IEEE Visualization 2004 Tutorial 4

Advanced Virtual Medicine: Techniques and Applications for Medicine-oriented Visualization

Dirk Bartz[1] and Y. Wu[2] (Editors and Tutorial organizers)
[1]University of Tübingen and
[2]TeraRecon

(c) 2004
Tübingen, Germany and Concord, USA

Tutorial Summary

One of the largest application domains of visualization is medicine. Numerous techniques ranging from medical imaging to virtual medicine are used in both daily health care practice and in research. In particular, recent developments in minimally-invasive surgery require an advanced planning and intra-operative support through computer science methods. Even until recently, 2D medical imaging remained to be the standard for the planning of these interventions, while 3D and virtual medicine methods are only slowly merging into hospitals. This however, has dramatically changed with the introduction of intra-operative navigation systems.

In this course, we will first give an introduction into medical imaging methods such as data acquisition, data analysis, segmentation, registration and rendering both in 2D and 3D. Based on this foundation, the course will further explore a variety of applications for virtual medicine. In particular, we will discuss virtual endoscopy, virtual reality methods for surgery, soft-tissue-, and motion simulation, functional Magnetic-Resonance Imaging some of the most actively researched fields in virtual medicine. Together, these topics form important components towards more realistic interaction with the virtualized human body. Besides the technical aspects, we will also discuss the advantages to traditional methods, but will also illustrate their specific and inherent limitations.

For updates and additional information, see
http://www.gris.uni-tuebingen.de/~bartz/tutorials/vis2004

1 Prerequisites

We assume a basic understanding of 3D graphics, visualization and medical-imaging technology. Although recent research results will be presented, we are not in particular targeting at experts in the field.

2 Tutorial Speaker

The material of this tutorial has been prepared by the following six people:

Dirk Bartz is head of the research group on Visual Computing for Medicine (VCM) of the University of Tübingen. His recent works covers interactive virtual medicine and scientific visualization. In 2002, he received the NDI Young Investigator Award for his work in virtual endoscopy and intra-operative navigation.

Dirk studied computer science and medicine at the University of Erlangen-Nürnberg and the SUNY at Stony Brook. He received a Diploma (M.S.) in computer science from the University of Erlangen-Nürnberg, and a Ph.D. in computer science from the University of Tübingen (both in Germany).

Jan Hardenbergh has worked in the 3D computer graphics industry for 25 years. He now works on software for VolumePro hardware engine at TeraRecon, Inc. He started when the project was still in research at MERL. He also helped bring up the Apollo DN10KVS and Oki TrianGL chips, both performance leaders for brief times. He has worked on the integration of 3D and the Internet by implementing products and serving on the architecture boards of both VRML and PEX (PHIGS and X). Jan also has a rich CAD (Design) background working with Computervision, Cognition and Revit and looks forward to working in CAD (Diagnostics).

Michael Hauth is a member of the research staff of the Computer Graphics Laboratory (GRIS) at the University of Tübingen. His current research focuses on simulation of deformable objects with applications in textile and medical animation. Michael studied computer sciences, physics and mathematics at the University of Tübingen. He received a Diploma in computer sciences in 1999 and a Diploma in mathematics in 2000. In 2000/2001 he held a visiting post as research assistant at the Numerical Analysis Group of the University of Geneva. He authored and co-authored several papers covering simulation and animation of deformable objects.

Klaus Mueller is currently an Assistant Professor at the Computer Science Department at Stony Brook University. His current research interests are computer graphics, medical-imaging, and virtual and augmented reality. He has authored or co-authored more than 30 conference and journal papers on volume rendering and on filter design. One of these papers won the Visualization "Best Paper" award in 1997. He has also won the Visualization "Best Hot Topic" award in 1999 and the NSF CAREER award in 2000. He has cochaired the 2001 and 2003 Volume Graphics Workshop and the 2002 Symposium on Volume Visualization and Graphics.

Markus Wacker is a member of the research staff of the Computer Graphics Laboratory (GRIS) at the University of Tübingen and currently head of the Virtual TryOn project in Tübingen. His research focuses on physically correct simulation of deformable objects for textile and medical applications. Markus studied mathematics and physics at the University of Tübingen and recevied a Ph.D. in mathematics from the University of Tübingen.

Yin Wu is a principal computer scientist in Terarecon Corporation, which is an imaging technology company that develops professional imaging applications, with special emphasis on medical imaging. Prior to her work at Terarecon she works on volume rendering and medical visualization in Mitsubishi Electric Research Lab (MERL). She received her Ph.D in Computer Science at University Geneva at 1998 on facial animation and biomechanics based aging simulation. Her research and development focus on physical simulation and interactive medical imaging visualization.

3 Schedule

The presentation itself is organized in four parts:

I.	**Welcome and Introduction** (Bartz, Wu)	**60min**
	Examples of Virtual Endoscopy Applications	
	Time Density Analysis of Tumors, Cardiac Time Volume Analysis, fMRI	
II.	**Foundations**	**120min**
	Medical Imaging Techniques (Bartz, Wu)	20min
	(X-Rays, Computed-Tomography, (functional) Magnetic-Resonance	
	Imaging, Rotational Angiography, Sampling-Theory)	
	Data Pre-processing (Bartz)	60min
	(Segmentation, Classification, Registration, and Fusion)	
	Coffee Break	
	Visualization and Navigation Techniques (Mueller, Wu)	40min
	(Indirect and direct surface volume rendering,	
	camera path generation vs. free fly-throughs	
	Lunch Break	
III.	**Applications of Virtual Medicine**	
	Virtual Endoscopy (Bartz)	20min
	(Available Systems, Advantages and Limitations of Virtual endoscopy)	
	Simulated Surgical Procedures (Wu)	40min
	fMRI for Surgical Planning (Hardenbergh)	45min
	Coffee Break	
	Soft-Tissue-Simulation (Wacker, Hauth)	**90min**
	General Basics	15min
	Physical and mathematical Background	45min
	State-of-the-art in simulation of Human Soft-Tissue	20min
	Soft-Tissue for minimally-invasive Surgery Simulation	10min
IV.	**Questions and Answers** (All)	15min

4 Introduction

These notes introduce into some topics covered in this tutorial. In particular, it introduces concepts of volumetric data representation and volume rendering (Section 5). Thereafter, Section 6 will introduce various scanning modalities. An overview of the various virtual medicine applications, in particular focusing on the skull, is given in Section 7. For a detailed overview on virtual endoscopy, we refer to the respective Eurographics 2003 STAR-report, which is included in the additional material on the conference DVD. Furthermore, the appendix contains interesting WWW links on the topic, to related papers, and – of course – the slides of the presentation. However, the webpage mentioned above probably contains a more up-to-date version of the presentation.

5 Volume Data

- Dirk Bartz, University of Tübingen, Germany

Traditionally, computer graphics represented a model as a set of vectors which were displayed on vector graphic displays [Foley et al. 1996]. With the introduction of raster displays, polygons became the basic rendering primitive, where the polygons of a model were rasterized into pixels, which represent the compounds of the framebuffer. The concept of an image composed of two-dimensional discrete pixels was later extended into a volume, composed of discrete voxels, arranged in a three-dimensional array. The representation of model data as a volume is particularly useful for scientific visualization, since the data is usually computed by a simulation on a multi-dimensional grid with three or more dimensions, or measured by a volumetric scanner like a CT scanner.

In the following sections, we will briefly introduce into the concepts of volume datasets (Section 5.1) and volume rendering. Specifically, we describe the two primary approaches to generate a visual representation of the volume dataset; indirect (Section 5.3) and direct volume rendering (Section 5.4). Both approaches use voxel-based shading, which will be reviewed in Section 5.2.

5.1 Volume Data

A volumetric dataset is a three-dimensional array of data values, the voxels. We address these voxels with three indices i, j, k, indicating the position of a voxel within the volume dataset. If we assume an array of 512 data values in each dimension, each index runs from

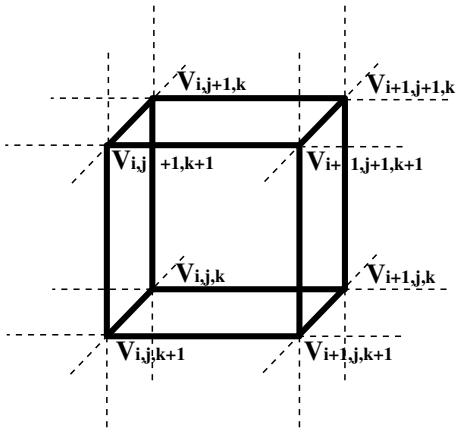


Figure 1: Volume cell in a rectilinear grid

$i, j, k = 0..511$, forming a $512 \times 512 \times 512$ volume dataset. The three-dimensional array can also be seen as a stack of two-dimensional arrays of data values and each of these two-dimensional arrays as an image (or slice), where each of the data values represents a pixel. This alternative view is motivated by the slice oriented, traditional way physicians look at a volumetric dataset).

Eight neighboring voxels ($V_{i,j,k}$, $V_{i+1,j,k}$, $V_{i,j+1,k}$, $V_{i+1,j+1,k}$, $V_{i,j,k+1}$, $V_{i+1,j,k+1}$, $V_{i,j+1,k+1}$, $V_{i+1,j+1,k+1}$) form a *volume cell*, or simply cell[1] (see Fig. 1). The *pixel distance* between the voxels within one slice or image is usually equal. However, the *slice distance* between two neighboring voxels in two neighboring slices is frequently not equal to the pixel distance. We call these datasets *anisotropic* volume datasets. If slice and pixel distance are equal, we speak of an *isotropic* volume dataset. The distances are also called *voxel spacing*. In most datasets, all voxels are aligned on a *cartesian grid*, according to voxel spacing. Therefore, we speak of a cartesian or *rectilinear grid* dataset. If the voxel spacing is constant, the grid is also called an *uniform grid*, or a *non-uniform grid* vice-versa. Some scanners (i.e., 3D ultrasound) produce volume datasets which are not aligned on a cartesian grid, but on a grid which is bent. In other words, the grid geometry has changed, but the grid topology (connectivity) has remained the same. These grids are called *curvilinear grids*. Together with cartesian grids, they are classified as *regular grids*. If also the topology is no longer of a cube-like (possibly bent) cell, the resulting grids are classified as *irregular* or *structured grids*. If the grid topology is using various cell types of different connectivity, we speak of an *unstructured grid*. In the course of this thesis, we will only encounter regular, cartesian gridded volume datasets.

5.2 Voxel-based Shading

The shading of an object has a significant influence on the appearance of the object, since it simulates the interaction of light and the material properties of the object. Besides material and light, the shading is determined by the normals of the object, which define its surface. In traditional computer graphics, these normals are defined at the vertices of a triangle and they are interpolated over the area of the triangle to compute the shading of the triangle.

In contrast, the normals of a volumetric representation are defined at the positions of the data values. Due to the lack of a specific surface in a volume dataset, the normals are approximated by gradient operators, which describe changes in the material properties of a volume dataset. This method was originally motivated by the *partial volume effect* (see Appendix 6), which generates smooth density changes between two different materials (i.e., bone and tissue) in CT datasets [Höhne and Bernstein 1986], thus the density gradient represents the surface of the material interface between bone and tissue. The partial volume effect also affects other image modalities, such as MRI, or rotational angiography (see Appendix 6).

The currently used standard gradient operator – which we also use later on – computes the central difference between the six direct neighbors of a voxel [Höhne and Bernstein 1986], without taking into account the value of the voxel itself. Other gradient operators include the intermediate difference operator (sometimes also called forward or backward difference operator) – which consider the voxel values at the current position and at three neighboring voxels [Lichtenbelt et al. 1998] – and the Sorbel operator, which considers the full 26-neighborhood of a voxel to approximate the gradient [Lichtenbelt et al. 1998].

Once the normals at the voxel positions are computed by the (normalized) gradient operators, the normals at the sample points within a volume cell are usually computed by trilinear interpolation of the normals at the eight voxels of this cell.

5.3 Indirect Volume Rendering

In indirect volume rendering (IVR), a "traditional" computer graphics representation is generated, usually by extracting an isosurface or an isocontour from the dataset. The respective graphics primitives (i.e., polygons) are then rendered using standard computer graphics hardware. The most popular approach of IVR is probably the Marching Cubes algorithm [Lorensen and Cline 1987], where each volume cell of eight

[1] An alternative, somewhat outdated terminology describes a voxel as a cubic cell around a data value which extents half way to the next neighboring voxels. The basic difference is the notion of interpolation between the data values, which is nearest-neighbor (non-continuous) for the older scheme, and trilinear (continuous, or C^0) for the current scheme.

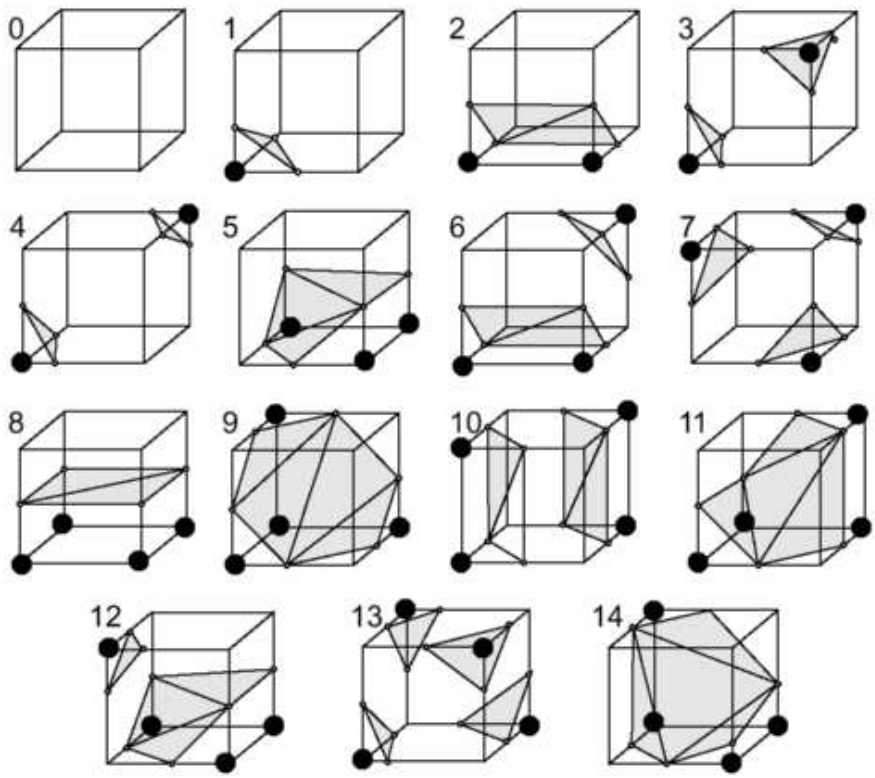


Figure 2: Reduced Marching Cubes case table [Kaufman et al. 1996]

neighboring voxels is classified according to the specified isovalue (which specifies the isosurface). 2^8 possible combinations whether the scalar value of a voxel of the cell is above or below (or equal to) the isovalue are stored in a table of 256 cases. In the original paper on Marching Cubes, these 256 cases were reduced to 15 cases by inverting or rotating the classification cubes (see Fig. 2). However, later research showed that these reduced case table generates inconsistencies, which result in holes in the isosurface [Kaufman et al. 1996]. Therefore, we only use a full 256 case table, similar to many other state-of-the-art implementations of the Marching Cubes algorithms [Schroeder et al. 1998].

Up to five triangles per cell are generated depending on the classification case. To approximate the normals at the vertices of the generated triangles, each normal is computed by a trilinear interpolation of the normals at the eight voxels of the current cell, which in turn are computed by a central difference operator (see Section 5.2) [Höhne and Bernstein 1986].

Several approaches addressed improvement of the basic Marching Cubes algorithm. Most of them focused on skipping of regions which do not contain the isosurface, by applying hierarchical data-structures, like quadtrees and octrees [Samet 1994], k-D-trees [Bentley 1975], and BSP-trees [Fuchs et al. 1980]. Wilhelms and van Gelder introduced the Branch-ON-demand-Octree (BONO) [Wilhelms and van Gelder 1992], which stores the isovalue interval of the octants to skip octants whose isosurface interval does not contain the specified isovalue.

Many other approaches are known for the efficient isosurface or contour extraction of large volumetric scalar fields. Value partitioning methods [Livnat et al. 1996; Shen et al. 1996; Cignoni et al. 1997] store the minimum and maximum values of the voxels of a cell as a pair in a 2D field. Livnat et al. used a k-D-tree structure [Livnat et al. 1996], and Shen et al. [Shen et al. 1996] used lattice subdivision to subdivide this field. Cignoni et al. used an interval tree as search index for optimal efficiency [Cignoni et al. 1997]. Space and value partitioning methods are used for an efficient contour propagation, starting from a small set of seed cells [Itoh and Koyamada 1995; Bajaj et al. 1996].

5.4 Direct Volume Rendering

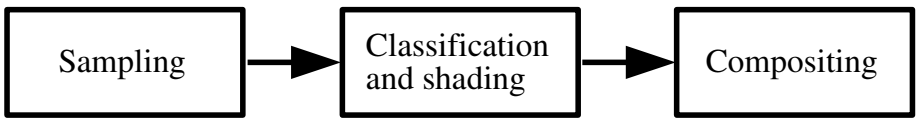


Figure 3: (Post-shading) Direct Volume Rendering Pipeline

In contrast to IVR, methods of direct volume rendering (DVR) are generating images without an intermediate polygonal representation.

Instead, the volume dataset is projected onto an image- or view-plane. Figure 3 shows an overview of the direct volume rendering pipeline using a post-shading scheme. After computing a sample according to the chosen algorithms, it is *classified* by the transfer functions and shaded according to the specified lights. Finally, the lit sample is *composited* with the previous samples using the over-operator [Porter and Duff 1984].

The transfer functions define the contribution of the samples by mapping their value(s) to a red, green, blue, and α (transparency) value. By evaluating the volume rendering integral (see Equation 1), the contribution of the volume data (through the transfer functions) lit by the possibly multiple light sources is computed for each pixel on the image- or view-plane, where I_λ is the color intensity of the pixel at $\vec{x}$, which receives a contribution of a ray of the length L – casted from $\vec{x}$ in direction $\vec{r}$ – and λ corresponds to the wavelength of a color (r, g, b). $C_\lambda(s)$ is the actual color of the sample at position s on the ray, with an extinction coefficient $\mu(s)$ at the sample, which is basically the opacity of the sample. The opacity-weight color is attenuated by the exponential opacity term, which collects the opacity of the ray through the volume (from entry point 0 to current sample s).

$$I_\lambda(\vec{x}, \vec{r}) = \int_0^L C_\lambda(s)\mu(s)e^{(-\int_0^s \mu(t)dt)} ds \tag{1}$$

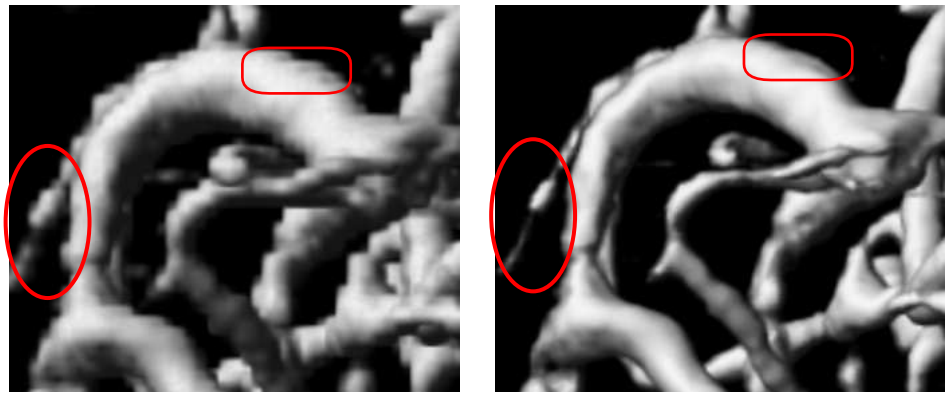
Unfortunately, the analytical volume rendering integral cannot be computed for the general case [Max 1995], hence a discrete approximation is used, where Δs is the width of the discretized integration interval:

$$I_\lambda(\vec{x}, \vec{r}) = \sum_{i=0}^{L/\Delta s} C_\lambda(s_i)\mu(s_i) \prod_{j=0}^{i-1} e^{(-\mu(s_j)\Delta s)} \tag{2}$$

The theoretical basis of the volume rendering integral is the *density emitter model* introduced by Sabella [Sabella 1988], which assumes a simplified model of the transport theory of light [Hege et al. 1993]. This simplified models takes only absorption and emission into account; physical terms like scattering, influence of the different wavelengths (red, green, and blue), or the influence of participating media are ignored.

Generally, DVR-algorithms are classified into front- and back-projection approaches. While front-projection approaches project the contributions of a dataset onto the view-plane, back-projection approaches trace the contribution for each pixel of the view-plane through the volume. Prominent candidates for front-projection are splatting [Westover 1990; Westover 1991; Mueller and Crawfis 1998], texture-mapping [Cullip and Neumann 1993; Cabral et al. 1994], or cell projection [Wilhelms and van Geldern 1991]. For back-projection, the most prominent candidate is ray-casting [Tuy and Tuy 1984; Levoy 1988].

Another important algorithmic classification is the shading scheme, which basically depends on when and how the normals at the sample points are calculated. Most approaches use the normalized gradient of the sample, based on central differences, to compute the normal at the sample [Höhne and Bernstein 1986] (also see previous section on voxel-based shading). However, this gradient can be computed before (pre-shading), or after the sample is computed (post-shading), as described above. Pre-shading pre-computes the normal gradients – and hence the illumination and shading – for all voxel values of the volume. The resulting colors and opacities are stored as a color volume instead of the original data. During the actual rendering process, the samples are computed based on the color volume. If the lighting conditions are changing (i.e., modifications of the lights for a diffuse lighting model, or of the view-point in a Phong-like lighting model), this requires the re-calculation of this color volume. Additionally, the colors need to be pre-multiplied by the opacities in order to avoid color bleeding artifacts [Wittenbrink et al. 1998] (see Fig. 4). In contrast, post-shading computes the classification and normals during rendering, after computing the sample.



