ELSEVIER

Computerized Medical Imaging and Graphics 29 (2005) 53–63

Computerized Medical Imaging and Graphics

www.elsevier.com/locate/compmedimag

Real time computation and temporal coherence of opacity transfer functions for direct volume rendering of ultrasound data

Bernhard Petersch[a,*], Markus Hadwiger[b], Helwig Hauser[b], Dieter Hönigmann[a]

[a]Advanced Computer Vision GmbH (ACV), Tech Gate Vienna, Donau-City-Strasse 1, A-1220 Vienna, Austria
[b]VRVis Research Center, TechGate Vienna, Donau-City-Strasse 1, A-1220 Wien, Austria

Received 27 May 2004; accepted 22 September 2004

Abstract

Opacity transfer function (OTF) generation for direct volume rendering of medical image data is an intensely discussed subject. Several automatic methods exist for CT and MRI data, which are not apt for ultrasound data, mainly due to its low signal-to-noise ratio. Furthermore, ultrasound (US) imaging is able to produce time-varying 3D datasets in real time thus opening the door to 4D visualization. However, OTF design for 4D datasets has not been exhaustively discussed until now. We present an efficient solution to generate an optimized OTF for a given 3DUS dataset in real time. Our method results in excellent visualization which we demonstrate using 3D fetus datasets. Finally, we discuss the applicability of our method to 4DUS visualization.

Keywords: 3D/4D ultrasound; Direct volume rendering; Transfer function

1. Introduction

While 2D ultrasound is an established medical imaging modality since long ago, 3D ultrasound imaging (3DUS) has continually gained importance in many medical fields during the last few years [1–5]. Ultrasound imaging has many advantages in comparison to computerized tomography (CT), magnetic resonance imaging (MRI). The absence of ionizing radiation renders the installation of special treatment rooms unnecessary and provides diagnostic possibilities wherever more invasive techniques are prohibitive, like for example in fetal imaging.

Not surprisingly, despite the rather noisy nature of the images, US imaging is the method of choice for many diagnostic tasks nowadays. With the ever increasing computational power of available computer hardware and advances in algorithms for visualization of three-dimensional digital data, 3DUS has found its way into everyday clinical life, e.g. in *radiotherapy*, 3DUS is used for applications as diverse as prostate segmentation for radiotherapy treatment planning [6–8], patient positioning for prostate treatment [9], or breast biopsy and monitoring of carotid atherosclerosis in response to therapy [10]. Within the field of *ophthalmology*, diagnosis and quantitative analysis of various ocular diseases like choroidal melanoma or retinoblastoma can be assisted by 3DUS [11,12]. *Cardiology* is a classic US domain and the extension from 2D to 3D scanning techniques promises for example better pre- and post-surgical planning, improved measurement of heart functions, decreased exam times, better quantification of size, shape and function of the heart, improved localization of abnormalities for surgical planning [13], facilitated analysis of septal defects [14], or more accurate quantification of cardiac chamber volume, mass and ventricular functions [15]. *Intravascular US* (IVUS) likewise profits from 3D scanning techniques, providing better quantitative analysis of degree and extent of coronary/artery plaque [16–19]. In *obstetrics* and *gynecology*, 3DUS has also been found to be a valuable and powerful diagnostic tool, fetal imaging being one of the most popular 3DUS applications [20,21]. Nelson and Pretorius [5] provide an overview over this exciting domain.

* Corresponding author. Tel.: +43 1 2696255 130; fax: +43 1 2696255 200.

E-mail address: bernhard.petersch@acv.ac.at (B. Petersch).

0895-6111/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compmedimag.2004.09.013

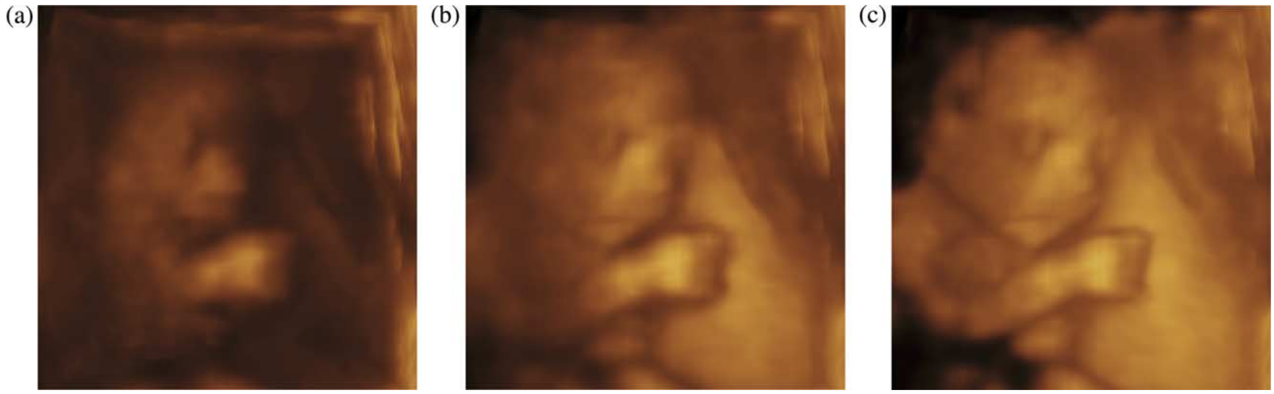

Fig. 1. Direct volume rendering of an ultrasound data set of a fetus using three different global opacity transfer functions (OTF). (a) Linear OTF (b) Manually designed piecewise linear. OTF (c) Adaptively designed OTF.

However, while the visualization of 2D medical data is rather trivial, visualization of 3D data is not. Basically, there are two approaches for visualization of these data sets, namely those based on *surface extraction* and *direct volume rendering*. The latter is a method which is readily available for visualization of volumetric data without the need to compute an explicit surface model prior to visualization. For this reason, direct volume rendering is particularly popular. This is especially true for modalities where the segmentation task is difficult to handle. 3DUS imaging is one of these modalities. The following US image characteristics make many conventional segmentation and visualization techniques fail:

- The coherent nature of the ultrasound imaging pulse is responsible for interference effects and the appearance of *speckle artifacts* which often exceed the specular echo intensity.
- The dynamic range is much lower than in CT or MRI.
- Variations in the intensity of neighboring voxels are high, even within areas of homogeneous tissue. Rather, signal intensity locally increases at the *interface* between adjacent tissues.
- Boundaries show varying gray level caused by the variation of surface curvature and orientation to the sound source. The regions representing boundaries are not sharp but show a width of several pixels.
- Surfaces are partially or completely shadowed from objects closer to and within the direction of the sound source.
- Sonography is a highly interactive modality: the physician moves the sound source and expects to immediately see the image from the new point-of-view. Thus, all visualization techniques need to work in or near real time.

It is important to understand that in the direct volume rendering approach the delineation of the surface by some kind of 3D segmentation of the original data set is replaced by applying an *opacity transfer function* (OTF). Proper design of this function, which maps voxel properties to opacities, is of great importance and determines the final result of the visualization process, as one can see in Fig. 1.

