Colon Centerline Calculation for CT Colonography using Optimised 3D Topological Thinning

Robert J.T. Sadleir and Paul F. Whelan
Vision Systems Laboratory, School of Electronic Engineering,
Dublin City University, Dublin 9, Ireland.
{Robert.Sadleir, Paul.Whelan}@eeng.dcu.ie

Abstract

CT colonography is an emerging technique for colorectal cancer screening. Using this technique virtual reality models of the human colon are generated from an abdominal CT study of a suitably prepared patient. Manual navigation through these models is a slow and tedious process. It is possible to automate navigation by calculating the centerline of the human colon. There are numerous well documented approaches for centerline calculation, most of these have been developed as alternatives to 3D topological thinning which is regarded as being far too computationally intensive for use with large volumetric datasets. This paper describes a fully automated, optimised version of 3D topological thinning that has been specifically developed for calculating the centerline of the human colon. This algorithm is fast, portable and robust. Initial tests have been performed on real patient data and the results are promising.

1. Introduction

Colorectal cancer is a major cause of cancer related death in developed countries. Statistics published by the *National Cancer Registry of Ireland* (NCRI) [14] indicate that colon cancer accounted for 9% of all cancer cases diagnosed in 1997, second only to non-melanoma skin cancer (36%). Colorectal cancer can be prevented if precursor polyps are detected early in their course and successfully resected. Regular screening is required in order to detect the presence of colorectal polyps. At present the most sensitive screening technique is conventional colonoscopy. This involves an endoscopic examination of the colonic mucosa using an instrument known as a colonoscope.

There are several problems associated with conventional colonoscopy. The patient must undergo complete bowel evacuation prior to the examination, in addition the patient must be placed under sedation for the duration of the procedure. The examination itself is extremely invasive and can lead to complications. The most serious complication is bowel perforation, which can, in extreme cases, result in death. The conventional colonoscopy examination is embarrassing and uncomfortable for the patient and has achieved limited acceptance among the those at risk of developing colorectal cancer.

CT colonography (CTC) [21] (also known as *virtual colonoscopy* (VC)) is an emerging technique for imaging the interior of the colon in a noninvasive manner. CTC provides an alternative to conventional colonoscopy for colorectal cancer screening. Using this technique, a suitably prepared patient undergos an abdominal CT scan. The patient is generally scanned in both the prone and supine position in order to reduce the effects of residual material in the colon [6]. 3D virtual reality models of the colon are then generated and inspected in a manner similar to conventional colonoscopy. CTC datasets can be automatically analysed to assist in the polyp detection process [15].

Manual navigation through a virtual reality model of the colon is both time consuming and awkward. It is possible to automate intraluminal navigation by calculating the centerline of the colon. The centerline is a single string of connected points that links the two endpoints of the colon. Each point must be located at the furthest possible point from the colon wall. The centerline can subsequently be used to guide the observer through the colon lumen, thus substantially reducing the time required to evaluate the colon. The centerline can also be used to generate fly-through movies of the virtual colon. This is achieved by creating renderings at regular intervals along the centerline.

The problem of colon centerline calculation has generated great interest in the research community. A large number of colon centerline calculation techniques have been proposed. Each technique requires different levels of user interaction and has different performance characteristics. This paper describes a novel alternative for centerline calculation. The proposed technique is fully automated and uses

Group	Year	Technique	Platform	CPU(Mhz)	RAM(MB)	Automatic	Time(s)
Deschamps & Cohen [9]	2001	Distance field	Sun	300	1000	✗	30
Bitter et al. [1]	2001	Distance field	Intel	1000	NA	✓	119
Chen et al. [4]	2000	Distance field	SGI	2x195	896	✗	36
Samara et ak. [17]	1999	Region Growing	SGI	NA	NA	✗	300
Zhou et al. [24]	1999	Voxel coding	SGI	NA	NA	✗	519
Ge et al. [10]	1999	3D topological thinning	SGI	NA	NA	✗	518(60†)
Horwich et al. [12]	1999	3D topological thinning	SGI	NA	1000	✓	<900
McFarland et al. [13]	1997	Radiologist marking	NA	NA	NA	✗	1080

† Centerline calculation time obtained through using subsampling.