(a) (b)

Figure 4: Color bleeding artifacts in a rotational angiography dataset. (a) Non-opacity weight, pre-shaded volume rendering; color bleeding artifacts are marked with red ellipse. (b) Post-shaded volume rendering.

In the following, we will briefly describe four of the most popular direct volume rendering approaches, where each of them has their own specific advantages and disadvantages.

Ray Casting

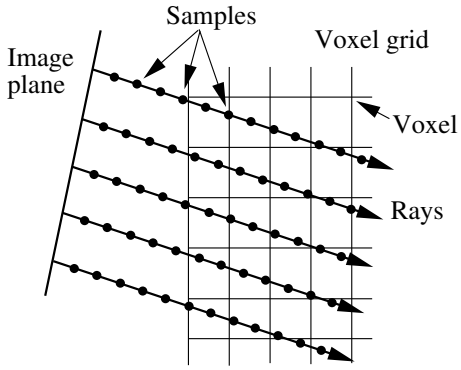


Figure 5: Ray casting of a volume dataset with parallel projection and uniform sampling.

In the image-space oriented ray casting approaches, rays are cast from the view-point through the view-plane into the volume. Along their way through that volume, samples are calculated usually at equal sampling distances between two sample points (see Fig. 5). A sample is computed based on trilinear interpolation within a cell of eight voxels. Thereafter, it is classified according to the transfer functions. If that sample has a contribution to the ray, the normal gradient is computed based on a trilinear interpolation of the normalized central differences at the eight voxels of the cell which contains the sample point. Finally, the sample is composited with the previous samples of the ray.

Acceleration methods include *early ray termination*, where the sampling along the ray is terminated, once (almost) full opacity has been reached, or space leaping, where a distance field or other data-structures indicate empty space where no sampling is required. Ray casting can also utilize oversampling within the view-plane and along the ray to account for high-frequency data in the data volume.

Shear-Warp Factorization

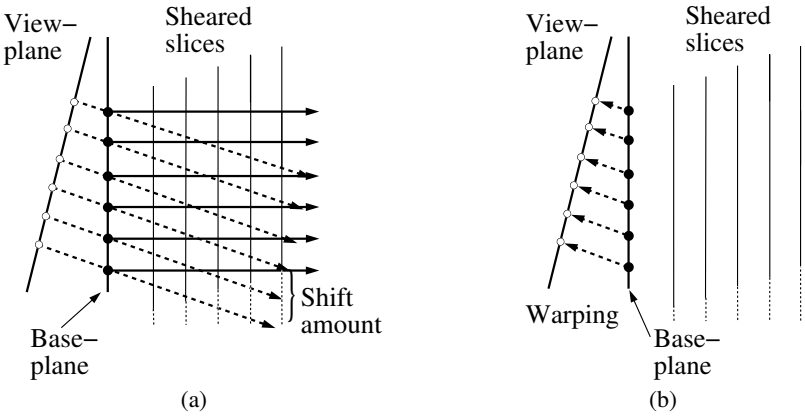


Figure 6: Shear-warp Factorization: (a) The volume slices are shifted (sheared) to account for the angle between view- and base-plane. (b) The base-plane image is warped to the view-plane.

A variation of the ray casting approach is the image-space oriented shear-warp factorization, which essentially factorizes the viewing transformation into a shearing and a warping matrix [Lacroute and Levoy 1994]. This enables the casting of rays from each pixel of a volume aligned base-plane, which is most parallel to the image- or view-plane, which in turn can be exploited for an optimized memory access. To take into account the angle between view- and base-plane, the individual slices are shifted (**sheared**) accordingly (see Fig. 6a). The samples of a whole scanline are computed simultaneously within one slice, employing a bilinear interpolation scheme. This results in a view-dependent sampling interval, which can vary between 1.0 for axis-aligned views, to 1.73 ($= \sqrt{3}$) for corner-on views. Perspective projection also requires the scaling of the slices to address the divergence of perspectively cast rays. Finally, the computed base-plane image is **warped** onto the view-plane (see Fig. 6b).

The Stanford VolPack-implementation of the shear-warp factorization is highly memory optimized. A pre-computed classification and a run-length-encoding of the opacity weighted voxels are used to rapidly skip the transparent, non-contributing volume space. VolPack uses a pre-shading scheme which computes the lighting on the pre-classified voxels, before generating the sample within a slice. Note that this (pre-)shading operation occurs during rendering, not in a pre-process, which allows the modification of the light settings without re-generating the pre-classified volume dataset.

Splatting

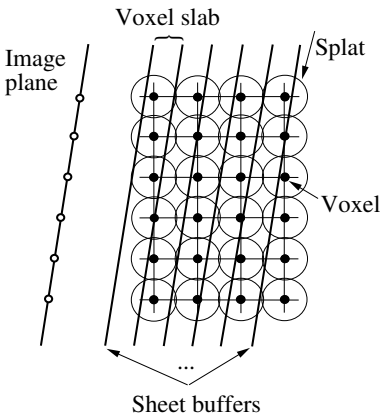


Figure 7: Splats arranged on view-plane aligned voxel slaps are projected into sheet buffers, which are in turn composited front to back into the final image on the view-plane.

The object-space oriented splatting approach projects each voxel onto the screen as an overlapping, orientation invariant Gaussian kernel with an amplitude scaled according to the voxel value [Westover 1990; Westover 1991]. High speed is obtained by a footprint lookup table of the pre-computed, radially symmetric kernel function. For correct compositing, the volume is processed by slices oriented most parallel to the view-plane, which causes severe brightness variations, such as popping artifacts in animated views. Mueller et al. modified the splatting approach such that *slabs* of the voxel kernels were processed in an image aligned fashion [Mueller and Crawfis 1998] and projected in *sheet buffers*, which in turn are composited into the final image on the view-plane. Other modifications include the *early splat elimination* for the removal of non-contributing splats from rasterization [Mueller et al. 1999]. Similar to early ray termination, the (average) opacity of the screen area covered by a splat is tested, if it is above a specified threshold. In this case, the respective splat will not have a significant contribution (or no contribution, if the chosen threshold specifies full opacity). The test itself is performed by convolving the associated opacity buffer area (which stores the up to now accumulated opacity of each sheet buffer) with a box filter of the screen size of the splat, resulting in the opacity average of that area [Mueller et al. 1999].

Instead of tri-/bilinear interpolation-based point sampling, splatting employs a sample average across the sampling distance in view direction. This introduces an additional low-pass filter operation, which reduces aliasing, but also tends to smooth signal characteristics. Finally, the recent splatting approaches [Mueller and Crawfis 1998; Mueller et al. 1999] also provide post-shaded DVR, in contrast to pre-shading provided by the original splatting approach.

Texture Mapping

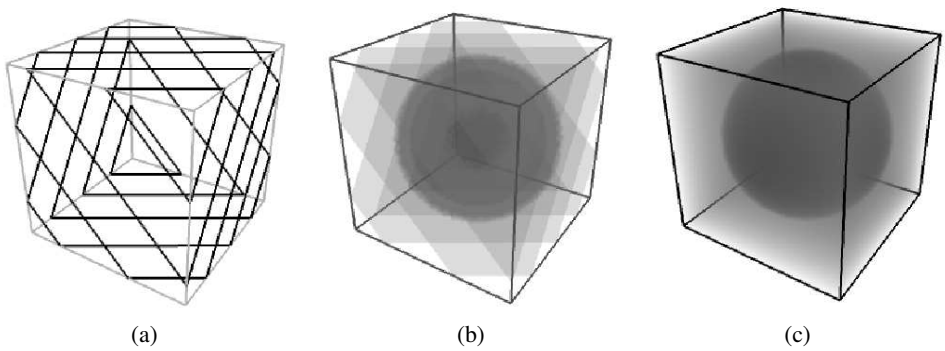


Figure 8: 3D Texture Mapping-based Volume Rendering [Westermann and Ertl 1998]: (a) 3D texture slices are generated from the volume, perpendicular to the view-plane; the texture slices are mapped onto the screen (b) and blended with the previous slice (c).

Object-space oriented texture-mapping is accumulating texture slices perpendicular to the view-plane back-to-front using the blending functionality of graphics hardware (see Fig. 8) [Cullip and Neumann 1993; Cabral et al. 1994]. The sampling of the texture slices from the volume are either trilinear, if 3D texture mapping hardware is available, or bilinear, if only 2D textures are supported. (On some graphics systems, quadlinear interpolation is also available, where an additional interpolation takes place between two mipmap-levels.)

Shading can be achieved by the computation of a pre-shaded color volume [van Gelder and Kim 1996], or by using multi-pass methods to visualize isosurfaces [Westermann and Ertl 1998; Sommer et al. 1998; Dachille et al. 1008], or transparent volumes [Meißner et al. 1999]. In

our comparison later-on, we use the color volume approach to provide high-speed volume rendering of semi-transparent volumes. However, due to the limited precision of the alpha channel of the framebuffer (only eight bits), we do not pre-multiply the voxels with the opacity which leads to color bleeding artifacts [Wittenbrink et al. 1998] (see Fig. 4).

6 Volume Data Acquisition Techniques

• Dirk Bartz, University of Tübingen, Germany

There are many sources of data for medical visualization. For our purposes, we rely on data from medical scanners, such as Computed Tomography (CT or CAT scanners), Magnetic Resonance Tomography (MRT or MR scanner), rotational biplane X-ray (or rotational angiography), 3D Ultrasound, Positive Emission Tomography (PET), and many more. These scanners produce a stack of images, where each image represents a slice from a data volume. Each entity or pixel of the images represents a sample point within the data volume and it is also called a voxel (see Section 5).

In the course of this section, we give an introduction into basic concepts of the scanning technology of the modalities of the volume datasets later used for VIVENDI. More details on the specific technology can be found in [Cho et al. 1993; Lehmann et al. 1997; Kalender 2000]. Several sources of artifacts are known; most notorious are aliasing problems due to undersampling, motion artifacts if the scanned object is moving during the scanning procedure (i.e., a heart), and *partial volume effects*, which are also related to undersampling of a structure. 3D data samples (voxels) which are reconstructed from projections of a 3D scanner are averages of the local volume. If material with a high voxel value is adjacent to material with a low voxel value, this averaging can lucidly distort these voxel values, causing the partial volume effect.

6.1 X-Ray

In classical X-ray imaging, electrons are shot at the focal spot of a specific target. The resulting radiation is absorbed (and scattered) by an object (i.e., body parts) behind the focal spot and hence the attenuated intensity is recorded on detectors behind the object. Depending on the quantity of the acceleration voltage, X-rays are denoted as hard or soft radiation. The interaction of X-rays with matter is basically determined by photo-electric absorption and by scattering (Compton scattering). With hard radiation, the scattering effect is dominating the interaction, resulting in an absorption only significant in object areas with high density (high atomic number), i.e. bones. For areas with a lower density (i.e., tissue), a sufficient absorption requires soft radiation. Today's X-ray devices feature a very high resolution (i.e., 4096×4096 pixels), but only as a 2D projection.

A variation of classical X-ray imaging is fluoroscopy. The emitted radiation is recorded with an image intensifier and displayed such that the progress of specific procedures can be observed. This permanent exposure to ionizing radiation requires a significant lower intensity than with classical X-rays. Hence, the resolution and general image quality are reduced.

6.2 Computed Tomography

The introduction of X-ray Computed Tomography (CT) in 1972 [Hounsfield 1972] provided for the first time a volumetric representation of objects, and not only a 2D projection of a volumetric object. Generally, it is seen as one of the major milestones in medical imaging [Cho et al. 1993].

As basic concept, X-rays are emitted through the object from different positions around that object (i.e., filling an angular range of 180 degree) and the intensity profile is recorded by a detector. Algorithms are applied to reconstruct an intensity attenuation layer of the object. A series of these projection with an incremental modified position generates a stack of intensity attenuation layers, which forms an intensity volume. The intensity attenuation is measured in Hounsfield units (HU), with water as reference material (0 HU). I.e., bone has a high measured intensity attenuation, while fat or air have negative HU [Lehmann et al. 1997]. Each of the reconstructed volumetric samples (voxels) represents the average of the local environment.

The major differences between the development stages of CT [Kalender 2000] are different projection reconstruction algorithms and the emitter and detector architecture (data acquisition). The first generation of CT scanners was basically the experimental setting of Hounsfield's CT scanner. It used a single pencil-like X-ray beam emitter and a single detector on the opposite side of the object. To acquire a data slice, the pencil beam is translated along the object and rotated afterwards for the next series of beams. All together, the costly mechanical movement of emitter and detector caused long scanning times, ranging from several minutes to several hours at a resolution of 80×80 pixels per scan/slice. Furthermore, the single emitter/detector architecture enabled only a poor utilization of the emitted radiation.

The next generation and first commercial generation of CT scanner used small angle fan beams and multiple detectors to scan two neighboring rotational projections at the same time. This technique reduces the number of necessary rotations, and hence the required scanning time (10 - 60 seconds, up to several minutes) needed for a sufficient reconstruction. It also provides a better utilization of the emitted radiation. Both first and second generation techniques are parallel beam devices, which use different reconstruction algorithms than the next fan beam devices.

The next improvement increased the fan beam angle and the number of detectors to cover the whole object, thus the translating movement became unnecessary and increased the scanning speed to five seconds per slice. Similar to the previous techniques, the radiation is enabled in fixed time intervals to be measured by the detectors.

In the fourth generation, the rotating detector was replaced by a fixed circular ring detector, which reduced the technical effort of moving the larger mass of emitter and detector. Here, the radiation was permanently emitted and only the detectors were enabled at certain intervals. However, several problems of ring detectors led to further developments in favor of third generation scanners. Besides the higher costs for the detector ring, specific X-ray scattering problems reduced the image quality of these systems, while collimator technology could reduce the scattering problems with a rotating emitter/detector system of the third generation.

Currently, the state-of-the-art are spiral or helical CT, where the emitter/detector system is rotating permanently around the object, while the object is moved continuously in the perpendicular direction to acquire a full data volume. This technique enables faster scanner due to

the continuous rotating movement of the emitter/detector system which saves the time for the previously needed time to accelerate and slow down these heavy parts of the scanner. Recently, multiple layers of emitters and detectors (twin or quad slices) where combined to create multi-slice CT scanners, which enable fast and isotropic scanning of large object areas.

Besides the architectural development, different volume/slice reconstruction algorithms differentiate the various systems and generations. The basic approach is the back transformation of the slice projections into the volume slices by the Radon transform [Cho et al. 1993]. In the beginning of computed tomography, algebraic reconstruction techniques (ART) were used to solve this back projection problem. However, the high computational costs of iterative solving the large matrices – the size of a reconstruction matrix is equal to the resolution of the slice – rendered this approach as not usable for standard applications [Kalender 2000]. The standard method today is filtered back-projection, where each projection is composed according to the measured direction. Parallel and fan beam methods are available to perform this back-projection. However, the current fan beam methods are more complex and less efficient than state-of-the-art parallel beam reconstruction algorithms. Hence, the projections of today's fan-beam scanners are re-sorted in parallel beams before the actual reconstruction. Another modification is required to address the continuously moving object tray of modern spiral CT scanners, where a z-interpolation corrects the measured projections according to the tray movement [Kalender 2000]. In the future, cone beam reconstruction algorithms will probably replace the current methods [Kalender 2000], which are already successfully used in rotational biplane X-ray (see below).

6.3 Rotational Biplane X-Ray

Rotational biplane X-ray is a recent scanning technology which started from digital subtraction angiography (DSA – in DSA, X-ray images are recorded before (mask) and after (filling) the injection of a contrast agent. The mask is subtracted from the filling image, thus presenting only the contrast agent enhanced object areas). Hence, most of the associated applications are angiography applications and this technique is frequently referred to as rotational angiography [Fahrig 1999; Gürvit et al. 2000].

To generate volumetric datasets, a series of up to 132 X-ray projections are taken from a rotation range of 200 degrees around the scanning object. In contrast to CT, rotational biplane X-ray is using a full array of up to 1024^2 detectors which enable the measurement of a full cone of rays. To account for reconstruction errors [Kalender 2000], a modified back-projection algorithm is used [Hornegger 2000]. Additionally, special filter kernels are used which further reduces potential artifacts. Current rotational angiography systems provides very high resolution, isotropic datasets, good reconstruction quality, and a high data acquisition speed of up to 13 seconds for a full scan.

6.4 Magnetic Resonance Imaging

The beginning of Magnetic Resonance Imaging (MRI) dates back to the early seventies, but it was not adopted into medical use until the nineteen-eighties. It is based on the nuclear resonance of hydrogen in a magnetic field, where each of the hydrogen nuclei can be considered as a small dipole magnet which aligns itself either parallel or anti-parallel along the magnetic field.

While aligned in that field, the protons (which are identical to the hydrogen nucleus) spin arbitrarily around the axis of the field. This spinning is called the *precession* of the nuclei. If energy is applied to the magnetic field as a radio-frequency pulse (RF) at the *Larmor*-frequency, the nuclear resonance forces the protons to receive some of the energy from the RF. This pulse also forces all the protons to spin synchronously, or *in phase* and flip increasingly into the anti-parallel orientation of higher energy, until the number of parallel protons is equal to the number of anti-parallel oriented protons. The duration of the RF pulse determines the amount of precession; i.e., a *90 degree pulse* will force the protons into a 90 degree precession, where the precession vector of the protons is completely perpendicular to the magnetic field, resulting in a zero z-component (along the magnetic field). After the stimulation of the protons, they slowly release the received energy, *de-phase* and re-align with the magnetic field. This *relaxation* is described as *free induction decay (FID)* and is divided into the transverse and longitudinal relaxation. The first relaxation of the transverse magnetization – also called spin-spin relaxation – describes the de-phasing of the x/y-component of the precession. The time required for this relaxation is called T_2 and is in the order of a few milliseconds. The longitudinal, or spin-lattice relaxation describes the re-alignment of the precession with the magnetic field, thus the restoring of the z-component. This relaxation time is called T_1 and is in the order of seconds. The actual measured volumetric information is the proton density σ which needs to be reconstructed at the specific voxels.

To reconstruct the spatial information of the measured signal, two additional gradient magnetic fields are applied. The first field selects the slice in z-direction of the volume, since only one layer of protons suffices the Larmor-frequency for the main and gradient magnetic fields. An additional gradient field in x-direction selects a y-slab. With a 2D Fourier reconstruction, the x/y-coordinate is encoded into that signal [Lehmann et al. 1997] by increasing de-phasing signal of the transverse relaxation along the x and y directions, which generate specific frequencies into the signal. Thus, the RF intensity (proton/spin intensity) is encoded into the intensity of the signal, while the position is encoded into the frequencies [Cho et al. 1993; Lehmann et al. 1997].

Different protocols describe sequences of various RF pulses and the actual measurement of the signal, where the time between the initial stimulation and the measurement is called *echo time* T_E and the time between two (initial) stimulation cycles is called *repetition time* T_R. By varying T_E and T_R, different weight data can be achieved; i.e., with short echo and repetition times, the proton density is mostly dominated by the T_1 relaxation time (T_1 weighted), with long T_E and T_R, the T_2 relaxation time is dominating the signal (T_2 weighted), or with a long T_R and a short T_E, the resulting data is neither T_1 nor T_2 and can be seen as "*the pure (proton) density function*" [Cho et al. 1993]. Scanning protocols such as Turbo Spin Echo (TSE) or Constructive Interference in Steady States (3D CISS) belong to these categories. A slightly different technique is used for angiography sequences, which focus on flow information. Specifically, the Time of Flight sequence (TOF) measures the spin saturation of the protons; the repeated stimulation of the protons in the data volume leads to a saturation of the signal of stationary samples. However, the particle flow in blood vessels are continuously introducing "fresh material" into the magnetic field which gives a strong signal. Typical MRI scans take between 2 and 25 minutes. This relatively long scanning time is dominated by the relaxation of the spins, not by the time required for the measurements.

Appendix: Virtual Endoscopy Resources

This text is also contained as html-file on the conference DVD.

Books

There are not many books on virtual endoscopy. If you are only interested in the clinical aspect, but not at all in the technical aspects, the book by Patrick Rogalla can be helpful:

P. Rogalla (ed.): Virtual Endoscopy and Related 3D Techniques. Springer Verlag 2001.

Other books (in particular books for medical students) sometimes briefly touch the subject, but don't go deeper into the subject. My thesis contains some related information, but the relevant parts are included in the tutorial notes or STAR report anyway.

Web Resources

In contrast, there are plenty of web pages related to the topic of virtual endoscopy. This list however, contains only a small subset of it. Try web search engines for more information on the web. Also on the conference DVD, we will add a (subjective) bibliography of related papers, mostly of our groups, with weblinks to the respective papers.

- **http://www.gris.uni-tuebingen.de/ bartz/tutorials/vis2003**: Course home page
- **http://www.gris.uni-tuebingen.de/ bartz/endoscopy.html**: Virtual Endoscopy Project, University of Tübingen (VIVENDI Project)
- **http://www.cs.sunysb.edu/ vislab/**: Center for Visual Computing, SUNY Stony Brook (VICON Project)
- **http://www.vec.wfubmc.edu**: Virtual Endoscopy Center, University of Wake Forest (FreeFlight Project)
- **http://3dradiology.org**: Medical Imaging Lab, Stanford University
- **http://splweb.bwh.harvard.edu:8000**: Surgical Planing Lab, Brigham and Women's Hospital
- **http://cobb.ee.psu.edu/users/krishnan/quicksee/quicksee.html**: Multi-Dimensional Image Processing Lab, Pennsylvania State University
- **http://www.rad.unipi.it/virtual/VE-forum/**: Virtual Endoscopy Forum
- **http://www.cg.tuwien.ac.at/research/vis/vismed/**: TU Vienna/TIANI Virtual Endoscopy Projects

Medical Imaging Links

- **http://www.kitware.com/vtk**: Visualization Toolkit (VTK)
- **http://www.itk.org**: National Library of Medicine's Insight Segmentation and Registration Toolkit (ITK)
- **http://www.image-registration.com**: CISG registration of King's College, London
- **http://dicom.offis.de/index.php.en**: OFFIS-DICOM toolkit

Related Medical Imaging Companies

- **http://www.viatronix.com**: Viatronix
- **http://www.vitalimages.com**: Vital Images
- **http://www.siemensmedical.com**: Siemens Medical Solutions
- **http://www.medical.philips.com**: Philips Medical Systems
- **http://www.gemedicalsystems.com**: GE Medical Systems
- **http://www.tiani.com**: TIANI

Visit also the volume dataset repository at **http://www.volvis.org** www.volvis.org.