Sakas et al. [20,21] were among the first to point out that MRI and CT surface reconstruction techniques do not yield reasonable results if applied to ultrasound data. Instead, they suggest a multi-scale *binarize*, *low-pass*, *threshold and propagate* (BLTP) method to pre-process the volumetric data. Subsequently, they employ direct volume rendering for surface visualization, using a slightly modified version of Levoy's [22] standard volume rendering pipeline. Their approach yields pleasing results in case of data sets with high contrast between the structures of interest and the background. In the conclusions of [20], they state that a surface extraction method which adapts to the local characteristics could improve the appearance of the rendered volume. We propose the adaptation of the OTF using surface information.

This paper is organized as follows. We briefly summarize previous work on OTF design in Section 2 and discuss the most commonly used OTF in 3D sonography, a piecewise linear OTF, in Section 3. In Section 4, we introduce our methodology to adjust OTFs to a specific data set in an automated way, presenting an alternative family of parabolic OTFs which outperform linear OTFs in matters of image contrast. Section 5 deals with the subject of 'temporal coherence', e.g. the question whether a single initial OTF or view-dependent OTFs are superior in visualizing *time-varying* datasets. The paper ends with results achieved by our approach in Section 6 and conclusions in Section 7.

2. Previous work

In the past, different OTF design approaches have been suggested. Basically, there are two classes, namely those which require user interaction, and those which work fully automatic. Among the latter is a stochastic optimization technique based on objective measures of the rendered image suggested by He et al. [23]. Unfortunately, their evolutionary approach is computationally demanding, and the definition of objective measures for rendered ultrasound data is not easy.

A different approach based on *histogram volumes* has been proposed by Kindlmann et al. in [24]. This approach unfortunately requires that the regions of interest are boundaries between different materials modeled by a step edge. This is a reasonable assumption for anatomical structures in CT data sets, but not for ultrasound data. An extension of this method toward multi-dimensional transfer functions by Kniss et al. [25] aims at the visualization of multi-variate volume data. An OTF design technique using 3D filter responses proposed by Sato et al. [26] uses even more detailed models of local structures such as sheets, lines, and blobs, and is thus not applicable to ultrasound data, either.

A 3D field topology approach by Fujishiro et al. [27] as well as the evaluation of the *contour spectrum* by Bajaj et al. [28] are based on the evaluation of isosurfaces. As discussed above, in sonography tissue boundaries often show varying gray levels caused by the variation of surface curvature and orientation to the sound source, or shadowing. The structure of interest will thus most probably not match with an isosurface.

Rezk-Salama et al. [29] suggested to use manually designed 'optimal' transfer functions as reference templates. By *non-linear time warping* they compute a non-linear distortion of the data value axis to obtain alignment of the normalized histograms of both the reference data set and the data set currently under examination. Unfortunately, for sonographic data alignment of the histograms is not a sufficient criterion to obtain a pleasing visualization.

Hence, although all these automated transfer function design techniques have proven to be useful in many visualization tasks the unique characteristics of sonographic data prevents their application.

3. Piecewise linear opacity transfer functions

The most commonly used transfer function for rendering of volumetric ultrasound data is based on Marc Levoy's seminal paper on direct volume rendering [22]. He suggested an opacity transfer function considering the intensity $I(x_i)$ of a voxel x_i as well as its gradient $\nabla I(x_i)$. Because of the low signal-to-noise ratio of sonographic data and the high sensitivity of the gradient to noise this transfer function is hardly ever used for ultrasound data visualization directly. Rather, it is common practice to either pre-process the data as proposed by Sakas in [20,21], or to base the transfer function solely on image intensity. In the latter case, a commonly used OTF is a simple piecewise linear monotonically increasing function

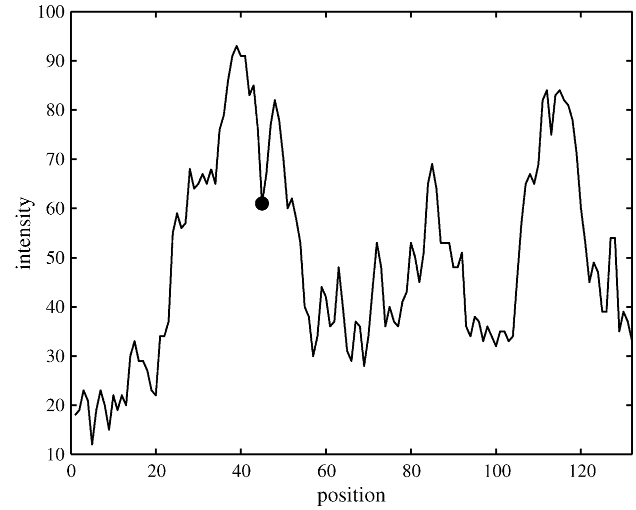

Fig. 2. Signal intensity profile of the ultrasound data set of Fig. 1. The position of an interface between tissue of different echogenicity is indicated by a dot. For the position of the profile see Fig. 3.

$$\alpha(I) = \begin{cases} 0 & \text{if } I < L_L, \\ a(I - L_L) & \text{if } L_L \le I \le L_H, \\ a(L_H - L_L) & \text{if } I > L_H. \end{cases} \tag{1}$$

This OTF resembles a 'fuzzy' segmentation of the entire data volume by an intensity threshold: Voxels with intensities I below L_L are classified as 'invisible', i.e. their opacity α is set to zero. Intensities between L_L and L_H have increasing opacities while intensities above L_H yield maximum opacity. An OTF of this form is useful for visualization of anatomical structures embedded in hypoechoic areas, i.e. areas with low signal intensity. A typical example is the data set of a fetus floating in amniotic fluid. For the result one obtains for $L_L = 20$, $a = 0.5$, and $L_H = 255$ see Fig. 1(b). The shape of the OTF is motivated by an increase in signal intensities near tissue boundaries, such as the interface between amniotic fluid and fetal tissue (see Figs. 2 and 3).

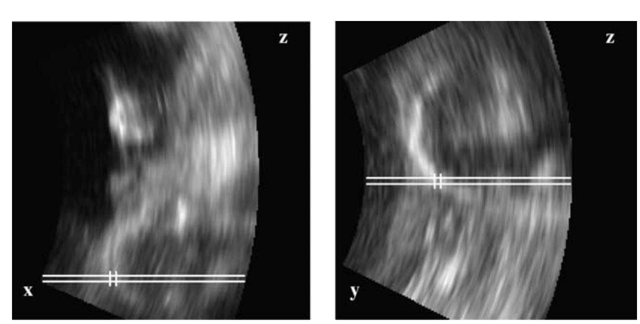

Fig. 3. Two orthogonal cross-sections of the ultrasound data set of Fig. 1. The position of the intensity profile of Fig. 2 is between the white lines. The position of the interface between amniotic fluid and fetal head detected by analysis of a tube core is indicated by short white lines.

Despite the simplicity of (1) three parameters need to be determined. Usually, this adjustment has to be done manually on a trial-and-error basis not only for every single data set but also depending on the view. It is also not clear whether a piecewise linear OTF allows to obtain an optimal visualization for a given data set. Thus, before going for automated OTF design, it was imperative to investigate in which way the choice of a specific initial OTF influences the contrast in rendered images.