Table 1. An overview of previously published centerline calculation algorithms

an optimised version of 3D topological thinning to calculate centerlines in a robust and extremely efficient manner. Initial tests have been carried out using real patient datasets. The test results indicate that our approach is a viable alternative for fast colon centerline calculation.

2. Previous Work

A large number of centerline calculation algorithms are described in the literature. The majority of these algorithms have been developed as efficient alternatives to 3D topological thinning which is generally regarded as being overly computationally intensive [1, 17, 24]. Some centerline calculation techniques do utilise 3D topological thinning, however, optimisation techniques must be employed in order to increase efficiency. Ge et al. [10] describe one such approach where subsampling is used to increase the efficiency of standard 3D topological thinning. This stratedgy increases performance by a factor of five, however, the accuracy of the result is ultimately compromised. The following paragraphs review a selection of previously published approaches for centerline calculculation.

Early techniques for centerline calculation required significant user interaction. McFarland et al. [13] describe a semiautomated technique where a radiologist identifies keypoints in the colon lumen. A cubic spline fit of these points is then calculated to approximate the colon centerline. Semiautomated techniques for colon centerline calculation are slow, inaccurate and require excessive user interaction. Modern techniques require substantially less user interaction and generate a more accurate approximation of the colon centerline in significantly less time.

Samara et al. [16] calculate the colon centerline using the center of mass of grown voxels. Two approximations of the centerline are calculated by region growing from seed points in the rectum and the caecum (the rectum and caecum are the two most distal points in the colon). These forward (antegrade) and reverse (retrograde) centerlines are then averaged in order to generate the final result. More recently this technique has been extended [17]. The plane perpendicular to each point in the centerline is determined. The center of mass of lumen voxels is calculated for each plane. These centers of mass are then connected to represent the ultimate centerline.

Chiou et al. [7, 8], Bitter et al. [3, 1] and Hong et al. [11] use centerline calculation techniques based on distance field analysis. The approach described by Hong et al. involves the computation of two seperate distance fields relative to a point $\mathbf{X}$ located in the colon lumen. These fields describe the distance from the surface of the colon ($D_s(\mathbf{X})$) and the distance from a target point located in the ceacum/appendix ($D_t(\mathbf{X})$). These distance fields are combined to generate a force field ($V(\mathbf{X})$) which represents an attraction to the target point in the ceacum/appendix and a repulsion from the colon wall. This force field is used to guide an observer along the centermost path of the colon.

These techniques, as well as others described in the litrature, are compared in Table 1. The best results are associated with the technique described by Deschamps and Cohen [9], however, this level of performance is achieved using high performance hardware.

3. Segmentation

Our centerline calculation algorithm requires a binary model of the air insufflated colon lumen. A CTC dataset contains an extremely large amount of information. In addition to the colon it can also represent sections of the lungs, kidneys, liver and stomach. The process of identifying voxels that represent the colon lumen from a CTC dataset is referred to as segmentation. The majority of segmentation techniques utilise seeded 3D region growing. The initial seed point(s) can be either user defined [16, 17] or automatically detected [23, 18]. There are alternative techniques for colon lumen segmentation. Chen et al. [5] describe a novel approach to extract the colon lumen using *principle component analysis* (PCA). Unfortunately this technique is rather computationally intensive and requires approximately 9 minutes to execute on an SGI Octane workstation with dual R10000 CPUs and 890MB of RAM. We

Material	Density (HU)
Air	-1500
Fat	-40
Water	0
Soft Tissue	80
Calcium	400
Metal	2000

Table 2. A range of common materials and their corresponding density values represented using Hounsfield units (HU).

have developed a segmentation technique that is based on seeded 3D region growing. Our technique incorporates automatic detection of the colon endpoints. These endpoints are ultimately used in the centerline calculation process (see Section 4).

