images. The paper is called "Visualization of multi attribute medical images" [128]. Visualization of multi attribute images is gaining increasing importance in many medical applications. In MRI, multi-parameter images are routinely acquired. Several algorithms have been proposed for visualization of multi attribute images, with special attention on remotely sensed data. However, most of the proposed algorithms deal with automatic segmentation into labeled regions. In many cases, the automatic classification procedure will introduce misclassified pixels and the displayed image may therefore contain misleading information. In our paper we concentrate on methods that provide the multi spectral information in a gray scale or color image without making any rigid classification. Gray scale images are of special interest as the human eye is considerably more sensitive to spatial variations in intensity than it is to chromatic variations. A nonlinear mapping is made from the original N -dimensional feature space to an M -dimensional output space where $M < N$ and $M \in \{1..3\}$. Two different nonlinear projection methods are investigated for this purpose. We first present a method based on Sammons nonlinear projection algorithm. Sammons algorithm is a gradient descent strategy which aims at preservation of inter pattern distances by minimizing a cost function which measures the so-called *Sammon stress*. We also introduce a new algorithm where the original multi attribute data is projected onto a curve in feature-space defined by an *ordered set* of reference vectors, and a gray scale is mapped along this curve. The optimal ordering of the reference vectors is found as a minimal cost permutation, where the cost function is a weighted sum of inter pattern distances in N space. Our algorithms are compared to principal component analysis (PCA) and a recently published algorithm based on Kohonens self organizing maps. The usefulness of the new algorithms are demonstrated for visualization of both reproducible synthetic images and real MR images.

In the fifth paper we present a general framework for fast visualization of multispectral volume

data. The paper is called “Shell rendering with hardware supported data extraction” [107, 106]. Dedicated hardware with a non-numeric co processor is utilized in the first step of the rendering pipeline to process the volume data and extract voxels according to feature characteristics. This capability is used to select voxels according to automatic classification results or real-time descriptions of regions of interest supplied by the user in an interactive environment. By this step we can in real-time reduce the number of voxels that have to be considered in the rendering and increase the speed of the volume rendering accordingly. The selected voxels are generated in a front-to-back (or back-to-front) order and projected to the view plane where a 3D rendering is accumulated with an adaptation of the shell rendering technique proposed by Udupa and Odhner. The paper includes an overview of the underlying hardware architecture and presents numerical experiments with a software simulator.

In the sixth paper, we present an algorithmic framework for fuzzy object segmentation. The framework is especially targeted at volume rendering applications which by definition are fuzzy display methods. The paper is called “An efficient algorithmic framework for fuzzy object segmentation”, and is based on an earlier paper [131]. *Surface rendering* and *volume rendering* are two basic approaches to visualization of 3D objects. In surface rendering, an explicit model of the imaged structure is extracted by using a surface detection scheme. This approach relies heavily on the robustness of the detection scheme. On the other hand, volume rendering techniques do not rely on explicit models of the imaged objects. The imaged scene is rather viewed as a semitransparent volume, where transitions between different materials may be smooth. In several of algorithms, opacity values are assigned to each voxel on the basis of local characteristics. In many cases it may be difficult or impossible to isolate an interesting structure purely on the basis of local features, since different materials may share the same local characteristics. Still, it may be possible to isolate an object as a separate region of spatially connected voxels with similar local features. This has motivated us to develop a segmentation algorithm that incorporates a measure of spatial connectedness into the opacity function. A cost is introduced for each connection between neighboring voxels. Dijkstras algorithm is utilized for computation of a minimum cost path from a user defined seed voxel to all other voxels. The minimum cost of connecting a voxel to the initial seed is then mapped into an opacity value. Different weight functions are discussed. The practical utility of the algorithm depends a lot on its efficiency. A simple and very efficient implementation of Dijkstras algorithm which exploits some specific properties of our application has therefore been developed. The presented algorithms can be directly related to a framework for fuzzy object segmentation which independently has been proposed by Udupa and Samasekara [150]. The presented algorithmic framework is demonstrated for visualization of tumor and vessel geometry using three dimensional ultrasonic images.

3 Acknowledgments

The work presented in this thesis has mainly been carried out at the Department of Computer Systems and Telematics (IDT) at The Norwegian Institute of Technology, partly at the National Center of Fetal Medicine in Trondheim and partly at Vingmed Sound in Horten. There are many people that I wish to thank for support and good cooperation. First of all, I am very grateful to my supervisor Bjørn Olstad at IDT who encouraged me to start a doctoral study after I finished my diploma thesis. Bjørn Olstad has been an excellent supervisor, always full of ideas and always giving constructive feedback on the work I have done. I am also grateful to Hans Torp at Department of Biomedical Engineering for technical support in several experiments and for sharing his knowledge on physics of ultrasonic imaging. I want to thank the clinical staff at the National Center of Fetal Medicine in Trondheim, especially Sturla Eik-Nes who established the experimental ultrasound laboratory and Torvid Kiserud who helped me a lot with acquisition of 3D ultrasound data. I would like to thank Gaute Myklebust, Sevald Berg and Klaus Schipper for good cooperation on several occasions. I would especially like to thank Anders Torp for good cooperation and many fruitful discussion during the whole working period. Vingmed Sound has provided the ultrasonic scanners used in several ultrasonic experiments. I am very grateful for this support. I would also like to thank Hans Jørgen Alker for providing excellent four dimensional acquisitions.

During my doctoral study I have been granted a scholarship by the Norwegian Research Council. I very grateful for this economic support.

4 List of papers

- Erik Steen and Bjørn Olstad: “Volume Rendering In Medical Ultrasound Imaging”, *In proc. of 8th Scandinavian Conference on Image Analysis, Tromsø*, 1993
- Erik Steen and Bjørn Olstad: “Volume Rendering in medical ultrasound imaging based on Non-Linear Filtering”, *In proc. of IEEE Winter Workshop On Nonlinear Digital Signal Processing*, 1993
- Erik Steen and Bjørn Olstad: “Scale-space and boundary detection in ultrasonic imaging, using signal-adaptive anisotropic diffusion”, *In proc. of SPIE Medical Imaging, Image processing conference, Newport Beach California*, 1994
- Erik Steen and Bjørn Olstad: “Volume Rendering of 3D Medical Ultrasound Data using Direct Feature Mapping”, *IEEE Transactions on Medical Imaging*, vol. 13, no 3, September 1994, pages 517-525
- E. Steen and G. Myklebust and B. Olstad: “Visualization of Multiattribute Medical Images”, *In proc. of SPIE Medical Imaging, Image Display conference, San Diego*, 1995
- E. Steen and B. Olstad and G. Myklebust and S. Berg: “Combined Visualization of Tumor and Vessel Geometry”, *In proc. of IEEE International Ultrasonics Symposium Cannes*, 1994
- Marit Holden and Erik Steen and Arvid Lundervold: “Segmentation and Visualization of Brain Lesions in Multispectral Magnetic Resonance Imaging”, *Computerized Medical Imaging and Graphics, volume 19, number 2*, 1995, pages 171-183
- Gaute Myklebust and Erik Steen: “Neural Networks in visualization of multispectral medical images”, *To appear in Information Sciences*
- B. Olstad and Erik Steen and Olav Sandstå: “Shell rendering with hardware supported data extraction”, *In proceedings of SPIE Medical Imaging, San Diego*, 1995
- B. Olstad and E. Steen and A. Halaas: “Image filtering techniques and VLSI architectures for efficient data extraction in shell rendering”, *In proceedings of IEEE ICIP 1995*

References

- [1] L.M Lifshitz and S.M Pizer. A multiresolution hierarchical approach to image segmentation based on intensity extrema. *Technical report, Dep Comput Sci and Radiol. Univ North Carolina, Chapel Hill,*, 1987.
- [2] Fritz Albrechtsen. Fractal texture signature estimated by multiscale min-max and max-min operators on landsat-5 mss images of the antarctic. In *Proceedings 6thSCIA Oulu Finland June 19-22*, pages 995–1002, 1989.
- [3] A. A. Amini, S. Tehrani, and T. E. Weymouth. Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints. In *Proc. Second Int. Conf. Computer Vision, Tarpon Springs, FL*, pages 95–99, 1988.
- [4] A. A. Amini, T. E. Weymouth, and R. C. Jain. Using dynamic programming for solving variational problems in vision. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(9):855–867, 1990.
- [5] Bjørn A.J Angelsen. *Waves, Signals and Signal Processing in Medical Ultrasonics*. Department of Biomedical Engineering, UNIT Trondheim, 1991.
- [6] L. Axel, G. T Herman, and J. K Udupa et.al. Three-dimensional display of NMR cardiovascular images. *Comput. Assist. Tomography*, 7:172–174, 1983.
- [7] R. Bajcsy. Computer description of textured surfaces. In *Proc. Third Int Joint Conf AI*, 1973.
- [8] J. C. Bamber and C. Daft. Adaptive filtering for reduction of speckle in ultrasonic pulse-echo images. *Ultrasonics*, 24:41–44, 1986.
- [9] J.C Bamber, R.J Eckersley, P. Hubregtse, N.L Bush, D.S Bell, and D.C Crawford. Data processing for 3D ultrasound visualization of tumor anatomy and blood flow. In *Visualization in Biomedical Computing*, Chapel Hill, NC, SPIE, 1992.
- [10] C. Barillot. Surface and volume rendering techniques to display 3D data. *IEEE Engineering in Medicine and Biology Magazine*, 12(1):111–119, 1993.
- [11] J. Besag. On the statistical analysis of dirty pictures. *J. Royal Stat. Soc., Series B*, 48(3):259–302, 1986.
- [12] J. Besag. Digital image processing: Towards Bayesian image analysis. *Journal of Applied Statistics*, 16(3):395–407, 1989.
- [13] R. W. Boesel and Y. Bresler. A new paradigm for optimal multi-parameter image visualization. *SPIE Visualization in Biomedical Imaging*, 1808, 1992.
- [14] Michael Bomans, Karl-Heinz Hohne, Ulf Tiede, and Martin Riemer. 3D segmentation of mr images of the head for 3D display. *IEEE Trans. on Medical Imaging*, 9(2), June 1990.

- [15] G Borgefors. Distance transformations in arbitrary dimensions. *Computer vision, graphics and image processing*, 27:321–345, 1984.
- [16] Alan C. Bovik. On detecting edges in speckle imagery. *IEEE Trans. on Accoustics, speech and signal processing*, 36(10), 1988.
- [17] J.F Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
- [18] Chi-Chang Chen, John S Daponte, and Martin D Fox. Fractal feature analysis and classification in medical imaging. *IEEE Trans. on Medical Imaging*, 8(2), June 1989.
- [19] H.S Choi, D.R Haynor, and Y Kim. Partial volume tissue classification of multichannel magnetic resonance images- a mixed model. *IEEE Transactions on Medical Imaging*, 10(3):395–407, 1991.
- [20] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. Mc Graw-Hill, 1990.
- [21] R. Bakalash D. Cohen, A. Kaufman and S. Bergman. Real-time-discrete shading. *The Visual Computer*, 8(1):16–27, February 1990.
- [22] R. A. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. *Computer Graphics*, 22(4):65–74, August 1988.
- [23] R. O Duda and P. E Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.
- [24] A. Weimann et. al. Wertigkeit der sonographie bei der diagnostik und verlaufskontrolle von hemangiom und fokal nodularer hyperplasie der leber. *Ultraschall Klin Prax*, 9:71–74, 1994.
- [25] Ahuja et al. Faster algorithms for the shortest path problem. *Technical report 193, MIT Operations Research Center*, 1988.
- [26] D. Harwood et al. A new class of edge-preserving smoothing filters. *Report CAR-TR-59 Univ of Maryland*, 1984.
- [27] D. N. Levin et al. Integrated three-dimensional display of MR and PET images of the brain. *Radiology*, 172:783–789, 1989.
- [28] M Verlande et al. 3D reconstruction of the beating left ventricle and mitral valve based on multiplanar tee. In *Proceedings. Computers in Cardiology*. IEEE Computer Society Press, 1991.
- [29] P. D. LeRoux et. al. A comparison between preoperative magnetic resonance and intraoperative ultrasound tumor volumes and margins. *J. Clin. Ultrasound*, 22:29–36, January 1994.

- [30] Paul A. Picot et al.. Three-dimensional colour doppler imaging. *Ultrasound in Medicine and Biology*, 19(2), 1993.
- [31] R. Pini et al. Echocardiographic three-dimensional visualization of the heart. *NATO ASI series, VOL. F60. 3D imaging in Medicine. Edited by K.H. Hohne et al. Springer-Verlag Berlin Heidelberg*, 1990.
- [32] R. Pini et al.. Computed tomography of the heart by ultrasound. In *Proceedings. Computers in Cardiology*. IEEE Computer Society Press, 1991.
- [33] R.A Reynolds et al.. Realistic presentation of 3D medical datasets. In *Proceedings of Graphics Interface '88*, pages 71–77, 1988.
- [34] RI Kitney et al.. 3D visualization for the study of arterial disease and tissue characterisation. In *Proceedings. Computers in Cardiology*. IEEE Computer Society Press, 1991.
- [35] S. Peleg et al. Multiple resolution texture analysis and classification. *IEEE Transactions on PAMI*, 6:518–523, 1984.
- [36] S. Webb et. al. *Physics of Medical Imaging*. Adam Hilger, 1988.
- [37] U. Schmiedl et. al. The utility of principal component analysis for the image display of brain lesions. a preliminary, comparative study. *Magn. Resn. Med.*, 4, 1987.
- [38] E. J. Farrell. Color display and interactive interpretation of three-dimensional data. *IBM J. Res. Develop.*, 17:356–366, April 1983.
- [39] M. Fatemi and A.C. Kak. Ultrasonic b-scan imaging: theory of image formation and a technique for restoration. *Ultrasonic Imaging*, 2:1–47, 1980.
- [40] J.D Foley, A. van Dam, S.K Feiner, and J.F Hughes. *Computer Graphics: Principles and Practice*. Addison Wesley, 1990.
- [41] M.D Fox. Visualization of multidimensional and multispectral biomedical images. *Electronic Imaging '88*, 2:1050–1053, 1988.
- [42] G. Frieder and R. Reynolds. Back-to-front display of voxel-based objects. *IEEE Computer Graphics and Application*, 5(1):52–60, January 1985.
- [43] N. Friedland and D. Adam. Automatic ventricular cavity boundary detection from sequential ultrasound images using simulated annealing. *IEEE Trans. Med. Imaging*, 8(4):344–352, 1989.
- [44] J. E. Gardener. Volume imaging of soft tissues with ultrasound. In *IEE Colloquium on 3D Imaging Techniques for Medicine*, 1991.
- [45] M. R. Garey and D. S. Johnson. *Computers and Intractability, a Guide to the Theory of NP-completeness*. Freeman, 1979.

- [46] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution, and Bayes restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.*, PAMI-6:721–741, 1984.
- [47] F. Godtlielsen. Noise reduction using markov random fields. *Journal of Magnetic Resonance*, 92:102–114, 1991.
- [48] K. J. Goldman and T. R. Miller. Interactive combining of functional images. *Eur. Journal of Nuclear Medicine*, (11), 1985.
- [49] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison Wesley, 1992.
- [50] D. Gordon and J. Udupa. Fast surface tracking in 3D binary images. *Comp. Vis. Graph. and Im. Proc.*, 45(2):196–241, 1989.
- [51] Åge Grønningsæther. *Reduction of Electronic Noise in Intravascular Ultrasound Imaging*. PhD thesis, The Norwegian Institute of Technology, 1992.
- [52] Gerig Guido, Kubeler Olaf, Kikinis Ron, and Jolesz F. A. Nonlinear anisotropic filtering of MRI data. *IEEE Trans on Medical Imaging*, 11(2), June 1992.
- [53] E. L. Hall. *Computer Image Processing and Recognition*. Academic Press, 1979.
- [54] R. M. Haralick and L. G. Shapiro. Glossary of computer vision terms. *Pattern Recognition*, 24(1):69–93, 1991.
- [55] G. Harikumar and Y. Bresler. Multi-parameter image visualization by projection pursuit. *SPIE Visualization in Biomedical Imaging*, 1808, 1992.
- [56] J. A. Hartigan. *Clustering Algorithms*. John Wiley & Sons, Ltd., New York, 1975.
- [57] J Haslett. Maximum likelihood discriminant analysis on the plane using a markovian model of spatial context. *Pattern Recognition*, 18:287–296, 1985.
- [58] K. H Hausser and H. R. Kalbitzer. *NMR in Medicine and Biology - Structure Determination, Tomography, In Vivo Spectroscopy*. Springer-Verlag, Berlin, 1991.
- [59] D.C Hemmy, D.J David, and G.T Herman. Three-dimensional reconstruction of craniofacial deformity using computed tomography. *Neurosurgery*, 13:534–541, 1983.
- [60] G. T. Herman, J. Zeng, and C.A. Bucholtz. Shape-based interpolation. *IEEE Computer Graphics and Applications*, 12(3):69–79, 1992.
- [61] Gabor T. Herman. A survey of 3D medical imaging technologies. *IEEE Engineering In Medicine and Biology*, pages 15–17, December 1990.
- [62] K.H. Hohne, M. Bomans, U. Tiede, and M. Riemer. Display of multiple 3D objects using the generalized voxel-model. In *Proc. of the SPIE - The International Soc for Optical Engineering*, 1988.

- [63] K.H. Hohne and U. Tiede. Surface rendering. investigation of 3D-rendering algorithms. *IEEE Computer Graphics & Applications*, March 1990.
- [64] K.H. Hohne and U. Tiede. Rendering tomographic volume data: Adequacy of methods for different modalities and organs. *NATO ASI series, 3D imaging in Medicine*. Edited by K.H. Hohne and U. Tiede F60, 90.
- [65] Jørn Hokland and Patrick A. Kelly. Markov models of specular and diffuse scattering in restoration of medical ultrasound images. *Submitted to IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 1995.
- [66] Marit Holden, Erik Steen, and Arvid Lundervold. Segmentation and visualization of brain lesions in multispectral magnetic resonance imaging. *Computerized Medical Imaging and Graphics*, 19(2):171–183, 1995.
- [67] F. Hottier and A. Collet Billon. 3D echography: Status and perspective. *NATO ASI Series, 3D Imaging In Medicine*, F. 60, 1990.
- [68] John Y HSIAO and Alexander A Sawchuk. Supervised textured image segmentation using feature smoothing and probabilistic relaxation techniques. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 11(12):1279–1292, December 1989.
- [69] P. J. Huber. Projection pursuit. *Annals of Statistics*, 13(2), 1985.
- [70] Kajiya J, T. The rendering equation. *Computer Graphics*, 20(4):143–149, August 1986.
- [71] A. K. Jain and J. Mao. Artificial neural network for nonlinear projection of multivariate data. In *IJCNN Intern. Joint conf. on Neural Networks*, volume III, 1992.
- [72] N.L. Johnson and S. Kotz. *Distributions in Statistics: Continuous Multivariate Distributions*. John Wiley and Sons, Inc., 1972.
- [73] I. T Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.
- [74] J. W. Sammon Jr. A nonlinear mapping for data structure analysis. *IEEE Trans. On Computers*, 18(5), 1969.
- [75] J. T. Kajiya and B. P. Von Herzen. Ray tracing volume densities. *Computer Graphics (Proc. Siggraph)*, 18(3):165–173, July 1984.
- [76] David N. Kennedy, Pauline A Filipek, and Verne S Jr. Caviness. Anatomic segmentation and volumetric calculations in nuclear magnetic resonance imaging. *IEEE Trans. on Medical Imaging*, 8, March 1989.
- [77] H. Knutsson and G. H. Granlund. Texture analysis using 2d quadrature filters. In *Proc. ICASSP 83 IEEE Conf on Acoust, Speech and Signal Processing*, 1983.
- [78] J. Koendrink. The structure of images. *Biol. Cybern*, 50:363–370, 1984.

- [79] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9), Sept 1990.
- [80] C. Kotropoulos, X. Magnisalis, I. Pitas, and M.G Strintzis. Nonlinear ultrasonic image processing based on signal-adaptive filters and self-organizing neural networks. *Submitted to IEEE Transactions on Image Processing*, 1993.
- [81] C. Kotropoulos and Ioannis Pitas. Optimum nonlinear signal detection and estimation in the presence of ultrasonic speckle. *Ultrasonic Imaging*, 14:249–275, 1992.
- [82] K.I Laws. Textured image segmentation. *PhD dissertation, Univ. Southern California Los Angeles USCPI Rep 940*, 1980.
- [83] R. A. Lerski, D. W. McRobbie, and J. D. de Certaines. Protocols and test objects for the assessment of MRI equipment. *Magnetic Resonance Imaging*, 6:195–199, 1988.
- [84] R. A. Lerski, D. W. McRobbie, K. Straughan, P. M. Walker, J. D. de Certaines, and A. M. Bernard. Multi-center trial with protocols and prototype test objects for the assessment of MRI equipment. *Magnetic Resonance Imaging*, 6:201–214, 1988.
- [85] M. Levoy. Volume rendering: Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, pages 29–37, May 1988.
- [86] M. Levoy. A hybrid ray tracer for rendering polygon and volume data. *IEEE Computer Graphics and Applications*, 10:33–40, March 1990.
- [87] W.J Lin, S. M Pizer, and V. E. Johnson. Boundary estimation in ultrasound images. In *Information Processing in Medical Imaging, 12th International Conference, IPMI 91*, pages 285–299, 1991.
- [88] Tony Lindberg. Scale-space for discrete signals. *IEEE Transactions on Pattern analysis and Machine Intelligence*, 12:234–254, 1990.
- [89] William E. Lorensen and Harvey E. Cline. Marching cubes: a high resolution 3D surface construction algorithm. *Computer Graphics*, 21(4):163–169, July 1987.
- [90] A. Loupas. Digital image processing for noise reduction in medical ultrasonics. *Ph.D dissertation, University of Edinburgh UK*, 1988.
- [91] T. Loupas, W. N McDicken, and P. L. Allan. An adaptive weighted median filter for speckle suppression in medical ultrasonic images. *IEEE Trans on Circuits and Systems*, CAS-36(1):129–135, 1989.
- [92] G.E. Lowitz. Can a local hisotgram really map texture information. *Pattern Recognition*, 16(2):141–147, 1983.
- [93] A. Lundervold and F. Godtlielsen. Noise reduction and brain tissue classification in MR images. In *IEEE Nuclear Science Symposium and Medical Imaging Conference, Orlando, Florida*, pages 1265–1267, October 25–31 1992.

- [94] A. Lundervold and F. Godtliebsen. Tissue classification in MR images using contextual and artificial neural network classifiers. In *Proceedings from the NOBIM Conference, Trondheim*, pages 263–275, June 15-16 1992.
- [95] A. Lundervold and G. Storvik. Segmentation of brain parenchyma and cerebrospinal fluid in multispectral magnetic resonance images. Submitted to *IEEE Trans. Med. Imaging*.
- [96] Benoni Mandelbrot. *The fractal Geometry of Nature*. Freeman San Fransisco, 1982.
- [97] A. Manduca. Multi-parameter image visualization with self-organizing maps. In *SPIE Medical Imaging 94*, 1994.
- [98] H.A McCann, J.S Sharp, T.M Kinter, C.N McEwan, C. Barillot, and J.F Greenleaf. Multidimensional ultrasonic imaging for cardiology. In *Proc IEEE*, volume 76, pages 1063–1073, 1988.
- [99] Hans-Peter Meinzer, Kirsten Meetz, et al. The heidelberg raytracing model. *IEEE Computer Graphics and Applications*, pages 34–43, November 1991.
- [100] T. R. Miller, J. B Starren, and R. A. Grothe. Three dimensional display of positron emission tomography of the heart. *J. Nuclear Medicine*, 29(4):530–537, 1988.
- [101] Gaute Myklebust and Erik Steen. Neural networks in visualization of multispectral medical images. *To be published in Information Sciences*, 1996.
- [102] N. nordstrøm. Biased anisotropic diffusion—a unified regularization approach to edge detection. *Image Vision Comput.*, 8:318–327, 1990.
- [103] R. Obuchi, D. Chen, and H Fuchs. Incremental volume reconstruction and rendering for 3D ultrasound imaging. In *Visualization in Biomedical Computing*, Chapel Hill, NC, 1992.
- [104] B. Olstad. Reasoning in entropy graphs for image operators. *Pattern Recognition*, 1994.
- [105] B. Olstad, S. Berg, S. Eik-Nes, and K. P. Schipper. 3d transvaginal ultrasound imaging for identification of endometrial abnormality. In *Proceedings of SPIE Medical Imaging '95*, San Diego, 1995.
- [106] B. Olstad, E. Steen, and A. Halaas. Image filtering techniques and vlsi architectures for efficient data extraction in shell rendering. In *IEEE ICIP*, 1995.
- [107] B. Olstad, Erik Steen, and Olav Sandstå. Shell rendering with hardware supported data extraction. In *Proceedings of SPIE Medical Imaging '95*, San Diego, 1995.
- [108] B. Olstad, A. Torp, and J. H. Husøy. A nonlinear loop filter model for video coding. In *IEEE Winter Workshop On Nonlinear Digital Signal Processing*, 1993.
- [109] B. Olstad and H. E. Tysdahl. Improving the computational complexity of active contour algorithms. In *8th Scandinavian Conference on Image Analysis, Tromsø*, 1993.

- [110] Bjørn Olstad. Automatic wall motion detection in the left ventricle using ultrasonic images. In *Proceedings of SPIE/SPSE Electronic Imaging, Science and Technology*, San Jose, 1991.
- [111] Bjørn Olstad. Motion compensation in filtering of time varying images. In *Proceedings of 6th International Conference on Image Analysis and Processing*, Como, Italy, 1991.
- [112] Bjørn Olstad. Noise reduction in ultrasound images using multiple linear regression in a temporal context. In *Proceedings of SPIE/SPSE Electronic Imaging, Science and Technology*, San Jose, 1991.
- [113] Bjørn Olstad. Maximizing image variance in rendering of volumetric data sets. *Journal of Electronic Imaging*, 1:245–265, July 1992.
- [114] Bjørn Olstad and Nils Thune. Visualizing 4-d medical ultrasound data. In *Proceedings IEEE Visualization '91*, San Diego, California, October 1991.
- [115] J.S Ostrem, A.D Valdes, and P.D Edmonds. Application of neural nets to ultrasound tissue characterization. *Ultrasonic Imaging*, 13(3):298–299, July 1991.
- [116] T. Pavlidis and P. C. Chen. Segmentation by texture using a cooccurrence matrix and a split and merge algorithm. *Computer Graphics and Image Processing*, 10, 1979.
- [117] T. Pavlidis and S.L Tanimoto. Texture identification by a directed split and merge procedure. In *Proc. Conf. Comput. Graph. and Pattern Rec. Data Struct.*, 1975.
- [118] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on Pattern Anal. and Machine Intell.*, 12(7), July 1990.
- [119] Bui-Thong Phong. Illumination for computer generated pictures. *Comm. ACM*, 18:311–317, June 1975.
- [120] M. Pietikainen and A. Rosenfeld. Image segmentation by texture using pyramid node linking. *IEEE Trans Syst. Man Cybern Volume SMC-11*, pages 822–825, 1981.
- [121] I. Pitas and A.N. Venetsanopoulos. *Nonlinear digital filter - Principles and Applications*. Kluwer Academic Publishers, 1990.
- [122] R. A. Rao. *A Taxonomy for Texture Description and Identification*. Springer-Verlag, New York, 1990.
- [123] Prasas Sai Raya. Low-level segmentation of 3D magnetic resonance brain images - a rule-based system. *IEEE Transactions on medical imaging*, 9(3):327–337, September 1990.
- [124] P. Sabella. A rendering algorithm for visualizing 3D scalar fields. *Computer Graphics*, 22(4):51–55, August 1988.
- [125] P. Saint-Marc, J. Chen, and G. Medioni. Adaptive smoothing: A general tool for early vision. *IEEE Trans. on Pattern Anal. and Machine Intell.*, 13(6), June 1991.

- [126] K. Å Salvesen, L. S Bakketeig, S.H Eik-Nes, J.O Undheim, and O. Økland. Routine ultrasonography in utero and school performance at age 8-9 years. *The Lancet*, 339, jan 1992.
- [127] R. Schalkoff. *Pattern Recognition, statistical, structural and neural approaches*. John Wiley & Sons, Inc, 1992.
- [128] E. Steen, G. Myklebust, and B. Olstad. Visualization of multiattribute medical images. In *SPIE Medical Imaging 95, Image display conf.*, volume 2431. SPIE, 1995.
- [129] E. Steen and B. Olstad. Scale-space and boundary detection in ultrasonic imaging, using signal-adaptive anisotropic diffusion. In *SPIE Medical Imaging, Image processing conference*, 1994.
- [130] E. Steen and B. Olstad. Volume rendering of 3d medical ultrasound data using direct feature mapping. *IEEE Trans. on Medical Imaging*, 13(3):517–525, september 1994.
- [131] E. Steen, B. Olstad, G. Myklubust, S. Berg, and K. P Schipper. Combined visualization of tumor and vessel geometry. In *IEEE International Ultrasonics Symposium Cannes*. IEEE, 1994.
- [132] Erik Steen. Volume rendering in medical imaging. Master’s thesis, The Norwegian Institute of Technology, dec 1992.
- [133] Erik Steen and Bjørn Olstad. Volume rendering in medical ultrasound imaging. In *8th Scandinavian Conference on Image Analysis, Tromsø*, 1993.
- [134] Erik Steen and Bjørn Olstad. Volume rendering in medical ultrasound imaging based on non-linear filtering. In *IEEE Winter Workshop On Nonlinear Digital Signal Processing*, 1993.
- [135] E. Stephansen. *MS160, Data book*. MRT International, 1993.
- [136] G. Storvik. A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing. *IEEE Trans. Pattern Anal. Machine Intell.*, 16:976–986, 1994.
- [137] G. Storvik and A. Lundervold. Segmentation of brain parenchyma and CSF in multispectral MR images of the head. In *Proceedings of the 8th SCIA Conference*, pages 3–4, Tromsø, Norway, May 25-28 1993.
- [138] M. R. Stytz, G. Frieder, and O. Frieder. Three-dimensional medical imaging: Algorithms and computer systems. *ACM Computing Surveys*, 23(4):421–499, 1991.
- [139] H. Suzuki and J. Toriwaki. Knowledge-guided automatic thresholding for 3D display of head MRI images. In *Proc. from 9th International Conference on Pattern Recognition*, volume 2, pages 1210–1212, 1988.
- [140] T. Taxt. A new class of gibbs-markov random field models for image restoration. *To be published*, 1992.

- [141] T. Taxt and A. Lundervold. Multispectral analysis of the brain using magnetic resonance imaging. *IEEE Trans. on Medical Imaging*, 13(3):470–481, September 1994.
- [142] T. Taxt, A. Lundervold, B. Fuglaas, H. Lien, and V. Abeler. Multispectral analysis of uterine corpus tumors in magnetic resonance imaging. *Magnetic Resonance in Medicine*, 23(1):55–76, 1992.
- [143] T. Taxt and E. Bølviken. Relaxation using models from quantum mechanics. *Pattern Recognition*, 24:695–709, 1991.
- [144] C. W. Therrien. *Decision, Estimation, and Classification: An Introduction to Pattern Recognition and Related Topics*. John Wiley & Sons, New York, 1989.
- [145] H. Torp, S. Dørum, E. Holm, E. Steen, K. Kristoffersen, K. A. Berg, H.J. Alker, and B. Olstad. Real-time three dimensional echocardiography for assessment of valvular function. *Presented at 11th Symposium on Echocardiology*, 1995.
- [146] Hans Torp. *Signal Processing in Real-time, two-dimensional Doppler Color Flow Mapping*. PhD thesis, The Norwegian Institute of Technology, 1990.
- [147] W. G. Totty and M. W Vannier. Analysis of complex musculoskeletal anatomy using three-dimensional surface reconstruction. *Radiology*, 150:173–177, 1984.
- [148] J. K Udupa. Interactive segmentation and boundary surface formation for 3D digital images. In *Comp. Graph. Im. Proc.*, volume 18, pages 213–235, 1982.
- [149] J. K. Udupa and Dewey Odhner. Shell rendering. *IEEE CG& A.*, pages 58–67, November 1993.
- [150] J. K. Udupa and S. Samarasekara. Fuzzy connectedness and object definition. In *SPIE Medical Imaging 95, Image Display conference*, volume 2431, pages 2–10. SPIE, 1995.
- [151] Michael Unser and Murray Eden. Multiresolution feature extraction and selection for texture segmentation. *IEEE transactions on PAMI*, 11(7), July 1989.
- [152] Craig Upson and Michael Keeler. V-buffer: Visible volume rendering. *Computer Graphics*, 22(4):59–64, August 1988.
- [153] Daniel J. Valentino. Volume rendering of multimodal images: Applications to MRI and PET imaging of the human brain. *IEEE Trans. on Medical Imaging*, 10(4):554–561, 1991.
- [154] M.C Vannier, D Jordan., and R. L Butterfield. Multispectral analysis of magnetic resonance imaging. *Radiology*, 154:221–224, 1985.
- [155] M.W Vannier, S. Gronemeyer, and F.R Gutierrez et al. Three-dimensional magnetic resonance imaging of congenital heart disease. *Radiographics*, 8(5):857–871, 1988.
- [156] M.W Vannier, J. L Marsh, and J. O. Warren. Three-dimensional ct reconstruction for craniofacial surgical planning and evaluation. *Radiology*, 150:179–184, 1984.

- [157] O.T. von Ramm, S.W. Smith, and H.G. Pavy Jr. High-speed ultrasound volumetric imaging system - part i-ii: transducer design and beam steering. *IEEE Trans. on Ultrasonics, Ferroelectrics and Frequency control*, 38, March 1991.
- [158] J. W. Wallis, T. R. Miller, C. A. Lerner, and E. C Kleerup. Three-dimensional display in nuclear medicine. *IEEE Trans. on Medical Imaging*, 8(4), 1989.
- [159] S. Webb, R. J Ott, and M. A. Flower et al. Three-dimensional display of data obtained by single photon emission tomography. *Brit. J. Radiology*, 60:557–562, 1987.
- [160] K. Zeger and A. Gersho. Pseudo-gray coding. *IEEE Trans. On Communications*, 38(12), December 1990.
- [161] Guownag Zhang, X. Liu, and M.D Fox. Visualization of ventricles of the brain by volume rendering. In *Proc of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 1989.

5 A brief overview of important medical imaging modalities

In this section we give a brief overview of important medical imaging modalities. The following four subsections are mainly taken from the book "The Physics of medical imaging", [36].

5.1 X-ray imaging

8. November 1895, Wilhelm Conrad Röntgen discovered the x-rays. Only about 10 weeks later, on 13 January 1896, a radiographic image was produced for clinical purposes by two UK doctors, showing a needle in a woman's hand. Since then, x-ray imaging has become clearly one of the most important medical imaging modalities. The radiographic image is formed by interaction of x-ray photons with the human body, and is a projection of the attenuating properties of the tissue along the x-ray paths. There are two basic modes of interaction, scattering and absorption. The absorption process is exploited to form the radiographic image, while the scattering reduces the signal to noise ratio. Commonly, an anti-scatter device is placed between the patient and the image receptor. The anti-scatter device can for instance be a grid of parallel lead strips, which will transmit most of the primary radiation, but reject most of the scatter. The signal to noise ratio can also be increased by increasing the number of x-ray photons used to form the image. This will however increase the dose to the patient. It is important to reduce the radiation dose, as x-ray radiation may cause damage to the patient. The transmission of monoenergetic photons through tissue increases as the energy increases, and decreases as the thickness of the tissue increases. If the transmission is close to unity, the X-ray image will have poor contrast. A very low transmission will lead to a very high radiation dose to the tissue. In practice, the photon energy used is in the range 17-150 keV.

Conventional radiological systems are analog and use screen-film receptors. These systems offer little possibility of image processing and are not very flexible when it comes to dose reduction. Digital systems have inferior resolution to the analog systems, but this is usually not of significance in clinical applications. Digital systems provide more flexible gray level mappings, greater flexibility in reduction of dose and possibilities for digital processing. Digital systems based on image intensifiers are very fast and can be used for real time imaging. Digital subtraction angiography is a technique which is used for visualization of blood vessels. With this technique, an image of the interesting region is obtained before injection of a contrast agent. Subsequent images are then subtracted from the first one in order to emphasize the vessels. The injection of contrast agents will however involve an element of risk.

5.2 X-ray computed tomography

One serious drawback of conventional X-ray imaging is that it is impossible to distinguish between different soft tissue structures from the radiographic images. Another severe limitation is that the X-radiograph is unable to resolve spatially structures along the direction of the x-rays. *X-ray com-*

puted tomography (denoted CT) is an imaging modality which does not have these limitations. With CT, a planar slice of the body is defined and x-rays are passed through this slice. All x-rays are then parallel with the slice and contained within it. The first CT scanner of practical utility was announced by Hounsfield in 1972. The first generation CT scanners utilized a single pencil beam and a single detector which were linearly stepped across the patient to obtain a parallel projection. The gantry was then rotated to a new position and a new projection was obtained. The scatter rejection was very high with these systems, but the acquisition time was very long, typically 4 minutes for a single section. The evolution of CT scanning technology has been tremendous. Today, fifth generation devices with no moving parts are used for real time acquisition with scan times reduced to only a few milliseconds providing frozen images of rapidly moving organs like the human heart. To reconstruct a CT image, one needs to recover the two-dimensional distribution of the linear attenuation coefficient from a set of projections. There are several ways to reconstruct a CT-image. The most widely used method today is called "filtered backprojection". In practice, the reconstruction is made onto a regular array of discrete samples, called pixels. The reconstruction is then done by backprojecting each filtered projection onto the array and summing the result for each projection angle. Before display, each pixel value is usually rescaled into a "CT-number" which measures the fractional difference of a tissue's linear attenuation coefficient relative to water.

CT imaging has a lot of important clinical applications. Only a few is mentioned here. A major clinical application of CT imaging is in radiotherapy planning. Radiotherapy is a common technique for treatment of cancer. For radiotherapy to be successful, It is necessary that the tumor is given a tumoricidal dose. At the same time, normal organs surrounding the tumor limit the radiation dose that can be given. CT images are ideal for radiotherapy planning because they are obtained in the transverse plane, and they provide detailed renderings of the tumor and its surrounding organs. CT images are therefore used directly for computation of carefully tailored radiation fields.

By acquiring a set of tightly sampled parallel slices, it is possible to obtain high resolution 3D images which can be used for reconstruction of entire organs. 3D reconstruction techniques are for instance used in planning of complicated skull surgery and jaw surgery.

5.3 Radioisotope imaging

In radioisotope imaging, radionuclide-labelled agents are injected into the body. The distribution of these agents within the body is then imaged. The agents, often called radiopharmaceuticals, are designed to show physiological function of individual organs. The injection of radiopharmaceuticals generally involves an element of risk, which is a disadvantage of all the methods used for radioisotope imaging. The first imaging devices, such as the rectilinear camera, were developed in the in 1950s. Today, mainly gamma cameras are used for radioisotope imaging. Planar static imaging, known as planar scintigraphy, is the basic and a common technique in nuclear medicine. The images produced by this technique are projections of the three-dimensional distributions of the activity in the field of view. Since only the detection position is known, collimation has to be used in order to produce an image. Temporal changes in the distributions can be registered by acquiring multiple images over a period of time. This form of imaging is known as dynamic scintigraphy.

A sectional image can be obtained from a set of projections, similar to X-CT. This technique is known as emission computed tomography, ECT. There are two different variants of ECT, Single-photon emission computed tomography (SPECT) and Positron emission tomography (PET). These two techniques are usually considered as different modalities. Because of basic limitations in spatial resolution, ECT is not very suitable for determining the anatomy of an organ, but rather its physiological function.

Radioisotope imaging is used for various clinical applications. For instance, radioisotope imaging is used to study physiological parameters of the brain, like blood flow, blood volume, glucose and amino-acid metabolism. Most of these measurements require that PET imaging is used. In cardiac imaging, radioisotope imaging can also be used to measure the output of blood from the left ventricle. Skeletal diseases can be imaged by using radiopharmaceuticals that show increased bone-blood flow and bone production. Perfusion of blood and ventilation in the lungs can be measured with radioisotope imaging in order to detect lung segments with malfunctions. Studies of liver function are also conducted with radioisotope imaging.

5.4 Nuclear Magnetic Resonance imaging

Nuclear magnetic resonance (NMR) was discovered as early as in 1946 by two independent research teams. The principle of utilizing the shift in resonant frequency resulting from imposition of a magnetic field gradient was proposed in 1973. Based on this principle, Damadian and co workers published the first whole-body image in 1977. In 1978, the first commercially NMR scanner became available. Since then, NMR imaging has developed very rapidly and has become a widely used imaging technique for many medical applications. NMR scanners are capable of producing tomographic images of the body with a very good spatial resolution. NMR imaging is also known to have a very low hazard to the health.

