

2007

# AVIDENSE: Advanced Video Analysis System for Colonoscopy Semantics

Yu Cao

*Iowa State University*

Follow this and additional works at: <https://lib.dr.iastate.edu/rtd>



Part of the [Computer Sciences Commons](#)

---

## Recommended Citation

Cao, Yu, "AVIDENSE: Advanced Video Analysis System for Colonoscopy Semantics" (2007). *Retrospective Theses and Dissertations*. 15757.

<https://lib.dr.iastate.edu/rtd/15757>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact [digirep@iastate.edu](mailto:digirep@iastate.edu).

**AVIDENSE: Advanced Video Analysis System for Colonoscopy Semantics**

by

Yu Cao

A dissertation submitted to the graduate faculty  
in partial fulfillment of the requirements for the degree of  
**DOCTOR OF PHILOSOPHY**

Major: Computer Science

Program of Study Committee:  
Wallapak Tavanapong, Major Professor  
Jennifer Davidson  
Vasant Honavar  
Dimitris Margaritis  
Johnny Wong

Iowa State University

Ames, Iowa

2007

Copyright © Yu Cao, 2007. All rights reserved.

UMI Number: 3337348

### INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.



---

UMI Microform 3337348  
Copyright 2009 by ProQuest LLC  
All rights reserved. This microform edition is protected against  
unauthorized copying under Title 17, United States Code.

---

ProQuest LLC  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106-1346

## TABLE OF CONTENTS

<b>LIST OF TABLES</b> . . . . .	v
<b>LIST OF FIGURES</b> . . . . .	vi
<b>ACKNOWLEDGEMENTS</b> . . . . .	ix
<b>ABSTRACT</b> . . . . .	x
<b>CHAPTER 1. INTRODUCTION</b> . . . . .	1
1.1 Problem Statement . . . . .	1
1.2 Proposed Approach . . . . .	3
1.3 Organization . . . . .	3
<b>CHAPTER 2. BACKGROUND ON COLONOSCOPY</b> . . . . .	4
2.1 Diagnostic and Therapeutic Operations . . . . .	5
2.2 Human Appendix . . . . .	6
<b>CHAPTER 3. CURRENT STATE-OF-THE-ART OF RELEVANT RESEARCH</b>	8
3.1 Gastrointestinal Endoscopic Research . . . . .	8
3.2 Content-based Video Segmentation . . . . .	9
3.3 Object Detection and Recognition . . . . .	11
<b>CHAPTER 4. SCENE SEGMENTATION</b> . . . . .	14
4.1 Challenges of Colonoscopic Scene Segmentation . . . . .	14
4.2 Audio Analysis Scene Segmentation Algorithm . . . . .	15
4.2.1 Phase 1: Audio Frame Classification . . . . .	16
4.2.2 Phase 2: Speech Segment Detection . . . . .	20
4.2.3 Phase 3: Speech Recognition . . . . .	21

4.2.4	Phase 4: Scene Identification . . . . .	22
4.3	Visual Model Approach to Refine the Scene Boundaries . . . . .	24
4.3.1	Visual Model for Scene Segmentation . . . . .	24
4.3.2	Feature Extraction and Analysis . . . . .	26
4.3.3	Scene Boundary Detection Algorithm . . . . .	27
4.4	Performance Study . . . . .	28
4.4.1	Performance Evaluation of Audio-based Scene Segmentation Method . .	30
4.4.2	Performance Evaluation of Visual Model Scene Segmentation Method .	31
4.5	Summary . . . . .	32
<b>CHAPTER 5. OPERATION SHOT DETECTION . . . . .</b>		<b>33</b>
5.1	Challenges of Operation Shot Detection . . . . .	33
5.2	Spatio-temporal Operation Shot Detection . . . . .	34
5.2.1	Image Preprocessing . . . . .	34
5.2.2	Identification of Instrument Insertion Direction . . . . .	36
5.2.3	Region Filtering, Merging, and Matching . . . . .	40
5.2.4	Shot Segmentation . . . . .	43
5.3	Performance Study . . . . .	45
5.3.1	Determining Important Parameters for the Proposed Approach . . . . .	45
5.3.2	Effectiveness of Cable Detection . . . . .	48
5.3.3	Effectiveness of Operation Shot Detection . . . . .	51
5.4	Summary . . . . .	54
<b>CHAPTER 6. APPENDIX IMAGE CLASSIFICATION . . . . .</b>		<b>55</b>
6.1	Challenges of Appendix Image Classification . . . . .	55
6.2	Feature-based Appendix Image Detection Approach . . . . .	56
6.3	Model-based Appendix Image Detection Approach . . . . .	63
6.3.1	Structure of the Statistical Model . . . . .	64
6.3.2	Learning . . . . .	67
6.3.3	Recognition . . . . .	73

6.4	Performance Study . . . . .	77
6.4.1	Datasets . . . . .	78
6.4.2	Model Training . . . . .	81
6.4.3	Test Results on Images from Colonoscopy Videos . . . . .	84
6.5	Summary . . . . .	89
<b>CHAPTER 7. CONCLUSION AND FUTURE WORK . . . . .</b>		<b>91</b>
<b>BIBLIOGRAPHY . . . . .</b>		<b>93</b>

## LIST OF TABLES

Table 4.1	Reserved terms for dictation for scene segmentation. . . . .	16
Table 4.2	Precision and recall on twenty colonoscopy videos. . . . .	30
Table 4.3	Effectiveness of fade-like detection models on ten colonoscopy videos. . . . .	31
Table 4.4	Precision and recall of three scene segmentation algorithms. . . . .	31
Table 5.1	Characteristics of “Video Set I”. . . . .	46
Table 5.2	Characteristics of “Image set”. . . . .	46
Table 5.3	Characteristics of “Video Set II”. . . . .	47
Table 5.4	Effectiveness of instrument insertion direction identification. . . . .	49
Table 5.5	Effectiveness of region filtering. . . . .	49
Table 5.6	Effectiveness of region merging. . . . .	50
Table 5.7	Accuracy of cable detection. . . . .	51
Table 5.8	Effectiveness of operation shot detection: Overview results. . . . .	53
Table 5.9	Effectiveness of operation shot detection: Detailed results for each video. . . . .	53
Table 6.1	Characteristics of “Image Set I”. . . . .	79
Table 6.2	Characteristics of “Image Set II”. . . . .	80
Table 6.3	Characteristics of “Video Set”. . . . .	80
Table 6.4	Effectiveness of the appendiceal orifice detection on colonoscopy videos. . . . .	90