Segmentation of the colon lumen is dependant on the ability to differentiate between voxels that represent air and voxels that represent other materials such as soft tissue and bone. This distinction is made by comparing the value of a particular voxel with a threshold. The threshold is determined by examining the difference between the density of the colon wall (approximately equal to soft tissue) D$_{colon}$ and the density of air D$_{air}$. The threshold density D$_{thresh}$ is taken to be approximately midway between these two values (see Equation 1). The units of density are measured using *Hounsfield units* (HU) and the densities of several common materials are presented in Table 2.

$$D_{thresh} \approx D_{air} + \frac{D_{colon} - D_{air}}{2} \quad (1)$$

By substituting the relevant values into Equation 1 it is apparent that the value for D$_{thresh}$ is approximately -790 HU, although a value of -800 HU is more commonly used [23, 10]. In subsequent sections any voxels with densities below the threshold value D$_{thresh}$ are taken as representing air while voxels with densities above this threshold are taken to represent solid material e.g. the colon wall. D$_{thresh}$ is the only constant threshold in our centerline calculation system. The use of this threshold does not represent a major problem as the values that are compared with it are always either much higher or much lower than the threshold itself. The only time the density of a particular voxel approaches that of D$_{thresh}$ is as a result of the *partial volume effect* (PVE). In this case the voxel in question is located on the air/tissue border and as such can be classified as either air or tissue without significantly affecting the outcome of the segmentation process.

The seed point for region growing is automatically detected from the CTC dataset using *a priori* knowledge about the data acquisition technique. This seed point is referred to as the *rectum point* (R_p) as it is located in the rectum. In addition to acting as the seed point for region growing the rectum point also represents one of the endpoints of the centerline of the colon.

We use a segmentation technique that is based on seeded 3D region growing. The initial seed point is the rectum point. Standard 3D region growing identifies contiguous voxels with values below a certain threshold. The threshold for colon lumen segmentation is D$_{thresh}$ as defined in Equation 1. We use a six neighbour kernal to reduce the effects of shine through. Shine through can cause oversegmentation. Our implementation of the region growing algorithm includes a slight modification. This modification allows us to detect the other endpoint of the colon centerline while performing segmentation. The second endpoint is referred to as the *caecum point* (C_p) as it is located in the caecum. The deliverables of the segmentation stage are twofold. In addition to generating a segmented version of the colon which is required for thinning the segmented stage also automatically detects the endpoints of the centerline, thus reducing the amount of user interaction required to calculated the centerline of the colon.

4. Centerline Calculation

Calculation of the centerline of the human colon is an extremely important part of the CTC process. The centerline of the colon is used to automate intraluminal navigation. It has been demonstrated that automated path planning increases the sensitivity of CTC while substantially reducing the amount of time required to evaluate the colon [19]. Bitter et al. [2] define the centerline of the colon as:

> *The shortest path between the two most distal points inside the colon that includes heavy penalties for coming close to the colon wall.*

We propose a fully automated centerline calculation algorithm based on 3D topological thinning as described by Tsao and Fu [20]. Centerline calculation using this approach is accurate, however, it has been discounted by others due to its computationally intensive nature [1, 17, 24]. Our novel approach enhances standard 3D topological thinning by employing optimisation techniques, thus significantly increasing performance. This performance increase is achieved without compromising accuracy.

Ge et al. [10] describe centerline calculation using 3D topological thinning as the removal of successive layers of surface voxels until certain topological and geometric constraints are violated. Topology preservation requires that the structure (number of holes) and connectivity of the object being thinned is preserved. The geometric constriants require that all skeleton endpoints are preserved. Our technique is different in this regard, only two endpoints are preserved, these are the precalculated centreline endpoints (R_p, C_p). We divide topology violation testing into two separate tests. These tests ascertain whether the deletion of a particular voxel will result in a connectivity violation or the introduction of a new hole in the object being thinned.

The correct centerline cannot be guaranteed after the completion of the 3D topological thinning algorithm. This is due to the fact that the topology of the original object is preserved. Holes can be present in the original binary model of the colon. These holes cause extraneous loops in the thinned version of the original object due to the topological constraints described above. These loops must be removed in order to generate the final centerline. The remainder of this Section describes the concepts of endpoint retention, connectivity violation testing, detection of new holes and extraneous loop removal.

4.1. Endpoint Retention

Standard 3D topological thinning requires that all skeletal branches are retained. Colon centerline calculation does not have the same geometric constraints and only requires the retention of two branches. The endpoints of these two branches are the rectum point and the caecum point i.e. the two most distal points of the colon. In the case of centerline calculation the geometric constraints simply require the retention of the predefined centerline endpoints.

4.2. Connectivity Violation Testing

While thinning, the removal of a surface voxel (deletion candidate) must not affect the connectivity of the binary object being thinned. Connectivity can be tested at a local level by ensuring that the number of distinct binary objects in the $3 \times 3 \times 3$ neighbourhood centered around the deletion candidate does not increase due to the deletion of the center voxel.