Classical theory of magnetism is usually adequate to describe the behavior of net nuclear magnetic moment of a material. The spinning electric charge distribution of a single proton generates a magnetic moment normal to the plane of charge circulation. If an external magnetic field is applied with a magnetic flux density B_0 , the magnetic dipole moment will start rotating about B_0 with a certain angular frequency which is called the Larmor frequency. The Larmor frequency is both related to the strength of B_0 and a nucleus specific constant. At equilibrium, the net magnetic moment M of a material will be aligned with B_0 , making it impossible to measure M directly. In order to measure M , it must be tilted away from the B_0 direction, to produce a measurable component in the plane normal to B_0 , called the xy -plane. This is the basis for NMR measurements. A second external magnetic field of magnetic flux density B_1 , oriented in the xy -plane and rotating at the Larmor angular frequency can be applied as a pulse. This will move M away from B_0 through an angle which depends on the magnitude and duration of B_1 . During this process, some nuclei will absorb energy. When the pulse is gone, the nuclei will radiate this energy which then can be detected with a suitable coil. Various kinds of sequences of RF-pulses are used in order to compute different parameters. One important parameter is the T_1 relaxation time. This relaxation time is a measure of how long time it takes for the nuclei to emit the absorbed energy and can be related

to the water content of the tissue. Another parameter is the transverse or T_2 relaxation time which measures the time it takes before the magnetization in the xy -plane is lost.

Many NMR imaging methods make use of the property that the resonant frequency of protons is proportional to the applied magnetic field. In order to compute images, magnetic gradient fields are applied in addition to the fields described above. In the most simple configuration, the gradient fields are made so that only one volume element at a time have a field strength which gives a Larmor frequency equal to the given RF frequency used. This is however very inefficient, and modern scanners utilize the gradient fields far more efficiently so that acquisition times can be reduced.

An NMR system basically consists of different major parts: A strong (usually superconducting) magnet producing the main magnetic field, a set of coils generating the gradient fields, an RF system generating RF signals, a receiver and detection system and a computer with peripherals.

As with most imaging methods, NMR images may contain different artefacts. One phenomenon is ringing effects close to high contrast tissue interfaces. Respiratory motion may result in geometric artefacts during reconstruction. NMR image also contain noise which is usually modeled as additive Gaussian noise. This noise can be greatly reduced by averaging several frames, at the cost of longer acquisition time.

In addition, there are several other phenomena which may occur and produce artefacts in the images. Despite this, modern NMR machines can produce multi spectral images with excellent contrast between different tissue types. Also, contrast agents are available which can greatly increase contrast between for instance a tumor and the surrounding tissue. Dynamic studies of the contrast agents can be conducted by acquiring and co registering images over a period of time. Recent developments have also made it possible to greatly reduce acquisition time in NMR images. There are presently NMR scanners which can acquire upto 20 frames per second available at a very few hospitals in the world. These highly sophisticated instruments makes it for instance possible to acquire cardiac images in real time. The main drawbacks of NMR imaging today is that the instruments are expensive, they are not mobile and they can usually not be used if the patient uses a pacemaker or has metal in the body.

NMR imaging is used in a wide range of clinical applications. NMR imaging is often the preferred imaging technique for examinations of pathological structures in the brain. NMR imaging is also commonly used for examinations of tumors or cysts in female breasts. In addition NMR imaging is used for cardiac imaging with the aid of ECG and respiratory triggering.

6 Ultrasound imaging

The following subsection is mainly taken from [5], [36] and [130].

6.1 Two dimensional ultrasound imaging

Ultrasonic imaging based on the pulse-echo principle became possible after the development of fast electronic pulse technology during the Second World War. The first two dimensional ultrasound scan was obtained in 1952 by Wild and Reid. They also developed the first two dimensional ultrasound scanner. Diagnostics based on medical ultrasound images has evolved dramatically during the past fifteen years. Technical breakthroughs in the development of new ultrasonic instrumentation has made this evolution possible. The image quality is continuously improving because of new probe technology and new signal processing algorithms.

6.1.1 Physical aspects of 2D ultrasound imaging

Ultrasound is defined as sound with a frequency above the audible range human hearing (approx. 20 kHz). Ultrasound transducers that transmit and receive sound waves in the range 2-40 MHz are used for clinical purposes. As the transmitted ultrasound waves propagate through a medium, the medium will be compressed and decompressed along the propagation direction. These kind of waves are called longitudinal waves. There may also be transverse waves, but they are heavily attenuated in biological material. The sound velocity is about 1540-1580 m/s in biological tissue (which is mainly water), except for fat where the velocity is approximately 1440 m/s. Ultrasound scanners are usually designed with the assumption that the speed of sound is 1540 m/s. The scattering of ultrasonic waves is caused by spatial variations in the acoustic impedance in the human body. The waves can be divided into specular reflections and diffuse scattering (volume scattering). Specular reflections appear between layers of tissues with different acoustic impedance when the boundary is smooth. The specular reflections are however only captured by the transducer from places where the layers are approximately normal to the radial direction. This is a direct consequence of the reflection laws. In an ultrasonic image, specular reflections appear as bright boundaries which are blurred according to the point spread function. High frequency spatial variations in the acoustic impedance give rise to diffuse scattering characterized by the backscatter coefficient per volume.

Ultrasound is also used to measure blood velocities, by exploiting the Doppler effect. By pulsing the beam, blood velocities in a localized region can be measured. If a sound pulse is reflected from a scatterer which is moving towards the transducer, there will be a positive shift in the frequency in the received pulse. Similarly, if a sound pulse is reflected from a scatterer which is moving away from the transducer, there will be a negative shift in frequency. The speed of which the scatterer is moving along the radial direction can be directly calculated from the measured frequency shift. A continuously transmitted wave is used for measuring the highest velocities, at the cost of reduced radial resolution. More recently, the Doppler effect has also been exploited to measure the movement of myocardial tissue in cardiac imaging.

There are many physical factors that limit the usability of ultrasound scanners. The speed of sound limits the amount of data which can be acquired per second. However, with modern ultrasound scanners, it is possible to increase acquisition speed by acquiring two or more beams simultaneously. Modern ultrasound scanners are capable of generating up to hundred images in a second

and even several hundred images if the number of beams in an image is reduced. This makes it possible to produce detailed renderings of fast moving organs like the valves in a human heart.

Due to the coherent detection, constructive and destructive interference between neighboring scatterers give rise to multiplicative noise, speckle, which modulates the gray level image, and often make the images hard to interpret. It is however possible to reduce speckle by using various analog and digital filtering techniques. There are also many other acoustical phenomena, such as multiple reflections between tissue layers and phase aberrations which may degrade the ultrasonic image. Some of these effects are very difficult to cope with

Ultrasound is attenuated by the tissue so that the intensity of a wave propagating through tissue decreases exponentially with the distance to the transmitter. Attenuation increases with the frequency of the sound. A high frequency probe can be used to obtain a high resolution image, at the cost of limited penetration into the tissue. Often, some part of the tissue may totally reflect or attenuate the ultrasound, leaving tissue farther from the transducer in the shadow. In some cases this effect can be reduced by using a technique called spatial compounding. With spatial compounding, ultrasound images which are acquired from different angles are mixed together. This technique also reduces speckle.

6.1.2 Ultrasound transducers

An ultrasonic transducer can be made from a thin plate of piezoelectric material with metal electrodes on each face. An oscillating voltage source can be coupled to the electrodes causing the plate to vibrate. The transducer will vibrate at resonance if the plate thickness is one half of a wave length. If the vibrating plate is in contact with biological tissue (or a fluid), a sound wave will be generated. By Huygen's principle, an ultrasonic beam is formed by the interference between spherical waves from each point on the transducer surface. The beam has a characteristic farfield region where it expands with a defined opening angle and a nearfield region which can be further divided into an extreme nearfield region and a transition region. In the extreme nearfield, the beam is a cylindrical extension of the transducer. The beam can be focussed by forming the transducer as part of a spherical shell. Electronically controlled beam forming can be obtained by building a transducer of an array of small elements each radiating a wide beam. The partial waves will interfere and form a resulting beam which can be steered and focussed electronically. There basically four different types of transducers that are in clinical use. Phased arrays, Linear arrays, CurveLinear arrays and annular arrays. Annular array probes are steered mechanically. The advantage of annular arrays is that the focus is symmetric, which increases the resolution normal to the scan plane. This may for instance be a benefit for three dimensional imaging.

6.1.3 Clinical applications of two dimensional ultrasound imaging

Ultrasound imaging has a wide range of clinical applications. Ultrasound imaging is widely used in obstetrics. In Norway, all pregnant women are offered an ultrasound examination in the 16th week

of their pregnancy. During such an examination, the midwife performs standardized measurements to determine fetal maturity, looks for abnormalities and studies the fetal activity.

In gynecology, ultrasound imaging is routinely used to determine size and shape of uterus and cervix and for diagnosis of tumors in the uterus. Ultrasound imaging is also used in examinations of the ovaries. Ultrasound imaging is also widely used in the upper abdomen for assessment of liver size and shape, examinations of liver tumors, detection of gall stones and kidney stones, examination of spleen and examination of pancreas.

Ultrasound imaging is widely used for examination of the cardiovascular system. A modern real time ultrasound scanner can be used to visualize the movement of the heart ventricles, the mitral valves and the aorta valves. The Doppler effect is exploited in order to visualize blood flow. This makes it possible to detect valve leakage.

2D ultrasound imaging is also used during many other clinical examinations in addition to the ones mentioned above and also during different kinds of surgery. The ultrasound instrumentation is faster, simpler and cheaper to use than for instance a computer tomograph. Also, the risk of hazard during an examination is very low. It has not been found any evidence that ordinary ultrasound examinations may cause damages to the patient [126].

6.2 Multidimensional medical ultrasound imaging

Three dimensional (3D) ultrasound imaging was suggested as early as in the 1950s, but then there was a lack of adequate computer equipment. Recent development in computer technology has made it possible to implement 3D and 4D instrumentation.

6.2.1 Benefits of 3D ultrasound imaging

In 2D ultrasound imaging, the operator must mentally construct a model of the organ he is examining by looking at many 2D images from different positions in sequence. This may be difficult in many cases, especially if the shape of the organ is abnormal due to a clinical condition. Very often, the operator want to estimate the volume of an organ. This is usually done from measurements of the diameter of the organ in one or several cross sections. The estimates may therefore be imprecise, especially if the organ has a complex shape. In 3D ultrasound imaging, it is possible to overcome many limitations with conventional 2D imaging. By using advanced visualization techniques, it is possible to generate images that render whole organs and not just cross sections. There are also techniques that can be used to build geometric models of the organs so that precise volume estimates can be made.

6.2.2 Acquisition of multidimensional medical images

Most of the existing solutions for 3D ultrasound acquisition are based on electro-mechanical devices which systematically change the orientation of the scan plane for each generated image. The acquired data is then often transferred to an external PC or workstation for further processing. Another approach to 3D acquisitions is to use a position censoring device which is mounted to an ordinary 2D probe. This is a very flexible way of acquiring 3D data. A normal 3D acquisition may take up to several seconds, depending on the size of the organ which is to be imaged. If the movement of the organ during this period can not be neglected, the acquisition is much more cumbersome. It is possible to acquire 3D data from a human heart by using ECG to synchronize the acquisition with the heart cycle of the patient. In addition, since the heart is moved during the respiration cycle, respiration must be monitored in order to validate each heart cycle. A four dimensional data set is obtained by acquiring a full cine loop in each scan plane. At present there is a big need to make such acquisitions less time consuming and easier to perform.

Visualization of 3D ultrasound data

A 3D data set is often represented as a regular array of small volume elements which are called "voxels". These are analog to pixels in 2D images. A scalar value or a vector value is associated with each voxel. There is a wide range of algorithms which have been developed for visualization of 3D data (see for instance [22, 152, 85, 113].) These methods generate two dimensional representations of three dimensional scenes. Several visualization algorithms make use of a technique which is called "ray casting". In this technique, an image is constructed in a view plane which is thought to be outside the space limited 3D scene. Each pixel is associated with a ray through the 3D scene. The value of the pixel is calculated as a function of all voxels through which the ray passes.

A very simple visualization method is to map the average value of all the voxels along each ray into their corresponding pixels. This method is analog to conventional X-ray imaging, where the intensity in each pixel depends on the total amount of light absorbed along each ray.

Surface shading techniques are often used in medical imaging applications. These techniques are based on either automatic, semi-automatic or manual detection of the surfaces. The detected surfaces are visualized by surface shading. Surface shading algorithms model light reflections from surfaces towards the viewer and can give a very good understanding of object geometries. Surface shading techniques can give misleading information if the detection is not reliable. In ultrasonic imaging, surfaces can be particularly hard to detect, due to different noise phenomena.

Much clinical attention has been paid to the problem of visualizing the surfaces of the heart valves. These surfaces may often be poorly defined in normal images acquired with Trans Thoracic probes. However, with a multiplane TEE probe (which is swallowed by the patient) it is possible to get closer to the target and a much better image quality can be achieved. An example of a special surface shading technique used on multiplane TEE data is illustrated in figure 1. The data was acquired

with the Tomtec Echoscan System and a Vingmed CFM-800 ultrasound scanner using ECG and respiration triggering and then transferred to an external workstation for further processing. This image is a rendering of a mitral valve with hole in it. Below the mitral valve, one can see the aortic valve which is open. This image is much more informative when displayed in a cineloop covering the whole heart cycle. The special shading technique used is called "Z-buffer shading" and it is described in [131].

Volume rendering techniques have become very popular for medical applications. Volume rendering techniques are "fuzzy" visualization methods that may be appropriate if the transitions between different tissues are diffuse. In many volume rendering algorithms, opacity values are assigned to each voxel. The opacity values define how much each voxel will contribute to the final image. Gradient shading is a fuzzy surface rendering technique in which surfaces are implicitly defined by computing gradients in each voxel. Figure 2 illustrates a volume rendering algorithm where the opacity values are computed on the basis of gradient magnitude values [130]. The rendering is a transparent view of the hands of a 30 week old fetus. It is possible to see small bone structure within the fetus hands. This kind of rendering technique is much more useful if it is possible to adjust the viewing direction interactively. Interactive rendering can be achieved by visualizing only a set of "interesting" voxels which is computed once for a given scene [149].

Ongoing research

3D and 4D medical ultrasound imaging is still in its childhood. Much research remains in order to exploit the possibilities better. A great challenge is to develop real-time 3D acquisition (4D acquisition) as a clinically useful modality. There are several groups working on 2D array probes [157]. There are still many theoretical and practical problems which has to be solved in order to develop clinically usable 2D probes. The requirement for computing power is tremendous in order to process data from 2D arrays in real-time. The number of connections which has to be made to the array elements may also be extremely high.

Currently, the author is involved in a research project that aims at developing real time 3D acquisition based on fast rotation of linear, curve-linear and phased array probes [145]. Several experiments have been conducted with a prototype system. The System Five ultrasound scanner from Vingmed Sound was used in these experiments. The System Five scanner is capable of scanning 2D sectors with a frame-rate exceeding 500 frames per second. This is achieved by reducing the number of beams in each frame and by use of multiple line acquisition. A probe adapter was mounted to a phased array probe and by use of a electro-motor, we were able to capture up to 20 small volumes per second with 10-15 scan planes per volume. Several acquisitions were made of the mitral valve of a healthy male person. The data was reconstructed off-line on a workstation and visualized with various rendering algorithms. Examples are shown in 3 and 4.

The interface between the computer and the user is a vital part of a 3D acquisition system. A general problem is orientation. To fully understand rendered images of a 3D scene, the user needs an intuitive control of the orientation of the view plane. This is difficult to achieve with common

pointer-devices such as a mouse or a joystick. GMD is a German research institute which has developed an intuitive user interface for orientation. A position tracking system with a small receiver mounted on a “dummy” ultrasound probe is used. The system is used to control the orientation of a cross-section through a model of the human heart. The user gain interactive control of the orientation of the cross section by altering the orientation of the probe. Such an interface may also be used for orientation after acquisition of real ultrasonic data.



Figure 1: Surface rendering of the mitral and aorta valves



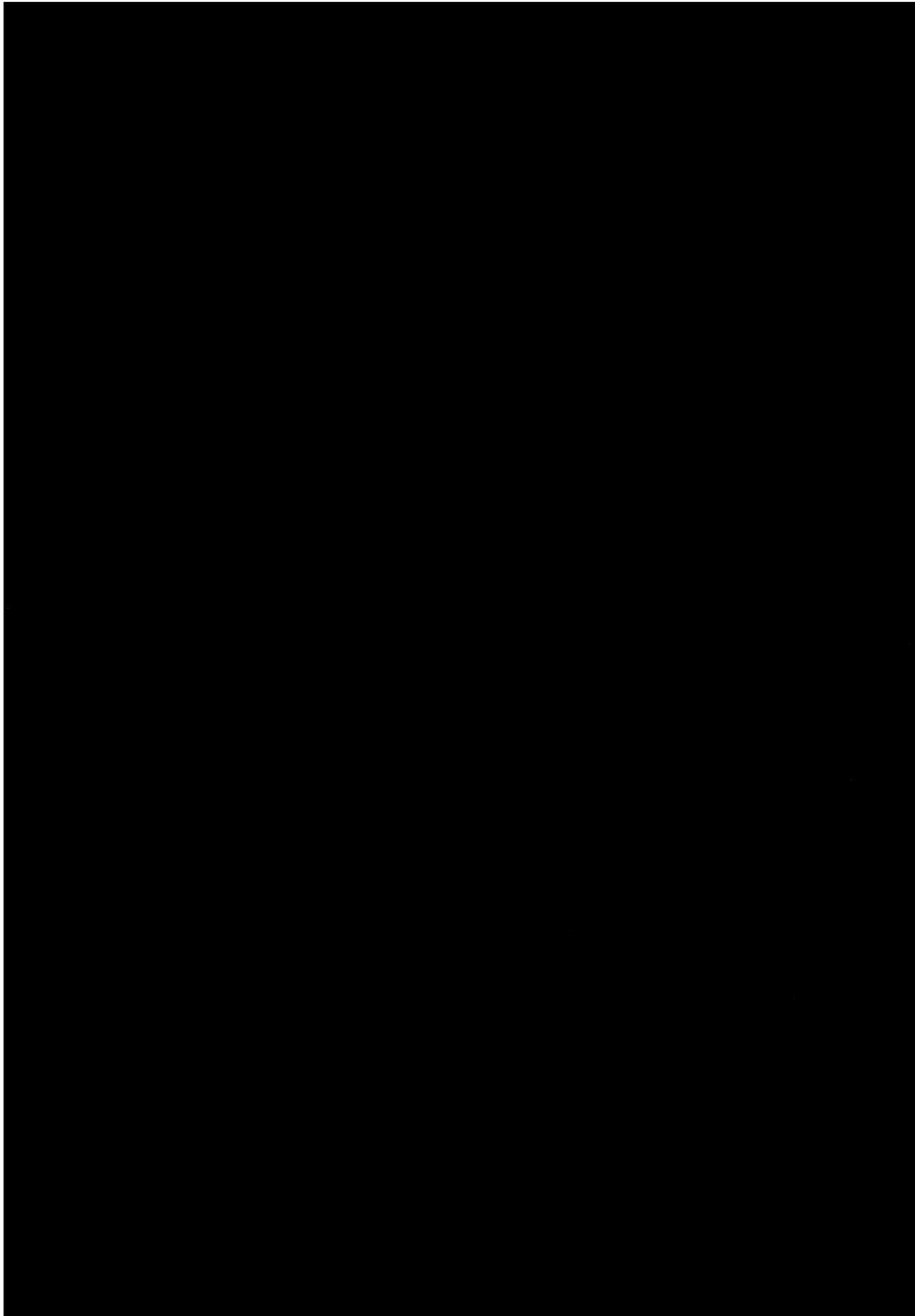
Figure 2: Transparent rendering of fetus hand and aorta valves



Figure 3: Cross sections of volumetric ultrasound data acquired in real time



Figure 4: Surface rendering of (part of) mitral valve



Scale-space and boundary detection in ultrasonic imaging using nonlinear signal-adaptive anisotropic diffusion

Abstract

In this paper we develop a strategy for scale-space filtering and boundary detection in medical ultrasonic imaging. The strategy integrates a signal model for displayed ultrasonic images with nonlinear anisotropic diffusion. The usefulness of the strategy is demonstrated for applications in volume rendering and automatic contour detection. The discrete implementation of anisotropic diffusion is based on a minimal nonlinear basis filter which is iterated on the input image. The filtering scheme involves selection of a threshold parameter which defines the overall noise level and the magnitude of gradients to be preserved. In displayed ultrasonic images the speckle noise is assumed to be signal dependent, and we have therefore developed a scheme which adaptively adjusts the threshold parameter as a function of the local signal level. The anisotropic diffusion process tends to introduce artificial image details due to edge enhancement. Another modification has therefore been made to avoid edge-enhancement by leaving significant monotone sections unaltered. We argue that this preservation of inaccuracies is important for applications such as volume rendering which is a fuzzy display technique. The proposed filtering strategy is evaluated both for synthetic images and real ultrasonic images. The results are very promising. The filtering scheme has been implemented in a complete rendering pipeline which now serves as a clinical tool at the National Center of Fetal Medicine in Norway.

1 Introduction

The boundaries in ultrasonic images carry structural information which is vital for diagnostic purposes [8]. It is therefore important in many clinical applications that the the location of the boundaries is preserved also at coarse resolutions. Boundary detection is particularly difficult in ultrasonic imaging because of high frequency speckle, acoustic reverberations and several other limitations in the imaging system. The scattering of ultrasonic waves is caused by spatial variations in the acoustic impedance in the human body. The waves can be divided into specular reflections and diffuse scattering (volume scattering). Specular reflections appear between layers of tissues with different acoustic impedance when the boundary is smooth. The specular reflections are however only captured by the transducer if the layers are approximately normal to the radial direction.

High frequency spatial variations in the acoustic impedance give rise to volume scattering characterized by the backscatter coefficient per volume. The volume scattering will therefore differ between different kinds of tissue and will cause gray level differences in the displayed image. This makes it possible to detect the boundaries between tissues by computing local gradient estimates. Due to the coherent detection, constructive and destructive interference between neighboring scatterers give rise to multiplicative noise, speckle, which modulates the gray level image. It is therefore essential that the gradient estimates are made as insensitive to speckle as possible. A gradient operator may however produce artificial double edges from single specular reflections. In this paper, we have assumed that these specular reflections are not prominent in the displayed images.

Several authors have proposed strategies for boundary detection in images corrupted with speckle or more specifically in ultrasonic imaging. Perhaps the best known approach to boundary detection in speckled images was presented by A. Bovik [2]. This approach was based on Laplacian-of-Gaussian (LOG) detection combined with Ratio-of-average (ROA) estimation. The idea was then to use a LOG based scheme to detect all potential edges. The ROA estimate was then used to filter out spurious edges detected with the LOG based scheme. LOG based schemes such as the Canny operator [3] do however not handle multiple interfering boundaries well, and the location of boundaries is accurate only for small window sizes [16]. A probabilistic approach to edge detection in ultrasound images was suggested by W.J Lin et al [9], but failed to locate the edges accurately.

Another approach to edge detection is to apply edge preserving smoothing prior to detection with a simple edge operator like Roberts or Sobel. Such a filtering scheme must satisfy the conflicting requirements of maximum noise suppression and preservation of edges and other image details, in order to be useful in medical applications [8]. An example of an edge preserving smoothing process is the anisotropic diffusion which was first introduced as an image processing tool by Perona and Malik [16]. This smoothing process can be thought of as a kind of scale-space filtering which smoothes within homogeneous image regions while preserving the boundaries. Several authors [18, 4, 22, 21], have presented filtering strategies based on essentially the same computational framework. anisotropic diffusion was used for noise suppression in MR-imaging by Gerig et. al [5]. Preliminary results on boundary detection in ultrasonic images with anisotropic diffusion has been presented by Steen and Olstad [20, 19]. In this paper we propose a strategy which integrates anisotropic diffusion with a signal model which was first presented by A. Loupas [11] for displayed

ultrasonic images.

The anisotropic diffusion scheme proposed by Perona and Malik tends to produce artificially sharp boundaries and artificial boundary corners which do not reflect on the diffuse transitions and smooth curves often found in real-world applications. Another modification has therefore been made to preserve significant monotone sections.

Boundary detection is important in applications such as volume rendering and automatic contour detection. As we will point out later, the proposed boundary detection scheme has several properties which are particularly useful in clinical applications.

This paper is organized as follows:

Section 2 reviews the anisotropic diffusion as it was presented by Perona and Malik [16] with some additional comments. In section 3 we develop our modified scheme which is specially adapted to filtering of ultrasonic images. In section 4 we discuss issues related to software and hardware implementation of the proposed strategy, and we present several numerical experiments. Finally, in section 5, we demonstrate the usefulness of the proposed strategy in applications to volume rendering and automatic contour detection.

2 Anisotropic diffusion

Perona and Malik has proposed a strategy for scale-space filtering and edge detection which is based on the anisotropic diffusion equation [16]. They criticized the standard scalespace paradigm, where only space invariant blurring is allowed. The “causality” criterion originally proposed by J. Koendrink [7] says that no spurious details should be generated when passing from finer to coarser resolutions. Mathematically, this means that no local maxima or minima should appear when passing from a finer to a coarse resolution. In addition, Perona and Malik stated that region boundaries should be sharp and coincide with the “semantically” meaningful boundaries at each resolution, and that intra region smoothing should be preferred to inter-region smoothing at all scales.

The anisotropic diffusion equation is:

$$I_t = \text{div}(c(x, y, t)\nabla I) \tag{1}$$

where I_t represents the image, and $c(x, y, t)$ is a conduction coefficient. Perona and Malik defined t as the scale-space parameter, thus larger values of t correspond to coarser resolution of the image.

It was proved by Perona and Malik that the causality criterion is satisfied by selecting $c(x, y, t)$ as a monotonically decreasing function $f(\cdot)$ (the weight function) of the image gradient with $1 \geq f(I_x(x, y, t)) \geq 0$. The proof is however only correct for $1D$ signals. It is easy to prove that the *isotropic* diffusion as presented by J. Koendrink [7] satisfy the causality criterion in the case of $1D$ signals. In the $2D$ case, local maxima (or minima) can in fact appear when passing from a fine to a coarser resolution [1, 10]. The following example is taken from Lifshitz and Pizer [1]:

Imagine a $2D$ image function consisting of two separate smooth and wide hills with slightly different heights. Assume that the two hills are connected by a narrow sloping ridge without any local extrema. The lower hill will then not be a local maximum. While smoothing this geometry with the isotropic diffusion, the ridge will erode much faster than the hills, such that the lower hill will become a local maximum.

Since the *isotropic* diffusion is a special case of the *anisotropic* diffusion, this result is also valid for anisotropic diffusion. An alternative but weaker $2D$ formulation of the causality criterion is:

The value of a local maximum on the gray-level image should not increase when passing from finer to coarser resolutions, similarly, the value of a local minimum should not decrease when passing from finer to coarser resolutions.

It can easily be shown that anisotropic diffusion satisfies the modified causality criterion. This criterion is similar to $2D$ scale-space formulations which were presented by Koendrink [7] and Lindberg [10].

Consider a step edge convolved with a Gaussian and let $\phi(I_x) = f(I_x) \cdot I_x$ denote the flux function. It was shown in [16] that if $\phi'(I_x) > 0$, the blurred edge will be *enhanced*, while if $\phi'(I_x) < 0$ the edge will be even more blurred.

Several weight functions have been proposed in the literature [16, 18, 20, 22, 21, 4], all for computational schemes which can be shown to be special cases of the discrete version of the anisotropic diffusion.

The following function has been proposed by several authors [16, 18, 4]:

$$f(I_x) = e^{-\frac{I_x^2}{2\sigma^2}} \quad (2)$$

The parameter σ determines the noise level which then is supposed to be space invariant. From equation 2 we find that $\phi'(I_x) = e^{-\frac{I_x^2}{2\sigma^2}} [1 - \frac{I_x^2}{\sigma^2}]$ and blurring will occur for $I_x < \sigma$ while $I_x > \sigma$ will lead to edge-enhancement.

Saint Marc et. al [18] developed an algorithm called iterative smoothing, which showed out to be an implementation of anisotropic diffusion. They also applied their algorithm to the first derivative of the image, generating patches of constant slope. Saint Marc et. al let the number of iterations be fixed, and σ was then used as the scale-space parameter.

A somewhat different weight-function was also suggested by Perona and Malik [16]

$$f(I_x) = \frac{1}{1 + (\frac{I_x}{\sigma})^{1+\alpha}}, \quad \alpha > 0 \quad (3)$$

By using this weight-function, small regions will be blurred before large ones.

It was argued by Nordstrøm [12] that in the diffusion process, the image will converge to a constant at the limit of infinite time. At some stage of the process, the edges will be very sharp, and

the image will be segmented into regions with nearly constant intensity. However, in a numerical implementation, the flow will be zero for sufficiently large gradients due to limited precision in the computation of the weight-functions, and the image may actually converge into several regions of constant intensity. Too guarantee that the algorithm converges within a limited number of steps, a regularization term may be added [12]. Nordstrøm called this modified scheme, biased anisotropic diffusion. Biased anisotropic diffusion will however lead to less noise smooting.

Anisotropic diffusion has been applied to medical images for noise suppression and boundary detection [5, 19, 20]

3 Signal Adaptive Anisotropic diffusion

It has been found by experimental measurements [11, 8] that *displayed* ultrasonic images can be modeled as corrupted with signal-dependent noise of the form:

$$I_0 = s + s^{1/2}n \quad (4)$$

where s is the original signal, I_0 is the observed signal and n is a zero-mean Gaussian variable with standard deviation σ_n . A more accurate model may be developed for a specific ultrasound scanner by taking specific nonlinearities in the signal processing chain into consideration. A filtering scheme based on a weight-function such as the one in eq. 2 will not work properly if we assume the signal model above, because the noise level varies within the image. Particularly, we both want to detect weak edges in regions where the signal-level is low, and stronger edges in regions where the signal level is high. We have therefore modified the algorithm by computing σ from local estimates of the signal-level \hat{s} :

$$\hat{\sigma} = \sigma_n \hat{s}^{1/2} \quad (5)$$

Now, substituting for $\hat{\sigma}$:

$$\phi(I_x) = e^{-\frac{I_x^2}{2\sigma_n^2 \hat{s}}} I_x \quad (6)$$

And differentiating:

$$\phi'(I_x) = e^{-\frac{I_x^2}{2\sigma_n^2 \hat{s}}} \left[1 - \frac{I_x^2}{\sigma_n^2 \hat{s}} \right] \quad (7)$$

Then, blurring will occur for $I_x < \sigma_n \hat{s}^{1/2}$ and edge enhancement for $I_x > \sigma_n \hat{s}^{1/2}$.

An implementation of the proposed scheme relies on the estimate of the local signal level \hat{s} . A simple solution is to use the *previous* output of the filter I_t as an approximation of the signal level. Because of the intra-region smoothing, the filtered value will gradually give a better estimate of the

signal level in the surrounding region as the filter is iterated. Initially, we may need a more robust estimate of the signal level. We will discuss how this can be achieved in the next section.

In displayed ultrasonic images, the speckle patterns may be large compared to the pixel resolution. With conventional smoothing filters, like the average filter and the median filter, the size of the filtering window is usually adjusted according to the point spread function. Anisotropic diffusion can only be implemented properly using minimal filtering windows. It is therefore very important that the original (isotropic) resolution of the image is adjusted so that the size of the minimal filtering window is approximately equal to the point spread function. Otherwise, the anisotropic diffusion filter may lead to spuriously detected edges.

The edge-enhancement property of the anisotropic diffusion may be desirable if the underlying true image consists of tissue regions which are separated by step edges. In medical applications, transitions between different tissues may be diffuse. Even if some transitions may be very sharp, a combination of noise, blurring and phase effects make a precise restoration of the corresponding edges very difficult and sometimes impossible without incorporation of very specific knowledge about the imaged object. We therefore argue that the edge enhancement property of the anisotropic diffusion is an undesired effect of the filtering in many applications such as volume rendering, which is a fuzzy display technique. More sophisticated filtering schemes, such as the one proposed in [6] may be used if the goal is to reconstruct image details which have been significantly blurred by the point spread function.

As was pointed out by Steen and Olstad [20, 14], the edge-enhancement property can be avoided by selecting the flux-function as a constant. Then any monotone section will remain unchanged. The corresponding weight-function is C/I_x , where C is a constant. This weight function should be selected only if $I_x > \sigma_n \hat{s}^{1/2}$, while a smoothing weight-function can be selected for $I_x \leq \sigma_n \hat{s}^{1/2}$. In this way, we permit intra-region smoothing while avoiding edge-enhancement. The constant C is then calculated according to the specific smoothing function to ensure a continuous weight-function. A modified version of the weight-function in eq. 2 will then be:

$$f(I_x) = \begin{cases} e^{-\frac{I_x^2}{2\hat{\sigma}^2}} & \text{if } I_x \leq \hat{\sigma} \\ \frac{e^{-\frac{1}{2}\hat{\sigma}}}{I_x} & \text{otherwise} \end{cases} \quad (8)$$

We note that $1 \geq f(I_x(x, y, t)) \geq 0$ also for this weight-function. The corresponding flux-function is illustrated in figure 1

4 Numerical experiments

4.1 Discrete implementation

The anisotropic diffusion equation can be discretized on a lattice-structure with brightness values associated to the vertices, and conduction coefficients to the arcs [16]. We have used the same

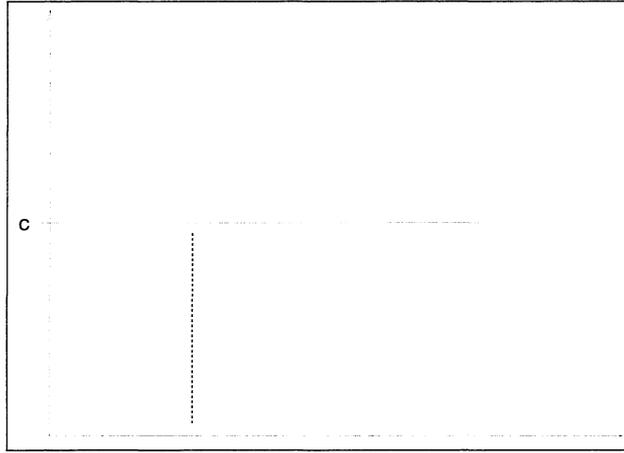


Figure 1: Proposed flux function

discretizations as proposed by Perona and Malik [16]. A 4- or 8- nearest neighbor discretization of the Laplacian operator is used, and the norm of the gradients is approximated with the absolute value of their projections along the direction of the arcs. This gives the following simple and stable numerical scheme (using 4 nearest neighbors to the north, east, south and west of the center-pixel):

$$\begin{aligned}
 I_{t+1}(x, y) &= I_t(x, y) \\
 + \frac{1}{4}[f(|\nabla_N I_t(x, y)|) \cdot \nabla_N I_t(x, y) & \\
 + f(|\nabla_E I_t(x, y)|) \cdot \nabla_E I_t(x, y) & \\
 + f(|\nabla_S I_t(x, y)|) \cdot \nabla_S I_t(x, y) & \\
 + f(|\nabla_W I_t(x, y)|) \cdot \nabla_W I_t(x, y)] &
 \end{aligned} \tag{9}$$

where ∇ denotes nearest-neighbor differences:

$$\begin{aligned}
 \nabla_N I_t(x, y) &= I_t(x, y - 1) - I_t(x, y) \\
 \nabla_E I_t(x, y) &= I_t(x + 1, y) - I_t(x, y) \\
 \nabla_S I_t(x, y) &= I_t(x, y + 1) - I_t(x, y) \\
 \nabla_W I_t(x, y) &= I_t(x - 1, y) - I_t(x, y)
 \end{aligned} \tag{10}$$

Iterating this scheme can be thought of as moving towards coarser resolutions in scale-space. An implementation based on an 8-point neighborhood that takes the distance between the neighbor-pixels into account is recommended for better isotropy.

A numerical implementation of the weight-function proposed in equation 8 requires some thought in order to keep the numerical process stable and to avoid blurring across boundaries. Significant monotone sections can be filtered properly by choosing the weight function $f(I_x) = \frac{\epsilon - \frac{1}{2}\sigma}{I_x} =$

C/I_x symmetrically around the center-pixel. Assume that in a 2-D situation we have $I_t(x, y) = \alpha$, $I_t(x - 1, y) = \beta$ and $I_t(x + 1, y) = -\gamma$, where $\alpha, \beta, \gamma \geq 0$.

We then see that:

$$\begin{aligned} & f(|\nabla_E I_t(x, y)|) \cdot \nabla_E I_t(x, y) + f(|\nabla_W I_t(x, y)|) \cdot \nabla_W I_t(x, y) \\ &= \frac{C}{(\beta - \alpha)} \cdot (\beta - \alpha) + \frac{C}{-(\beta + \gamma)} \cdot (\beta + \gamma) \\ &= 0 \end{aligned} \quad (11)$$

Hence, the two flow-contributions cancel out, and can thus be ignored.

The constant flux-function should then be selected (by ignoring the flow-contributions) symmetrically if *either* $|\nabla_E I_t(x, y)| \geq \hat{\sigma}$ or $|\nabla_W I_t(x, y)| \geq \hat{\sigma}$. and $\nabla_E I_t(x, y) \cdot \nabla_W I_t(x, y) < 0$. This test ensures that the flow is ignored only in significant *monotone* sections. A similar test can be done for pixels lying north and south to the center pixel.

As was pointed out in the previous section, we need an initial estimate of the local signal level. The signal level can then be estimated from local statistics in a *minimal* window $W_{x,y}$ surrounding a pixel with coordinates (x, y) . The window should be kept minimal to minimize the probability of sampling from different regions. We have then used the ML- estimate [8] of the original signal, given the signal model in eq. 4 with a window of size 3×3 .

$$s_{ML} = \frac{-\sigma_n^2}{2} + \sqrt{\frac{\sigma_n^4}{4} + \frac{1}{9} \sum I(i, j)^2}, (i, j) \in W_{x,y} \quad (12)$$

The implementation scheme presented in this subsection was mainly selected due to computational simplicity. In a software implementation, extensive use of look-up tables can dramatically speed up the computations. In fact the total flux in any direction can be pre-computed for all possible combinations of three consecutive discrete samples. For a four-neighborhood implementation, computation of each new pixel value will amount to two look-up operations and two summations. The scheme is also parallel in nature and well suited for implementation on dedicated hardware with several simple processing elements. Saint Marc et. al [18] implemented the anisotropic diffusion on the connection machine, gaining a dramatic speed up. They also presented a multi-grid implementation of anisotropic diffusion.

4.2 Experimental results, synthetic images

We have applied the proposed boundary detection strategy both to synthetic images and to real ultrasonic images. All experiments were done with the Application Visualization System (AVS) from Advanced Visual Systems. Initially, we constructed an experimental setup with various synthetic images to which we could add signal dependent white noise using the signal model in equation 4. The purpose of these initial experiments was to verify the theoretical properties of the proposed boundary detection scheme. In our first experiment, we constructed an artificial image with two "background regions" with different signal levels (pixel values were set to 25 and 100). In each

background region there were several smaller regions with higher intensity (pixel values were set to 50 and 175). We then added signal dependent Gaussian noise with $\sigma_n = 1.25$. Figure 2 is a topographic visualization of the test image. In figure 3 shows the test image with added noise. The proposed boundary detection scheme was applied to the test image with the parameter $\sigma_n = 1.25$. Figure 4 is a topographic visualization of the image after 25 iterations of the smoothing scheme using an 8-point neighborhood. Note how well the boundaries between the different regions are preserved, while spurious edges are almost completely removed.

In the second experiment we blurred the test image with a 5×5 Gaussian mask, and we then added signal-dependent noise with $\sigma_n = 0.8$. We applied both the proposed strategy with the weight function in eq. 8 and a signal adaptive version of the weight-function in eq. 2:

$$f(I_x) = e^{-\frac{I_x^2}{2\sigma^2}} \quad (13)$$

The results are shown in figure 5 and 6. From figure 5 we see that the edge-enhancement property of the weight function in eq. 13 in some places have lead to incorrect boundary locations and artificial boundary corners. Figure 7 shows edges detected with anisotropic diffusion using the weight-function in 13 and figure 8 shows edges detected with the scheme proposed in this paper.