## LIST OF FIGURES

Figure 2.1	The colon endoscopic segments: 1-cecum, 2-ascending colon, 3-transverse colon, 4-descending colon, 5-sigmoid, 6-rectum. . . . .	4
Figure 2.2	Examples of instruments. . . . .	6
Figure 2.3	Colon images with surgical instruments during colonoscopic procedure. . . . .	6
Figure 2.4	Examples of colon images with appendix: (a) Appendix image with a clearly seen appendiceal orifice in the top center of the image; (b) Appendix image with a clearly seen appendiceal orifice in the top left corner of the image; (c) Appendix image with a clearly seen appendiceal orifice in the top middle of the image. . . . .	7
Figure 4.1	Scenes of endoscopic procedures. . . . .	15
Figure 4.2	Overview of the proposed audio-analysis scene segmentation. . . . .	16
Figure 4.3	Typical pattern of frame tokens. . . . .	17
Figure 4.4	Audio frame classification. . . . .	19
Figure 4.5	Sensitivity analysis for the selection of thresholds. . . . .	20
Figure 4.6	Speech segment detection using a finite state automaton. . . . .	21
Figure 4.7	Finite state automaton for scene identification. . . . .	23
Figure 4.8	Examples of the cornering action. . . . .	25
Figure 4.9	Cornering pattern around a scene boundary. . . . .	26
Figure 4.10	Pattern of standard deviations of DC images in the cornering pattern. . . . .	27
Figure 4.11	Original colon image and the image after the removal of the black surrounding region. . . . .	27
Figure 4.12	Fade-in sequence detector for a cornering pattern. . . . .	29



Figure 5.1	Overview of operation shot detection. . . . .	35
Figure 5.2	Image examples for image preprocessing step: (a) Original color image; (b) Image after removing the light reflected regions; (c) Segmented image using JSEG. . . . .	36
Figure 5.3	Possible triangular areas and insertion directions of instruments: (a) Various components of the tip of a current endoscope model projected on top of the image area; note the position of the working channel in relation to the lens; (b) Eight insertion directions that correspond to eight triangular areas; (b) Eight triangular areas that correspond to eight insertion directions. . . . .	37
Figure 5.4	Image examples for insertion direction determination. . . . .	40
Figure 5.5	(a) The triangular filter in area 6; (b) Eight triangular filters. . . . .	41
Figure 5.6	Example of region matching between an instrument image and a template region. . . . .	43
Figure 6.1	(a)Original appendix image; (b) Location, orientation, and magnitude of interesting points identified by SIFT feature detector. . . . .	57
Figure 6.2	Image examples before and after segmentation: (a) Original lumen image; (b) Lumen image after segmentation; (c) Original appendix image; (d) Appendix image after segmentation. The average intensity of the darkest region for the lumen image is much smaller than the one for the appendix image, which is consistent with our observation (1). . . . .	58
Figure 6.3	Image examples for image enhancement based on Hessian Matrix: (a) Original appendix image; (b) Edge image for image(a) after enhancement; (c) Original image without a clearly seen appendiceal orifice; (e) Edge image for image(b) after enhancement. . . . .	59
Figure 6.4	Derivation of the ideal ellipse A from curve ACu, which is part of the ideal ellipse A and resides in the boundary of ellipse A. . . . .	62
Figure 6.5	Relationship between the appendix curve and the ideal ellipse. . . . .	62

Figure 6.6	Illustration of the mapping between the image and the graph. . . . .	65
Figure 6.7	System overview of detecting the appendiceal orifice under the part-based statistical framework. . . . .	74
Figure 6.8	Positive training images for the appendix image class I. . . . .	79
Figure 6.9	Positive training images for the appendix image class II. . . . .	80
Figure 6.10	The relationship between the number of parts and the error rates for the two image classes. . . . .	85
Figure 6.11	Positive training image examples with parts superimposed for training the model of image class I. (This figure is best viewed in color) . . . .	85
Figure 6.12	Six sub-images for part 0 (root), part 1, part 2, and part 3 cropped from images in Figure 6.11. . . . .	86
Figure 6.13	The first six largest-eigenvalue eigenvectors for part 0 (root), part 1, part 2, and part 3. . . . .	86
Figure 6.14	Illustration of the shape model for the spatial relations between root and leaves. (This figure is best viewed in color) . . . . .	87
Figure 6.15	Thirty ROC curves of thirty runs on training set I of “Image Set II”. .	87
Figure 6.16	Thirty ROC curves of thirty runs on training set II of “Image Set II”. .	88

## ACKNOWLEDGEMENTS

Pursuing a Ph.D degree is a painful but enjoyable journey. It's just like climbing a high peak, step by step, accompanied with bitterness, hardship, frustration, encouragement, trust, and delight. I could never have reached the heights or explored the depths without the help, support, and guidance of a lot of people.

First, I would like to thank my major advisor Dr. Wallapak Tavanapong for her guidance, patience, and support throughout this research and the writing of this dissertation. I would also like to thank my committee members for their efforts and contributions to this work: Dr. Jennifer Davidson, Dr. Vasant Honavar, Dr. Dimitris Margaritis, and Dr. Johnny Wong. I would additionally like to thank Dr. Dimitris Margaritis for his inspiring suggestions.

