

Intra-patient Prone to Supine Colon Registration for Synchronized Virtual Colonoscopy

Delphine Nain¹, Steven Haker², W. Eric L. Grimson¹, Eric Cosman Jr¹,
William W. Wells^{1,2}, Hoon Ji², Ron Kikinis², and Carl-Fredrik Westin^{1,2}

¹ Artificial Intelligence Laboratory, Massachusetts Institute of Technology
Cambridge MA, USA

{delfin,welg,ercosman,sw}@ai.mit.edu

<http://www.ai.mit.edu>

² Surgical Planning Laboratory

Harvard Medical School and Brigham and Women's Hospital

{haker,hooni,kikinis,westin}@bwh.harvard.edu

<http://www.spl.harvard.edu>

Abstract. In this paper, we present an automated method for colon registration. The method uses dynamic programming to align data defined on colon center-line paths, as extracted from the prone and supine scans. This data may include information such as path length and curvature as well as descriptors of the shape and size of the colon near the path. We show how our colon registration technique can be used to produce synchronized fly-through or slice views.

1 Introduction

Colorectal cancer is one of the most common forms of cancer, and is associated with high mortality rates. Various screening methods used to detect colorectal cancers and pre-cancerous polyps are available, each with its own costs and benefits. In particular, fiber-optic colonoscopy is a well established and highly effective screening method, but is also invasive, expensive, time consuming and uncomfortable for the patient.

A more recently developed screening method is computed tomographic colonography. In this screening method, a radiologist views a sequence of CT images, typically from one or more axial volumetric scans, and inspects the colon wall for structures likely to be polyps. Keys to identifying these structures can include their shape and cross sectional image intensity profiles. The doctor may also be presented with a 3D reconstructed view of the colon in a process known as virtual colonoscopy. The idea of this approach is to simulate, using computer graphics techniques, the appearance of the colon wall as it would be seen by a doctor performing fiber-optic colonoscopy. This simulation may include a virtual “fly-through” of the colon. Recent work has indicated that these methods have the ability to provide doctors with the information needed to detect small polyps [11]. As CT technology has improved, providing higher resolution images obtained in shorter periods of time and with lower radiation doses to the patient, these virtual methods have become more attractive as routine diagnostic procedures.

The presence of pseudo-polyps, material such as stool which may adhere to the colon wall and appear much like a polyp, can make the task of finding true polyps difficult. In order to better differentiate polyps from pseudo-polyps, and to better view the lumen surface in the presence of fluid, it is common practice to obtain two CT scans of the patient, one with the patient in the prone position and one in the supine. Fluid in the colon will naturally appear on the anterior colon wall when the patient is in the prone position, and on the posterior wall when in the supine. Pseudo-polyp material may also change its position between the two scans, allowing the radiologist to differentiate these structures from true polyps. Further, a second view of the colon after re-positioning may help the doctor determine if a structure is a polyp or simply a fold in the haustra [9].

Naturally, the ability to compare corresponding positions in the prone and supine scans is required before all the benefits mentioned above can be accrued. However, the change in shape and position of the colon within the body between the prone and supine scans can be surprisingly large. Peristaltic action and changes in pressures applied to the body are among the causes of these deformations. The colon, insufflated with room air or carbon dioxide, can behave more like a lightly filled bladder than a rigid structure. Registering the colonic wall between the two scans can therefore be quite challenging. In practice, the radiologist can attempt a manual registration by using anatomical landmarks such as spinal vertebrae to observe images through similar axial planes, and then scroll through adjacent slices to try to find similar structures in the colon wall. Such methods however, are difficult, inaccurate and time consuming.

Figure 1 shows an axial slice that intersects the same vertebra through both the prone and supine scan. As seen on the figure, the colon and other anatomical structures are deformed and shifted due to gravity and pressure factors, and look very different from one scan to the other. The deformation and shift between the colon and other anatomical structures is non-linear, so that surrounding structures are unlikely to provide enough information to match the two colons. An automatic volumetric deformable registration of one entire grayscale scan to the other would be desirable, but is an extremely difficult and time consuming task.