4.2.1 Quantitative evaluation of filtering scheme

The smoothing algorithm presented in this paper can be viewed as a simple restoration scheme. It is therefore desirable to measure the quality of the filtered images. A common practice is to use the SNR value as a quality measure for restored images. The SNR value does not emphasize on edge localization, and is therefore a poor measure of image quality when it comes to localization of edges. Olstad [13] has proposed a quality measure which does not have this defficiency. The quality measure is developed on an information theoretic basis and is defined as an estimate of how well thresholded versions of the orginal image can be approximated by thresholding the restored image. Let $g \in Re$ denote the image intensity fuction and T be a given threshold. In Olstads framework, a thresholded image is considered as a binary partition $\xi_g^T \xi = \{X_1, X_2\}$. x_i^T is defined as follows:

$$\xi_g^T = \{ \{g < T\}, \{g \geq T\} \} \quad (14)$$

From basic information theory we know that entropy is a measure of information content in a given partition. Entropy is defined as follows:

Definition 1 (Entropy) *The entropy of a partition $\xi = \{X_1, \dots, X_n\}$ is the nonnegative value :*

$$H(\xi) = - \sum_{i=1}^n \mu(X_i) \log \mu(X_i).$$

μ is a probability measure.

Conditional entropy records the information given in a partition, ξ , if another partition, η , is already known:

Definition 2 (Conditional entropy) *The conditional entropy of ξ given η is :*

$$H(\xi/\eta) = - \sum_{i,j} \mu(X_i \cap Y_j) \log \frac{\mu(X_i \cap Y_j)}{\mu(Y_j)},$$

where the terms with $\mu(Y_j) = 0$ are omitted from the sum.

Let g denote the image computed from the undisturbed original image, and \tilde{g} the restored image computed from a corrupted version of the original image.

$$e(g, \tilde{g}) = \sup_{T_g} \inf_{T_{\tilde{g}}} H(\xi_g^{T_g} / \xi_{\tilde{g}}^{T_{\tilde{g}}}) \quad (15)$$

This measure computes the information in the stochastic experiment associated with knowing the result of a fixed threshold T_g of g , given that the thresholded images $T_{\tilde{g}}$ of \tilde{g} are known. The infimum operator assures that the best possible threshold value of \tilde{g} for guessing the result of g is selected. The supremum operator then finally selects the g -threshold that is most difficult to estimate in this manner. If $e(g, \tilde{g}) = 0$, all the information in the original image can be recovered.

To give a quantitative evaluation of the proposed boundary detection scheme, we wanted to measure the quality of the smoothed images. Experiment 1 was repeated, but with σ_n increased to 1.75. In addition, well known image operators such as running average filter and running median filters were used for comparison. Table 1 contains the results of the different filtering operations.

Filter type	Quality measure (scale 10^{-3})
No filter	113
Signal Adaptive anis. diff.	6
3x3 average filter	44
5x5 average filter	62
3x3 median filter	24
5x5 median filter	27

Table 1: Results of filtering operations

Evidently, the anisotropic diffusion filter performs much better than the median and average filter in terms of the quality measure. The average filter is known to suppress Gaussian noise better than

the median filter, but blurs the edges. The median filter is known to have better edge preservation capabilities than the average filter [17] which can be seen from the table. The performance of both the average filter and the median filter decreases as the the filter size increases. This can be explained by the fact that edge localization will be less precise when large filtering windows are used.

It is interesting to measure how the filtering effect increases as a function of time (number of iterations). The table below shows the intermediate measurements from the previous experiment. The filtering effect is significantly reduced after a relatively small number of iterations.

Number of iterations, proposed filter	Quality measure (scale 10^{-3})
0	113
1	107
2	101
4	89
8	66
16	32
32	10
64	6

We also wanted to see if it was possible to measure the observed effect that edge enhancement may lead to incorrect edge localization. We made several experiments with both 3×3 and 5×5 Gaussian blurring masks and added various amounts of noise. We applied the signal adaptive anisotropic diffusion filter both with and without edge enhancement to all images. In all cases, the edge enhancing filter gave the lowest performance in terms of the quality measure used. Table 2 contains the measurements for an experiment with $\sigma_n = 0.8$ and a Gaussian mask of size 5×5 . The difference between the unfiltered and the filtered images increased significantly when the noise level was increased.

Filter type	Quality measure (scale 10^{-3})
No filter	60
Signal Adaptive anis. diff. with edge enhancement	50
Signal Adaptive anis. diff.	40

Table 2: Filtering with and without edge enhancement

4.3 Experimental results, real ultrasonic images

In our experiments with real ultrasonic images, we made an experimental setup with a CFM750 ultrasound sector-scanner from VingMed Sound using a 7.5 Mhz annular array probe. A typical scan

consists of 128 digitized beams each containing 512 8-bit samples. The direction of the ultrasound rays is steered mechanically at high speed, and the scanner is capable of digitizing approximately 50 scans per second. The data was digitally transferred to a HP series 700 workstation. The experiments were done on this platform with AVS. The effect of the proposed filtering strategy is illustrated in figure 9, 10. Figure 9 shows gradients computed from an original ultrasonic image of the hand of the fetus. The image was first scaled down to half the original resolution. Figure 10 shows gradients computed from a filtered image (after 25 iterations of the smoothing scheme). This and many other experiments have shown that the proposed boundary detection scheme has the property of preserving clinically relevant details while smoothing out spurious edges.

5 Applications

We have found that the proposed boundary detection scheme has several properties that are useful in many medical applications.

- It is possible for a clinician to interactively define the level of detail in the gradient images, both by adjusting the noise-parameter σ_n and the scale parameter t (the number of iterations). This property is very useful in *volume rendering* because information from an entire volumetric dataset is projected onto a two dimensional view-plane. Rendered images can then be made at a coarse scale in order to give a global overview of the volumetric dataset.
- The location of the edges is not altered by going from fine to coarser resolution. Instead, spurious edges are gradually removed while relevant details are preserved. This property makes it possible to generate detailed rendered images with only a very small amount of disturbing noise. In automatic contour detection, it is possible to remove spurious edges which might lead to incorrect detection.
- The discrete implementation is simple and well suited for implementation on dedicated hardware, for instance in a future ultrasound scanner.

The effect of the filtering strategy is clearly demonstrated in figure 11 which shows a volume rendering of a fetus hand computed from gradients detected with a 3D version of the boundary detection scheme. The initial filtering was strong, but still relevant details are preserved. One can clearly see the small bone structures within the fetus hand. The clinical value of such images becomes even more prominent when displayed with stereo vision equipment and/or in animated sequences.

The *active contour* algorithm has been proposed as a tool for automated object detection in medical images [15]. An active contour is defined as an energy-minimizing spline under the influence of internal and external forces. The internal forces serve as a smoothness constraint and the external forces typically guide the contour towards image locations with high gradients. Noise and spurious edges may therefore lead to imprecise location of the contour. We have then used the proposed smoothing strategy to remove noise and spurious edges while preserving relevant details which

should attract the contour. Figure 12 shows the contour of a urine bladder located with the active contour algorithm on a 2D gradient image computed after 25 iterations of the smoothing scheme.

6 Conclusion

In this paper we have presented a new strategy for scale-space filtering and boundary detection in ultrasonic images. The strategy integrates a signal-model for displayed ultrasonic images [11] with anisotropic diffusion [16]. A more accurate signal model may be developed in order to compensate for scanner specific non linearities in the signal processing chain. Anisotropic diffusion is an edge enhancing smoothing process. Numerical experiments showed that the edge enhancement property of Perona and Maliks original filtering scheme may introduce artificial image details and lead to incorrect edge localization. A modification to the original anisotropic diffusion scheme was made in order to preserve monotone signal transitions, and thereby avoid introduction of additional artificial details. In fact, the modification turned out to improve edge localization both visually and quantitatively. Since the filtering scheme does not include deblurring, uncertainties in the imaging system are preserved. On the other hand, more sophisticated filtering schemes should be used if the goal is to reconstruct image details which have been blurred significantly by the point spread function. This is however an ill-posed problem due to a combination of blurring, noise, reverberations, phase aberrations and other degradation phenomena.

Numerical experiments on both synthetic images and real ultrasonic images showed that the proposed filtering scheme preserves edges much better than both the running average filter and the running median filter. The discrete version of the filtering strategy is simple and well suited for implementation on dedicated hardware. The usefulness of the strategy has been demonstrated for applications in volume rendering and automatic contour detection.

7 Acknowledgments

The authors would like to thank the Norwegian Technical Research Council for supporting this study. The results presented in this paper were accomplished in cooperation with the clinical staff in the 3D ultrasound laboratory at the National Center of Fetal Medicine in Trondheim, Norway. We would like to thank the clinical staff for good cooperation and for providing the ultrasonic images used in the numerical experiments.

References

- [1] L.M Lifshitz and S.M Pizer. A multiresolution hierarchical approach to image segmentation based on intensity extrema. *Technical report, Dep Comput Sci and Radiol. Univ Noth Car-*

olina, Chapel Hill,, 1987.

- [2] Alan C. Bovik. On detecting edges in speckle imagery. *IEEE Trans. on Accoustics, speech and signal processing*, 36(10), 1988.
- [3] J.F Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
- [4] F. Godtlielsen. Noise reduction using markov random fields. *Journal of Magnetic Resonance*, 92:102–114, 1991.
- [5] Gerig Guido, Kubeler Olaf, Kikinis Ron, and Jolesz F. A. Nonlinear anisotropic filtering of MRI data. *IEEE Trans on Medical Imaging*, 11(2), June 1992.
- [6] Jørn Hokland and Patrick A. Kelly. Markov models of specular and diffuse scattering in restoration of medical ultrasound images. *Submitted to IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 1995.
- [7] J. Koendrink. The structure of images. *Biol. Cybern*, 50:363–370, 1984.
- [8] C. Kotropoulos, X. Magnisalis, I. Pitas, and M.G Strintzis. Nonlinear ultrasonic image processing based on signal-adaptive filters and self-organizing neural networks. *Submitted to IEEE Transactions on Image Processing*, 1993.
- [9] W.J Lin, S. M Pizer, and V. E. Johnson. Boundary estimation in ultrasound images. In *Information Processing in Medical Imaging, 12th International Conference, IPMI 91*, pages 285–299, 1991.
- [10] Tony Lindberg. Scale-space for discrete signals. *IEEE Transactions on Pattern analysis and Machine Intelligence*, 12:234–254, 1990.
- [11] A. Loupas. Digital image processing for noise reduction in medical ultrasonics. *Ph.D dissertation, University of Edinburgh UK*, 1988.
- [12] N. nordstrøm. Biased anisotropic diffusion—a unified regularization approach to edge detection. *Image Vision Comput.*, 8:318–327, 1990.
- [13] B. Olstad. Reasoning in entropy graphs for image operators. *Pattern Recognition*, 1994.
- [14] B. Olstad, A. Torp, and J. H. Husøy. A nonlinear loop filter model for video coding. In *IEEE Winter Workshop On Nonlinear Digital Signal Processing*, 1993.
- [15] Bjørn Olstad. Maximizing image variance in rendering of volumetric data sets. *Journal of Electronic Imaging*, 1:245–265, July 1992.
- [16] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on Pattern Anal. and Machine Intell.*, 12(7), July 1990.

- [17] I. Pitas and A.N. Venetsanopoulos. *Nonlinear digital filter - Principles and Applications*. Kluwer Academic Publishers, 1990.
- [18] P. Saint-Marc, J. Chen, and G. Medioni. Adaptive smoothing: A general tool for early vision. *IEEE Trans. on Pattern Anal. and Machine Intell.*, 13(6), June 1991.
- [19] Erik Steen and Bjørn Olstad. Volume rendering in medical ultrasound imaging. In *8th Scandinavian Conference on Image Analysis, Tromsø*, 1993.
- [20] Erik Steen and Bjørn Olstad. Volume rendering in medical ultrasound imaging based on nonlinear filtering. In *IEEE Winter Workshop On Nonlinear Digital Signal Processing*, 1993.
- [21] T. Taxt. A new class of gibbs-markov random field models for image restoration. *To be published*, 1992.
- [22] T. Taxt and E. Bølviken. Relaxation using models from quantum mechanics. *Pattern Recognition*, 24:695–709, 1991.

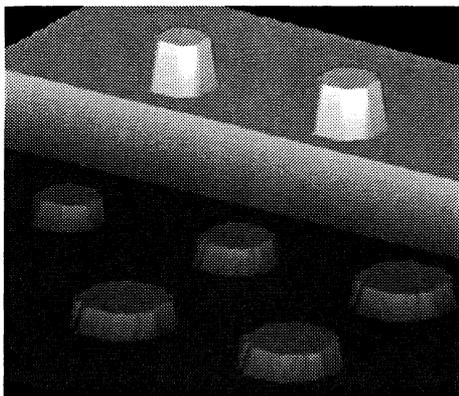


Figure 2: Topographic visualization of synthetic image

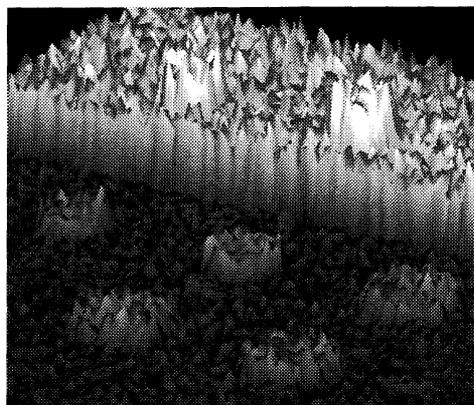


Figure 3: Topographic visualization of noisy synthetic image

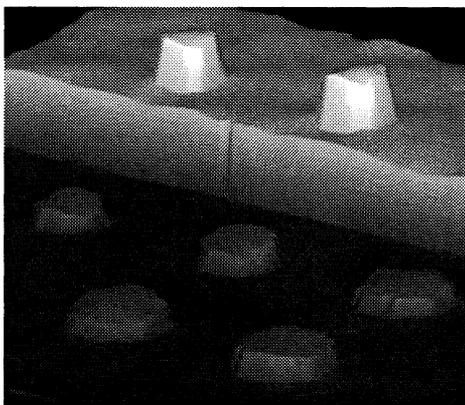


Figure 4: Topographic visualization of noisy synthetic image smoothed with proposed strategy

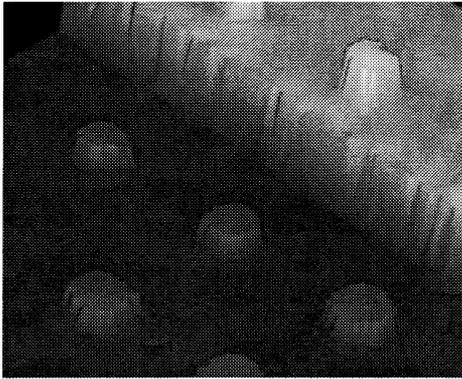


Figure 5: Topographic visualization of noisy blurred synthetic image smoothed with edge-enhancement



Figure 6: Topographic visualization of noisy blurred synthetic image smoothed with proposed strategy

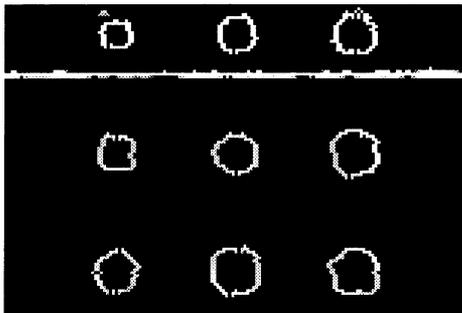


Figure 7: Edges detected from noisy blurred synthetic image which has been smoothed with edge-enhancement

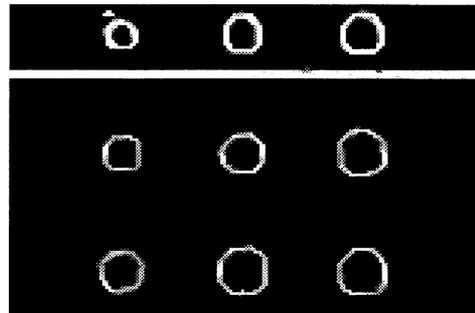


Figure 8: Edges detected from noisy blurred synthetic image which has been smoothed with proposed strategy

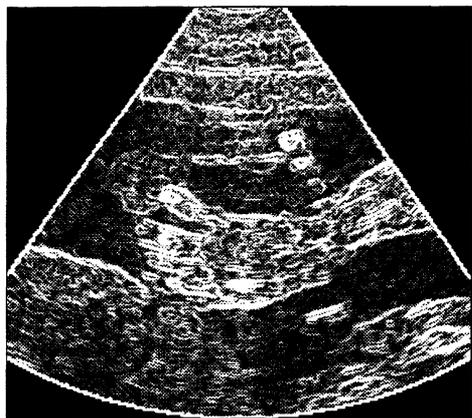


Figure 9: Gradients computed from original image

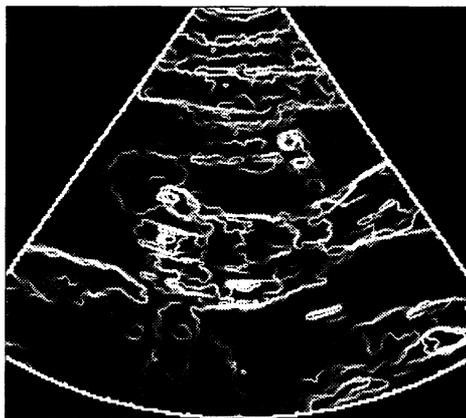


Figure 10: Gradients computed from filtered image



Figure 11: Volume rendering of fetus hand

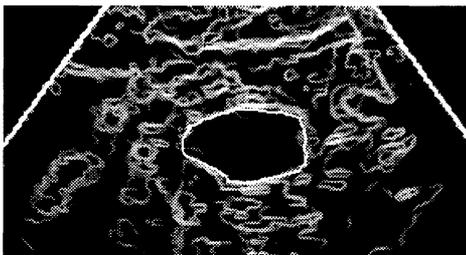
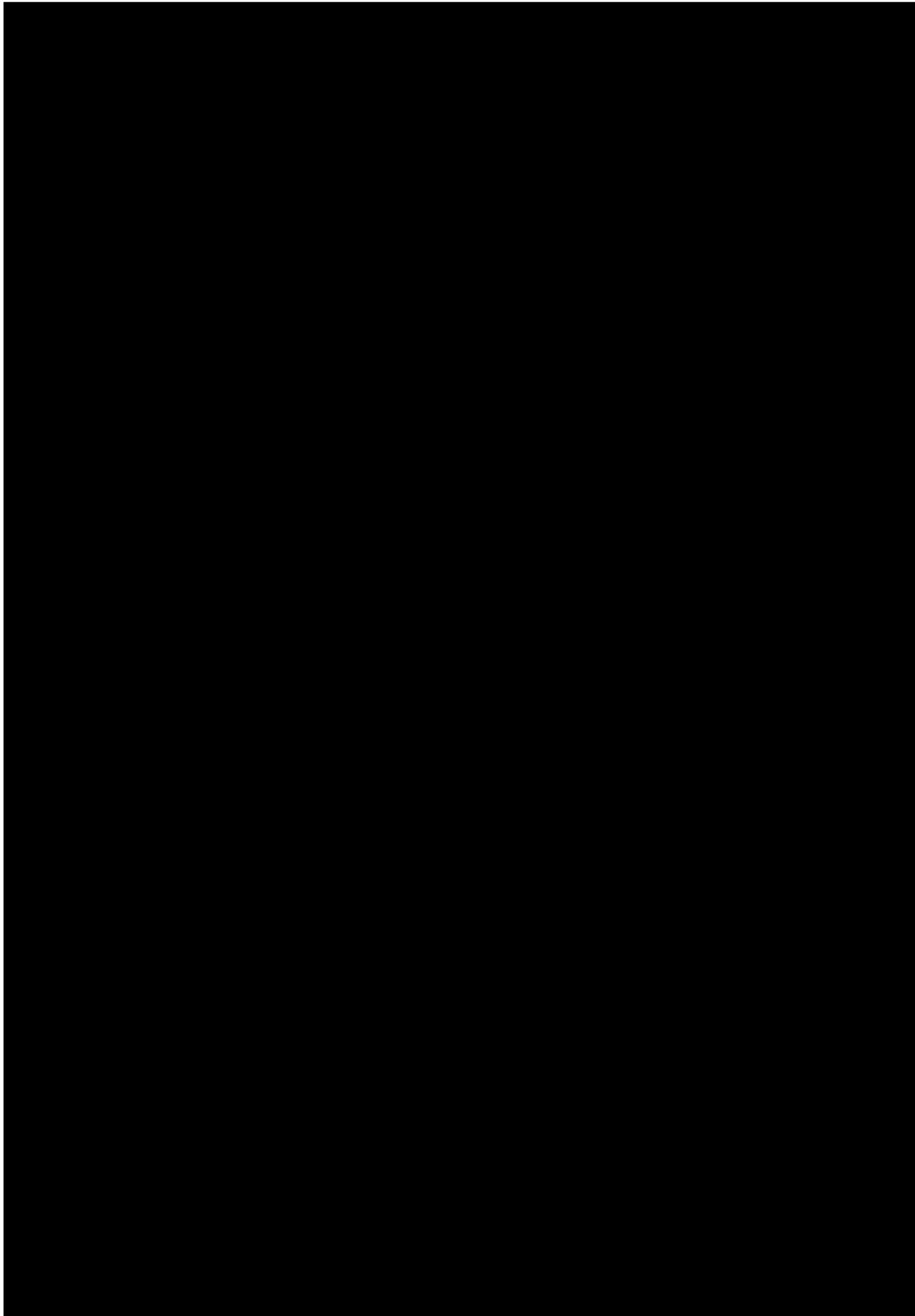


Figure 12: Contour of urine bladder



the 1990s, the number of people in the UK who are aged 65 and over has increased from 10.5 million to 13.5 million (19.5% of the population).

There is a growing awareness of the need to address the needs of older people, and the Government has set out a strategy for the 21st century in the White Paper on *Ageing Better: The Government's Strategy for Older People* (Department of Health 1999).

The White Paper sets out a number of key objectives for the Government, including:

- to improve the health and well-being of older people;
- to improve the quality of life of older people;
- to improve the opportunities for older people to participate in society;
- to improve the support available to older people.

The White Paper also sets out a number of key actions for the Government, including:

- to improve the health and well-being of older people;
- to improve the quality of life of older people;
- to improve the opportunities for older people to participate in society;
- to improve the support available to older people.

The White Paper also sets out a number of key actions for the Government, including:

- to improve the health and well-being of older people;
- to improve the quality of life of older people;
- to improve the opportunities for older people to participate in society;
- to improve the support available to older people.

The White Paper also sets out a number of key actions for the Government, including:

- to improve the health and well-being of older people;
- to improve the quality of life of older people;
- to improve the opportunities for older people to participate in society;
- to improve the support available to older people.

The White Paper also sets out a number of key actions for the Government, including:

- to improve the health and well-being of older people;
- to improve the quality of life of older people;
- to improve the opportunities for older people to participate in society;
- to improve the support available to older people.

The White Paper also sets out a number of key actions for the Government, including:

- to improve the health and well-being of older people;
- to improve the quality of life of older people;
- to improve the opportunities for older people to participate in society;
- to improve the support available to older people.

The White Paper also sets out a number of key actions for the Government, including:

- to improve the health and well-being of older people;
- to improve the quality of life of older people;
- to improve the opportunities for older people to participate in society;
- to improve the support available to older people.

Volume Rendering of 3D Medical Ultrasound Data using Direct Feature Mapping

Abstract

In this paper we explore the application of volume rendering in medical ultrasonic imaging. Several volume rendering methods have been developed for X-ray Computed Tomography (X-CT), Magnetic Resonance Imaging (MRI) and Positron Emission Tomography (PET). Limited research has been done on applications of volume rendering techniques in medical ultrasound imaging because of a general lack of adequate equipment for 3D acquisitions. Severe noise sources and other limitations in the imaging system make volume rendering of ultrasonic data a challenge compared to rendering of MRI and X-CT data. Rendering algorithms that rely on an initial classification of the data into different tissue categories have been developed for high quality X-CT and MR-data. So far, there is a lack of general and reliable methods for tissue classification in ultrasonic imaging. This paper focuses on volume rendering methods which are not dependent on any classification into different tissue categories. Instead, features are extracted from the original 3D data-set, and projected onto the view plane. We found that some of these methods may give clinically useful information which is very difficult to get from ordinary 2D ultrasonic images, and in some cases renderings with very fine structural details. We have applied the methods to 3D ultrasound images from fetal examinations. The methods are now in use as clinical tools at the National Center of Fetal Medicine in Trondheim, Norway.

I. Introduction

Three-dimensional (3D) display techniques have been studied extensively for medical applications for the past several years and are well established as a clinical tool in X-CT imaging [23, 58, 54]. Visualization techniques for MRI and PET have been explored in [1, 57, 4, 60, 41]. Recently, special techniques have also been proposed for multi-modal images with combination of PET and MR-data [8, 56]. Only limited research has been done on visualization of 3D ultrasonic data. One reason for this is that the scanning technology is still immature and also that the quality of ultrasonic images is relatively low compared to X-CT and MRI. Examples can be found in visualization of polygon data which are extracted automatically or semi-automatically from the left ventricle of the heart and the Mitral Valve [12, 10, 13]. Others have applied similar methods to visualize the arterial wall in intra-vascular ultrasound [14]. Standard volume rendering techniques have been used in 3D color Doppler ultrasound imaging to visualize quantitative maps of the Doppler shift of blood flow [11]. Bamber et al.[3] have also made visualizations of 3D Doppler echo data. Mc Cann et al.[39] have made transparent renderings of beating heart. Tomographic Technologies has developed a commercial system for rendering of both 3D and 4D ultrasonic data.

A volumetric data set is made of samples from a 3D *scalar field* and is usually represented as a 3D array. The sample points, or volume elements, are called *voxels*. A *cuberille* is a regular 3D array of voxels. In this paper, we assume that the 3D data is represented as a cuberille. Other geometries can

usually be scanconverted to a regular cuberille prior to the analysis. A common way to visualize 3D data is by interactive slicing of the cuberille. Thin slices can be displayed directly as intensity images. By successive rendering of parallel slices, the viewer can at least mentally get an overview of the volumetric data set.

Another well known technique is threshold rendering. This technique has been found useful for display of isovalue surfaces from Computed Tomography data, where each material falls within a specific density range. The surfaces can be rendered by using back to front traversal [16], ray tracing [30] or surface reconstruction [36].

Photo realistic renderings can be obtained of polygonal surfaces extracted from the data-set by thresholding or by using automatic surface detection schemes [44, 43, 20, 24]. Shading techniques can then be applied to the extracted polygon data. These approaches rely heavily on the robustness of the surface detection schemes.

Volume rendering is used to give a 2D representation of the interior of an entire volumetric data set. In medical imaging, the usefulness of such methods relies on their ability to extract clinically useful information. Several algorithms have been proposed [7, 55, 33, 48, 34, 40, 45] for this purpose. Drebin et al. [7] developed a technique for rendering of data-sets which represents a mixture of materials. This paper has become a “classic” and the methodology has been used by several researchers for medical applications [8, 56]. Drebin emphasizes that volume rendering algorithms should not incorporate discrete decisions. Such decisions tend to produce artificially sharp boundaries and do not reflect the diffuse transitions often found in real world applications. Instead, Drebin suggests that statistical analysis should be used to decide the amount of each material present within each voxel. Drebin[7] and Levoy[33] suggest that local gradients should be computed within the cuberille. Levoy applies the density values directly while Drebin computes the gradients from densities assigned to the different materials. These local gradients are used in the shading computations together with an *opacity* and *color* value, which has been assigned to the various tissue categories. The methods in [7, 33] give realistic impressions of the implicit surface geometry within the cuberille if the initial classification stage is reliable.

Meinzer et al. [40] have developed a rendering pipeline for use in medical imaging, which is based on the work of Kayija [29]. Several simplifications are made to reduce the computational complexity. The model is achromatic in contrast to the methods presented in [7, 33], and combines properties of surface shading and light scattering by particles. Heidelberg et al. demonstrate their method on high quality X-CT data and the results are quite impressive. It is however obvious that data with a substantial noise component will give a less attractive result. Evaluations of volume rendering techniques in medicine can be found in [25, 27, 26]. Problems specially related to rendering of ultrasound data are discussed in [28, 17].

Several fundamental limitations in the imaging system, make volume rendering a difficult challenge for ultrasonic data. First of all, ultrasonic images are distorted by several kinds of noise. This includes thermal noise caused by the amplifying circuits, acoustic noise, and phase effects (speckle noise and phase aberrations.) The substantial presence of noise makes the ultrasound images hard to interpret. Another severe limitation is that the received echo signal from structures lying parallel

to the radial direction can be very weak and thus difficult to render. Structures lying normal to the radial direction give a much stronger echo. Yet another problem is that high density material may totally absorb the transmitted ultrasound waves, leaving structures further from the transducer in the shadow.

The limitations suggest that binary thresholding schemes, which has been successfully used in X-CT-imaging, will generally be unreliable in ultrasound imaging. Rendering algorithms such as the one presented by Drebin [7] will be of limited use unless more sophisticated tissue classification schemes are developed. Tissue characterization schemes based on fractal dimension [9] and neural networks [46] have been proposed for classifying tissue types as normal or abnormal. However, so far there is a lack of general and reliable methods for tissue classification in ultrasonic imaging.

In this paper we focus on methodologies which do not rely on any initial classification into different tissue categories. Instead, features are extracted from the original 3D data-set, and projected into the view-plane. The feature extraction then serves as a low-level segmentation process which then is meant to emphasize on clinically interesting features of the data-set.

Several authors have presented rendering methods which fall into this category [48, 45]. We explore both old and more recent approaches, which have been designed to meet the difficult noise conditions in ultrasonic imaging. The aim of this study has been to find methods which give clinically interesting renderings, **not** photo-realistic images. The methods are evaluated for several fetal studies.

II. Methods

A. Image acquisition

All ultrasound data used in this study has been acquired using the CFM-750 ultrasound annular array sector-scanner from VingMed Sound. A typical scan consists of 128 digitized beams each containing 512 8-bit samples. The direction of the ultrasound rays is steered mechanically at high speed, and the scanner is capable of digitizing approximately 50 scans per second.

A prototype system for 3D ultrasound acquisition, which works with the CFM-750, has been developed at the Department of Biomedical Engineering in Trondheim, Norway. A 2D probe is placed in a holder, and the scan plane is either tilted, translated or rotated using a step motor. The step motor is synchronized with the scanner, giving the scanner time to process the data before the scan plane is moved into a new position. A typical acquisition takes approximately 2 seconds and consists of 95 scans. It is essential that the probe-holder is kept stable during the acquisition, to avoid motion-artifacts. The tilting probe covers a sector of 51 degrees in the motion plane, while the rotating probe covers a complete cone. The acquired data is directly transferred to an external computer and scan converted into a regular volume. Trilinear interpolation is used to fill in the gaps between the adjacent scan converted rays. Similar acquisition equipment has been developed by others. An overview of different acquisition strategies can be found in [28].

B. Boundary detection

The boundaries in ultrasonic images carry structural information which is vital for diagnostic purposes [31]. In [51, 52] boundary information was exploited by computing local *gradient magnitude* values within the cuberille. These values can be computed directly from the raw data by simple operators such as *Roberts gradient* [19]. Schemes like this are however very noise-sensitive, and it was therefore found in [51, 52] that more robust boundary-detection schemes should be used.

Boundary detection is particularly difficult in ultrasonic imaging because of speckle, acoustic reverberations and several other limitations in the imaging system. The scattering of ultrasonic waves is caused by spatial variations in the acoustic impedance in the human body. The waves can be divided into specular reflections and diffuse scattering (volume scattering). Specular reflections appear between layers of tissues with different acoustic impedance when the boundary is smooth. The specular reflections are however only captured by the transducer if the layers are approximately normal to the radial direction. This is a direct consequence of the reflection laws. In an ultrasonic image, specular reflections appear as bright boundaries which are blurred according to the point spread function. The blurring means that two specular reflections from both sides of a thin tissue layer actually will appear as a single bright boundary in an ultrasonic image.

High frequency spatial variations in the acoustic impedance give rise to volume scattering characterized by the backscatter coefficient per volume. The volume scattering will therefore differ between different kinds of tissue and will cause gray level differences in the displayed image. This makes it possible to detect the boundaries between tissues by computing local gradient estimates. Due to the coherent detection, constructive and destructive interference between neighboring scatterers give rise to multiplicative noise, speckle, which modulates the gray level image. It is therefore essential that the gradient estimates are made as insensitive to speckle as possible. A gradient operator may however produce artificial double edges from single specular reflections. In this paper, we have assumed that these specular reflections are not prominent in the displayed images.

A probabilistic approach to edge detection in ultrasound images was suggested by Lin et al.[35], but failed to locate the edges accurately. Lin also argued that a boundary detection scheme should fill in the boundary gaps that sometimes appear due to image dropouts. This approach may however introduce artificial details.

A. Bovik suggested to combine Laplacian- of- Gaussian (LOG) edge detection with Ratio- of- average estimation [5] as a general approach to edge detection in speckled images. LOG-based schemes such as the Canny-operator [6] will however not handle multiple interfering boundaries properly, and the localization of boundaries is accurate only for small window-sizes [47].

Another approach is to apply edge-preserving filtering schemes prior to the gradient calculations. Such a filtering scheme must satisfy the conflicting requirements of maximum noise suppression and preservation of edges and other image details, in order to be useful in medical applications [31].

It has been found by experimental measurements [37, 38, 31] that a reasonable image formation

model for *displayed* ultrasonic images is given by the following equation:

$$x = s + s^{1/2}n \quad (1)$$

where s is the original signal, and n is a zero-mean Gaussian variable which then models the noise process. This signal model assumes that the original signal has been log-compressed and low-pass filtered. A more accurate model could be developed for a specific ultrasound scanner by taking time-gain-compensation and other nonlinearities in the signal processing chain into considerations.

The non-stationary nature of ultrasonic images has motivated several authors to develop adaptive spatial filtering schemes which adjust their smoothing properties according to local image statistics [2, 38, 32, 31].

A signal adaptive maximum likelihood filtering scheme has been proposed by Kotropoulos et al.[31]. The filter uses a signal adaptive weighting factor to adjust the filtering strength locally in each voxel. The filter preserves edges while smoothing homogeneous regions. The filter computes signal statistics locally in windows centered around each pixel in the image. If an edge is present within the current window, the window-size is reduced, but then the output-variance of the filter increases. Thus, noise will not be reduced very well close to the tissue boundaries. This will also be a problem with other filtering schemes which adaptively control the window-size.

Steen and Olstad[50] developed a boundary detection scheme for ultrasonic images which was based on anisotropic diffusion. The anisotropic diffusion is a smoothing process which smooths within homogeneous regions while preserving or enhancing the region boundaries. It was first introduced as an image processing tool by Perona and Malik[47]. Others have also presented filtering schemes based on essentially the same computational framework[49, 53, 18]. The anisotropic diffusion has also been used for noise suppression in MR-imaging by Gerig et al.[21].

The anisotropic diffusion equation is:

$$I_t = div(c(x, y, t)\nabla I) \quad (2)$$

where I_t represents the image, and $c(x, y, t)$ is a conduction coefficient which is chosen as a monotonically decreasing function $f(\cdot)$ (the weight function) of the image gradient with $0 \leq f(I_x(x, y, t)) \leq 1$. Perona and Malik defined t as the scale-space parameter, thus larger values of t correspond to coarser resolutions of the image. Consider a step edge convolved with a Gaussian and let $\phi(I_x) = f(I_x) \cdot I_x$ denote the flux function. It was shown in [47] that if $\phi'(I_x) > 0$, the blurred edge will be *enhanced*, while if $\phi'(I_x) < 0$ the edge will be even more blurred.

The following weight function has been proposed by several authors[47, 49, 18]:

$$f(I_x) = e^{-\frac{I_x^2}{2\sigma^2}} \quad (3)$$

It can then be shown that blurring will occur for $I_x < \sigma$ while $I_x > \sigma$ will lead to edge-enhancement. The parameter σ determines the noise level which is supposed to be space invariant. According to the signal model in eq. 1, this assumption is not valid for ultrasonic images. It was therefore argued

in [50] that the parameter σ should be computed from local estimates of the signal level:

$$\hat{\sigma} = \sigma_n \hat{s}^{1/2} \quad (4)$$

where σ_n is a global parameter which defines the level of detail to be preserved. Regular Gaussian blurring can be obtained by setting σ_n to infinity. A numerical implementation relies on the estimate of the local signal level \hat{s} . A simple solution is to use the *previous* output of the filter I_t as an approximation of the signal level. Because of the intra-region smoothing, the filtered value will gradually give a better estimate of the signal level in the surrounding region as the filter is iterated. Initially, a more robust estimate can be achieved by computing the ML-estimate[31] of the original signal, given the signal model in eq. 1 within a small window.

Numerical experiments in [50] showed that the edge-enhancement may introduce artificial boundary details in 2D images. The edge-enhancement property can be avoided by using a weight function of the form C/I_x , where C is a constant [50, 52, 42]. This weight function should be selected only if $I_x > \sigma_n \hat{s}^{1/2}$, while a smoothing weight function can be selected for $I_x \leq \sigma_n \hat{s}^{1/2}$. In this way, one achieves intra-region smoothing while avoiding edge-enhancement. The constant C is then calculated according to the specific smoothing function to ensure a continuous weight-function. A modified version of the weight-function in eq. 3 will then be:

$$f(I_x) = \begin{cases} e^{-\frac{I_x^2}{2\hat{\sigma}^2}} & \text{if } I_x \leq \hat{\sigma} \\ \frac{e^{-\frac{1}{2}\hat{\sigma}}}{I_x} & \text{otherwise} \end{cases} \quad (5)$$

The anisotropic diffusion equation can be discretized on a lattice-structure with brightness values associated to the vertices, and conduction coefficients to the arcs[47]. A 4- or 8- nearest neighbor discretization of the Laplacian operator can be used, and the norm of the gradients can be approximated with the absolute value of their projections along the direction of the arcs. A multi-grid implementation of anisotropic diffusion was presented by Saint Marc et al.[49].

A numerical implementation of the weight-function in eq. 5 requires some thought in order to keep the numerical process stable and to avoid blurring across boundaries. Significant monotone sections can be filtered properly by choosing the weight function $f(I_x) = \frac{e^{-\frac{1}{2}\hat{\sigma}}}{I_x} = C/I_x$ *symmetrically* around the center-pixel, hence, the two flow-contributions cancel out, and can thus be ignored.

Iterating the basis filter can be thought of as scale-space operations [47]. By iterating the filter, noise and small details will be smoothed, while significant boundaries will be preserved. By allowing the end-user to adjust the number of iterations, the level of detail in the rendered images can be interactively defined. This last property is very important in clinical applications, as all filtering of data is a compromise between noise suppression and image-detail preservation. In volume rendering, strong filtering can often be allowed, as information from the *whole* volume will be compressed into one 2D image. Boundaries can then often be extracted at a coarser scale. It is also possible to fix the number of iterations and use the parameter $\hat{\sigma}$ in eq. 5 as the scale space parameter[49]. The scale parameter then approximately defines the magnitude of gradients to be preserved.

Numerical experiments in [51, 52] showed that the initial filtering of the data was essential in order to achieve satisfactory results. Figure 2 shows gradients computed from an original ultrasonic image captured during a fetal examination. Figure 3 shows gradients computed from a filtered version of the image.

C. Volume Rendering

Rendering based on Direct Feature Mapping

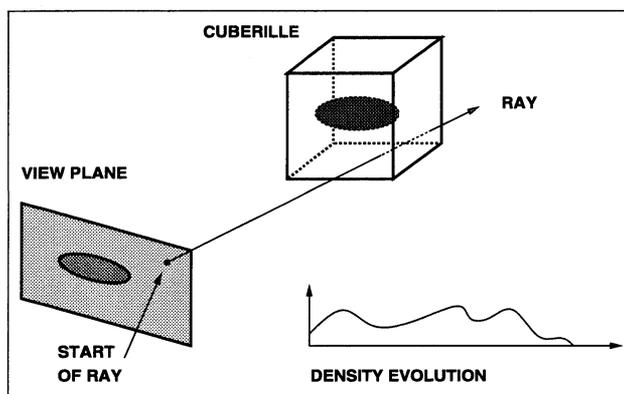


Figure 1: Rendering process

Several rendering algorithms that do not rely on an initial classification of the voxels into different material categories have been proposed in the literature[48, 45]. These rendering models are based on extracting characteristic parameters from the density evolutions, $\rho(t)$. The attributes are displayed at the associated pixel in the view plane. The rendering methods can be applied to the entire data-set in order to give a global overview of the data, or smaller portions of the data-set may be selected interactively for rendering. In the latter case, the aim may be to reveal the interior of smaller structures which are difficult to interpret from the global overview. The following attributes were evaluated in [45].

1) Maximum value projection:

$$M = \sup_{t_1 \leq t \leq t_2} \rho(t) \quad (6)$$

This method has shown to give clinically interesting renderings, but only for a limited number of views. Only a single voxel is projected from each evolution, which means that most information from the evolution is lost, and that noisy voxels with high intensities may be projected instead of voxels actually corresponding to the highest densities. This method was evaluated for Nuclear medicine data in [59] as a technique for “hot spot” imaging. The projected voxel-values would

then come from regions with high nuclear activity. In ultrasonic imaging, the maximum value projection method can be thought of as a technique for projecting the voxels from areas with maximum variation in acoustic impedance.