Over the past five years, I am fortunate to have the opportunity to work with a group of energetic and talented schoolmates (Danyu Liu, Kihwan Kim, Jie Bao, Hua Ming, Sean Stanek, Kung-En Dean Lin, and Dalei Li, etc) in the Department of Computer Science at Iowa State University. I enjoyed every moment that we have worked together including all those late night lab activities. I appreciate the friendships and all their encouragements to finish this dissertation.

I am very grateful for my parents, who have encouraged and assisted me constantly throughout my life, especially in attaining this degree.

Last, but certainly not least, I must acknowledge with tremendous and deep thanks to my wife, Xiaoling Dai, for being so supportive of my efforts and so understanding of the endless deadlines.

## ABSTRACT

Colonoscopy is an important screening tool for colorectal cancer. During a colonoscopic procedure, a tiny video camera at the tip of the endoscope generates a video signal of the internal mucosa of the colon. The video data are displayed on a monitor for real-time analysis by the endoscopist. We call videos captured from colonoscopic procedures “colonoscopy videos”. To the best of our knowledge, they are not captured for post procedural review or analysis in the current practice. Because of the unique characteristics of colonoscopy videos, new types of semantic units and new image/video analyzing techniques are required. In this dissertation, we aim to develop new image/video analysis techniques for these videos to extract important semantic units, such as colonoscopic scenes, operation shots, and appendix images. Our contributions include two parts: (a) new definitions of semantic units (colonoscopic scene, operation shot, and appendix image); and (b) novel image/video analysis algorithms, including novel scene segmentation algorithms using audio and visual information to recognize scene boundaries, new computer-aided detection approaches for operation shot detection, and new image analysis methods for appendix image classification. The new image processing and content-based video analysis algorithms can be extended to videos from other endoscopic procedures, such as upper gastrointestinal endoscopy, EGD, enteroscopy, bronchoscopy, cystoscopy, and laparoscopy. Our research is very useful for the following platforms and resources: (a) platforms for new methods to discover unknown patterns of diseases and cancers; (b) platforms for improving and assessing endoscopists procedural skills; and (c) education resources for endoscopic research.

## CHAPTER 1. INTRODUCTION

### 1.1 Problem Statement

Colorectal cancer is the second most common cancer in both men and women (Jemal et al., 2007). As the name implies, colorectal cancers are malignant tumors that are developed in the colon and rectum. The survival rate is higher if (1) the cancer is found and treated early before metastasis to lymph nodes or other organs occurs, or (2) the colonic polyps are removed during the colonoscopy procedure. Colonoscopy is rapidly becoming the single most important endoscopic screening modality for colorectal cancer. This is because colonoscopy allows inspection of the entire colon (unless there are large lesions in the colon) and provides the ability to perform a number of diagnostic and therapeutic operations (e.g., hot biopsy, polyp removal) during a single procedure.

During a colonoscopic procedure, a tiny video camera at the tip of the instrument generates a sequence of images (frames) of the internal mucosa of the colon. These frames are displayed on a monitor. The endoscopist interprets the displayed images and acts based on his/her knowledge regarding the condition of the patient combined with his/her colonoscopic expertise. The entire procedure typically lasts 20 and 45 minutes. The endoscopist may take pictures of normal or abnormal mucosa of a certain part of the colon for educational purposes or to document the extent of the procedure or specific findings. In current clinic practice, video signals generated during endoscopic procedures are typically not recorded. Recent advances in video compression and capturing hardware and software present excellent opportunities to record these videos in a digital format for real-time and post-procedure analysis for early detection and diagnosis of diseases. We refer to the videos captured from colonoscopic procedures as “*colonoscopy videos*”. Compared with produced videos (for instance, news videos and sports

videos), colonoscopy videos have unique characteristics. For example, due to frequent shifts of camera focus while moving along the colon, colonoscopy videos contain many blurry frames. The lens of the current endoscopes cannot be focused because they are single, wide-angle lens. Many manipulations, such as optimization of the sharpness, brightness and contrast of the image, removal of stool and light scattering substances (via irrigation with water, suction of cleansing material and water, and dispersal of air bubbles), minimal tip movement, appropriate distance from the mucosal surface area, tangential illumination of the mucosa to prevent direct light reflexes, and appropriate light intensity and color settings of the instrument (through balancing of equipment prior to entering the patient), are reflected in colonoscopy videos and may add noise to these videos. Hence, a sequence of images for the same object of interest may exhibit totally different visual properties. Solving these technique challenges requires new types of semantic units and new image/video analysis techniques.

Automatic analysis of colonoscopy semantics from colonoscopy videos is a very important research problem for the following areas: (1) platforms for new methods to discover unknown patterns of diseases and cancers. Currently, we lack easy ways to access colonoscopy video data. Data access is important to apply data mining techniques to discover new abnormal patterns that may lead to better understanding of diseases and cancers; (2) education resources for endoscopic research. Currently, endoscopists typically capture images of interest using proprietary software or occasionally record the entire procedure onto a VHS tape. Although the captured analog video allows post-procedure analysis of the entire colonoscopic procedure, images in the VHS tape are of relative poor quality. It is also time consuming to locate a few interesting images within the entire video; (3) platforms for improving and assessing endoscopists' procedural skills. Currently, endoscopists' skills are indirectly measured. It is important to develop a system that can compare the quality of the procedures performed by different operators, and provide quality measurements as well as educational means to improve procedural skills automatically.

## 1.2 Proposed Approach

To solve the problem of analyzing colonoscopy semantics, we first define three new semantic units: colonoscopic scenes, operation shots, and appendix images. A colonoscopic scene is defined as a segment of visual and audio data that corresponds to an endoscopic segment of the colon. Scene segmentation is the first and necessary step to provide important statistics such as the number of polyps appearing in a scene, various therapeutic operations performed in a scene, and changes in the internal mucosa of the same scene of the same patient over time. These statistics are valuable for diagnosis of colonic diseases. An operation shot is defined as a segment of visual and audio data that corresponds to a diagnostic or therapeutic operation in a colonoscopy video. Operation shots are useful for reviewing causes of complications due to diagnostic or therapeutic operations. An appendix image is defined as a colon image that contains the shape of the opening of the appendix. Appearances of appendix images in colonoscopy videos indicate the complete inspection of the colon, which is one of the important measurements for evaluating the quality of colonoscopic procedure. Based on these new definitions, we first investigate new video segmentation techniques to extract colonoscopic scenes. Then we present our algorithms for operation shot detection. We also introduce two approaches on appendix image classification. To validate our new image/video analysis algorithms, we develop software packages that implement the above algorithms. The software is being integrated into a novel system aiming at automatic analysis for quality measures of colonoscopy.