We test for local connectivity breaches using a standard labelling algorithm [22] which has been slightly modified and extended to operate in three dimensions. If the number of labelled objects in the local $3 \times 3 \times 3$ neigbourhood is greater that one after the removal of a deletion candidate then a local connectivity violation will be generated. If this is the case then the deletion candidate cannot be removed. The concept of local connectivity violation testing is illustrated in Figure 1.

4.3. New Hole Detection

Simply avoiding connectivity violations is not sufficient to guarantee the generation of a valid skeleton for a 3D object. It is possible to introduce an unwanted hole by deleting

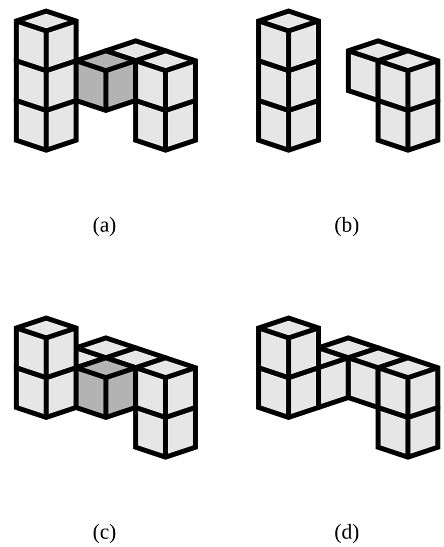


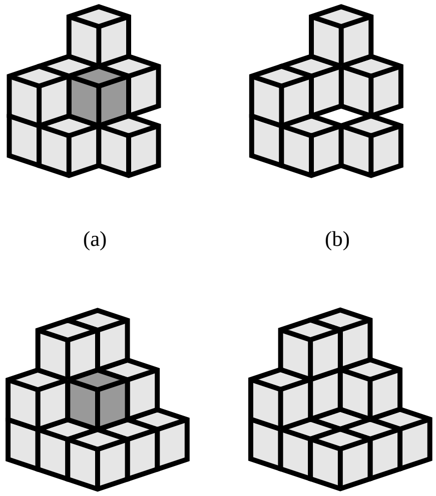
Figure 1. Examples of where the removal of the center voxel violates connectivity (a) & (b) and preserves connectivity (c) & (d).

a voxel that does not generate a local connectivity violation. If this occurs then the topology of the object that is being thinned is altered. It is possible to detect the introduction of new holes at a local level by examining the $3 \times 3 \times 3$ neigbourhood centered about the current deletion candidate.

The test for the introduction of new holes is significantly more complex and time consuming than the test for local connectivity violations. The test itself is based on *ad hoc* principles and requires exhaustive analysis of the local $3 \times 3 \times 3$ surrounding the deletion candidate. The test involves checking for scenarios where, following the removal of the deletion candidate, a clear path exists that passes through the center voxel. If this path is surrounded by a ring of connected foreground voxels in any configuration then a new hole has been introduced. The concept of hole detection is illustrated in Figure 2.

4.4. Extraneous Loop Removal

Standard topological thinning of the colon lumen does not generate a correct centerline. The reason for this is that the calculated centerline includes extraneous loops. These loops are due to holes that are present in the original binary object being thinned. In the case of the colon lumen holes are normally present and are caused due to folds in the colonic mucosa. Because of the local geometric constraints placed on the thinning algorithm these holes are re-



(a) (b) (c) (d)

Figure 2. Examples of where the removal of the center voxel introduces a hole (a) & (b) and does not introduce a hole (c) & (d).

tained in the final centerline in the form of extraneous loops. Extraneous loops can be removed by examining each centerline voxel and sequently removing the voxels closest to the surface. A voxel must not be removed if it causes a global connectivity violation. This can be checked after the removal of each voxel by ascertaining whether or not a path exists between the caecum point and the rectum point. This process is outlined in Figure 3.

5. Optimisation

Centerline calculation as described in Section 4 is extremely inefficient, however, it can be optimised. This Section describes two optimisation techniques that dramatically increase the performance of centerline calculation using 3D topological thinning without compromising the accuracy of the resulting centerline.