2) Average value projection:

$$A = \frac{\int_{t_1}^{t_2} \rho(t) dt}{t_2 - t_1} \tag{7}$$

This method was evaluated in [45] and turned out to give low contrast renderings of little clinical value in most cases. A similar method which consisted of projecting the sum of all voxels along each ray was also evaluated in [59] and was found to give low contrast renderings. The average value projection method is nevertheless much more insensitive to noise than the maximum value projection method, and may be useful for a limited range of depths. In addition, a threshold value can be set to project only voxels with values above this threshold. The latter method will intuitively be of a limited use for ultrasonic images because the individual voxel-values generally have no absolute interpretation. (On the contrary, this is very much the case for X-CT images). Further improvements can be made by using depth attenuation and by applying local contrast enhancement filtering to the rendered image. Some experiments on ultrasonic studies with this method were presented in [28].

Sabella[48] has suggested several parameters that were mapped into the HSV color model[15]. In addition to the maximum value M , Sabella defined the following attributes:

3) Distance to maximum value:

$$D = t_0 \text{ such that } \rho(t_0) = M \tag{8}$$

4) Center of gravity:

$$C = \frac{\int_{t_1}^{t_2} t\rho(t) dt}{\int_{t_1}^{t_2} \rho(t) dt} \tag{9}$$

5) Attenuated intensity: The attenuated intensity is modeling the scattering of light in clouds. The density voxels transmit light towards the viewer and at the same time absorb light from more distant voxels:

$$B = \int_{t_1}^{t_2} e^{-\tau \int_{t_1}^t \rho^\gamma(\lambda) d\lambda} \rho^\gamma(t) dt \tag{10}$$

The transmitted light is given by $\rho^\gamma(t)$ which becomes filtered according to the factor $e^{-\tau \int_{t_1}^t \rho^\gamma(\lambda) d\lambda}$. τ and γ are constants that control the thickness of the cloud.

6) Principal component analysis: In [45] *principal component analysis - PCA* (see [22, 19]) was proposed as a general tool for compressing information found in the density evolutions into one or three orthogonal attributes that can be displayed in the rendered view. Numerical experiments

[45] indicated that the methodology can be well suited for rendering of anomalies in homogeneous tissue. Other methods should be used if the goal is to focus on the geometry of the different tissue structures. Also, PCA focuses on image variance which may not be the best measure of information content from a clinical point of view.

7) Gradient Magnitude projection: The methods described above project information from a large volumetric data-set into one 2D image. Their success for a given application depends on their ability to reflect clinically interesting features of the data-set. For many applications it is interesting to render the surfaces between different tissues. The surfaces define the geometry of the different tissue structures. Also, it is interesting to see how different surfaces are placed relative to each other. This can be accomplished by allowing the surfaces to be transparent. The implicit surfaces are however difficult to extract in ultrasonic images because of the limitations in the imaging system. In [52] local gradients were computed within the cuberille after applying an edge preserving smoothing scheme similar to the one in the previous section. Only the gradient magnitude values were computed with a simple scheme:

$$\begin{aligned}
 I_{surf}(x, y, z) &= |\nabla I(x, y, z)| \approx \\
 & \text{sqr}t(\text{sqr}(I(x-1, y, z) - I(x, y, z)) + \\
 & \text{sqr}(I(x, y-1, z) - I(x, y, z)) + \\
 & \text{sqr}(I(x, y, z-1) - I(x, y, z)))
 \end{aligned} \tag{11}$$

Voxels from homogeneous regions will not contribute to the rendered image, as opposed to the average value-projection method. The voxel-values within these homogeneous regions give no information about the geometry of the tissue structures. Projecting the gradients can therefore be thought of as a feature extraction method that enhances contrast between different tissues. A rendered view can be constructed by using a scheme similar to the one found in [33] where each sample location the ray passes makes a contribution of emitted white light. The contribution is then the sampled gradient magnitude value ρ_i multiplied with the corresponding opacity $o(\rho_i)$. At the same time, incoming light from behind will be filtered by the factor $(1 - o(\rho_i))$. Light is only emitted in the viewing direction. Let the density evolution $\rho(t)$ be given by N samples $(\rho_1, \rho_2, \dots, \rho_N)$. The following attribute is then extracted:

$$L = \rho_1 o(\rho_1) + \sum_{i=2}^N \rho_i o(\rho_i) \prod_{j=1}^{i-1} (1 - o(\rho_j)) \tag{12}$$

$o(\rho)$ is a function which maps the gradient values directly into opacity values. This mapping may just be a scaling, or more sophisticated functions could be used to emphasize on special features of the data-set. Low opacity values will make the surfaces highly transparent. In addition, each sample can be weighted with a depth-dependent attenuation factor to differentiate structures lying close to the view-plane from those lying far from the view-plane.

In some cases, more realistic images can be composed by modeling directional light sources and take the directions of the gradients into account. This technique is called surface gradient shading

and is included in many rendering algorithms [7, 33, 40]. Surface shading is especially suited if the surfaces are well defined and have a smooth appearance. However, shading modulates the appearance of the surface strength by the surface orientation which may also confuse the visualization. In ultrasound imaging the magnitude of the local gradients will also depend on the surface orientation relative to the radial direction. The transducer can in fact be compared to a directional light source which illuminates the tissue during acquisition.

Because the structures are made transparent, visual ambiguities may arise in the rendered images. Numerical experiments in [52] showed that these visual ambiguities were removed by computing an animated sequence of rendered images or by allowing the end user to modify the viewing parameters interactively. Depth information can also be improved by presenting a pair of renderings as stereo images, or by using a single color image with depth encoded as color saturation.

III. Results

All methods evaluated below were implemented in C under AVS (Application Visualization System from Advanced Visualization Systems Inc.) on an HP-7000 workstation (78 MIPS peak performance). All renderings were computed in a few seconds. The most time-consuming operation was the initial filtering required by method 7 which took approximately 30 seconds per iteration for a 3 Mb data-set. This operation could however be optimized by using a multi-scale algorithm proposed in [49]. The filtering is also well suited for implementation on dedicated hardware [47]. Method 7 was also implemented by thresholding the gradient-values and projecting only the remaining voxels directly into the view-plane. This approach reduced the computation time to less than one second for each rendered view.

In our first experiments we tested several different volume rendering techniques on a 3D image of the hands of a 30 week old fetus. In this case it was interesting to render the small bone-structures in the fingers. We then made renderings by applying the different methods explained in the previous section. The renderings were carefully examined together with medical researchers at the center of Fetal Medicine in Trondheim. The methods were evaluated by their ability to render the clinically interesting structures and also by sensitivity to viewing parameters and noise.

Figure 4 shows a maximum value projection of the data-set. Most of the interesting structures are visible, but this was not the case for other viewing orientations. In the maximum value projection, only one voxel is selected from each density evolution, which makes this method very sensitive to noise and viewing parameters. Slightly better results were obtained by applying noise reduction filtering prior to the rendering. Still, useful results were obtained only for a limited number of viewing orientations.

We also applied the average value projection method, but these renderings were foggy with little clinical value. This is not surprising, as averaging the whole density evolution tends to smear out small details. Better results were obtained by using a limited range of depths, but only for some viewing orientations.

The method proposed by Sabella [48] was tested by mapping the maximum value (M), distance to maximum value (D) and attenuated intensity (B) into HSV color space. The resulting images were dominated by the M and D attributes and gave similar structural information as the maximum value projection method. The D attribute gave somewhat better understanding of the spatial positions of the voxels with the maximal value. The B attribute did not give any additional clinical value. Clinically interesting renderings were obtained only for some viewing directions.

The gradient magnitude projection method was applied to the data-set, including a 3D version of the initial filtering scheme described in this paper. Figure 6 illustrates this technique. The bone structures, which are not grown together, can be clearly seen in this image. Also one can see that surfaces lying normal to the radial direction are especially highlighted. It is however difficult to decide the position of some structures relative to each other. An animated sequence of images was therefore made. The use of animation gave a better understanding of the geometry of the structures, and removed visual ambiguities. Further improvements were made by using depth-dependent attenuation to discriminate structures lying near the view-plane from those lying far from the view-plane. The rendered images were carefully examined by clinicians and they did not find any artificial details inside the fetus hands. Surface shading was also applied to the filtered data and turned out to give nice renderings of the outer surfaces of the hands. However, visual ambiguities made it very difficult to see the bone structures within the hands as the surfaces were made more transparent.

Figure 5 shows an image generated by the same technique, but this time without the initial filtering. The image appears substantially more foggy, and much of the small structural details have disappeared. For other viewing orientations, the effect of the filtering was even more prominent. This was not surprising because of the speckle found in the original data.

Figure 7 and 8 show the head and parts of the body of a 20 week old fetus. The renderings were made by using method 7. One can clearly see details of the fetus face and parts of its body.

IV. Discussion

Volume rendering techniques have been used with success for several medical applications with X-CT and MR-imaging. The high image quality obtained in these image modalities has motivated the development of highly sophisticated rendering techniques. Volume rendering techniques have shown to be superior to polygon based surface-rendering methods for several applications which involve displaying of soft-tissue structures. The success of such volume rendering methods are due to their ability to render information which is difficult to obtain from ordinary 2D images. The introduction of 3D data has given new possibilities also in ultrasonic imaging but the low signal-to-noise ratio and several other fundamental limitations make volume rendering a difficult challenge. These limitations suggest that surface rendering based only on binary thresholding of the ultrasonic data in most cases will be unreliable.

Several rendering algorithms which rely on an initial classification of the voxels into different materials have been proposed in the literature. Some of these algorithms have successfully been applied

to MR and X-CT data. So far there is a lack of general and reliable methods for tissue characterization in ultrasonic imaging. Therefore, we have focused on rendering methods that do not rely on any initial classification into different tissue categories. Several rendering methods that fall into this category have been evaluated for rendering of fetal organs. The maximum value projection method can be thought of as a method of projecting voxels from regions with maximal spatial variation in acoustic impedance. The method turned out to give clinical useful results, but is very sensitive to noise and viewing parameters. Nevertheless, it is a very simple rendering method which can easily be included as an option in a system for visualization of ultrasonic data. The average value projection method gave low contrast renderings with little details. The method due to Sabella was dominated by the maximum value attribute and gave similar results as the maximum value projection method.

The best results in this study were obtained by using the gradient magnitude projection method. The results were obtained by first applying an edge preserving smoothing scheme, and then computing gradients from the smoothed data. Finally, a weighted sum of gradients were projected onto the view-plane. The method produced rendered images with very clear details and also turned out to be less sensitive to viewing parameters. The initial smoothing was essential to achieve satisfactory results. Surface shading was found to be useful for visualizing opaque surfaces, but gave confusing results when the surfaces were made transparent.

Computing gradients in ultrasonic images may in some cases produce artificial double edges from strong specular reflections that are captured by the transducer. We did not find this to be a problem in our experiments with images of fetal organs. A possible solution to this potential problem is to use a mixed scheme which combines separately detected specular reflections with gradients computed from boundaries between tissues with different backscatter coefficients per volume. The methods are now implemented as clinical tools at the National Center of Fetal Medicine in Trondheim.

V. Acknowledgments

The authors would like to thank the Norwegian Technical Research Council for supporting this study. The results presented in this paper were accomplished in close co-operation with the clinical staff in the 3D ultrasound laboratory established by prof. Dr. med. Sturla Eik-Nes at the National Center of Fetal Medicine in Trondheim, Norway. We would like to thank the clinical staff for participating in the evaluation of the rendered images, and especially Torvid Kiserud who provided us with excellent 3D acquisitions. Also, we would like to thank Hans Torp at Dept for Biomedical Engineering for providing detailed knowledge about ultrasonic imaging both in general and also specific to the scanner used in our numerical experiments. Finally, we would like to thank the reviewers for many useful comments and suggestions.

References

- [1] L. Axel, G. T Herman, and J. K Udupa et.al. Three-dimensional display of NMR cardiovascular images. *Comput. Assist. Tomography*, 7:172–174, 1983.
- [2] J. C. Bamber and C. Daft. Adaptive filtering for reduction of speckle in ultrasonic pulse-echo images. *Ultrasonics*, 24:41–44, 1986.
- [3] J.C Bamber, R.J Eckersley, P. Hubregtse, N.L Bush, D.S Bell, and D.C Crawford. Data processing for 3D ultrasound visualization of tumor anatomy and blood flow. In *Visualization in Biomedical Computing*, Chapel Hill, NC, SPIE, 1992.
- [4] Michael Bomans, Karl-Heinz Hohne, Ulf Tiede, and Martin Riemer. 3D segmentation of mr images of the head for 3D display. *IEEE Trans. on Medical Imaging*, 9(2), June 1990.
- [5] Alan C. Bovik. On detecting edges in speckle imagery. *IEEE Trans. on Acoustics, speech and signal processing*, 36(10), 1988.
- [6] J.F Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
- [7] R. A. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. *Computer Graphics*, 22(4):65–74, August 1988.
- [8] D. N. Levin et al. Integrated three-dimensional display of MR and PET images of the brain. *Radiology*, 172:783–789, 1989.
- [9] I. Akiyama et al. Tissue characterization by using fractal dimension of b-scan image. In *IEEE Ultrasonics Symposium Proceedings*, volume 3, pages 1353–5, 1990.
- [10] M Verlande et al. 3D reconstruction of the beating left ventricle and mitral valve based on multiplanar tee. In *Proceedings. Computers in Cardiology*. IEEE Computer Society Press, 1991.
- [11] Paul A. Picot et al.. Three-dimensional colour doppler imaging. *Ultrasound in Medicine and Biology*, 19(2), 1993.
- [12] R. Pini et al. Echocardiographic three-dimensional visualization of the heart. *NATO ASI series, VOL. F60. 3D imaging in Medicine*. Edited by K.H. Hohne et al. Springer-Verlag Berlin Heidelberg, 1990.
- [13] R. Pini et al.. Computed tomography of the heart by ultrasound. In *Proceedings. Computers in Cardiology*. IEEE Computer Society Press, 1991.
- [14] RI Kitney et al.. 3D visualization for the study of arterial disease and tissue caharecterisation. In *Proceedings. Computers in Cardiology*. IEEE Computer Society Press, 1991.

- [15] J.D Foley, A. van Dam, S.K Feiner, and J.F Hughes. *Computer Graphics: Principles and Practice*. Addison Wesley, 1990.
- [16] G. Frieder and R. Reynolds. Back-to-front display of voxel-based objects. *IEEE Computer Graphics and Application*, 5(1):52–60, January 1985.
- [17] J. E. Gardener. Volume imaging of soft tissues with ultrasound. In *IEE Colloquium on 3D Imaging Techniques for Medicine*, 1991.
- [18] F. Godtlibsen. Noise reduction using markov random fields. *Journal of Magnetic Resonance*, 92:102–114, 1991.
- [19] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison Wesley, 1992.
- [20] D. Gordon and J. Udupa. Fast surface tracking in 3D binary images. *Comp. Vis. Graph. and Im. Proc.*, 45(2):196–241, 1989.
- [21] Gerig Guido, Kubeler Olaf, Kikinis Ron, and Jolesz F. A. Nonlinear anisotropic filtering of MRI data. *IEEE Trans on Medical Imaging*, 11(2), June 1992.
- [22] E. L. Hall. *Computer Image Processing and Recognition*. Academic Press, 1979.
- [23] D.C Hemmy, D.J David, and G.T Herman. Three-dimensional reconstruction of craniofacial deformity using computed tomography. *Neurosurgery*, 13:534–541, 1983.
- [24] G. T. Herman, J. Zeng, and C.A. Bucholtz. Shape-based interpolation. *IEEE Computer Graphics and Applications*, 12(3):69–79, 1992.
- [25] Gabor T. Herman. A survey of 3D medical imaging technologies. *IEEE Engineering In Medicine and Biology*, pages 15–17, December 1990.
- [26] K.H. Hohne and U. Tiede. Surface rendering. investigation of 3D-rendering algorithms. *IEEE Computer Graphics & Applications*, March 1990.
- [27] K.H. Hohne and U. Tiede. Rendering tomographic volume data: Adequacy of methods for different modalities and organs. *NATO ASI series, 3D imaging in Medicine*. Edited by K.H. Hohne and U. Tiede F60, 90.
- [28] F. Hottier and A. Collet Billon. 3D echography: Status and perspective. *NATO ASI Series, 3D Imaging In Medicine*, F. 60, 1990.
- [29] Kajiya J, T. The rendering equation. *Computer Graphics*, 20(4):143–149, August 1986.
- [30] J. T. Kajiya and B. P. Von Herzen. Ray tracing volume densities. *Computer Graphics (Proc. Siggraph)*, 18(3):165–173, July 1984.
- [31] C. Kotropoulos, X. Magnisalis, I. Pitas, and M.G Strintzis. Nonlinear ultrasonic image processing based on signal-adaptive filters and self-organizing neural networks. *Submitted to IEEE Transactions on Image Processing*, 1993.

- [32] C. Kotropoulos and Ioannis Pitas. Optimum nonlinear signal detection and estimation in the presence of ultrasonic speckle. *Ultrasonic Imaging*, 14:249–275, 1992.
- [33] M. Levoy. Volume rendering: Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, pages 29–37, May 1988.
- [34] M. Levoy. A hybrid ray tracer for rendering polygon and volume data. *IEEE Computer Graphics and Applications*, 10:33–40, March 1990.
- [35] W.J Lin, S. M Pizer, and V. E. Johnson. Boundary estimation in ultrasound images. In *Information Processing in Medical Imaging, 12th International Conference, IPMI 91*, pages 285–299, 1991.
- [36] William E. Lorensen and Harvey E. Cline. Marching cubes: a high resolution 3D surface construction algorithm. *Computer Graphics*, 21(4):163–169, July 1987.
- [37] A. Loupas. Digital image processing for noise reduction in medical ultrasonics. *Ph.D dissertation, University of Edinburgh UK*, 1988.
- [38] T. Loupas, W. N McDicken, and P. L. Allan. An adaptive weighted median filter for speckle suppression in medical ultrasonic images. *IEEE Trans on Circuits and Systems*, CAS-36(1):129–135, 1989.
- [39] H.A McCann, J.S Sharp, T.M Kinter, C.N McEwan, C. Barillot, and J.F Greenleaf. Multidimensional ultrasonic imaging for cardiology. In *Proc IEEE*, volume 76, pages 1063–1073, 1988.
- [40] Hans-Peter Meinzer, Kirsten Meetz, et al. The heidelberg raytracing model. *IEEE Computer Graphics and Applications*, pages 34–43, November 1991.
- [41] T. R. Miller, J. B Starren, and R. A. Grothe. Three dimensional display of positron emission tomography of the heart. *J. Nuclear Medicine*, 29(4):530–537, 1988.
- [42] B. Olstad, A. Torp, and J. H. Husøy. A nonlinear loop filter model for video coding. In *IEEE Winter Workshop On Nonlinear Digital Signal Processing*, 1993.
- [43] B. Olstad and H. E. Tysdahl. Improving the computational complexity of active contour algorithms. In *8th Scandinavian Conference on Image Analysis, Tromsø*, 1993.
- [44] Bjørn Olstad. Automatic wall motion detection in the left ventricle using ultrasonic images. In *Proceedings of SPIE/SPSE Electronic Imaging, Science and Technology*, San Jose, 1991.
- [45] Bjørn Olstad. Maximizing image variance in rendering of volumetric data sets. *Journal of Electronic Imaging*, 1:245–265, July 1992.
- [46] J.S Ostrem, A.D Valdes, and P.D Edmonds. Application of neural nets to ultrasound tissue characterization. *Ultrasonic Imaging*, 13(3):298–299, July 1991.

- [47] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on Pattern Anal. and Machine Intell.*, 12(7), July 1990.
- [48] P. Sabella. A rendering algorithm for visualizing 3D scalar fields. *Computer Graphics*, 22(4):51–55, August 1988.
- [49] P. Saint-Marc, J. Chen, and G. Medioni. Adaptive smoothing: A general tool for early vision. *IEEE Trans. on Pattern Anal. and Machine Intell.*, 13(6), June 1991.
- [50] E. Steen and B. Olstad. Scale-space and boundary detection in ultrasonic imaging, using signal-adaptive anisotropic diffusion. In *SPIE Medical Imaging, Image processing conference*, 1994.
- [51] Erik Steen and Bjørn Olstad. Volume rendering in medical ultrasound imaging. In *8th Scandinavian Conference on Image Analysis, Tromsø*, 1993.
- [52] Erik Steen and Bjørn Olstad. Volume rendering in medical ultrasound imaging based on non-linear filtering. In *IEEE Winter Workshop On Nonlinear Digital Signal Processing*, 1993.
- [53] T. Taxt and E. Bølviken. Relaxation using models from quantum mechanics. *Pattern Recognition*, 24:695–709, 1991.
- [54] W. G. Totty and M. W Vannier. Analysis of complex musculoskeletal anatomy using three-dimensional surface reconstruction. *Radiology*, 150:173–177, 1984.
- [55] Craig Upson and Michael Keeler. V-buffer: Visible volume rendering. *Computer Graphics*, 22(4):59–64, August 1988.
- [56] Daniel J. Valentino. Volume rendering of multimodal images: Applications to MRI and PET imaging of the human brain. *IEEE Trans. on Medical Imaging*, 10(4):554–561, 1991.
- [57] M.W Vannier, S. Gronemeyer, and F.R Gutierrez et al. Three-dimensional magnetic resonance imaging of congenital heart disease. *Radiographics*, 8(5):857–871, 1988.
- [58] M.W Vannier, J. L Marsh, and J. O. Warren. Three-dimensional ct reconstruction for cranio-facial surgical planning and evaluation. *Radiology*, 150:179–184, 1984.
- [59] J. W. Wallis, T. R. Miller, C. A. Lerner, and E. C Kleerup. Three-dimensional display in nuclear medicine. *IEEE Trans. on Medical Imaging*, 8(4), 1989.
- [60] S. Webb, R. J Ott, and M. A. Flower et al. Three-dimensional display of data obtained by single photon emission tomography. *Brit. J. Radiology*, 60:557–562, 1987.

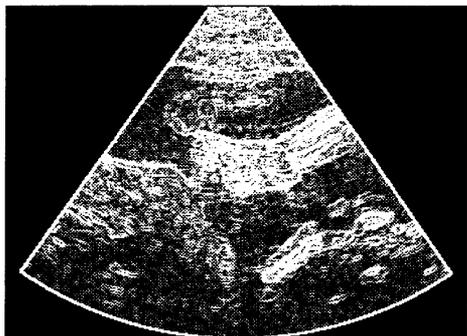


Figure 2: Gradients computed from an original ultrasonic image

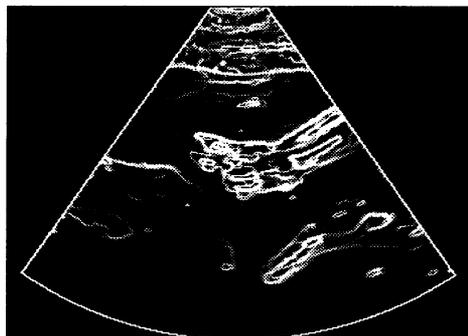


Figure 3: Gradients computed from a filtered image

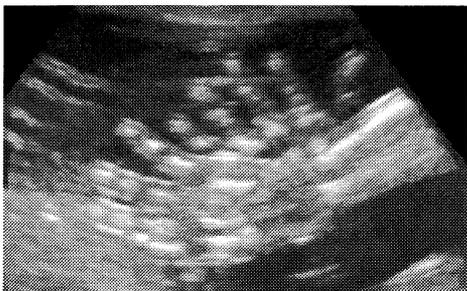


Figure 4: Method 1: The hands of a fetus.

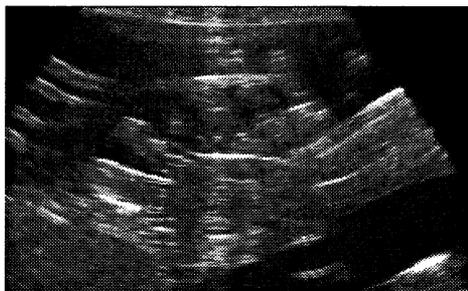


Figure 5: Method 7: The hands of a fetus, unfiltered.



Figure 6: Method 7: The hands of a fetus.

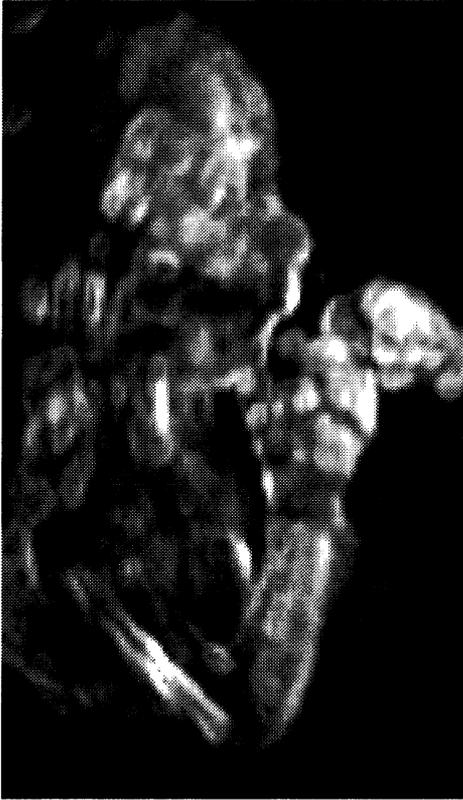


Figure 7: Method 7: Upper part of fetus.



Figure 8: Method 7: Upper part of fetus.



0895-6111(94)00031-X

SEGMENTATION AND VISUALIZATION OF BRAIN LESIONS IN MULTISPECTRAL MAGNETIC RESONANCE IMAGES

Marit Holden,*¹ Erik Steen,[†] and Arvid Lundervold*[‡]

*Department of Pattern Recognition and Image Processing, Norwegian Computing Center, P.O. Box 114
Blindern, N-0314 Oslo, Norway

[†]Department of Computer Systems and Telematics, The Norwegian Institute of Technology,
N-7034 Trondheim, Norway

[‡]University of Bergen, Section for Medical Image Analysis and Pattern Recognition, Årstadveien 19, N-5009
Bergen, Norway

(Received 8 September 1993)

Abstract—In this study we focus on the problem of segmentation and visualization of soft tissue structures in three-dimensional (3D) magnetic resonance (MR) imaging. We introduce a classification method which is a combination of a recently proposed contour detection algorithm and Haslett's contextual classification method extended to 3D. This classification method is used in the classification step of a rendering model suggested by Drebin et al. for visualizing normal and pathological tissue structures in the brain. We evaluate the combination of these two methodologies, and identify some problems which have to be solved in order to develop a clinical useful tool.

Key Words: Magnetic resonance imaging, Segmentation, Visualization, Brain lesions

1. INTRODUCTION

Essential to diagnosis by any medical imaging modality is detectability and locability. During the last decennium, magnetic resonance imaging (MRI) have become one of the most important modalities because of its high soft tissue contrast with multiparameter dependence of signals from tissue voxels (i.e., detectability) and its freedom of slice sectioning together with high spatial resolution (i.e., locability). Even if MRI is able to generate three-dimensional (3D) datasets from a volume of interest, having voxels (volume elements) less than 1 mm³, the MR images are usually stored and displayed as 2D slice images and require the physician to do sequential examination of images and mental 3D reconstruction.

To lighten this burden on the physician and to increase his or hers diagnostic abilities, several investigators and manufacturers have looked into the field of computer graphics and taken advantage of recent techniques of volume rendering which is considered to be the most sophisticated and general method to display volume data in medical imaging (1). Applying volume rendering techniques in a proper way, the following proclamation (2) might become closer to reality: ". . . Three-dimensional medical imaging presents the remotely sensed morphological and physiological patient data in such a way that the physician is relieved

of the chore of mentally reconstructing and orienting the volume and instead can concentrate on the practice of medicine."

Volume rendering techniques have been studied extensively for the past few years. Several algorithms have been proposed (3-11) and some have been evaluated for magnetic resonance imaging and X-ray computed tomography (CT) (12-14). Many of the reported methods have shown to give high quality renderings for several applications. One approach, which is the one considered here, is based on a partition (binary or probabilistic) of the 3D dataset. To provide useful information from the 3D dataset the partition of the imaged voxels should be computed as close as possible to the "true" anatomical or pathological structures (lesions) which are embedded in the 3D patient voxel space. To achieve this, *contrast-enhancing agents* (i.e., pharmaceuticals used to increase contrast in diagnostic images by altering the physical characteristics of target tissue) are often given to the patient previous to image acquisition. Then, given the image data, the partition step essentially involves *image segmentation*, which is the main concern of this paper.

Gray level segmentation, which can be either binary (crisp) or probabilistic (fuzzy), is performed to display the structures or lesions of interest from the imaged volume. Using a ray-casting technique, the rendering can be defined according to a particular function $f: (i, \rho) \rightarrow \mathbf{R}$ of each pixel i on the screen and the voxel gray level values met by the ray ρ through

¹ To whom correspondence should be addressed.

the investigated volume. The set of transmitted rays can be cone shaped or be parallel, as in this study. When the MR acquisitions are multispectral, which means that we have a set of registered slices or volumes to our disposal, the power of *multispectral and contextual statistical tissue classification* techniques could be applied at the segmentation step of the transmission oriented volume rendering.

In this paper we have combined the classification strategy presented in (15) with a rendering model suggested by Drebin et al. (5) for visualizing soft tissue structures and lesions in the brain from multispectral MR acquisitions. The rendering model is based on probabilistic classification of the multi spectral signal intensity values into different tissue categories, taking spatial context into account. Each tissue is assigned a color where the color mixture in each voxel reflects the amount of different tissues present. The multispectral 3D anatomical volume was obtained through the sequential acquisition (with and without a contrast-enhancing agent) of spatially nearly contiguous and aligned slices piled on top of each other. In (16) visualization was combined with classification results from (15). The results suffered among other things from many misclassifications to tumor. In this work we have concentrated on methods which reduce the occurrence of misclassification. The results reported are preliminary in that only one patient with a tumor is studied from which the interslice spacing of the 18 multispectral 2D acquisitions was large (2.5 mm) as well as the slice thickness (5 mm). Thus, the data set is suboptimal with respect to the full strength of our 3D contextual classification technique.

The rest of the paper is organized as follows.

A brief overview of volume rendering concepts and methods is given in Section 2.2, emphasizing the model suggested in (5). The main contribution of this paper, introduction of a 3D contextual and multispectral classification procedure in volume rendering, is thoroughly described in Section 2.1 (and more details are found in (15)). Our experimental results together with clinical data and the MR imaging protocol, is presented in Section 3. Finally, in Section 4, we discuss the combination of these two methodologies of classification and rendering, and identify some problems which have to be solved in order to develop a clinical useful tool.

2. METHODS

The main issue of this paper is tissue characterization in MR images and how this may be used in connection with visualization of MR-data. In the following sections we will describe and give a background

for methods used in the experiments described in Section 3.

2.1 Tissue characterization

The notion of *image segmentation* is a key step in image analysis and can be defined as "a process which typically partitions the spatial domain of an image into mutually exclusive subsets, called regions, each one of which is uniform and homogeneous with respect to some property such as intensity or texture and whose property value differs in some significant way from the property value of each neighboring region" (17). In medical applications the different regions may correspond to different tissue types. Two main segmentation techniques are relevant to our study; clustering and classification.

The simplest way of segmenting an image is by saying that pixels with grey levels in a certain interval belong to one region, the others belong to other regions. This method is called *thresholding*. Thresholding is often used as a segmentation method for X-ray CT images because the Hounsfield units are such that certain intervals correspond to certain tissue types. When it comes to MRI, thresholding is not appropriate. The grey levels of a certain tissue varies between examinations. In addition some tissues are well separated in one channel (i.e., pulse sequence with its timing parameter values) while others are better separated in another channel. In this case we then take advantage of the multispectral nature of MR imaging. For each voxel we then obtain a vector of gray level values—one value from each channel. This vector is called a feature vector. In addition to the component values obtained from the image acquisitions, we may use texture measures or other features computed from the different channels to increase the number of components in each feature vector. A qualitative attribute of image texture can be a property of *fineness, coarseness, granulation, smoothness, randomness, lineation* or the texture can be described as *mottled, irregular, speckled, lamellar or striped* among others [see (18)]. In contrast to conventional X-ray radiology and medical ultrasound, the relevance of texture in diagnostic processing of MR images is not thoroughly investigated.

To perform feature-based segmentation both classification and clustering techniques can be used.

Let us first consider the method of *clustering*. The situation here is that we have available a set of feature vectors and the task is to partition these vectors into "natural" subsets (clusters) based only on rules, a similarity measure, and the structure in the data. There is no "supervisor" which tells the true number of clusters, nor give any prior cluster assignment information. One of the most commonly used clustering methods is the

k-means algorithm and its variants (19–21) which can be roughly described as follows:

1. Begin with an arbitrary set of k cluster centers and assign each feature vector to the cluster with nearest center.
2. Compute the sample mean of each cluster.
3. Reassign each feature vector to the cluster with nearest mean.
4. If the assignment of samples to clusters has not changed, then stop; else: go to step two.

Let us next consider the method of *classification*. Here we have a predefined set of K classes (tissue types) and a sample of feature vectors from each class. This is called a *training set*. We assume that feature vectors coming from a certain class are distributed according to a certain probability density. The density is estimated from the samples of feature vectors from the corresponding class. This and other a priori information is used to design a classifier. This classifier is then applied to novel acquisitions to perform tissue classification. It is this segmentation method of classification we will discuss and use in the rest of this paper.

2.1.1 Training–estimation. The purpose of training is to build a classifier, incorporating for instance estimates of multinormal class conditional probability densities for each of the predefined classes. During training, a set of labeled pixels is accumulated from the MR images dedicated to the training phase. Each of these training pixels is associated with a class label and a feature vector. The feature vector is given by the feature extraction method. In examination of the head one might use the intensity values of the pixel in each of the image channels as the feature vector components. The class label of a pixel is decided by the trainer who employs general medical knowledge and specific diagnostic information about the patient and the MR examination.

The training part can be facilitated by employing unsupervised classification techniques such as *k*-means clustering of the channel images combined with supervised labeling of the segmented clusters. This was the approach in (22).

REMARK. In MR-images the mean gray level of a given tissue in a channel can show great variability from one acquisition to another, even if extrinsic parameters are kept constant (23, 24). Thus, to train and apply the classifier to different examinations this effect has to be compensated by performing *calibration* of the channel images. This can be done either manually or by automated techniques for updating of parameter estimates. In (25) an automated parameter updating technique was used for solving this problem. This approach seems to be very promising.

2.1.2 Test classification. Let us first describe the situation. We have a data set $y = y_1, \dots, y_N$ of voxel-based feature vectors which should be classified to the classes C_1, \dots, C_N , where C_k is one of K predefined classes found in the training procedure. There are two main methods for determining C_1, \dots, C_N . The first one is to consider the joint posterior probability $P(C_1, \dots, C_N | y)$ and compute C_1, \dots, C_N which optimize this expression. Examples of this approach may be found in (26) and (27) (simulated annealing).

The second approach, which is the one we will concentrate on here, is to compute $P(C_i = k | y)$ for all classes k and choose C_i equal to the class k which maximizes this term. This corresponds to minimizing the expected error rate. How this $P(C_i = k | y)$ is computed depends on which classification model we use. In a non-contextual classification model we make use only of the extracted features related to the voxel of interest when we classify the voxel, that is $P(C_i = k | y) = P(C_i = k | y_i)$. $P(C_i = k | y)$ is then computed as follows:

$$P(C_i = k | y) = P(C_i = k | y_i) = \frac{\pi_k f_k(y_i)}{\sum_{k=1}^K \pi_k f_k(y_i)}$$

where f_k is the probability density distribution for class k and π_k is the a priori probability of having class k .

In a *contextual* classification model, we make in addition use of the extracted features in a spatial neighborhood, δ_i , of the voxel of interest, that is $P(C_i = k | y) = P(C_i = k | \delta_i)$ (i.e. a Markov property is assumed). Contextual decision rules can also make use of the fact that the voxels of MR images is usually much smaller than the size of the tissue structures one wish to study. In most cases, adjacent voxels are likely to have similar tissue constituents, and field of view, acquisition matrix and anatomy constrains the probability of two given tissue classes being neighbors. Let us now study one such specific contextual model.

Haslett's model. In Haslett's model (28, 29) the neighborhood consists of the four nearest neighbors, $y_{iS}, y_{iN}, y_{iW}, y_{iE}$, which are assumed conditionally independent. Further, isotropy and stationarity is assumed. $P(C_i = k | y)$ is then computed as follows:

$$P(C_i = k | y) = P(C_i = k | y_i, y_{iS}, y_{iN}, y_{iE}, y_{iW}) \\ = \frac{\pi_k f_k(y_i) T_k(y_{iS}) T_k(y_{iN}) T_k(y_{iE}) T_k(y_{iW})}{\sum_{k=1}^K \pi_k f_k(y_i) T_k(y_{iS}) T_k(y_{iN}) T_k(y_{iE}) T_k(y_{iW})}, \quad (1)$$

where

$$T_k(y_j) = \sum_{l=1}^K \pi(l|k) f_l(y_j), \quad (2)$$

and $\pi(l|k)$ is the transition probability of having class l in a pixel given that its neighbor is k . The classification is then done by using Bayes decision rule (19, 29) with minimum expected cost of misclassification. To make better use of the voxel information when dealing with a 3D data set or a stack of nearly contiguous 2D images, we have made a simple generalization of Haslett's 2D model to 3D.

This we do by using information from the slice on each side of the slice we want to classify. We then get two additional nearest neighbors y_{iD} and y_{iU} . The equation for computing $P(C_i = k|y)$ then becomes:

$$\begin{aligned}
 P(C_i = k|y) &= P(C_i = k|y_i, y_{iS}, y_{iN}, y_{iE}, y_{iW}, y_{iD}, y_{iU}) \\
 &= \frac{\pi(k)f_k(y_i)T_k(y_{iS})T_k(y_{iN})T_k(y_{iE})T_k(y_{iW})T_k(y_{iD})T_k(y_{iU})}{\sum_{k=1}^K \pi(k)f_k(y_i)T_k(y_{iS})T_k(y_{iN})T_k(y_{iE})T_k(y_{iW})T_k(y_{iD})T_k(y_{iU})}
 \end{aligned}
 \tag{3}$$

where $T_k(y_i)$ is defined in Eq. (2).

In addition to this 3D generalization of Haslett's model we use a recently developed contour detection model (30-33) to compute the outer border of brain parenchyma. A priori we know that an intracerebral tumor with edema do not occur outside this border. This information is then used in the classification scheme by excluding the tumor class and the edema class from the possible classes when we classify the extracerebral voxels.

2.1.3 Contour detection. Recognition of the outer boundary of the brain consists of finding an appropriate closed curve representing the border between brain parenchyma and the subarachnoid CSF. The traditional way of finding boundaries has been to detect edges (typically by defining a threshold value of some gradient operator and select all locations with values above the threshold value as potential edges) and then link these together. For such methods, there is no guarantee of obtaining one closed curve.

Recognition of closed curves in medical images is a topic that has received much attention in recent years. A well studied application is recognition of the endocardial border of the left ventricular cavity in ultrasound images. Many of the techniques developed for this particular problem may, however, be used in other applications with minor adjustments. Common to many of the techniques developed has been the use of a Bayesian approach through specification of a priori distributions of boundary features such as shape, size and smoothness. Combining the a priori distribution with a conditional distribution for the data (given the boundary), the *posterior* distribution is calculated. The *maximum a posteriori* (MAP) solution is then used as

an estimate of the boundary. The approaches used differ from each other in mainly three aspects.

- The representation part, which specifies how the contour $x \in X$ is formally represented (e.g., X is the sample space of linked cyclic lists of nodes with fixed internode distance where the number of nodes are stochastic; 31).
- The modeling part, which include properties of the contour (such as smoothness and shape) and a model for the observed data (images) $y \in Y$. The models, which fall into the Bayesian framework (34), are defined through energy-functions $U_1, \dots, U_n : X \times Y \rightarrow \mathbf{R}$.
- The numerical algorithm for finding the optimal contour (the MAP solution), defined to be the one with minimal energy.

Friedland and Adam (35) assumed a star-shaped region model where the contour is star-shaped and is described by a set of radii with respect to a given centerpoint. The contour is then specified through the length of each radius. In (35) an a priori model including both spatial and temporal smoothing was used together with a gradient-operator for recognizing the radius-lengths. An algorithm based on simulated annealing was defined for finding the MAP solution.

