A Robust and Efficient Approach to Detect 3D Rectal Tubes from CT Colonography

Xiaoyun Yang[1, a)] and Greg Slabaugh[1]

Medicsight PLC, Kensington Centre, 66 Hammersmith Road, London,
UK

(Dated: 15 September 2011)

Abstract

Purpose: The rectal tube (RT) is a common source of false positives (FPs) in computer-aided detection (CAD) systems for CT colonography. A robust and efficient detection of RT can improve CAD performance by eliminating such "obvious" FPs and increase radiologists' confidence in CAD.

Methods: In this paper, we present a novel and robust bottom-up approach to detect the RT. Probabilistic models, trained using kernel density estimation on simple low-level features, are employed to rank and select the most likely RT tube candidate on each axial slice. Then, a shape model, robustly estimated using Random Sample Consensus (RANSAC), infers the global RT path from the selected local detections. Subimages around the RT path are projected into a subspace formed from training subimages of the RT. A quadratic discriminant analysis (QDA) provides a classification of a subimage as RT or non-RT based on the projection. Finally, a bottom-top clustering method is proposed to merge the classification predictions together to locate the tip position of the RT.

Results: Our method is validated using a diverse database, including data from five hospitals. On a testing data with 21 patients (42 volumes), 99.5% of annotated RT paths have been successfully detected. Evaluated with CAD, 98.4% of FPs caused by the RT have been detected and removed without any loss of sensitivity.

Conclusion: The proposed method demonstrates a high detection rate of the RT path, and when tested in a CAD system, reduces FPs caused by the RT without the loss of sensitivity.

Keywords: Rectal Tube, RANSAC, CAD, eigenspace, CT colonography

[a)] Electronic mail: Xiaoyun.Yang@medicsight.com.

I. INTRODUCTION

A. Motivation

Colorectal cancer is the second leading cause of cancer related death in western countries and the third most common human malignancy in the United States[1]. Most colorectal cancers arise from pre-malignant adenomatous polyps in the colon that develop into cancer over time. The progression from polyp to cancerous lesion takes more than ten years for most patients. Because of this slow growth rate, colorectal screening[2] is an effective method for polyp detection, and subsequent removal reduces the risk of colorectal cancer by up to 90 percent[3]. Colonoscopy, the endoscopic examination of the colon using a flexible video camera, is the standard approach to colorectal screening; however, in recent years there has been much interest in CT Colonography (CTC), which uses CT images of the cleansed and insufflated colon[4]. Polyps can then be detected by a clinical reader who examines the CT data using advanced visualization software that provides both 3D endoluminal views of the colon as well as 2D multi-planar reformatted slices.

Although CTC has been demonstrated to be an effective colorectal screening approach[5], it is possible for the clinical reader to fail to detect lesions, due to the large quantity of data generated (typically 800 - 2000 images per patient). Computer-aided detection (CAD) software assists the clinical reader by automatically analyzing the CT data and highlighting potential lesions. CAD may detect lesions that would have otherwise been missed by the reader, and may give the reader more confidence that lesions were not missed during the read. It has been demonstrated that the use of CAD leads to improvements in reader performance (at all levels of experience) in detecting lesions in CTC data[6,7].

It has been reported that tagged stool, ileocecal valve, thick folds, extra-colonic structures and the RT or RT balloon are common sources of false positives (FPs)[6–8] in CTC CAD systems. The RT has a (potentially bent) cylindrical shape and includes a hemispherical tip that often has a polyp-mimicking appearance. As such, the rectal tube generates "obvious" CAD FPs[8–11] that can be easily dismissed by the clinical reader. However, radiologists may lose confidence in the CAD system when presented such "obvious" FPs. To improve the overall CAD performance and increase radiologists' confidence, it is therefore desirable to have a robust and efficient way to identify the RT and remove its resulting FPs from the

CAD marks presented to the reader.

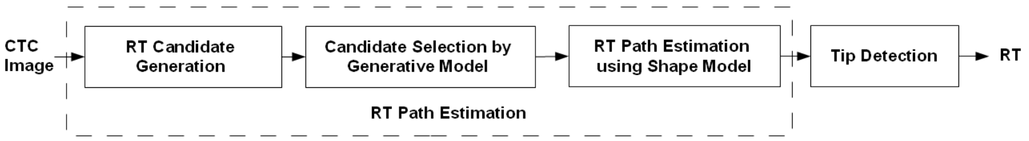

CTC Image
RT Candidate Generation
Candidate Selection by Generative Model
RT Path Estimation using Shape Model
Tip Detection
RT
RT Path Estimation

FIG. 1: Rectal Tube Detection Scheme.

Some research has been proposed[9–11] to address this problem. Iordanescu *et al.*[9] developed an image segmentation based method that detects, via template matching, the air inside the RT in the first nine CT slices, tracks the tube, and performs segmentation using morphological operations. The template is generated in a fixed definition of shape. It is a relatively rigid method for detection, and may fail to take account of the shape variability by tracking 2D slices. It assumes that the RT can be detected by a set of fixed-shape template matching continuously during tracking, which could be too strong of an assumption. Suzuki *et al.*[10] employed a Massive Training Artificial Neural Network (MTANN) to distinguish between polyps and FPs due to the RT. In the 73 patient data (146 volumes), MTANN detected all 20 RT FPs as well as 53 FPs due to other sources among the 224 FPs in total, without removal of any true positives. The method requires manual selection of representative training samples. Imperfect selection of training samples could degrade the performance[10]. This could introduce uncertainty to the performance. Barbu *et al.*[11] designed a method to detect the RT with the assumption that the RT can have large variable shape and appearance, which increased the complexity of their method. They first started detecting 3D circles using voting and Probabilistic Boosting Tree (PBT) methods, with a feature set consisting of 6720 features computed from gradient, curvatures, etc. A second classifier trained with PBT is used to detect short tubes from the detected 3D circles. Dynamic programming is employed to find the best RT segmentation from the detected parts.

B. Our contribution

In this paper, we present a novel method to detect the RT using a global shape model, unlike the methods discussed above. A quadratic 3D space curve models the global rectal tube path, and is robustly estimated using RANSAC[12] from local (and potentially outlier) 2D detections of the air in the rectal tube on axial slices. These 2D detections are made through a probabilistic model learned through kernel density estimation (KDE) on simple low-level features. With the rectal tube path robustly estimated, the RT tip is then estimated using a classifier trained on projection coefficients in an eigen-subspace of rectal tube subimages. The classifier provides a label for each axial slice if it contains the RT or not. From these outputs, a bottom-top clustering method merges the labels predicted by the classifier into a few major clusters, from which the position of the RT tip can be localized.