5.1. Surface Voxel Tracking

3D topological thinning only examines the surface voxels of the object that is being thinned. A surface voxel is a foreground (or lumen) voxel where one of the 6 directly connected neighbours is a background voxel. Repetitive raster scanning of the entire volume to identify surface voxels is unnecessary. We propose a technique similar to that described by Ge et al [10] for tracking surface voxels, thus removing the need to perform unnecessary raster scans of the entire volume. Using this technique a single raster scan of the volume is required to identify the initial set of surface voxels. The co-ordinates of the surface voxels (or deletion candidates) are stored using a Java `Vector` object. This object supports the storage, management and retrieval of any type of data. The thinning process removes successive layers of surface voxels. As each voxel is removed its six directly connected neighbours are examined. If any of these neighbour voxels become surface voxels then their co-ordinates are added to the `Vector` object for inspection during the next thinning iteration. Using this technique, only the voxels of interest or the surface voxels are examined during each iteration of the thinning algorithm. The amount of data processing is substantially reduced when compared with the standard raster scan based approaches.

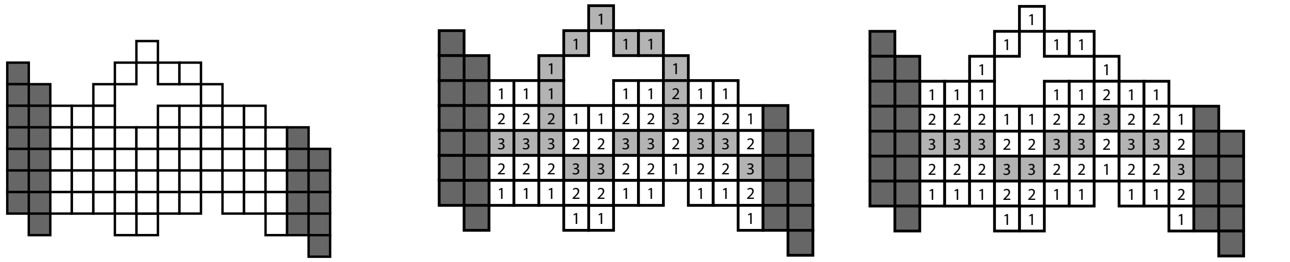
5.2. Partial Precalculation of Results

In order to increase the performance of the local neighbourhood analysis we precalculate the deletion value $d_i(X)$ of the center voxel for each possible $3 \times 3 \times 3$ binary neighbourhood. The deletion value is binary and indicates whether or not the particular voxel can be deleted. This involves the generation of each possible neighbourhood configuration and testing if deletion of the center voxel causes a breach in local topology. The deletion value is subsequently stored in a results table. A unique index is associated with each neighbourhood configuration and this index is generated by examining the values of the neighbouring voxels.

The center voxel in a $3 \times 3 \times 3$ neighbourhood has 26 neighbours. Each of these neighbour voxels is binary and as a result there can be 67,108,864 (2^{26}) possible neighbourhood configurations. Each of these neighbourhood configurations is generated and tested using the techniques outlined in Sections 4.2 and 4.3 respectively. If either violation is generated then the deletion value is false, alternatively if no violation is generated then the deletion value is true. The index for each neighbourhood is unique and is generated using equation 2, $d_i(X)$ represents the deletion value, i represents the neighbour index and $I(n)$ presents the value of the neighbour at index n.

$$d_i(X) = \sum_{n=1}^{26} 2^{n-1} I(n) \qquad (2)$$

Generation of the deletion table is performed once and the same table is used for thinning every CTC dataset. Thus the processes of local neighbourhood analysis is reduced to a simple task of index generation and both local topology tests are effectively performed in parallel. The results table is quite large (67MB), and its size could be reduced using various techniques. However, this would ultimately



(a) (b) (c)

Figure 3. A simple example of extraneous loop removal illustrated using a 2D object. Consider the highlighted segment of the 2D object (a). After thinning and distance field generation we have an extraneous loop (b). This loop can be eliminated by examining each centerline voxel starting with those closest to the surface and removing them provided that the path between the two endpoints of the object is not broken, the final result (c) represents the correct centerline with no extraneous loops.

increase the complexity of index generation, thus reducing the performance of the centerline calculation algorithm.

