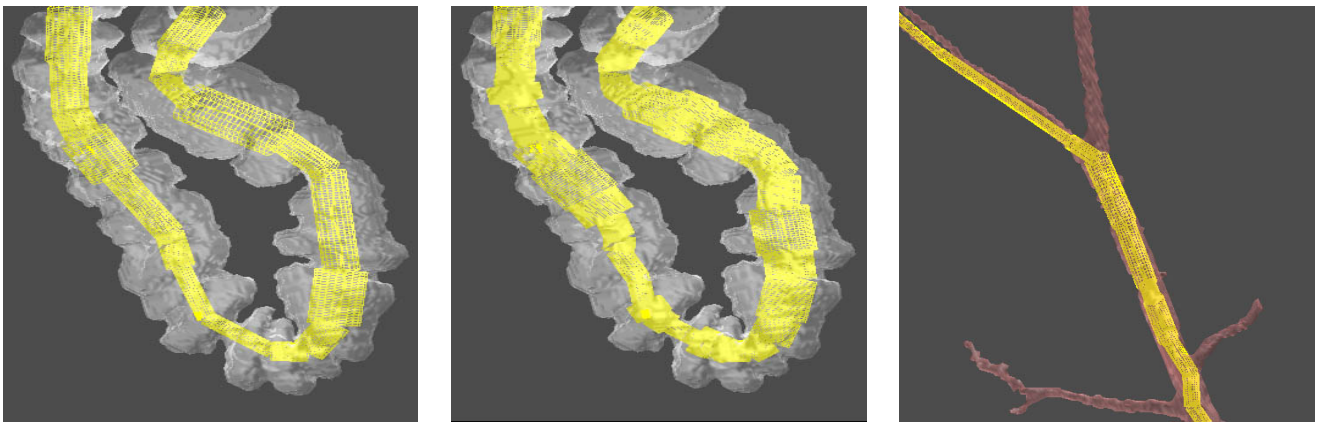
Cylindrical Approximation of Tubular Organs for Virtual Endoscopy

Anna Vilanova Eduard Gröller Andreas König

* Institute of Computer Graphics
Vienna University of Technology



a) b) c)

Figure 1: Surface of the segmented organ together with the cylindrical approximation: a) CT colon data. b) The same CT colon data as in a but using *a* different wall thickness. c) CT kidney data set. The segmented organ is the aorta.

Abstract

Virtual endoscopy is a promising medical application of volume visualization techniques. A virtual endoscopy system requires high quality and perspective projection rendering, as well as real-time navigation. In this paper the generation of a cylindrical structure for tubular shaped organs (i.e. colon, aorta) is presented. This structure represents an approximation of the real organ. The cylindrical structure will be used to accelerate high quality volume rendering.

Keywords: Volume Rendering, Virtual Endoscopy, Space Leaping, Bounded Cylinders

1 Introduction

The continuous improvement of medical imaging techniques, like CT and MRI, has led to the investigation of different methods to visualize the acquired volumetric data. Virtual endoscopy is one of the most promising medical applications using volume rendering techniques. It can be defined as the exploration of hollow organs and anatomic cavities using computer visualization techniques. Its fields of application, amongst others, are diagnosis avoiding the risks associated with a real endoscopy, training, and exploration of organs that cannot be reached with a real endoscope.

Several virtual endoscopy systems have been proposed in recent years [5, 15, 18]. These systems are concerned with navigation inside the volume data and therefore with fast volume rendering. A virtual endoscopy system has mainly the following components: segmentation of the organ, user interaction, and rendering.

* email:{anna,meister,koenig}@cg.tuwien.ac.at, http//:www.cg.tuwien.ac.at

Dealing with medical data implies that the quality of the rendering results has to be high. Unfortunately the methods which produce high quality images in volume rendering are also computationally expensive.

There are basically two methods to visualize volume data: surface based techniques and direct volume rendering. Surface based rendering is fast since it enables the use of common graphics hardware acceleration. The volume data is replaced by a polygonal intermediate structure which represents the organ surface [10]. This method implies the loss of quality and information due to the segmentation step. Furthermore, since just the surface of the organ is visible, there are medical procedures which cannot be performed in a surface based model (e.g. decide whether a polyp is benign or not). Moreover, the surface volume rendering looks artificial especially in places where the polygons are near the view point (e.g., narrow tubes).

The second visualization technique, called direct volume rendering, produces a projected image directly from the volumetric data without using a segmentation step or intermediate constructs. A transfer function, which maps the medical volume data to material properties (e.g., color and opacity) has to be defined as well as the optical model to be used [11].

The image is computed by integrating (i.e. compositing) along each viewing ray. There are different approaches to compute this integration; usually it is sampling along the ray with a distance smaller than a voxel and compositing the sample results (i.e. raycasting) [7]. The processing of a single sample point is expensive due to interpolation and shading calculations.