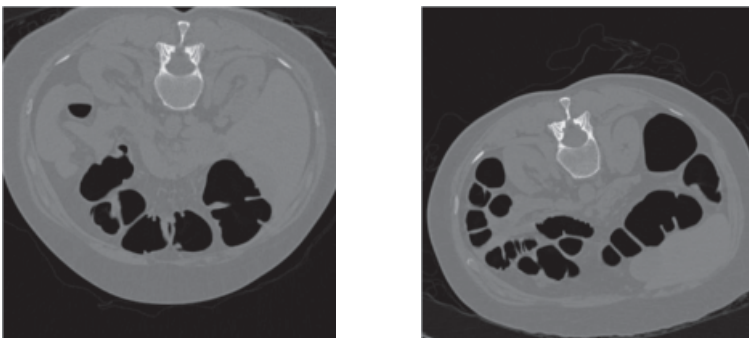


Fig. 1. Axial slices through supine (left) and prone (right) scans. Although the same vertebra is pictured, the colon and other anatomical structures are not aligned.

2 Related Work

Some commercial tools used to view CT image slices have an option to display a supine and prone scan side-by-side, with the option to flip the prone scan, so that images are presented in the same orientation. Effectively, the clinician has to register the two volumes manually in order to find interesting corresponding landmarks.

Acar [1] *et al.* have developed an automatic method that registers the medial axis of supine and prone colon surface models using linear stretching and shrinking operations. They sample centerlines of both colon models at 1mm intervals and examine the inferior/superior coordinates of path points to find local extrema and path inflection points. In order to find corresponding points, they linearly interpolate between inflection points using path length information.

This method only takes into account the local extrema located on the inferior/superior axis. If the colon shifts obliquely when the patient changes position, then some local extrema may not be accounted for. Further, the method does not allow for colon surface information to be taken into account. Information such as radial distance from the medial axis or circumference cannot be used in the matching process. Our method by contrast, is designed to address these points.

3 Registration Methodology

In this section, we present our method for the registration of colon centerlines extracted from prone and supine colon scans. Our method uses dynamic programming and geometric information to find an optimal match between sampled centerpoints. Once the centerlines are matched, fly-throughs are generated for synchronized virtual colonoscopy by stepping incrementally through the matched centerpoints.

3.1 Motivation for Our Approach

The centerline through both colons can provide meaningful information for colon registration, such as length and curvature. Centerline registration is also a simpler and less computationally intensive problem to solve than volumetric registration, since there is only one dimension to match. However, using only centerline information such as length and curvature might not be enough for a robust solution in cases where the colon is severely stretched and deformed between the two scans.

In this case, other important geometric quantities could help the centerline matching, such as radius, circumference or surface curvature information. To resolve this issue, we have developed a dynamic programming algorithm for 1-dimensional point matching, and we incorporate 3-dimensional surface information into our metric, or objective function, to match each point of the centerline. Dynamic programming has the advantage of being a fast and efficient algorithm that finds a globally optimal matching of centerlines, with respect to the objective function, while preserving centerpoint ordering [2].

In Section 3.2 we briefly describe a method for automatically extracting centerlines from CT scans. In Section 3.3, we describe our centerline registration method using dynamic programming. Section 4 presents our results, including the use of the matching technique for synchronized virtual colonoscopy. Section 5 describes intended future work.

3.2 Centerline Extraction

In this section, we describe the method by which we obtain centerlines of a surface model. Our starting point is a grayscale volume that is segmented and then used to produce a triangulated surface model of the colon, of the kind obtained through the use of the Marching Cubes algorithm [12], or other similar isosurface extraction algorithm.