We note that this paper extends our preliminary work on RT detection presented in[13] by inferring the RT tip location using the eigen-subspace, classifier, and clustering method mentioned above. Accurate tip estimation improves the FP detections per volume and reduces the likelihood that the method would produce a false negative when used in a CAD system to reduce false positives.

Our method is computationally inexpensive and reliable. The experimental results demonstrate a high detection rate in a diverse dataset, and when evaluated by a CAD system, achieve a good reduction of FPs caused by the RT without any loss of sensitivity.

II. METHODS

A. Overview

We present a learning framework to combine probabilistic models for low level detection of the air in the rectal tube with a global shape model of the RT path. The overview of the system is presented in Figure 1.

We start with simple image processing, applied to each 2D axial slice, to detect air regions (RT candidates) within the body in the most caudad slices, starting at the anal verge and moving up the abdomen towards the lungs. For each RT candidate, three simple low level features are computed: the normalized spatial position x and y of the centroid and the size h of the region.

A probabilistic model using KDE is trained for prone and supine data respectively and then used to rank the RT candidates. In each slice, the most probable tube candidate is selected and the others are discarded. From these 2D detections, RANSAC fits a global 3D shape model representing the RT path. RANSAC is a robust statistical technique that can infer the real RT path even in the presence of strong outliers resulting from incorrect

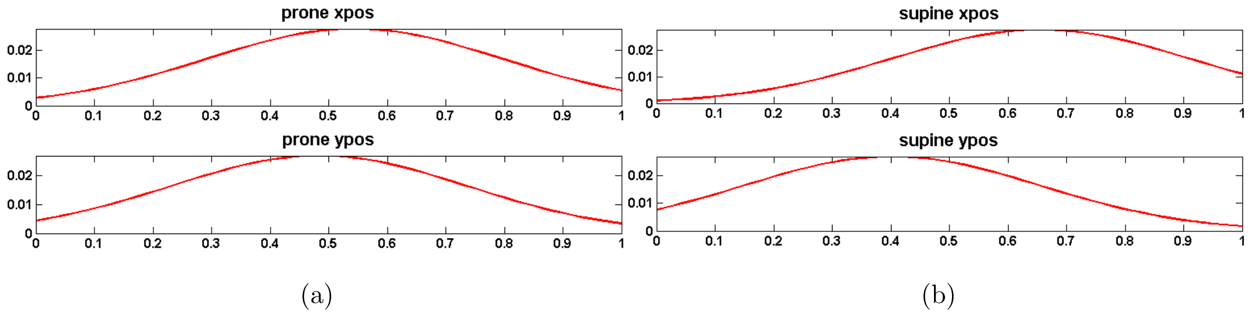


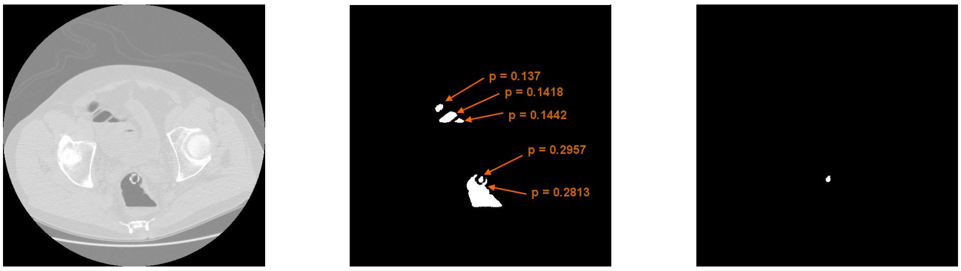
FIG. 2: The probability distributions of normalized candidate position x-direction, y-direction. (a) supine, (b) prone.

KDE predictions. We have two assumptions in this paper. First, the spatial distribution of the RT within the body can be approximately described by a probabilistic model built from training data. Intuitively, this assumes that the candidate is more likely selected if its position is close to the mean position of training samples. With an additional constraint that the RT must located within or near the colon mask, we can further remove the bad (false) candidates if a colon mask is provided. Second, the RT is a possibly bent cylindrical structure placed on the bottom of body. With this knowledge, by seeking a quadratic path supported by maximum number of candidates, we can differentiate good candidates and bad ones selected by the probabilistic model. The RT path can thus be estimated from good candidates.

Once the RT path is located, we search for the RT tip with a prior knowledge that the RT is less than 140mm in the body. Firstly, for each slice, centering around the path point at that slice, a set of 2D sub-images is extracted and projected into a eigen-subspace; and a quadratic discriminant classifier is applied on the projected subspace to make the RT/non-RT prediction; a bottom-up clustering approach is then proposed to apply on those predictions to merge them iteratively into a small number of clusters, the position of the RT tip can then be located. The details are discussed in the following subsections.

B. Probabilistic models for 2D candidate detection of RT regions

In this subsection we demonstrate how we generate local 2D RT candidates, and train a probabilistic model from simple low-level features. The generated model is then employed



(a) (b) (c)

FIG. 3: Examples of the candidate selected by KDE. (a) the CT image, (b) the candidate regions, and (c) the region selected by KDE. This region corresponds to the air in the RT.

for selection of the most probable candidate on a given slice.

The air region in the RT can be identified on an axial image slice using standard image processing techniques. For each slice, we apply simple thresholding using a threshold value ranged between $[-700, -850]$[9,14], e.g. -750 HU. All the air pixels then belong to the background and all the remaining pixels are assigned to the foreground pixels; the largest foreground region is morphologically filled and then regarded as the body region at that slice. The air regions located within this body region are then kept as inside-body air and the others are discarded as outside-body air. A 3D minimum bounding box *rect* to enclose all the 2D body regions can be uniquely determined with simple boolean min or max operations on the x-y-z coordinates of all the pixels in body regions, where the x-y-z coordinates of the bounding box is denoted to be $(rect_{x1}, rect_{x2}, rect_{y1}, rect_{y2}, rect_{z1}, rect_{z2})$.

Each candidate is represented by its centroid (C_x^i, C_y^i, C_z^i). Three low level features, (V_x^i, V_y^i, V_s^i), for each candidate are computed: the normalized spatial position (V_x^i) and (V_y^i) in the x and y direction, and the region size (V_s^i). The normalized spatial positions are determined as $V_x^i = (C_x^i - rect_{x1})/(rect_{x2} - rect_{x1}), V_y^i = (C_y^i - rect_{y1})/(rect_{y2} - rect_{y1})$.