Active contours (36-39) may be thought of as a generalization of the star-shape-representation. In this case the contour is represented through a set of vertices with straight lines or parametric curves in between. For each vertex, a set of possible positions can be defined. The models used for this approach include spatial smoothness or stretching and bending energy of the curve in the a priori models and gradient operators for the datamodels.

Storvik (30, 31) introduced yet another approach where the simply closed curve is supposed to follow the edges of the pixels in the observed image. Spatial characteristics may be included in the a priori models. For the datamodel, class-descriptions of the regions inside and outside the contour are used. An algorithm based on simulated annealing was developed for finding the MAP solution in such a way that the curve will iteratively move towards the optimal solution.

In the experiments described later we chose the approach by Storvik mainly because this approach was available for us at the present time. Below we will therefore give a somewhat more detailed explanation of this method. A full description of the contour representation and optimization algorithm is given in (31) and (30). We will here only discuss the model that has been used.

We assume a region model where the planar brain image is divided into two regions. These regions are outside brain ($r = 1$) and inside brain ($r = 2$).

Assume x denote the contour and y the observed image. The energy is then defined by

$$E = U_1(x) + U_2(x; y), \quad (4)$$

where U_1 is an energy-function giving low energy for smooth curves while U_2 is an energy-function describing the distribution of the grey-levels in the image. In this work we have chosen U_1 to be related to the “fractal” property of the closed contour by

$$U_1(x) = \beta \frac{(\text{length of contour})^2}{\text{area inside contour}}, \quad (5)$$

where $\beta \in \mathbf{R}$ is a smoothing parameter. Regarding the data, we assume the image consists of 13 tissue classes, *air and bone, fat, connective tissue, CSF, CSF + gray* (three different classes for voxels which are mixels of CSF and gray matter), *tumor* (three different classes), *edema and white matter and gray matter* making up the brain parenchyma. Further, we have assumed region 1 to contain *air and bone, fat, connective tissue, CSF and CSF + gray*; region 2 to contain *brain parenchyma, edema and tumor*.

Each class $k = 1, \dots, 13$ is assumed to have a multivariate Gaussian distribution with expectation μ_k and covariance matrix Σ_k . For the two-region model, the likelihood function in this case will be

$$f(y|x) = \prod_{r=1}^2 \prod_{i \in R_r} \sum_{k \in C_r} \pi_k N(y_i; \mu_k, \Sigma_k). \quad (6)$$

where R_r is the set of pixels inside region r and C_r is the set of classes that appears in region r , and π_k is the prior probability of class k . We define the energy-function U_2 as the log-likelihood

$$U_2(x; y) = -\log(f(y|x)). \quad (7)$$

$$= \sum_{r=1}^2 \sum_{i \in R_r} \log \left(\sum_{k \in C_r} \pi_k N(y_i; \mu_k, \Sigma_k) \right). \quad (8)$$

Given the energy-functions, we seek to find the minimum energy curves, corresponding to the maximum a posteriori (MAP) estimates of the curves. In (31) and (30) an iterative algorithm based on simulated annealing was constructed for solving this optimization problem.

2.2 Volume Rendering

Volume rendering is a set of techniques for obtaining a 2D representation of the interior of a volumetric data set. This is useful in medical imaging because it may increase the physicians diagnostic ability by improving his/her mental 3D reconstruction of the data set. The volumetric data set is made of samples from a 3D scalar field, or vector field in the multispectral case, and is usually represented as a 3D array.

In this study, we assume that the 3D data is represented as a cuberille, that is as a 3D array of equal sized cubic voxels.

A common way to visualize 3D data is by interactive slicing of the cuberille. Thin slices can be displayed directly as intensity images. By successive rendering of parallel slices, the viewer can at least mentally get an overview of the volumetric data set.

Another well known technique is threshold rendering. This technique has been found useful for display of isovalue surfaces from X-ray CT, where each tissue falls within a specific density range (Hounsfield units). This is not the case for MR data. The surfaces can be rendered by using back to front traversal (3), ray tracing (40) or surface reconstruction (4).

Experiments with volume rendering techniques on MR data are numerous [see (41–43)]. In (41) volume rendering is used to validate the result of a segmentation technique based on iterative thresholding. In (42) the aim is to visualize the ventricle system of the brain. In (43) volume rendering is used in reconstruction of the brain coupled with interactive “electronic dissection.”

In the next section we will consider a specific rendering algorithm which was used in our experiments.

2.2.1 The rendering algorithm by Drebin and co-workers. Drebin et al. (5) have developed a technique for rendering of datasets which represent a mixture of tissues. This paper has already become a “classic” and has been used by several others (see for instance (44)). Much effort is paid to avoid artifacts, and all-or-none decisions based on thresholding are avoided. Such decisions tend to produce artificially sharp boundaries and do not reflect on the diffuse transitions and fine detail often found in real world applications. Drebin et al. demonstrate their method on X-ray CT data and also for several other applications. The steps in the volume rendering process applied to medical images, are as follows [for more details see (5)]:

1. *Classification. tissue percentage volumes* are computed by using statistical analysis to decide the amount of each tissue present within each voxel. In this paper the tissue percentage volume is set equal to $T(P(C_i = k|y))$. Here $T: [0,1] \rightarrow R$ is a suitable transform (see Section 3 for a detailed explanation of this transform) and $P(C_i = k|y)$ is the value of the conditional probability for class k at voxel i given the data, as described in Section 2.1.

Then a color and an opacity value is assigned to each tissue. Each voxel will then contain a mixture of colors individually weighted by the probability that the given tissue is present in the voxel. The output of this stage is called the *color volume*. In addition so-called

matting volumes can be made to smoothly remove portions of the volume occupied by one or more tissues. The matting volumes are made by specifying tissue categories which should be made transparent.

2. *Surface determination.* A density $\rho_k \in R$ is assigned to each tissue, $k = 1, \dots, K$. The composite density, D_i , of a voxel, $i = 1, \dots, N$, will then be

$$D_i = \sum_{k=1}^K T(P(C_i = k|y))\rho_k$$

where T is the transform mentioned above. These voxels will then form a *density* volume. The boundaries between tissues can then be detected by calculating gradients in each voxel of the density volume. The magnitude of the gradient is stored in a *surface strength volume*, and the direction of the gradient is stored in *surface normal volumes*. A simple nearest neighbor calculation is suggested for the gradient calculation.

3. *Shading.* Shading is based on a lighting model which takes into account the position and color of the light sources, the position of the eye, the *surface normal volumes*, the *surface strength volume* and the *color volume*. The voxels may be luminous and emit outgoing light, they may act as translucent filters or they may contain surface scatters. Only a single scattering of light from the light source to the eye is assumed. The reflected surface color is a function of the surface normal, the strength of the surface, the diffuse color of the surface, the direction and color of the light source and the eye position. If no surface is present, no scattering will take place.

4. *Transformation.* After shading the volume, it is re-sampled and transformed into the viewing coordinate system. Re-sampling and transformation is done to avoid visual artifacts, and is preferred to ray-tracing.

5. *Projection.* The transformed volume is finally projected onto the view plane. This is done by computing a weighted sum of voxels on columns parallel to the viewing direction.

3. RESULTS OF PRELIMINARY EXPERIMENTS

3.1 MR data

In the experiments we employed MR acquisitions from the Norwegian Radiumhospital (1.5 T GE Signa scanner) (15). We selected one patient from a set of 36 patient examinations (referred to as DNR.A.03 in (15)). The patient was a 53-yr-old man with a metastatic brain tumor from a bronchogenic carcinoma. We obtained 18 axial slices from this patient, but only slices 3–17 were used. The slice thickness was 5 mm, the interslice

distance was 2.5 mm, the pixel size about 1 mm \times 1 mm using a 256 \times 256 display matrix.

For each slice, four geometrically equivalent channels were obtained representing different SE sequences pre and post Gd contrast. Three pre contrast channels were acquired with T2 (2000/80), T1 (700/20) and proton density-weighting (2000/20), respectively, and one T1 weighted channel was acquired after injection of Gd-DTPA.

The images were calibrated as described in (15).

T1-images (with an outer brain contour superimposed) of four slices from the selected patient examination are shown in the middle column of Fig. 2 and all four channel images of slice 14 are shown in Fig. 1.

3.2 Training, Classification and Visualization

Training. To build a classifier we need descriptions of the different classes. The class descriptions used for classification of the images here were those used in (15). These class descriptions were created from parameter estimates derived from marked regions (masks) in calibrated images, with some modifications (see below). Calibrated images from 11 different patients were used in this process.

The following classes were modeled exclusively from training: *white matter, gray matter, fat, CSF, edema, connective tissue.* Air was modeled as a uniform distribution, with limits set arbitrarily after inspection of a set of feature vectors from air-pixels. An *outlier* class was also introduced, modeled as a uniform distribution with pixel values ranging from zero to 400 in all channels (all pixel values in the data set were smaller than 400).

In addition, there was a tumor class derived from training only, on contrast-positive tumor tissue. In initial classifications (using patients which were not part of the test set), this class description was found to give an unacceptable level of misclassification to *tumor*. Therefore, some modifications were made and the tumor class was split up into three different tumor classes. Details of how this was done are given in (15).

To further reduce the problems of misclassification, some additional modifications were made. It was observed that misclassification chiefly occurred subcortically (i.e., in the transition zone between gray and white matter), and on the cerebral surface (i.e., in the transition between gray matter and CSF). Therefore, classes simulating these transition pixels (pixel originating from a voxel which contains two different tissues) were made by simple interpolation.

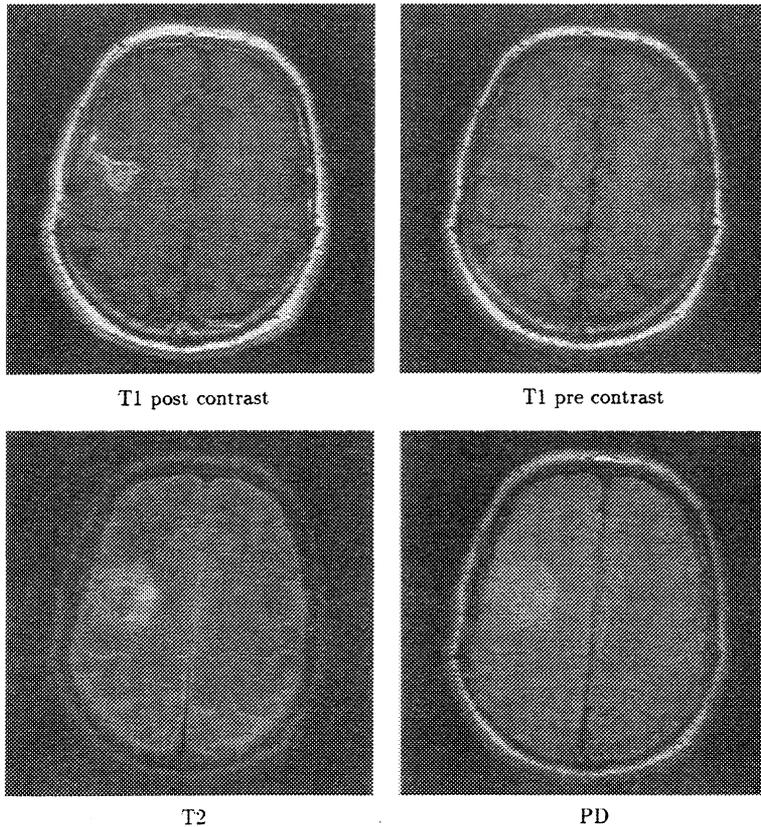


Fig. 1. Example of a slice (No. 14) of MR-images of the brain.

Classification and visualization. We started by classifying the images with Haslett's classification method in 2 dimensions, just as was done in (15). In the experiments the transitions probabilities $\pi(k|l)$ were chosen such that $\pi(k|l) = 0.9$ if $k = l$, $\pi(k|l) = 0.1/(K - 1)$ otherwise; $k, l = 1, \dots, K$. A 0-1 loss function was used and all prior probabilities, π_k , were set equal to $1.0/K$. The classes used during classification and rendering are described in Table 1.

Classification results from four different slice levels are shown in the left column of Fig. 2. Note that voxels from the normal tissue categories are properly classified with only a small number of misclassifications. Further, almost all voxels from the tumor are correctly classified to one of the 3 tumor classes. However, misclassifications were present in other parts of the brain. In particular, several voxels lying close to extracerebral fat are misclassified to tumor.

We then found the posterior probabilities $P(C_i = k|y)$, computed as in the Haslett classification al-

gorithm described above, but now with student distribution instead of normal distribution (cfr. (15) where the results indicate that this is better when probability images are made). These posterior probabilities were used as input to the visualization algorithm. First tissue percentage volumes were made by transforming the posterior probability values such that they became suitable for the display. Several transforms were tested in (15), and the following transform was suggested:

$$T(P(C_i = k|y)) = a - c(\log(\max_{i,k}\{P(C_i = k|y)\}) - \log(P(C_i = k|y))), \quad (9)$$

where a denote the maximum display value (=255) and c was set to 10 according to the recommendations in (15). In addition $\max_{i,k}\{P(C_i = k|y)\}$ was 1.0 and the tissue percentage volumes were computed as follows:

Table 1. Class information

Class	Tissue type	Color	Distribution	PD	T2	T1 pre-contrast	T1 post-contrast
1	White matter	Yellow/white	Student Normal	100	40	91	85
2	Gray matter	Gray	Student Normal	103	41	83	79
3	White matter		Student Normal	105	42	76	72
4	Gray matter	Gray	Student Normal	100	51	66	67
5	CSF		Student Normal	98	56	61	64
6	Gray matter	Gray	Student Normal	96	60	56	61
7	CSF		Student Normal	91	70	46	56
8	Edema	Green	Student Normal	114	70	80	76
9	Tumor1	Red	Student Normal	101	41	71	92
10	Tumor2	Red	Student Normal	105	56	72	106
11	Tumor3	Red	Student Normal	107	58	70	141
12	Connective tissue	Brown	Student Normal	55	22	80	65
13	Fat	Ochre	Student Normal	98	25	158	154
14	Air	Black	Uniform	0-15	0-7	0-15	0-15
15	Outlier	White	Uniform	0-400	0-400	0-400	0-400

In the color column the colors used in the classified images are given. The values in the PD column are the mean values in the PD images for that tissue type if it is Student distributed. If it is uniformly distributed it is the interval in which the distribution is non-zero (if it is within the interval of the other channels too). Similarly for T2 and T1 pre- and post-contrast. See (15) for more information about the classes used (like f.ex. covariance matrices).

$$T(P(C_i = k|y)) = \begin{cases} 0 & \text{if } P(C_i = k|y) = 0 \text{ or} \\ & 10 \cdot \log(P(C_i = k|y)) < -255 \\ 255 + 10 \cdot \log(P(C_i = k|y)) & \text{otherwise} \end{cases} \quad (10)$$

Because of a nonnegligible interslice distance, we employed bicubic interpolation to form each of the tissue percentage volumes. Each tissue was then given a color value, a density value and an opacity value, and a color volume was formed. The opacity values were set low for the outer structures in order not to occlude inner ones. Also the density values were set equal in each tissue. Therefore, no surface shading was performed. Matting volumes were made to reveal interior structures of the brain. Instead of transforming the volume into the view-plane, we extracted tri-linearly interpolated samples with a ray-tracer. No additional visible artifacts were introduced by using this method. A much more serious limitation, however, was the low spatial resolution.

Several renderings were made from different viewing points, showing fat, CSF, brain parenchyma

and the tumor with surrounding edema (Figure 3). Generally, the quality of these renderings suffer from the poor spatial resolution in the axial direction (due to the large slice thickness and interslice distance compared to the in plane pixel size).

The substantial number of misclassifications of voxels to tumor for the Haslett 2D classification (see left column of Fig. 2) are clearly visible in these images (Fig. 3b). Especially in the outer parts of the brain, artificially high probability density values for tumor have been assigned to a large set of voxels. The actual tumor can however be clearly seen with its surrounding edema in the right fronto-parietal region of the brain.

We observed that the rendered images suffered from the same misclassifications to tumor as observed in the Haslett classified images. We therefore tried to avoid these misclassifications by using a priori information that there should be no tumor or edema outside the brain. In addition we employed data also from slices above and below the slice of interest. To incorporate a priori information of intracerebral tumor localization we partitioned the imaged voxel slice into two regions, one inside and one outside the brain. To obtain this segmentation we utilized the

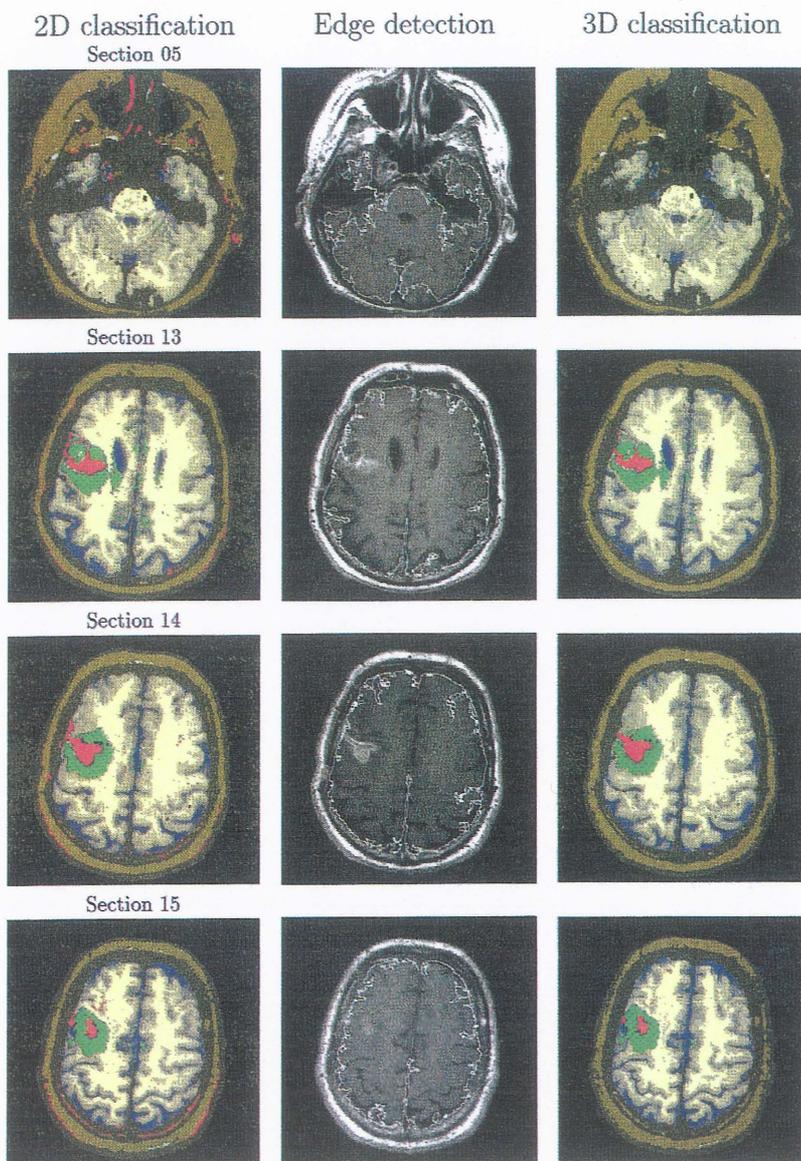


Fig. 2. Classified images and contour detection results.

automatic *contour detection* algorithm described in Storvik (30–33). In the contour finding procedure we assumed that white matter (1), gray matter (3), edema (8) and tumor (9–11), respectively, could occur inside the contour and that CSF (7), CSF + gray (4–6), connective tissue (12), fat (13) and air (14)

could occur outside the contour. The result of contour detection is shown in the middle column of Fig. 2. For the air class used here, the normal distribution is assumed because the contour detection algorithm was implemented only for that case. We assume that this was of no importance for the result.

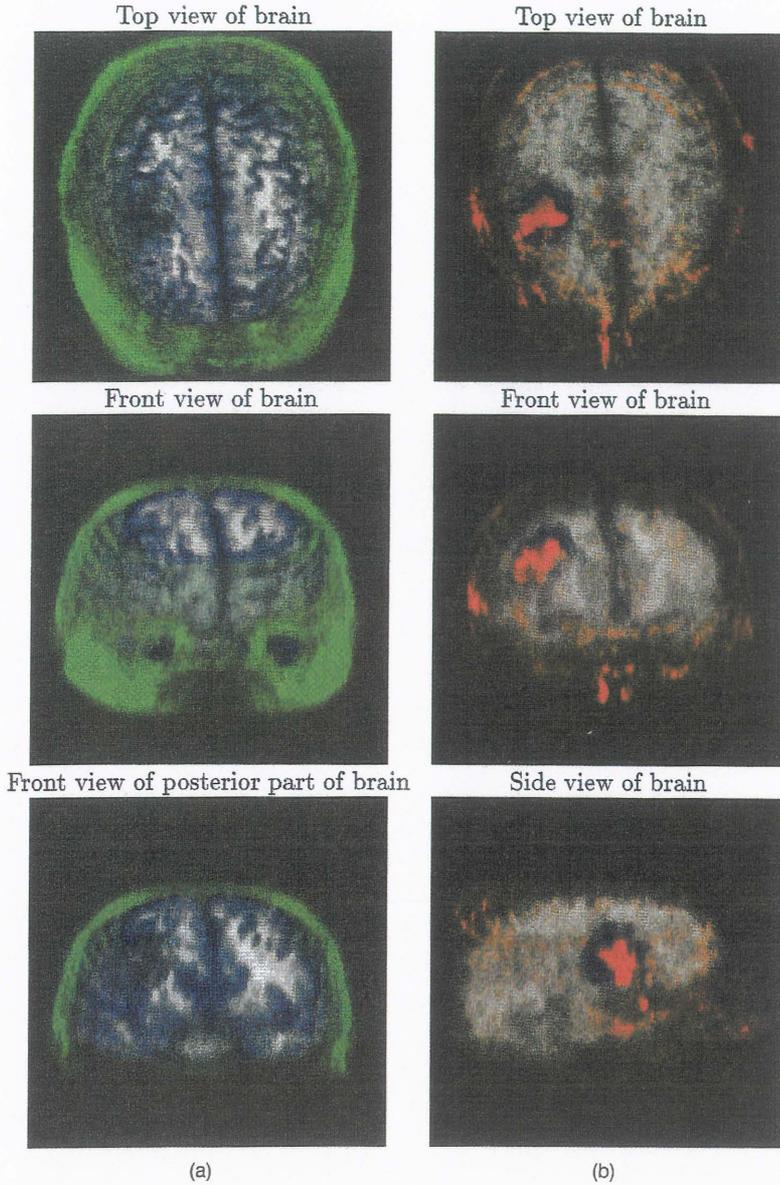


Fig. 3. (a) Visualization of fat, CSF and parenchyma in the brain. The color values are chosen such that fat is green, CSF is blue, and parenchyma is gray. Fat is given low opacity in order not to occlude inner brain parenchyma. (b) Visualization of tumor, surrounding edema and parenchyma in the brain. The color values are chosen such that tumor is red, edema is blue, and parenchyma is gray. The images were classified using Haslett 2D contextual classification. Note the misclassification of voxels to tumor in the peripheral part of the volume, but good localization of the tumor proper with surrounding edema in the right fronto-parietal region.

The two regions were then classified separately with *Haslett's* classification algorithm generalized to *three dimensions*. The interior of the brain was classified to all classes 1–15, and the extracerebral region was classified to all classes except tumor and edema. The two classification results were then merged into one classified image. The result of this classification is shown in the right column of Fig. 2. There we observe that the combination of contour detection and 3D classification gives a much better result than the use of 2D classification alone. We also observe that inside the brain we have avoided some misclassifications to tumor, but not all.

4. DISCUSSION

In this work we have introduced an improved multispectral and contextual statistical classification procedure (15) to the rendering technique of Drebin and coworkers (5) for effective visualization of MR image volumes.

The volume data we used in the preliminary experiments were acquired from 15 parallel slices from one patient. The interslice distance was large (2.5 mm) as well as the slice thickness (5.0 mm), compared to the in plane pixel size (1×1 mm). The chosen patient was from the training set, but was used here because he had a large tumor in the classified image, a good classification result (no significant magnetic field inhomogeneities) and there were problems near fat with misclassification to tumor. All this made this dataset suitable for visualization, and for showing how to improve the classification result by combining contour detection and classification. Other studies show that classifying the test set work quite well, see for instance (22). We observed that *Haslett's* 2D classification gave reasonable results except for several misclassifications to tumor, especially outside the brain. Our main contribution is improvement of this result from standard slice by slice classification by using anatomical and pathological a priori information together with additional contextual information in our classifier. The first improvement was obtained by using a contour detection technique to define the outer border of the brain. Then the interior and the exterior region relative to this contour were classified separately, where tumor and edema were excluded from classification of the exterior region. By this, most misclassifications outside the brain disappeared. Small improvements were also made by extending *Haslett's* method to 3D. The rendering method suggested by Drebin (5) gave very promising results for rendering of normal tissue structures, but misclassifications in the 2D *Haslett* classified images were clearly visible in the rendering result.

We should expect improvements in the results if more densely sampled voxels were obtained from the investigated volume, such as 3D-acquisitions. Three-dimensional Fourier MR imaging have several advantages to 2D MRI (45). First there is better signal-to-noise ratio (SNR), since with each measuring pulse the whole volume sample contributes to the signal. In planar 2D-imaging only the nuclear spins of one slice produce the signal, while the whole volume contributes to the noise. Next there is an optimum spatial resolution in all direction.

Further improvements of classification could be achieved by better calibration of the images, by using noise filtering (46) or employ even more a priori information. Another approach is to use mixture classes in the classification, that is classes which consist of a mixture of base classes (see f.ex. (47)). Such a mixel model is relevant to the rendering method used in this work, because one step in the method consists of computing tissue percentage volumes.

Another strategy towards extraction and visualization of diagnostic information could be to introduce measures like "malignancy" and "fluidness" in Drebin's density function δ . Other diagnostic problems than brain tumor could also be subject to our approach. One example could be the intracerebral ventricular system with CSF and its surrounding structures, which is relevant to conditions of hydrocephalus and brain atrophy. Regarding quantitative measurements, the segmentation techniques may easily be extended to compute volumes and location of tumor or to compute the volume of the ventricle system or other relevant anatomical structures.

To really evaluate the methods described in this paper, more experiments should be performed on better 3D data. From our preliminary experiments we have indicated that there is a valuable potential in the described methods. However, these should be developed further to make them a clinical useful tool.

DISTRIBUTIONS

Here we describe the multivariate distributions used in the experiments.

Normal Distribution

The multinormal distribution is given by

$$f(x; \mu, \Sigma) = (2\pi)^{-d/2} |\Sigma|^{-1/2} e^{-1/2(x-\mu)^T \Sigma^{-1}(x-\mu)},$$

where μ is the mean feature vector and Σ is the covariance matrix.

Student Distribution

Sometimes the distributions have too heavy tails to make the normal distribution a reasonable assumption.

tion. Then a student distribution is a better approximation to the data. One possible definition of a multistudent distribution is found in Johnson & Kotz (48). There it is defined as follows:

Y is multistudent distributed with ν degrees of freedom, $\nu > 2$, if

$$f_Y(y) = \frac{\Gamma\left(\frac{\nu+m}{2}\right)}{(\pi\nu)^{\frac{m}{2}} \Gamma\left(\frac{\nu}{2}\right) |R|^{1/2}} \times (1 + \nu^{-1}(y - \mu)R^{-1}(y - \mu))^{-\frac{\nu+m}{2}}$$

where $Y^j = \frac{x^j}{S} \sqrt{\nu}$, $i = 1, \dots, m$, $X \sim N(\mu, R)$ and $S \sim \chi_\nu$ and independent of X^1, \dots, X^m .

Uniform Distribution

The multiuniform distribution may be defined as follows:

$$f(x; a, b) = f((x)_1, \dots, (x)_p; (a)_1, \dots, (a)_p, (b)_1, \dots, (b)_p) = \begin{cases} \prod_{i=1}^p \frac{1}{(b)_i - (a)_i} & \text{if } (a)_i \leq (x)_i \leq (b)_i \text{ for all } i, \quad 1 \leq i \leq p; \\ 0 & \text{otherwise.} \end{cases}$$

SUMMARY

Essential to diagnosis by any medical imaging modality is detectability and locability. During the last decennium, magnetic resonance imaging (MRI) have become one of the most important modalities because of its high soft tissue contrast with multiparameter dependence of signals from tissue voxels (i.e., detectability) and its freedom of slice sectioning together with high spatial resolution (i.e., locability). In this study we focus on the problem of tissue characterization and visualization of soft tissue structures in 3D MRI. We introduce a classification method. The method is a combination of a contour detection algorithm proposed by Storvik et al. (30, 31) combined with Haslett's classification method (28, 29) which we extend to 3D. This classification method is used in the classification step of a rendering model suggested by Drebin et al. in (5) for visualizing soft tissue structures from the central nervous system. We evaluate the proposed segmentation algorithm and the combination of the two methodologies. We also discuss some problems which have to be solved in order to develop a clinical useful tool. Preliminary experiments were performed on 15 slices from the brain of one patient with a tumor. The experiments show that the methods have a potential for being part of a useful tool for diagnosing, but the meth-

ods should be tested more thoroughly on better 3D data before firm conclusions can be drawn.

Acknowledgments—This study has been made possible by financial support from The Norwegian Research Council, NTNf section. We thank Geir Storvik for contributing to the section on contour detection algorithms. Images were obtained from the Norwegian Radium Hospital.

REFERENCES

1. Barillot, C. Surface and volume rendering techniques to display 3D data. *IEEE Engineering in Medicine and Biology Magazine* 12(1):111-119; 1993.
2. Styzt, M. R.; Frieder, G.; Frieder, O. Three-dimensional medical imaging: Algorithms and computer systems. *ACM Comput. Surveys* 23(4):421-499; 1991.
3. Frieder G.; Reynolds, R. Back-to-front display of voxel-based objects. *IEEE Comput. Graphics Applications* 5(1):52-60; 1985.
4. Lorensen, W. E.; Cline, H. E. Marching cubes: a high resolution 3D surface construction algorithm. *Comput. Graphics* 21(4):163-169; July 1987.
5. Drebin, R. A.; Carpenter, L.; Hanrahan, P. Volume rendering. *Comput. Graphics* 22(4):65-74; 1988.
6. Upson, C.; Keeler, M. V-buffer: Visible volume rendering. *Comput. Graphics* 22(4):59-64; 1988.
7. Levoy, M. Volume rendering: Display of surfaces from volume data. *IEEE Comput. Graphics Applications* 8(3):29-37; 1988.
8. Sabella, P. A rendering algorithm for visualizing 3D scalar fields. *Comput. Graphics* 22(4):51-55; 1988.
9. Levoy, M. A hybrid ray tracer for rendering polygon and volume data. *IEEE Comput. Graphics Applications* 10:33-40; March 1990.
10. Meinzer, H.-P.; Meetz, K.; Scheppelmann, D.; Engelmann, U.; Baur, H.-J. The Heidelberg ray tracing model. *IEEE Comput. Graphics Applications* 11(6):34-43; 1991.
11. Olstad, B. Maximizing image variance in rendering of volumetric data sets. In: *Proceedings of SPIE/SPSE electronic imaging, science and technology San Jose, USA: 1992.*
12. Herman, G. T. A survey of 3D medical imaging technologies. *IEEE Eng. Med. Biol.* 15-17; December 1990.
13. Höhne, K. H.; Bomans, M.; Pommert, A.; Riemer, M.; Tiede, U.; Wiebecke, G. Rendering tomographic volume data: Adequacy of methods for different modalities and organs. In: *Höhne, K. H.; Fuchs, H.; Pizer, S. M. 3D imaging in medicine. Berlin: Springer-Verlag; 1990:197-215.*
14. Tiede, U.; Höhne, K. H.; Bomans, M.; Pommert, A.; Riemer, M.; Wiebecke, G. Surface rendering. Investigation of medical 3D-rendering algorithms. *IEEE Comput. Graphics Applications* 10:41-53; 1990.
15. Bosnes, V.; Blomlie, V.; Hald, J. K.; Holden, M.; Høgaasen, G.; Lundervold, L.; Milvang, O.; Storvik, G.; Øksendal, A.; Aas, K. An evaluation of statistical modeling in magnetic resonance imaging using images from the central nervous system acquired with and without a Gadolinium-based contrast medium. Technical Report No. 855 from the Norwegian Computing Center, Oslo, 1992 ISBN 82-539-0348-0.
16. Steen, E. Volume rendering in medical imaging. Technical report, Department of Computer Systems and Telematics, The Norwegian Institute of Technology, 1992. Diploma thesis.
17. Haralick, R. M.; Shapiro, L. G. Glossary of computer vision terms. *Pattern Recognition* 24(1):69-93; 1991.
18. Rao, R. A. A taxonomy for texture description and identification. New York: Springer-Verlag; 1990.
19. Duda, R. O.; Hart, P. E. *Pattern classification and scene analysis.* New York: John Wiley & Sons; 1973.
20. Hartigan, J. A. *Clustering algorithms.* New York: John Wiley & Sons; 1975.

21. Therrien, C. W. Decision, estimation, and classification: An introduction to pattern recognition and related topics. New York: John Wiley & Sons; 1989.
22. Taxt, T.; Lundervold, A.; Fuglaas, B.; Lien, H.; Abeler, V. Multispectral analysis of uterine corpus tumors in magnetic resonance imaging. *Magn. Reson. Med.* 23(1):55-76; 1992.
23. Lerski, R. A.; McRobbie, D. W.; de Certaines, J. D. Protocols and test objects for the assessment of MRI equipment. *Magn. Reson. Imaging* 6:195-199; 1988.
24. Lerski, R. A.; McRobbie, D. W.; Straughan, K.; Walker, P. M.; de Certaines, J. D.; Bernard, A. M. Multi-center trial with protocols and prototype test objects for the assessment of MRI equipment. *Magn. Reson. Imaging* 6:201-214; 1988.
25. Storvik, G.; Holden, M.; Bosnes, V. Improving statistical image classification by updating model parameters using unclassified pixels. Technical Report No. 857, Norwegian Computing Center, Oslo, ISBN 82-539-0350-2; 1992.
26. Besag, J. On the statistical analysis of dirty pictures. *J. Royal Stat. Soc. Series B* 48(3):259-302; 1986.
27. Geman, S.; Geman, D. Stochastic relaxation, Gibbs distribution, and Bayes restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.* PAMI-6:721-741; 1984.
28. Haslett, J. Maximum likelihood discriminant analysis on the plane using a markovian model of spatial context. *Pattern Recognition* 18:287-296; 1985.
29. Lundervold, A.; Godtliebsen, F. Tissue classification in MR images using contextual and artificial neural network classifiers. Proceedings from the NOBIM conference. Trondheim. 15-16 June, 1992; 263-275.
30. Storvik, G. A bayesian approach to dynamic contours. Technical Report No. 860, Norwegian Computing Center, Oslo, October 1992. Thesis for Dr. Scient degree at the University of Oslo.
31. Storvik, G. A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing. *IEEE Trans. Pattern Anal. Machine Intell.* 16:976-986; 1994.
32. Lundervold, A.; Storvik, G. Segmentation of brain parenchyma and cerebrospinal fluid in multispectral magnetic resonance images. Submitted to *IEEE Trans. Med. Imaging*.
33. Storvik, G.; Lundervold, A. Segmentation of brain parenchyma and CSF in multispectral MR images of the head. The 8th SCIA Conference May 1993; 3-4.
34. Besag, J. Digital image processing: Towards Bayesian image analysis. *J. Appl. Statistics* 16(3):395-407; 1989.
35. Friedland, N.; Adam, D. Automatic ventricular cavity boundary detection from sequential ultrasound images using simulated annealing. *IEEE Trans. Med. Imaging* 8(4):344-353; 1989.
36. Kass, M.; Witkin, A.; Terzopoulos, D. Snakes: Active contour models. *Int. J. Comput. Vision* 1(4):321-331; 1988.
37. Amini, A. A.; Tehrani, S.; Weymouth, T. E. Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints. In: Proceedings Second International Conference Computer Vision, Tarpon Springs, FL: 1988: 95-99.
38. Amini, A. A.; Weymouth, T. E.; Jain, R. C. Using dynamic programming for solving variational problems in vision. *IEEE Trans. Pattern Anal. Machine Intell.* 12(9):855-867; 1990.
39. Olstad, B. Automatic wall motion detection in the left ventricle using ultrasonic images. In: SPIE/SPSE symposium on electronic imaging. Science & Technology San Jose: 1991.
40. Kajiya, J. T.; Von Herzen, B. P. Ray tracing volume densities. *Comput. Graphics (Proc. Siggraph)*, 18(3):165-173; 1984.
41. Suzuki, H.; Toriwaki, J. Knowledge-guided automatic thresholding for 3D display of head MRI images. Proceedings from 9th International Conference on Pattern Recognition 2:1210-1212; 1988.
42. Zhang, G.; Liu, X.; Fox, M. D. Visualization of ventricles of the brain by volume rendering. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society 2; 1989.
43. Fox, M. D. Visualization of multidimensional and multispectral biomedical images. *Elect. Imaging '88* 2:1050-1053; 1988.
44. Foley, J. D.; van Dam, A.; Feiner, S. K.; Hughes, J. F. *Computer Graphics: Principles and practice*. Addison Wesley; 1990.
45. Hausser, K. H.; Kalbitzer, H. R. NMR in medicine and biology—Structure determination, tomography, in vivo spectroscopy. Berlin: Springer-Verlag; 1991.
46. Lundervold, A.; Godtliebsen, F. Noise reduction and brain tissue classification in MR images. IEEE nuclear science symposium and medical imaging conference. Orlando, Florida; October 25-31 1992; 1265-1267.
47. Choi, H. S.; Haynor, D. R.; Kim, Y. Partial volume tissue classification of multichannel magnetic resonance images—A mixed model. *IEEE Trans. Med. Imaging* 10(3):395-407; 1991.
48. Johnson, N. L.; Kotz, S. Distributions in statistics: Continuous multivariate distributions. John Wiley and Sons, Inc.; 1972.

About the Author—MARIT HOLDEN received her Dr. Scient. degree (Ph.D. equivalent) in computer science in 1988, and her Cand.Scient. degree (M.Sc. equivalent) in 1985. Both degrees were received at the Department of Informatics at the University of Oslo in Norway. Presently she is working as a research scientist in the group for image analysis and pattern recognition at the Norwegian Computing Center. Her principal research interests are concerned with statistical classification, image segmentation, and tissue characterization.

About the Author—ERIK N. STEEN received the Norwegian degree of "Sivilingeniør" (M.S.E.) from the Norwegian Institute of Technology at the Faculty of Electrical Engineering and Computer Science, Computer Science Section Dec. 1992, with the thesis "Volume Rendering in Medical Imaging." He is currently working on a PhD thesis on analysis and visualization of medical images. Previous work includes boundary detection in ultrasonic imaging and rendering of 3D ultrasonic data.

About the Author—ARVID LUNDERVOLD has a B.Sc. in mathematics/philosophy from University of Oslo and graduated in Medicine from the same university in 1982. Previous to his M.D. he was engaged in experimental neurophysiological research on the hippocampal slice preparation. In the period 1984-87 he worked on medical informatics and knowledge based systems at the National Hospital, University of Oslo. From 1987-93 he worked as a research scientist in the "BILD" group at the Norwegian Computing Center in Oslo. Presently, he holds a research position at the University of Bergen, Section for medical image analysis and pattern recognition with a grant from the Research Council of Norway (NFR/MH-IT). His principal research interests are concerned with multispectral statistical classification, tissue characterization and, more recently, functional magnetic resonance imaging.

the 1990s, the number of people in the UK who are employed in the public sector has increased from 10.5 million to 12.5 million, and the number of people in the public sector who are employed in health care has increased from 2.5 million to 3.5 million (Department of Health 2000).

There are a number of reasons for this increase. One of the main reasons is the increasing demand for health care services. The population of the UK is increasing, and the number of people who are aged 65 and over is increasing rapidly. This has led to an increase in the number of people who are in need of health care services, and this has led to an increase in the number of people who are employed in health care.

Another reason for the increase in the number of people employed in health care is the increasing demand for health care services. The population of the UK is increasing, and the number of people who are aged 65 and over is increasing rapidly. This has led to an increase in the number of people who are in need of health care services, and this has led to an increase in the number of people who are employed in health care.

A third reason for the increase in the number of people employed in health care is the increasing demand for health care services. The population of the UK is increasing, and the number of people who are aged 65 and over is increasing rapidly. This has led to an increase in the number of people who are in need of health care services, and this has led to an increase in the number of people who are employed in health care.

A fourth reason for the increase in the number of people employed in health care is the increasing demand for health care services. The population of the UK is increasing, and the number of people who are aged 65 and over is increasing rapidly. This has led to an increase in the number of people who are in need of health care services, and this has led to an increase in the number of people who are employed in health care.