Different software techniques to accelerate direct volume rendering have been proposed [1, 3, 6, 20, 21]. Some techniques are based on space leaping which avoids sampling in empty space regions. These methods achieve speed-up without affecting the image quality. In some other techniques there is a tradeoff between image

quality and speed. There are also some special hardware acceleration techniques like 3D Texture Mapping [2] and the VolumePro system [13]. The VolumePro system enables volume rendering in real-time but currently has the disadvantage of not producing perspective views, which are necessary when the camera is inside the volume. The perspective projection also reduces the possibility of using other algorithmic acceleration techniques which exploit the coherence of parallel rays in parallel projection views [22].

Therefore, most of the systems that use high quality direct volume rendering for virtual endoscopy compute an off-line movie sequence.

In this paper, an algorithm that generates cylindrical structures for tubular shaped organs (e.g., colon, vessels) is presented. These structures roughly approximates the organ cavity (see figure 1). These structures are used to accelerate high quality direct volume rendering for virtual endoscopy using space leaping.

The next section gives a brief description of the previous work done in this field; in section 3, the outline of the method is presented. Section 4 describes the algorithm to generate the cylindrical structures. In section 5, the algorithm that uses the cylinder structure to accelerate volume rendering is described. The results are presented in section 6. Conclusions and future work are discussed in section 7.

2 Previous Work

Space leaping avoids the computation of those voxels that are not contributing to the final image. A good overview about these techniques can be found in Šrámek's thesis [17].

A group of space leaping approaches exploits object space coherence. The voxels that do not contribute to the final image usually occupy connected regions. These voxels can be joined in macro regions or into hierarchical structures (e.g., pyramid [8]).

A space leaping approach based on the distance transform was introduced by Zuiderveld et al. [24]. The distance transform [9] operation is applied to a binary volume data (e.g., segmented volume) and its result is a distance map. A distance map is a volume data of the same size as the original volume which for every background voxel saves the distance to the nearest object voxel. The distance map can also be computed to save for every object voxel the distance to its nearest background voxel. The distance map can be used to accelerate ray-casting increasing the distance between samples along a ray when there is empty space. Wan et al. [21] presented a distance transform approach which was used for virtual colonoscopy.

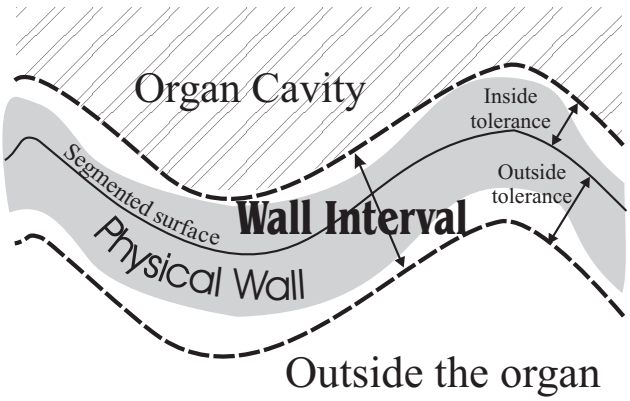


Figure 2: Sketch representation of a 2D cut of an organ. It shows the wall and the cavity of the organ together with a possible wall interval definition.

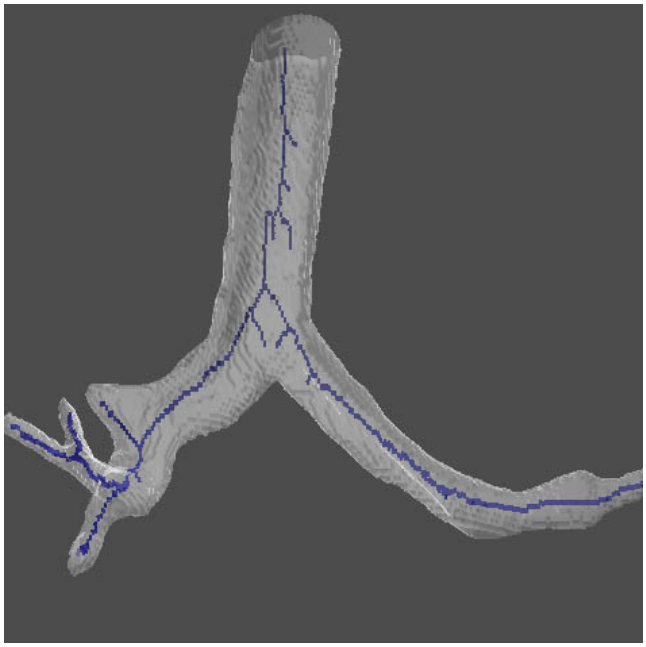


Figure 3: Surface of a segmented trachea together with its skeleton.