## 1.3 Organization

The remainder of this paper is organized as follows. Chapter 2 provides background on colonoscopy, diagnostic and therapeutic operations, and appendix. Chapter 3 introduces the current state-of-the-art of relevant research. Chapter 4 presents the scene segmentation techniques. Methods for operation shot detection are introduced in Chapter 5. We discuss detection algorithms for appendix images in Chapter 6. Finally, we offer our concluding remarks and future work in Chapter 7.

## CHAPTER 2. BACKGROUND ON COLONOSCOPY

The colon is a hollow, muscular tube about 150 cm long (NationalCancerInstitute, 2007), as illustrated in Figure 2.1. A normal colon consists of six parts: cecum with appendix, ascending colon, transverse colon, descending colon, sigmoid and rectum. Anatomical landmarks, such as the appendiceal orifice and the terminal ileum, appear in the most proximal part of the colon.

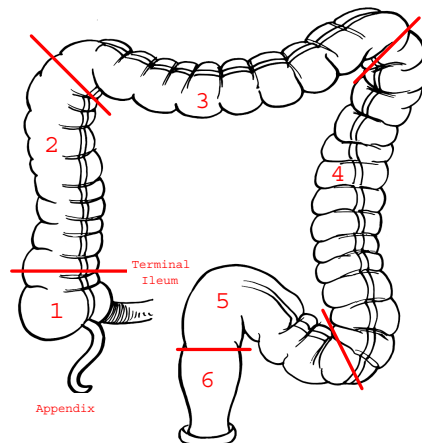


Figure 2.1 The colon endoscopic segments: 1-cecum, 2-ascending colon, 3-transverse colon, 4-descending colon, 5-sigmoid, 6-rectum.

Colonoscopy is a procedure that allows inspection of the colon. Prior to colonoscopic procedures, patients are asked to cleanse the colon. During the colonoscopic procedure, a flexible endoscope (a flexible tube with a tiny video camera at the tip) is advanced under direct vision via the anus into the rectum and then gradually into the most proximal part of the colon or the terminal ileum. Colonoscopy allows inspection of the colonic mucosa and provides the



ability to perform a number of therapeutic operations during a single procedure. Besides the detection of pre-malignant (polyps) or malignant colonic lesions, colonoscopy has a number of other diagnostic and therapeutic applications. These include inspection of the mucosa of the colon and terminal ileum for inflammatory or hemorrhagic lesions, diagnostic tissue sampling, ablation of polypoid lesions, treatments of hemorrhagic lesions, and decompression of distended colonic segments. A colonoscopic procedure consists of two phases: the insertion phase and the withdrawal phase. During the insertion phase, the endoscopist rapidly advances the tip of the endoscope to the most proximal location possible (cecum or terminal ileum). Frequently, but not always, the endoscopist is able to identify important anatomic landmarks such as the end of the sigmoid, the splenic flexure, and the hepatic flexure. Careful mucosal examination, diagnostic and therapeutic operations are typically performed during the withdrawal phase when the endoscope is gradually withdrawn. In a colonoscopic procedure, images with the shape of the opening of the appendix usually appear in the end of the insertion phase or the beginning of the withdrawal phase.

## 2.1 Diagnostic and Therapeutic Operations

An endoscope has instrument channels that allow the insertion of flexible accessories such as biopsy forceps, cytology brushes, sclerotherapy needles, and diathermy snares from a port on the endoscope control head through the shaft and into the field of view. These instruments are used for tissue-sampling and therapeutic procedures. Biopsy forceps used for tissue sampling consist of a pair of sharpened cups, a spiral metal cable, and a control handle. The tissue specimen is used for microscopic examination of its structure or for searching for the presence of infectious agents or *Helicobacter pylori*. “Hot” biopsy forceps (allowing the passage of current) and diathermy snares are used for polyp removal. Figure 2.2 shows some examples of these instruments. Figure 2.3 depicts images from actual colonoscopic procedures when a snare and biopsy forceps are in use. Depending on endoscope models, the instrument may appear in the images at a different position, e.g. the bottom right corner or the bottom left corner.

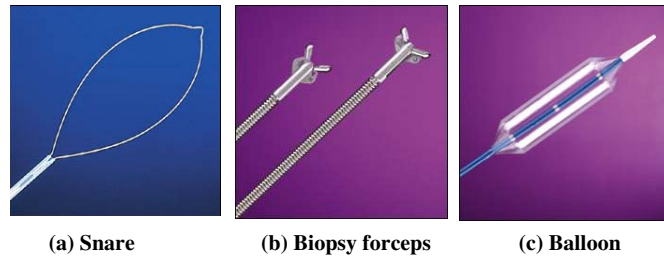


Figure 2.2 Examples of instruments.

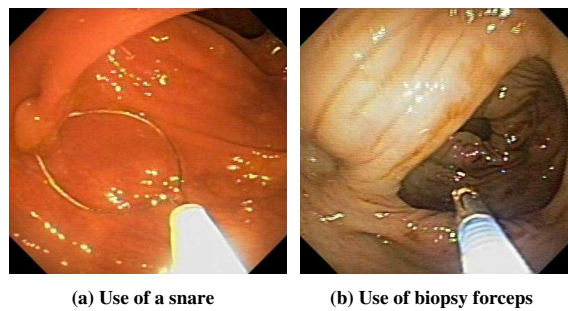


Figure 2.3 Colon images with surgical instruments during colonoscopic procedure.