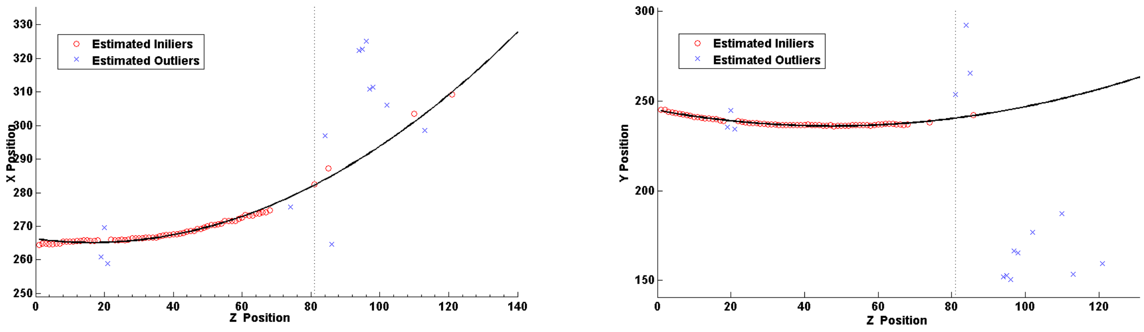
In the training stage, each candidate is sorted by the distance from its centroid (C_x^i, C_y^i, C_z^i) to the annotated RT center point in the same slice. The candidate with the smallest distance (the distance must be very small, e.g. less than 6 mm) can be considered as a good candidate to represent RT, and selected as training data to build a probabilistic model. If the candidate's size is too large (e.g. above 130 mm^2, the candidate may come from non-RT air regions or connected with them. If the size is too small (e.g. one or two pixels with area less than 0.5 mm^2), it may come from CT noise. Therefore we limit the set of admissible candidates to have area in the range $[0.5, 130]mm^2$, which we set empirically. A kernel density function (KDE) can be built by summing a set of kernels[15] with the selected feature data (V_x^i, V_y^i, V_s^i), as shown in Equation 1. From the constructed KDE, the probability of a given candidate V being RT can be estimated as

$$P(RT|V) = \begin{cases} \frac{1}{T}\sum_{i=1}^{N} e^{-\frac{(V_x - V_x^i)^2}{2h_x^2}} \sum_{i=1}^{n} e^{-\frac{(V_y - V_y^i)^2}{2h_y^2}} \\ \qquad \text{if } V_s \in [0.5, 130]\text{mm}^2 \\ 0 \qquad \text{otherwise} \end{cases} \tag{1}$$

where $P(RT|V)$ represents the probability of a given candidate being RT, (V_x, V_y, V_s) is the feature vector of the candidate V. (V_x^i, V_y^i, V_s^i) $(i = 1, .., N)$, as defined earlier, is the ith example used for training. If V_s is too big or too small, a penalty is given and the probability is zero, and T is a normalization factor so that the probability sums to 1. The KDE model is generated from prone and supine data separately using Gaussian kernels with bandwidth h_x, h_y in x and y direction with value 0.25 for both. In this function, each data point has a contribution to the probability controlled by a Gaussian function. Figure 2 gives the probability distributions of normalized spatial position of RT for prone and supine respectively. With a large bandwidth value, the distribution behaves as a single Gaussian, where the candidates located near the center of body are more likely to be selected than the candidates at the border of body, particularly in the x-direction. In the y-direction, the RT candidates at the supine position are located relative closer to the bottom than the ones at the prone position. These reflect the anatomical knowledge of the RT position in the CT scan.

In the testing stage, we apply the appropriate (prone or supine) probabilistic model to the candidate's feature vector. A probability can then be estimated for each RT candidate. The candidates are ranked by the estimated probability and only the one with the maximum probability in each slice will be selected. Figure 3 (a) shows an example CT image, (b) the candidate regions and (c) most likely candidate selected by the KDE model.

The length of the RT within the body may vary depending upon how deeply the RT is inserted. We use the most $140mm$ caudad of CT slices as our processing range which are enough to cover the RT in our experimental data. The majority of RTs are located in the most $90mm$ caudad CT slices. An example of RT detections (prone position) is shown by Figure 4, where the depth resolution is $1mm$ and the slice range is $[1, 140]$. In the figure,



(a)

(b)

FIG. 4: Examples of RANSAC fitting for two volumes data in the xz and yz planes. Red circles represent inliers, blue crosses represent outliers, the black dotted line is the fitted path using RANSAC, and the black dotted line indicates the end of the RT.

the red circles (inliers, usually corresponding to the RT) and blue crosses (outliers, typically not corresponding to the RT) illustrate the selected candidate centroids, plotted separately, i.e., in the xz and yz planes. The tube is present in the patient for the most caudad slices with lower z value (here, the tip of the RT is located at slice 81 indicated by black dot line). Ideally, the KDE would predict all the air regions of the RT; however, KDE can wrongly pick up the candidates from other structures. To reduce the number of outliers, we add a constraint using a colon segmentation generated by the algorithm described by Slabaugh[8]. At each slice, if there are candidate regions partially or fully covered by a morphological hole-filled colon segmentation (filling holes[16] in the colon segmentation if there is any), the KDE model only operates on those covered regions and chooses the one with maximum likelihood. If there are no regions covered by the colon segmentation, the KDE model operates on all candidates. The colon segmentation and KDE model together can reliably choose good candidates especially in the rectum, as shown by Figure 4a) (slice ranges $[1, 65]$). Most outliers appear near the RT tip or beyond (slice ranges $[65, 140]$ in Figure 4a). A few outliers appear in the rectum near the anal verge (around slice 20 in Figure 4a), where the RT air regions may connect with the colon air and the centroid can deviate away from the true locations.

C. RT path estimation using a global shape model

In this subsection we describe how the underlying RT path is inferred from the RT candidate centroids selected in the previous step.