Sobierajski et al. [16] propose a polygon assisted ray-casting for volume rendering. Their approach generates a polygon bounding box from the faces of the cells that contains the external surface of the object. The bounding box is rendered in Z-buffer graphics hardware. The Z-buffer is then used to identify the first voxel which contributes to the result image. Similar approaches where introduced by other authors [23, 19, 20]

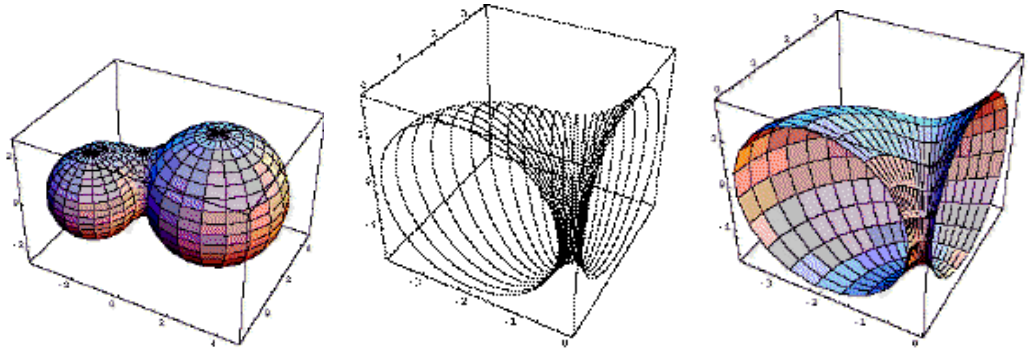
In our approach we use a combination of the approach proposed by Sobierajski and distance transform techniques. In virtual endoscopy the camera is situated inside the organ. We generate a structure formed by cylinders which approximate the organ cavity. This structure is used to accelerate ray-casting by identifying the ray pieces that are traversing voxels inside the cavity of the organ, and therefore, do not contribute to the final image.

Generating cylindrical structures from volume data has been described before by other authors (e.g.,Puig et al. [14]). In Puig et al. approach the cylindrical structures are used as symbolic trees of blood vessels. This structures give a sketch representation of the vessels, and they are used, for instance, to identify changes in the diameter of the vessels (i.e., to detect a stenosis).

3 Method Overview

The method presented in this paper is proposed for virtual endoscopy applications where the organ has a tubular shape and the physicians want to visualize the organ surface and its vicinity (i.e. the organ wall). This situation arises in many endoscopy applications, like colonoscopy or bronchoscopy.

Our algorithm is based on the definition of a safety region around the physical organ wall. This safety region will be called the wall interval of the organ. We define the wall interval of an organ as the approximated organ surface (surface obtained from a segmentation) together with two tolerance values (see figure 2). The tolerance values are two distances from the approximated surface, one in the direction of the organ cavity and one pointing to outside of the organ. The physical organ wall to be visualized must be enclosed within the wall interval, since only what is within the wall interval will be rendered. The tolerance values of the wall interval are currently given by the user as parameters of the presented method.



a) b) c)

Figure 4: Canal Surfaces: a) 1-parameter spheres family b) Circles of intersection c) Canal surface

The algorithm can be divided in the following steps:

- Segmentation of the organ to inspect (e.g., colon, trachea).
- Calculation of the distance map and the skeleton of the segmented organ.
- Cylindrical approximation of the organ cavity.
- Accelerated direct volume rendering using ray-casting.

The segmentation is applied to the volume data to identify the organ and obtain a binary volume. The segmentation does not have to be highly accurate. The algorithm will correct the segmentation error using the tolerance values of the wall interval (see figure 2). Therefore, the tolerance values must be an overestimation. They should include the wall thickness and the possible error generated in the segmentation step.

Using the segmented volume, the organ skeleton is computed (see figure 3). A skeleton is the locus of points that are geometrically centered with respect to the border. The skeleton can be automatically calculated from the segmented volume using topological thinning or distance transform methods [4].

The skeleton is also used to find the camera path which will be used for the camera navigation metaphor and therefore for the 3D interaction.

The cylindrical approximation of the organ cavity and the accelerated direct volume rendering will be discussed in detail in the next sections.

4 Cylindrical Approximation

In this section, the algorithm to generate the cylindrical approximation of the organ cavity is described. The input of the algorithm are the skeleton and the distance map of the segmented object. The distance map contains for each object voxel the minimum distance to the background.

The main idea about using a cylindrical approximation of the organ started with the canal surfaces [12], which are commonly used in geometric modeling. A canal surface is the envelope of a 1-parameter family of spheres, whose radius and position change continuously. The envelope of the spheres is defined as the union of all circles of intersection of infinitesimally neighboring pair of spheres (see figure 4). The envelope represents the surface of the object.

Given the skeleton and the distance map, a piece-wise continuous canal surface can be defined. The family of spheres is defined with the centers on the skeleton and the radii equal to the minimal distance to the nearest background voxel.

Using ray-casting as rendering technique, we would like to use the canal surface as a bounded structure to calculate the pieces of the viewing rays that cross the cavity of the organ. Unfortunately the canal surfaces require an expensive ray-intersection calculation. Therefore, we will approximate the canal surface to a set of primitives with a fast ray-intersection, i.e., cylinders. It is also possible to use other approximations apart from cylinders, such as generalized cylinders, but the overhead due to more costly intersection calculation would increase.