References

BAJAJ, C., PASCUCCI, V., AND SCHIKORE, D. 1996. Fast Isocontouring for Improved Interactivity. In *Proc. of Symposium on Volume Visualization*, 39–46.

BENTLEY, J. 1975. Multidimensional Binary Search Trees Used for Associative Search. *Communications of the ACM 18*, 9, 509–516.

CABRAL, B., CAM, N., AND FORAN, J. 1994. Accelerated Volume Rendering and Tomographic Reconstruction Using Texture Mapping Hardware. In *Proc. of Symposium on Volume Visualization*, 91–98.

CHO, Z., JONES, J., AND SINGH, M. 1993. *Foundations of Medical Imaging*. John Wiley, New York, NY.

CIGNONI, P., PARINO, P., MONTANI, E., PUPPO, E., AND SCOPIGNO, R. 1997. Speeding Up Isosurface Extraction Using Interval Trees. *IEEE Transactions on Visualization and Computer Graphics 2*, 12, 158–170.

CULLIP, T., AND NEUMANN, U. 1993. Accelerating Volume Reconstruction with 3D Texture Hardware. Tech. Rep. TR93-027, University of North Carolina at Chapel Hill.

DACHILLE, F., KREEGER, K., CHEN, B., BITTER, I., AND KAUFMAN, A. 1008. High Quality Volume Rendering Using Texture Mapping Hardware. In *Proc. of Eurographics/SIGGRAPH Workshop on Graphics Hardware*, 69–76.

FAHRIG, R. 1999. *Computed Rotational Angiography*. PhD thesis, University of Western Ontario.

FOLEY, J., VAN DAM, A., FEINER, S., AND HUGHES, J. 1996. *Computer Graphics: Principles and Practice*, 2nd ed. Addison Wesley, Reading, MA.

FUCHS, H., KEDEM, Z., AND NAYLOR, B. 1980. On Visible Surface Generation by a Priori Tree Structures. In *Proc. of ACM SIGGRAPH*, 124–133.

GÜRVIT, O., SKALEJ, M., RIEKMANN, R., ERNEMANN, U., AND VOIGT, K. 2000. Rotational Angiography and 3D Reconstruction in Neuroradiology. *electro medica 68*, 1, 31–37.

HEGE, H., HÖLLERER, T., AND STALLING, D. 1993. Volume Rendering – Mathematical Foundations and Algorithmic Aspects. Tech. Rep. TR 93-7, Konrad-Zuse Zentrum für Informationstechnik Berlin.

HÖHNE, H., AND BERNSTEIN, R. 1986. Shading 3D-Images from CT using Gray-level Gradients. In *IEEE Transactions on Medical Imaging*, vol. MI-5, 45–47.

HORNEGGER, J., 2000. Rotational Angiography. Siemens Medical Systems, Personal Communications.

HOUNSFIELD, G., 1972. A Method of and Apperatus for Examination of a Body by Radiation such as X-ray or Gamma Radiation. British Patent No. 1283915.

ITOH, T., AND KOYAMADA, K. 1995. Automatic Isosurface Propagation Using an Extrema Graph and Sorted Boundary Cell Lists. *IEEE Transactions on Visualization and Computer Graphics 1*, 4, 319–327.

KALENDER, W. 2000. *Computer Tomography*. PUBLICIS MCD Verlag, München, Germany.

KAUFMAN, A., AVILA, R., SOBIERAJSKI, L., AND YAGEL, R. 1996. Volume Visualization Algorithms and Applications. In *IEEE Visualization, tutorial 1*.

LACROUTE, P., AND LEVOY, M. 1994. Fast Volume Rendering Using a Shear-Warp Factorization of the Viewing Transformation. In *Proc. of ACM SIGGRAPH*, 451–458.

LEHMANN, T., OBERSCHELP, W., PELIKAN, E., AND REPGES, R. 1997. *Bildverarbeitung für die Medizin: Grundlagen, Modelle, Methoden, Anwendungen*. Springer Verlag, Heidelberg, Germany.

LEVOY, M. 1988. Display of Surfaces from Volume Data. *IEEE Computer Graphics and Applications 8*, 3, 29–37.

LICHTENBELT, B., CRANE, R., AND NAQVI, S. 1998. *Introduction into Volume Rendering*. Prentice Hall, Upper Saddle River, NJ.

LIVNAT, Y., SHEN, H., AND JOHNSON, C. 1996. A Near Optimal Isosurface Extraction Algorithm Using the Span Space. *IEEE Transactions on Visualization and Computer Graphics 1*, 2, 73–84.

LORENSEN, W., AND CLINE, H. 1987. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. In *Proc. of ACM SIGGRAPH*, 163–169.

MAX, N. 1995. Optical Models for Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics 1*, 2, 99–108.

MEISSNER, M., HOFFMAN, U., AND STRASSER, W. 1999. Enabling Classification and Shading for 3D Texture Mapping Based Volume Rendering. In *Proc. of IEEE Visualization*, 207–214.

MUELLER, K., AND CRAWFIS, R. 1998. Elminating Popping Artifacts in Sheet Buffer-Based Splatting. In *Proc. of IEEE Visualization*, 239–246.

MUELLER, K., SHAREEF, N., HUANG, J., AND CRAWFIS, R. 1999. High-quality Splatting on Rectilinear Grids with Efficient Culling of Occluded Voxels. *IEEE Transactions on Visualization and Computer Graphics 5*, 2, 116–134.

PORTER, T., AND DUFF, T. 1984. Compositing Digital Images. In *Proc. of ACM SIGGRAPH*, 253–259.

SABELLA, P. 1988. A Rendering Algorithm for Visualizing 3D Scalar Fields. In *Proc. of ACM SIGGRAPH*, 51–58.

SAMET, H. 1994. *The Design and Analysis of Spatial Data Structures*. Addison-Wesley, Reading, MA.

SCHROEDER, W., MARTIN, K., AND LORENSEN, B. 1998. *The Visualization Toolkit*, 2nd ed. Prentice Hall, Upper Saddle River, NJ.

SHEN, H., HANSEN, C., LIVNAT, Y., AND JOHNSON, C. 1996. Isosurfacing in Span Space with Utmost Efficiency (ISSUE). In *Proc. of IEEE Visualization*, 287–294.

SOMMER, O., DIETZ, A., WESTERMANN, R., AND ERTL, T. 1998. TIVoR: An Interactive Visualization and Navigation Tool for Medical Volume Data. In *Proc. of WSCG*, 361–372.

TUY, H., AND TUY, L. 1984. Direct 2D Display of 3D Objects. *IEEE Computer Graphics and Applications 4*, 10, 29–33.

VAN GELDER, A., AND KIM, K. 1996. Direct Volume Rendering with Shading via Three-dimensional Textures. In *Proc. of Symposium on Volume Visualization*, 23–30.

WESTERMANN, R., AND ERTL, T. 1998. Efficiently Using Graphics Hardware in Volume Rendering Applications. In *Proc. of ACM SIGGRAPH*, 169–177.

WESTOVER, L. 1990. Footprint Evaluation for Volume Rendering. In *Proc. of ACM SIGGRAPH*, 367–376.

WESTOVER, L. 1991. *SPLATTING: A Parallel Feed-Forward Volume Rendering Algorithm*. PhD thesis, University of North Carolina at Chapel Hill.

WILHELMS, J., AND VAN GELDER, A. 1992. Octrees for Faster Isosurface Generation. *ACM Transactions on Graphics 11*, 3, 201–227.

WILHELMS, J., AND VAN GELDERN, A. 1991. A Coherent Projection Approach for Direct Volume Rendering. In *Proc. of ACM SIGGRAPH*, 275–284.

WITTENBRINK, C., MALZBENDER, T., AND GOSS, M. 1998. Opacity-weighted Color Interpolation for Volume Sampling. In *Proc. of Symposium on Volume Visualization*, 135–142.

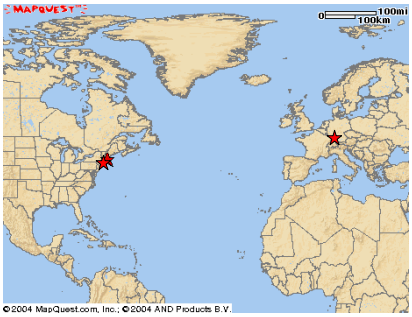
Advanced Virtual Medicine: Techniques and Applications for Medicine-Oriented Visualization

Introduction

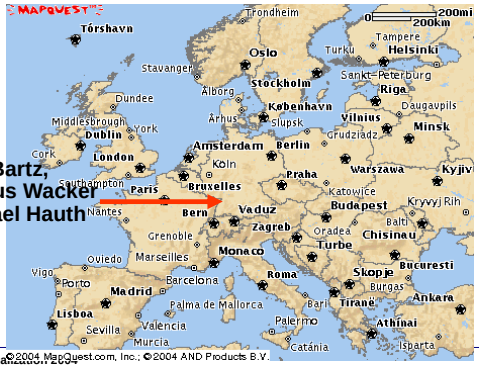
Yin Wu — TeraRecon — wu@terarecon.com

Dirk Bartz, — WSI/GRIS – VCM, University of Tübingen — bartz@gris.uni-tuebingen.de

Cross-Atlantic Speakers



From Europe



Dirk Bartz,
Markus Wacker
Michael Hauth

University of Tübingen (1)

Dirk Bartz

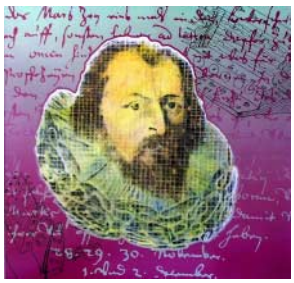
Head of VCM,
Univ. of Tübingen
Tübingen,
Germany

Markus Wacker

Michael Hauth

Univ. of Tübingen
Tübingen,
Germany

University of Tübingen (2)



Wilhelm Schickard,
1592 - 1635

Prof. of Orientalic Languages, Astronomy, Mathematics, and Geodesy
at University of Tübingen

1623 invention of first calculation machine

(Blaise Pascal's Pascaline 1642)

University of Tübingen (3)

Dirk Bartz

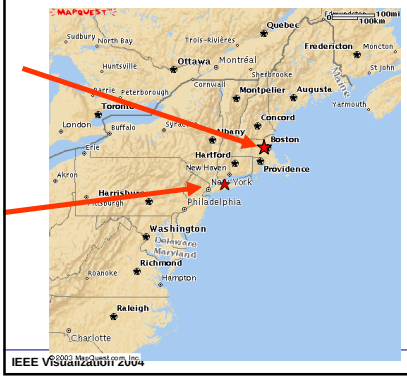
Head of VCM,
Univ. of Tübingen
Tübingen,
Germany

Michael Hauth

Univ. of Tübingen
Tübingen,
Germany

University of Tübingen (4)

USA Speakers (1)



TeraRecon (1)

Yin Wu,

Senior Scientist,

TeraRecon
Concord, MA

Jan Hardenbergh
,

TeraRecon
Concord, MA

TeraRecon (2)

Yin Wu,

Senior Scientist,
TeraRecon
Concord, MA

Stony Brook University (1)

Klaus Müller

Assist. Professor,
Stony Brook
University,
Stony Brook, NY

Stony Brook University (2)

Outline

Introduction (Bartz, Wu)

Basics of Medical Imaging
- Data Acquisition (Bartz, Hardenbergh)
- Data Pre-Processing (Bartz)
- Visualization and Navigation Techniques (Mueller, Wu)

Applications of Virtual Medicine (Bartz, Wu)

fMRI for Surgical Planning (Hardenbergh)

Soft-Tissue-Simulation and Deformation (Wacker, Hauth)

Questions and Answers (All)

Additional Tutorial Notes

The conference DVD does contain additional material:

- Textual tutorial notes on data acquisition and volumetric datasets
- Virtual endoscopy state-of-the-art report of Eurographics 2003
- Links to related websites and papers (to avoid copyright clashes …)
- and the slides, of course

Tutorial Note Update

New and updated material will be available at

http://www.gris.uni-tuebingen.de/~bartz/tutorials/vis2004

Prerequisites

- Basic 3D graphics (polygons, triangles, shading, ...)
- (Very) basic medical imaging knowledge
- Scientific visualization helps also

Introduction (1)

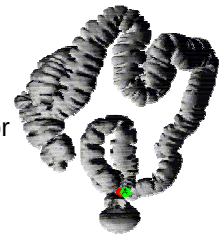
Virtual Endoscopy Applications:

- Colonoscopy
- Ventriculoscopy
- Angioscopy
- Bronchoscopy
- ....

Virtual Colonoscopy (1)

- Colorectal cancer has high incidence: **150,000 cases/yr**
- Early detection is critical for successful treatment

→ Purpose: Screening/Diagnosis



rectum

Virtual Colonoscopy (2)

Clinical screening procedures:

- optical colonoscopy
- barium enema

Alternative screening procedure

- virtual colonoscopy

Virtual Colonoscopy (3)

Optical Colonoscopy:

+ shape and texture information
+ physical tissue interaction
- low patient compliance due to uncomfortable procedure
- no access to areas behind colon collapses
- potential risk of lesions due to endoscope
- expensive (approx. 1h procedure time)

Virtual Colonoscopy (4)

Barium Enema

+ relatively simple procedure
+ relatively cheap
- low sensitivity (structures < 10mm)
- depends on experience of radiologist
- radiation exposure (although not very high)

Virtual Colonoscopy (5)

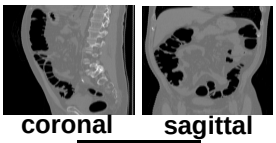
Virtual Colonoscopy:

+ same patient preparation as optical
+ cheaper than optical endoscopy
+ can inspect areas behind collapses
- no texture information
- no biopsy / removal possible
- quality depends on segmentation

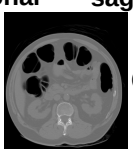
Virtual Colonoscopy (6)

Protocol:

- Cleanse colon
- Pump air into colon
- CT-scan abdomen
- **Reconstruct & visualize colonic surface**
- If polyps are found, do optical Colonoscopy



coronal sagittal



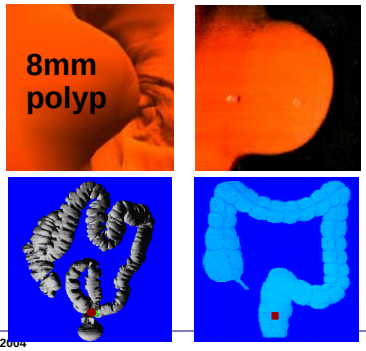
CT

transverse

Different modalities possible

Virtual Colonoscopy (7)

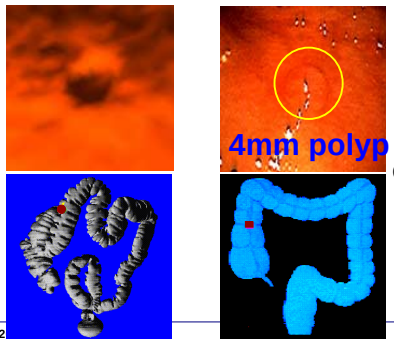
virtual



optical

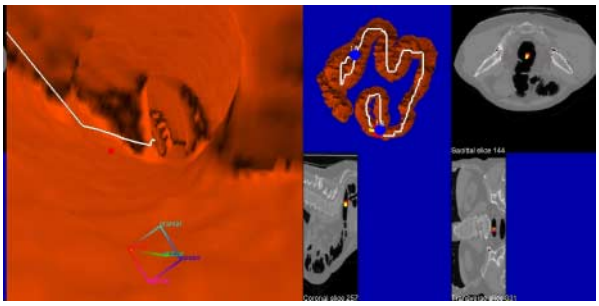
Virtual Colonoscopy (8)

virtual



optical

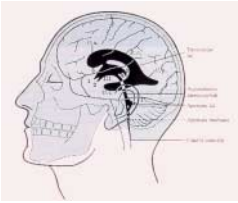
Virtual Colonoscopy (9)



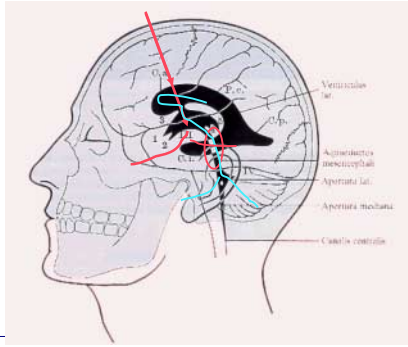
Virtual Ventriculoscopy (1)

- Ventricular System (VS) produces CSF (brain fluid)
- Occlusion in VS causes hydrocephalus
- Requires new drain

➔ Purpose:
Diagnosis and planning



Virtual Ventriculoscopy (1)



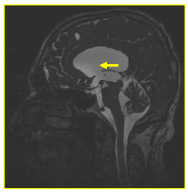
Virtual Ventriculoscopy (2)

Procedures:

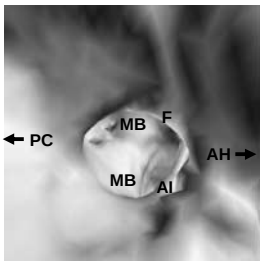
- placement of stent
  - requires replacement after a few years
  - causes degenerations of VS
- ventriculoscopy:
  - \+ no implantation required
  - \- complex procedure

Virtual Ventriculoscopy (3)

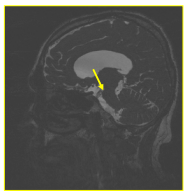
Left Lateral Ventricle, approach from Posterior Horn:
AH - Anterior Horn, CP - Choroid Plexus, CPV - Choroid Plexus Vein



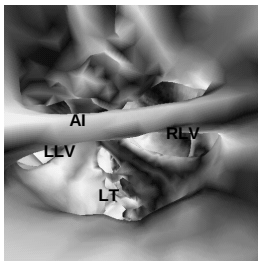
Virtual Ventriculoscopy (4)



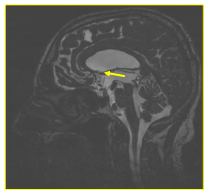
Foramen of Monro, approach from Right Ventricle: PC - Pars Centralis, AH - Anterior Horn, F - Fornix, AI - Adhesio Interthalamica, MB - Mamillary Bodies



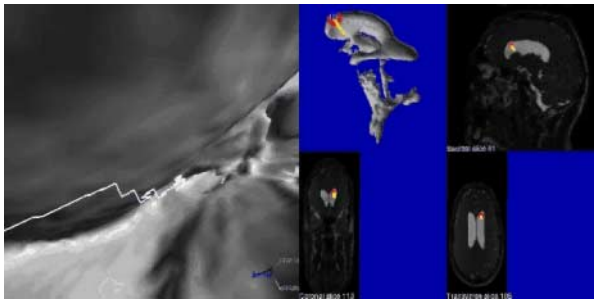
Virtual Ventriculoscopy (5)



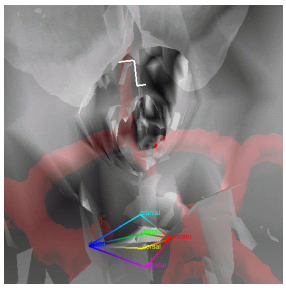
Foramen of Monro, approach from Third Ventricle: AI - Adhesio Interthalamica, LLV - entrance to Left Ventricle, RLV - entrance to Right Ventricle, LT - Lamina Terminalis



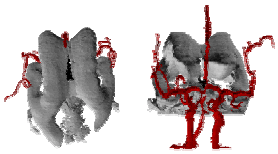
Virtual Ventriculoscopy (6)



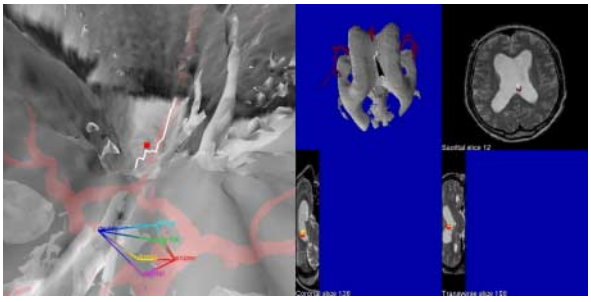
Virtual Ventriculoscopy (7)



Multi-modal visualization of **ventricular system** and **angio-architecture**

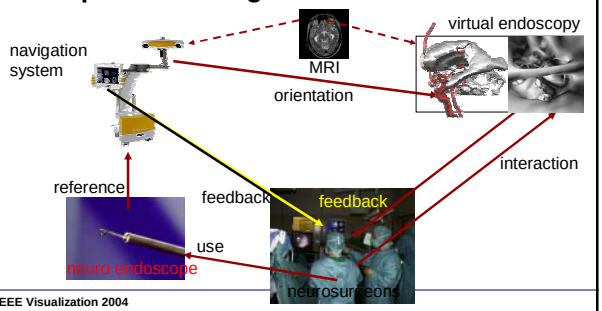


Virtual Ventriculoscopy (8)



Virtual Ventriculoscopy (9)

Intraoperative navigation



Virtual Angioscopy (1)

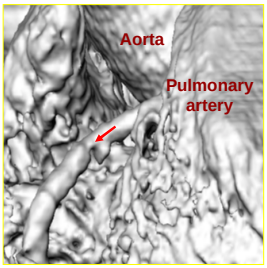
Among leading medical problems are vascular diseases:

- stenosis/occlusion
- aneurysms

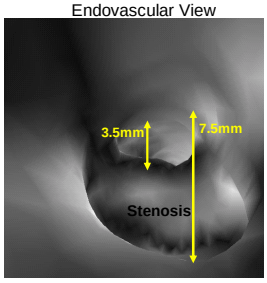
➔ Purpose:
Diagnosis and Planning

Virtual Angioscopy (2)