The RT is a cylindrical tube placed in the patient's colon, and once the colon is insufflated with room air and there is no force to twist the RT. The path of the RT can be approximated as a quadratic curve (which includes a straight line as a special case). While there may occasionally be some other air-filled structure or noise giving a quadratic path, the RT is easily identified as the longest path along a smooth and continuous quadratic curve starting at the bottom of the patient. To estimate the correct RT path, one must differentiate the outliers from the inliers that represent the true locations of the RT. From the inliers, we can infer the other 2D RT locations missed by KDE. With the prior model of the global shape information that the RT path is a quadratic curve and continuously appears in the most caudad slices, we use that as a criteria to seek a maximum set of inlier points that can fit the quadratic curve which can be resolved by RANSAC[12], as shown in Equation 2:

$$\hat{\theta} = \arg\max_{\theta} \sum_{i=1}^{n} f(e_i^2|\theta) \tag{2}$$

where

$$f(e_i^2|\theta) = \begin{cases} 1 & e_i^2 < \delta^2 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where θ is the quadratic model to be estimated, e_i is the error or the distance between the data V_i and the estimated curve. δ is a threshold under the hypothesis that the error is generated by a true inlier contaminated with a Gaussian noise $P[e_i \leq \delta]$, where we expect the value of P is 0.95. The estimation of 3 parameters for a quadratic curve requires 3 points at least; and RANSAC adopts a random selection process to choose any 3 points to estimate a quadratic model θ, this process is repeated k times. And the one with the maximum number of inliers shown in Equation 2 is chosen. Let P_k be probability that the points selected to estimate the model are all inliers (n is total number of selected points). p is the probability that any selected point is inliers ($p = \frac{number of inliers in data}{number of points in data}$). Their relationship can be characterized as follows[12]:

$$1 - P_k = (1 - p^n)^k \tag{4}$$

the required number of trials is

$$k = \frac{log(1 - P_k)}{log(1 - p^n)} \tag{5}$$

From Equation 5, assuming that the percentage of inliers in the data is $p = 0.5$, and we know the required number of points for the estimation of quadratic model is 3 ($n = 3$), to attain a 0.99 probability of success ($P_k = 0.99$), it only requires only 35 (k=35) times to repeat this process, which can be computed quickly.

A $3D$ space curve quadratic in z models the path as $[C_x(z), C_y(z), z]^T = [\theta_{0x} + \theta_{1x}z + \theta_{2x}z^2, \theta_{0y} + \theta_{1y}z + \theta_{2y}z^2, z]^T$, where the estimation of $[\theta_{0x}, \theta_{1x}, \theta_{2x}]$ and $[\theta_{0y}, \theta_{1y}, \theta_{2y}]$ is performed separately in the xz and yz planes respectively using RANSAC.

As shown in Figure 4, the black line illustrates the inferred RT path from the inliers (red circles). Even in the presence of large outliers (non-RT locations), the estimated RT path estimated by RANSAC is quite reliable. After RANSAC fitting, given a slice number, we can predict the RT path location. RANSAC can be viewed as a method to achieve a robust regression to fit the global shape model to data containing outliers.

D. RT Tip Detection

In this subsection we describe how to efficiently detect the RT tip along the RT path given by the previous step.

1. Eigentubes: Eigenspace of the RT

Kirby and Sirovich[17] first proposed to represent an arbitrary face with a set of eigenvectors using principal component analysis. Turk and Pentland[18] further developed this idea to construct a fast matching algorithm to recognize the face with the coefficients of eigenfaces. Under a much smaller subspace constructed by the first few eigenvectors, the classification of new samples by computing its distance to the training samples of different classes can be efficiently solved. We propose to use such an eigen-picture method for the detection the RT region along the estimated path, forming a subspace of RT images we call *eigentubes*.

To add robustness to localisation errors along the path, for each location (C_x, C_y, C_z) on the estimated RT path, we shifted it by moving 5 pixels around in the x-direction and y-directions within the plane, and extracted a sub-image with a 2D window 21×21 centered

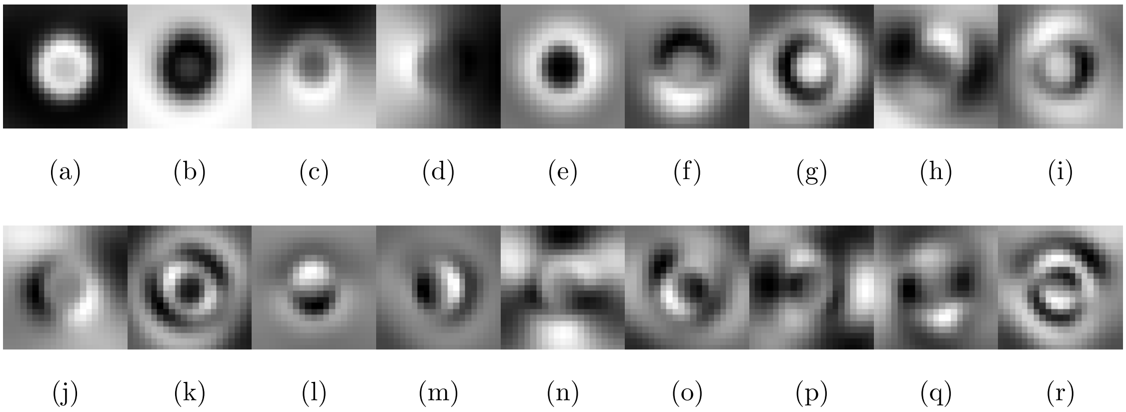


FIG. 5: Eigentubes: the images of the first 18 eigenvectors.

on each position $(C_x + \{-5, 0, 5\}, C_y + \{-5, 0, 5\}, C_z)$. Thus, we gathered in total of 9 sub-images for each given location (C_x, C_y, C_z). We then have a set of representative images with a $[\hat{\vec{x}}_1$, ..., $\hat{\vec{x}}_n]$, where the image $\hat{\vec{x}}_i$ is a $d \times 1$ dimension vector (d $= r \times c$, r $(r = 21)$) represents the number of rows and c $(c = 21)$ represents the number of columns of the $2D$ image) formed by appending each row of the image into the vector sequentially. Any image can be represented in such a form:

$$\vec{x_i} = U_q \vec{\lambda_i} \tag{6}$$

and

$$\vec{x_i} = \hat{\vec{x_i}} - \vec{\mu} \tag{7}$$

where $\vec{\mu}$ (dimension $d \times 1$) is a mean image of the set images, $\vec{x_i}$ is the observation of data $\hat{\vec{x_i}}$ centered at $\vec{\mu}$. We define a matrix $X = (\vec{x_1} ... \vec{x_n})$, where $\vec{x_i}$ is the ith column of the matrix. To best represent the n images of this set X, each of the image is $d \times 1$ dimension with a much smaller dimension q, the problem of seeking such subspace is formulated as minimizing the reconstruction error:

$$U_q = \arg\min_{U_q} \sum_{i=1}^{n} \|\vec{x_i} - U_q \vec{\lambda_i}\| \tag{8}$$

where U_q is a $d \times q$ $(q \ll n)$ matrix and it defines a subspace spanned by its column, each column vector of U_q is a basis vector of this space which is orthogonal to each other. $\vec{\lambda}$