4. Adaptive OTF design

One of the most important applications of volume rendering of 3DUS data is the visualization of a fetus embedded in amniotic fluid. We therefore based our theoretical considerations about contrast enhancement on a simple two-intensity data model mimicking this scenario: tissue of high signal intensity c_1 is located behind some lower intensity area with signal intensity c_2. In order to produce a 2D projection of voxel values in the viewing plane we use the *volumetric compositing stage* proposed by Levoy in [22]: rays are cast from the eye into the voxel arrays and color and opacity information are combined into single values to provide a final pixel intensity. For our data model, we assume that, in viewing direction, the rays first pass through voxels with intensity c_2, and that the intensity changes to c_1 at a distinct position k^*. For a single voxel at position k along the ray

$$C_{\text{out}} = C_{\text{in}}(1 - \alpha(k)) + c(k)\alpha(k), \tag{2}$$

with C_{out} the outgoing intensity and color for the voxel, C_{in} the incoming intensity, $\alpha(k)$ the opacity assigned to the voxel at position k along the ray and $c(k)$ its color or intensity. The final intensity $C(r)$ due to the set of K voxels that intercept the ray r is given by

$$C(r) = \sum_{k=0}^{K} \left(c(r,k)\alpha(r,k) \prod_{i=k+1}^{K} (1 - \alpha(r,i)) \right), \tag{3}$$

where (r, k) is the kth voxel along the rth ray, and $c(r, 0)$ is the color of an opaque background, i.e. $\alpha(r, 0) = 1$.

Based on this rendering scheme and our data model, we compared the performance of linear, parabolic ($\alpha(I) = a \times I^2$, $\forall I$) and 4th order parabolic OTFs ($\alpha(I) = a \times I^4$, $\forall I$). We found that, from a theoretical point-of-view, parabolic OTFs outperform linear OTFs concerning image contrast. The complete analysis can be found in [30].

We now provide means to adjust an OTF to a specific data set. We aim at an automated adjustment to allow the online-computation of an optimal OTF for acquired volume data sets at high frame rates. Unfortunately, we lack an objective function which would allow to tune the OTF by an optimization approach. Nevertheless, we aim at an automated adaptation of OTFs. To achieve this goal we take an heuristic approach as follows. We detect the interface between tissue H with high echogenicity and low echogenic area L. Subsequently we use information about H and L to modify an initial OTF. The task of choosing an appropriate surface is not easy. There are the following difficulties. Because of the noisy characteristics of ultrasound data, it might be difficult to identify the interface between H and L (see Fig. 2). There might be multiple interfaces of equal magnitude, but at different intensities. If we have identified the interface between H and L, most probably it will be some kind of noisy ramp rather than one distinct step edge.

4.1. Surface detection

The proposed methodology for estimating the tissue surface is based on the evaluation of so called *tube cores*. Having defined the z-axis parallel to the ray-casting direction, a tube core is a collection of voxels gathered by traversing the volume in z-direction with a specific diameter. Considering the discrete framework, we specify the diameter by its width in x- and y-direction in voxels. A tube core resembles an intensity profile as in Fig. 2, but has an extension greater than one in x- and y-direction. Using a tube core width greater than one is motivated by the noise inherent in ultrasound data. We place these tube cores in an equally spaced grid. In order to further reduce the effects of low signal-to-noise ratio we evaluate the signal intensities along a tube core t within cells C_i^t of specific depth. The size of these cells is thus defined by the diameter of the tube core and the cell depth. We may then assess statistical parameters of the signal intensities within the cells, such as the mean signal intensity $I(C_i^t)$. Our goal is the detection of the position of a surface within the tube core. The measure we use for this task is the *variance of the intensities* within the cells. Thus, we use this parameter as *interface indicating function*. We will expunge the only disadvantage of this parameter, the missing information about edge direction, by subsequent processing.

Choosing the interface indicating function does not provide the final solution of our problem, the detection of the interface we want to visualize. A closer look at the graph of the interface indicating function in Fig. 4 unveils that it is not sufficient to look for the global maximum. Rather, we are looking for the first significant local maximum, in viewing direction.

We adopted the *scale space filtering* approach, originally proposed by Witkin [31], to this problem. This method describes signals in terms of their extrema, managing ambiguity of scale in an organized and natural way. The signal is expanded by convolution with Gaussian kernels of decreasing size over a continuum scales. We track interfaces, i.e. maxima of the interface indicating function, in the vicinity of the result of the interface detection in

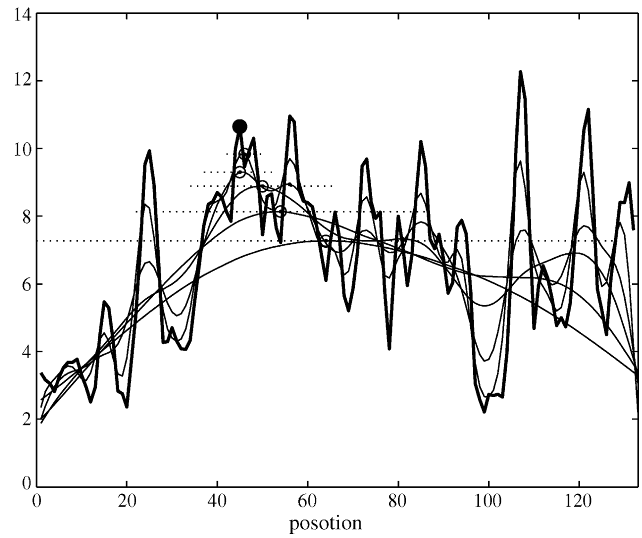

Fig. 4. The interface indicating function and the position of the interface between amniotic fluid and fetal head in Figs. 2 and 3 are depicted by the thick line and the dot. Multi-scale detection starts with the detection of the first local maximum at a coarse scale of the interface indicating function. We then compute the range (dotted lines) for subsequent maximum detection at finer scales.

the previous scale. Within this range we choose the first local maximum in viewing direction. We iterate this procedure until we finally locate an interface in the original scale. The multi-scale detection of the interface in Fig. 3 is depicted in Fig. 4. The suggested method does not necessarily detect the most prominent nor the very first interface in the tube core.

4.2. Transfer function modification

The aim of modification of the OTF is to reduce opacity for intensities which typically appear prior to the detected surface while enhancing opacity of the intensities located at the surface. Our approach is a multiplicative one. We start with an OTF of parabolic shape and modify the initial OTF by one basis function per tube core. Let x denote the intensity and T the number of tube cores used, then the OTF $\alpha(x)$ reads

$$\alpha(x) = \prod_{t=1}^{T} f_t(x) \times x^2 \tag{4}$$

We will now elaborate on the calculation of $f_t(x)$. First of all we extract two parameters for every tube core t. The mean signal intensity I_L^t of all cells prior to the position of detected surface s_t, and the mean signal intensity I_H^t of the cell located exactly at the detected surface. Then, we design each individual basis function $g_t(x)$ to have a minimum at I_L^t and a maximum at I_H^t. One family of functions suitable for this task are functions of the form

$$g_t(x) = ax\,\mathrm{e}^{-b(x-c)^2}, \tag{5}$$

which we scale to the range of values $[-1;\ 1]$ with a maximum at I_H^t and a minimum at I_L^t by choosing

$$a = \sqrt{2be}, \quad b = \frac{2}{(I_H^t - I_L^t)^2}, \quad c = \frac{1}{\sqrt{2b}} + I_L^t. \tag{6}$$

We now have to transform the basis function to an appropriate range for multiplication, i.e.

$$f_t(x) = (1 + d_t)^{g_t(x)}. \tag{7}$$

Since the range of $g_t(x)$ is $[-1;\ 1]$, $f_t(x)$ ranges from $1/(1+d_t)$ to $(1+d_t)$, thus ‘raising’ or ‘lowering’ the parabolic basis function at the corresponding intensity x (see Eq. (4)). The parameter d_t (>0) determines the weight of the tube core t. Its calculation is based on the ‘quality’ of the tube core, that is the reliability of the surface extraction within this tube core. The proposed method for detecting tissue interfaces works very well even for data with low signal-to-noise ratio, but it does not detect the correct interface in every single case. On the other hand, it might even happen that there is no interface at all to detect, since the position of the tube core is outside the anatomical structures we want to visualize. Therefore, we have to provide means to detect these cases and eliminate their influence on the final OTF. This is done by assigning *higher values of* d_t to tube cores with *highly reliable surface detection* and vice versa.