## 2.2 Human Appendix

Appendix is a small, worm-shaped blind tube. It is about 7.6 cm long and 0.64 cm to 2.53 cm thick, projecting from the cecum on the right side of the lower cavity (ColumbiaUniversity, 2004). The appendix only appears in cecum. Figure 2.4 shows some colon images that contain the shape of the opening of the appendix. The appendiceal orifice is annotated by a white dot rectangle in each image. There are several ellipse shape rings around the center of the appendix. The appendix forms the beginning of the colon. It indicates the end of the insertion phase or the beginning of the withdrawal phase in a colonoscopic procedure.

Recent research on colonoscopy indicates that there is a significant miss-rate for the detection of even large polyps and cancers (Lieberman, 2005). The miss-rate may be related with the experience of the endoscopist and the location of the lesion in the colon, but no prospective

studies related to this have been done so far. In current practice, there is no objective way to measure in detail what exactly is achieved during the procedure although a number of indirect markers of quality have been proposed. These include duration of the withdrawal phase and average number of polyps detected per screening colonoscopy, and thoroughness of inspection of the colon. The presence of a sufficient number of images showing a closely inspected appendiceal orifice is one of the important objective indicators that the most distal end of the colon has been reached during the procedure. Other indicators include presence of small bowel mucosa and ileocecal valve. Reaching the end of the colon is one of the prerequisites for complete inspection. If few or no appendix images are found in a colonoscopy video, the video may require a second opinion to determine whether the entire colon indeed was visualized or not. If we still could not find any evidence of showing the complete inspection of the colon, the patient may have to undergo a second procedure.

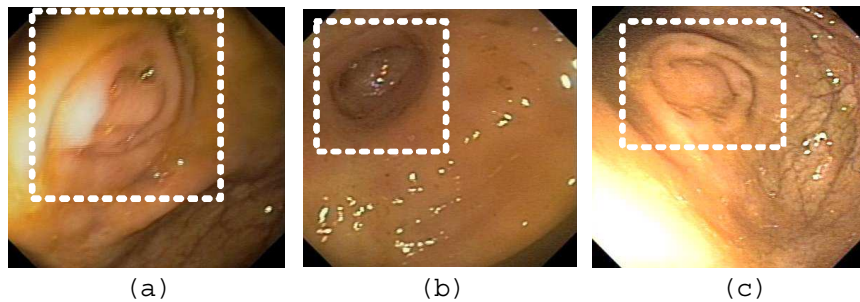


Figure 2.4 Examples of colon images with appendix: (a) Appendix image with a clearly seen appendiceal orifice in the top center of the image; (b) Appendix image with a clearly seen appendiceal orifice in the top left corner of the image; (c) Appendix image with a clearly seen appendiceal orifice in the top middle of the image.

## CHAPTER 3. CURRENT STATE-OF-THE-ART OF RELEVANT RESEARCH

Despite a large body of knowledge in medical image analysis, very little research has been conducted to analyze colonoscopy videos or to provide efficient access to important images and video segments from such videos, or to investigate automatic measurement method to evaluate the quality of the colonoscopic procedure. The most related research efforts are in the areas of gastrointestinal endoscopic research, content-based video segmentation, and object detection and recognition. The following sections introduce the state-of-the-art in these three areas.

### 3.1 Gastrointestinal Endoscopic Research

Research efforts in this area include techniques for guiding a colonoscope (Sucar and Gillies., 1990; Phee and Ng., 1998; Koh and Gillies., 1994) during a colonoscopic procedure, development of colonoscope hardware (Dario and Lencioni., 1997; Khessal and Hwa., 2000; Lim and Lee., 2001), analysis of images from biopsies of colon tissues (Todman et al., 2000; Hamilton et al., 1997; Shuttleworth et al., 2002), classification and identification of colonic carcinoma using microscopic images (Esgiar et al., 1998), detection of tumor in endoscopic videos (Karkanis et al., 2003), and virtual colonoscopy (Lakare et al., 2002; Chen et al., 2000; Lee et al., 2000; Haker et al., 2000; Sharghi and I.W., 2001; Hietala and Oikarinen, 2000). Microscopic images of the colon are captured from tissue samples using a light microscope mounted with a CCD camera. Tissue samples are obtained from sequential resections of the colon. Unlike the colonoscopy videos, microscopic images only reflect the morphology of the tissue in a specific location of the colon mucosa. In (Karkanis et al., 2003), an approach to detect tumors in colonoscopic video is proposed. The main focus of this paper is to detect the small size

adenomatous polyps, given a colonoscopy image. However, we are more interested in identifying important semantic video segments and semantic video objects from a colonoscopy video that corresponding to one colonoscopy procedure. In virtual colonoscopy, a virtual colon is reconstructed from Computer Tomography (CT) cross-sectional images of the abdomen of a patient. CT images are significantly different from those of colonoscopic procedures. Virtual colonoscopy is still in its infancy. Actual colonoscopic procedures are still needed for definitive examinations, histologic samplings, and therapeutic procedures.

### 3.2 Content-based Video Segmentation

Content-based video analysis is a promising paradigm that lets users browse and retrieve desired video segments effectively and efficiently. The first and necessary step for content-based video analysis is video segmentation, which segment the video into smaller but meaningful chunks. Automatic video segmentation techniques are desired since manual segmentation is very time consuming (i.e., ten hours of work for one hour of video (Bimbo, 1999)). Existing video segmentation techniques typically divide a video file into shots, which is defined as a contiguous sequence of video frames recorded from a single camera operation. High-level aggregates of relevant shots termed scenes are then generated for browsing and retrieval. Scenes are important as (i) users are more likely to recall important events rather than a particular shot or frame (Hanjalic et al., 1999); and (ii) the number of shots may be too large for effective browsing (e.g., about 600-1500 shots for a typical film). A typical automatic video segmentation involves three important steps. The first step is *shot boundary detection (SBD)*. A shot boundary is declared if a dissimilarity measure between consecutive frames exceeds a threshold. Examples of recent SBD techniques are (Zhuang et al., 1998; Aigrain and Joly, 1994; Zhang et al., 1993, 1997; Yeo and Liu, 1995; Shin et al., 1998; Gamaz et al., 1998; Dawood and Ghanbari, 1999; Nang et al., 1999). The second step is *key-frame selection* that extracts one or more frames that best represent the shot, termed key-frame(s). The third step is *scene segmentation*.