The cylindrical approximation to be used for space leaping should fulfill the following requirements:

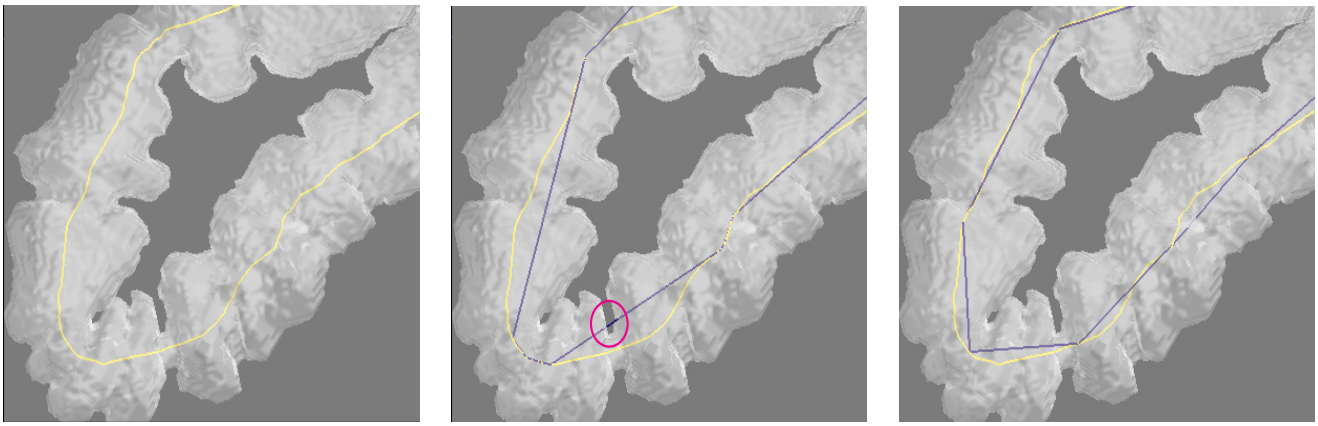
1. The cylindrical structure is entirely enclosed in the approximated organ and does not intersected the wall interval of the organ.
2. It should have a limited number of cylinders. With a high number of cylinders the intersection procedure with the structure will be too expensive.
3. The cylinders should be as long as possible to enable long jumps, especially for central rays. This is quite advantageous in the typical case when the camera is looking straight into the organ cavity.
4. The cylinders should have large volumes to cover a considerable portion of empty space within the organ cavity.
5. The structure shall closely approximate the organ cavity. The intersection point of a viewing ray with the cylinder structure shall be close to the wall interval of the organ (i.e., close to where the sampling of the volume data must start).

The cylinders do not have to be connected and they can also overlap. However, overlapping reduces efficiency in the intersection calculation.

A cylinder is defined by a radius, and an axis, given by position, direction and length. These parameters are assumed to be variables of an optimization problem to find the optimal cylindrical structure. This optimization problem should be solved using characteristics 1-5 as optimization criteria. However, the optimal solution is complex and time-consuming to achieve since there are too many variables. Therefore an heuristical approach has been used.

The goal is to have bounded cylinders that fit as much as possible into the organ cavity. We use the skeleton as a base to define the cylinder axes. Keeping the axes in the center as much as possible will give fewer and larger cylinders. Once the cylinders axes has been defined we define the radius of the cylinders using the distance transform.

The creation of the cylinder structure can be divided into two steps:



a) b) c)

Figure 5: Surface of the segmented organ together with the cylinder axis generation based on the central path. **a)** Central path. **b)** Central path, cylinder axes generated from curvature information. The circle shows an area where an axis is outside of the organ. **c)** Central path, cylinder axes generated according to distance errors.

- Cylinder axis definition.
- Cylinder radius definition.

4.1 Cylinder Axis Definition

The axes of the cylinders will be generated based on the skeleton. The skeleton obtained by either topological thinning or resulting from the distance transform is a set of 26-connected voxels. From the skeleton a central path is defined. For simplicity, we deal with paths which do not have branches. An ordered sequence V_i of voxels that go from a starting voxel until a target voxel of the skeleton is obtained. Two consecutive voxels are 26-connected and there is no cycle, so a voxel is represented at most once. The central path is then a polyline whose edges are the junction of the center points, P_i, of consecutive voxels.

This path is usually highly folded and noisy. In order to reduce the high frequencies and to smooth the path a low pass filter is applied. If m is the kernel dimension and n is the number of path points, with $0 \leq i \leq n-m$:

$$P_i = \sum_{h=0}^{m-1} P_{i+h} * w_h \;\; where \;\; \sum_{j=0}^{m-1} w_h = 1$$

The weights w_h of the kernel are determined using a tent filter. However, other low pass filters could have also been used.