There are two constraints at hand for determining the reliability of surface detection. First, the maximum intensity before the surface must not exceed a certain value. Otherwise, we either detected too deep an interface, or the entire tube core is located within hyperechoic tissue. Hence, we compute a quality measure $w^{\mathrm{t}}\hat{\imath}$ for each tube core t based on the evaluation of the mean intensities $I(C_i^t)$ of all cells C_i^t prior to the detected surface s_t. Let $\hat{I}_L^t$ be the maximum of these intensity values, i.e.

$$\hat{I}_L^t = \max_{i<s_t} I(C_i^t). \tag{8}$$

We then compute the histogram $h\hat{I}$ of the maximum intensities $\hat{I}_L^t$ of all tube cores t. By accumulation and normalization of the histogram we finally obtain the quality factor $w^{\mathrm{t}}\hat{\imath}$ for a specific tube core t with maximum intensity $\hat{I}_L^t$

$$w\hat{I}^t = 1 - \frac{1}{N} \sum_{I=0}^{\hat{I}_L^t} h\hat{I}(I), \tag{9}$$

with N the total number of tube cores.

We obtain a second quality measure w_S^t in a similar way based on the histogram h_S of intensity slopes S^t at the detected surface position s_t of every tube core t. Again, accumulation and normalization of the histogram

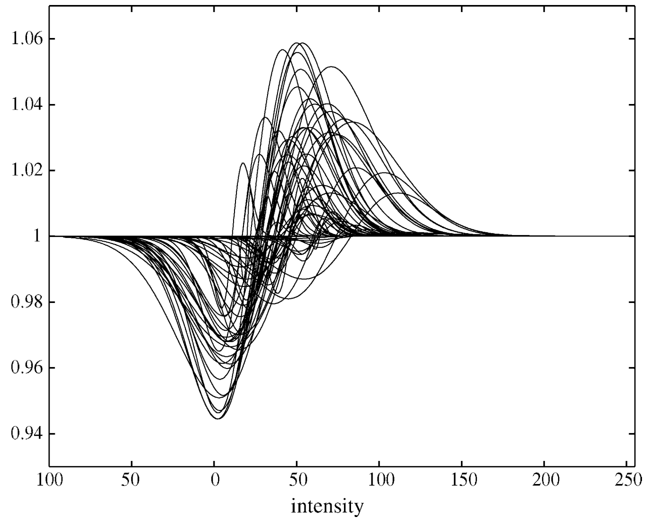

Fig. 5. Graph of all basis functions f_t used to adapt the initial OTF to the data set of Fig. 1 in a multiplicative way. Each basis function has a minimum at the mean intensity I_L^t prior to the detected surface and a maximum at the mean intensity I_H^t of the cell which is located at the surface detected for each tube core t. The range of f_t depends on tube core quality parameters w_S^t and $w^t_{\hat{I}}$.

allows us to assess a quality factor w_S^t for a specific tube core t with slope S^t

$$w_S^t = \begin{cases} \dfrac{1}{N}\displaystyle\sum_{S=0}^{S^t} h_S(S) & \text{if } S > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

$w^t_{\hat{I}}$ and w_S^t allow the elimination of any influence of tube cores located within tissue of constant echogenicity, but also a reduction of the influence of surfaces with weak manifestation. In addition, we also exclude all tube cores with negative slope of intensity at the surface position, thereby eliminating the drawback of the chosen interface indicating function. We use the quality measures $w^t_{\hat{I}}$ and w_S^t to modulate the parameter d_t in (7): We choose

$$d_t = d_{\max} \times w_{\hat{I}}^t \times w_S^t. \tag{11}$$

The graph of all basis functions for the volume rendered in Fig. 1 is depicted in Fig. 5.

5. Temporal coherence

Ultrasound imaging is a priori a real time imaging technique. In 2DUS imaging, physicians are used to having the acquired images displayed instantaneously, and to see the image change corresponding to the movement of the transducer if a hand held device is being used. Not surprisingly, the same is expected from 3DUS and modern 3DUS scanners are capable of providing real time 4D visualization. Three different 4D scenarios can occur:

1. static transducer—moving object
2. static object—moving transducer
3. a combination of 1 and 2.

If an OTF which is dependent on the view of the scanned anatomy—like in our case—is used for direct volume rendering of such 4D datasets, the following question arises: is it sufficient to calculate an OTF based on the initial view and render all subsequent 3D views using this same initial OTF? Transfer functions are usually designed or generated globally without taking the actual view point into account. Even in 4D data sets involving multiple volumes, a transfer function is usually generated for a single frame and then used identically for all frames in the sequence. The alternative would be to calculate a new OTF every time the viewing angle changes or the object moves. The latter approach would be feasible since our method is able to provide OTFs in real time (see Section 6.1). The advantage of a view-dependent OTF would obviously be that the rendering result is optimized for the *given* viewing direction. Theoretically, this could improve the contrast of details in comparison to a rendering using an OTF generated from a *different* viewing angle. On the other hand, the OTF could show large variations with viewing direction. This may cause incoherence in intensity and contrast of the rendered images of a time-varying sequence of volumes.

To the best of our knowledge, this matter has not been exhaustively discussed in the literature until now. Previous work on multiple transfer functions for time-varying data exclusively considers the view-independent case where the data set itself is changing over time [32].

However, if the transfer function generation itself is view-dependent, rotations of the view lead to either the same problem of coherence between multiple frames using different transfer functions, or a single global transfer function that is not optimal for all frames.

A general problem with time-dependent transfer functions is that what is actually visualized changes over time. These automatic changes are generally not obvious to the user and may thus be difficult to interpret appropriately.

In order to evaluate which strategy produces superior visualization results, usage of a single initial OTF for rendering of time-varying 3DUS datasets or recalculation of the OTF for every new viewing angle and/or time increment, we have performed the following tests. Since we had no access to real 4DUS datasets, we have simulated a 4D scenario of type (2) by rotating static 3D datasets. The rotation simulates the movement of the transducer relative to a static anatomy/object. Results of the rendering of these 4D sequences can be compared using a single OTF in one case and an OTF individually recalculated for every viewing angle in the other case. We applied this technique to both a real world dataset (the fetus of Fig. 1) and an artificial dataset. The latter was aimed to mimic the scenario of a fetus embedded in amniotic fluid, only using basic geometric shapes to make differences in the rendered images (if any) more apparent. An ellipsoid, a cylinder and various spheres were used to represent object surfaces (high signal intensity), surrounded by tissue of intermediate signal intensity which featured a ‘window’ of low signal intensity (amniotic fluid). We added speckle noise as it occurs in real world US datasets to our artificial dataset. Due to the coherent nature of the radiation used in US imaging, the statistics of the noise apparent in US images is not Gaussian but rather similar to noise as occurring in Radar images or images from applications using optical laser radiation which has Rayleigh characteristics (see [33–35]). We rotated both datasets within a range of $[-45°, +45°]$ relative to an initial view in steps of 1°.