Exterior View



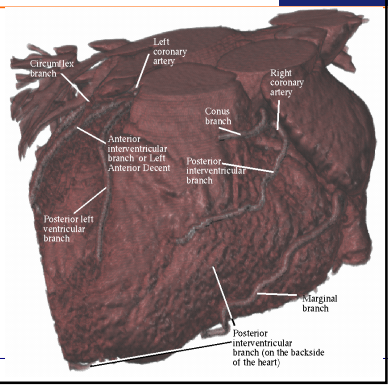
Endovascular View



Stenosis of Right Coronary Artery

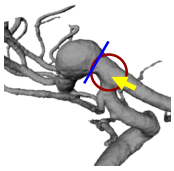
Virtual Angioscopy (3)

Coronary arteries of the heart

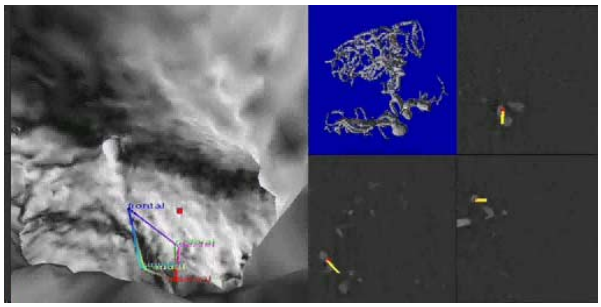


Virtual Angioscopy (4)

Aneurysm of cerebral artery



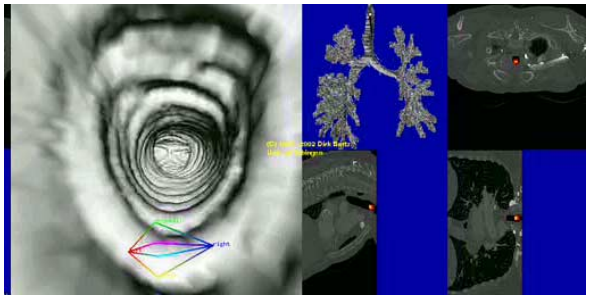
Virtual Angioscopy (5)



Virtual Bronchoscopy (1)

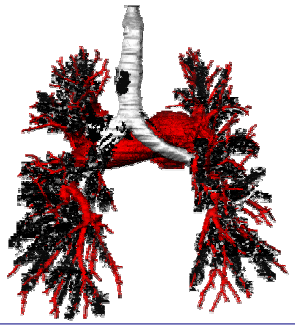
- Multi-Slice CT
- Model-based Segmentation (Univ. Mainz)
- More complex segmentation
  - ➔ More complex reconstruction
- Purpose: Diagnosis and Planning

Virtual Bronchoscopy (2)



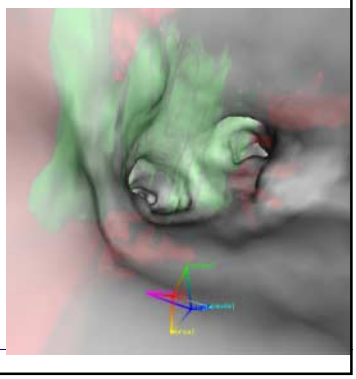
Virtual Bronchoscopy (3)

Bronchial tree and pulmonary arteries

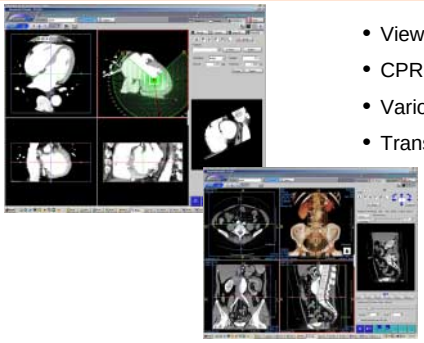


Virtual Bronchoscopy (4)

Bronchial tree with pulmonary arteries and tumor



Interactive Medical Visualization



- Viewing 2D, 3D, 4D
- CPR, cut, masking
- Various templates
- Transfer functions

Cardiac Time Volume Analysis

Cardiac Application:

- Automatic contour detection of LV endocardial and epicardial surfaces
- Manual correction of the surfaces.
- Calculation of myocardial volume (and mass).
- Regional wall thickness, thickening, and ejection fraction can be displayed on polar maps and graphs.
- Time-volume graph, calculation of peak ejection, peak filling rate and their corresponding time.
- 3D display of endocardium and epicardium, with color coded regional indexes on the surface.

Cardiac Time Volume Analysis

Cardiac Application:

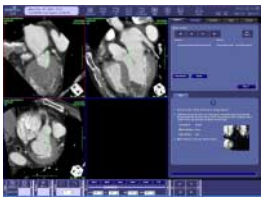
- Automatic contour detection of LV endocardial and epicardial surfaces
- Manual correction of the surfaces.
- Calculation of myocardial volume (and mass).
- Regional wall thickness, thickening, and ejection fraction can be displayed on polar maps and graphs.
- Time-volume graph, calculation of peak ejection, peak filling rate and their corresponding time.
- 3D display of endocardium and epicardium, with color coded regional indexes on the surface.

IEEE Visualization 2004

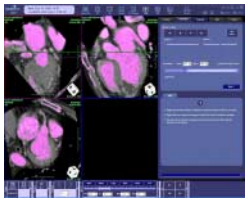
Cardiac Time Volume Analysis

Steps:

Position

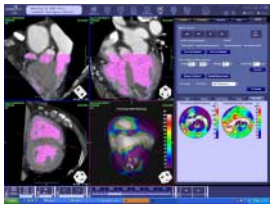


Threshold



Cardiac Time Volume Analysis

Results:



Summary

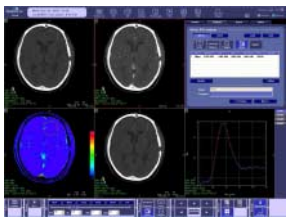
- Visualization 4D Volume
- Detect 2D-3D Boundary Recursively
- Trace the Boundary During Deformation
- Calculate the Volume/Time Curve
- Visualize Heart Wall Thickness

Time Density Analysis

- Study of time-dependence of contrast enhancement

Example:

Brain Tumor

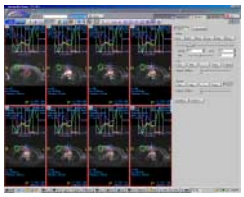


Contrast enhancement of brain tissue due to perfusion with contrast enhanced blood

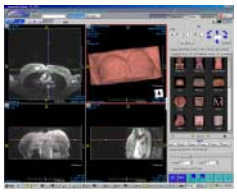
Time Density Analysis

Example:

Breast cancer



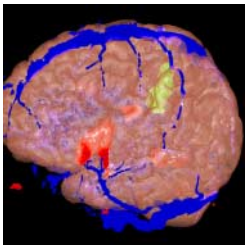
Detect the cancer according to the time density curve



3D View

Functional MRI

MRI for Functional areas of the brain:



for example
left motor
right motor
language acquisition
etc.

Advanced Virtual Medicine: Techniques and Applications for Medicine-Oriented Visualization

Foundations

Dirk Bartz

WSI/GRIS – VCM, University of Tübingen

bartz@gris.uni-tuebingen.de

Outline

Introduction

Foundations

- Medical Imaging Techniques
- Data Pre-Processing
- Visualization and Navigation Techniques

Applications of Virtual Medicine

Motion Simulation

Soft-Tissue Simulation /Deformation

Question and Answers

Foundations

Data Acquisition
using medical imaging techniques

Pre-Processing
segmentation, classification, etc.

Exploration
using visualization and navigation techniques

Foundations

Data Acquisition
using medical imaging techniques

Pre-Processing
segmentation, classification, etc.

Exploration
using visualization and navigation techniques

Medical Imaging Techniques (1)

© Siemens Medical Solutions

X-Ray

2D projection images based on absorption and scattering

- Very high resolution
- Bone/tissue contrast by selectin hard/soft radiation
- Only 2D

Medical Imaging Techniques (1)

X-Ray

2D projection images based on absorption and scattering

- Very high resolution
- Bone/tissue contrast by selectin hard/soft radiation
- Only 2D

Hand of wife of C. Röntgen

Medical Imaging Techniques (2)

© Siemens Medical Solutions

Rotational Angiography

3D volume is reconstructed from **series of X-ray** scans

- Very high resolution
- Isotropic spacing (reduces artifacts)

Medical Imaging Techniques (3)

© Siemens Medical Solutions

Computed Tomography (CT)

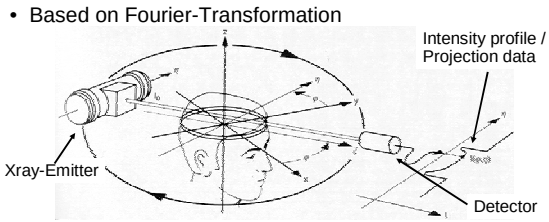
3D volume is reconstructed from X-ray projections
(Spiral CT, Multi-Slice CT 4/16)

- Fast image acquisition
- High resolution
- Different reconstruction approaches

Medical Imaging Techniques (4)

Computer Tomography (CT):

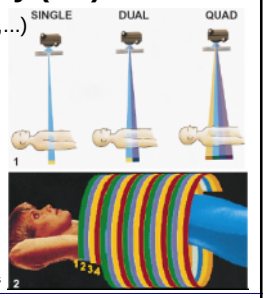
- Radon-Transformation reconstructs images from projection data/-profiles
- Based on Fourier-Transformation



Medical Imaging Techniques (5)

Computer Tomography (CT):

- Spiral- und Multi-Slice CT (4,16,...)
- Cone-beam reconstruction
- Flat panel detector: 256 slices
- Pros:
  - Better radiation usage
  - Faster
  - Higher resolution
- But more expensive



© Philips Medical Systems

Medical Imaging Techniques (6)

Computer Tomography (CT):

First images from Hounsfield

Abdomen CT

Medical Imaging Techniques (7)

© Siemens Medical Solutions

Magnetic Resonance Imaging (MRI)

3D volume is reconstructed from measured proton spin
(1.5T, 3T, ...)

- Relatively slow image acquisitio
- Resolution depends on magneti field strength
- Many different protocols

Medical Imaging Techniques (8)

Functional MRI (fMRI)

- Blood flow increases to active regions of the brain – saturates it with oxygen.
- Deoxyhemoglobin is paramagnetic (no O_2) and can be imaged with fMRI.
- While in scanner, subject exercises mental functions.
- This is extremely useful in neurosurgical planning.

Medical Imaging Techniques (9)

Functional MRI (fMRI)

- Reconstructed scanning dataset includes volumes of
  - Anatomy of brain
  - Vasculature (blood vessels)
  - Each volume for an activation area

Medical Imaging Techniques (8)

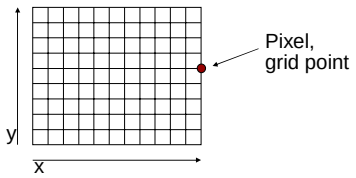
Many other imaging techniques:

- (3D) Ultrasound
- Positron Emission Tomography (PET)
- SPECT
- ....

Medical Imaging Techniques (9)

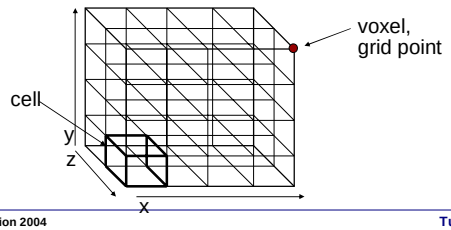
Volume data / stack of images:

- Images are composed of image elements
  → **pixel** (picture element)



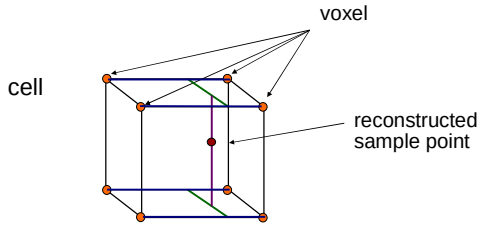
Medical Imaging Techniques (10)

- Volumes are composed of a stack of images (image stack).
  Volume elements are called voxels.



Medical Imaging Techniques (11)

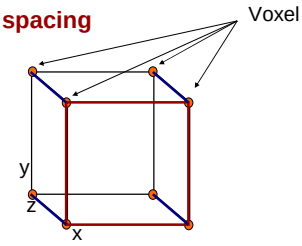
Trilinear volume interpolation:



Medical Imaging Techniques (12)

Keywords in this context:

- **Volume cell** or simply cell
- Voxel distance or **voxel spacing**
  - **Pixel distance** (x/y) - Distances within a slice
  - **Slice distance** (z) – Distance between slices



Medical Imaging Techniques (13)

Limitations of volume data
- Aliasing problems

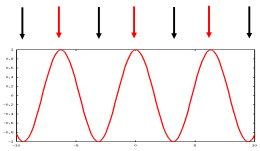
Most image/volume artifacts can be traced back to

- violating the **sampling theorem**, or
- **partial volume effects**
- **interpolation artifacts**

Medical Imaging Techniques (14)

Sampling Theorem (Nyquist, Shannon):

The proper reconstruction of a signal requires a sampling of **at least two times** as fast (frequency) as the signal (**Nyquist - Rate**)



1x +1x => 2x

Medical Imaging Techniques (15)

Sampling Theorem:

- Sampling rate at least twice as high
- Better three times higher

Medical Imaging Techniques (16)

Partial Volume Effects:

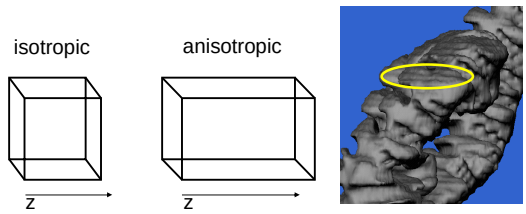
- Basically also due to **undersampling** (at volume reconstruction)
- Large **intensity difference** between neighboring materials
- Sampling does not reflect high frequencies

➔ **Material interface artifacts** (ie., holes, false connections) due to inherent

Medical Imaging Techniques (17)

Stair case artifacts

- Interpolation problem in anisotropic datasets



Foundations

Data Acquisition
using medical imaging techniques

Pre-Processing
segmentation, classification, etc.

Exploration
using visualization and navigation techniques

Data Pre-Processing

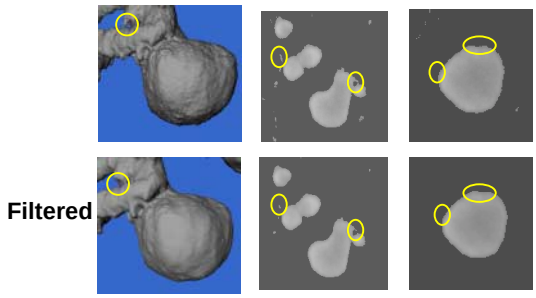
Several pre-processing operations:

- Filtering/smoothing of data
- Segmentation of structures of interest (ie., organs)
- Classification - rendering parameters
- Registration of dataset with environment
- Fusion of multiple datasets of different origins (multi-modal representations)

Filtering (1)

- Volume data can be noisy
  - → low-pass filter to remove/reduce noise
- Data looses accuracy
- Small features which disappear might be below Nyquist rate
- Careful filter design

Filtering (2)



Segmentation (1)

Problem: Structures easily detected by the human eye are **difficult to specify** for a computer

- **Many different segmentation** approaches and variations available
- **Specific image acquisition** protocols can ease segmentation difficulties

Segmentation (2)

- Automatic segmentation frequently segments **too much**, or **not all** structures
- Manual segmentation is usually **too expensive** for daily practice (ie., visible human datasets)
- Semi-automatic segmentation with little interaction only:
  can consist of **several steps**

Check out: http://www.itk.org

Segmentation (3)

Typical (and possible most used) semi-automatic segmentation is 3D Region growing:

- specify **seed point** inside structure of interest
- specify **threshold interval** which describes material interfaces
- successively **selects neighboring voxels** until threshold interval is violated

Segmentation (4)

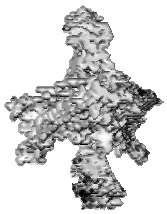
Potential problems of 3D region growing:

- Inappropriate threshold interval
- False/missing connections due to **partial volume effect** or **signal attenuation**
- Resolution too low
- Contrast too low;
  good contrast: feature intensity high,
  surrounding intensity low

Segmentation (5)

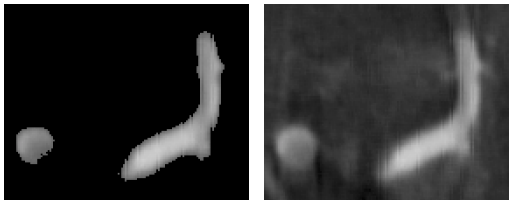
Binary segmentation can result in bumpy appearance due to interpolation artifacts (similar to staircasing)

➔ **Add boundary** to segmentation



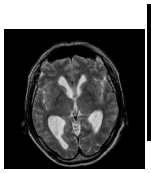
Segmentation (6)

Which is the correct threshold interval?

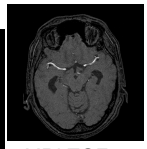


Segmentation (7)

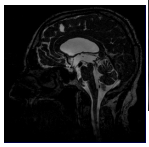
Examples of good contrast:



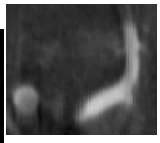
MRI TSE: Fluid filled cavities



MRI TOF: blood vessels



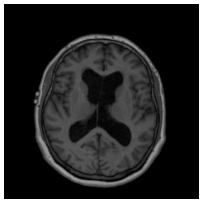
MRI 3D CISS: Fluid filled cavities



Rot. Angiography contrast agent enhanced cavities

Segmentation (8)

Examples of insufficient contrast:



Differentiation ventricles / empty space

Differentiation corpus callosum / brain tissue

MRI Flash/T1

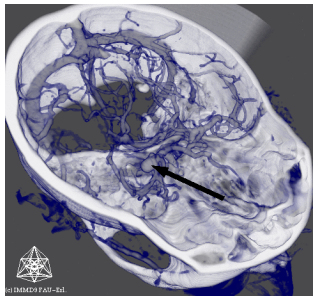
Segmentation (9)

CT Angiography:
- **good bone contrast**
- **good angio contrast**
- **poor contrast of ventricles (noisy surfaces)**

Segmentation (9)

CT Angiography:
- **good bone contrast**
- **good angio contrast**
- **poor contrast of ventricles (noisy surfaces)**

Hastreiter et al., Univ. Erlangen-Nürnberg



Segmentation (10)

3D Scanner data are usually **12-16 bits**, while volume datasets / display provide often only **8 bits**

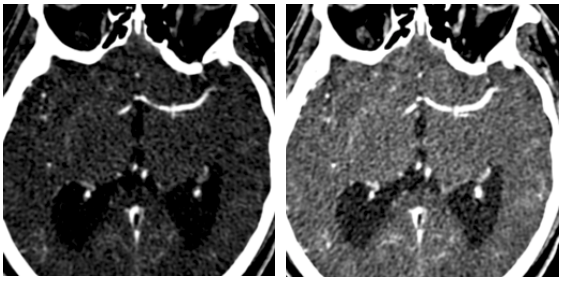
➔ requires **windowing**:

- select **sub range** of data

- **down sample** data

- inappropriate window can ruin contrast

Segmentation (11)



Different window ranges

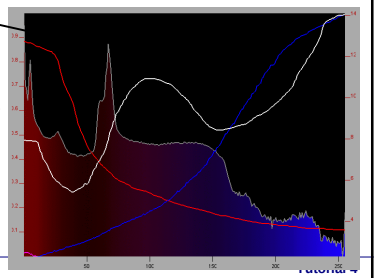
Classification (1)

- Classification specify **how data is rendered** (for direct volume rendering)
- Often **confused with segmentation**
- Are expressed by **transfer functions**
- Are usually based on **histogram**: every possible scalar value is assigned to a tuple of color and opacity
- Focuses on **material interfaces**
- May introduce **high frequencies**

Classification (2)

Example of transfer function

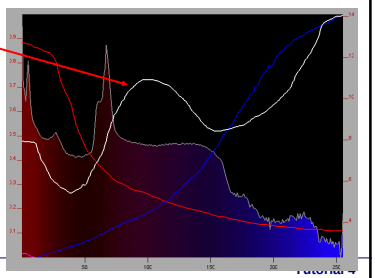
- Histogram:
- **Rising or falling edges indicate material interfaces**
- **Gradient magnitude can be also provide useful information**



Classification (2)

Example of transfer function

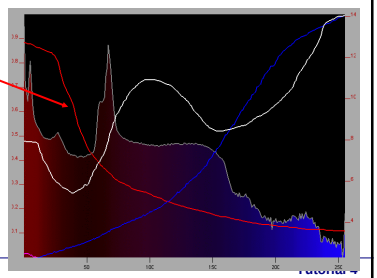
- Histogram
- Opacity



Classification (2)

Example of transfer function

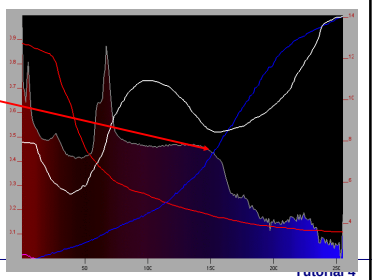
- Histogram
- Opacity
- Red channel



Classification (2)

Example of transfer function

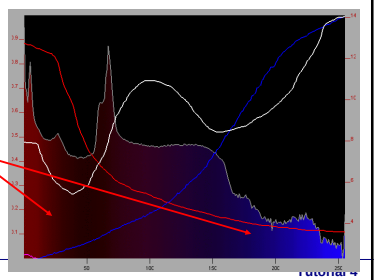
- Histogram
- Opacity
- Red channel
- Blue channel



Classification (2)

Example of transfer function

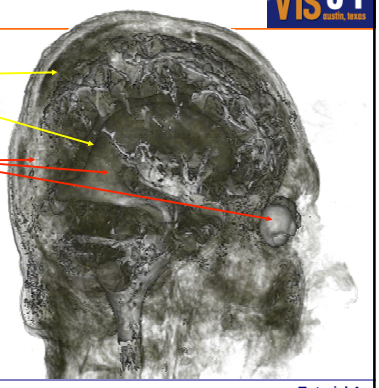
- Histogram
- Opacity
- Red channel
- Blue channel
- Resulting color