The maximum variation of the new path compared with the old one depends on the kernel size. If the kernel size increases then the path is smoother but the path also moves from the center of the hollow organ.

Once filtered we have a smooth path represented by a polyline with a large amount of small edges. The polyline is approximated by longer edges to be used as axes for the cylinders. Furthermore the variation from the original path has to be controlled in order that the axes are not moving outside the organ.

One approach to create the cylinder axes is undersampling the central path by taking into account the curvature of the path. In the areas of the polyline with lower curvature the axes can be long and in the areas with higher curvature we need more but shorter axes. The number of cylinders that the structure shall contain is defined by the user with a value s. The curvature in the different points of the central path is computed. Then the first and last points of the central path together with the $s-1$ points with highest curvature are joined in the path order to create the axes.

This method has the problem that s is not easy to estimate. The axes are also not well distributed since they are concentrated in the areas where the curvature is high. Furthermore there is no control of the deviation between the original path and the axes. Therefore it can happen that the axes are partially outside the organ if s is not large enough (see figure 5b).

A second approach which prevents the axes to move outside of the organ was used. A distance error ε is specified as the maximum distance difference between the original path and the cylinder axes. The error ε is defined as a portion of the minimum distance from the central path to the wall interval of the organ. In this way the axes are generated ensuring that they will stay within the organ cavity (see figure 5c).

The algorithm treats the central path points sequentially (see figure 6). Path pieces of the central path, sub-paths of the central path, are consecutively used to generate the axes. It is ensured that the distance difference between a sub-path and the corresponding axis is not larger than the defined error ε.

A sub-path is a sequence of consecutive points of the central path. An axis A_j is created joining the first and the last points of the current sub-path. $\varepsilon_{i,j}$ is defined as the distance of the point P_i to the axis A_j. If for each point P_i of the current sub-path the distance $\varepsilon_{i,j}$ to the axis A_j is smaller than ε, then the next point of the central path is added to the sub-path. This process is repeated until an iteration in which there exists a P_i whose $\varepsilon_{i,j}$ is greater than ε. Then the axes generated in the previous iteration, A_{j-1}, is defined as a valid axis. The sub-path is erased. The next current sub-path is defined by the last point of the axis A_{j-1} and its next two consecutive points in the central path. Initially the current sub-path contains the first point of the central path together with the next two consecutive points. If in the first iteration already $\varepsilon_{i,0}$ is greater than ε then the valid axis is created between the first and the second point of the sub-path.

In figure 5 we observe that with the method that uses error distance the resulting axes are better distributed along the path than the curvature based method.

4.2 Cylinder Radius Definition

Once the axes for the cylinders have been defined, the radius of the cylinders have to be specified. Every cylinder axis is sampled with a rate smaller than the voxels distance. For each sample point, the distance map is used to determine the distance to the nearest wall point. The radius of each cylinder is fixed as the minimum distance

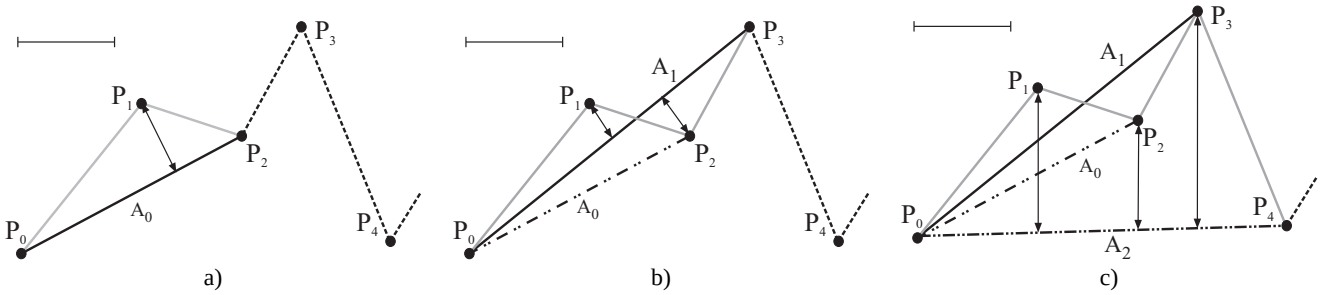


Figure 6: Illustration of the axes generation algorithm using the distance error information. A_j is the axis generated in iteration j; P_i denote the central path points. The central path is drawn as a discontinuous line and the current sub-path is painted in grey. 3 iterations of the algorithm are shown. In this example the final valid axis would be A_1.

from the axis to the wall. If the radius of a cylinder is smaller than a size defined by the user (e.g., one voxel size), this cylinder is discarded.

Some examples showing the cylinder structure can be seen in figure 1.

5 Volume Rendering Accelaration using the Cylindrical Structure

The cylindrical structure can be used to accelerate volume rendering using space leaping. In this section we describe how this can be done and we discuss the use of the distance map for early ray-termination.

The cylindrical structure generated is used to calculate the ray segments which go through empty space. Therefore for each ray the intersection points with the cylindrical structure are calculated. Besides, frustum culling is applied to avoid the intersection with cylinders that are not visible for the current camera.