6. Results

6.1. OTF design

For the visualization of the $199\times197\times199$ voxel data set in Fig. 1 we use a grid of 15×15 tube cores with an inter-tube core distance of 13 voxels in each direction and a cell size of $5\times5\times5$ voxels. The size of the Gaussian kernels employed for setting up the scale space is 2^n, $3\le n\le7$. Both the original and the modified OTF are depicted in Fig. 6.

By a highly efficient implementation of the proposed algorithm using the Intel® C++ Compiler 7.0 and Intel® VTune™ Performance Analyzer 7.0 we yield an OTF for this volume within 29.38 ms on a 2.5 GHz Intel® Pentium® 4-based PC with 512 MB RAM. Thus, we could compute an optimal OTF at a rate of 34 volumes per second. It is worth mentioning that we did not optimize volume data access assuming that data access is one task of the ray-casting/interpolation subsystem. Expectedly, surface detection is by a factor of more than 5 more demanding than the iterative OTF design, even though scalespace has been implemented by an FFT.

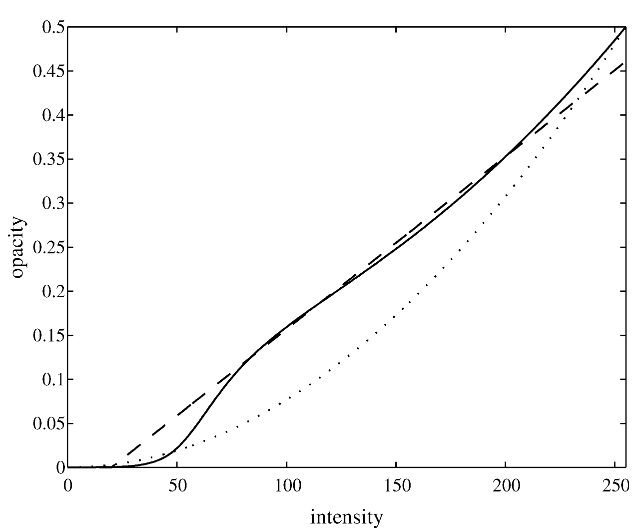

Fig. 6. Dotted: initial parabolic OTF. Solid: result of the modification by basis functions of Fig. 5, used in Fig. 1(c). Dashed: manually adjusted piecewise linear OTF used in Fig. 1(b).

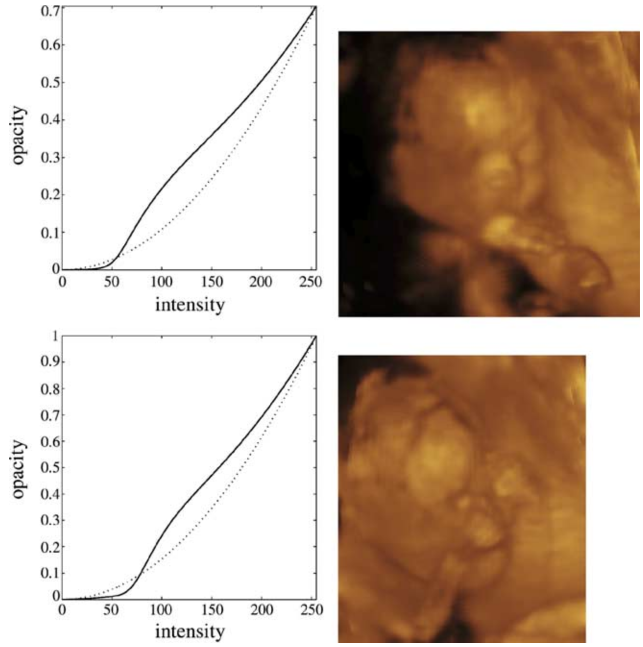

Fig. 7. Further results of adaptive OTF design applied to moderate quality data sets arising frequently in clinical setting. The original OTF is indicated as dotted line, the adapted OTF as solid line.

Further examples of the quality one can obtain by following our approach are depicted in Fig. 7. These are data sets of moderate quality as they often arise in 3DUS imaging in clinical settings. Of course the proposed method also yields excellent results for data sets of superior quality, such as in Fig. 8.

All images presented here have been rendered in real time using a high-quality volume renderer [36] running on current consumer graphics hardware such as an ATI Radeon 9800. The entire volume is stored in a 3D texture, which allows to perform re-sampling and evaluation of the volume rendering integral by slicing this texture with multiple view-aligned polygons [37] and employing hardware compositing.

Compositing of sample contributions along viewing rays has been done with floating point accuracy, which is crucial

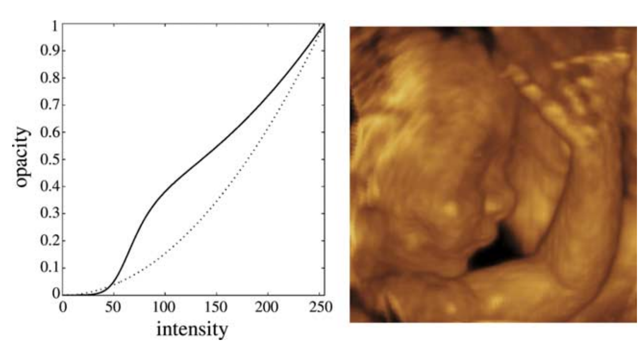

Fig. 8. The adaptive OTF design also yields superior results for high-quality data sets.