Current research efforts on *SBD* focus on detection of three types of transitions: hard cut,

fade, and dissolve. Here we briefly describe these techniques as follows.

1. **Hard Cut Detection:** A hard cut is a direct concatenation of two shots, which indicates a temporal visual discontinuity in the video. Existing hard cut detection algorithms detect significant changes in either intensity/color histograms (U.Gargi et al., 2000; Yusoff and Kittler, 2000; Naphade et al., 1998) or edge pixels (R.Zabih, 1999) or motions (Hanjalic and Zhang, 1999) between consecutive frames. For example, if the changes is over a threshold, a shot boundary is declared.
2. **Fade Detection:** A production model of a fade sequence  $S(x, y, t)$  of duration  $T$  is defined as the scaling of pixel intensities/color of a video sequence  $S_1(x, y, t)$  by a temporally monotone scaling function  $f(t)$  (Hampapur et al., 1995).

$$S(x, y, t) = f(t) \times S_1(x, y, t), t \in [0, T] \quad (3.1)$$

For a fade-in sequence,  $f(0) = 0$  and  $f(T) = 1$ , while  $f(0) = 1$  and  $f(T) = 0$  for a fade-out sequence. Typically,  $f(t)$  is a linear function. It was observed that a fade detector based on edge changes does not perform as well as a fade detector based on changes in standard deviations of pixel intensities (Lienhart, 1999).

3. **Dissolve Detection:** A dissolve sequence is defined as a combination of two sequences where the first sequence is fading out and the second sequence is fading in. Existing dissolve detectors utilize changes in pixel intensities (Nam and Tewfik, 2000) or variances (Truong et al., 2000) or edges/contours (Lienhart, 1999) to detect dissolves.

Existing scene segmentation techniques can be divided into two categories: one using only visual features (Rui et al., 1999; Hanjalic et al., 1999; Yeung and Liu, 1995; Corridoni and Bimbo, 1998; Lin and Zhang, 2000; Veneau et al., 2000) and the other using both visual and audio features (Sundaram and Chang, 2000a,b; Adams et al., 2000; Cao et al., 2003, 2004b,c). In both categories, visual similarities of entire shots or key-frames (i.e., global color histograms or color moments) are used for clustering shots into scenes. That is, global visual features of nearby shots are compared. If the dissimilarity measure of the features representing the shots

is within the threshold, these shots and the shots between them are considered in the same scene. Global features, however, tend to be too coarse for shot clustering because they include noise-objects that are excluded when human beings group shots into scenes. Determining the appropriate areas of video frames (or objects) to use for correct shot clustering is challenging even if objects can be reliably recognized using advanced object recognition techniques.

### 3.3 Object Detection and Recognition

Object detection and recognition is one of the fundamental problems in computer vision. Much of the previous work has focused on extracting features from the image followed by matching or classification algorithms. We call this approach feature-based approach. Methods in this category range from simple template matching to sophisticated model-based methods. The idea of template matching is to create a template (or kernel) of an object and search over the image of interest to identify the object by measuring the similarity between the image and the template. Common examples of features used to measure the image similarity include the cross-correlation coefficient (Briechele and Hanebeck, 2001), Fourier descriptors (Aguado et al., 1996), and texture features (Tan, 1998). Model-based approaches treat the recognition task as a combinatorial problem and focus on efficiently searching for correspondences between the model and the image features. Instead of searching through all possible locations in simple template matching methods, model-based approaches always use various heuristics to guide and improve the search. Different statistic models are frequently used in this approach. Under the statistic framework, combined with different types of machine learning techniques, the selection of the model parameters are usually more flexible and more accurate. Typical examples include tree search-based methods (Ayache and Faugeras, 1986; Grimson and Lozano-Prez, 1987), alignment-based methods (Fischler and Bolles, 1997; Huttenlocher and Ullman, 1990).

Another class of recognition methods processes images directly instead of extracting features first. Eigenfaces method (PCA) (Turk and Pentland, 1991) was a classic example. There are two stages (learning and recognition) in this method. In the learning stage, principal component analysis is performed on the training images and the principal subspace (or feature

space) is obtained. In the recognition step, the testing image is linearly reconstructed in the obtained feature space. The distance between the reconstructing weights of the testing image and the reconstructing weights of each training image is computed. Then the smallest distance is selected and this distance is used to determine the existence of the object in the testing image. Similar approaches can also be found in (Huttenlocher et al., 1993; Murase and Nayar, 1995).

Most of the above efforts typically solve the problem of recognition of specific objects such as faces. A more difficult problem is object class recognition, which categorizes the objects into object classes. It requires a generic model that can handle a large intra class variance. Part-based model that represents the object in terms of a set of parts offers a possible solution. This distributed model captures the appearances of the local parts and the spatial relations among parts. There are many flavors of part-based representations. Early research on part-based object detection focused on deterministic approaches with energy minimization (M.A.Fischler and R.A.Elschlager, 1973). In (Weber et al., 2000; Fergus et al., 2003), a joint probabilistic model called constellation model was proposed. It models multiple parts distributed normally in appearance and location space. Appearance variations of object parts are modeled by Principal Component Analysis. The spatial relations among parts are captured by a global joint Gaussian probability distribution function. Weakly supervised learning of the model parameters has been developed using the Expectation-Maximization algorithm. Another well-known part-based model is called pictorial model (Felzenszwalb and Huttenlocher, 2005). It models an object as a collection of parts arranged in a deformable configuration. It treats an object as a graph-like entity. The nodes represent the object parts and the edges indicate the spatial relations among parts. Originally, the pictorial model was used for object localization. In (Crandall et al., 2005), the authors expand the capacity of the pictorial model by providing efficient matching algorithms using general  $K$ -fan graphs. The parameter  $K$  controls both the representational capacity of the models and the computational cost of doing inference with them. For example, when  $k = 0$ , the locations of different object parts are independent (no dependence exists). When  $k = n - 1$  (where  $n$  is the number of parts in the model), there are



dependencies between all pairs of parts. Generally, the larger value of  $k$ , the more computation cost is needed to capture the relations among parts. This model provides a natural way of relating different spatial priors that have been used for recognizing generic classes of objects.