Classification (3)

Fat tissue

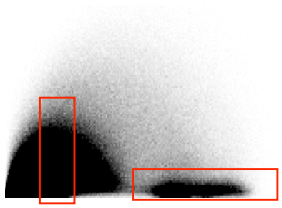
Fluid

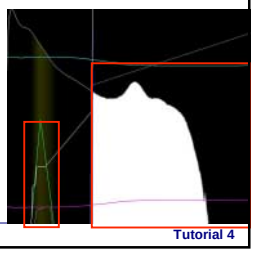


Classification (4)

2D/3D Histograms

1D Histogram

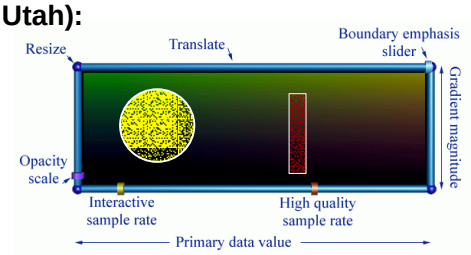




[G. Kindlmann et al., 1998]

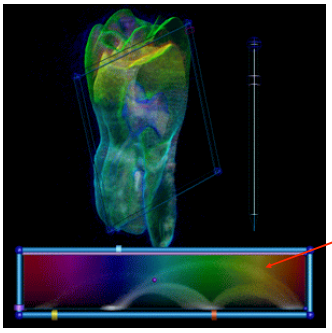
Classification (5)

3D Transfer function editor (Univ. of Utah):



[J. Kniss et al., 2001]

Classification (6)



Color table

[J. Kniss et al., 2001]

Registration (1)

- Datasets are put in context with environment
- Also referred to as **matching**
- Provides a **reference frame** for tools, ie., scalpels, endoscopes, etc.
- **Intra-operative navigation systems** register dataset with OR coordinate system

Registration (2)

- Usually an **optimization problem**
- Optimized are
  - Mutual information (entropy) or
  - Landmark matching
- The more data points, the higher the accuracy

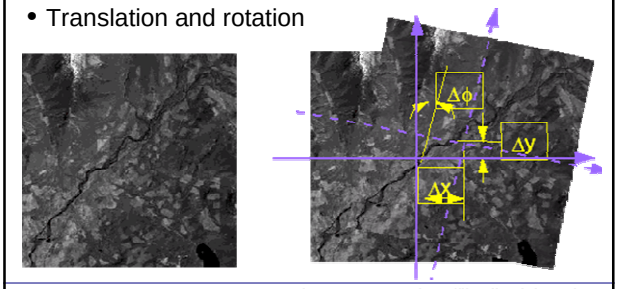
Registration (3)

- **Rigid registration**: linear transformations (translations, scaling, rotations, ...) of data volume/images (2D or 3D)
- **Non-rigid registration**: non-linear deformations of data volume; usually only 3D
- Check: http://www.image-registration.com and http://www.itk.org

Registration (4)

Registration of aerial photography

- Translation and rotation



images source: http://lis1.iis.sinica.edu.tw

Registration (5)

- The **more sample points**, the better the **accuracy**

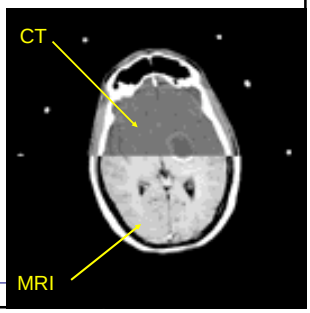
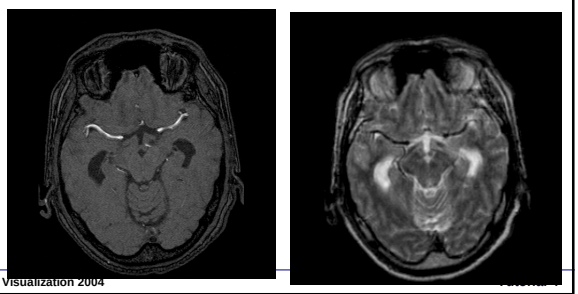


image source: http://???

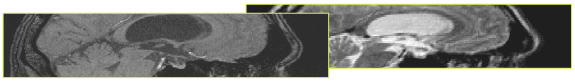
Registration (6)

Registration can be very simple:



Registration (7)

Registration can be very simple:



- Only vertical translation and scaling
- Patient movement negligible

Data Fusion (1)

- Combined representation of different datasets
- Usually requires registration
- Datasets can be from different modalities (ie., CT, MRI, rotational angiography, ...)
- Can be from different sources: Fully segmented/annotated medical atlas and patient datasets

Data Fusion (2)

Consider rendering parameters how to incorporate data from different sources:

- (Relatively) simple for surfaces
- Difficult for accumulative volume rendering

Advanced Virtual Medicine: Techniques and Applications for Medicine-Oriented Visualization

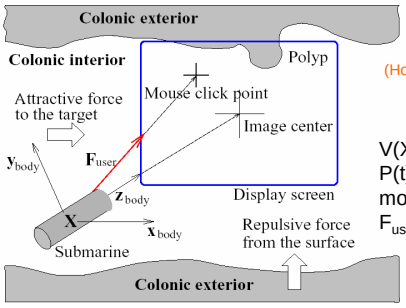
Navigation and Rendering in Virtual Endoscopy

Klaus Mueller

Computer Science Stony Brook University

mueller@cs.sunysb.edu

Physically-Based Navigation Control



(Hong et al., Siggraph '97)

V(X): potential field
P(t): linear momentum
F_{user}: user force

Submarine dynamics $\dot{P}(t) = -\nabla V(X) - kP(t) + F_{user}(t)$

Physically-Based Navigation Control

The influence of the target D_t and distance D_s potential fields can be adjusted by constants C_t and C_s

$$V(X) = \begin{cases} C_t D_t(X) + C_s(\rho / D_s(X) - 1)^2, & 0 < D_s < \rho \\ C_t D_t(X), & otherwise \end{cases}$$

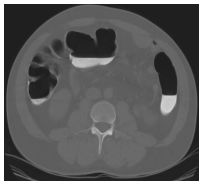
The user has full control over C_t and C_s

Large C_t accelerates the submarine towards target

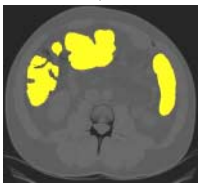
Small C_s allows users to come close to the boundary

F_{user} allows users to control the camera orientation

Data Generation and Preparation



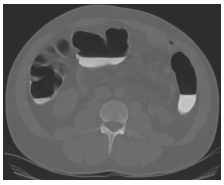
Segmentation (3D region growing)



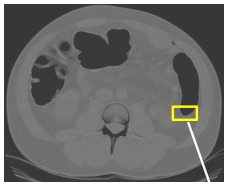
Cull remainder

3D Colon - acquired via helical CT

Data Generation and Preparation



Remove tagged fluid
(Lakare et al., Vis '00)



Reconstruct smooth surface under fluid
(Lakare and Kaufman VisSym '03)

Visualization Paradigms

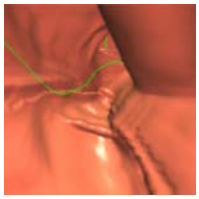
Available visualization paradigms

- 2D viewing, slice by slice (non-intuitive)

Visualization Paradigms

Available visualization paradigms

- 2D viewing, slice by slice (non-intuitive)
- 3D visualization (most appropriate)



Visualization Paradigms

Available visualization paradigms
- 2D viewing, slice by slice (non-intuitive)
- 3D visualization (most appropriate)

3D visualization paradigms (use colon as example)
- Section colon into straight pieces, slice in the center, and visualize on a virtual tray
- Unroll and flatten colon and view as a 2D sheet
- Leave as is and perform a virtual fly-through

Visualization Paradigms

Available visualization paradigms
- 2D viewing, slice by slice (non-intuitive)
- 3D visualization (most appropriate)

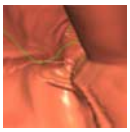
3D visualization paradigms (use colon as example)
- Section colon into straight pieces, slice in the center, and visualize on a virtual tray
- Unroll and flatten colon and view as a 2D sheet
- Leave as is and perform a virtual fly-through

3D Fly-Through Paradigms

Options range from fully passive to fully interactive

Pre-compute a video and just watch
- Cannot stop and explore
- Easy to fall asleep (TV-like)

path



Pre-compute a path along which to travel
- Better - more immersive

Pre-compute a path, but allow users to "get off"
- Allows users to explore and inspect structures

Unguided navigation
- Requires navigation skills ("driver's license")

Centerline

A good guiding path is the centerline (medial axis)

Additional desirable features for navigation:
- Provide a "pull" towards the goal (target)
- Provide a "pull" to stay on course, away from walls

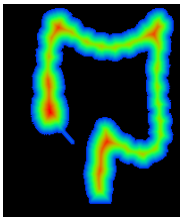
Thus, we need two potential fields

(Hong et al., Siggraph '97)

Potential Field Computation

Basically a distance transform

Two distance criteria: wall and target



Distance to boundary

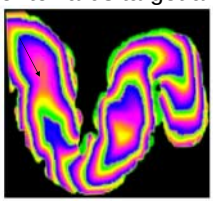
Distance to target

(Hong et al, Siggraph '97)

Potential Field Computation

Merge the two potential fields

Each voxel has potential according to both criteria
- Pull user towards target and away from wall



zoomed

overall

(Hong et al, Siggraph '97, Bitter et al., Vis '01)

Rendering: Two Options

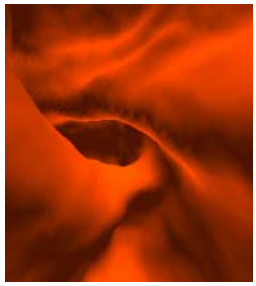
Extract polygon mesh from boundary and visualize with graphics hardware

- Discards the original volume data
- Will not allow user to “drill” into the wall to reveal inside-structures

Visualize the volume with direct volume rendering

- Nothing is lost, “drilling” possible
- Will also give more realistic (softer) images since there is no (linear) mesh approximation
- Downside: computationally expensive

Surface vs. Volume Rendering: Human Colon



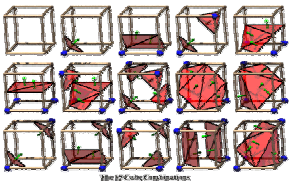
surface rendered

volume rendered

Rendering With Polygons: Mesh Generation

Extraction of polygon mesh with *Marching Cubes*

- Set iso-value *iso* to boundary
- Label all voxels below *iso* as “*out*”, else as *”in”*
- Then each voxel 8-cell fits one of 15 base cases:



(Lorensen et al., Siggraph ‘87)

Rendering With Polygons: Mesh Generation

Assemble mesh given the extracted polygons

Render with polygon graphics hardware

Problem:

- Will likely get very large meshes
- Graphics pipeline will be overwhelmed
- Rendering will not be interactive

Solution:

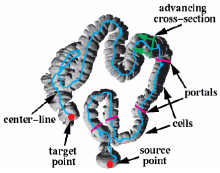
- Simplify mesh - undesirable since loss of detail
- Perform smart occlusion culling during rendering - twisted nature of object helps here

Rendering With Polygons: Occlusion Culling Preparation

Subdivide the colon into cells of about the same number of polygons and/or center path length

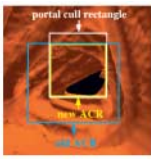
At each end of a cell, erect a bounding “portal” polygon perpendicular to the center line

Use the portals to compute visibility during rendering



Rendering With Polygons: Algorithm With Occlusion Culling

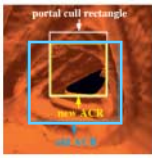
Locate cell containing camera, render its polygons



Rendering With Polygons: Algorithm With Occlusion Culling

Locate cell containing camera, render its polygons

Initialize Aggregate Cull Rectangle (ACR) to screen

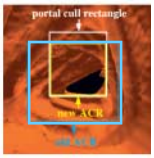


Rendering With Polygons: Algorithm With Occlusion Culling

Locate cell containing camera, render its polygons

Initialize Aggregate Cull Rectangle (ACR) to screen

Render polygons located in the two neighbor cells



Rendering With Polygons: Algorithm With Occlusion Culling

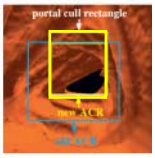
Locate cell containing camera, render its polygons

Initialize Aggregate Cull Rectangle (ACR) to screen

Render polygons located in the two neighbor cells

Perform two ACR operations for culling: Limit ACR

- by the far portals of these cells



Portal limit

Rendering With Polygons: Algorithm With Occlusion Culling

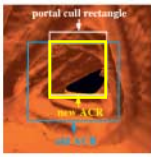
Locate cell containing camera, render its polygons

Initialize Aggregate Cull Rectangle (ACR) to screen

Render polygons located in the two neighbor cells

Perform two ACR operations for culling: Limit ACR

- by the far portals of these cells



Portal limit

Rendering With Polygons: Algorithm With Occlusion Culling

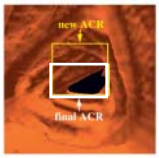
Locate cell containing camera, render its polygons

Initialize Aggregate Cull Rectangle (ACR) to screen

Render polygons located in the two neighbor cells

Perform two ACR operations for culling: Limit ACR

- by the far portals of these cells
- by z-buffer



Z-buffer limit

Rendering With Polygons: Algorithm With Occlusion Culling

Locate cell containing camera, render its polygons

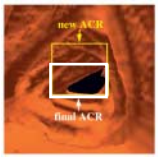
Initialize Aggregate Cull Rectangle (ACR) to screen

Render polygons located in the two neighbor cells

Perform two ACR operations for culling: Limit ACR

- by the far portals of these cells
- by z-buffer

Visit next cells, render polygons, limit ACR



Rendering With Polygons: Algorithm With Occlusion Culling

Locate cell containing camera, render its polygons

Initialize Aggregate Cull Rectangle (ACR) to screen

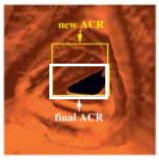
Render polygons located in the two neighbor cells

Perform two ACR operations for culling: Limit ACR

- by the far portals of these cells
- by z-buffer

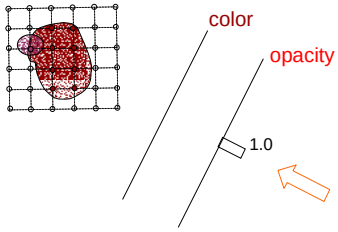
Visit next cells, render polygons, limit ACR

Stop when ACR degenerates to zero



Direct Volume Rendering: Raycasting Fundamentals

volumetric compositing

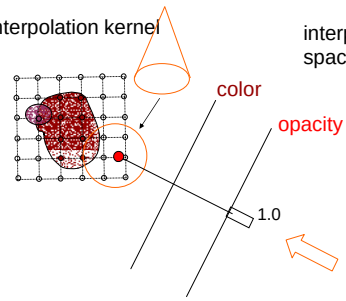


object (color, opacity)

Direct Volume Rendering: Raycasting Fundamentals

interpolation kernel

interpolated empty space

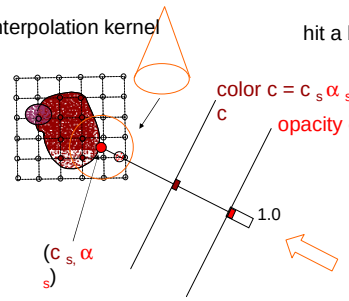


object (color, opacity)

Direct Volume Rendering: Raycasting Fundamentals

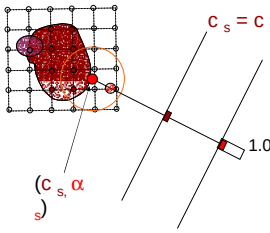
interpolation kernel

hit a boundary surface

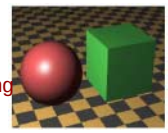


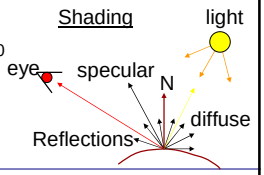
object (color, opacity)

Direct Volume Rendering: Raycasting Fundamentals



$c_s = c_s \cdot$ shading

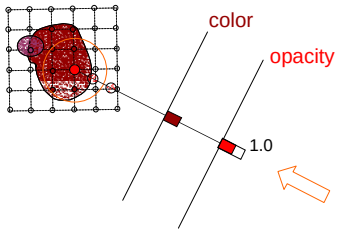




object (color, opacity)

Direct Volume Rendering: Raycasting Fundamentals

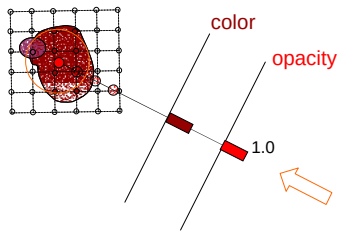
volumetric compositing



object (color, opacity)

Direct Volume Rendering: Raycasting Fundamentals

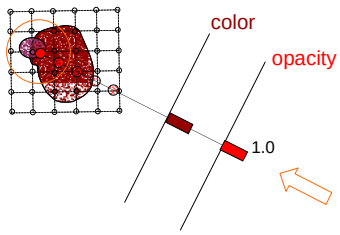
volumetric compositing



object (color, opacity)

Direct Volume Rendering: Raycasting Fundamentals

volumetric compositing

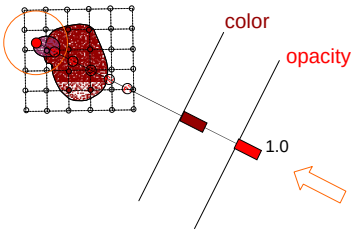


object (color, opacity)

IEEE Visualization 2004 Tutorial 4

Direct Volume Rendering: Raycasting Fundamentals

volumetric compositing

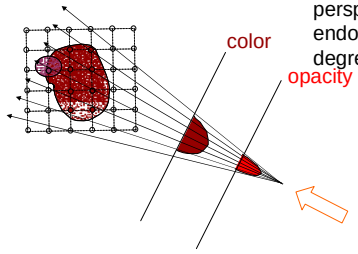


object (color, opacity)

IEEE Visualization 2004 Tutorial 4

Direct Volume Rendering: Raycasting Fundamentals

whole image - perspective angle in endoscopy often 60 degrees and more



object (color, opacity)

IEEE Visualization 2004 Tutorial 4

Direct Volume Rendering: Raycasting Fundamentals

Shortcoming:

- Traversing and interpolating empty space until a boundary is hit is time consuming and limits performance

Require some form of accelerated empty-space traversal - "*space leaping*"

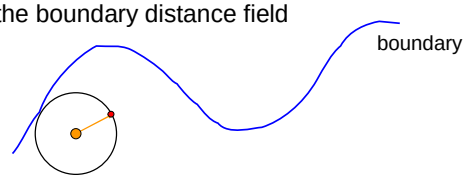
Solutions

- Render polygons into z-buffer, then use the z-depths to start rays (large polygon overhead)
- Better: use the potential field to speed up rays

IEEE Visualization 2004 Tutorial 4

Potential Field Assisted Raycasting

Use the boundary distance field

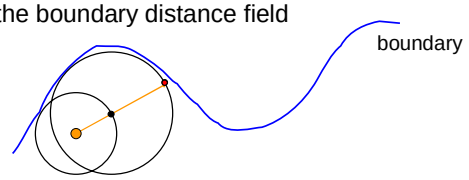


Wan et al. (Vis'99)

IEEE Visualization 2004 Tutorial 4

Potential Field Assisted Raycasting

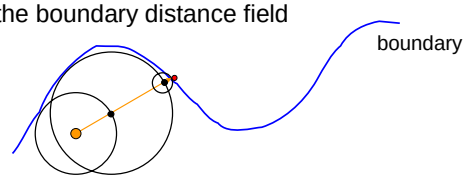
Use the boundary distance field



Wan et al. (Vis'99)

Potential Field Assisted Raycasting

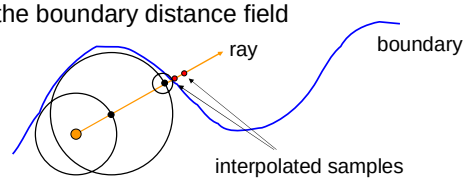
Use the boundary distance field



Wan et al. (Vis'99)

Potential Field Assisted Raycasting

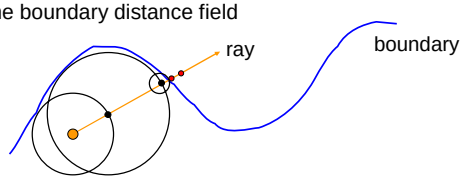
Use the boundary distance field



Wan et al. (Vis'99)

Potential Field Assisted Raycasting

Use the boundary distance field



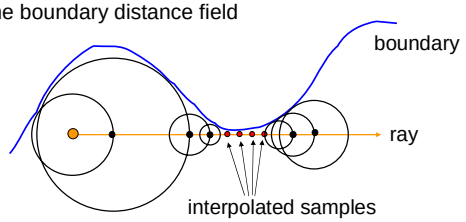
Two hops are required instead of ten samples

This, in fact, corresponds to the average case

These speedups (and further optimizations) allow the required interactive frame rate of 10 fps

Potential Field Assisted Raycasting

Use the boundary distance field



Rays that graze the surface will not accelerate well

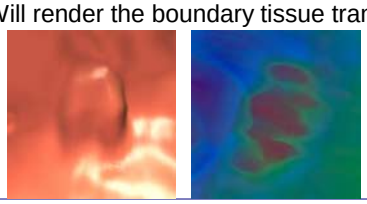
- Fortunately these are rare

Opaque vs. Translucent Rendering

Setting a high opacity at the boundary will render the surface opaque

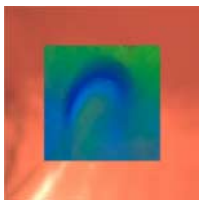
Selecting low opacities allows rays to penetrate further into the boundary tissue

- Will render the boundary tissue translucent

opaque  translucent

Digital Biopsy

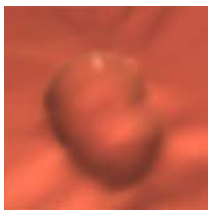
Translucent rendering allows one to visualize tissue underneath the surface (Lakare et al., SPIE '01)