$(q \times 1)$ is a vector of coefficients, and this can be obtained by projecting $\vec{x_i}$ onto the subspace defined by U_q. Since the column of U_q is orthogonal, the coefficients can be obtained by using product of each basis vector with $\vec{x_i}$, $\vec{\lambda} = U_q^T \vec{x_i}$, Equation 8 can then be represented as:

$$U_q = \arg\min_{U_q} \sum_{i=1}^{n} \|\vec{x_i} - U_q U_q^T \vec{x_i}\| \tag{9}$$

where $U_q U_q^T$ acts as a projection matrix and maps each image $\vec{x_i}$ into its rank q. This can be solved by a singular value decomposition of X:

$$X_q = U_q \Sigma V^T$$

where X_q is the approximate representation of X using the first q eigenvectors; U_q is $d \times q$ matrix, the column of U_q is a eigenvector of XX^T, Σ is a diagonal matrix, the value is its square of eigenvalue in a decreasing order to the covariance matrix XX^T. The first eigenvector of covariance XX^T can be intuitively explained as the direction that the data set has maximal variance.

Figure 5 illustrates the first 18 eigenvectors, each of which is a column of U_q (if $q = 18$). The appearance of the eigenvectors includes more ring-like shapes due to the round shape of RT at each section. The first eigenvector (a) is a bright disk surrounded by dark areas, which is similar to the top (near the tip) part of the RT. The second eigenvector (b) is dark inside and bright outside, which is similar to the bottom (near the anal verge) part of the RT. The fifth eigenvector (e) is dark inside, bright in the middle and gray outside, which is similar to some middle part of the RT. In the third eigenvector (c), there is half dark outside the RT, which behaves like a colon air on the top of the RT. Figure 6 illustrates the distribution of each candidate ROI image by projecting them into the subspace spanned by the first three eigenvectors. The red points represent the samples from a RT image, and the blue points represent the samples from a non-RT image. It shows a certain degree of separation between the two classes even in a three-dimensional subspace.

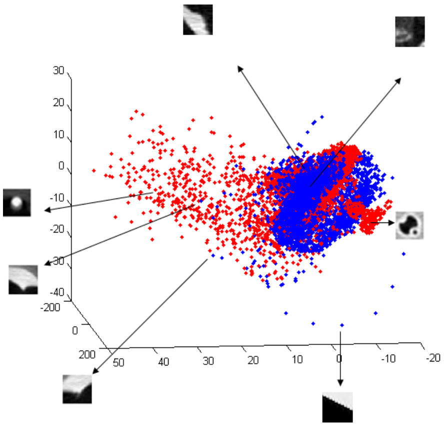


FIG. 6: PCA (3D) mapping of $2D$ candidate image for RT, red dots represent RT candidate, blue dots represent non-RT candidates.

2. Classification and Clustering

In the constructed eigenspace, each sub-image is represented as a data point. Quadratic discriminant analysis (QDA) is applied on the eigen-subspace for the detection/classification of the RT. Figure 7 illustrates a 5-fold validation result by projecting into a different number of subspace dimensions. We use the first 17 eigenvectors to construct the subspace since there is no clear additional gain in the performance with more dimensions.

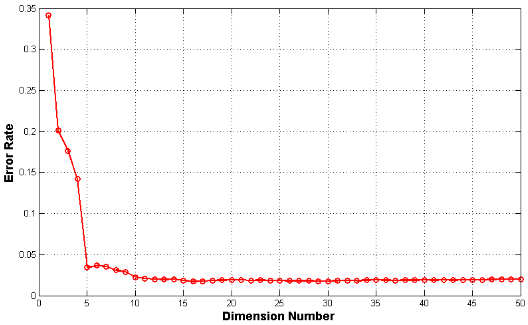


FIG. 7: RT classification, 5 fold cross validation.

For each slice z, the predictions above a certain confidence threshold (0.85) are labeled as positive (1) representing RT, otherwise labeled as negative (0) representing non-RT. We first group neighbors with the same label value together to form a set of clusters. The clusters are sequentially arranged in order of z values. Starting from the cluster with the smallest size, its two neighbor clusters which both have larger or equal size must have the