5.3. Revised Technique

An extremely efficient centerline calculation algorithm has been developed. This level of efficiency has been achieved by optimising conventional 3D topological thinning. Partial precalculation of results replaces the computationally intensive tasks of local topology testing with a simple task of index generation using Equation 2. Surface voxel tracking significantly reduces the amount of data processing required for the detection of new surface voxels after each thinning iteration by identifing any new surface voxels as they appear. The revised centerline calculation algorithm employing these optimisation techniques can be described as follows:

1. Locate all surface voxels by performing an initial raster scan of the entire segmented CTC volume.
2. Store the co-ordinates of all surface voxels in a `Vector` object for further processing.
3. Generate an index for each surface voxel using Equation 2.
4. Obtain the deletion value that corresponds to this index from the results table and flag the voxel for deletion depending on this value.
5. Delete all flagged voxels from the volume and remove their co-ordinates from the `Vector` object. Examine all of the directly connected neighbours for each deleted voxel. If any of these are new surface voxels then add their co-ordinates to the `Vector` object.
6. The predefined endpoints of the skeleton must not be deleted.
7. Continue the thinning process until no more voxels can be deleted.
8. On completion the `Vector` object should contain only the co-ordinates of centerline voxels.
9. Finally the extraneous loops are removed from the centerline to yield a string of minimally connected voxels connecting the two most distal points of the colon.

6. Results

We have performed initial tests using 5 CTC datasets obtained from the department of Radiology in the Mater Misericordiae Hospital, Dublin, Ireland. All algorithms were implemented using Java and executed on a standard PC with a 700Mhz Intel PIII Processor and 512MB of RAM. Segmentation and automatic endpoint detection was successful in all cases. Centerline calculation was then performed, requiring an average of 24.415 seconds. Detailed results are presented in Table 3 and renderings of the colon centerline are presented in Figure 4. These results compare very favorably with those listed in Table 1.

Patient	Sex	Age	Orientation	Volume Size	Thinning Time(s)
1.	Male	68	Supine	512×512×299	23.564
2.	Female	68	Prone	512×512×284	24.746
3.	Female	56	Supine	512×512×244	24.866
4.	Male	67	Prone	512×512×266	23.473
5.	Female	52	Supine	512×512×286	25.424

Table 3. Centerline calculation times for 5 CTC datasets with varied patient sex, age and orientation

7. Conclusions

This paper describes an extremely efficient algorithm for detecting the centerline of the human colon for use with CTC. This technique uses automatically detected centerline endpoints, thus, no user interaction is required. Optimisation techniques that significantly increase the performance of 3D topological thinning are introduced, however, these optimisation techniques do not compromise the acuracy of the resulting centerline. Our technique is fast, portable and robust while requiring a relatively modest hardware platform for execution. The results obtained from initial tests carried out on real patient data compare quite favorably with previously published results (see Table 1).

Acknowledgements

The authors wish to thanks members of the *Vision Systems Laboratory* (VSL) and the *Virtual Reality Applications Laboratory* (VRAL) in Dublin City University particularly Dr. Derek Molloy, Dr. Ovidiu Ghita and Mr. John Mallon. We also wish to acknowledge contributions from our medical colleagues from the Gastrointestinal Unit and Department of Radiology at the Mater Misericordiae Hospital in Dublin, particularly Dr. Helen Fenlon, Dr. Padraic MacMathuna, Dr. Alan Moss and Dr. John Bruzzi.

References

[1] I. Bitter, A. Kaufman, and M. Sato. Penalized-Distance Volumetric Skeleton Algorithm. *IEEE Trans Vis Comp Graph*, 7:195–206, 2001.

[2] I. Bitter, M. Sato, M. Bender, A. Kaufman, M. Wan, and M. Wax. Automatic, Accurate and Robust Colon Centerline Algorithm. *Radiology*, 217:Suppl:705, 2000.

[3] I. Bitter, M. Sato, M.Bender, K. McDonnell, A. Kaufman, and M. Wan. CEASAR: A Smooth, Accurate and Robust Centerline Extraction Algorithm. In *Proc. of IEEE Visualisation 2000*, 2000.

[4] D. Chen, B. Li, . Liang, M. Wan, A. Kaufman, and M. Wax. Tree-branch-searching multiresolution approach to skeletonization for virtual endoscopy. In *Proc. SPIE, Medical Imaging 2000: Image Processing*, volume 3979, pages 726–34, 2000.