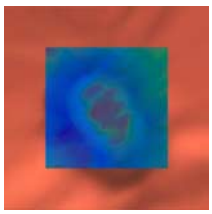


Hyperplastic polyp: surface translucent

Digital Biopsy

Translucent rendering allows one to visualize tissue underneath the surface (Lakare et al., SPIE '01)





Adenoma: surface translucent

Advanced Virtual Medicine: Techniques and Applications for Medicine-Oriented Visualization

Visualization and Navigation Techniques II

Yin Wu

TeraRecon

wu@terarecon.com

Visualization and Navigation Techniques

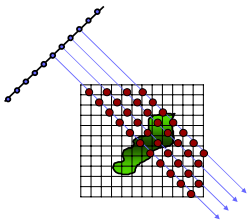
- Real time volume visualization techniques
  - Texture Method: GPU method
    - 2D Texture: Shear Warp
    - 3D Texture: Image Order

Dependent on Graphic Card Memory

On the fly gradient calculation need pixel shading functions

Visualization and Navigation Techniques

- **Real time volume visualization techniques**
  - **Shear-Image Ray Casting Method**
    - **Rays through pixels on image plane**
      - **Image quality equiv. to full image order**
      - **No 2nd resampling**
    - **Samples organized in planes parallel to volume axes**
      - **Traverse & resample data in memory order**
      - **Maximize memory performance**
      - **Share neighbor interpolation values**



Visualization and Navigation Techniques

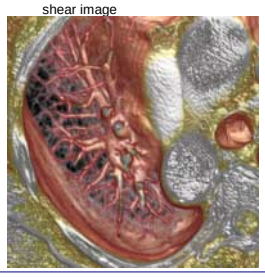
- Real time volume visualization techniques
  - Shear-image: Algorithm
- Voxel processing part
  - Traverse data slice-by-slice in memory order
  - Read voxels for each slice of samples
  - Voxel-oriented processing (e.g., gradient estimation)
  - Store in on-chip buffers
- Sample processing part
  - Define sample points where rays intersect a slice
  - Traverse & interpolate on-chip buffer in pixel order
  - Sample-oriented processing
    - e.g., illumination, filtering, depth testing, compositing
  - Output to image

Visualization and Navigation Techniques

- Real time volume visualization techniques
  - Shear warp vs shear image:Image Quality Comparison

Shear warp

shear image



Visualization and Navigation Techniques

- Real time volume visualization techniques
  - VolumePro 1000: implementation

Single-Board, Special Purpose Computer

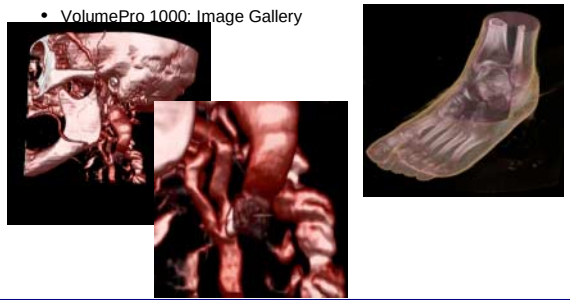
- Full-size PCI card for 64 bit slot
- 40W Internal or external powered
- Memory configurations
  - 1 GB & 2 GB
- Half-size PCI card for 64 bit slot
- 30W Internal powered
  - 512 MB & 1 GB

Visualization and Navigation Techniques

- **Real time volume visualization techniques**
  - **VolumePro 1000 Features…**
- ***Feature Rich* – render tools, opaque and translucent polygons, perspective**
- ***Superior Image Quality* – Shear-Image(XY image)order with 16 bit data paths**
- ***Platform Agile* – Windows, Solaris, Linux**
- ***Ease of Integration* – VLI SDK, TGS Open Inventor, Kitware VTK**
- ***Interactive Performance* - large 3D and 4D volume data sets on the desktop**
- ***Scalable Solutions* – multiple board load sharing**
- **.....**

Visualization and Navigation Techniques

- Real time volume visualization techniques
  - VolumePro 1000: Image Gallery



Visualization and Navigation Techniques

- Real time volume visualization techniques
  - VolumePro 1000: Image Gallery

Outline

Introduction

Foundations

Applications of Virtual Medicine

- **Virtual Endoscopy**
- Distributed Virtual Medicine
- fMRI for Surgical Planning
- Soft-Tissue Simulation /Deformation

Question and Answers

Advanced Virtual Medicine: Techniques and Applications for Medicine-Oriented Visualization

Applications of Virtual Medicine
Virtual Endoscopy

Dirk Bartz

WSI/GRIS – VCM, University of Tübingen

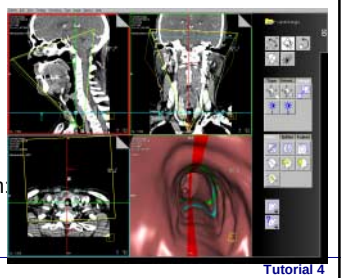
bartz@gris.uni-tuebingen.de

Virtual Endoscopy (1)

Commercial systems

Siemens Syngo

© Siemens Medical Solutions



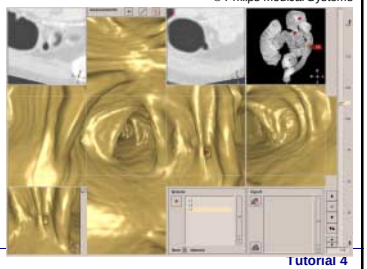
- Syngo is new system
- Uses ray casting/ VolumePro
- Manual and automatic navigation
- (Virtuoso is older system: 2D texture mapping based)

Virtual Endoscopy (2)

Commercial systems, cont´:

Philips EasyVision Endo 3D

© Philips Medical Systems



- Automatic navigation
- Colon unfolding
- Ray casting

Virtual Endoscopy (3)

Commercial systems, cont´:

GE Advantage Windows

© GE Medical Systems

- Semiautomatic path generation
- Ray casting

Virtual Endoscopy (4)

Commercial systems, cont´:

Viatronix V3D Viewer/Colon

© Viatronix

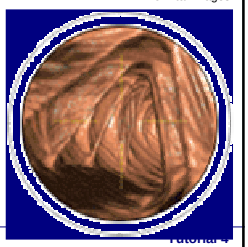
- Ray Casting
- Guided/manual
- Technology partially based on SUNY SB VICON

Virtual Endoscopy (5)

Commercial systems, cont´:

Vital Images
Vitrea2/CT Colonography

© Vital Images



- Automatic/manual
- 2D/3D Texture Mapping

Virtual Endoscopy (6)

Research Systems:

- VIVENDI - U Tübingen
  - OpenGL/Visibility culling
  - Guided navigation
- VICON - SUNY Stony Brook
  - similar to VIVENDI, but bound to colonoscopy
- CRS4 Sardinia
  - 3D TEX on SGI RE2
- FreeFlight, U Wake Forest
  - OpenInventor
- TU Wien / TIANI Medgraph
  - 3D Movie (similar to QuickTimeVR)

Advantages and Limitations (1)

Some statements:

- VE (Virtual Endoscopy) **will not replace** optical endoscopy
- VE is only as good as dataset/image acquisition
- Will cause a shift of revenue from GE/surgery to radiology

Advantages and Limitations (2)

Advantages of VE:

- Non-invasive
- Better patient compliance
- Control of lighting and orientation
- No physical access limitations
- All (segmented) areas are accessible
- More economical than optical endoscopy

Advantages and Limitations (3)

Limitations of VE:

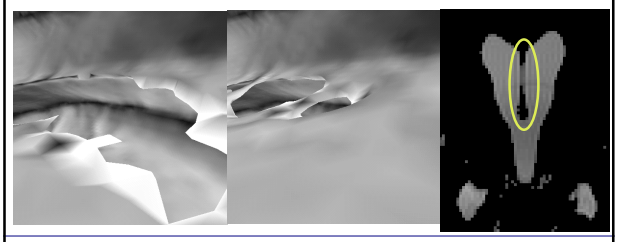
1. Segmentation

- Represented information depends totally on **segmentation quality**
- Flaws of segmentation can (usually) not be fixed at rendering
- Which is the correct classification?

Advantages and Limitations (3)

Limitations of VE:

1. Segmentation



Advantages and Limitations (4)

Limitations of VE:

2. Rendering

- Rendering never better than data
- Resolution of **real biopsy is unmatched**
- Transparent rendering of inner surface ("look through", "**virtual biopsy**") **does not replace real biopsy**

Advantages and Limitations (5)

Limitations of VE:

3. Concept

- **No physical interaction** with tissue: no removal of pathological tissue
- No actual intervention
- Structure of interest might have changed since acquisition ("**brain shift**")

Advantages and Limitations (6)

Requirements for VE:

- Sufficient accuracy
- Identifiable (segmentable) structures of interest
- Interactivity (> 19 fps)
- Awareness of reasons for artifacts

Outline

Introduction

Foundations

Applications of Virtual Medicine

- Virtual Endoscopy
- **Distributed Virtual Medicine**
- fMRI for Surgical Planning
- Soft-Tissue Simulation /Deformation

Question and Answers

Advanced Virtual Medicine: Techniques and Applications for Medicine-Oriented Visualization

Applications of Virtual Medicine: Distributed Virtual Medicine

Yin Wu

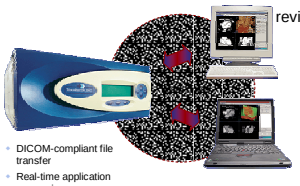
TeraRecon

wu@terarecon.com

Distributed Virtual Medicine

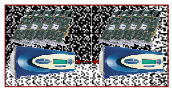
- The client and power server system can distribute images and processing power over the network for medical distribution.

Example: AquariusNET enterprise server with thin-client or workstation-client software



Distributed Virtual Medicine

- AquariusNET Enterprise 3D Serve.
  - scalable Image Network Server



Multiple AquariusNET servers can be placed around the network, wherever a system needs processing power and image distribution. The servers can work on many different datasets at once, for many users, or they can work in concert on the most demanding datasets.

- Interactive report

Radiology creates reports with embedded dynamic images. 3D reports are delivered over the Internet by email in Microsoft® Word. Referring physician dynamically reviews, interacts with, and manipulates 3D images within the Word report.

Outline

Introduction

Foundations

Applications of Virtual Medicine

- Virtual Endoscopy
- Distributed Virtual Medicine
- **fMRI for Surgical Planning**
- Soft-Tissue Simulation /Deformation

Question and Answers

Advanced Virtual Medicine: Techniques and Applications for Medicine-Oriented Visualization

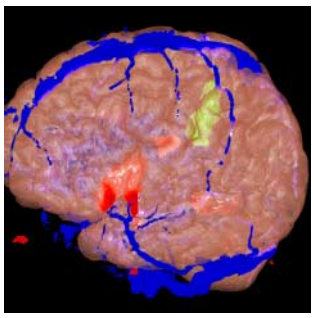
Applications of Virtual Medicine: fMRI for Surgical Planning

Jan Hardenbergh

TeraRecon

hardenbergh@terarecon.com

fMRI Intro



fMRI images functional areas of the brain, for example left motor, right motor, language acquisition, etc.

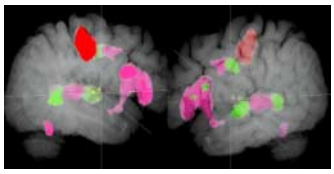
Motivation

Neurosurgeon wants to get to tumor without impairing functions such as speech and movement

Tumor

Inside (R)

Outside (L)



fMRI Background

Blood flow increases to active regions of the brain – saturates it with oxygen.

Deoxyhemoglobin is paramagnetic (no O_2) and can be imaged with fMRI.

While in scanner, subject exercises mental functions.

fMRI Scanning

Done in a standard MRI scanner

Scan takes up to 2 hours to scan anatomy, vasculature and functional areas.

Patient performs various tasks, moving hands, reading, verbalizing.

fMRI Analysis

Correlate stimulus sequence timing with MRI data

Lots of background activity

Network analysis

fMRI volume data

2 volumes + one for each functional area

Example has three functional areas

- Right hand motor
- Language acquisition
- Verbal Generation

Each volume is 256x256x124 = 8MB.

Packing Activation data

8 channels is hard (especially with masking)

40 MB of texture memory is getting smaller, but, still hard to fit in 64MB.

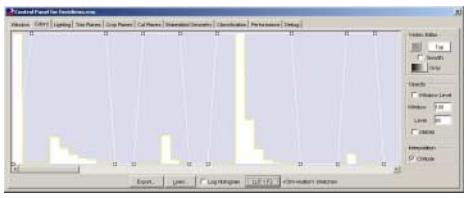
Pack the 3 activation datasets into 1.

Offset the voxel values (remember how)

RHM 2 4 9 AC 14 16 5 VG 22 24 9 -VG-AC 36 36 4

Packing Activation data

RHM 2 4 9 AC 14 16 5 VG 22 24 9 -VG-AC 36 36 4



Overlapping Activations

Language Acquisition and Verbal Generation share part of the brain.

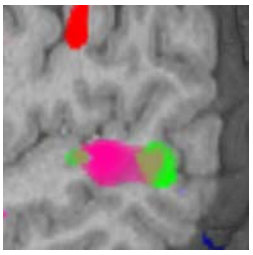
When an overlap is detected a new “activation” area is created – using a single voxel value.

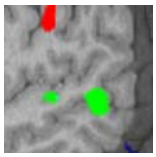
Displayed as blend if both areas are visible, else in the color of the visible activation.

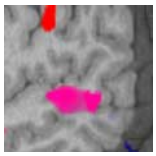
-VG-AC 36 36 4

Overlap Display

Blend below (olive)
Each separate area to right

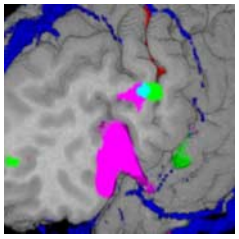


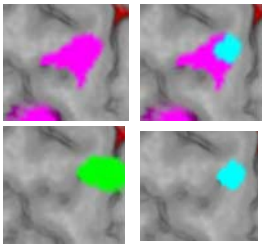




Overlap Display 2

Don't blend and don't hide overlaps: Picture on the left is the same. Middle hides overlaps. Right shows overlap separate.



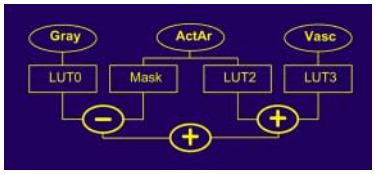


Channel Masking

Needed to clearly see the activation area.

Use a masking LUT for the activations

Diagram shows VP1000 classifier method



Channel Masking on Radeon

Activation as RGBA (needed for CI)

Read Gray and Vascular channels, pass Act

Setup dependent read of 1D textures (LUT)

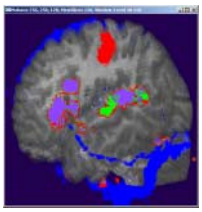
Read LUTs for Gray&Vasc, + Act RGBA

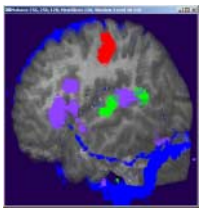
Subtract Act's Alpha from all channels of Gray/LUT result

Add result + Act RGBA + Vasc/LUT

Interpolate First?

Examples from Radeon, IC, nearest neighbor – for CI, we need to create RGBA





fMRI UI

Panel allows each area to

be turned on and off.

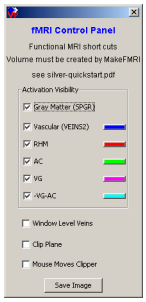
Window/Level allows easy modification of transparency

Clip Plane is easy.

fMRI UI

Alternatively,

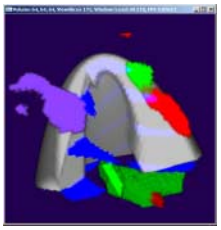
We can display the activation areas as independent Activations.

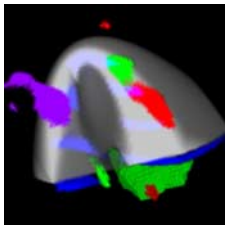


Radeon and VP1000

VP1000 image quality better. Much faster.

Demo data does not fit in 64MB with RGBA





Brain Imaging

CT / MRI - anatomy

fMRI – functional areas

PET = positron emission tomography

SPECT = single positron …

MRS = magnetic resonance spectroscopy

MEG = magnetoencephalography

SPECT & PET

SPECT (single-photon emission computed tomography)

shows blood flow and other chemicals in the brain.

"radiotracers" injected; gamma rays measured. 8mm resolution

PET (Positron Emission Tomography)

Need cyclotron nearby. 4mm resolution

www.musc.edu/psychiatry/fnrd/primer_spectpet.htm

MEG

Magnetoecephalograpyh

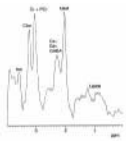
- Millisecond time resolution
- 2mm spatial res.

Better at detecting some functional areas.

MRS

MRI spectroscopy

Spectograph per voxel



atoms such as 31P, 23Na, K, 19F, or Li are detected.

More Info

This information as of August 2004:

www.jch.com/volumes/fmriviz.htm

New books on fMRI published

Outline

Introduction

Foundations

Applications of Virtual Medicine

- Virtual Endoscopy
- Distributed Virtual Medicine
- fMRI for Surgical Planning
- **Soft-Tissue Simulation /Deformation**

Question and Answers

Advanced Virtual Medicine: Techniques and Applications for Medicine-Oriented Visualization

Soft-Tissue-Simulation and Deformation

Markus Wacker, Michael Hauth,

WSI/GRIS, University of Tübingen

{wacker,hauth}@gris.uni-tuebingen.de

Generations of Medical Simulators

1st Generation Simulator: Geometry/Anatomy/Navigation
- Shape
- Surface & Volume
- Morphology

2nd Generation Simulator: Physics/Interaction
- Deformation & Forces
- Physical Properties

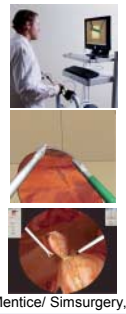
3rd Generation Simulator: Functional aspects/Physiology
- Pathology
- Function of organs and cells

[Satava 1996,Delinquette 1998]

What we want to do: Education / Training / Assessment

Classification of training in virtual minimal invasive surgery

- Skill Level
  - Moving instruments
  - Touching locations
- (Basic) Task Level
  - Suturing
  - Clip & Cut a vessel
- Procedure Level
  - Detach and remove gallbladder
- Intervention Level
  - Cholecystectomy



©Images Mentice/ Simsurgery, VSOne

Soft Tissue Modelling and Simulation

Soft tissue simulation module

- Key technology for 2nd and 3rd generation simulator
- for training beyond task level
- for improving visual realism
- for faithful force feedback
  (Non-authentic force-feedback may lead to the adoption of false impressions.)

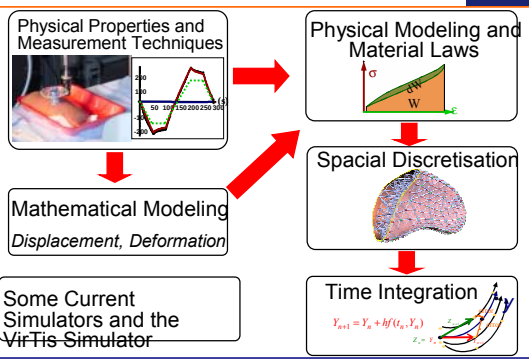
Accuracy vs. Computation time

Outline

Requirements for Soft Tissue Simulation

i. Physical Properties and Measurement Techniques
ii. Mathematical Modeling
iii. Physical Modeling and Material Laws
iv. Spatial Discretisation
v. Time Integration
vi. Some Current Simulators

Outline: Requirements for Soft Tissue Simulation



i. Physical Properties and Measurement Techniques

Physical properties of Soft Tissue

Soft Tissue deformation is:

1. **elastic:** retreats to rest (spring)
2. **viscous**: damping
   but neither uniform nor immediately
3. **plastic**: non-reversible process

Blood filled tissue is highly incompressible

Measurements in Biomechanics

Tissue is Point of Interest in **Biomechanics**

Focusing on skin, vessels, muscles, bone and cells

"Standard" textbook: Fung, *Biomechanics: mechanical properties of living tissue*

Most work up to mid 90s: Ex vivo

Now focussing: In vivo

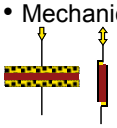
©Images MIC Tübingen, CIMIT

Measurements in Biomechanics

Overview

Ex vivo

- Mechanical Tear
- Mechanical Shear



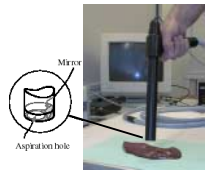
[Gross et al. 2000, Ishihara 2000, Sakuma 2003]

In vivo

- Aspiration, Indentation
- Tactile Imaging
- Mechanical Shear
- Elastographie: MRE, Ultrasound

In Vivo: Experiments

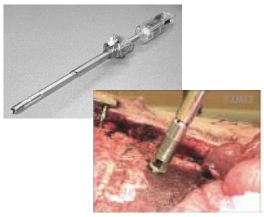
Aspiration



Result

[Kauer 2001; Nava, Mazza et al. 2003]

Indentation
TeMPeST 1-D

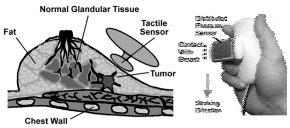


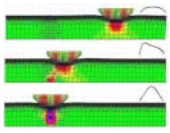
Ottensmeyer et. al. 2002 (TeMPeST 1-D);

©Images ETH Zürich, CIMIT

Measurements

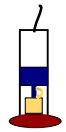
Tactile Imaging

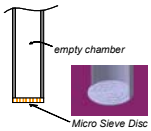




Wellman 1999, Galea and Howe 2003 (Tactile Imaging)

ROSA II



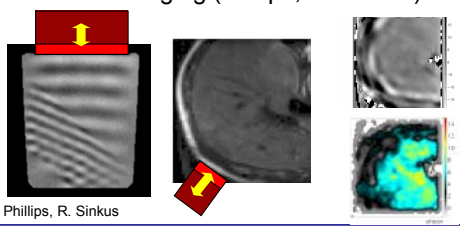


[Gross et. al. (ROSA II) 2003; Mazza et. al. 2003]