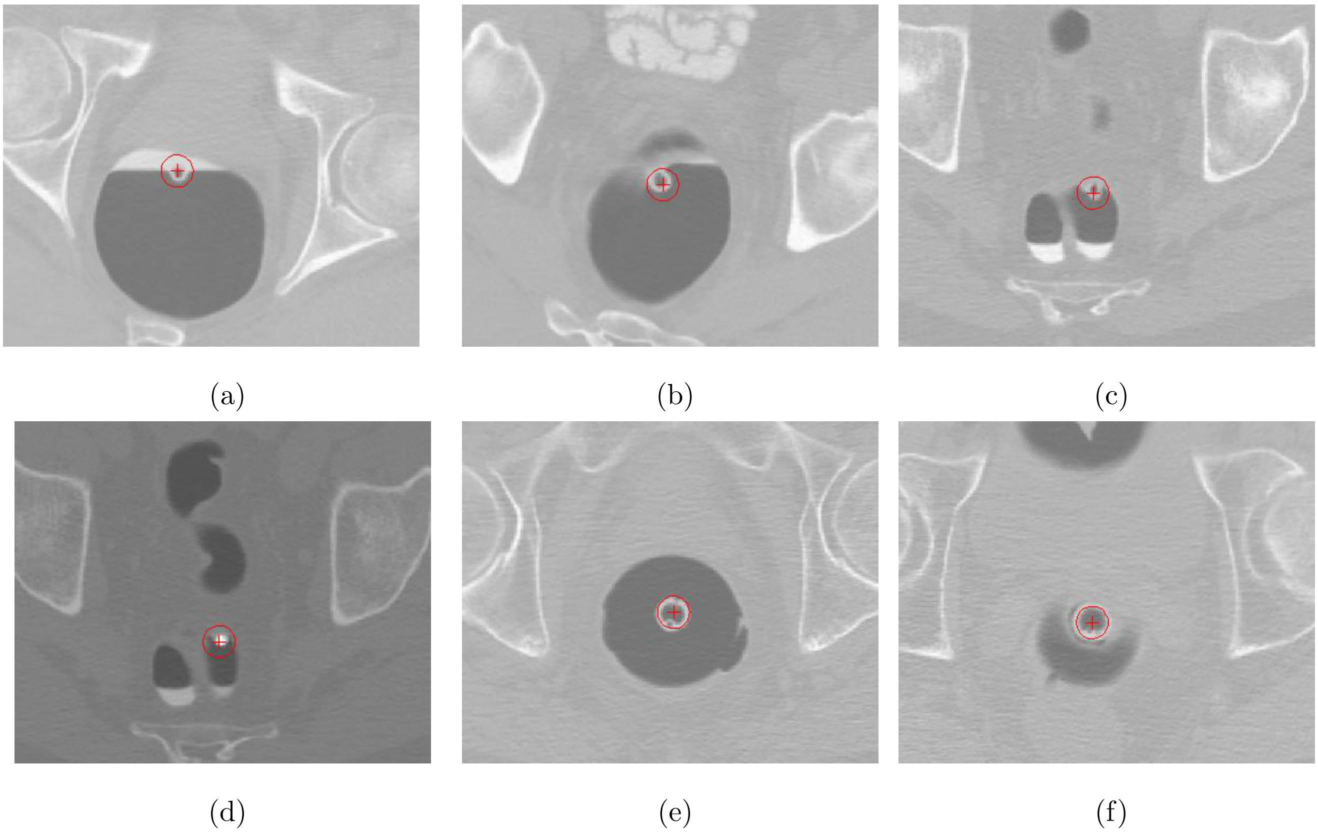


FIG. 8: Examples of detected RTs scanned from three hospitals, (a-b), (c-d), (e-f) are from the same hospital respectively.

same label value differing from the label in the middle (the cluster with the smallest size). We merge the three clusters into a new cluster: the label of the newly merged cluster is then the label of its neighbor clusters, and the size is the sum of the three clusters. The three clusters should be replaced by this newly merged cluster, and this merging process may be iteratively continued until there are two clusters left. As shown in Table I, 10 clusters (at the first row) are formed initially. We can see the first cluster at the second row is grouped from the first three clusters from the first row. Ideally, we could continue this process until we have only two clusters, one being the RT cluster and the other being non-RT. However, since we consider there could be a middle part of the RT which can be misclassified as non-RT with some length, the merging process is stopped when only four clusters are left or the minimum cluster size exceeding a threshold (25). The pseudocode of this clustering

TABLE I: A table demonstrating grouping process.

Cluster No	1	2	3	4	5	6	7	8	9	10
label	1	**0**	1	0	1	0	1	0	1	0
size	29	**1**	2	3	1	20	12	1	12	59
Cluster No	1	2	3	4	5	6	7	8		
label	1	0	**1**	0	1	**0**	1	0		
size	32	3	**1**	20	12	**1**	12	59		
...										
Cluster No	1	2	3	4						
label	1	0	1	0						
size	32	24	25	59						

algorithm is shown in Figure 9.

```
RT Clustering
begin do if $number(cluster) > 4$ then exit fi;
            $i := FindSmallestCluster(cluster)$;
                          find the cluster with the smallest size
            if $Size(cluster(i)) > 25$ then exit fi
            merge($cluster, i-1, i, i+1$);
   od                          merge cluster i and its neighbors
where
proc merge($node, i-1, i, i+1$) $\equiv$
   $node(i).label = node(i-1).label$;
   $node(i).size = node(i-1).size+$
   $node(i).size+$
   $node(i+1).size$;
   delete($node(i-1), node(i+1)$).
end
```

FIG. 9: Pseudocode of the RT clustering.

Once the grouping finished, we choose the last positive cluster with label size greater than 10% of the total length. We also choose the first negative cluster with label size greater than 10% of the total length. If the last z value of the chosen positive cluster is greater than the first z value of the chosen negative cluster, the z value of the positive cluster is determined as the top of the RT, otherwise, the first z value of the negative cluster. The estimated RT path below this chosen z value is the final RT path, and the rest is discarded.

III. RESULTS

The data are collected from five institutions. CT images were generated using scanners from all the major manufacturers, including Siemens, GE, Philips, and Toshiba, with 4, 8, 16, 32, and 64 multi-slice configurations, KVp ranged from $120 - 140$, and exposure ranged from 29 - 500 mAs. All subjects were scanned within the years between 1999 and 2008, and roughly 80% were administered fluid and fecal tagging. The RTs from those data were manufactured by Bracco and VMPAP, with or without hemispherical tip. The diameter ranged within $6 - 11$mm, the maximum intensity of the RT wall can be $550HU$ approximately, and the maximum intensity at tip position can be approximately 3072.

In the first experiment, we annotated the tube location (approximate center point) for 86 volumes of prone and supine data from 43 patients from five different hospitals. The data is divided 50% (44 volumes of 22 patient) for training and 50% (42 volumes of 21 patient) for testing by patient. Using KDE, separate probabilistic models for the prone and supine data were trained using the training data. During testing, we used the probabilistic models as a ranking method for the selection of the most probable candidate from each slice. If the in-plane distance between the prediction of KDE model and the corresponding annotated RT position is less than 5mm, that annotated RT position is viewed as being correctly detected by KDE, otherwise, the prediction given by KDE is viewed as a false location. In this experiment, the KDE method alone detected 83.7% of annotated RT centers but with 72.3 false locations (the location belonging to the non-RT mistakenly predicted as belonging to the RT by the algorithm) per volume in average. The center points of the selected candidates were then used by RANSAC to estimate the RT path using the global shape model, followed by the tip detection. With the estimation of the RT path and tip, the proposed approach detected 99.5% of annotated RT positions with only 1.21 false locations per volume in average. Figure 8 shows diverse examples of RT detection for different positions and patients using our approach. The red cross is the predicted center point. Figure 10 further illustrates a 3D view of the estimated RT path in the colon. Each row of Figure 10 shows a patient respectively, the first two columns (prone position) and the last two columns (supine position). The first and third column illustrates the estimated RT path (red line), rendered directly through a segmented colon. The second and fourth column illustrates the estimated RT path (red line) and circular path along the axial plane (white circle), rendered directly through the gray-scale image.

Our second experiment evaluates the RT detection for suppression of FPs in CAD. Any detection by CAD is removed as a FP if the distance between the center of the detected region and the center of the detected RT is less than $10mm$ in-plane and $7mm$ out-of-plane. On the 42 volume testing datasets, we could reduce 119 FPs generated by CAD (RT filtering is turned off)[8], and the average FPs per volume is reduced to 5.3 from 8.3, which is 35.6% of improvement in FP reduction, without any loss of sensitivity. We repeat this RT filtering but by replacing the estimated RT position with the annotated RT position, where 121 FPs can be removed in total, including the 119 FPs identified by our approach and with only 2 additional missed FPs. This shows that our approach has detected 98.4% of FPs caused by the RT and didn't remove any CAD mark from other sources, particularly true positives. In our data, we have found no true positives touching the RT and therefore no sensitivity loss due to this reason. The approach proposed in [11] was tested on a 210 datasets, achieving 94.7% detection rate and removing additional 26 CAD marks caused by other sources. Although the data set is different and our testing dataset is smaller, our performance has shown a very competitive performance in removing RT FPs and keeping the CAD sensitivity. We attribute the improved results of our method to the global shape model, which provides robustness to potentially erroneous local features.