[5] D. Chen, Z. Liang, M. Wax, L. Li, B. Li, and A. Kaufman. A novel approach to extract colon lumen form CT image for virtual colonoscopy. *IEEE Trans on Med Img*, 19:1220–6, 2000.

[6] S. Chen, D. Lu, J. Heicht, and B. Kadell. CT Colonography: Value of scanning in both the supine and prone positions. *Am J Roentgenol*, 172:595–9, 1999.

[7] R. Chiou, A. Kaufman, Z. Liang, L. Hong, and M. Achniotou. Interactive Path Planning for Virtual Endoscopy. In *Proc. of the IEEE Nuclear Science and Medical Imaging Conference*, 1998.

[8] R. Chiou, A. Kaufman, Z. Liang, L. Hong, and M. Achniotou. An Interactve Fly-Path Planning Using Potential Fields and Cell Decomposition for Virtual Endoscopy. *IEEE Trans Nucl Sci*, 46:1045–9, 1999.

[9] T. Deschamps and L. Cohen. Fast extraction of minimal paths in 3D images and applications to virtual endoscopy. *Med. Image Anal.*, 5:281–99, 2001.

[10] Y. Ge, D. Stelts, J. Wang, and D. Vining. Computing the Centerline of the Colon: a Robust and Efficient Method Based on 3D Skeletons. *J Comput Assist Tomogr*, 23:786–94, 1999.

[11] L. Hong, S. Muraki, A. Kaufman, D. Bartz, and T. He. Virtual Voyage: Interactive Navigation in the Human Colon. In *Proc. of SIGGRAPH'97*, 1997.

[12] P. Horwich, D. Chen, B. Li, M. Wax, G. Gindi, and J. Liang. The Centerline of the Colon: Automating a Key Step in Virtual Colonoscopy. *Radiology*, 213P:691, 1999.

[13] E. McFarland, G. Wang, J. Brink, D. Balfe, J. Heiken, and M. Vannier. Spiral Computed Tomographic Colonography: Determination of the Central Axis and Digital Unraveling of the Colon. *Acad Radiol*, 4:367–73, 1997.

[14] NCRI. *Cancer in Ireland, 1997: Incidence and mortality*. Healy & Associates, 2000.

[15] R. Sadleir, P. Whelan, P. MacMathuna, and H. Fenlon. Automated detection & flagging of colorectal polyps for virtual colonoscopy: A preliminary study. *Eur Radiol*, 11:suppl:C6, 2001.

[16] Y. Samara, M. Fiebich, A. Dachman, K. Doi, and K. Hoffmann. Automated centerline tracking of the human colon. In *Proc. of the SPIE Conference on Image Processing*, 1998.

[17] Y. Samara, M. Fiebich, A. Dachman, J. Kuniyoshi, K. Doi, and K. Hoffmann. Automated Calculation of the Centerling of the Human Colon on CT Images. *Acad Radiol*, 6:352–9, 1999.

[18] M. Sato, S. Lakare, M. Wan, A. Kaufman, and M. Wax. An automatic colon segmentation for 3D virtual colonoscopy. *IEICE Trans Inf & Sys*, E84D:201–8, 2001.

[19] G. Schmutz, S. Hue, L. Fournier, F. Leproux, and A. Lepennec. Contribution of Automated Path Planning in Virtual Colonoscopy. *Radiology*, 213P:693, 1999.

[20] Y. Tsao and K. Fu. A parallel thinning algorithm for 3-D pictures. *Comp Vis Graph & Im Proc*, 17:315–31, 1981.

[21] D. Vining, D. Gelfand, R. Bechtold, E. Scharling, E. Grishaw, and R. Shifrin. Technical Feasibility of Colon Imaging with Helical CT and Virtual Reality. *Am J Roentgenol*, 162:Suppl:104, 1994.

[22] P. Whelan and D. Molloy. *Machine Vision Algorithms in Java: Techniques and Implementations*, chapter 3, page 103. Springer, 2000.

[23] C. Wyatt, Y. Ge, and D. Vining. Automatic segmenation of the colon for virtual colonoscopy. *Comp Med Im Graph*, 24:1–9, 2000.

[24] Y. Zhou and A. Toga. Efficient Skeletonization of Volumetric Objects. *IEEE Trans Vis Comp Graph*, 5:196–209, 1999.