©Images Harvard Biorobotics, MIC Tübingen, E. Mazza ETH Zürich

Non Invasive: Elastography

- (Static) Ultrasound measurements
- Dynamic Magnetic Resonance Elastography (MRE)
- 3D Ultrasound imaging (Philips, R. Sinkus)

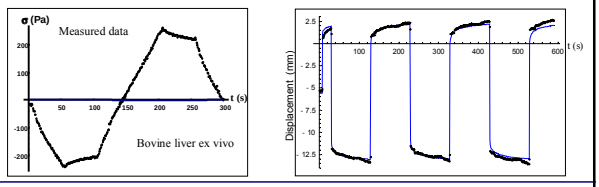


©Images Phillips, R. Sinkus

Results: Characteristics of Soft Tissue

- Data is limited to specific frequency ranges
- Instruments often designed for specific organs (mainly liver, kidney)

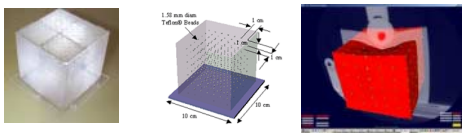
Common result: The visco-elastic properties depend on the frequency of excitation



©Images MIC Tübingen, ETH Zürich E. Mazza

Current Activities

- Enhance database
- Comparative measurements [Ottensmeyer 2003]
- Define standard benchmarks: Truthcube.org, E. Mazza



- Develop better in vivo instruments

 © Images Truthcube.org

Measurement Goals

Near Term:

Single organ (or limited region) "as good as possible"

⇒ Minimal invasive procedures

Long Term: (5+ years)

- Besides "standard training": Patient specific training (data acquisition ?)
- Open surgery (haptic & visual devices?)
- 3rd generation simulators

Outline

i. Physical Properties and Measurement Techniques
ii. Mathematical Modeling
iii. Physical Modeling and Material Laws
iv. Spatial Discretisation
v. Time Integration
vi. Some Current Simulators

iii. Mathematical Modeling

Mathematical Model

Parametrized Surface: A: $\mathrm{R}^2 \supset K \rightarrow \mathrm{R}^3$

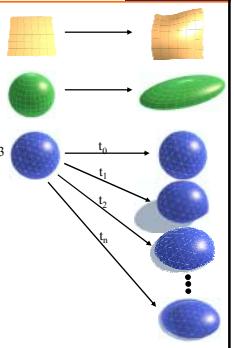
Rigid Body: S: $\mathrm{R}^3 \supset K \rightarrow \mathrm{R}^3$

Deformable Solid: ϕ: $\mathrm{R}^3 \supset K \times [0,\infty) \rightarrow \mathrm{R}^3$

Configuration ϕ

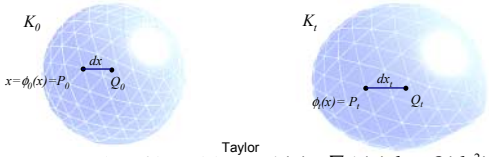
Often: initial configuration $K = \phi(\cdot\,,0) = \mathrm{id}$

Displacement Field $\mathrm{u} := \phi - \mathrm{id}$



©Images M.Hauth

Measure of Deformation: Strain ε



$$dx_t = Q_t - P_t = \phi_t(x+dx) - \phi_t(x) \overset{\text{Taylor}}{=} \phi_t(x) + \nabla\phi_t(x)dx + O(dx^2) - \phi_t(x)$$

$$\approx \nabla\phi_t(x)dx = (\nabla u_t(x) + id)dx$$

$$\Rightarrow \|dx_t\|^2 - \|dx\|^2 = \overbrace{\langle \nabla\phi_t^T \nabla\phi_t dx, dx\rangle - \langle dx, dx\rangle}^{\langle \nabla\phi_t dx, \nabla\phi_t dx\rangle} = \langle (\nabla\phi_t^T \nabla\phi_t - id)dx, dx\rangle$$

Define $\boxed{\varepsilon := \tfrac{1}{2}(\nabla\phi_t^T \nabla\phi_t - id) = \tfrac{1}{2}(\nabla u_t^T + \nabla u_t + \nabla u_t^T \nabla u_t)}$

Alternatively: Use *metric tensor* g_{ij}: $\varepsilon_{ij}(P) = \frac{1}{2}[g_{ij}(P) - G_{ij}(P)]$

Different Strain Tensors

Green's strain tensor ε:

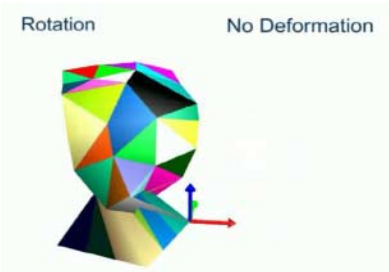
$$\varepsilon_{ij} = \tfrac{1}{2}(\nabla u_t^T + \nabla u_t + \nabla u_t^T \nabla u_t) = \frac{1}{2}\left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \sum_{k=1}^{3} \frac{\partial u_k}{\partial x_j}\frac{\partial u_k}{\partial x_i}\right]$$

Cauchy's tensor (linear)

Green's tensor:
- ↗ *second* order strain description
- ↗ invariant under rigid body rotations
- ↗ suitable for large deformations *and* large strain
- ↘ nonlinear

Cauchy's tensor:
- → *first* order strain description
- ↘ **not** invariant under rigid body rotations
- ↗ **linear** for small deformation/strain problems

Cauchy Ghost-Forces



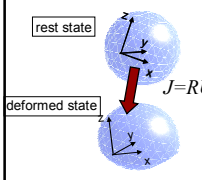
© M.Hauth

Strain Tensors

$$\varepsilon_{ij} = \frac{1}{2}\left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right] + \sum_{k=1}^{3} \frac{\partial u_k}{\partial x_j}\frac{\partial u_k}{\partial x_i}$$

Green's tensor Cauchy's tensor

→Use local **rotated** coordinate system

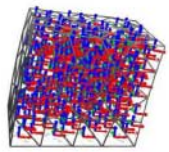


$$\varepsilon^{CR}(\phi) := \varepsilon^{C}(\mathrm{R}^{\mathrm{T}}\phi)$$

$J = RU$ polar decomposition

Co-rotational Cauchy stress tensor
- ↗ invariant under rigid body rotations
- ↗ linear

Tracking local coordinate systems



© M.Hauth

Summary

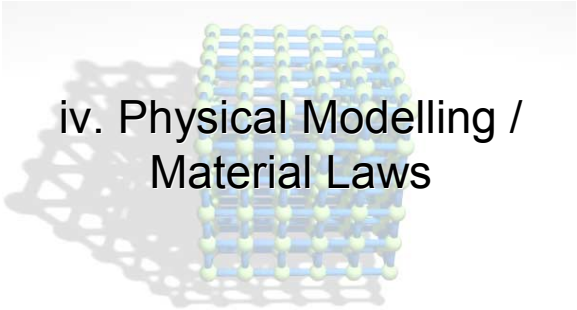
Deformation as displacement field:

$$u(t,x) := \phi(t,x) - \phi(0,x) = \phi(t,x) - x$$

- Green's Tensor: **non**linear for finite displacement, rotationally invariant
- Cauchy tensor: linear *not* invariant under finite rotations
- Co-rotated Chauchy: linear **and** invariant under finite rotations

Outline

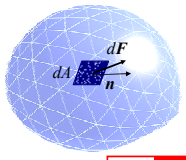
i. Physical Properties and Measurement Techniques
ii. Mathematical Modeling
iii. Physical Modeling and Material Laws
iv. Spatial Discretisation
v. Time Integration
vi. Some Current Simulators



iv. Physical Modelling / Material Laws

Material laws

strain tensor ε: geometrical state.
stress tensor σ: internal forces.



$$\frac{d\boldsymbol{F}}{dA} = \boldsymbol{\sigma} \cdot \boldsymbol{n}$$

Assumption: The stress σ is a function of strain (-history).

$$\sigma = \sigma(\varepsilon) = \sigma(\varepsilon(u))$$

Material law. linear or non-linear

Non-linear for large strain/displacement (*geometrical non-linearity*)

Physical properties of Soft Tissue

Soft Tissue deformation is:

1. **elastic**: retreats to rest (spring)
2. **viscous**: damping

   but neither uniform nor immediately

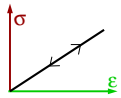
3. **plastic**: non reversible process

Blood filled tissue is highly incompressible

Hooke's law

Elastic, isotropic, homogeneous solid:

$$\sigma = C\varepsilon$$



$$\sigma_{11} = \lambda(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}) + 2\mu\varepsilon_{11} \qquad \sigma_{12} = 2\mu\varepsilon_{12}$$
$$\sigma_{22} = \lambda(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}) + 2\mu\varepsilon_{22} \qquad \sigma_{13} = 2\mu\varepsilon_{13}$$
$$\sigma_{33} = \lambda(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}) + 2\mu\varepsilon_{33} \qquad \sigma_{23} = 2\mu\varepsilon_{23}$$

with Lamé's constants $\lambda, \mu > 0$.

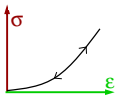
Young's modulus $E = \mu \frac{3\lambda + 2\mu}{\lambda + \mu}$

Poisson's ratio $\nu = \frac{\lambda}{2(\lambda + \mu)}$

Nonlinear elastic models

Nonlinear elastic solid:

$$\sigma = f(\varepsilon)$$



- Mooney-Rivlin
- Neo-Hooke
- Veronda-Westermann
- …

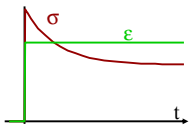
General Material Law

For any linear, causal material there is a relaxation function R(t), s.t.

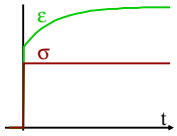
$$\sigma(t) = \int_{-\infty}^{t} R(t-\tau)\,\dot{\varepsilon}(\tau)\,d\tau$$

Time dependant effects

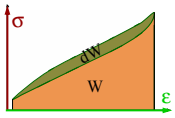
Stress relaxation



Creep

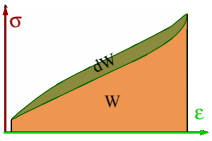


Hysteresis



Mechanical Quality Q

Fung, Biomechanics: "In biological tissues, the shape of the hysteresis loop is independent of frequency in a range of several orders of magnitude"



$$Q(\omega) = \frac{4\pi\ W}{\Delta W}$$

Purely viscous materials: Q = 0

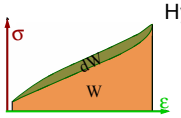
Purely elastic materials: Q = ∞

Q is constant

Confirmed for liver tissue by several ex vivo *and in vivo* experiments (in [0.1,…,50] Hz.; by J.Groß)

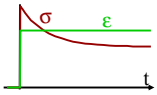
E. Kjartansson: constant Q (CQ) theory for rocks

Constant Q



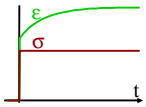
Hysteresis

$$\sigma(t) = \int_{-\infty}^{t} R(t-\tau)\,\dot{\varepsilon}(\tau)\,d\tau$$



Stress Relaxation

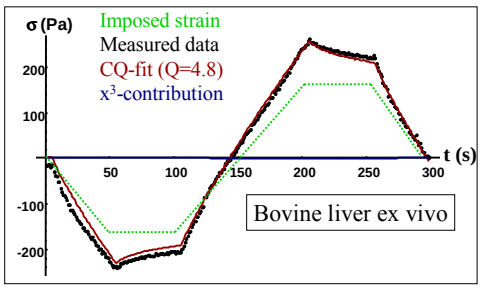
$$R(t) = \frac{M_0}{\Gamma(1-2\gamma)}\left(\frac{t}{t_0}\right)^{-2\gamma} \quad t>0,\ \gamma = \frac{1}{\pi}\tan^{-1}\left(\frac{1}{Q}\right)$$



Creep

$$C(t) = \frac{1}{M_0\Gamma(1+2\gamma)}\left(\frac{t}{t_0}\right)^{2\gamma} \quad t>0$$

Measurements and CQ-fit



Why CQ? ↗ Permits a good reproduction of the measured data

©Images MIC Tübingen ↗ **Single** parameter: Q

Prony Series for Numerical Simulation

Fit the relaxation function with an exponential Prony series:

$$R(t) \approx \mu_0 + \sum_{i=1}^{m} \mu_i\, e^{-t/\lambda_i}$$

$$\sigma(t) = \int_{-\infty}^{t} R(t-\tau)\,\dot{\varepsilon}(\tau)\,d\tau$$

$$\Rightarrow \quad \sigma(t) = \mu_0\varepsilon(t) + \sum_{i=0}^{m}\mu_i\left[\int_{t_0}^{t} e^{-\frac{t-\tau}{\lambda_i}}\dot{\varepsilon}(\tau)\,d\tau + \boxed{e^{-\frac{t-t_0}{\lambda_i}}\int_{-\infty}^{t_0} e^{-\frac{t_0-\tau}{\lambda_i}}\dot{\varepsilon}(\tau)\,d\tau}\right]$$

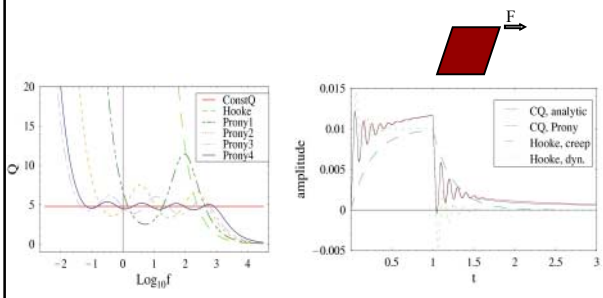
memory

$$\Rightarrow \quad \sigma(t) = \mu_0\varepsilon(t) + \sum_{i=0}^{m}\mu_i q_i(t)$$

$$q_i(t) = e^{\frac{\Delta t}{\lambda_i}} q_i(t_n) + \Delta q_i$$

$$\Delta q_i = \frac{\Delta t}{\lambda_i}\left(1 - e^{\frac{\Delta t}{\lambda_i}}\right)\left(\varepsilon(t) - \varepsilon(t_n)\right)$$

Quality of the Approximation

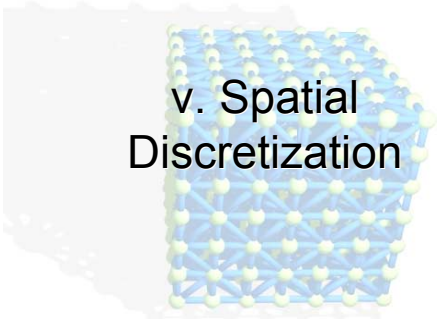


Summary

- Measurements imply a **frequency dependant** material model
- A simple and currently adequate model is given by assuming a **constant Q**
- **Prony series** fit
  - allows *arbitrary* frequency dependencies (with sufficient parameters)
  - standard in material sciences
  - can be generalized to large strain settings

Outline

i. Physical Properties and Measurement Techniques
ii. Mathematical Modeling
iii. Physical Modeling and Material Laws
iv. Spatial Discretisation
v. Time Integration
vi. Some Current Simulators



v. Spatial Discretization

Equations for deformable objects

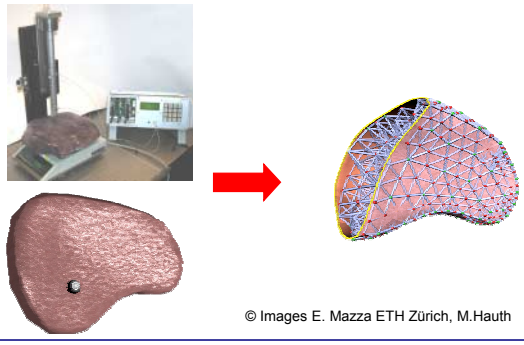
Equation of equilibrium

$$\operatorname{div}\sigma - f = 0$$

No inertia and viscosity (dynamical effects)
➡ Equation of motion

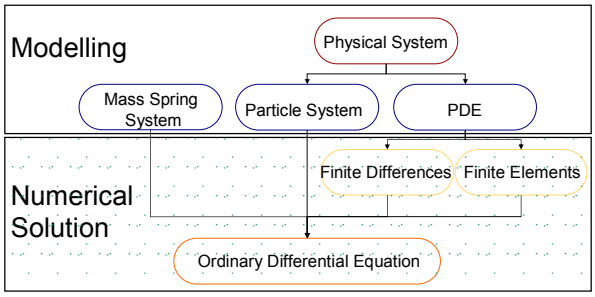
$$\operatorname{div}\sigma + f = \rho \frac{\mathrm{d}}{\mathrm{dt}} v$$

From Displacements to Simulation

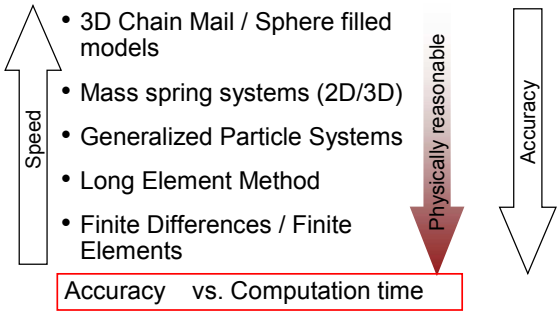


© Images E. Mazza ETH Zürich, M.Hauth

Physically Based Simulation



Methods

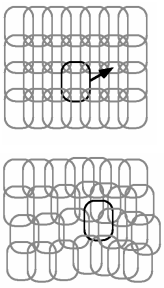


- 3D Chain Mail / Sphere filled models
- Mass spring systems (2D/3D)
- Generalized Particle Systems
- Long Element Method
- Finite Differences / Finite Elements

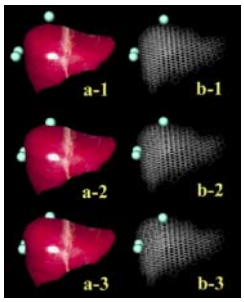
Accuracy vs. Computation time

3D Chainmail / Sphere filled Models

Images taken from Gibson and Suzuki

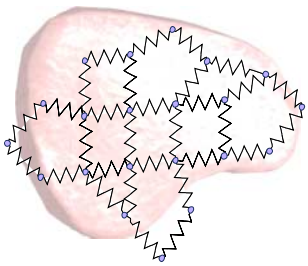


[Gibson 1996]



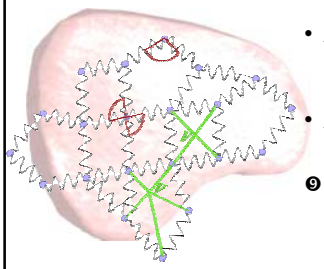
[Suzuki et al. 1997-2000]

Mass-Spring/Damper Systems



- Arbitrary Topology
- Very difficult to fit to physical data (impossible?)
- Forces defined on edges (pairs)
- Can be over- or underdetermined

Generalized Particle System



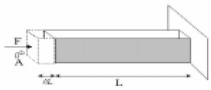
- Additional forces to fix shortcomings: Shear springs, Bend springs
- Additional non-spring forces, like volume preservation

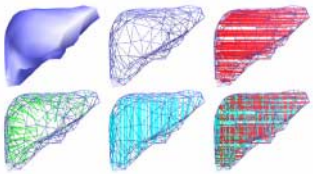
❾ Generalized Particle System

(Forces between particles)

LEM/REM

- Physical simulation of a Long Bar filled with an incompressible fluid.
- Object decomposed into bar elements





[Balaniuk et al. 2000, 2001, 2002; Laugier et al. 2002]

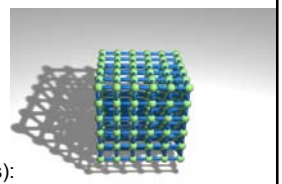
Images taken from Balaniuk and Laugier

Finite Differences / Finite Element Method

- Toolbox to cast a physical formulation into a simulation
- Sound derivation from continuum mechanics
- Standard method in engineering
- Computationally expensive

Finite Differences Method (FDM)

$\boxed{\text{div}\ \sigma - f = 0}$ results in a PDE

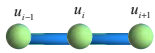


- u sampled on *regular* grid
- differential quotients

➡ difference quotients (stencils):

$$\frac{\partial u_i}{\partial x} = \frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}}$$

$$\frac{\partial^2 u_i}{\partial x^2} = \frac{u_{i+1} - 2u_i - u_{i-1}}{(x_{i+1} - x_{i-1})^2}$$



©Images M. Hauth

FDM: Dynamics

Quasi-static description:

$$K(u)\ u - f = 0 \quad \Rightarrow \quad \text{Solve LES}$$

Adding mass and damping matrices:

$$M(u)\frac{d^2u}{dt^2} + D(u)\frac{du}{dt} + K(u)u - f = 0$$

$$\Leftrightarrow \quad \frac{d^2u}{dt^2} = -M(u)^{-1}\left(D(u)\frac{du}{dt} + K(u)u - f\right)$$

Energy

(Elastic) Energy function:

$$\Pi = \tfrac{1}{2}\int_K \varepsilon(u) : \sigma(u)\,\mathrm{dx}$$

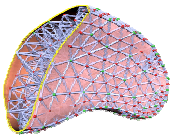
In a non-inertia setting, the configuration of the elastic solid will always be a minimum of the (elastic) energy

Hence: $\Pi \to \min! \quad \Leftrightarrow \quad \frac{\partial \Pi}{\partial u} \overset{!}{=} 0$

Finite Elements

Approximation with finite d.o.f., with basis φ_i („shape functions“)

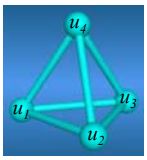
$$u \to \sum u_i \varphi_i$$



time dependent problem $\Rightarrow$ **ODE**

For example: *Linear* trivariate functions :

$$\vec{u}(\vec{x}) = \vec{\mu}_0(1 - x_1 - x_2 - x_3) + \vec{\mu}_1 x_1 + \vec{\mu}_2 x_2 + \vec{\mu}_3 x_3$$



©Images M. Hauth, TU Darmstadt

FE: Putting it together

Shape function: $\vec{u}(\vec{x}) = \vec{\mu}_0(1 - x_1 - x_2 - x_3) + \vec{\mu}_1 x_1 + \vec{\mu}_2 x_2 + \vec{\mu}_3 x_3$