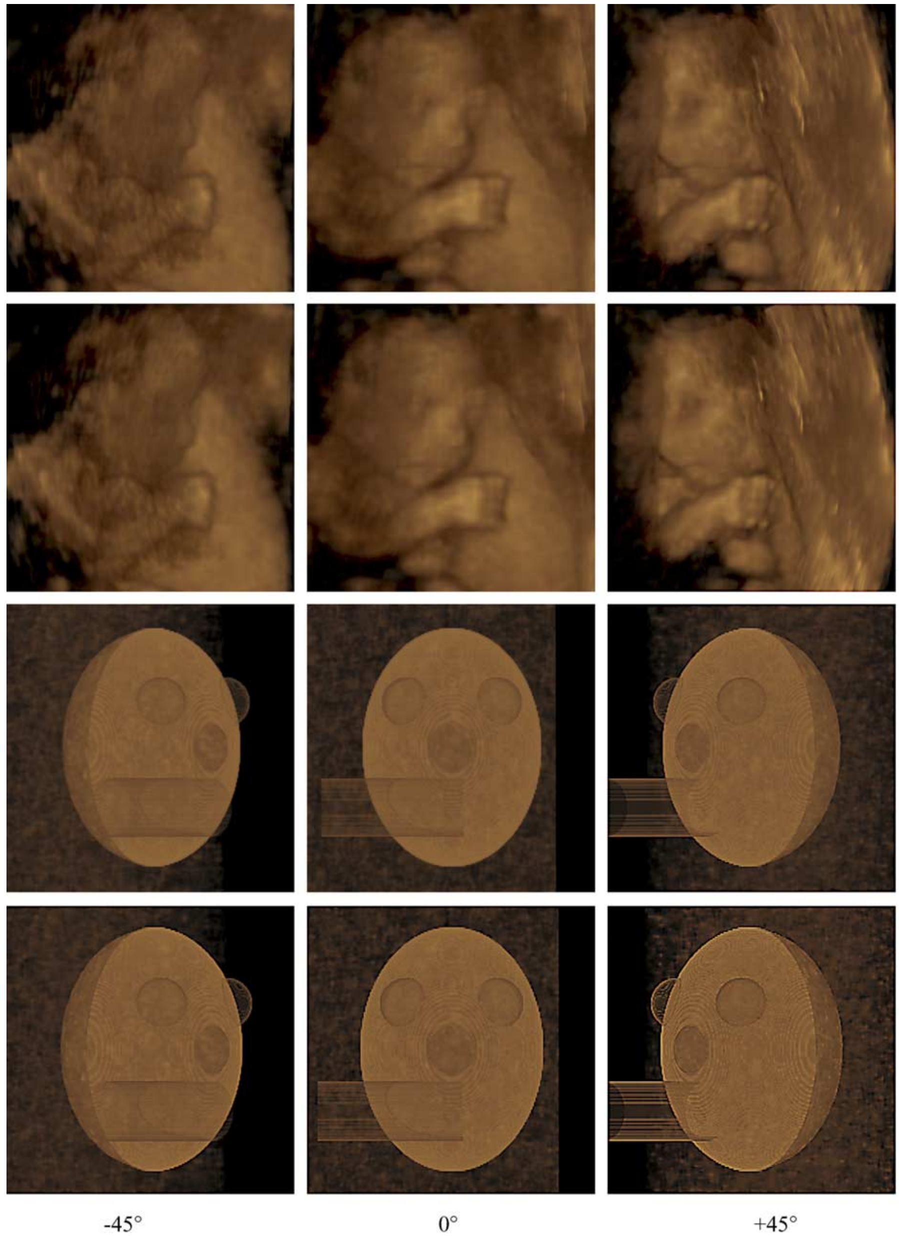

Fig. 9. Renderings of a fetus and an artificial dataset using different viewing directions and OTFs (top row of each dataset: same OTF has been used for rendering of all frames; bottom row: individual OTFs have been used for each frame).

for high-quality results when high sampling rates are used. Floating point rendering and compositing has become possible on the latest generation of graphics cards.

The transfer function itself is stored in a 1D texture map and can be changed interactively. Moreover, each rendered frame is allowed to use an individual transfer function.

Typical frame rates on an ATI Radeon 9800 are 30 fps for preview compositing quality (eight bits per color channel) and 10 fps for high-quality compositing (16 bit floating point per color channel).

Thus, the interactive generation of an OTF can easily be integrated with real time volume rendering for display purposes.

6.2. Temporal coherence

The most natural way to visualize rendering results of 4D datasets is of course to view the corresponding movies. We have therefore provided the movies showing the rotation of the fetus dataset and the artificial dataset as described in Section 5 on the WWW.[1]

Fig. 9 shows stills taken from these movies at distinct rotation angles. The OTFs calculated for these angles do of course differ in both datasets (see Fig. 10), but only to an

[1] http://www.acv.ac.at/publications.

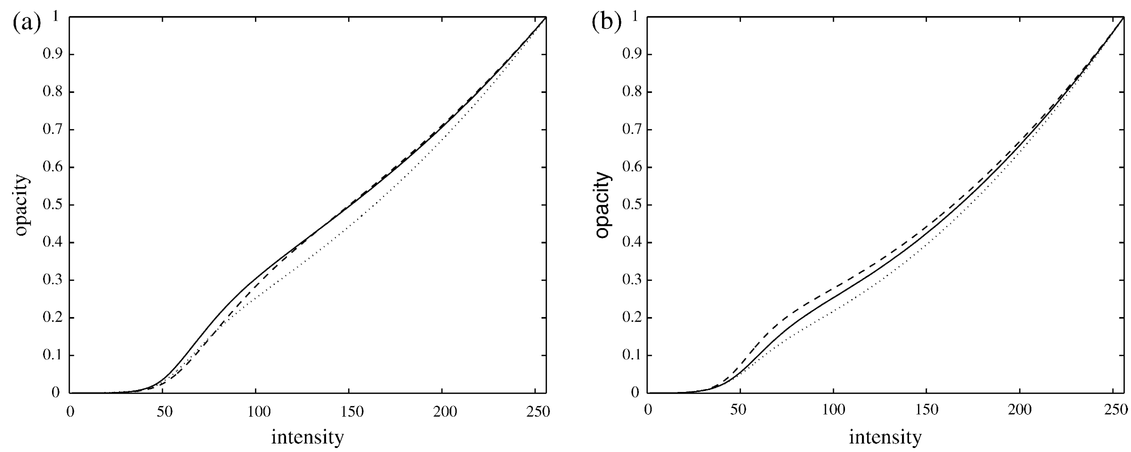

Fig. 10. Dependence of the OTF on the viewing angle. (a) Fetus dataset and (b) artificial dataset (dotted: −45°, solid: 0°, dashed: +45°).

extent that the final rendered images are visually almost identical. This is not surprising since the surface extraction method uses information from various locations throughout the volume and thus is robust with respect to slight shifts of the tube core positions. When viewing the movies generated using an individual OTF for each viewing angle (i.e. for each frame) it appears that, while the rotational movement is smooth, the image is slightly flickering. This is due to the slight variation of the OTFs between consecutive frames. On the other hand, if one single OTF is used for rendering of the complete 4D sequence, there is no flickering and the overall impression is smooth. One possible remedy for the flickering is to locally smooth the OTF within the time domain. The OTF at (α_n/t_n) thus incorporates information from OTFs at previous instances (α_{n-m}/t_{n-m}) which avoids abrupt changes in the OTF. Any reasonable smoothing technique can be employed, for example a moving least squares algorithm or a simple mean value. We have used both a mean value smoothing and a moving least squares calculation with a polynomial of order two, taking the past five OTFs to determine the current OTF. This OTF was then used for rendering of the current frame, while the originally calculated (unsmoothed) OTF was incorporated into the calculation of the next OTF. Our tests have shown that this slight time domain smoothing effectively eliminates the flickering (the corresponding movie can be found on the WWW site mentioned). Fig. 11 shows the effect of time domain smoothing on a subset of the OTFs generated for the 90° rotation of the fetus dataset. The additional calculations needed embarking on such a strategy are not time consuming but nevertheless only justified if view-dependent OTFs yielded highly superior visualization of 4D datasets in comparison to using a single static OTF. If details were revealed at specific viewing angles which were not distinguishable using a single static OTF, a view-dependent OTF would be highly advantageous.