A fifth reason for the increase in the number of people employed in health care is the increasing demand for health care services. The population of the UK is increasing, and the number of people who are aged 65 and over is increasing rapidly. This has led to an increase in the number of people who are in need of health care services, and this has led to an increase in the number of people who are employed in health care.

A sixth reason for the increase in the number of people employed in health care is the increasing demand for health care services. The population of the UK is increasing, and the number of people who are aged 65 and over is increasing rapidly. This has led to an increase in the number of people who are in need of health care services, and this has led to an increase in the number of people who are employed in health care.

A seventh reason for the increase in the number of people employed in health care is the increasing demand for health care services. The population of the UK is increasing, and the number of people who are aged 65 and over is increasing rapidly. This has led to an increase in the number of people who are in need of health care services, and this has led to an increase in the number of people who are employed in health care.

Visualization of multi attribute medical images

Abstract

In this paper we present two new algorithms for visualization of multi attribute medical images. The aim of the algorithms is to provide as much information as possible from the multi attribute image in one gray scale or color image without making any rigid classification into different tissue categories. Gray scale images are of special interest as the human eye is considerably more sensitive to spatial variations in intensity than chromatic variations. A nonlinear mapping is made from the original N -dimensional feature space to a M -dimensional output space where $M < N$ and $M \in \{1..3\}$. Two different nonlinear projection methods are investigated for this purpose. We first present a method based on Sammon's nonlinear projection algorithm. Sammon's algorithm is a gradient descent strategy which aims at preservation of inter pattern distances by minimizing a cost function which measures the so-called *Sammon stress*. To reduce computational complexity, we first find a set of X reference vectors in feature space by using a standard clustering technique such as the c-means algorithm. Each feature vector in N -space is associated with its nearest reference vector which we then map to a lower dimensional M -space by using Sammon's algorithm. Finally, we introduce a new algorithm which can be used to create gray scale images when the number of reference vectors is sufficiently small. The original multi attribute data is then projected onto a curve in feature-space defined by an *ordered set* of reference vectors, and a gray scale is mapped along this curve. The optimal ordering of the reference vectors is found as a minimal cost permutation, where the cost function is a weighted sum of inter pattern distances in N space. Our algorithms are compared to principal component analysis (PCA) and a recently published algorithm based on Kohonen self organizing maps. The usefulness of the new algorithms are demonstrated for visualization of both reproducible synthetic images and real MR images.

1 Introduction

Visualization of multi attribute images is gaining increasing importance in many medical applications. In MRI, multi-parameter images are routinely acquired. Also, fusion of different image modalities such as MR and PET images is becoming more widespread.

Several algorithms have been proposed for visualization of multi attribute images. Special attention has been paid to the problem of visualizing remotely sensed data. However, most of the proposed algorithms deal with automatic segmentation into labeled regions. In many cases, the automatic classification procedure will introduce misclassified pixels and the displayed image may therefore contain misleading information. In this paper we concentrate on methods that provide the multi spectral information in a gray-level or color image without making any rigid classification. The aim is to provide as much information as possible into one image such that the experienced diagnostician can perform the final analysis.

Purely interactive methods have been suggested for visualization of medical data [5]. Such methods tend to be very time consuming when applied to images with more than two components per

pixel. Standard linear projection methods such as *principal component analysis* [10, 6] is commonly used in remote sensing and has also been used for applications in medical imaging [2]. Principal component analysis focuses on image variance which in many cases may not be the best measure of image quality [1, 7]. A more recent approach to visualization of multi attribute images was proposed by Harikumar and Bresler [7]. Their method was based on selecting one or few out of many possible linear projections by using projection pursuit [8]. Four different optimality criteria were suggested including three that takes the spatial relations between neighboring pixels into consideration.

A general problem with linear transform techniques is that it may be difficult, and in many cases impossible, to find a linear projection that differentiates between all the important clusters in feature space. In such cases nonlinear projection methods may be more appropriate. Manduca [13] recently proposed a nonlinear projection method which was based on Kohonens self-organizing map (SOM) algorithm [12]. A nonlinear projection was then made of the multi attribute image data onto a 1D SOM. The 1D SOM was then thought of as an ordered set of nodes spaced non-uniformly along a curve in feature space, and a gray scale was mapped along this curve. The SOM algorithm places a pre-specified number of reference vectors into the feature space in an ordered fashion. The position of each reference vector is found iteratively using a simple gradient descent technique. In each iteration, a sample feature vector is presented to the map and the closest matching node (called best matching unit - *bmu*) is found among all the nodes in the map. All nodes contained in a *neighborhood set* centered around the *bmu* are then updated according to the presented feature vector. The result of the process is however very much dependent on the initialization of the map and also on several different parameters which is used in the updating of the reference vectors. The mathematical properties of the SOM is only known for very simplified cases [12].

In this paper we propose two new approaches to multi-spectral image visualization. We first present a method based on Sammon's nonlinear projection algorithm [11]: Sammon's algorithm is a gradient descent technique that aims at preservation of inter pattern distances by minimizing a cost function which measures the so-called *Sammon stress*. The number of computations in each iteration is however proportional to the square of the number of sample vectors which can be as many as 512^2 in a typical medical image. To reduce computational complexity, we first find a set of X reference vectors in feature space by using a standard clustering technique such as the c-means algorithm [15]. Each feature vector in the N -space is associated with its nearest reference vector which we then map to a lower dimensional M -space by using Sammon's algorithm. A neural network version of Sammon's algorithm [9] was evaluated for image visualization in [14] and found to be a useful alternative to linear projection methods such as PCA. Both versions of Sammon's algorithm are however gradient descent strategies and does not necessarily yield the best solution to the optimization problem.

We also introduce a new algorithm which can be used to create gray scale images when the sufficient number of reference vectors is small. The original multi attribute data is then projected onto a curve in feature-space defined by an *ordered set* of reference vectors, and a gray scale is mapped along this curve. The optimal ordering of the reference vectors is then found as a minimal cost permutation. Several different cost functions has been considered. Sammon's cost function measures

the preservation of inter pattern distances. In many cases, it is impossible to find a mapping that approximately preserves *all* the inter pattern distances. To illustrate this, consider a situation where a number of clusters is evenly distributed along a circle in a 2 dimensional feature space. In this case, it is obviously impossible to find a mapping to a 1 dimensional output space in which all inter pattern distances are approximately preserved. In image visualization, a reasonable criterion may be that reference vectors that are close to each other in feature space should preferably be mapped to gray scale values that are similar. The problem of finding an optimal ordering of reference vectors has also been studied in communication theory [16] where the aim was to reduce quantization errors in transmission of the reference vectors over a noisy channel.

2 Projection methods for multi attribute image visualization

2.1 Principal Components Analysis

Principal Component Analysis (PCA) is a commonly used technique for multi attribute image visualization [6]. Let $\xi = (\xi_1, \dots, \xi_N)$ denote a multi attribute image ξ with N components. Furthermore, let C_ξ denote the covariance matrix of the image vectors ξ . Correlation between the components ξ_i can be utilized to compute a new image $\eta = (\eta_1, \dots, \eta_M)$ with a smaller number of components M . η can be computed as linear combinations of the components.

$$\eta = \mathbf{A}\xi \quad (1)$$

\mathbf{A} is a matrix where the rows are the eigenvectors of C_ξ sorted by decreasing size of their associated eigenvalues γ_i . \mathbf{A} is then an orthonormal matrix that represents a basis transformation of the n -dimensional space containing ξ . It can be shown that the variance of the components η_i is equal to the eigenvalues γ_i . η_1 is therefore the gray scale image with maximum variance. η_1, \dots, η_3 can be mapped into a color image using the *HSV* or the *RGB* color models.

2.2 Sammon's nonlinear mapping

Sammon's nonlinear mapping[11] is a nonlinear projection technique that attempts to preserve all the inter pattern distances as well as possible.

Let $\xi(\tau), \tau \in \{1..X\}$ denote vectors in the N -dimensional feature space containing the multi attribute image ξ . Furthermore, let $\eta(\tau), \tau \in \{1..X\}$ denote vectors in the M -dimensional output space where $M < N$. Sammon defined the mapping error, called *Sammon's stress* as follows:

$$E = \frac{1}{\sum_{\tau=1}^{X-1} \sum_{v=\tau+1}^X d^*(\tau, v)} \sum_{\tau=1}^{X-1} \sum_{v=\tau+1}^X \frac{[d^*(\tau, v) - d(\tau, v)]^2}{d^*(\tau, v)} \quad (2)$$

where $d^*(\tau, v)$ is the distance between the vectors $\xi(\tau)$ and $\xi(v)$ in feature space and $d(\tau, v)$ is the distance between the vectors $\eta(\tau)$ and $\eta(v)$ in output space.

Usually the Euclidean distance is applied, but any other distance measure can be used instead. Sammon's stress measures how well inter pattern distances are preserved in the nonlinear projection. Sammon proposed a gradient descent technique to find a configuration of the X vectors in the M -dimensional space, such that E is minimized. The algorithm involves computation of $X(X-1)/2$ distances in each iteration. The algorithm is therefore computationally expensive if X is large. In image visualization, the number of feature vectors in an image I will typically be in the range $I \in \{256^2 \dots 512^2\}$, and it will therefore be impractical to use the algorithm directly. A simple solution to this problem is to find a smaller number of X reference vectors in the feature space containing ξ using a standard clustering algorithm such as the c-means algorithm. The c-means algorithm is also an iterative gradient descent technique, but the number of computations in each iteration is proportional to $I \cdot X$, and the algorithm is thus computationally much lighter than Sammon's algorithm. Sammon's algorithm can then be used to find a mapping of these reference vectors to the M dimensional output space containing η . For $M = 1$, X can be set to 256 or less since this is usually the largest number of gray levels that can be simultaneously displayed on a computer screen. To display the image, one may first label each sample vector with the index of the nearest reference vector using the nearest neighbor rule. The complete algorithm can be stated as follows:

1. Find a set of X reference vectors in the N dimensional feature space, using a standard clustering technique such as the c-means algorithm. Then label all feature vectors with the index of their nearest reference vector.
2. Use Sammon's algorithm to find a mapping f of the X reference vectors from N space to M space: $f : \mathfrak{R}^N \rightarrow \mathfrak{R}^M$.
3. Apply f to each labeled feature vector.
4. Normalize the component values of the output vectors to lay in the range 0..255

If color images are desired ($M = 3$), the number of possible colors that can be displayed on a computer screen will typically be 256^3 . Obviously, it will be impractical to use such a large number of reference vectors. A useful alternative is then to use a much smaller number of reference vectors and label each sample vector with the index of its three nearest neighbors. The color components of the projected sample vector can then be computed as linear combinations of the colors assigned to its neighbors.

A general weakness of Sammon's algorithm is that it is a simple gradient descent technique which unavoidably will get stuck in a local minimum on the error surface. The algorithm can be run several times with different random initializations, but there is no guarantee that the globally optimal configuration will be found. An artificial neural net version of Sammon's projection algorithm was proposed by Jain and Mao[9], and evaluated for image visualization in [14]. The neural algorithm is however also a gradient descent strategy.

2.3 Nonlinear projection of ordered reference vectors

In this section we present a new algorithm for computing gray scale images (eg. $M = 1$). Gray scale images are of particular interest as the human eye is considerably more sensitive to spatial variations in intensity than it is to variations in chromaticity [3].

Assume that we have an *ordered* set of X reference vectors (or points) such that they define a piecewise linear curve in feature space. A gray scale can be mapped non uniformly along this curve with the value zero assigned to the lowest ranked reference vector. We then assign a gray scale value to each reference vector which is a function of arc length along the curve in feature space, as in the SOM algorithm presented by Manduca[13]. We have then defined a mapping $f : \mathfrak{R}^N \rightarrow \mathfrak{R}$ from an N dimensional feature space to a 1 dimensional output space. The remaining problem is to find an ordering of the reference vectors which is optimal in some sense. This is a *combinatorial* problem which can be solved by computing a minimal cost permutation of the reference vectors. We have then used the following cost function:

$$E_\pi = \sum_{\tau=1}^{X-1} \sum_{v=\tau+1}^X \frac{d^*(\tau, v)}{|\pi_\tau - \pi_v|^\gamma} \quad (3)$$

Here, $|\pi_\tau - \pi_v|$ is the difference between the permutation indexes associated with the respective vectors. γ is a constant factor. The cost function is thus a *weighted* sum of inter pattern distances in N -space, where each inter pattern distance $d^*(\tau, v)$ is multiplied with a weight $w = \frac{1}{|\pi_\tau - \pi_v|^\gamma}$. Distances between vectors that are close in the permutation are given heavier weights than distances between vectors that are far apart. As $\gamma \rightarrow \infty$, the cost function reduces to a measure of the total distance along the curve. The corresponding optimization problem is then analogous to the problem of finding the shortest Hamiltonian cycle in a graph. This problem is known to be NP-complete [4]. The minimal cost permutation can be found using exhaustive search, but the running time then grows exponentially with the number of reference vectors X , so this technique is only applicable if X is small. Branch and bound techniques [4] can be used to speed up the algorithm significantly. A lower bound L for the cost function defined in Eq. 3 can be found as follows:

- Sort all $X(X - 1)/2$ N -space inter pattern distances in ascending order: $[d_1^* \dots d_{X(X-1)/2}^*]$
- Sort all $X(X - 1)/2$ weights in ascending order: $[w_1 \dots w_{X(X-1)/2}]$
- Compute $L = \sum_{i=1}^{X(X-1)/2} w_i d_i^*$

Alternatively, heuristic algorithms such as k -opting [4] can be used to find a pseudo-optimal ordering of a larger number of reference vectors.

256 gray levels can still be used for display by projecting each feature vector onto the closest line segment defined by two consecutive reference vectors on the curve in feature space. The feature vector can then be mapped to a gray-level value which is a linear combination of the gray level values associated with the two reference vectors. Preferably, the path should connect all reference

vectors that are likely to belong to the same cluster before jumping to a new cluster. Since the distribution of feature vectors within different clusters may vary, weighted Euclidean distance measures such as the *Mahalanobis distance* should be considered [15].

The algorithm presented in this section can be stated more formally as follows:

1. Find a set of reference vectors $\xi(\tau), \tau \in 1..X$ in the N dimensional feature space, using a standard clustering technique such as the c-means algorithm. We want to find a corresponding set of output values $\eta(\tau), \tau \in 1..X$ in the 1-dimensional output space.
2. Let π_τ denote the rank of the vector $\xi(\tau)$ in an ordering (permutation) of the reference vectors. The ordered reference vectors define a piecewise linear curve in feature space. We define a mapping $f : \mathfrak{R}^N \rightarrow \mathfrak{R}$ as follows: $\eta(\pi_\tau) = f(\xi(\tau)) = d^+(\pi_\tau)$, where d^+ is the arc distance along the curve in feature space defined by the ordered set of reference vectors. $d^+(1) = 0$, which means that the lowest ranked reference vector will map to the lowest gray level value. The algorithm seeks the permutation of reference vectors that gives the lowest cost.

3 Numerical Experiments

All experiments presented in this section were done with AVS (Application Visualization System) running on a SPARC IPX workstation.

3.1 Experiment 1

In our first experiment we created a multi attribute image with two components. The values in each component ranged between 0 and 255. The image consisted of seven different regions. All feature vectors within a region were drawn from the same Gaussian distribution. The seven distributions had different mean vectors but the same covariance matrix $\Sigma = \sigma \mathbf{I}$, $\sigma = 20.0$. The cluster configuration was chosen such that it was impossible to differentiate between all the clusters in any linear projection. A scatterplot which shows the configuration of the 7 clusters is shown in Fig. 1. The x-axis in the scatterplot is aligned with the first principal axis, and the y-axis is aligned with the second principal axis. The aim of the experiment was to verify how well each projection algorithm was able to separate the 7 different clusters when the dimensionality was reduced to 1. The principal component images are shown in Fig. 2 and 3. It is not possible to separate all clusters in any of the components.

The two nonlinear projection methods presented in this paper were applied to the synthetic image. In both algorithms, Euclidean distance was used as a distance measure, and the number of reference vectors X was set to be 7 or more. The reference vectors were found with the C -means algorithm. For $X = 7$, Sammon's algorithm resulted in an image where it was possible to separate all the different clusters very well. The result is shown in Fig. 4. As Sammon's algorithm is a simple

gradient descent strategy, it was necessary to run the algorithm several times with different random initializations and then choose the configuration with the lowest stress in order to get a satisfactory solution. As X was increased, Sammon's algorithm did only give a satisfactory result if a very large number of trials were made. For $X = 15$ approximately 1 out of 100 runs resulted in an image where it was possible to separate all the different clusters.

The new algorithm described in section 2.3 resulted in an image where it was possible to separate all the different clusters. A branch and bound technique was used to speed up the algorithm.

For $X \leq 10$ the ordering was done in less than 1 second. The result of the algorithm ($X = 7$) is shown in Fig. 5 together with plots which shows the curves in feature space for $X = 7$ and $X = 15$.

The 1D SOM algorithm proposed by Manduca[13] also resulted in an image where it was possible to separate all the different clusters. The 1D SOM map was then initially aligned with the first principal axis, and network parameters were set according to the recommendations given by Kohonen[12]. As X was increased, the algorithm resulted in different mappings from one run to another.

3.2 Experiment 2

In our second experiment we created a synthetic image of size 150^2 with three different textured regions. In the first (background) region, each pixel value was drawn from a normal distribution $N(\mu_1, \sigma_1^2)$, with $\mu_1 = 100$ and $\sigma_1 = 18$. In the second region, each pixel value was drawn from a normal distribution with a different mean value $\mu_2 = 128$ but with the same standard deviation σ_1 . Finally, in the third region, each pixel was drawn from a normal distribution with the same mean value as in the second region μ_2 , but with a higher standard deviation $\sigma_2 = 36$. The synthetic image is shown in Fig. 6. Region 3, which is circular and lies inside region 2, is barely visible in the original image.

A multi attribute representation of the textured image was then made by computing local feature vectors for each pixel in the image. The intensity value of each pixel and the 8 closest neighbors were sorted and used as a 9 dimensional feature vector. The feature vectors then represented the local distribution around each pixel. The three different image regions then appeared as three different clusters in feature space, but with considerably overlap between each pair of clusters. The 9 dimensional multi attribute image was transformed with principal component analysis. The first principal component is shown in Fig. 7. The contrast between the first and the second region has been enhanced from the original image, but it is still difficult to separate region 2 from region 3. The second principal component is shown in Fig. 8. In this image, the contrast between region 2 and region 3 has been enhanced, but the contrast between region 1 and region 2 is low. The two nonlinear projection methods presented in this paper were applied to the image. In both algorithms, a weighted Euclidean distance measure was used. Each component was weighted with the inverse standard deviation. The nonlinear projection based on Sammon's algorithm gave good contrast between the three different clusters for $X = 3$ but as in the first experiment, a substantial number of runs had to be made as X increased. The nonlinear projection algorithm described in section 2.3

resulted in an image with good contrast between the three different regions. The ordering of the reference vectors did only change locally as X was increased. The result of the algorithm ($X = 12$) is shown in Fig. 9 together with a plot which shows the corresponding curve in feature space projected onto the plane spanned by the first and second principal axes (See Fig. 10). The results of the 1D SOM algorithm was dependent both on initialization and on different learning parameters. The parameters were adjusted according to the recommendations given by Kohonen[12]. A typical result achieved with the SOM algorithm is shown in Fig. 11 together with a plot of the corresponding curve in feature space (See Fig. 12). In this case, the 1D SOM was originally aligned with the first principal axis.

3.3 Experiment 3

In our last experiment, we wanted to evaluate our new algorithms on real MR images of a human brain. The MR images were acquired from a 53 year old male patient with diagnosis Ca. bronchiale with metastasis. The slice thickness was 5mm and the pixel size about 1mm^2 . Four co registered channels were acquired by varying the parameter settings of the MR equipment. The channels were T2, T1, proton density and T1 weighted in the presence of a contrast agent. The aim of this experiment was to investigate how the different algorithms were able to produce single component images with good contrast between the different tissue types present in the brain. A visualization pipeline was made in the AVS system, where the user could mark one or more regions of interest inside the brain. All feature vectors inside the selected region(s) were then used as input to the algorithms. In this way, it was possible to tune the algorithms to perform particularly well within clinically interesting regions. The nonlinear projection algorithms were compared with Principal Component Analysis. In most cases, the algorithms gave similar results. In other cases the nonlinear projection algorithm presented in section 2.3 gave a better differentiation between the interesting tissues. Such an example is shown to the left in Fig. 13. In the middle left of the brain there is a (dark) Metastasis surrounded with edema. White and gray matter appears with different gray shades. The central spinal fluid appears black. For comparison, the first principal component is shown in the middle of Fig. 13. The two images are similar, except that the contrast between edema and the other tissues are better in the non linear projection. The second principal component is shown to the right in Fig. 13. In this image, the contrast between edema and the other tissues is better than in the first principal component. The non linear projection contains important information from both components.

4 Discussion

In this paper we have presented new algorithms for visualization of multi attribute images. A nonlinear mapping is made from the original N -dimensional feature space to a M -dimensional output space where $M \in \{1..3\}$. Two different nonlinear projection methods have been investigated for this purpose. We have developed one method based on Sammon's nonlinear projection algorithm.

Sammon's algorithm is a gradient descent strategy which aims at preservation of inter pattern distances by minimizing a cost function which measures the so-called *Sammon stress*. One of the main problems with Sammon's algorithm was that it had to be run several times with different initializations in order to get a satisfactory result. This is not surprising, since the cost function may have many different local minima. We have also developed a new algorithm which projects the original multi attribute data onto a curve in feature-space defined by an ordered set of reference vectors. The ordering is found as a minimum cost permutation, where the cost function is a weighted sum of inter pattern distances. Other cost functions could be considered. The algorithm can be compared to Kohonens self-organizing map (SOM) algorithm. In the SOM approach the ordering process is integrated with the clustering. The process is heavily dependent on both initialization and learning parameters, and the mathematical properties of the algorithm is only known for very simplified cases. In our algorithm the clustering and the ordering problems are treated separately. The ordering process is transformed into a combinatorial problem which can be solved optimally for a small or moderate number of reference vectors. However, if the number of reference vectors is large, heuristic algorithms has to be used.

Several experimental setups were made in order to evaluate the suggested algorithms. In all experiments, the aim was to produce gray scale images with as much information as possible. The experiments showed that the nonlinear projection algorithms can provide information which is difficult or impossible to obtain from linear algorithms such as Principal Component Analysis. The drawback with the nonlinear algorithms is the increased computational complexity. In conclusion, we have found that the suggested methods are useful alternatives to linear projection methods such as Principal Component Analysis.

5 Acknowledgments

The authors would like to thank the Norwegian Technical Research Council for supporting this study. We would also like to thank Marit Holden at the Norwegian Computing Center for supplying the MR data which was originally acquired at Radiumhospitalet in Oslo.

References

- [1] R. W. Boesel and Y. Bresler. A new paradigm for optimal multi-parameter image visualization. *SPIE Visualization in Biomedical Imaging*, 1808, 1992.
- [2] U. Schmiedl et. al. The utility of principal component analysis for the image display of brain lesions. a preliminary, comparative study. *Magn. Resn. Med.*, 4, 1987.
- [3] J.D Foley, A. van Dam, S.K Feiner, and J.F Hughes. *Computer Graphics: Principles and Practice*. Addison Wesley, 1990.

- [4] M. R. Garey and D. S. Johnson. *Computers and Intractability, a Guide to the Theory of NP-completeness*. Freeman, 1979.
- [5] K. J. Goldman and T. R. Miller. Interactive combining of functional images. *Eur. Journal of Nuclear Medicine*, (11), 1985.
- [6] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison Wesley, 1992.
- [7] G. Harikumar and Y. Bresler. Multi-parameter image visualization by projection pursuit. *SPIE Visualization in Biomedical Imaging*, 1808, 1992.
- [8] P. J. Huber. Projection pursuit. *Annals of Statistics*, 13(2), 1985.
- [9] A. K. Jain and J. Mao. Artificial neural network for nonlinear projection of multivariate data. In *IJCNN Intern. Joint conf. on Neural Networks*, volume III, 1992.
- [10] I. T Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.
- [11] J. W. Sammon Jr. A nonlinear mapping for data structure analysis. *IEEE Trans. On Computers*, 18(5), 1969.
- [12] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9), Sept 1990.
- [13] A. Manduca. Multi-parameter image visualization with self-organizing maps. In *SPIE Medical Imaging 94*, 1994.
- [14] Gaute Myklebust and Erik Steen. Neural networks in visualization of multispectral medical images. In *Joint Conference on Information Sciences (JCIS'94)*. Duke University, November 1994.
- [15] R. Schalkoff. *Pattern Recognition, statistical, structural and neural approaches*. John Wiley & Sons, Inc, 1992.
- [16] K. Zeger and A. Gersho. Pseudo-gray coding. *IEEE Trans. On Communications*, 38(12), December 1990.

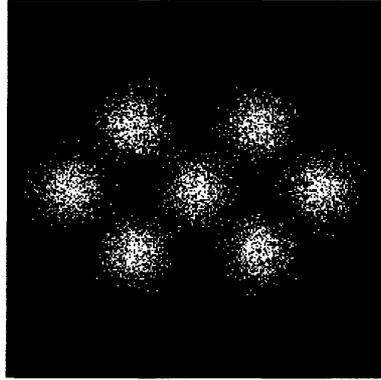


Figure 1: Scatterplot showing the configuration of 7 clusters in a 2-dimensional feature space

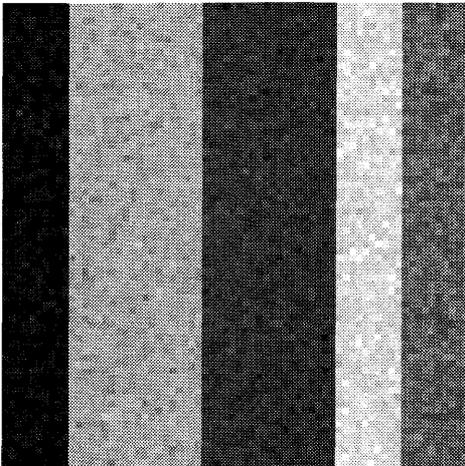


Figure 2: First principal component of image with 2 attributes containing 7 Gaussian clusters

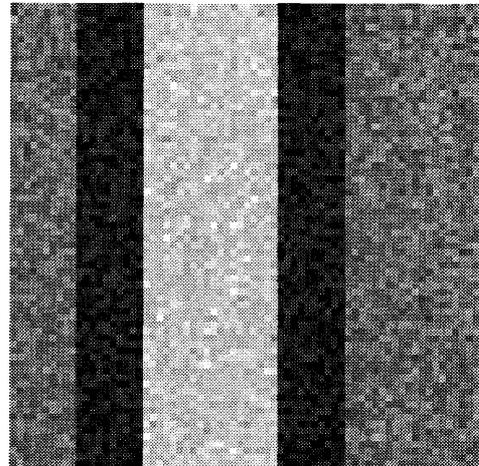


Figure 3: Second principal component of image with 2 attributes containing 7 Gaussian clusters

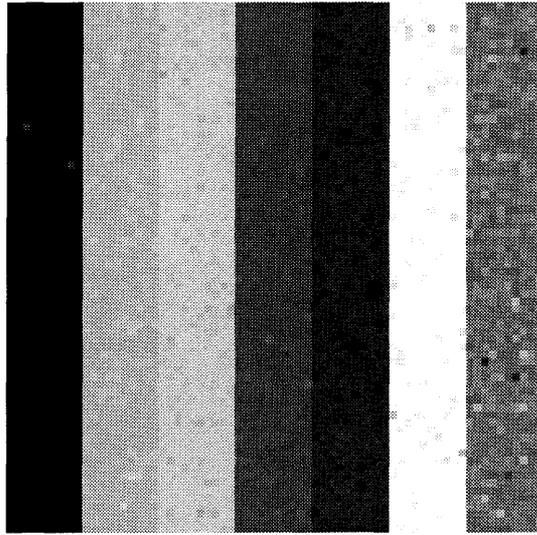


Figure 4: Sammon projection of image with 2 attributes containing 7 Gaussian clusters

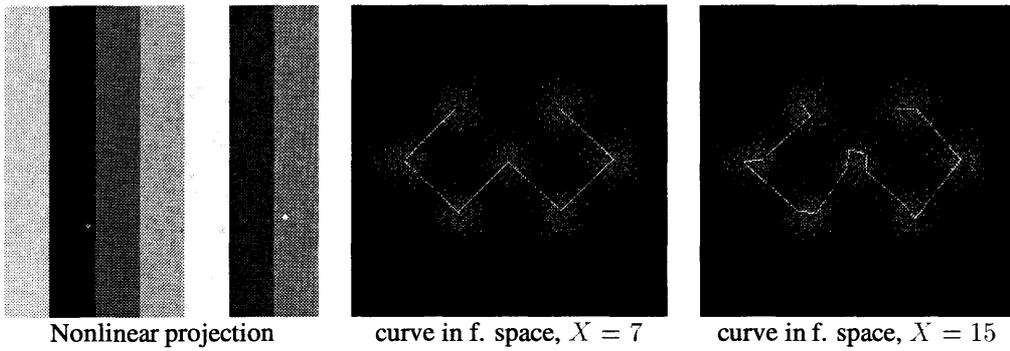


Figure 5: Nonlinear projections of 2-attribute image

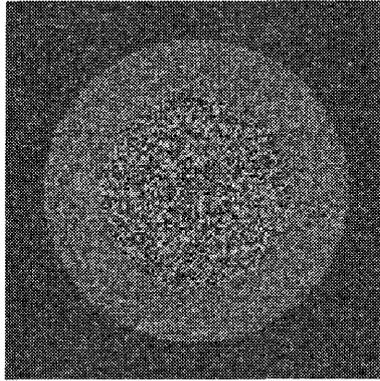


Figure 6: Synthetic texture image

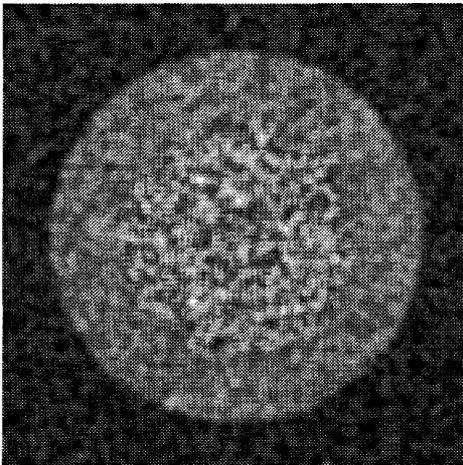


Figure 7: 1st principal component of texture image

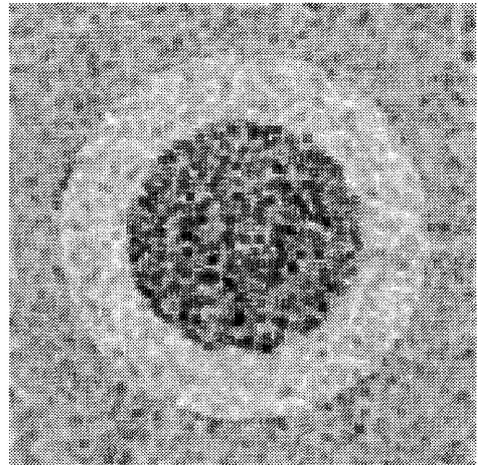
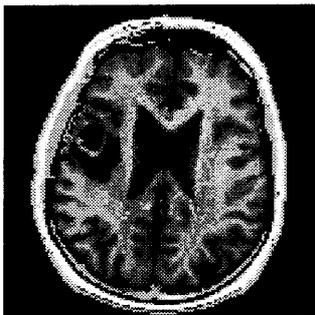
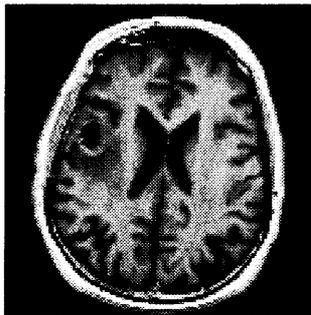


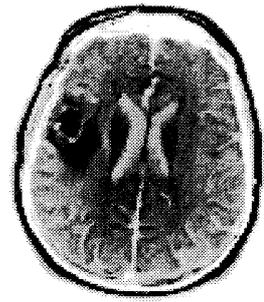
Figure 8: 2nd principal component of texture image



Nonlinear projection



First principal component



Second principal component

Figure 13: Projections of four channel MR image of a human brain

the 1990s, the number of people in the UK who are aged 65 and over has increased from 10.5 million to 13.5 million (1990-2000).

There are a number of reasons why the number of people aged 65 and over has increased. One reason is that people are living longer. The life expectancy at birth in the UK is 77 years for men and 81 years for women. This is an increase from 72 years for men and 76 years for women in 1950. The increase in life expectancy is due to a number of factors, including improvements in diet, housing, and healthcare.

Another reason why the number of people aged 65 and over has increased is that people are having children later in life. This is due to a number of factors, including the fact that people are staying in education longer and the fact that people are having children later in life.

There are a number of challenges that the UK faces as a result of the increasing number of people aged 65 and over. One challenge is the need for more social care services. As people age, they are more likely to need help with daily activities, such as shopping, cooking, and cleaning.

Another challenge is the need for more housing for older people. Many older people live in overcrowded and poorly maintained housing. This is a problem because it can lead to health problems and a poor quality of life.

There are a number of ways in which the UK can address these challenges. One way is to invest in social care services. This could include providing more home care services and more residential care services.

Another way is to invest in housing for older people. This could include providing more affordable housing and more housing that is designed for older people.

There are a number of other ways in which the UK can address these challenges. For example, it could invest in education and training for older people. This could help them to stay active and engaged in society.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

The UK faces a number of challenges as a result of the increasing number of people aged 65 and over. However, there are a number of ways in which the UK can address these challenges. It is important that the UK takes action now to ensure that all older people have a good quality of life.

Shell rendering with hardware supported data extraction

Abstract

A general framework for fast visualization of multispectral volume data is presented. Dedicated hardware with a non-numeric coprocessor is utilized in the first step of the rendering pipeline to process the volume data and extract voxels according to feature characteristics. This capability is used to select voxels according to automatic classification results or real-time descriptions of regions of interest supplied by the user in an interactive environment. By this step we can in real-time reduce the number of voxels that have to be considered in the rendering and increase the speed of the volume rendering accordingly. The selected voxels are generated in a front-to-back (or back-to-front) order and projected to the view plane where a 3D rendering is accumulated with an adaptation of the shell rendering technique proposed by Udupa and Odhner. The paper includes an overview of the underlying hardware architecture and presents numerical experiments with a software simulator.

1 Introduction

Three-dimensional (3D) display techniques have been studied extensively for medical applications for the past several years and are well established as a clinical tool in X-CT imaging [6, 18, 21]. Visualization techniques for MRI and PET have also been explored [1, 2, 10, 20, 23]. Ultrasonic 3D and 4D acquisitions have also attracted much interest during the last years [14, 17, 22]. Volume rendering is used to give a 2D representation of the interior of a volumetric dataset. The usefulness of such methods relies on their ability to extract clinically useful information. Interesting structures are made opaque and obscuring structures are made transparent. The opacity assignment implies an underlying classification that is useful for the visualization quality. In addition, we will exploit this classification to reduce the processing requirements for volume rendering. Volume rendering with uniform opacity assignment reduces to average value projection which has been found to be an inferior technique in many applications [14]. Volume rendering techniques have been presented which are based on an initial classification of the voxels (volume elements) into different tissue categories. Simple thresholding can be used to render isovalue surfaces [5, 7, 9]. A more sophisticated approach to volume rendering was presented by Drebin et. al. [4] An initial probabilistic classification is here used to classify voxels into different tissue categories. Drebin stressed the fact that binary classification schemes will usually result in artifacts, and he therefore allowed voxels to contain a mixture of materials. The amount of a material in a given voxel was then decided from a probabilistic classification. Some image modalities provide a sufficient signal-to-noise ratio to allow for a reliable unsupervised classification into the various materials or tissue categories of interest. This is not the case in other important modalities such as ultrasonic imaging and seismic imaging. In MRI-imaging, the multi-parameter dependence of the MRI-data can be utilized to obtain initial classifications into the various tissue types. Multi-dimensional statistical analysis has been used to estimate the probability density distributions for the different tissue categories from standard multi variate distributions such as the multi normal distribution [16]. The parameters for

these distributions are estimated from training sets. A general problem with this approach is that the classification parameters obtained from one dataset may not be directly transferable to datasets with different or additional tissue. Parameters for the additional tissue categories must then be specified and old parameters must be adjusted by some calibration procedure.

There are two basic approaches to visualization of 3D objects: *surface rendering* and *volume rendering*. In surface rendering an explicit model of the imaged structure is extracted. Typically, a polygonal representation of the boundaries between the various materials or tissue categories is defined and used in the subsequent rendering of the 3D scene. This approach rely heavily on the robustness of the surface detection schemes. In contrast to surface rendering, volume rendering does not compute an explicit model of the imaged structure. The imaged scene is viewed as a semitransparent volume. Different characteristics can be extracted in the rendering by modifying the strategy for assigning opacity to each individual voxel. Volume rendering is therefore almost by definition a more fuzzy oriented technique than the surface rendering approach. This can be utilized to allow for more flexibility in the classification stage and in the specification of rendering characteristics through opacity assignment. The full power of this approach is not revealed before renderings are generated in real-time with 1-30 frames/s. The user can in this case do interactive inspection/classification or modifications of the opacity assignment and have immediate visual feedback in the 3D scene.

2 Ray casting vs. voxel projection

Most volume rendering algorithms are based on the ray casting paradigm. This paradigm is illustrated together with the alternative voxel projection technique in Fig. 1. The ray casting paradigm loops through the pixels in the view plane and for each pixel a line through the 3D dataset is generated according to the selected viewing transformation. A 1-dimensional density evolution is then extracted by trilinearly interpolation of voxels along the line that passes through the domain of the 3D dataset. The rendering is then generated by assigning a color to the pixel in the view plane that reflects some properties of the extracted density evolution.

The disadvantage of the ray casting approach is that the extraction of each of the individual density evolutions usually requires values spread all over the address space occupied by the 3D dataset. Modern RISC based workstations rely heavily on efficient utilization of their memory architectures with high speed cache-memories close to the CPU. Realistic 3D datasets are unfortunately usually too large to fit in cache and the almost randomized address generation caused by the ray casting principle makes it difficult to utilize the high speed memory architectures. Elaborate pre-fetching strategies in the operating system might in fact turn the presence of cache memory into a disadvantage for ray casting.

Voxel projection is an alternative data driven approach. The dataset is read in a systematic manner and each voxel is transformed into the associated location in the view plane based on the selected viewing transformation. The shell rendering algorithm [19] is based on voxel projection.

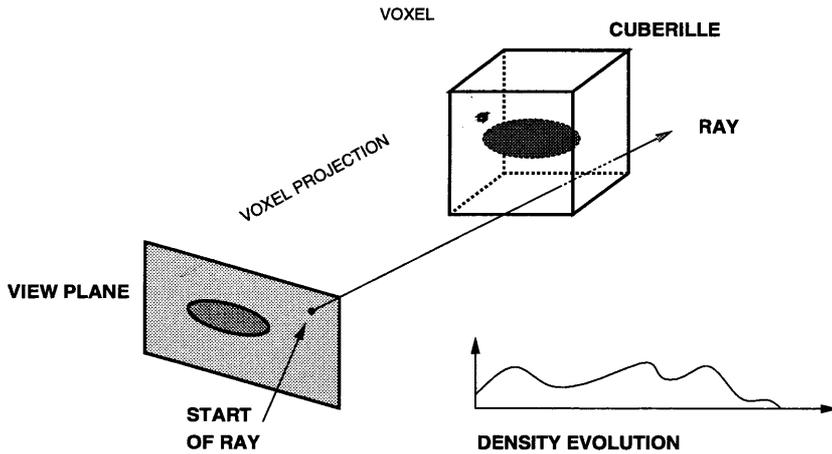


Figure 1: The ray casting and voxel projection paradigms for volume rendering.

Volume rendering software has earlier usually preferred the ray casting principle because:

1. It is easier to obtain high quality interpolations in the view plane if the resolution of the view plane image significantly exceeds the resolution of the 3D dataset. Voxel projection techniques must solve this problem with anti-aliasing and proper adjustment of the resolution in the 3D dataset.
2. The density evolutions extracted by the ray casting paradigm provide a greater flexibility in the definition of the actual rendering principle. The voxel projection paradigm can only guarantee that voxels arrive at a given pixel in the front-to-back (FTB) or back-to-front (BTF) order. The associated rendering principle is therefore constructed as an iterative updating algorithm. On the other hand, most of the important rendering algorithms fall into this category including max rendering, average rendering and Drebins rendering method.