Strain tensor: $\varepsilon_{ij} = \frac{1}{2}\left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \sum_{k=1}^{3}\frac{\partial u_k}{\partial x_j}\frac{\partial u_k}{\partial x_i}\right]$

Stress tensor: $\sigma_{ij} = \delta_{ij}\lambda(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}) + 2\mu\varepsilon_{ij}$

Energy: $\Pi(\vec{\mu}_0, \vec{\mu}_1, \vec{\mu}_2, \vec{\mu}_3) = \tfrac{1}{2}\int \varepsilon(\vec{u}) : \sigma(\vec{u}) = V\sum_{i,j=0}^{3}\varepsilon_{ij}\sigma_{ij}$

Variation: $f_{i,j} = \frac{\partial \Pi}{\partial \mu_{i,j}}$

& sum up over all elements

Summary

Finite Differences:

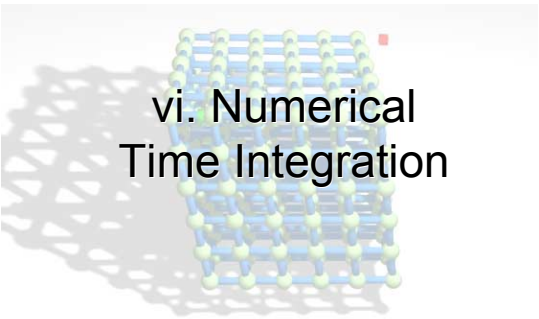
- Simple and easily optimized
- Moderate numerical approximation quality
- Difficult to fit to arbitray geometry
- Problem: Unstructured meshes

Finite Elements:

- Good numerical approximation quality
- For linear shape functions: $\varepsilon(u)$=const per tetrahedron!
- Good shape fitting properties

Outline

i. Physical Properties and Measurement Techniques
ii. Mathematical Modeling
iii. Physical Modeling and Material Laws
iv. Spatial Discretisation
v. Time Integration
vi. Some Current Simulators



vi. Numerical Time Integration

General ODE

2nd order mechanical system:

$$\frac{d}{dt}x(t) = v(t)$$
$$m\frac{d}{dt}v(t) = f_v(t, x, x')$$

Reduction to 1st order

$$\frac{d}{dt}\begin{bmatrix} x(t) \\ v(t) \end{bmatrix} = \begin{bmatrix} v(t) \\ \frac{1}{m} f_v(t, x, v) \end{bmatrix}$$

Abstract form

$$\boxed{\frac{d}{dt}y(t) = f(t, y(t))}$$

Discretize: Solve for time steps

Evaluating a Method

Criteria:

- Convergence timestep →0
- Accuracy u_{approx} - u
- Stability solution → ◎
- Efficiency

Numerical Integration

Abstract problem:

$$y' = f(t, y)$$

Goal:

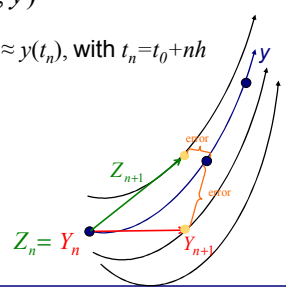
Find approximations $Y_n \approx y(t_n)$, with $t_n = t_0 + nh$

Explicit Euler

$$Y_{n+1} = Y_n + hf(t_n, Y_n)$$

Implicit Euler:

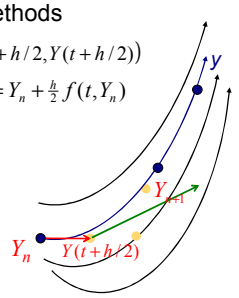
$$Z_{n+1} = Z_n + hf(t_{n+1}, Z_{n+1})$$



Time integration methods

Explicit Methods

$$Y_{n+1} = Y_n + hf\big(t+h/2, Y(t+h/2)\big)$$
$$Y(t+h/2) = Y_n + \tfrac{h}{2} f(t, Y_n)$$



Time integration methods

Explicit Methods

$$Y_{n+1} = Y_n + hf\big(t+h/2, Y(t+h/2)\big)$$
$$Y(t+h/2) = Y_n + \tfrac{h}{2} f(t, Y_n)$$

Generalisation:

a_{ij} (coeff.), b_i (weights), c_j (abscissae).

$$k_i = Y_n + h\sum_{j=1}^{s} a_{ij} k_j', \quad k_i' = f(t_n + c_i h, k_i), \; i = 1,..,s$$

$$Y_{n+1} = Y_n + h\sum_{i=1}^{s} b_i k_i'$$

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & 1 \end{array}$$

explicit midpoint

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} \end{array}$$

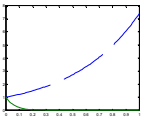
"The" RK4 method

Analysis

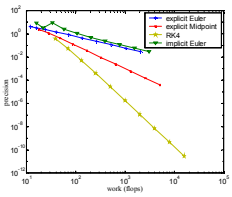
$y' = \lambda y \quad \lambda \in \{2, -15\}$

$y(0) = 1$

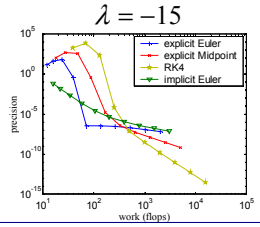
Solution: $y(t) = e^{\lambda t} y$



$\lambda = 2$



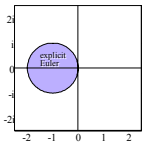
$\lambda = -15$



Linear Stability Analysis

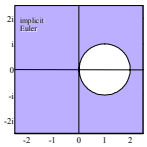
Explicit Euler

$$Y_n = (1 + h\lambda)^n Y_0 \quad < \infty$$
$$\Leftrightarrow \quad |1 + h\lambda| < 1$$



Implicit Euler

$$Z_n = (1 - h\lambda)^{-n} Z_0 \quad < \infty$$
$$\Leftrightarrow \quad |1 - h\lambda|^{-1} < 1$$



For fixed λ condition for stepsize h

stable for h, λ :⇔ solution of Dahlquist is bounded

A-stable :⇔ stable for all Re $\lambda < 0$ and $h > 0$

Relevance of Dahlquists Equation

Abstract setting: $y' = f(t, y)$

Linearise locally around a solution by Taylor expansion:

$$y' = f(t, y) = f(t, \tilde{y}(t)) + \frac{\partial}{\partial y} f(t, \tilde{y}(t))(\tilde{y}(t) - y(t)) + \ldots$$
$$\approx const + J(\tilde{y}(t) - y(t)) + \ldots$$

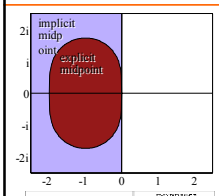
Use an eigenbasis of J to obtain

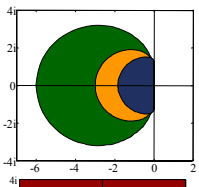
$$\hat{y}_n' = \lambda_n \hat{y}_n \quad n = 1 \ldots N$$

Observation: **All** *passive physical systems* come to **rest**

$\Rightarrow \quad y(t) < \infty \quad$ *and* $\quad \operatorname{Re} \lambda < 0$

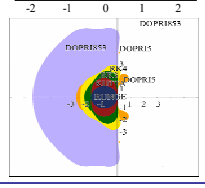
Stability regions

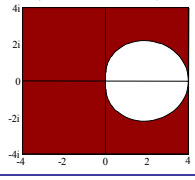




Adams Methods:
(order k+1)

$$Y_{n+1} = Y_n + h\sum_{j=0}^{k} \beta_j f_{k-j+1}$$





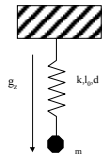
BDF Methods:
(order k)

$$\sum_{j=0}^{k} \alpha_j Y_{k-j+1} = h\beta_k f_1$$

Example 2

A simple spring system:

$$\frac{d^2 z}{dt^2} = \frac{k}{m}(l_0 - z) + \frac{d}{m}\frac{dz}{dt} + g_z$$



Reduction to 1st order:

$$\frac{d}{dt}\begin{bmatrix} z \\ v_z \end{bmatrix} = \begin{bmatrix} 0 & I \\ -\frac{k}{m}I & \frac{d}{m}I \end{bmatrix}\begin{bmatrix} z \\ v_z \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k}{m}l_0 + g_z \end{bmatrix}$$

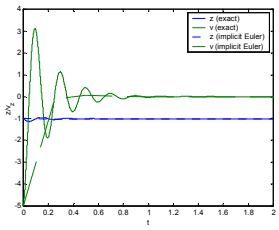
Parameter:

$$k = 1000, l_0 = -1, d = 10, g = -10, m = 1$$

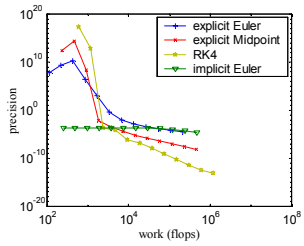
Example 2 Analysis

$$\frac{d^2 z}{dt^2} = \frac{k}{m}(l_0 - z) + \frac{d}{m}\frac{dz}{dt} + g_z$$

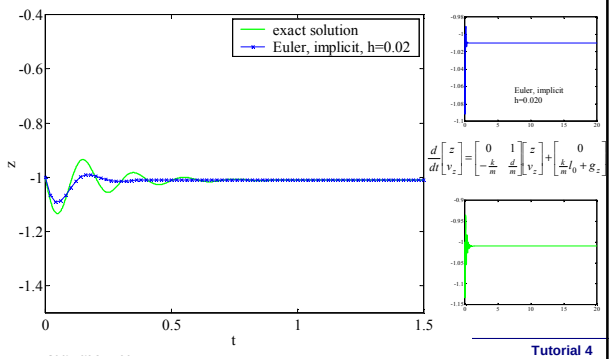
Analytical and a numerical solution



Precision-work

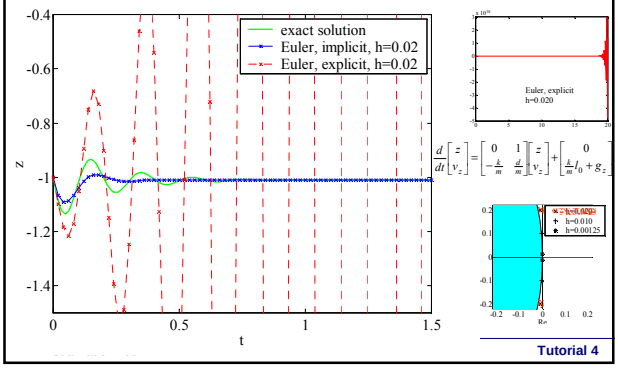


In Detail: Stability for Ex.3



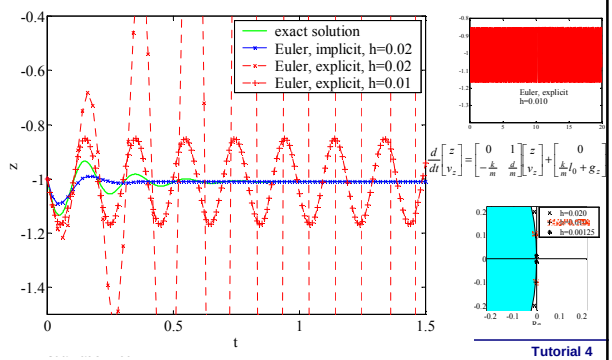
$$\frac{d}{dt}\begin{bmatrix} z \\ v_z \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & \frac{d}{m} \end{bmatrix}\begin{bmatrix} z \\ v_z \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k}{m}l_0 + g_z \end{bmatrix}$$

In Detail: Stability for Ex.3



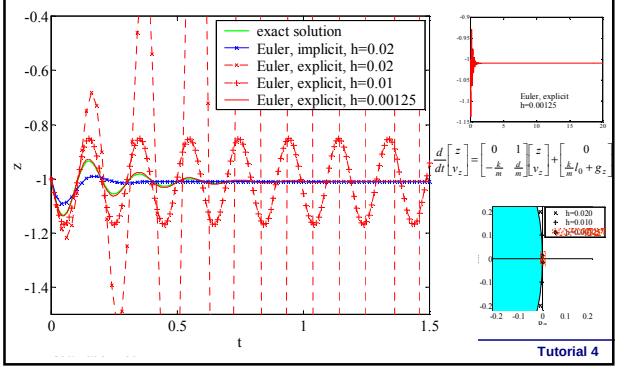
$$\frac{d}{dt}\begin{bmatrix} z \\ v_z \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & \frac{d}{m} \end{bmatrix}\begin{bmatrix} z \\ v_z \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k}{m}l_0 + g_z \end{bmatrix}$$

In Detail: Stability for Ex.3



$$\frac{d}{dt}\begin{bmatrix} z \\ v_z \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & \frac{d}{m} \end{bmatrix}\begin{bmatrix} z \\ v_z \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k}{m}l_0 + g_z \end{bmatrix}$$

In Detail: Stability for Ex.3



$$\frac{d}{dt}\begin{bmatrix} z \\ v_z \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & \frac{d}{m} \end{bmatrix}\begin{bmatrix} z \\ v_z \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k}{m}l_0 + g_z \end{bmatrix}$$

Summary: Integration Methods

Implicit methods:

- ↗ arbitrary large time steps
- ↘ need to solve large (non-)linear systems

Explicit methods:

- ↘ Have *strict* time step restrictions
- ↗ only need a few function evaluations

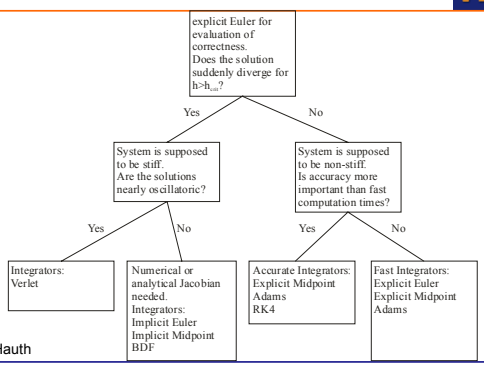
For **elastic equations**:

- ↗ Implicit methods are superior due to large time steps

But they are computationally expensive

- ↘ Because the Jacobian changes due to the nonlinearity

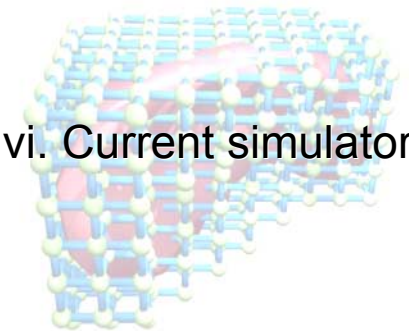
Selecting a Method



© M.Hauth

Outline

i. Physical Properties and Measurement Techniques
ii. Mathematical Modeling
iii. Physical Modeling and Material Laws
iv. Spatial Discretisation
v. Time Integration
vi. Some Current Simulators



vi. Current simulators

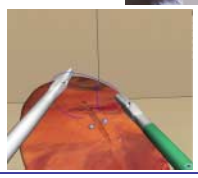
Some Simulators

State of the art:

- Mist — Commercial
- Kismet / Surgical Science / Xitact
- INRIA Epidaure — Research
- Lasso / CoMe

MIST by Mentice

- Modules available also from Simsurgery
- Focuses on *basic* skills and tasks
- Studies show improved learning
- No forcefeedback, thus comparably *cheap*

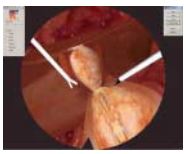


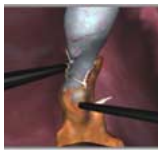
©Images Mentice/ Simsurgery

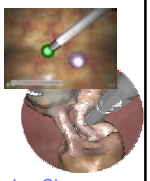
Commercial: VsOne (Kismet) / LapSim / Xitact / ReachIn

- Supposed to use Mass Spring and/or variations of LEM
- Include basic lessons
- Mostly cholecystectomy modules
- Force Feedback

Xitact LS500

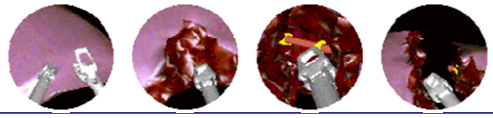

VSOne


ReachIn


LapSim

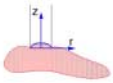
INRIA/Epidaure

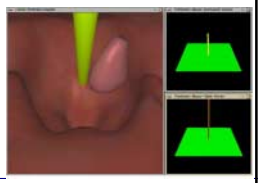
- Since 1992, includes tissue modelling as subproject
- Finite elements and equivalent tensor Mass model
- Anisotropic Hook law
- Force extrapolation for haptic loop



Lasso / CoMe

- Two large Swiss National projects (Lasso 1996-2000, CoMe 2001-)
- Soft tissue part focuses on hysteroscopy
- Includes measurement subproject
- Uses partly finite elements but also investigates customized mass spring models
- Includes automated texture and pathology generation (e.g. fibroids), and special effects (e.g. bleeding)





Summary

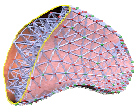
- Cheap basic simulators
- Current commercial simulators lack tissue fidelity
- In comparison to flight simulators, simulators are in the early 1930s (flight simulators were accepted in the mid 50s)
- Research simulators are maturing, using almost exclusively the finite element toolbox

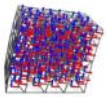
The VirTis Simulator

by WSI/GRIS, Uni Tübingen,
M. Hauth, J. Gross, W. Strasser

Overview

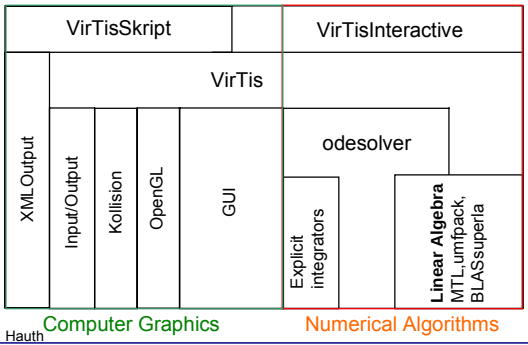
- Tetrahedral finite elements with linear shape functions
- Uses measured data as Prony series, different material laws, e.g. constQ
- Implicit or explicit time integration
- Haptic feedback with PHANToM
- Solves the rate problem by a local buffer model
- Optionally: Uses co-rotational strain





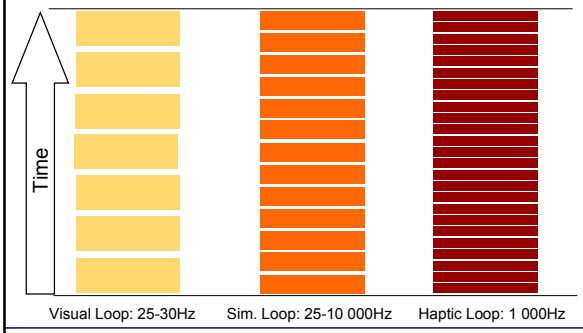
©Images M. Hauth

Architecture



© M. Hauth

The rate problem

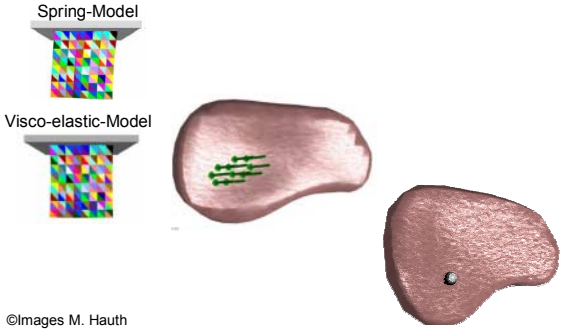


Local Buffer Model

- Simplified geometrical and physical model in the haptic loop.
- Different techniques possible:
  - planes, spheres or a triangle subset as geometric model
  - constant, extrapolated or physical proxy for force computations
- Local Buffer is updated from the simulation loop.

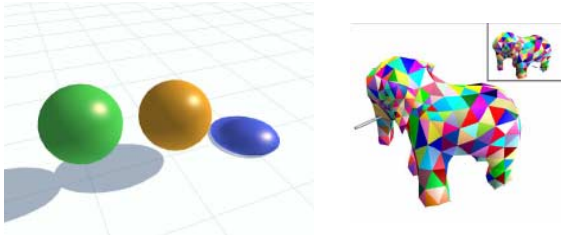
[Zilles and Salisbury 1994, Ruspini et al. 1997, Balaniuk 2000]

Demo Movies



©Images M. Hauth

Demo Movies



©Images M. Hauth

Summary: The VirTis Simulator

Provides real time finite elements with

- 5 memory parameters
- ~ 1000 (full Green strain tensor) – 5000 (co-rotational tensor) elements in real time on a first generation P4@2GHz

Theorie and Applications

Ph.D. Thesis Michael Hauth

Directions of further work

- (Better in vivo measurement devices)
- (Better/more complex force feedback devices)
- Cutting
- Adaptive Subdivision/Coarsening
- Numerical Methods (FEM, time integrators)
- Evaluation of haptic perception
- Parallelization
-

Outline

Introduction

Foundations

Applications of Virtual Medicine

Motion Simulation

Soft-Tissue Simulation /Deformation

Question and Answers

Question and Answers

- Was tutorial content helpful?
- Are you missing a topic(s)
- ....

Acknowledgements (Bartz)

Thanks to

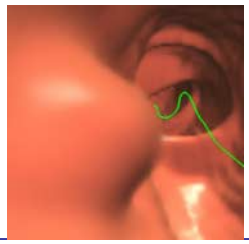
collaboration partners

- Anxo del Rio, Jan Fischer, WSI/GRIS - VCM, Univ. Tübingen
- Özlem Gürvit, Univ. Hospital Marburg
- Dirk Mayer, Univ. Hospital Mainz
- Dirk Freudenstein, Jürgen Hoffmann, Martin Skalej, Florian Dammann, Andreas Kopp, Univ. Hospital Tübingen

and

- Joachim Hornegger, Siemens Medical Systems
- Rainer Wegenkittl, TIANI MedGraph
- Roel Truyen, Philips Medical Systems
- Karel Zuiderveld, Vital Images

Acknowledgments (Mueller)

Prof. Arie Kaufman and collaborators for various images used in this presentation

NIH, NSF, ONR