Our method for centerline extraction is based on the following physical model. Let Σ denote the tubular colon surface with open ends described by two closed space curves σ_0 and σ_1 . We suppose that these boundary curves are held at a constant temperature of 0 and 1 degrees respectively, and seek the steady-state distribution of temperature u across the surface. The standard theory of partial differential equations [7] tells us that this temperature distribution will smoothly vary between 0 and 1 degrees from end to end, and will be free of local maxima and minima away from the boundary curves. In fact, the function u will be harmonic, i.e. will satisfy Laplace's equation $\Delta u = 0$, and each level set $u^{-1}(t), t \in [0, 1]$ will consist of a loop around the colon surface. Our centerline is then formed by the centers of mass of these loops.

The numerical method used to find the temperature distribution function is based on finite element techniques [5]. In [8], there is a closely related method for colon mapping. The function u may be found by solving the sparse linear system of equations using standard methods from numerical linear algebra. We have found that the solution of this system can be found in under 5 minutes on a single processor Sun Ultra 10, for a surface consisting of 100,000 triangles. Once the solution u is found, the center points may be found simply by dividing up the interval $[0, 1]$ into a number of sub-intervals, and calculating for each sub-interval the center of mass of the vertices with corresponding values of u .

Since each centerpoint is associated with a loop around the colon surface, surface measures such as average radial distance, circumference and curvatures can be mapped to the centerpoint for use in our dynamic programming matching technique.

3.3 Dynamic Programming

Overview. Once we have the centerpoints extracted for both colons, we match them using dynamic programming. Dynamic programming solves optimization problems by finding and recursively combining the optimal solutions to sub-problems. A dynamic programming algorithm is efficient since it solves every subproblem only once and caches the solution, thereby avoiding the work of recomputing the answer every time the subproblem is encountered [2].

Dynamic programming has been used in a variety of contexts including for DNA and protein sequence alignment [10] as well as special cases of pose estimation in object recognition [4],[6].

Registration Algorithm. In our case, we wish to align two sets of centerpoints, P_1^N *i.e.* the centerline of the prone colon containing the points indexed from 1 to N and S_1^M , the centerline of the supine colon indexed from 1 to M . The subproblems of this optimization task are all the pairwise matching of the subsequences of P_1^i , ($1 \leq i \leq N$) and S_1^j , ($1 \leq j \leq M$). We now describe the two steps of our centerline registration algorithm.

1. Recursive Definition of an Optimal Solution

Let $f(P_i, S_j)$ be a cost associated with matching centerpoint P_i with S_j .

Let us further assume that we already have an optimal minimal cost solution for the matching of the pairs of subsequences (P_1^i, S_1^{j-1}) , (P_1^{i-1}, S_1^{j-1}) , and (P_1^{i-1}, S_1^j) . If we define F to be a metric that evaluates the matching of the argument subsequences, we can find the optimal alignment of the centerlines (P_1^i, S_1^j) by solving the recursion:

$$F(P_1^i, S_1^j) = f(P_i, S_j) + \min \begin{cases} F(P_1^i, S_1^{j-1}); & \text{(expansion at } P_i) \\ F(P_1^{i-1}, S_1^{j-1}); & \text{(no expansion/compression)} \\ F(P_1^{i-1}, S_1^j); & \text{(expansion at } S_j) \end{cases} \tag{1}$$

With this recursive expression, we fill in an $N \times M$ matrix with entries at $(i, j) : F(P_1^i, S_1^j)$, along with pointers in the direction of the preceding sequence matching which led to this optimal value. It is important to note that dynamic programming allows many-to-many mappings, resulting in mappings that can be locally compressions or expansions. For example, if $F(P_1^i, S_1^{j-1})$ is chosen as the optimal subsequence of $F(P_1^i, S_1^j)$, then centerpoint P_i will match to both points S_{j-1} and S_j , which would mean that in the locality of point P_i , the matching to the sequence S_1^j is a compression. We chose to penalize the amount of stretching and compression allowed with a penalty function $g()$ added to the first and third line of equation 1. We experimented with different values of $g() = 0.0$ and $g() = 0.1$ and experimentally found that the latter gives us a better match.

2. Extracting the Sequence Alignment

By following the pointers from entry (N, M) to entry $(1, 1)$, we obtain a sequence of (i, j) pairs that define a point-to-point correspondence between point P_i and point S_j .