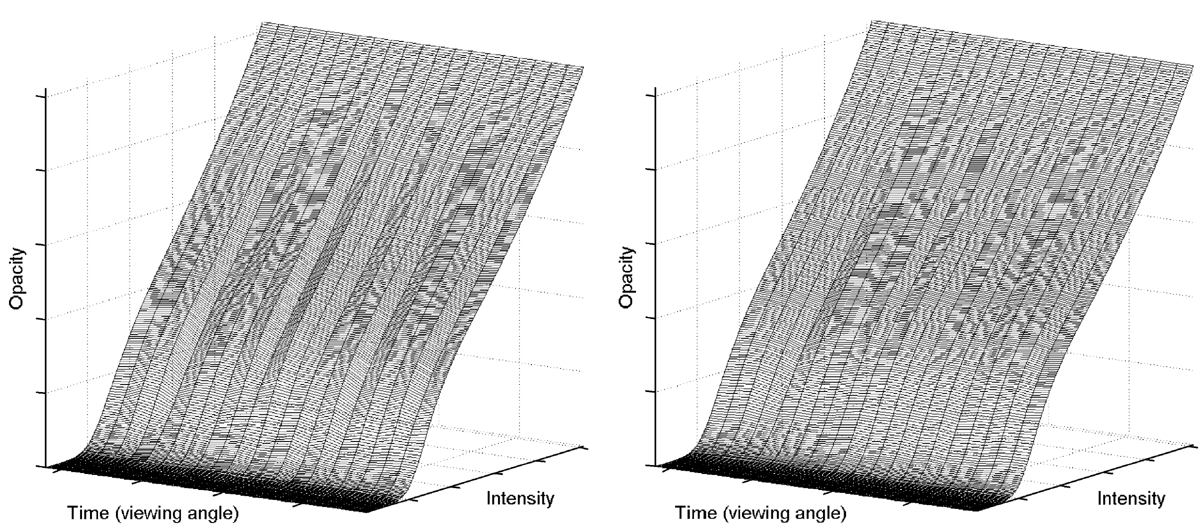

Fig. 11. Effect of time domain smoothing on a subsequence of OTFs generated for a rotating fetus dataset (left, smoothed; right, original). The time domain shown covers a rotation of the dataset of 20°.

7. Conclusions

We presented a novel technique for adaptively designing an opacity transfer function for a sonographic data set in real time. By analyzing tube cores we yield an estimate for the position of the most prominent tissue transition, in rendering direction, in a highly efficient way. We analyze the tube core cells prior to and at the detected interface and use this information to adaptively design an initial, parabolic OTF in a multiplicative way. Of course, the tube core modification can also be applied to OTFs of different shape. Our experiments on temporal coherence have shown that the proposed method for OTF calculation is not only very efficient but also very robust. It thus allows online-computation of OTFs for an entire sequence of acquired volumes. In a clinical setup, time domain smoothing can be employed to produce coherent image sequences whenever view-dependent/time-varying OTFs are desired. In our tests, OTFs calculated from a large range of viewing angles (90°) of a given 3D dataset did only differ slightly. Therefore, computation of a single OTF for an entire 4D sequence might be sufficient in many cases, guaranteeing a coherent view of the 3D scenario.

8. Summary

Three-dimensional ultrasound imaging is an ever increasing domain within the spectrum of medical imaging. Its various advantages over CT and MR imaging have made it a popular diagnostic tool despite its main inconvenience—the low signal-to-noise ratio. The latter together with real time data acquisition make 3D visualization of ultrasound data challenging. Direct volume rendering is a method frequently used to accomplish this task. While there are a couple of transfer function design approaches for CT and MRI data, direct volume rendering of ultrasound data commonly still relies on manual adjustment of an inflexible piecewise linear opacity transfer function (OTF) on a trial-and-error basis.

In this article, we present a method for automatic computation of optimized OTFs for visualization of sonographic datasets. The proposed algorithm extracts information about possible surfaces positions (interfaces between tissues of different echogenicity) and intensities within the viewing direction. This information is then used to modify a parabolic transfer function in a multiplicative way thus that tissue interfaces are accentuated while lower signal tissues prior to the interface in viewing direction are omitted. Our approach is inspired by a frequent scenario: imaging of a fetus embedded in amniotic fluid.

We show that our technique is efficient and capable of providing OTFs in real time. We demonstrate the appropriateness of our approach on data sets of moderate quality arising frequently in clinical settings. Furthermore, we present considerations on temporal coherence, e.g. visualization of 4D (time-varying 3D) datasets.

Acknowledgements

The authors would like to acknowledge the contributions of Armin Schoisswohl of GE Medical Systems Kretztechnik, Zipf, Austria, in providing the 3DUS data sets and valuable discussions, as well as the contributions of Christoph Berger from VRVis Research Center.

This work has been carried out within the K*plus* Competence Center ADVANCED COMPUTER VISION and was funded by the K*plus* program. Parts of this work have been done at the VRVis Research Center, which is also funded by the Austrian K*plus* program.

References

[1] Fenster A, Downey DB. Three-dimensional ultrasound imaging. In: Beutel J, Kundel HL, Van Metter RL, editors. Handbook of medical imaging, 1st ed, vol. 1. SPIE; 2000. p. 463–509 [chapter 7].

[2] Fenster A, Downey DB, Cardinal HN. Three-dimensional ultrasound imaging. Phys Med Biol 2001;46(5):R67–99.

[3] Lees W. Ultrasound imaging in three and four dimensions. Semin Ultrasound CT MRI 2001;22(1):85–105.

[4] Sakas G, Walter S, Grimm M, Richtscheid M. Free hand acquisition, reconstruction and visualization of 3d and 4d ultrasound. Radiologe 2000;40(3):295–303.

[5] Nelson TR, Pretorius DH. Three-dimensional ultrasound imaging. Ultrasound Med Biol 1998;24(9):1243–70.

[6] Ghanei A, Soltanian Zadeh H, Ratkewicz A, Fang FY. A three-dimensional deformable model for segmentation of human prostate from ultrasound images. Med Phys 2001;28(10):2147–53.

[7] Hu N. Prostate surface segmentation from 3d ultrasound images. In: Proceedings IEEE symposium on biomedical imaging 2002 p. 613–16.

[8] Holmes DR, Davis BJ, Bruce CJ, Robb RA. 3d visualization, analysis, and treatment of the prostate using trans-urethral ultrasound. Comput Med Imaging Graph 2003;27(5):339–49.

[9] Bouchet LG, Meeks SL, Goodchild G, Bova FJ, Buatti JM, Friedman WA. Calibration of three-dimensional ultrasound images for image-guided radiation therapy. Phys Med Biol 2001;46(2): 559–77.

[10] Fenster A, Surry K, Smith W, Gill J, Downey DB. 3d ultrasound imaging: applications in image-guided therapy and biopsy. Comput Graph 2002;26(4):557–68.

[11] Finger PT, Khoobehi A, Ponce-Contreras MR, Rocca DD, Garcia JP. Three dimensional ultrasound of retinoblastoma: initial experience. Br J Ophthalmol 2002;86(10):1136–8.

[12] Grasbon T, Schriever S, Hoops JP, Mueller AJ. 3-d ultrasound initial experiences in various eye diseases. Ophthalmologe 2001;98(1): 88–93.

[13] Lang R, Sugeng L. A fantastic journey: 3d cardiac ultrasound goes live. Radiol Manage 2002;24(6):18–22.

[14] Lassige TA, Benkeser PJ, Fyfe D, Sharma S. Comparison of septal defects in 2d and 3d echocardiography using active contour models. Comput Med Imaging Graph 2000;24(6):377–88.