If the camera is inside the cylinder then the first intersection is taken as the first possible sampling point. Afterwards, the distance map is used to jump until the distance to the wall interval is smaller than the sample rate, and then the usual slow compositing is applied [21]. It may happen that the ray enters another cylinder again. In that case the next intersection point in the list of intersections between the viewing ray and the cylinder approximation is used as a possible sampling point. If the camera is outside the cylinder a usual distance jumping is applied till the ray enters a cylinder or is close to the interval wall. Figure 7 illustrates how the algorithm works for three sample rays.

The most common early ray-termination technique terminates the ray if the accumulated opacity approaches one. In our approach another early ray-termination condition is added. Assuming that we want to sample points just within the wall interval of the organ, we do not want to go on sampling the rays in areas outside the interval wall. To determine when the ray is outside the wall interval, we calculate a distance map with the distance of the background voxels to the nearest object voxels. The algorithm consults the distance map to decide if the ray has to be terminated at a given sample point. This early ray-termination approach cannot be used if the organ tissue is rendered transparently and the region outside of the organs shall be visualized.

6 Results

Figure 1a and 1b shows the result of applying the cylinder structure generation to a colon. Figure 1c shows the cylinder approximation of one central path in the aorta (for the moment the algorithm does not use branches). Figure 1a and 1b present the difference if the

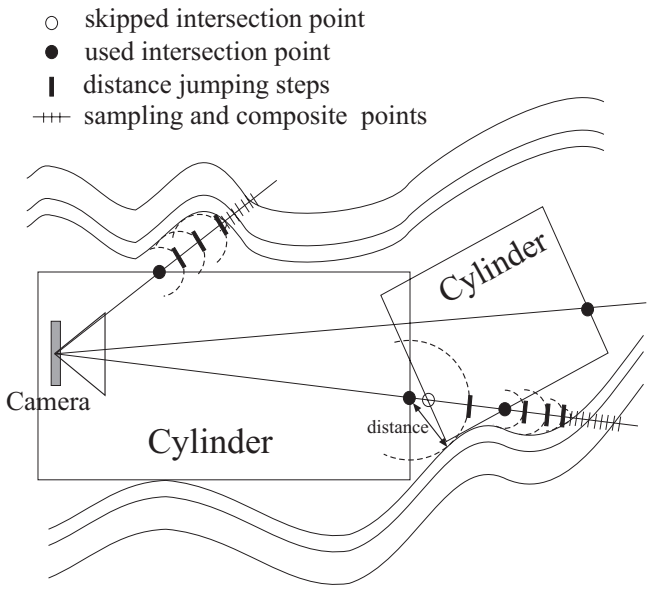


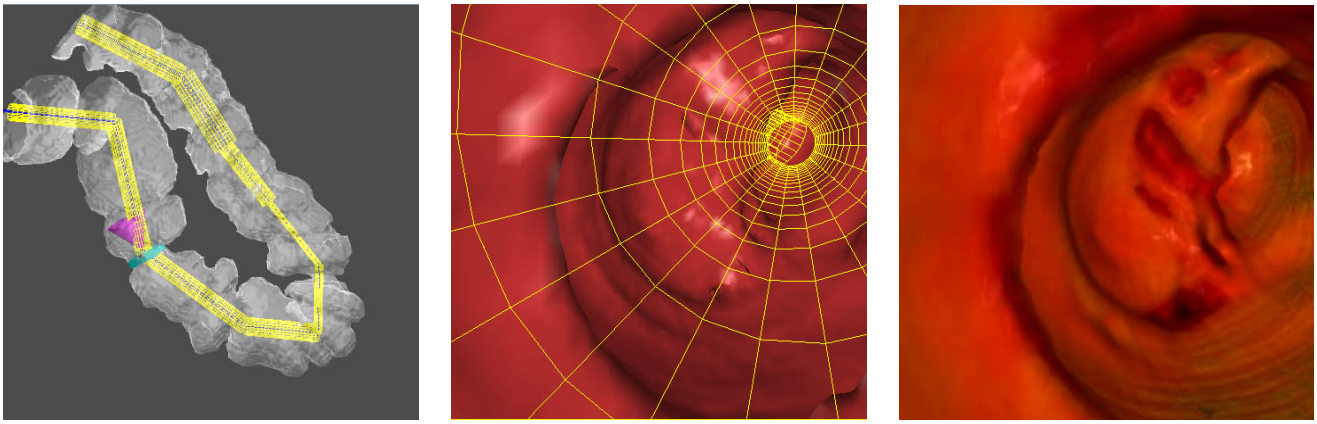
Figure 7: Illustration of the volume rendering algorithm using the cylindrical approximation structure.

wall interval tolerance values are modified (Figure 1a has more inside tolerance of the wall interval than 1b). In figure 1b the structure fits better to the organ cavity but the number of cylinders generated also increases. Figure 8a illustrates for the same data as figure 1a and 1b the effect of decreasing the ε value.