Objective Function $f(P_i, S_j)$. As mentioned in Section 2, we use both centerline and geometrical information to give a value to each centerpoint.

- The centerline information is the distance from the first centerpoint to the current centerpoint normalized to the total length of the centerline.
- The geometrical information is the average radial distance from the centerpoint to the surface loop centered at the centerpoint.

The objective function $f(P_i, S_j)$ evaluates how closely two centerpoints P_i and S_j match. We have defined it as:

$$f(P_i, S_j) = \alpha(r_i^P - r_j^S)^2 + (1 - \alpha)(d_i^P - d_j^S)^2 \quad (2)$$

where r_i^P is the average radial distance at the i th centerpoint in the prone scan, d_i^P is the distance along the path of the centerpoint, and similarly for r_j^S and d_j^S . The parameter α is used to balance the two terms of the functional. The results for the colon registration presented in Section 4 are for $\alpha = 0.5$. Other functionals incorporating other surface information can also be used.

4 Results

Here we show the results of our algorithm applied to supine and prone CT scans taken of the same patient 8 minutes apart. Both scans had 512 matrix size with slice thickness of 1mm, and 362 slices. Pixel size was 0.6mm.

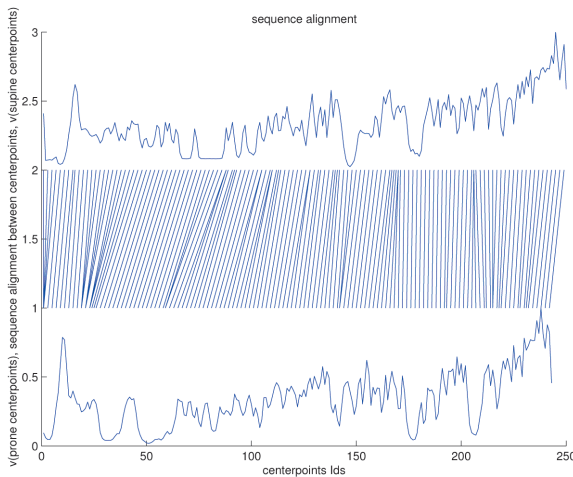


Fig. 2. Sequence alignment of prone and supine centerpoints.

Figure 2 shows an objective value with $\alpha = 0.5$, plotted for both colons as well as the sequence alignments found by our algorithm between points using a penalty for excessive stretching or compression (we used $g() = 0.1$). Each line in the middle of the figure shows a correspondence between a prone and supine point. As can be seen, there are areas of slight expansion and stretching that are detected. In order to have a preliminary evaluation of the fly-throughs produced by our algorithm, we recorded how many frames matched for different values of α out of the total number of frames (278). These results are presented in Table 1.

From the results, we see that using the distance metric alone ($\alpha = 0$) fails because the initial 6 centerpoints of the supine centerline do not exist on the prone

Table 1. Performance results with different objective functions.

α	% of matched frames
0	40
0.5	94
1	88

centerline. This causes a misalignment of frames throughout the fly-through. Using average radius information alone ($\alpha = 1$) is considerably better, except at particular centerpoints where there is a collapse of the colon due to fluid leftovers. This causes a temporary misalignment until the next correct radius information is matched. We found that a combination of both metrics is optimal and gives us a 94 % frame match rate.

4.1 Synchronized Virtual Colonoscopy (SVC)

In order to visualize synchronized colonoscopies, we use a virtual endoscopy tool that we developed [3]. This tool displays the two surface models and the location of both virtual endoscopes relative to the surface models, as well as the views of each virtual endoscope updated simultaneously to show the same location in the colon (see Figure 3). In addition, the user has the option to display a reformatted CT slice through each volume that moves along with the virtual endoscope and stays parallel to the view plane (Figure 3, right view). This functionality allows the user to compare the grayscale image registered to match the colon data. During the fly-throughs, the positions of the virtual endoscopes