3 Combining voxel projection with real-time data extraction

It is generally recognized that speed becomes necessary - rather than just convenient - to further development of volume rendering. The data driven aspect of voxel projection is an attractive property for both efficient utilization of dedicated hardware and interfacing with real-time data sources. Voxel projection can also handle non-uniform spatial sampling in a very direct and efficient manner. Further speed improvements can be achieved if the stream of voxels is reduced before they enter into major computations. Udupa and Odhner have proposed a technique called shell rendering [19] which is based on voxel projection of a limited dataset extracted from the original com-

plete 3D cuberille. Udupa and Odhner's technique is a software based approach where the volume is preprocessed (a step reported to require approximately 5 min. on a high performance workstation). The preprocessing discards the voxels that *a priori* are known not make a significant contribution to the final volume rendering. The extracted voxels are rendered in software with 2-3 seconds per frame. The method is called shell rendering because the extraction principle can be thought of as a fuzzy generalization of a simple boundary extraction. Shell rendering allows for interactive manipulation of the viewing parameters with almost immediate visual feedback. Hardware accelerators and the possibility to subsample the dataset can bring this methodology further to a truly real-time environment for volume rendering.

The visualization speed of shell rendering is attractive, but the initial data extraction is slow and far from interactive rates even on powerful workstations. Immediate visual feedback in the 3D scene on the geometry of the extracted data will make volume rendering an efficient tool for automatic and interactive exploration of the geometry of multispectral volume data. Immediate visual feedback on classification results will also allow for visual sensitivity analysis of the classifications results which is important for immediate validation of clinical findings. Before presenting a hardware accelerator that offers real-time shell extraction we will explore some basic principles that can be utilized in the classification stage.

3.1 Preprocessing techniques for automatic classification

3.1.1 Multi-spectral classification

Classification of voxels is used to define an appropriate opacity function $\omega : V \rightarrow [0, 1]$. Based on a given opacity function Udupa and Odhner [19] define a shell as

$$B = \{v \in V \mid \omega(v) > \Omega_l \text{ and } \omega(v') \leq \Omega_h\} \quad (1)$$

where $0 < \Omega_l \leq \Omega_h < 1$ and v' is a voxel in a given neighborhood of v . A surface is extracted if $\Omega_l = \Omega_h$. Eq. 1 must be modified in order to allow for efficient data extraction with a general purpose non-numeric coprocessor. The shell membership must usually be determined by an isolated multi-spectral analysis of the attributes recorded or computed at a given voxel. In MRI-imaging, the multi-parameter dependence of the MRI-data can in this way be utilized to obtain initial classifications into the various tissue types. Multi-dimensional statistical analysis has earlier been used to estimate the probability density distributions for the different tissue categories from standard multi variate distributions such as the multi normal distribution [16].

3.1.2 3D edge detection

It is usually sufficient to obtain a good description of the implicit surface geometry inside the 3D cuberille in order to produce meaningful volume visualizations. Voxel elements far from significant transitions can therefore usually be discarded. Non-linear diffusion filters can be used to define

edges at the various levels in scale space. We have earlier reported how these filters can be explicitly used as a data extraction parameter within the shell rendering framework [14, 13].

3.1.3 Using the distance transform in 3D

The neighborhood operator in Eq. 1 must be replaced by a preprocessing where for example the minimal element in the given neighborhood of v is recorded as an additional attribute on voxel v . A more flexible approach is to process the detected boundaries with a three-dimensional distance transform [3]. The shortest distance to features of interest can in this way be recorded as additional attributes and used in the real-time process for data extraction. We have earlier studied one application of this approach combined with both automatic 3D edge detection filters and manual/automatic procedures for object recognition in 3D data sets [11].

3.1.4 Region growing with the Dijkstra algorithm

Region growing can be used to either automatically segment the scene into separated components or the user can interactively extract specific objects. The volume rendering framework can handle uncertainties explicitly through opacity assignments that increase the transparency of uncertain classifications. Fig. 2 and Fig. 3 illustrate two renderings generated with shell rendering of 3D ultrasonics. In both cases a number of seeds have been interactively positioned by the user inside the 3D volume. The selected seeds are used as the source in a Dijkstra algorithm computing for every voxel the shortest path to one of the selected seeds. The volume is interpreted as a graph with a node for each voxel element and an edge between neighboring voxels. The cost is computed as a function of the difference in image characteristics between each individual voxel and the set of voxels that are used as seeds. Finally, we use the shortest distance computed by the Dijkstra algorithm as a basis for opacity assignments. The drawback with this attribute is that the region growing is not accelerated and can only be modified interactively if the size of the volume is small.

3.1.5 Utilizing specific voxel variations along a predetermined structure

The underlying VLSI architecture can also do selection based on signatures found in a sequence of voxels that have been sent through the data extraction module in Fig. 5. The interpretation of this mode depends on the address generation which triggers the data reservoir. One possibility is to augment each voxel with the voxels found in a predetermined spatial 3D neighborhood.

Some 3D applications, including seismic imaging, provide a rather unique direction in 3D space that can be used to formulate data extraction criteria. In seismic imaging the signatures recorded across a sequence of layers can be used as a basis for fuzzy data extraction in volume rendering highlighting similar layer sequences in the 3D scene.



Figure 2: Visualization of liver veins.

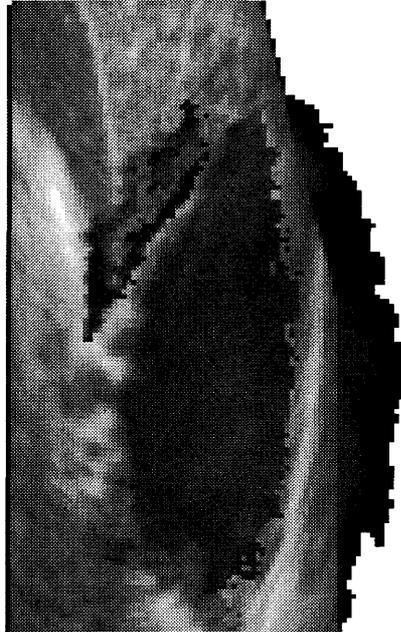


Figure 3: Visualization of hemangioma and portal vein. The visualization was made with a color encoding of the different objects.²

3.2 Interactive classification

The speed of the visualization pipeline makes the presented volume rendering framework attractive for *interactive* classification procedures. Interactive methods can be used to extend, calibrate or replace the automatic classification procedures. Interactive classification will include real-time mouse tracking in arbitrary 2D slices with a statistical characterization of a preselected neighborhood relative to the mouse position. Alternatively a region of interest (ROI) can be outlined by accumulation of voxel elements indicated in possibly different 2D slices. A mean feature vector is computed from the data within the selected ROI together with lower and upper bounds for each feature. These characteristics are used to discard voxels in the volume rendering pipeline. Only voxels similar to the specified ROI content are selected and assigned a non-zero opacity value.

The ROI can be implicitly defined as a small neighborhood of the current mouse position. The renderings will then continuously change as the mouse cursor is moved between the various tissue categories in the original 2D slices. Hence, volume rendering becomes an interactive tool for exploration of the implicit geometry contained in the 3D dataset. The stability of the renderings will in addition give visual feedback on the sensitivity and stability of the classification results. Fig. 4 illustrates a possible image display with a 2D slice from the original data set that can be arbitrarily positioned inside the complete 3D dataset. A ROI is indicated in the 2D slice with bright intensities and a 3D rendering from a coarse MRI study of the brain with only 15 parallel slices is shown in the 3D rendering window.

The interactive tissue characterization can be repeated for different tissue categories and integrated in the 3D rendering. Each tissue is then separated by a specific color hue in the 3D rendering.

4 A non-numeric coprocessor design

Fig. 5 gives a block diagram of a PCI board implementing the necessary data routing and extraction facilities for efficient shell rendering [12]. White arrows indicate data paths for volume data and black arrows indicate control signals that can be specified by the host computer. The volume data is read through a stack of four non-numeric coprocessors with 264 Mbytes per second. The sequencing of the volume data is controlled by address generation from the host computer. The addresses are generated in such a manner that the voxels will accumulate in the view plane in either a Front-To-Back (FTB) or Back-To-Front (BTF) order. The FTB order is selected for efficiency reasons if the rendering is generated in software on the host computer. The non-numeric coprocessors are 4 MS160 [15] chips which perform general range queries on the multi-spectral voxel data in real-time.

The MS160 non-numeric coprocessor is a general purpose information filtering processor that has found applications in fuzzy, interactive free text searching, genetic databases, Internet servers (Archie and WWW), volume rendering and knowledge- / property-addressable image databases. The co-

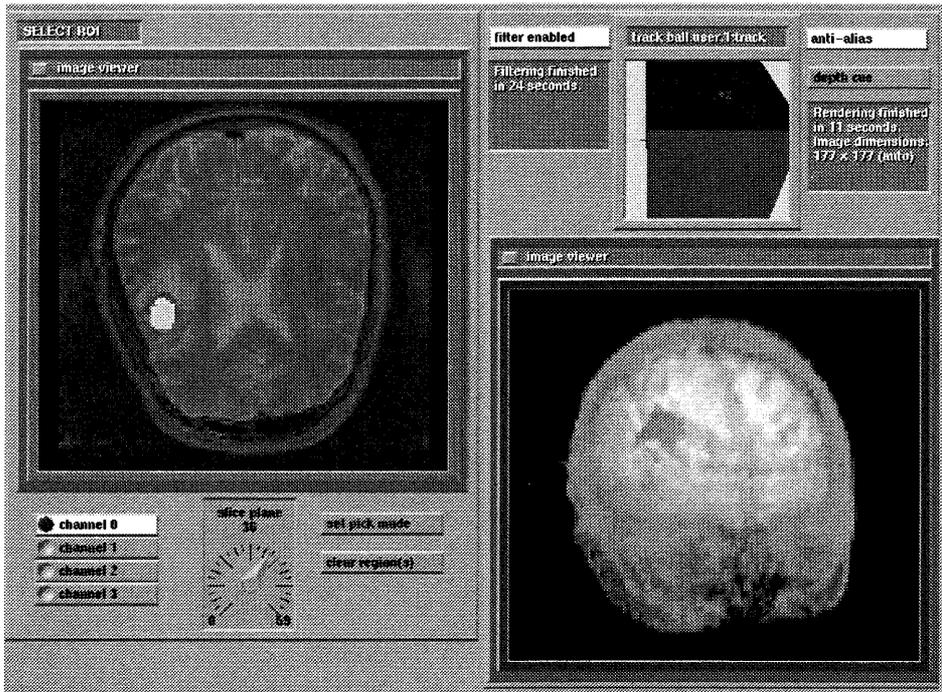


Figure 4: Prototype display in AVS

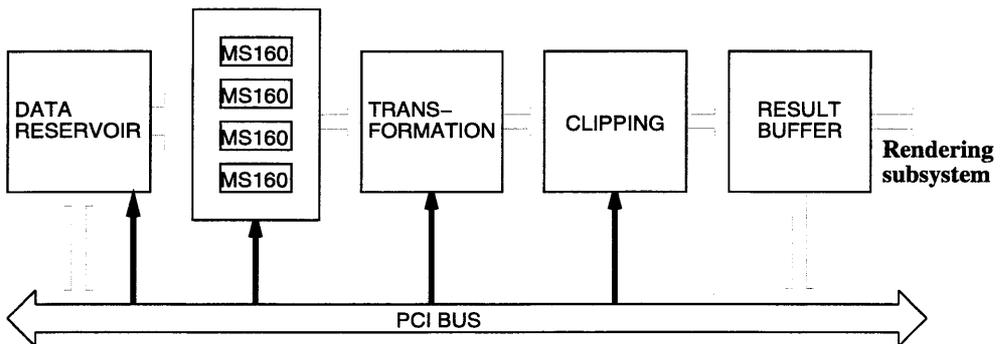


Figure 5: Block diagram for the search engine.

processor has a high bandwidth data input that can read 264 Mbytes per second³ and feed the data through a configurable parallel architecture of 256 processing elements (PEs). The PEs perform a total of more than 10^{10} comparisons per second. These comparisons are combined with match logic in order to provide a single hit indicator for the voxel element being analyzed. The final hit/no-hit decision is used to define which voxel elements that are included in the current volume visualization. The decision can also include contextual information.

The 256 PEs are grouped in 8 windows, each equipped with an array of 32 processing elements. A *window* is thus an evaluation unit for a byte-stream shared with, or distinct from, streams of other windows. Each PE is individually programmed to work on-the-fly. Programming modes include testing for equality, rank and range membership, the latter type having the former and *don't care* as special cases. To support various applications, the word-lengths of the operands are also made programmable. Lengths may vary from 1 to 8 bytes for operations involving e.g. lexical-ordering-testing on textual/numeric strings, or simply large integers. A data distribution network can be programmed to feed input data to specific windows and pipeline the operation of multiple windows in order to handle complex queries.

A MS160 window is performing a completely new evaluation of its resident data for each shift position of its stream. In principal, the window function could have been selected as a threshold operator on a selected metric measuring the distance from a reference vector to the resident window data. A good alternative for such a metric would be the Minkowski p -metric:

$$L_p : d_p(u, v) = \left(\sum_{i=1}^k |u_i - v_i|^p \right)^{\frac{1}{p}} \quad (2)$$

L_1 , L_2 and L_∞ represent the Hamming-/Manhattan-, the Euclidean- and the Maximum-distance metrics respectively. L_∞ can be viewed as a limiting case because all except the maximal vector component difference can be ignored when $p \rightarrow \infty$. The L_p metric has two deficiencies related to fuzzy pattern matching:

- *Noise tolerance*: Individual vector components might contain "garbage". It might for example be interesting to extract patterns that agree in 80 % of the vector components with a reference pattern even if the remaining 20 % of the vector components contain large differences producing high L_p -distances.
- *Arithmetic*: All the L_p metrics, except L_∞ , require heavy computations.

In order to meet these challenges, the window function of the MS160 is based on General Range Queries (GRQ) [15], for which acceptable vectors v relative to a reference vector u should satisfy:

$$\text{GRQ: } \bigwedge_{i=1}^k (|u_i - v_i| \leq \Delta_i) \quad (3)$$

³Operating at 33 MHz.

The reference vector u and the slack parameters Δ_i are parameters that are down-loaded to the MS160 from the host CPU. Future versions of the MS160 will include fuzzy versions of the GRQ definition in Eq. 3 where a hit is reported when a specified number of the k inequalities are satisfied. If more computationally expensive metrics like the L_2 metric is needed, the GRQs can still be applied as a preprocessing step selecting the patterns elements that should be reevaluated with the desired algorithm/metric on for example the host computer.

Finally, the MS160 contains match logic that can evaluate logical combinations of hits reported by individual windows. Arbitrary logical combinations of the 8 windows can be specified. In addition, individual hits reported by a given window can be remembered for a programmable number of input patterns. This property can be utilized to specify contextual selection criteria depending on how the spatial sequencing of the voxel data is organized.

The GRQs and subsequent match logic can be utilized to perform a discrete, statistical classification by approximating a given density distribution with a set of overlapping GRQs. Voxels that don't satisfy the GRQs in the MS160s are discarded. The selected voxels are transferred to a pipelined computation of the viewing transformation. Transformed voxels are finally clipped in the x , y and z directions defined by the view plane before the results are stored in a large result buffer. For each selected voxel element v this result buffer will hold: **1)** An attribute $f(v)$. **2)** A tag $t(v)$ which holds the result of the range queries on the MS160's. The tag can usually be translated into a probability for membership in a certain tissue category and is therefore a useful parameter for opacity assignments. **3)** Transformed coordinates relative to the selected view plane. (x , y and depth z)

The result buffer with extracted voxel information including opacity information, screen location and depth is transferred to the host CPU or an accelerated PCI based rendering subsystem for accumulation of 3D visualizations with the shell rendering algorithm [19, 12].

5 Shell rendering

5.1 A graphics accelerator on the PCI bus

A PCI based accelerator is under development. The rendering engine offers shell rendering of 16 Mvoxels per second. The voxel data is transferred on a direct link from the search engine and will therefore not compete for the bandwidth on the PCI bus. The maximal capacity of the complete system is therefore given by $S \cdot 16$ Mvoxels per second where S is the average data reduction obtained in the classification. The capacity is in addition limited by the maximal rate of 264 Mbytes that can be read from the data reservoir. If we for example assume that each voxel is labelled with eight bytes of information, then 33 Mvoxels can be rendered per second.

5.2 A software prototype based on AVS

A software simulator for the rendering system has been developed using AVS 5.0 (Application Visualization System). The software prototype is currently not optimized for speed. The purpose of the prototype has so far been to validate and optimize algorithmic principles that are included in the hardware accelerators. Both the search engine and the shell rendering functionality of the rendering engine can on the other hand be embedded as general purpose modules in the AVS framework. Experiments with shell rendering of X-CT data can be found in Udupa and Odhners paper [19]. Our contribution is mainly in terms of performance and making the classification/opacity assignment part of the interactive feedback-loop.

5.3 Some aspects of shell rendering

5.3.1 Irregular spatial sampling

The voxel projection paradigm only requires that the spatial 3D world coordinates of every voxel can easily be determined. For regular cuberilles these coordinates can be computed by decoding of the voxel address inside the data reservoir. Irregular sampling can be processed if each voxel element is labelled with attributes that hold the x , y and z world coordinates. These coordinates will in this case be extracted from the voxel and used as input to the transformation unit in Fig. 5. This mode of operation makes it possible to process digital data from medical scanners that record the images in non-regular geometries. The polar geometries usually encountered in ultrasonic scanners are such examples. The 3D acquisition can also consist of irregularly sampled 2D slices that are difficult to convert to a regular cuberille without loss of image detail. 3D acquisition with position sensing is such an example. Even systematic acquisition strategies like tilting of a 2D scanplane with fixed angle increments generate 3D studies that are difficult to convert to a regular cuberille without loss of image detail. The convention into a regular cuberille might generate erroneous interpolations that are included in the 3D renderings.

5.3.2 Mixing 2D slices with 3D renderings

Mixing 2D images with 3D renderings is especially important in medical applications for validation of findings in 3D scenes. A rendering based on surface shading is for example heavily dependent on the underlying boundary extraction mechanism. A hole in the 3D surface might for example either be a physical defect or a consequence of a slightly unfortunate opacity assignment. The interactivity obtained for classification/opacity assignment reduces this specific problem, but still many 3D scenes should be validated with 2D imaging. Fig. 3 illustrates a visualization of two tissue categories (a hemangioma and the portal vein) that are intercepted by a 2D slice from the underlying data set. The 2D slice can be arbitrarily positioned and interactively moved through the 3D scene.

Orthogonal 2D slices can be included in the 3D scene by a proper sequencing of the voxel elements

behind and in front of the selected slice. Arbitrarily positioned 2D slices can be mixed with any number of 3D renderings if a z -buffer is associated with each rendering. In addition, it should be noted that the geometry transformation unit has been duplicated on the hardware accelerators. Both the search engine and the rendering engine include a transformation unit. This is partly to make each of the PCI boards more useful as individual accelerators. The two transformations can also be used to process 2D slices with the following procedure:

1. The address generation to the data reservoir is computed such that a minimal number of voxels including the selected 2D slice are generated.
2. The coordinates of the extracted voxels are transformed and clipped on the rendering engine such that a specified thickness around the selected 2D slice is selected.
3. The extracted voxels are finally transformed on the rendering engine such that the 2D slice is properly adjusted to the selected viewing direction.

5.3.3 Shading and computation of transparent renderings

Transparent views can be constructed by using a scheme due to Levoy [8] where each sample location the ray passes makes a contribution of emitted white light. The contribution is the sampled value ρ multiplied with the corresponding opacity value o . At the same time, incoming light from behind will be filtered by the factor $(1 - o)$. Let the voxel elements arriving at a given location in the view plane be denoted by ρ which consists of the N samples $\rho_1, \rho_2, \dots, \rho_N$. The following attribute is then extracted:

$$L = \rho_1 o_1 + \sum_{i=2}^N \rho_i o_i \prod_{j=1}^{i-1} (1 - o_j) \quad (4)$$

A colored dataset can be rendered by treating each color component (R,G,B) separately using the same formula as above. Depth information can also be recovered by coding depth (D) as an attribute. D can for instance be the distance t to the voxel with the maximal value M . An opacity based estimate of depth is given by:

$$D = \frac{z_1 o_1 + \sum_{i=2}^N z_i o_i \prod_{j=1}^{i-1} (1 - o_j)}{o_1 + \sum_{i=2}^N o_i \prod_{j=1}^{i-1} (1 - o_j)} \quad (5)$$

where z_i denotes the associated depth in the scene.

In many cases, better understanding of the geometry of the structures can be obtained by shading the D image. Fig. 2 illustrates this principle. Shading of “ z ”-buffers is an attractive procedure for computation of diffuse and specular reflections with arbitrarily positioned light sources in the shell rendering framework. Alternatively, local image gradients can be computed and included with the voxel description.

6 Real-time data sources

The data reservoir on the search engine has been designed as a plug in module for memory banks. The memory banks can on the other hand easily be replaced by smaller memory banks and communication circuitry for direct data input from real-time digital data sources. Real-time 3D acquisition of ultrasonic data is currently being explored with both electronic 2D arrays [22] and hybrid control mechanisms [17]. The acquisition rate of 3D ultrasound is limited by the speed of sound. Typically, 6000 beams per second can be used to sample a 3D object at normal depths (10-15 cm). Approximately 3 Mvoxels or 1.5 Mvoxels are generated each second if we assume that each beam is sampled with either 512 or 256 samples respectively. This number should be multiplied with the number of parallel beams that are utilized in the acquisition.

Fig. 6 illustrates a possible architecture for truly real-time 3D in ultrasonic imaging. The figure gives details on the content of the data reservoir in Fig. 5 for real-time applications. The ultrasonic scanner fills the memory bank with a digital 3D study while the search engine reads the previous 3D study into the rendering pipeline as in the post-processing mode of operation with conventional memory banks in the data reservoir.

Fig. 6 indicates a possible attribute content for each of the digital voxel elements. 3 attributes (6, 7 and 8) are used to give the true 3D coordinates in a world coordinate system. These values depend on the beam steering principles and can be precomputed. The labelling of voxels with spatial 3D coordinates makes it possible to render digital data recorded with an arbitrary scanner geometry. It might still be useful to utilize the polar-to-Euclidean scanconverter hardware found in all ultrasound scanners. The pixel resolution in the scanconverted, Euclidean 2D images should in this case correspond to the selected pixel resolution in the view plane where the rendering is accumulated. Five bytes are used to characterize the voxel content.⁴ In addition to displayed amplitude data, the ultrasound scanner can utilize this space to label the voxels with information extracted by digital signal processing of the RF signal. This information could include both related modalities and digital image processing like Power Doppler, color flow, blood/tissue characterization, contrast agent indicators, image gradients, distance transforms, etc. These attributes can then be utilized both for data extraction and as a basis for opacity and color assignments. This selection of parameter setup would give a rendering capacity of 33 Mvoxels per second if the subsampling factor obtained in the data extraction exceeds 2.

The rendering capacity in this specific example should be compared with the acquisition rate of 3 or 1.5 Mvoxels per second per beam. If we assume that each beam is sampled with 256 samples, then a maximum of 22 parallel beams can be utilized in the acquisition and still truly real-time display of 3D visualizations can be achieved.⁵ For lower numbers of parallel beams the high rendering rates can be utilized to either render each acquisition from different views in each update or data from a longer acquisition period could be integrated to a 3D or 4D study with better spatial and temporal resolution.

⁴Different parameter setups for voxel characterization are of course possible.

⁵The theoretical maximum is eight times higher: $8 \cdot 22 = 176$ parallel beams.

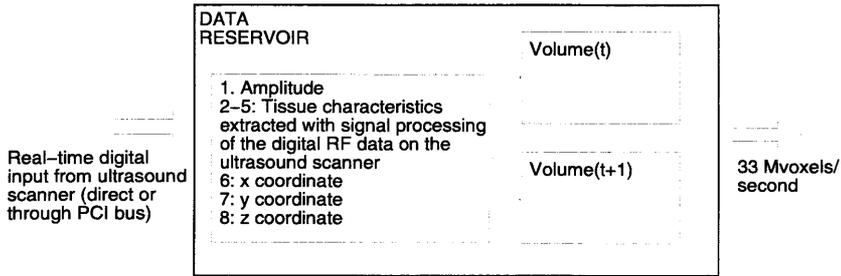


Figure 6: Example of possible modification of the data reservoir for digital interfacing with ultrasonic scanners delivering real-time 3D data.

7 Conclusion

This paper has presented a methodology and hardware accelerators for real-time volume rendering with interactive feedback on classification and opacity assignments. Dedicated hardware it utilized to extract voxels according to feature characteristics before volume renderings are generated with the voxel projection paradigm of shell rendering. Visual feedback in the 3D scene is obtained both for modification of viewing parameters and for modification of automatic or interactive classification parameters. Typical performance figures are in the range of 33 to 264 Mvoxels per second. In particular, we describe how the hardware accelerators can be utilized to achieve truly real-time imaging with 3D ultrasonics.

8 Acknowledgments

The authors would like to thank Hafslund Nycomed, MRT International and the Norwegian Research Council for supporting this study. The authors would also like to thank K.P. Schipper at the regional hospital in Trondheim and T. Hausken at the regional hospital in Bergen for providing data for the liver vein study and the hemangioma study respectively. The AVS prototype for shell rendering was initially developed by Robert Schmidt.

References

- [1] L. Axel, G. T Herman, and J. K Udupa et.al. Three-dimensional display of NMR cardiovascular images. *Comput. Assist. Tomography*, 7:172-174, 1983.
- [2] Michael Bomans, Karl-Heinz Hohne, Ulf Tiede, and Martin Riemer. 3D segmentation of mr images of the head for 3D display. *IEEE Trans. on Medical Imaging*, 9(2), June 1990.

- [3] G Borgefors. Distance transformations in arbitrary dimensions. *Computer vision, graphics and image processing*, 27:321–345, 1984.
- [4] R. A. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. *Computer Graphics*, 22(4):65–74, August 1988.
- [5] G. Frieder and R. Reynolds. Back-to-front display of voxel-based objects. *IEEE Computer Graphics and Application*, 5(1):52–60, January 1985.
- [6] D.C Hemmy, D.J David, and G.T Herman. Three-dimensional reconstruction of craniofacial deformity using computed tomography. *Neurosurgery*, 13:534–541, 1983.
- [7] J. T. Kajiya and B. P. Von Herzen. Ray tracing volume densities. *Computer Graphics (Proc. Siggraph)*, 18(3):165–173, July 1984.
- [8] M. Levoy. Volume rendering: Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, pages 29–37, May 1988.
- [9] William E. Lorensen and Harvey E. Cline. Marching cubes: a high resolution 3D surface construction algorithm. *Computer Graphics*, 21(4):163–169, July 1987.
- [10] T. R. Miller, J. B Starren, and R. A. Grothe. Three dimensional display of positron emission tomography of the heart. *J. Nuclear Medicine*, 29(4):530–537, 1988.
- [11] B. Olstad, S. Berg, S. Eik-Nes, and K. P. Schipper. 3d transvaginal ultrasound imaging for identification of endometrial abnormality. In *Proceedings of SPIE Medical Imaging '95*, San Diego, 1995.
- [12] B. Olstad, Erik Steen, and Olav Sandstå. Shell rendering with hardware supported data extraction. In *Proceedings of SPIE Medical Imaging '95*, San Diego, 1995.
- [13] E. Steen and B. Olstad. Scale-space and boundary detection in ultrasonic imaging, using signal-adaptive anisotropic diffusion. In *SPIE Medical Imaging, Image processing conference*, 1994.
- [14] E. Steen and B. Olstad. Volume rendering of 3d medical ultrasound data using direct feature mapping. *IEEE Trans. on Medical Imaging*, 13(3):517–525, september 1994.
- [15] E. Stephansen. *MSI60, Data book*. MRT International, 1993.
- [16] T. Taxt and A. Lundervold. Multispectral analysis of the brain using magnetic resonance imaging. *IEEE Trans. on Medical Imaging*, 13(3):470–481, September 1994.
- [17] H. Torp, S. Dørum, E. Holm, E. Steen, K. Kristoffersen, K. A. Berg, H.J. Alker, and B. Olstad. Real-time three dimensional echocardiography for assessment of valvular function. *Presented at 11th Symposium on Echocardiology*, 1995.
- [18] W. G. Totty and M. W Vannier. Analysis of complex musculoskeletal anatomy using three-dimensional surface reconstruction. *Radiology*, 150:173–177, 1984.

- [19] J. K. Udupa and Dewey Odhner. Shell rendering. *IEEE CG& A.*, pages 58–67, November 1993.
- [20] M.W Vannier, S. Gronemeyer, and F.R Gutierrez et al. Three-dimensional magnetic resonance imaging of congenital heart disease. *Radiographics*, 8(5):857–871, 1988.
- [21] M.W Vannier, J. L Marsh, and J. O. Warren. Three-dimensional ct reconstruction for cranio-facial surgical planning and evaluation. *Radiology*, 150:179–184, 1984.
- [22] O.T. von Ramm, S.W. Smith, and H.G. Pavy Jr. High-speed ultrasound volumetric imaging system - part i-ii: transducer design and beam steering. *IEEE Trans. on. Ultrasonics, Ferro-electrics and Frequency control*, 38, March 1991.
- [23] S. Webb, R. J Ott, and M. A. Flower et al. Three-dimensional display of data obtained by single photon emission tomography. *Brit. J. Radiology*, 60:557–562, 1987.

the 1990s, the number of people in the UK who are employed in the public sector has increased from 10.5 million to 12.5 million (12% of the population).

There are a number of reasons for this increase. One is that the public sector has become a more important part of the economy. Another is that the public sector has become more efficient. A third is that the public sector has become more attractive to workers. A fourth is that the public sector has become more diverse.

The public sector has become a more important part of the economy. This is because the public sector has become more efficient.

The public sector has become more efficient. This is because the public sector has become more attractive to workers.

The public sector has become more attractive to workers. This is because the public sector has become more diverse.

The public sector has become more diverse. This is because the public sector has become more important to the economy.

The public sector has become more important to the economy. This is because the public sector has become more efficient.

The public sector has become more efficient. This is because the public sector has become more attractive to workers.

The public sector has become more attractive to workers. This is because the public sector has become more diverse.

The public sector has become more diverse. This is because the public sector has become more important to the economy.

The public sector has become more important to the economy. This is because the public sector has become more efficient.

The public sector has become more efficient. This is because the public sector has become more attractive to workers.

The public sector has become more attractive to workers. This is because the public sector has become more diverse.

The public sector has become more diverse. This is because the public sector has become more important to the economy.

The public sector has become more important to the economy. This is because the public sector has become more efficient.

The public sector has become more efficient. This is because the public sector has become more attractive to workers.

The public sector has become more attractive to workers. This is because the public sector has become more diverse.

The public sector has become more diverse. This is because the public sector has become more important to the economy.

The public sector has become more important to the economy. This is because the public sector has become more efficient.

The public sector has become more efficient. This is because the public sector has become more attractive to workers.

The public sector has become more attractive to workers. This is because the public sector has become more diverse.

The public sector has become more diverse. This is because the public sector has become more important to the economy.

The public sector has become more important to the economy. This is because the public sector has become more efficient.

The public sector has become more efficient. This is because the public sector has become more attractive to workers.

The public sector has become more attractive to workers. This is because the public sector has become more diverse.

The public sector has become more diverse. This is because the public sector has become more important to the economy.

An efficient algorithmic framework for fuzzy region oriented segmentation

Abstract

An algorithmic framework for fuzzy region oriented segmentation is presented. The framework is aimed at volume rendering methods, which by definition are fuzzy display techniques. In several volume rendering methods, opacity values are assigned to voxels (volume elements) in order to emphasize the “interesting” parts of the data and make “uninteresting” parts more or less transparent. The opacity assignments may be thought of as a fuzzy segmentation or classification procedure which allows diffuse transitions between different materials. Opacity values are usually computed as a function of local feature vectors associated with each voxel. In many cases this kind of segmentation will be insufficient, as different objects may share similar local features. Still it may be possible to isolate an object as a region of spatially connected voxels. Traditional region oriented segmentation methods divide the image into disjoint regions and will therefore not fit well with the volume rendering paradigm. The algorithmic framework presented in this paper can be viewed as a fuzzy region growing technique. Each connection between neighboring voxels is given a weight which is a function of the difference in image characteristics between these voxels and a set of seed voxels. Dijkstras algorithm is utilized for computation of a minimum weight path from the seed voxels to all other voxels in the 3D image. The weight of this path is then mapped into an opacity value. Different weight functions are discussed. The proposed framework is especially useful for interactive manipulation of 3D images. The practical utility of the algorithm therefore depends a lot on its efficiency. A simple and very efficient implementation of Dijkstras algorithm which exploits some properties specific to our application has been used. Numerical experiments have been conducted in order to evaluate the proposed framework. Finally, the presented algorithm is demonstrated for visualization of tumor and vessel geometry using three dimensional ultrasonic images.

1 Introduction

Surface rendering and *volume rendering* are two approaches to visualization of 3D objects. In surface rendering, an explicit model of the imaged structure is extracted by using a surface detection scheme. This approach relies heavily on the robustness of the detection scheme. On the other hand, volume rendering techniques do not rely on explicit models of the imaged objects. The imaged scene is rather viewed as a semitransparent volume, where transitions between different materials are allowed to be diffuse. A number of algorithms have been proposed for volume rendering of medical data [2, 15, 7, 10, 9, 8, 13, 11]. In several of these algorithms, opacity values are assigned to each voxel on the basis of local characteristics. In the simplest case, opacity values are given as the scalar voxel values multiplied by a global scaling factor.

In Drebins algorithm [2], different opacity values are assigned to each material present in the image. Statistical analysis of the scalar or vector valued voxels is used to decide the amount of each material present within a voxel. The opacity of a voxel is computed as a normalized sum of opacities of all materials present within the voxel, where the opacity value of a material is weighted

by the probability that the material is present within the voxel. A specific material can be emphasized by assigning a high opacity value to it and low opacity values to other materials in the image. Drebin states that only probabilistic classification schemes should be used, since binary classification may introduce artifacts in the rendered images. This is obviously true if the distributions of feature vectors from the different materials overlap. The uncertainty inherent in the outcome of the classification can be built into the opacity function with Drebins algorithm.

In many cases it can be difficult to isolate an object purely on the basis of local features, since different objects in the image may share the same local characteristics. Still, an object may be isolated as a region of spatially connected voxels. Region oriented segmentation methods [6] view segmentation as a process of partitioning the image into *disjoint* regions. Since the regions have to be disjoint, traditional region oriented segmentation methods do not fit well with the volume rendering paradigm, since diffuse boundaries between different regions are not allowed.

A well known strategy for region oriented segmentation is called region growing. A basic approach to region growing is point aggregation, where regions are grown from a set of seed points by iteratively appending neighbor points with similar properties until no more such points can be found. This approach eventually leads to a hard partition of the image into disjoint regions.

In this paper we present an algorithmic framework which can be viewed as a fuzzy approach to region growing. Each connection between neighboring voxels is given a weight which is a function of the difference in image characteristics between these voxels and a set of seed voxels. Dijkstras algorithm [1] is utilized for computation of a minimum weight path from the seed voxels to all other voxels. The weight of this path is then mapped into an opacity value.

Interactive adjustment of the segmentation parameters is a necessity in many applications. This requires that the algorithm has a low computational complexity. In most practical applications we can map opacities into a set of discrete values. With this assumption in mind, we have made a simple and very efficient implementation of Dijkstras algorithm with a worst case running time proportional to the number voxels in the 3D image. The work presented here is a further development of work which was presented in a preliminary paper [12].

Udupa and Samarasekera [14] have, independently of this work, recently presented a fuzzy-set theoretic approach to multidimensional image segmentation. In their framework, an imaged scene was represented as a *fuzzy digital space*. Udupa and Samarasekera based the notion of connectedness on the mathematical theory of fuzzy sets, and relations between voxels were restricted to be reflexive and symmetric. They developed an iterative dynamic programming scheme in order to determine fuzzy connectedness. However, no explicit analysis of the time complexity of the algorithm was given. The algorithm proposed in this paper can also be used to compute fuzzy connectedness as defined by Udupa and Samarasekera.

In the next section we review some basic concepts on image representations, connectivity and segmentation. In section 3 we introduce the concepts of weighted connectivity, weighted paths and fuzzy region oriented segmentation. Dijkstras algorithm is presented as an efficient algorithm for solving the fuzzy region growing problem. Numerical experiments are conducted in order to eval-

uate the proposed algorithm. These experiments are presented in section 5. The presented algorithm is also demonstrated for visualization of tumor and vessel geometry using three dimensional ultrasonic images. Appendix *A* describes a highly efficient implementation of Dijkstras algorithm. Finally, appendix *B* relates the algorithmic framework presented in this paper to the fuzzy segmentation framework presented by Udupa and Samarasekara.

2 Image representation, K -connectivity and region oriented segmentation

The following is partially taken from [6]. A 2D image is usually represented as a 2D array of regularly spaced image elements, called pixels. An attribute vector $\mathbf{A}(p(x, y))$ is associated with each pixel $p(x, y)$. A pixel $p(x, y)$ at coordinates (x, y) has four horizontal and vertical (nearest) neighbors which are called 4-neighbors, denoted $N_4(p(x, y))$. Similarly, a 8-neighborhood can be defined by also considering diagonal pixels.

K -connectivity is defined as follows:

Definition 1

Two pixels $p(x, y)$ and $q(s, t)$ are said to be K -connected if they are K -neighbors and if $\mathbf{A}(p(x, y)) \in C$ and $\mathbf{A}(q(s, t)) \in C$, where C is the set of attribute values defining connectivity.

Definition 2

A K -path between two pixels $p(x, y)$ and $q(s, t)$ is a sequence of distinct pixels with coordinates $(x_0, y_0), \dots, (x_n, y_n)$ where $(x_0, y_0) = (x, y)$, $(x_n, y_n) = (s, t)$ and (x_{i-1}, y_{i-1}) is K -connected with (x_i, y_i) , $i \in \{1..n\}$.

Two pixels $p(x, y)$ and $q(s, t)$ in an image subset S are said to be *connected* in S if there exist a path between them consisting entirely of pixels in S . For any pixel $p(x, y)$ in S , the set of pixels in S that are connected to $p(x, y)$ is called a *connected component* of S .

A 3D image is usually represented as a 3D array of regularly spaced volume elements, called voxels, each with an associated attribute vector $\mathbf{A}(v(x, y, z))$. A voxel $v(x, y, z)$ at coordinates (x, y, z) has six nearest neighbors denoted $N_6(v)$. A 26-neighborhood can be defined by also considering all the diagonal voxels. Definitions of K connectivity and K -paths are analog to the 2D case.

2.1 Region oriented segmentation

Given the entire image I , traditional region oriented segmentation methods partitions I into n sub-regions, I_1, \dots, I_n such that:

- a) $\cup_{i=1}^n I_i = I$
- b) I_i , $i \in \{1..n\}$ consist only of connected pixels (or voxels)
- c) $Pr(I_i) = TRUE$, $i \in \{1..n\}$
- d) $I_i \cap I_j = \emptyset$ for all i and j , $i \neq j$

where $Pr(I_i)$ is a logical predicate defined over the points in I_i , and \emptyset is the empty set.

Condition a) means that the segmentation must be complete.

Condition b) means that the all points in a region must be connected.

Condition c) deals with the properties which must be shared by all points in a region.

Condition d) means that the regions must be disjoint.

Condition d) implies that diffuse transitions between different regions are not allowed. We therefore have to relax this condition in order to develop region oriented segmentation algorithms useful for volume rendering.

3 Weighted K -connectivity, weighted K -paths and fuzzy region growing

In this section we will introduce the concepts of weighted K -connectivity and weighted K -paths. Although we will relate the presentation to 3D images, all the concepts will be equally relevant for 2D images.

3.1 Weighted K -connectivity

The basic definition of K -connectivity does not account for situations where distributions of feature vectors within different materials overlap (which is the normal case for most imaging modalities.) We therefore introduce the concept of *weighted K -connectivity*:

Definition 3

Any two voxels $p(x, y)$ and $q(s, t)$ that are K neighbors are K -connected with a weight $w_M(p(x, y), q(s, t)) \in \mathbb{R}^+$.

In general, no restrictions are given on w_M except that it can not be negative. In practice, w_M may depend on $\mathbf{A}(p(x, y))$, $\mathbf{A}(q(s, t))$ and maybe also on the coordinates of $p(x, y)$ and $q(s, t)$. In the following we will assume that the weight $w_M(p(x, y), q(s, t))$ *decreases* as the probability that $p(x, y)$ and $q(s, t)$ belongs to the region which we are interested in, *increases*.

For now, we will only mention two possible weight functions which are based on image statistics.