We further test our approach on a large database of 382 volumes (without annotated RT). The average FP is reduced to 4.17 from 5.64 per volume. Overall, this improved the CAD with a 20.2% FP reduction without missing any polyp, therefore CAD sensitivity is unaffected. The FPs arising from RT are typically polyp-mimicking, and can pass the classifier designed based on the shape and contextual features, unlike FPs from other sources which are more easily rejected by the classifier. The RT filter presented in this paper is independent of our CAD system and can be used by other CAD system. The algorithm runs in 3.6 seconds per volume on a standard quad-core 2.4 GHz processor while requiring at most 150 MB of RAM regardless of the number of images in the series, which compares favorably to [11] which required 5.3 seconds running time (PC configuration is unknown).

IV. DISCUSSION

The method proposed in [9] determined a path by the first nine air hole regions. Tracking a long path, the RT can be detected using template matching with a set of fixed shape templates. The method is relatively rigid for detection, and may fail to take account of the shape variability when tracking 2D slices. The assumption that the RT can be continuously detected by a set of fixed-shape template matching during tracking could be very strong and may lead to incorrect or partly detection only. Indeed, in our work we have noticed that local detections can be erroneous on individual slices, thereby suggesting a more global approach achieved through the shape model in our approach. The method proposed by Barbu *et al.*[11] made a very weak assumption that the RT path can be any curved shape, which we believe underconstrains the problem and introduces additional complexities into the algorithm, such as two level PBT training. In[11] a large feature set is required for the task of classification, e.g. approximate 6720 features for circular region detection in PBT. In this paper, we assume the path of RT to be a straight line or quadratic curve given that the knowledge that the RT exhibits a slight bending or straight-like shape inserted into the rectum of the body, matching observation in practice. Our experiments demonstrate the effectiveness of this assumption by giving a high detection rate of the RT in our data collected from different sources. This simplifies the complexity of our algorithms and requires only 3 simple low level features to be computed: 2D normalized positions (x,y) and the size of air regions.

We have tested this approach on the image without RT. When the input image does not have RT, the estimated path is not on the RT path. Searching along the path, most extracted sub-images don't have the appearance of RT tip. After tip classification and clustering stage, the tip detector is going to end at the staring slice 0. This correctly reflects the fact that there is no RT in the image.

The RT tip appearance can be varied and is difficult to detect in isolation. Therefore, our method classifies each slice as having the RT tip, and merges the classification labels to produce the final tip location. This helps identify the tip location using long sections (possibly the entire) tube. To fulfill this task, we could collect each possible sectional appearance of RT and perform template matching for each of them. This may increase the cost of computation. In this paper, we propose to detect the RT tip along the estimated path with an eigentube method. A QDA classifier is applied to the transformed 17-eigenvectors constructed subspace. Each 2D sectional image of RT can be projected into a data point in this subspace. In the subspace, the similar appearance object is much closer to each other than the very dissimilar appearance ones. This allows some degree of variance in the appearance among the samples from RT candidates. The classification only requires the computation of the data points distribution in this subspace, which is quite cost-effective. We design a bottom-up clustering method to have a high-level view of the individual local detections on each section, with which an optimal decision to determine the place of the RT tip can be made.

Our method is designed generally to be robust from outlier data's influence. However, the accuracy of tip detector relies on the training samples to be representative. The RT path should generally pass through axial section perpendicularly or with a large angle with the axial plane. Some extreme cases that the angle between RT path and axial section could be small, the image of the intersection and axial plane may look very different from training sample images. In these cases, the eigentube detector may fail to accurately detect the tip. However, in practice, we have not observed cases where the detection failed due to this reason.

The balloon inflated at the top of the RT can be another potential source of FPs. Due to partial volume effect, there is not a clear boundary of the balloon, which makes its detection difficult. Our future work will try to estimate the balloon position using the estimated RT path and reduce the balloon's resulting FPs.

V. CONCLUSIONS

In this paper, we present a novel and robust learning approach for RT detection and removal of its resulting FPs in CAD. The approach starts from simple image processing operations and simple low-level 2D feature extraction for locally detected objects, which are then probabilistically ranked using KDE. Then, RANSAC robustly estimates the RT path from the most likely 2D candidates by fitting a 3D global shape model. The tip of the RT is estimated quickly, computed in an eigenspace of low dimension. A bottom-up clustering method is designed to put each individual local detections on each section together and make an optimal decision of the tip location in a high-level view. Our RT detection method has shown a state-of-the-art performance in detecting the RT path and removing FPs resulting from the RT in our CAD system.

REFERENCES

[1] "Cancer facts and figures," American Cancer Society Annual Report **12**, 11–12 (2007).

[2] M. E. Zalis, M. A. Barish, J. R. Choi, A. H. Dachmann, H. M. Fenlon, J. T. Ferrucci, S. N. Glick, A. Laghi, M. Macari, E. G. McFarland, M. M. Morrin, P. J. Pickhardt, J. Soto, and J. Yee, "CT colonography report and data system: A consensus proposal," Radiology **236**, 3–9 (2005).

[3] S. Winawer, "The achievements, impact, and future of the national polyp study," Gastrointest. Endosc. **64**, 975–978 (2006).

[4] C. D. Johnson and A. H. Dachman, "CT Colonography: The next colon screening examination?" Radiology **216**, 311–319 (2000).

[5] C. D. Johnson, M. Chen, A. Toledano, J. Heiken, A. H. Dachman, M. D. Kuo, C. Menias, B. Stewert, J. I. Cheema, R. G. Obregon, J. L. Fidler, P. Zimmerman, K. M. Horton, K. Coakley, R. B. Iyer, A. K. Hara, R. A. Halvorsen, G. Casola, J. Yee, B. A. Herman, L. J. Burgart, and P. J. Limburg, "Accuracy of CT colonography for detection of large adenomas and cancers," N. Engl. J. Med. **259**, 1207–1217 (2008).

[6] D. Burling, A. Moore, M. Marshall, J. Weldon, C. Gillen, R. Baldwin, K. Smith, P. Pickhardt, L. Honeyfield, and S. Taylor, "Virtual colonoscopy: effect of computer-assisted detection (CAD) on radiographer performance," Clin. Radiol. **63**, 549–556 (2008).