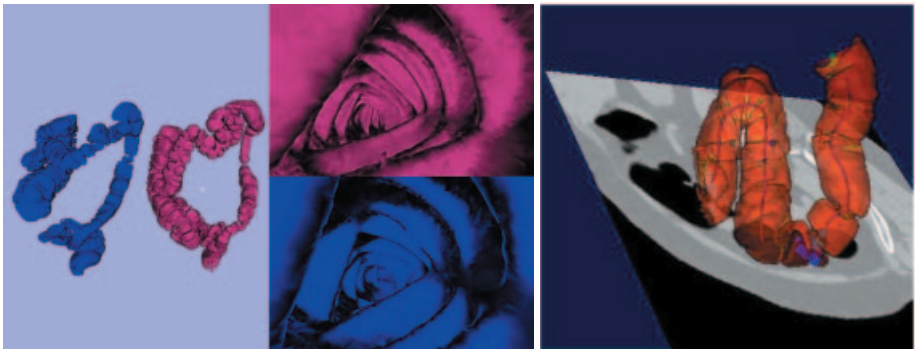


Fig. 3. Left. Virtual colonoscopy tool screenshot with surface models and endoscopic views – Right. Reformatted slice with colon model and centerpath.

are defined by the centerpoint sequence alignment. However, the rotation of the virtual endoscopes (the “ViewUp” vector) is not defined and so both views could be rotated relative to each other. In practice, if the ViewUp of both virtual endoscopes are manually matched on the first corresponding centerpoint, then

the rotation difference between the two views stays small. But in the future, we would like to find a corresponding ViewUp for both endoscopes by comparing slices perpendicular to the centerline at corresponding points and aligning them by moments.

5 Future Work

In the future, we will incorporate other surface information, such as curvature and moments, into our objective functional. We also plan to test our method in a clinical setting and obtain expert classification of matched/mismatched frames.

6 Conclusion

We presented a method for automatic supine and prone colon centerline registration based on the dynamic programming principle. Our method can include information such as path length and curvature as well as descriptors of the shape and size of the colon near the path to find an optimal matching. We showed that our colon registration technique can be used to produce synchronized fly-throughs.

References

1. B. Acar, S. Napel, D.S. Paik, P. Li, J. Yee, C.F. Bealieu, R.B. Jeffrey. Registration of supine and prone ct colonography data: Method and evaluation. Radiological Society of North America 87th Scientific Sessions, 2001.
2. T. H. Cormen, C. E. Leisterson, and R. L. Rivest. *Introduction to Algorithms*. McGraw-Hill, New York, 1998.
3. D. Nain, S. Haker, R. Kikinis, W. Grimson. An interactive virtual endoscopy tool. *Satellite Workshop at the Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'2001)*, 2001.
4. E. Cosman, W. Wells. Slice-wise, non-rigid volumetric image registration by dynamic programming. <http://www.ai.mit.edu/people/ercosman>, 2001.
5. T. Hughes. The finite element method. Prentice-Hall, New Jersey, 1987.
6. A. L. Ratan. Learning visual concepts for image classification. *Ph.D. Thesis, A.I. Lab, MIT*, 1999.
7. J. Rauch. Partial differential equations. Springer-Verlag, New York, 1991.
8. S. Haker, S. Angenent, A. Tannenbaum, and R. Kikinis. Nondistorting flattening maps and the 3d visualization of colon ct images. *IEEE Trans. on Medical Imaging*, 19:pp. 665–670., 2000.
9. S.C. Chen, D.S. Lu, J.R. Hecht et al. Ct colonography: value of scanning in both the supine and prone positions. *Am. J. Rad.*, 172:pp. 595–599., 1999.
10. J. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology*. PWS Publishing Co., New York, 1997.
11. D. Vining. Virtual endoscopy: Is it a reality? *Radiology*, 200:pp. 30–31., 1996.
12. W. Schroeder, H. Martin, and B. Lorensen. *The visualization toolkit*. Prentice-Hall, New Jersey, 1996.