[15] Corsi C, Borsari M, Consegnati F, Sarti A, Lamberti C, Travaglini A, et al. Left ventricular endocardial surface detection based on real-time 3d echocardiographic data. Eur J Ultrasound 2001;13:41–51.
[16] Klingensmith JD, Vince DG. B-spline methods for interactive segmentation and modeling of lumen and vessel surfaces in three-dimensional intravascular ultrasound. Comput Med Imaging Graph 2002;26(6):429–38.
[17] Shekhar R, Cothren RM, Vince DG, Chandra S, Thomas JD, Cornhill JF. Three-dimensional segmentation of luminal and adventitial borders in serial intravascular ultrasound images. Comput Med Imaging Graph 1999;23(6):299–309.
[18] Haas C, Ermert H, Holt S, Grewe P, Machraoui A, Barmeyer J. Segmentation of 3d intravascular ultrasonic images based on a random field model. Ultrasound Med Biol 2000;26(2):297–306.
[19] Kovalski G, Beyar R, Shofti R, Azhari H. Three-dimensional automatic quantitative analysis of intravascular ultrasound images. Ultrasound Med Biol 2000;26(4):527–37.
[20] Sakas G, Schreyer L-A, Grimm M. Preprocessing and volume rendering of 3d ultrasonic data. Comput Graph Appl, IEEE 1995; 15(4):47–54.
[21] Sakas G, Walter S. Extracting surfaces from fuzzy 3d ultrasound data. In: ACM SIGGRAPH 1995.: ACM; 1995 p. 465–74.
[22] Levoy M. Display of surfaces from volume data. IEEE Comput Graph Appl 1988;8(3):29–37.
[23] He T, Hong L, Kaufman A, Pfister H. Generation of transfer functions with stochastic search techniques. In: Proceedings visualization.: IEEE; 1996 p. 227–34, 489.
[24] Kindlmann G, Durkin JW. Semi-automatic generation of transfer functions for direct volume rendering. In: IEEE symposium on volume visualization.: IEEE; 1998 p. 79–86, 170.
[25] Kniss J, Kindlmann G, Hansen C. Multidimensional transfer functions for interactive volume rendering. IEEE Trans Vis Comput Graph 2002;8(3):270–85.
[26] Sato Y, Westin C, Bhalerao A, Nakajima S, Shiraga N, Tamura S, et al. Tissue classification based on 3d local intensity structures for volume rendering. IEEE Trans Vis Comput Graph 2000;6(2):160.
[27] Fujishiro I, Takeshima Y, Azuma T, Takahashi S. Volume data mining using 3d field topology analysis. IEEE Comput Graph Appl 2000;20(5):46–51.
[28] Bajaj CL, Pascucci V, Schikore DR. The contour spectrum. In: Proceedings visualization.: IEEE; 1997 p. 167–73, 539.
[29] Rezk-Salama Ch, Hastreiter P, Scherer J, Greiner G. Automatic adjustment of transfer functions for direct volume rendering. In: Proceedings vision, modeling, and visualization 2000 p. 357–64.
[30] Hoenigmann D, Ruisz J, Haider C. Adaptive design of a global opacity transfer function for direct volume rendering of ultrasound data. In: Visualization, VIS 2003.: IEEE; 2003 p. 489–96.
[31] Witkin AP. Scale-space filtering. In: Proceedings international joint conference on artificial intelligence. vol. 2. Los Altos, CA: Morgan Kaufmann; 1983 p. 1019–22.
[32] Jankun-Kelly TJ, Kwan-Liu M. A study of transfer function generation for time-varying volume data. In: Volume graphics 2001 proceedings 2001 p. 51–68.
[33] Burckhardt CB. Speckle in ultrasound b-mode scans. IEEE Trans Sonics Ultrason 1978;25(1):1–6.
[34] Wagner RF. Statistics of speckle in ultrasound b-scans. IEEE Trans Sonics Ultrason 1983;30(3):156–63.
[35] Petrou M. Modelling the histogram of various classes in sar images. Pattern Recognit Lett 2002;23:1103–7.
[36] Hadwiger M, Berger C, Hauser H. High-quality two-level volume rendering of segmented data sets on consumer graphics hardware. In: Visualization VIS 2003.: IEEE; 2003 p. 301–8.
[37] Westermann R, Ertl T. Efficiently using graphics hardware in volume rendering applications. In: SIGGRAPH 98, computer graphics proceedings 1998 p. 169–77.

Bernhard Petersch graduated in 2001 from the Vienna University of Technology (TU Wien), Austria as a Dipl.-Ing. (≈ MSc) after studying technical physics there and at the Swiss Federal Institute of Technology Lausanne (EPFL). During his diploma thesis, which he concluded at the Institute of Medical Radiation Physics, Karolinska Institute/Stockholm/Sweden, he started working in the field of medical imaging and radiotherapy. In 2003, he received the degree of Dr Techn. (PhD) from the TU Wien after having worked for BrainLAB (Munich, Germany) in the field of real time image processing for radiotherapy surveillance. Since then, Bernhard Petersch works as researcher at Advanced Computer Vision (ACV) in Vienna. His main interests include 3D medical imaging, image guided radiotherapy and surgery, as well as medical engineering in general.

Markus Hadwiger received a masters degree (Dipl.-Ing.) in computer science from the Vienna University of Technology in 2000, working on the design and implementation of a portable 3D game engine. In 2004, he received a PhD degree (Dr Techn.) in computer science also from TU Vienna, in cooperation with the VRVis Research Center, concentrating on high-quality volume rendering and texture filtering with graphics hardware. He has been a researcher at the VRVis Research Center in Vienna since 2000, working in the Basic Research on Visualization group, and the Medical Visualization group (since 2004). His research interests are real time volume rendering, medical visualization and segmentation, and computer games.

Helwig Hauser graduated in 1995 from Vienna University of Technology, Austria, as a Dipl.-Ing. (≈ MSc) after studying computer science with special focus on computer graphics. In 1998, after a research project in the field of scientific visualization (visualization of dynamical systems), he received the degree of Dr Techn. (PhD) also from TU Wien. From 1994 until mid-2000, Helwig Hauser worked as a Univ.-Ass. (assistant professor) at the Institute of Computer Graphics at TU Wien, where he was co-leader and founding member of the visualization group around Prof. E. Gröller (http://www.cg.tuwien.ac.at/research/vis/). Since 2000, he works for the VRVis Research Center in Vienna, Austria, as a key researcher in the field of visualization (volume/flow/information visualization, http://www.VRVis.at/vis/). In 2003, Helwig Hauser became the scientific director of VRVis (http://www.VRVis.at/, ≈50 researchers). His main interests include volume, flow, and information visualization, and the combination thereof. Helwig Hauser is member of IEEE and EUROGRAPHICS.

Dieter Hönigmann studied Informatics at the Vienna University of Technology, Austria. He specialized in Medical Imaging and Neuroinformatics and received (with distinction) his Diploma in Informatics in 1998. From 1999 to 2000 he was with the IT management centre (EMB) of the Vienna General Hospital. Since 2001, he is with Advanced Computer Vision (ACV), concerned with the development of 3D/4D ultrasound data processing algorithms. He is currently pursuing his PhD research at the Faculty of Mathematics and Geoinformation of the Vienna University of Technology. Dieter Hönigmann is member of IEEE and ACM.