[7] E. M. Lawrence, P. J. Pickhardt, D. H. Kim, and J. B. Robbins, "Colorectal polyps: Stand-alone performance of computer-aided detection in a large asymptomatic screening population1," Radiology **256**, 791–798 (2010).

[8] G. Slabaugh, X. Yang, X. Ye, R. Boyes, and G. Beddoe, "A robust and fast system for CTC computer-aided detection of colorectal lesions," Algorithms **3**, 21–43 (2010).

[9] G. Iordanescu and R. M. Summers, "Reduction of false positives on the rectal tube in computer-aided detection for CT colonography," Med. Phys. **31**, 2855–2862 (2004).

[10] .

[11] A. Barbu, L. Bogoni, and D. Comaniciu, "Hierarchical part-based detection of 3D flexible tubes: Application to CT Colonoscopy," in *MICCAI* (2006) pp. 462–470.

[12]M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” Commun. ACM **24**, 381–395 (1981).

[13]X. Yang, G. Beddoe, and G. Slabaugh, “Learning to detect 3D rectal tubes in CT colonography using a global shape model,” in *MICCAI Virtual Colonoscopy Workshop* (2010) pp. 61–66.

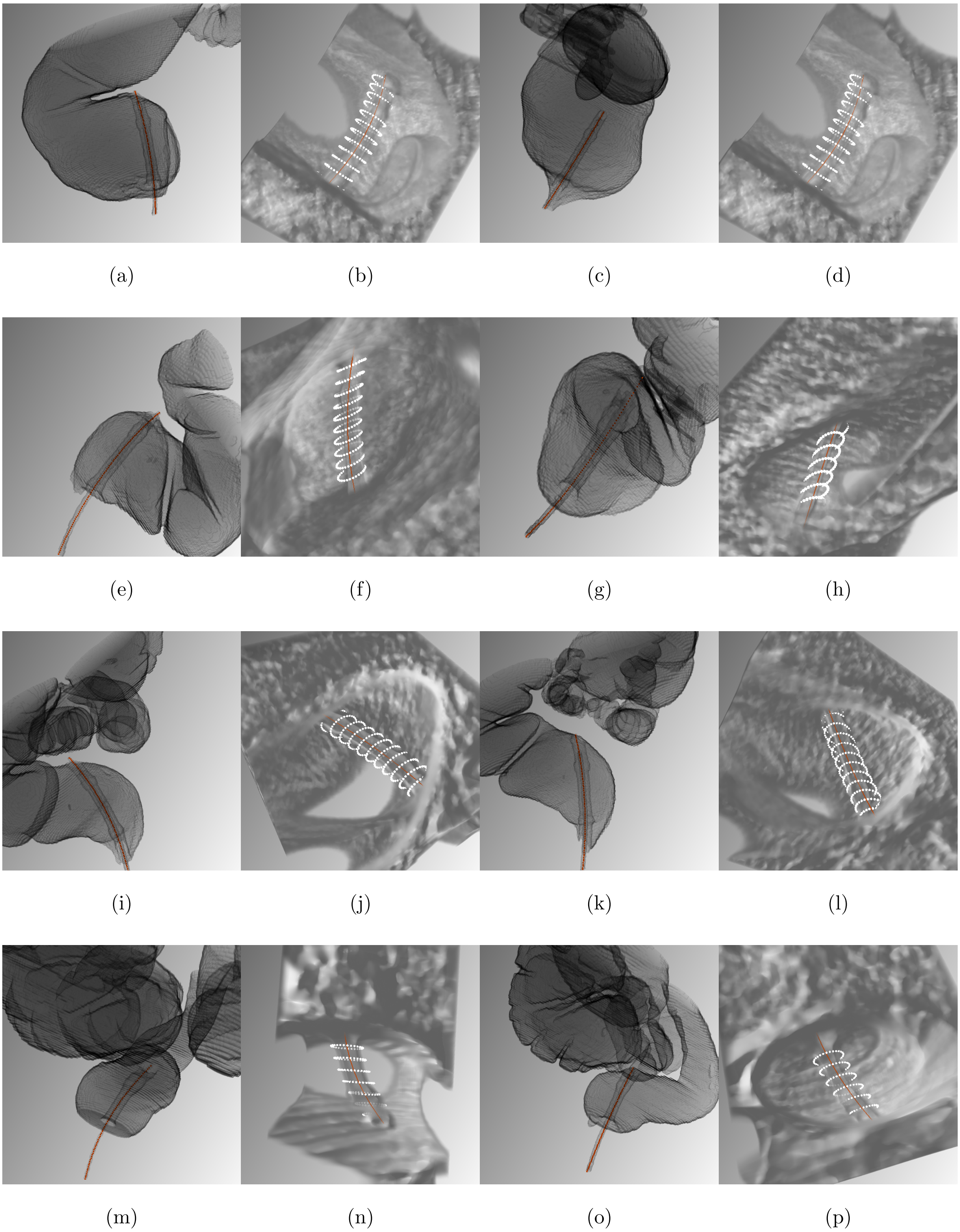
[14]P. Lia, S. Napel, B. Acar, D. S. Paik, R. B. Jeffrey, and C. F. Beaulieu, “Registration of central paths and colonic polyps between supine and prone scans in computed tomography colonography: pilot study,” Med. Phys. **31**, 2912 (2004).

[15]A. Bowman and A. Azzalini, *Applied smoothing techniques for data analysis: the kernel approach with S-Plus illustrations*, Vol. 18 (Oxford University Press, USA, 1997).

[16]P. Soille, *Morphological Image Analysis: Principles and Applications*, 2nd ed. (Springer-Verlag New York, 2003).

[17]M. Kirby and L. Sirovich, “Application of the karhunen-loeve procedure for the characterization of human faces,” IEEE Trans. Pattern Anal. Mach. Intell. **12**, 103–108 (1990).

[18]M. Turk and A. Pentland, “Eigenfaces for recognition,” J. Cognitive Neurosci. **3**, 71–86 (1991).



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

(e) (f) (g) (h)

(i) (j) (k) (l)

(m) (n) (o) (p)

FIG. 10: The visualization of RT detection. Each row shows a patient respectively, the first two (prone position) and the last two (supine position). The first and third column illustrates the estimated RT path (red line), rendered directly through a segmented colon. The second and fourth column illustrates the estimated RT path (red line) and circular path along axial slice (white circle), rendered directly through the gray-scale image.