Recent years have seen some interests in applying a part-based approach to medical imaging analysis. For example, Towers et al. (Toews et al., 2006) proposed a part-based appearance model to address the inter-subject MR brain image matching. In (Toews et al., 2006), the part-based model consists of a collection of localized image parts whose appearance, geometry and occurrence frequency are quantified statistically instead of global image representations such as active appearance models. This model addresses the problem that one-to-one correspondence does not exist between subjects due to anatomical differences. Solving the inter-subject variability problem is important for us to understanding how individuals vary within a population in the task of inter-subject registration, for example, determining correspondence between images of different subjects of a population.

## CHAPTER 4. SCENE SEGMENTATION

In this chapter, we present our new audio-visual analysis approach for scene segmentation. In Section 4.1, we define scene, a new type of semantic unit for colonoscopy videos and introduce the challenges of scene segmentation. In Section 4.2, we present the audio-based scene segmentation algorithm as the first step. In Section 4.3, we discuss how to apply the visual analysis approach as the second step to refine the scene boundaries resulting from the first step. Finally, we present our experimental results on colonoscopy videos in Section 4.4.

### 4.1 Challenges of Colonoscopic Scene Segmentation

We define a scene as a segment of visual and audio data that correspond to an endoscopic segment of the colon. Since a typical colon has six different parts and as the terminal ileum is also reachable during endoscopy, in a complete colonoscopic procedure, a total of thirteen scenes are expected: seven scenes from the insertion phase and six scenes from the withdrawal phase, as shown in Figure 4.1(a). Because a scene corresponds to an important endoscopic segment of the colon, the identification of scenes is necessary to provide important statistics such as the number of polyps appearing in a scene, various therapeutic operations performed in a scene, and changes in the internal mucosa of the same scene of the same patient over time. These statistics are valuable for diagnosis of colonic diseases. Note that scenes with the same name in the insertion phase and the withdrawal phase typically do not contain similar images. The length of the scenes with the same name is typically different. This is because during the withdrawal phase, the endoscopist carefully examines each part of the colon whereas in the insertion phase, the endoscopist typically attempts to reach to the most proximal part of the colon as fast as possible. To apply this new scene definition to other endoscopic

procedures, we only need to change “colon” to the organ of interest. For instance, scenes of videos from upper gastrointestinal endoscopy (EGD) are shown in Figure 4.1(b). Because of camera movements, patient’s conditions, and different ways an endoscopist can maneuver the endoscope, visual properties alone are insufficient for scene segmentation. Common color features that are popularly used for segmentation of produced videos (for instance, news videos and sports videos) are not so useful since the different parts of the colon have similar color.

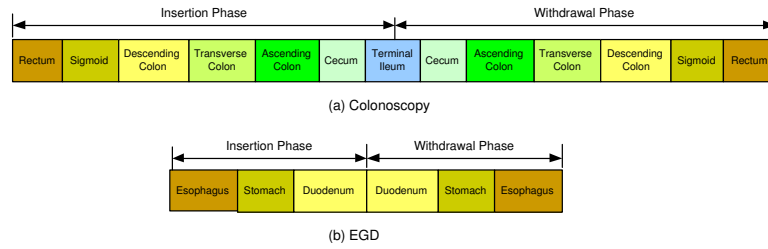


Figure 4.1 Scenes of endoscopic procedures.

## 4.2 Audio Analysis Scene Segmentation Algorithm

In this dissertation, we investigate an alternative approach that utilizes the endoscopist’s comments and domain knowledge for scene segmentation. Since an endoscopy unit only generates a sequence of images with no audio information, we developed a capturing system that allows recording of both the video signal from the endoscopy unit and the endoscopist’s dictation when the tip of the endoscope is moving from one colonic segment into the next in real-time. The captured videos do not contain any patient identifiable information. Although the endoscopist can readily identify the location of the tip in straight, clean colons during the insertion and withdrawal phases, this is not always the case in tortuous colons or in conditions with limited visibility due to faeces. In such colons, definite location is only possible during the withdrawal phase, when the endoscopist has an impression of the entire colon and its position within the abdominal cavity. To facilitate scene segmentation, we have developed a set of reserved terms to be spoken by the endoscopist during the colonoscopic procedure (see Table

4.1). These terms were tested and found practical. The endoscopist also provides dictation when seeing tumors, polyps or when performing biopsy and other therapeutic procedures. The overview of the scene segmentation algorithm is depicted in Figure 4.2 (Cao et al., 2004b).

Table 4.1 Reserved terms for dictation for scene segmentation.