Assume that the attribute values of voxels sampled from a material M are distributed according to a multidimensional normal distribution. The probability that a particular voxel $q(s, t)$ is a sample of M are:

$$P(q(s, t) \text{ sample of } M) = (2 \cdot \pi)^{-\frac{d}{2}} \left| \Sigma \right|^{-\frac{1}{2}} e^{-\frac{1}{2}((\mathbf{A}(q(s, t)) - \mu)' \Sigma^{-1} (\mathbf{A}(q(s, t)) - \mu))} \quad (1)$$

Here μ is the mean feature vector and Σ is the co variance matrix which together define the distri-

bution of attribute values sampled from the material M . A possible weight function is:

$$w_M(p(x, y), q(s, t)) = 1.0 - e^{\frac{-1}{2}((\tilde{q}(s,t)-\mu)'\Sigma^{-1}(\tilde{q}(s,t)-\mu))} \quad (2)$$

Here $\tilde{q}(s, t) = q(s, t) * G_{m \times m}(s, t)$, where $*$ is the convolution operator and $G_{m \times m}(s, t)$ is a Gaussian kernel of size $m \times m$. By increasing m , the weight will be less influenced by noise, at the cost of less spatial resolution. Note that this weight function is non-symmetric.

The following symmetric weight function is a modified variant of a scalar weight function proposed by Udupa and Samasekara [14].

$$w_{M_2}(p(x, y), q(s, t)) = 1.0 - e^{((\tilde{q}(s,t)+\tilde{p}(x,y))\cdot\frac{1}{2}-\mu)'\Sigma^{-1}((\tilde{q}(s,t)+\tilde{p}(x,y))\cdot\frac{1}{2}-\mu))} \quad (3)$$

The multivariate student distribution may be used to derive similar weight functions if the distribution of attribute values have heavy tails compared to the normal distribution.

The parameters μ and Σ can sometimes be obtained in advance by training. In other cases, it may be difficult to perform a general training, and it will then be more appropriate to use the sample mean vector and sample covariance matrix computed directly from a user defined region within the object of interest.

3.2 Weighted K -paths

In the previous subsection we introduced the concept of weighted K -connectivity. We also need to define the concept of a *weighted K -path*.

Definition 4

A *weighted K -path P between two voxels p_1 and p_n is a sequence of distinct voxels $p_1..p_n$, where p_{i-1} is a K -neighbor of p_i , $i \in \{1..n\}$.*

Definition 5

The weight of a path $w(P)$ is a function $f(w(p_0, p_1), \dots, w(p_{n-1}, p_n)) \in \mathbb{R}^+$ of the weights along the path.

In the previous subsection we assumed that the weight of a connection between two voxels $p(x, y)$ and $q(s, t)$ decreases with the probability that $p(x, y)$ and $q(s, t)$ both belong to the interesting region. To be consistent with this assumption, we only want to deal with weight functions that are monotonically increasing with the length of the path. This means that each time a path is expanded with an additional weighted connection, the weight of the path will either increase or remain the same. If the weight of a path between two voxels $p(x, y)$ and $q(s, t)$ is large, then it is likely that at least some of the voxels along the this path do not belong to the interesting material. We have found the following functional form of f useful for practical applications:

$$w(P) : f(w(p_0, p_1), \dots, w(p_{n-1}, p_n)) = \left(\sum_{i=1}^{n-1} w(p_i, p_{i+1})^\gamma \right)^{\frac{1}{\gamma}} \quad (4)$$

We will later refer to this general weight function as the f_γ function. The f_γ function can be related to the Minkowski L_p metric, $L_p : d_p(\eta, \xi) \rightarrow \mathfrak{R}^n$ for measuring the distance between two n -dimensional vectors ξ and η .

As γ increases, the weight of the path will be more and more influenced by the largest weight along the path. For $\gamma = 1$, the weight of a path is simply the sum of weights along the path. Hence, the f_1 weight function will depend strongly on the length of the path. $\gamma = \infty$ can be viewed as a limiting case because all except the maximal weight can be ignored.

In general, the number of paths between any two voxels p_1, p_n will grow exponentially with the total number of voxels in the image. In the set $Y(p_1, p_n)$ of all possible paths connecting p_1 and p_n there are at least one with a minimum weight $\delta(Y(p_1, p_n))$. This minimum weight $\delta(Y(p_1, p_n))$ defines the weight of the connection between p_1 and p_n .

Definition 6

$$\delta(Y(p_1, p_n)) = \min_{P \in Y(p_1, p_n)} w(P)$$

3.3 Region growing on images with weighted connections

According to the definitions of weighted K -connectivity and weighted K -paths, all voxels in a regular grid will be connected. In order to do segmentation, it is useful to define a threshold δ_{Max} , which limits the maximal weight of a path.

Definition 7

Two voxels p_1 and p_n are said to be connected in a hard sense if $\delta(Y(p_1, p_n)) \leq \delta_{Max}$

A fuzzy variant of the region growing problem can now be formulated as follows:

Region growing on images with weighted connections

Given a complete image I , a seed s , or more generally a set of seeds S , compute $\delta(Y(S, p(x, y)))$ for all $p(x, y) \in I_{\delta_{Max}}$, where $I_{\delta_{Max}}$ is the set of voxels satisfying $\delta(Y(S, p(x, y))) \leq \delta_{Max}$

For the purpose of volume rendering, $\delta(Y(S, p(x, y)))$ can be mapped into opacity values for all $p(x, y) \in I$ by a simple scaling:

$$o(u) = \begin{cases} 0 & \text{if } \delta(Y(S, p(x, y))) > \delta_{Max} \\ \beta \cdot \frac{\delta_{Max} - \delta(Y(S, p(x, y)))}{\delta_{Max}} & \text{otherwise} \end{cases} \quad (5)$$

where β serves as a scaling constant. It is important that δ_{Max} is set sufficiently high so that all parts of the object of interest are assigned positive opacity values.

In many cases, it is useful to render different objects simultaneously. This can be done by assigning a color C_i and opacity scaling constant β_i to each individual object $i \in \{1..i_{Max}\}$, as proposed by Drebin et. al. [2]. The following opacity function is useful for rendering of the colored objects:

$$\begin{aligned} o_i(u) &= 0 && \text{if } \delta(Y(S_i, p(x, y))) > \delta_{Max_i} \\ o_i(u) &= \beta_i \cdot \frac{\delta_{Max_i} - \delta(Y(S_i, p(x, y)))}{\delta_{Max_i}}, && \text{otherwise} \end{aligned} \quad (6)$$

Here S_i is the seed region within object i and δ_{Max_i} is the threshold used to define the extent of the object. o_i is the opacity ‘‘contribution’’ from object i . Given o_i for all $i \in \{1..i_{Max}\}$, the color $C(u)$ of a specific voxel u can be determined as follows:

$$C(u) = \frac{\sum_{i=1}^{i_{Max}} C_i \cdot o_i(u)}{\sum_{i=1}^{i_{Max}} o_i(u)} \quad (7)$$

Similarly, the opacity of a given voxel can be determined by using the following equation:

$$o(u) = \frac{\sum_{i=1}^{i_{Max}} o_i(u)}{\sum_{i=1}^{i_{Max}} \beta_i} \quad (8)$$

It is of course also possible to combine this opacity function with other opacity functions computed from local features.

In the next section we will present an efficient solution to the fuzzy region growing problem, using Dijkstras algorithm for finding shortest paths in directed graphs.

4 An efficient algorithm for fuzzy region growing

In this section we will present Dijkstras algorithm as an efficient solution to our fuzzy region growing problem. Dijkstras algorithm works on directed graphs with weighted edges. We will therefore first review some basic graph related concepts.

4.1 Basic graph related concepts

Definition 8

A directed graph $G(V, E)$ with weighted edges consists of:

- A set of vertices (or nodes) V .
- A set of edges (or connections) E .
- A weight $w(e(u, v))$ assigned to each edge $e(u, v) \in E$.

Note that, since the graph is directed, $e(u, v) \in E$ does not imply $e(v, u) \in E$. Also, if $e(u, v) \in E$ and $e(v, u) \in E$, $w(e(u, v))$ may or may not be equal to $w(e(v, u))$.

An undirected graph, which is a special case of a directed graph, is a graph where $e(u, v) \in E \implies e(v, u) \in E$ and $w(e(u, v)) = w(e(v, u))$, for all $e(u, v) \in E$.

A 2D or 3D image can be directly mapped into a graph $G(V, E)$, where each image element is represented as a vertex $u \in V$, and where each weighted connection between image elements is represented as an edge $e(u, v) \in E$ with a weight $w(e(u, v))$.

Definition 9

A path P is a sequence of vertices u_i , $i \in \{1..n\}$ such that $e(u_i, u_{i+1})$, $i \in \{1..n-1\}$ are edges .

In traditional graph related problem, the *weight* of a path is defined as the sum of all weights along the path. For our purposes, the weight of a path will be defined according to equation 4:

$$w(P) = \left(\sum_{i=1}^{n-1} w(e(u_i, u_{i+1}))^\gamma \right)^{\frac{1}{\gamma}} \quad (9)$$

For $\gamma = \infty$, the weight of a path can be computed efficiently as follows:

$$w(P) = \max_{i=1}^{n-1} w(e(u_i, u_{i+1})) \quad (10)$$

4.2 Dijkstras algorithm

The fuzzy region growing problem stated in section 3.3 can be transformed directly into the following graph problem:

Given a source vertex s , or more generally a set of source vertices S , compute the minimum weight path $\delta(S, v)$ for all $v \in V_{\delta_{Max}}$, where $V_{\delta_{Max}}$ is the set of vertices satisfying $\delta(S, v) \leq \delta_{Max}$

A well known solution to the shortest path problem is Dijkstras algorithm [1].

In the following, the set of vertices to which the minimum weight path from any vertex in S has been determined is denoted R . Q is the set of remaining vertices $V - R$.

Q is implemented as a priority queue. A priority queue is an abstract datatype which supports basic queue operations (such as *Insert* and *Delete*) and in addition the special operation *ExtractMin* which extracts the element from Q which has the highest priority (In our case, the vertex u with the currently smallest estimate for $\delta(S, u)$.)

An array $d[u]$ is used to hold the current minimum weight estimate for all nodes $u \in V$. $d[u]$ is initialized so that for all vertices $u \in V$ except the source vertices $u \in S$, $d[u]$ are set to infinity. For all source vertices $u \in S$, $d[u] = 0$.

In the following, $N_k[u]$ is the set of the k neighbors of u . If the topology of the graph is a regular $2D$ array, we may have $k = 4$ or $k = 8$. For $3D$ arrays, we may have $k = 6$ or $k = 26$.

The algorithm can be stated as follows:

- (1) For all $u \in V - S$, set $d[u] = \infty$
- (2) For all $u \in S$, set $d[u] = 0$
- (3) $R \leftarrow \emptyset$ // Initially, R should be empty
- (4) $Q \leftarrow V$ // Put all vertices from V into the queue Q
- (5) **while** $Q \neq \emptyset$
- (6) **do** $u \leftarrow ExtractMin(Q)$
- (7) **if** $d[u] > \delta_{Max}$ **then break**
- (8) $R \leftarrow R \cup \{u\}$
- (9) **for each vertex** $v \in N_k[u]$
- (10) **do Relax** (u, v, w)

By termination, the algorithm will have determined the minimum weight path from S to all vertices $u \in V_{\delta_{Max}}$. The minimum weights are then given in $d[u]$, $u \in V_{\delta_{Max}}$. It is important to note that once $d[u]$ have been determined for $\delta_{Max} = T$, $d[u]$ can be determined for all $\delta_{Max} \in [0, T]$ simply by thresholding $d[u]$

The relax operation compares the current weight estimate $d[v]$ with the weight of the alternative path arriving at v through u . For instance, if f_1 is used as weight function, the weight of the alternative path will be $d[u] + w(e(u, v))$. $d[v]$ is then updated by the weight of the alternative path if the weight of the alternative path is smaller than $d[v]$. A very important property of the *Relax* operation is that it either decreases an estimate or leaves it unchanged.

Dijkstras algorithm is known to solve the single source shortest path problem when the weight of a path is simply the sum of the weights $w(e(u, v))$ along the path. A crucial property of Dijkstras algorithm is that once a vertex u is included in R , there can be no other path to u with a smaller weight. To see why there can not be any other shorter path to u , imagine that a possible shorter path goes from S to a vertex $x \in V - R$, and then arrives at u . But if this is the case, the path from S to x must be shorter than the path from S to u and x would therefore have been selected by the *ExtractMin* operation (in line 5) *before* u . The key assumption here is that each time a path is expanded, the weight of the path either increases or remains the same. This assumption is also satisfied by the weight functions given in equation 9 and 10. A formal proof that Dijkstras algorithm really works can be found in [1].

There are at most $|V|$ *ExtractMin* operations and $|E|$ relax operations in Dijkstras algorithm. The priority queue can be implemented as a binary heap [1]. Then each *ExtractMin* and *Relax* operation will execute in $\log(V)$ time. The best known general implementation of Dijkstras algorithm uses a fibonacci heap [1] for the priority queue and thereby achieves a running time of $O(V \log V + E)$

4.3 Efficient implementations of Dijkstras algorithm

In some cases, it is possible to solve the single source shortest path problem more efficiently than in the general case. Ahuja et. al. [4] have reported an algorithm that runs in $O(E + V\sqrt{\log w_{Max}})$ time, which can be used if edge weights are integer values in the range $0..w_{Max}$. They also developed a simpler variant that runs in $O(E + V\sqrt{w_{Max}})$ time.

In practical applications we can map each weight to a discrete value in the range $0..w_{max}$. The algorithms due to Ahuja et. al could therefore be used. We may however also map the minimum path weights to discrete values in the range $0..\delta_{Max}$, since we are only interested in paths with a weight equal to or smaller than δ_{Max} . With this additional property in mind, we have developed an even simpler variant of the algorithm due to Ahuja et. al with a running time proportional to the number of edges in the graph. The algorithm is described in Appendix A.

5 Numerical experiments

5.1 Evaluation of weight functions

In order to evaluate the proposed segmentation algorithm, we first made various synthetic 2D images to which we added Gaussian noise. A 2D version of the segmentation algorithm was used during the evaluations. Different weight functions were tested, but the one given in eq. 3, turned out to give the best results. The weight function given in eq. 2 turned out to be somewhat more noise sensitive. A Gaussian operator with a mask size of 3×3 was used in both cases. The f_γ function was used for computing path weights, with $\gamma \in \{1, 2, \infty\}$

We created an artificial image of size $256 * 256$ consisting of a background region with pixel values equal to 120 and several foreground regions with pixel values 136. The image is shown in figure 2. The foreground regions are circular discs, some of which are connected, and some of which are close to be connected. In the middle of the image, there is a disc which is connected to another disc above and also one below it. The goal was then to isolate this region without connecting any of the other discs.

In one experiment we blurred the image with a Gaussian mask of size 5×5 and added Gaussian noise with $\sigma = 8.0$. Different seed regions within the object and various threshold levels (δ_{Max}) were used as input to the algorithm. Signal statistics were computed from the seed regions. The computed path weights were mapped into gray level values using the opacity function given in equation 5. As can be seen from figure 3 it was possible to isolate the object of interest using any of f_1 , f_2 or f_γ as a weight function by selecting an appropriate threshold value. The f_∞ function gave, as expected, segmentation results which were least sensitive to the location of the seed region within the object of interest. The implementation turned out to be very efficient. The algorithm spent less than 0.4 seconds in generating the results shown in the figures on a Sparc-10/51 workstation. In the worst case, when the threshold was set such that the whole image was included as a part of

the object, the algorithm spent less than 2.0 seconds.

We repeated the previous experiment but with Gaussian noise with $\sigma = 16.0$ added to the original image. (shown in figure 2). In this case, the segmentation results depended much more on the weight function used for computing path weights. Results are shown in figure 4. The f_1 weight function gave segmentation results which were more dependent on the location of the seed region within the object of interest, but were least sensitive to noise. The f_∞ function gave, as in the previous experiment, segmentation results which were least sensitive to the location of the seed region within the object of interest, but several noisy pixels within the object of interest were not included. The f_2 function gave segmentation results which were less sensitive to the location of the seed region compared the results obtained by using the f_1 function, and was less sensitive to noise than the f_∞ function.

Several other experiments were made with both synthetic and real 2D images in order to evaluate the different weight function. From these experiments we conclude that the f_∞ weight function seems to be most appropriate to use for object segmentation if the noise level is low. We have found the f_2 weight function to be a better alternative if the noise level is higher, especially if the images contain impulsive noise. The f_1 weight function is least sensitive to noise and was found to give the best results in some cases where there were considerable signal variations within the object of interest. However, the segmentation results are generally sensitive to the location of the seed region within the object of interest with this weight function. The f_1 weight function can be used more successfully by giving several different seed regions within the object of interest as input to the algorithm.

The fuzzy region growing algorithm can also be used interactively to *erase* parts of the 3D image which occludes an interesting object. In such cases it will be desirable that only voxels which are close to the location of the seed region are erased, and that no noisy voxels are left unerased. The f_1 weight function may then be the best alternative.

5.2 Experiments with ultrasonic images

Udupa and Samarasekara demonstrated their fuzzy segmentation framework on 2D and 3D MR-images and reported very satisfactory results for this application. In this paper we will focus on applications in medical ultrasound imaging. Ultrasound images are generally suffering from a low signal to noise ratio compared to MR-images, and automatic segmentation is therefore in general quite difficult [11].

The problem of finding useful visualization methods for display of tumor and vessel geometry from 3D ultrasonic data is a clinically important but yet challenging problem. The interpretation of tumor and vessel geometry is important for surgery, diagnosis and monitoring of tumor response to therapy [3, 5]. 2D ultrasound tissue and Doppler imaging is currently used for examinations of the liver function. Very often, important clinical decisions are based on these diagnostic images. Today, invasive methods (such as angiography) are necessary to determine the relation between tumor and the neighboring vessels. It is believed that 3D ultrasound imaging may replace some of these

invasive methods and add more safety to the diagnostic process.

5.3 Fuzzy segmentation of tumors using ultrasonic images

In our first experiments with ultrasonic images, we wanted to see if the proposed algorithm was useful for fuzzy segmentation of various kinds of tumors. In one experiment, a 3D image from the liver of a patient with two tumors was obtained by using a Vingmed Sound CFM-750 ultrasound scanner with a probe mounted in a motor steered tilting device (see for instance [11]). A cross section of the liver is shown in figure 10 with white curves indicating the boundary of the tumors. The curves were manually drawn by an experienced physician. The fuzzy segmentation algorithm proposed in this paper was applied to the same image by selecting a seed region within each tumor and using an appropriate threshold value. A segmentation of the largest tumor is shown in figure 11. The f_1 weight function was used in this case, because of considerable variations in signal intensity within the tumor. The smaller tumor was only possible to isolate by using several seed regions. The proposed algorithm was also tested on other images with tumors. In some cases the algorithm worked satisfactory. In other cases the contrast between the tumor and the surrounding tissue was very low and part of the tumor boundaries were completely missing. In these cases the algorithm did not work satisfactory.

5.4 Visualization of liver veins from 3D ultrasonic images

3D in vivo acquisitions of liver from an experimental animal setup were conducted by connecting a Vingmed Sound CFM-750 with the Tomtec Echoscan system. After general anesthesia and intubation, the liver of the pig was exposed and the ultrasound 3D probe placed directly on the pig liver. Image acquisition and machine parameters were optimized to the intra-operative situation. Only the tissue echo data was acquired. There was no need for any triggering as respiration was turned off during the few seconds of acquisition time. The video grabbed data was transferred to a workstation. A cross section of the liver is shown in figure 5.

Our fuzzy segmentation algorithm was used to emphasize the vessels which were clearly visible from the 2D images. This gave a very precise definition of the vessels. A surface shading technique was applied to the segmented data. The resulting image is shown in figure 6. The image gives a detailed interpretation of the geometry of the liver veins. For comparison, we computed a segmented volume based only on binary thresholding of the raw images. Due to shadowing effects some parts of the liver tissue was segmented as blood. This was clearly visible in the rendered image 7.

5.5 Combined visualization of hemangioma and liver veins

In another experiment, the CFM-750 scanner was used to acquire digital 3D data from the liver of a 36 year old female patient with a benignant hemangioma in the right liver lobe. A cross section

through the liver is shown in figure 8. The fuzzy segmentation algorithm was used to emphasize two fuzzily defined segments, namely the hemangioma and the portal vein. Different colors were assigned to the different segments, and a transparent rendering model was used. A transparent rendering of the scene is shown in figure 9.

6 Conclusion

An algorithmic framework for fuzzy region oriented object segmentation has been developed. The framework is targeted at volume rendering methods where each voxel is given an opacity value which reflects the strength of connectivity between the voxel and a user defined set of seed voxels. The algorithm can also be used for interactive removal of voxels which occlude an interesting object and even for fuzzy volume estimation. A very efficient implementation of Dijkstras algorithm has been developed to compute the shortest path between each voxel and the seed region according to the selected weight function. Numerical experiments were conducted in order to evaluate the proposed framework. From these experiments we conclude that the f_∞ weight function should be used if the noise level is low. We have also found that the f_2 weight function is a better alternative if the noise level is higher. Due to the efficiency of the algorithm, the user can interactively isolate objects in 2D and even small 3D images.

The computational framework presented in this paper can be used to compute fuzzy connectedness as defined by Udupa and Samasekara [14]. Udupa and Samarasekara demonstrated their fuzzy segmentation framework on 2D and 3D MR-images and reported very satisfactory results for this application. In this paper we have focused on applications in ultrasonic imaging. Ultrasound images are generally suffering from a low signal to noise ratio compared to MR-images, and automatic segmentation is therefore in general quite difficult [11]. The proposed algorithm was applied to ultrasonic images containing tumors and vessels. The algorithm turned out to work satisfactory in some cases but in other cases where the contrast between the object of interest and surrounding tissue was too low, the algorithm did not give satisfactory results. A possible improvement of the algorithm would be to incorporate a measure of region shape into the edge weight function. This should be considered in further work.

7 Acknowledgments

The authors would like to thank the Norwegian Research Council for supporting this study. The authors would also like to thank Trygve Hausken at Haukeland regional hospital in Bergen, and Klaus Schipper at Trondheim regional hospital in Trondheim for providing the ultrasonic data used in the experiments.

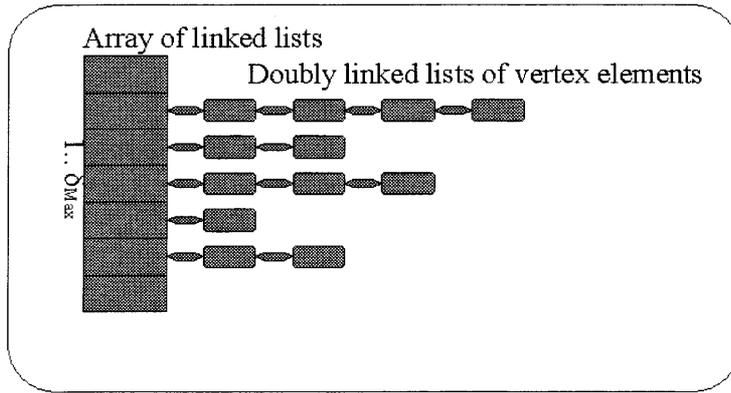


Figure 1: Priority queue implemented as a table of linked lists

Appendix A

To speed up the implementation of Dijkstras algorithm we impose the following restrictions on the weights and the minimum path length:

$$w(e(u, v)) \in 0..W_{Max}, \text{ for all } e(u, v) \in E$$

$$\delta(S, v) \in 0..\delta_{Max}, \text{ for all } v \in V$$

We have implemented a priority queue as an array $q[0..\delta_{Max} + 1]$ of linked lists of vertex elements. Each vertex element contains an index which identifies the vertex and a pointer to the next and previous element. The q array is indexed by path weights. An auxiliary variable d_{Min} keeps track of the currently minimum weight (initially 0). The q array is mainly used to speed up the *ExtractMin* operation.

In addition to the q array, we use an additional array $v_p[1.. |V|]$ of pointers which give direct access to all vertex elements in the queue. This array is mainly used to speed up the *relax* operations.

Initially, all vertices in S are placed in the queue as a linked list in $q[0]$ (zero weight) and all vertices in $V - S$ are placed in a linked list in $q[\delta_{Max} + 1]$ (infinite weight). The data structure is illustrated in figure 1.

The *ExtractMin* operation can be implemented as follows on the suggested data structure:

```
function ExtractMin( $q$ )
(1) if  $d_{Min} > \delta_{Max}$  return (NULL)
```

```

(2)  $u = \text{ExtractFirstNodeInList}(q[d_{Min}])$ 
(3) while  $q[d_{Min}] = 0$  and  $d_{Min} \leq \delta_{Max}$  do
(4)    $inc(d_{Min})$ 
(5)  $dec(qSize)$ 
(6)  $v_p[u] = \text{REMOVED}$ 
(7) return ( $u$ )

```

The function *ExtractFirstNodeInList* extracts the first node in the linked list pointed to by $q[d_{Min}]$. The loop in lines (3-4) finds the next non-empty entry in q and updates d_{Min} . The size of the queue $qSize$ is then decreased and the pointer $v_p[u]$ (which is used by *Relax* marks the node u as removed.

All lines in the code above execute in $O(1)$ time except for line (3), which in the worst case will execute in $O(\delta_{Max})$ time. However, if we amortize the time spent in line (3) in all the necessary *extractMin* operations, we see that the total time spent inside (3) will not exceed $O(\delta_{Max})$. (If $d_{Min} > \delta_{Max}$, the queue is empty and no further action is taken.) There are at most $|V|$ *ExtractMin* operations and the total time spent in this function is then $O(V + \delta_{Max})$.

The *Relax* operation involves several sub-operations. *Relax* checks if the weight of an alternative path through u to v is cheaper than the current minimum weight path estimate for v . If so, the function *DecreaseKey* is called. *DecreaseKey* updates the priority queue with the new estimate for v . *DecreaseKey* is simply a deletion of v followed by an insertion of v with the new weight.

```

function Relax( $q, u, v, w$ )
(1) if ( $v_p[v] == \text{REMOVED}$ ) return "Node does not exist anymore"
(2) if ( $weight(q, u) + w > weight(q, v)$ )
(3)   DecreaseKey( $q, v, weight(u) + w$ )

```

```

function DecreaseKey( $q, u, v, w$ )
(1) Delete( $q, v$ )
(2) Insert( $q, v, weight(u) + w$ ) "insert v with new weight"

```

```

function Delete( $q, v$ )
(1) RemoveNode( $q[weight(v)], v_p[v]$ )
(2)  $v_p[v] = \text{REMOVED}$ 
(3)  $dec(qSize)$ 

```

$q[weight(v)]$ identifies the linked list which holds v . $v_p[v]$ identifies directly the node within the list. The *RemoveNode* operation simply removes the specified node from the list and executes in $O(1)$ time. The whole *Delete* operation, can therefore be executed in $O(1)$ time.

function $Insert(q, v, weight)$

- (1) if $(weight < d_{min})d_{min} = weight$ “update d_{min} if necessary”
- (2) $v_p[v] = InsertNode(q[weight], v)$
- (3) $inc(qSize)$

The $InsertNode$ operation simply inserts a new node in the front of the linked list $q[weight(v)]$ or creates a new list, if $q[weight(v)]$ is empty. $v_p[v]$ is updated with a pointer to the new node. The $Insert$ operation can therefore execute in $O(1)$ time. The whole $Relax$ operation involves several sub-operations which all execute in $O(1)$ time, and $Relax$ is therefore $O(1)$. There are at most E $Relax$ operations and the total running time of Dijkstras algorithm is then $O(V + \delta_{Max} + E)$. If the topology of the graph is a regular 3D array, $E = 6 \cdot V$ and the algorithm will run in $O(6 \cdot V + \delta_{Max})$ time.

In the worst case (in terms of computational complexity), δ_{Max} is set so that all vertices in the graph can be reached by a path with a weight $w(P) \leq \delta_{Max}$. It is interesting to find the running time of the proposed algorithm in this worst case scenario.

If the weight of a path is given as in equation 10 ($\gamma = \infty$), we will have $\delta_{Max} = W_{Max}$ and δ_{Max} is therefore a (small) constant which is independent of the size and topology of the graph. The worst case running time of Dijkstras algorithm will in this case be $O(6 \cdot V)$ which is a major improvement compared to the general $O(E + V \log V)$ implementation, considering that the number of vertices in a 3D image may become quite large.

If the weight of a path is given as in equation 10, δ_{Max} can not be assumed to be independent of the graph size. In the worst case, the number of connections between two vertices can be proportional to the number of vertices in the graph. (For instance if the topology of the graph is a 1D array.) If the weight of a path is defined as the sum of all weights along the path, an upper limit for δ_{Max} is $V \cdot W_{Max}$

However, for most practical purposes, δ_{Max} can be regarded as a constant compared to the number of vertices V in the graph. We will justify this by an example:

Consider that the original image is a regular 2D array of size $V^{\frac{1}{2}} \cdot V^{\frac{1}{2}}$, where all vertices are connected with their 4 nearest neighbors. We want to find an upper limit for δ_{Max} as a function of V . Such a limit can be found as the weight of the most expensive minimum weight path that can exist between any two vertices in V . In the “worst” case, $w(e(u, v)) = W_{Max}$, for *all* $e(u, v) \in V$. To realize why this is a worst case situation, consider that the weights of some edges in E are decreased below W_{Max} . The weight of the most expensive minimum weight path will then either decrease or remain the same, since the weight of any path will decrease if any of the edge weights along it is decreased. If all weights are equal to W_{Max} , the longest possible minimum weight path will also be the most expensive one. The longest possible minimum weight path in a regular 2D array will be between two corner vertices. The longest possible minimum weight path between two corner vertices will therefore consist of $2 \cdot V^{\frac{1}{2}} - 1$ edges. An upper limit for δ_{Max} can then be found: $\delta_{Max} < ((2 \cdot V^{\frac{1}{2}} - 1) \cdot W_{Max}^{\gamma})^{\frac{1}{\gamma}} < 2 \cdot V^{\frac{1}{2}} \cdot W_{Max}$. Since W_{Max} can be regarded as a (small)

constant independent of V , $O(\delta_{Max} + 4 \cdot V) = O(2 \cdot V^{\frac{1}{2}} \cdot W_{Max} + 4 \cdot V) = O(4 \cdot V)$

Appendix B

Udupa and Samarasekara have recently proposed a general framework for fuzzy image segmentation based on the theory of fuzzy sets [14]. In this appendix we will review some of the theory presented in [14] and relate this to the work presented in this paper.

A fuzzy subset A of a set X is a set of ordered pairs

$$A = \{(x, \mu_A(x)) \mid x \in X\} \quad (11)$$

where

$$\mu_A : X \rightarrow [0, 1] \quad (12)$$

is a membership function of A in X . A fuzzy relation ρ in X is a fuzzy subset of $X \times X$

$$\rho = \{((x, y), \mu_\rho(x, y)) \mid (x, y) \in X \times X\} \quad (13)$$

where $\mu_\rho : X \times X \rightarrow [0, 1]$ ρ is called a *similitude relation* if it is reflexive, symmetric and transitive.

Udupa and Samarasekara divides an n -dimensional image into *spels* with coordinates defining points in Z^n . A reflexive and symmetric fuzzy relation in Z^n α was called a *fuzzy spel adjacency*. The pair (Z^n, α) was then called a *fuzzy digital space*.

Given a pair $C = (C, \{ \})$ where $C = \{c \mid -b \leq c \leq b \text{ for some } b \in Z_+^n\}$ and f is a function whose domain is C . C is called *scene* over a digital space (Z^n, α) . C is a *membership scene* over (Z^n, α) if the range of f is a subset of $[0, 1]$. A fuzzy relation $\kappa \in C$ is said to be a *fuzzy spel affinity* in C if it is reflexive and symmetric.

Fuzzy κ -connectedness is a relation in C :

$$\mu_K(c, d) = \max_{p \in P_{cd}} [\mu_N(p)] \quad (14)$$

where c and d are spels in C , P_{cd} is the set of possible paths between c and d , p is one such path, and $\mu_N(p)$ is a membership function assigning the minimum of the pairwise spel affinity of spels in p . This means that the strength of a connection between two spels c and d is determined by the weakest link on the strongest path between c and d . It is important to note that a path in this context is an arbitrary sequence of spels within C .

A membership scene can easily be translated into an undirected graph $G(V, E)$ with weighted edges using the following rules:

Rules for translating membership scenes to graphs

- Map each spel $c \in C$ into a corresponding vertex $v_c \in V$
- Map *each* pair of spels $c \in C, d \in C$ into an edge $e(v_c, v_d) \in E = V \times V$
- Assign the weight of each edge $e(v_c, v_d) \in E$ as given by $\delta_{Max} - \delta_{Max} \cdot \mu_\kappa(c, d)$, where κ is a fuzzy spel affinity

Note that the graph has to be undirected, since κ is a symmetric relation.

The problem of computing $\mu_\kappa(c, d)$ for all $c, d \in C$ can now be directly translated into the problem of computing the minimum weight path $\delta(v_c, v_d)$ for all $v_c \in V, v_d \in V$, where $\delta(v_c, v_d)$ is defined as in equation 10. We can then apply our implementation of Dijkstras algorithm to this problem.

The following rule can be used to map the minimum weights $\delta(v_c, v_d)$ into values $\mu_\kappa(c, d)$:

$$\mu_\kappa(c, d) = (\delta_{Max} - \delta(v_c, v_d)) / \delta_{max}$$

References

- [1] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. Mc Graw-Hill, 1990.
- [2] R. A. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. *Computer Graphics*, 22(4):65–74, August 1988.
- [3] A. Weimann et. al. Wertigkeit der sonographie bei der diagnostik und verlaufskontrolle von hemangiom und fokal nodularer hyperplasie der leber. *Ultraschall Klin Prax*, 9:71–74, 1994.
- [4] Ahuja et al. Faster algorithms for the shortest path problem. *Technical report 193, MIT Operations Research Center*, 1988.
- [5] P. D. LeRoux et. al. A comparison between preoperative magnetic resonance and intraoperative ultrasound tumor volumes and margins. *J. Clin. Ultrasound*, 22:29–36, January 1994.
- [6] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison Wesley, 1992.
- [7] M. Levoy. Volume rendering: Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, pages 29–37, May 1988.
- [8] Hans-Peter Meinzer, Kirsten Meetz, et al. The heidelberg raytracing model. *IEEE Computer Graphics and Applications*, pages 34–43, November 1991.
- [9] Bjørn Olstad. Maximizing image variance in rendering of volumetric data sets. *Journal of Electronic Imaging*, 1:245–265, July 1992.

- [10] P. Sabella. A rendering algorithm for visualizing 3D scalar fields. *Computer Graphics*, 22(4):51–55, August 1988.
- [11] E. Steen and B. Olstad. Volume rendering of 3d medical ultrasound data using direct feature mapping. *IEEE Trans. on Medical Imaging*, september 1994.
- [12] E. Steen, B. Olstad, G. Myklubust, S. Berg, and K. P Schipper. Combined visualization of tumor and vessel geometry. In *IEEE International Ultrasonics Symposium Cannes*. IEEE, 1994.
- [13] J. K. Udupa and Dewey Odhner. Shell rendering. *IEEE CG& A.*, pages 58–67, November 1993.
- [14] J. K. Udupa and S. Samarasekara. Fuzzy connectedness and object definition. In *SPIE Medical Imaging 95, Image Display conference*, volume 2431, pages 2–10. SPIE, 1995.
- [15] Craig Upson and Michael Keeler. V-buffer: Visible volume rendering. *Computer Graphics*, 22(4):59–64, August 1988.

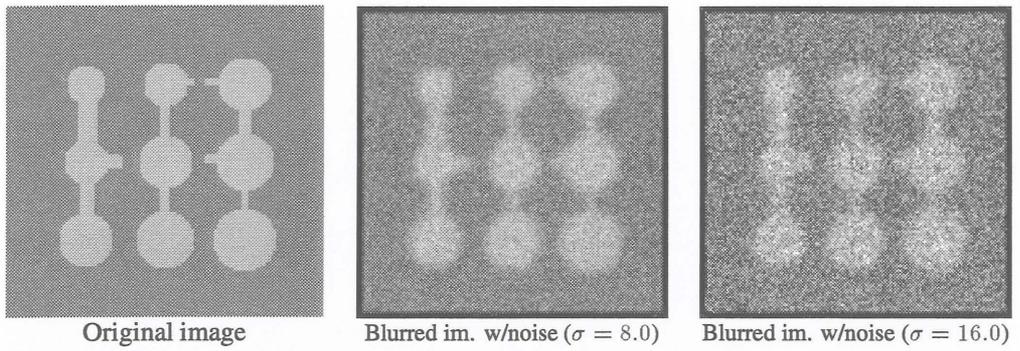


Figure 2: Synthetic images

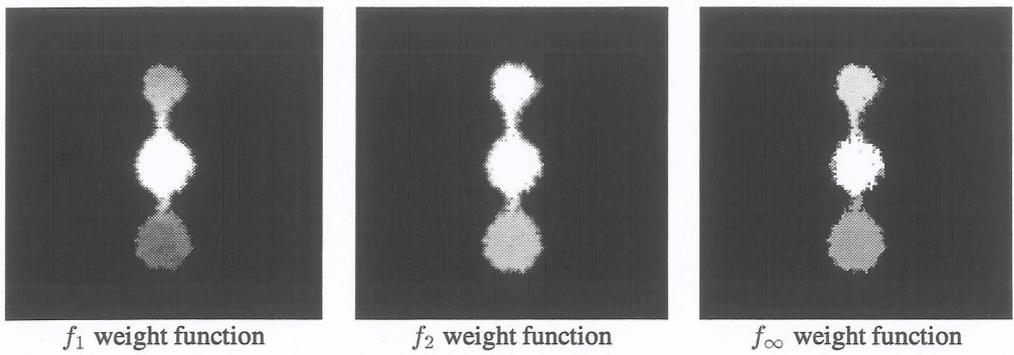


Figure 3: Results of fuzzy segmentation, noise level $\sigma = 8.0$

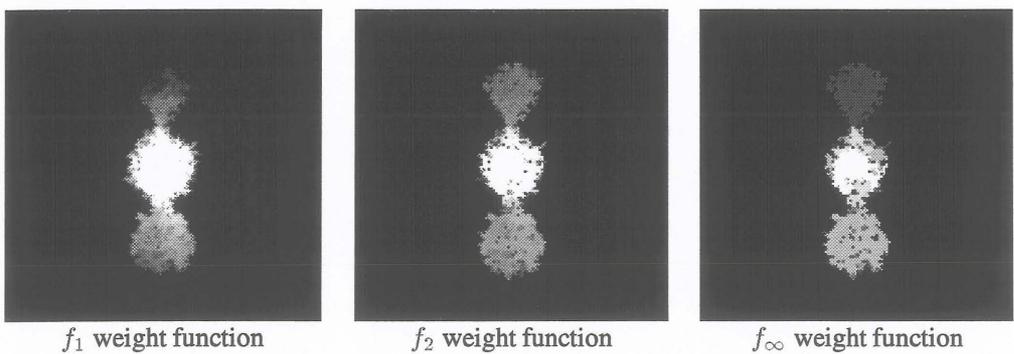


Figure 4: Results of fuzzy segmentation, noise level $\sigma = 16.0$



Figure 5: Cross section showing liver veins

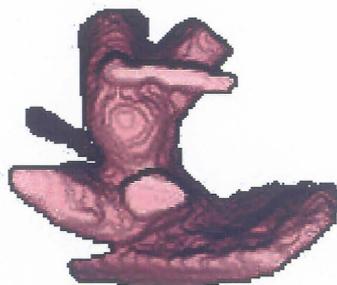


Figure 6: Visualization of liver veins



Figure 7: Visualization of liver veins, opacity computed only from intensity values

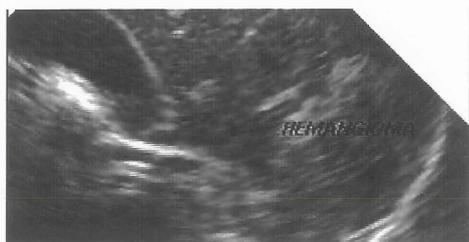


Figure 8: Cross section through human liver



Figure 9: Vis. of hemangioma and portal vein



Figure 10: Ultrasound image of tumor

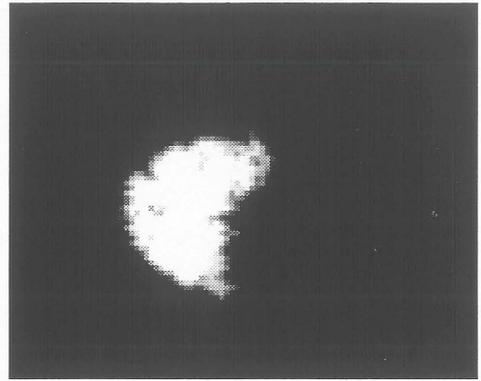


Figure 11: Segmented tumor