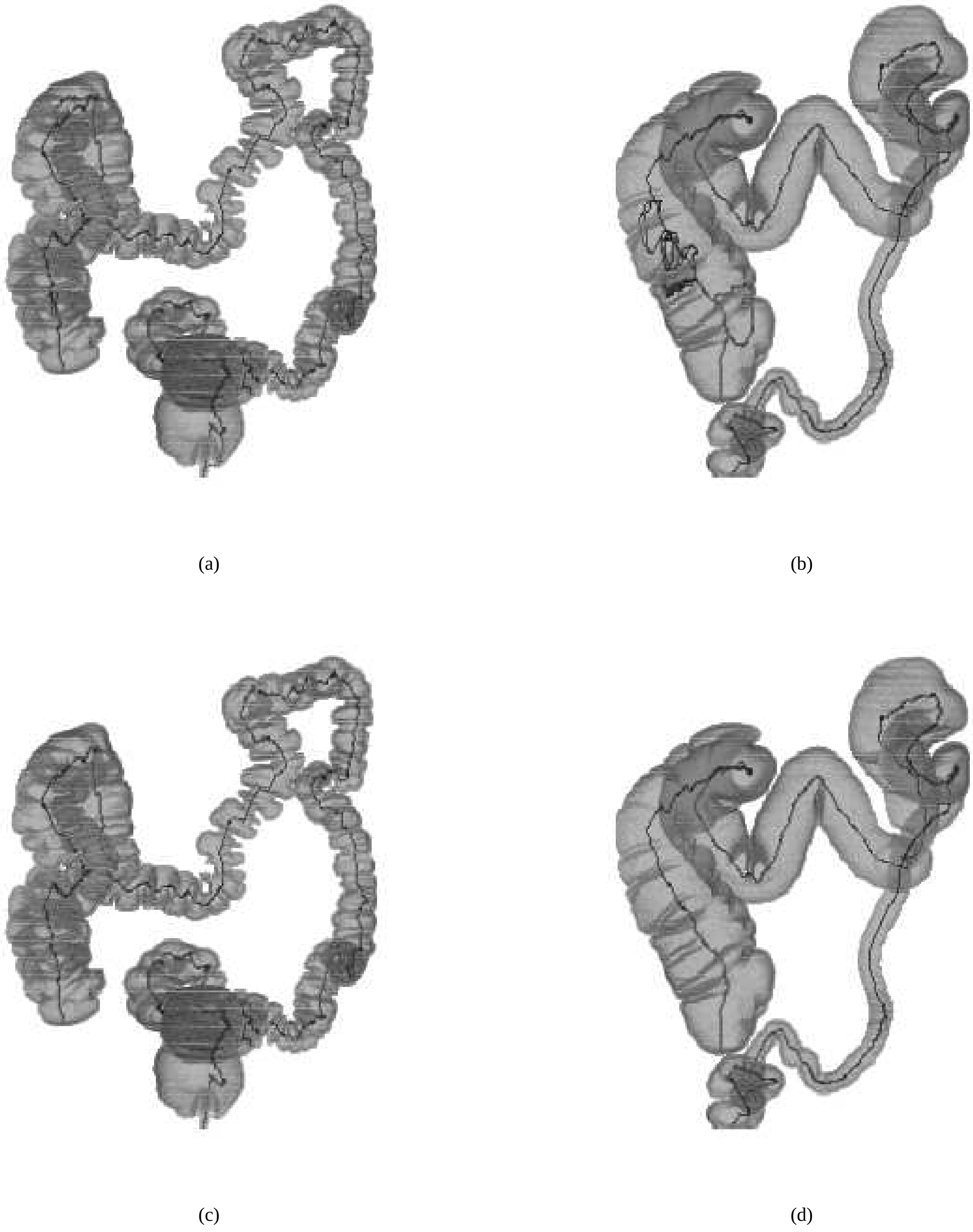


Figure 4. Two examples of colon centerlines generated using the technique described in this paper for patient 1 (a) & (c) and patient 4 (b) & (d). before (a) & (b) and after (c) & (d) the removal of extraneous loops. The actual centerline points are overlaid on a transparent renderings of the relevant colon lumen. Note the deviation in the centerline that is apparent in the rectum for patient 1 is due to the presence of the rectal catheter that is used to inflate the colon.