In figure 8, a virtual endoscopy example for a 200x115x300 CT colon data set is shown. Figure 8a shows an outside view of the colon portion to be inspected and the camera position. The endoscopic view in 8b shows a surface extracted using marching cubes together with the cylinder structure which contains about 9 cylinders. Figure 8c presents the result of applying direct volume rendering using the cylindrical approximation with the same result as applying a brute force approach. However, the rendering time of the 256x256 image using the cylindrical structure was four times faster than with the brute force approach.

In figure 9, a 205x83x105 CT dataset of a trachea is used. The rendering time was 3.5 times faster than the direct volume rendering using a brute force approach for a 256x256 image and using 14 cylinders.

The performance of the presented algorithm depends on several parameters, such as the transfer function, the organ shape, the number of cylinders generated and the characteristics of the cylinders.



a) b) c)

Figure 8: Virtual endoscopy of a 200x115x300 CT dataset of a colon : a) Outside view of the segmented object surface together with the camera and the cylindrical approximation. b) Endoscopic view of the cylindrical structure and a surface based rendering. The polygons have been generated using marching cubes. c) Endoscopic view using direct volume rendering and the cylindrical approximation.

To obtain good results these parameters have to be tuned carefully.

If the algorithm presented by Wan [21] is extended to use the presented wall interval concept, and also use the proposed early ray-termination, then in our implementation the performance of this algorithm is similar to the presented cylindrical approximation. The advantage of the cylindrical structure is that it has the potential of being used for other purposes, for example, being a sketch representation of the organ cavity.

7 Conclusions and Future Work

We have presented an algorithm to generate a cylindrical approximimation for the cavity of tubular like organs. This approximation has been used as bounded cylinders to accelerate direct volume rendering for virtual endoscopy. An early termination criteria for virtual endoscopy has also been presented. The acceleration achieved does not allow real-time direct volume rendering on a common PC (i.e., Pentium II 400MHz), but accelerates the generation of high quality precalculated movies.

One research direction to follow is the automatization of the parameter tuning. Other structures, like general cylinders, to obtain a better approximation of the organ cavity, will be studied further , as well as new generation algorithms to obtain better approximations.

The cylindrical structure might also be helpful for polygon assisted ray-casting as outlined in [16].

The cylindrical structure also has the potential of being used for image based rendering. Currently we are investigating some approaches in this direction.

Acknowledgements

The work presented in this publication has been funded by the $V^{is}M^{ed}$ project. $V^{is}M^{ed}$ is supported by *Tiani Medgraph*, Vienna (`http://www.tiani.com`), and the *Forschungsförderungs fonds für die gewerbliche Wirtschaft*, Austria.

See `http://www.vismed.at` for further information on this project.

We thank Katja Bühler for her information about canal surfaces.

We thank Dr. Martin C. Freund and the Department of Radiology at Leopold-Franzens-University of Innsbruck for their collaboration and for providing some of the data used in this paper.

References

[1] M. L. Brady, K. K. Jung, H. T. Nguyen, and T. PQ. Nguyen. Interactive volume navigation. *IEEE Transactions on Visualization and Computer Graphics*, 4(3):243–255, July – September 1998.

[2] B. Cabral, N. Cam, and J. Foran. Accelerated volume rendering and tomographic reconstruction using texture mapping hardware. In *1994 Symposium on Volume Visualization*, pages 91–98. ACM SIGGRAPH, October 1994.

[3] J. Danskin and P. Hanrahan. Fast algorithms for volume ray tracing. In *Proceedings of the Workshop on Volume Visualization*, pages 91–98, New York, October19–20 1992. ACM Press.

[4] N. Gagvani. Skeletons and volume thinning in visualization. Master's thesis, New Brunswick, Rutgers, The State University of New Jersey, 1997.

[5] L. Hong, S. Muraki, A. Kaufman, D. Bartz, and T. He. Virtual voyage: Interactive navigation in the human colon. In *SIGGRAPH 97 Conference Proceedings*, Annual Conference series, pages 27–34. ACM SIGGRAPH, Addison Wesley, August 1997.

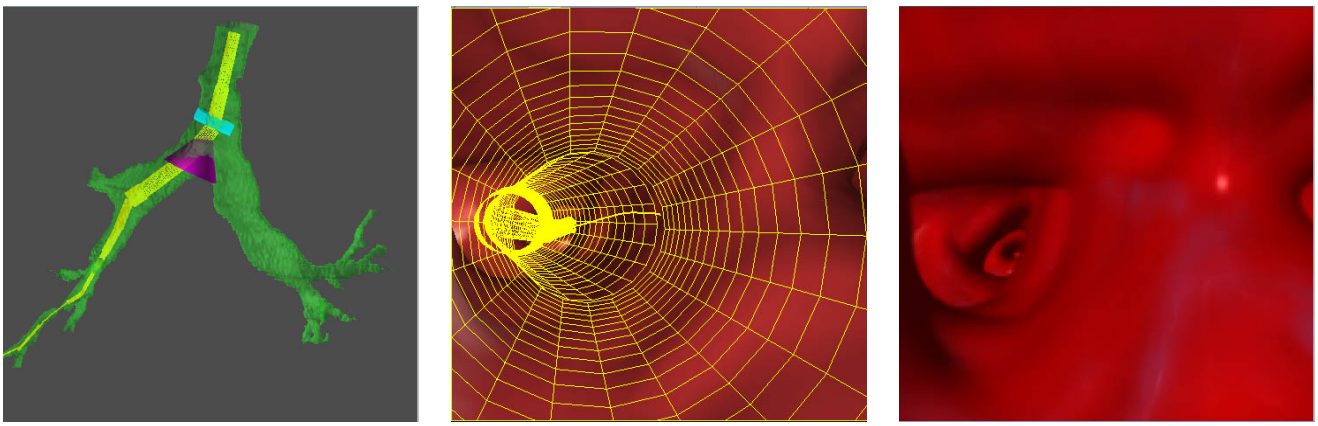
[6] P. Lacroute and M. Levoy. Fast volume rendering using a shear–warp factorization of the viewing transformation. In *SIGGRAPH' 94 Conference Proceedings*, pages 451–458. ACM SIGGRAPH, ACM Press, July 1994.

[7] M. Levoy. Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, 8(3):29–37, February 1987.

[8] Marc Levoy. Efficient ray tracing of volume data. *ACM Transactions on Graphics*, 9(3):245–261, July 1990.

[9] G. Lohmann. *Volumetric Image Analysis*. Chichester Wiley, 1998.

[10] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In Maureen C. Stone, editor, *SIGGRAPH 87 Conference Proceedings*, pages 163–169. ACM SIGGRAPH, Addison Wesley, July 1987.



a) b) c)

Figure 9: Virtual endoscopy of a 205x83x105 CT dataset of a trachea: a) Outside view of the Segmented object surface together with the camera and the cylindrical approximation. b) Endoscopic view of the cylindrical structure and a surface based rendering. The polygons have been generated using marching cubes. c) Endoscopic view using direct volume rendering and the cylindrical approximation.

[11] N. Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995.

[12] M. Paluszny and K. Buehler. Canal surfaces and inversive geometry. In *Mathematical Methods for Curves and Surfaces II*, pages 367–374. Vanderbilt Univ. Press, 1997.

[13] H. Pfister, J. Hardenbergh, J. Knittel, H. Lauer, and L. Seiler. The VolumePro real-time ray-casting system. In *ACM Siggraph'99*, pages 251–260, 1999.

[14] A. Puig, D. Tost, and I. Navazo. An interactive cerebral blood vessel exploration system. In Roni Yagel and Hans Hagen, editors, *IEEE Visualization '97*, pages 443–446. IEEE, November 1997.

[15] R.M. Satava and R.A. Robb. Virtual endoscopy: Application of 3D visualization to medical diagnosis. *Presence*, 6(2):179–197, April 1997.

[16] L. M. Sobierajski and R. S. Avila. Hardware acceleration method for volumetric ray tracing. In *IEEE Visualization'95*, pages 27–34. IEEE, 1995.

[17] M. Šrámek. *Visualization of Volumetric Data by Ray Tracing*. PhD thesis, Technische Universität Wien - Institut für Computergraphik, 1996.

[18] A. Vilanova, A. König, and E. Gröller. VirEn: A virtual endoscopy system. *Machine GRAPHICS & VISION*, 8(3):469–487, 1999.

[19] M. Wan, S. Bryson, and A. Kaufman. Boundary cell-based acceleration for volume ray casting. *Computers and Graphics*, 22(6):715–722, December 1998.

[20] M. Wan, A. Kaufman, and S. Bryson. High performance presence-accelerated ray casting. In *IEEE Visualization '99*, pages 379–386. IEEE, nov 1999.

[21] M. Wan, Q. Tang, A. Kaufman, Z. Liang, and M. Wax. Volume rendering based interactive navigation within the human colon. In *IEEE Visualization '99*, pages 397–400. IEEE, nov 1999.

[22] R. Yagel and A. Kaufman. Template-based volume viewing. *Computer Graphics Forum*, 11(3):C153–C167, 1992.

[23] S. You, L. Hong, M. Wan, K. Junyaprasert, A. Kaufman, S. Muraki, Y. Zhou, M. Wax, and Z. Liang. Interactive volume rendering for virtual colonoscopy. In Roni Yagel and Hans Hagen, editors, *IEEE Visualization '97*, pages 433–346. IEEE, nov 1997.

[24] K.J. Zuiderveld, A.H.J. Koning, and M. A. Viergever. Acceleration of ray-casting using 3D distance transform. In *Visualization in Biomedical Computing II. SPIE 1808*, pages 324–335, 1992.