Insertion Phase	Withdrawal Phase
Entering rectum	Leaving terminal ileum
Leaving rectum, entering sigmoid	Back in cecum and ascending colon
Leaving sigmoid, entering descending colon	Leaving ascending colon, back in transverse colon
Leaving descending colon, entering transverse colon	Leaving transverse colon, back in descending colon
Leaving transverse colon, entering ascending colon	Leaving descending colon, back in sigmoid colon
Cecum	Leaving sigmoid back in rectum
Entering terminal ileum	Retroflexion
	All done, taking out air

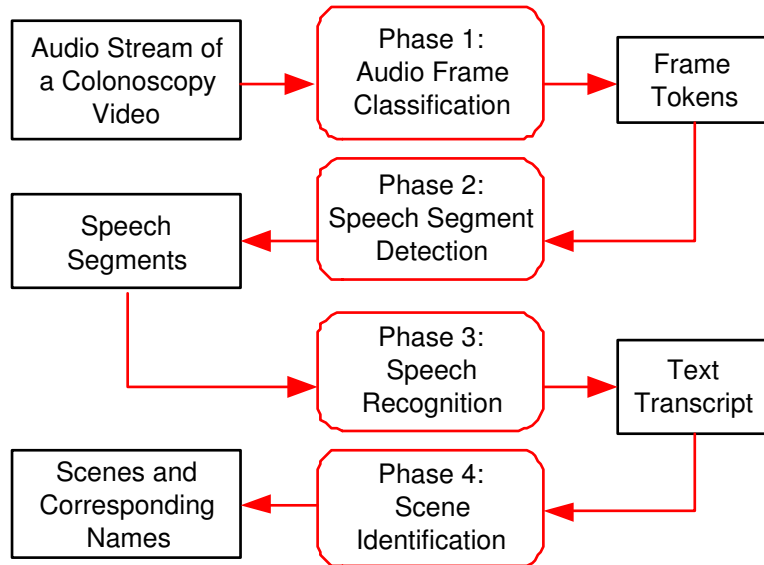


Figure 4.2 Overview of the proposed audio-analysis scene segmentation.

#### 4.2.1 Phase 1: Audio Frame Classification

This is the first phase in our algorithm. It accepts audio stream (composed of a sequence of audio frames) as input and classifies each audio frame into four categories: silence, marker, speech, and background types. The silence type is assigned to audio frames with very low

amplitude whereas the speech type is assigned to audio frames with the endoscopist’s voice. The marker type is for audio frames with special noise indicating the change in the status of the microphone. The background type is assigned to audio frames with unvoiced speech such as the paging system, and electrical noise of the microphone.

Hereafter, a classified audio frame is called a frame token. The rationale for the four types is based on the observation that most speech segments in our collection of colonoscopy videos have the pattern of frame tokens as shown in Figure 4.3.

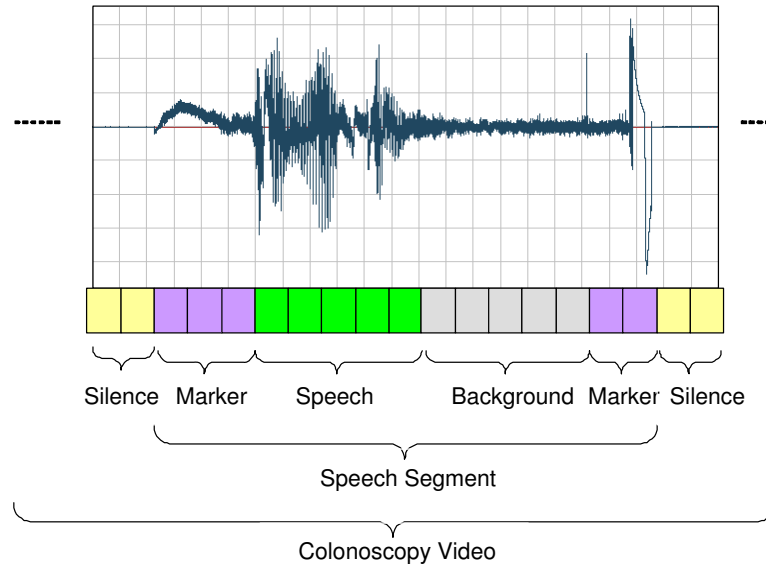


Figure 4.3 Typical pattern of frame tokens.

We select four existing audio features (Lu et al., 2001): Short-Time Energy, Zero-Crossing Rate, Pitch, and Spectrum Flux, to classify each audio frame. We provide the formula to compute these features here to make the dissertation self-contained. Let  $N$  denote the frame length (the number of audio samples in a frame), and  $S_n(i)$  denotes the  $i$ th sample of the  $n$ th audio frame.

- Short Time Energy (STE) is a reliable indicator for silence detection. Specifically, the STE value of frame  $n$  is computed as follows.

$$STE(n) = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} S_n(i)^2} \quad (4.1)$$

- Zero-Crossing Rate (ZCR) is a useful feature to characterize different non-silence audio signals. It is one of the most indicative and robust features to distinguish unvoiced speech. The ZCR value of a frame is defined as the number of times the audio waveform crosses the zero axis. It can be expressed mathematically as follows.

$$ZCR(n) = \frac{1}{2} \left( \sum_{i=1}^{N-1} \left| \text{sign}(S_n(i)) - \text{sign}(S_n(i-1)) \right| \right) \frac{f_s}{N} \quad (4.2)$$

where  $f_s$  is the sampling rate, and  $\text{sign}(\cdot)$  is a sign function.

- Pitch is typically computed using the fundamental frequency of an audio waveform. Normally, only voiced speech and harmonic music have well-defined pitch. There are several different methods to extract pitch information. Here, we use a simple temporal estimation method to extract pitch.

$$R_n(l) = \sum_{i=0}^{N-l-1} S_n(i) S_n(i+l) \quad (4.3)$$

where  $l$  is a constant in the range of 1 to  $N$ .

- Spectrum Flux (SF) is defined as the average variation value of spectrum between two adjacent frames in a short-time analysis window of length one second. SF has been shown to be effective for discriminating speech and environmental sound.

$$SF = \frac{1}{(N-1)(K-1)} \sum_{n=1}^{N-1} \sum_{k=1}^{K-1} \left( \log(A(n, k) + \delta) - \log(A(n-1, k) + \delta) \right) \quad (4.4)$$

where  $A(n, k)$  is the Discrete Fourier Transform (DFT) of the samples in frame  $n$  with order  $k$ .

$$A(n, k) = \left| \sum_{m=-\infty}^{\infty} x(m)w(nL - m)e^{\delta \frac{2\pi}{L} km} \right| \quad (4.5)$$

where  $x(m)$  is the original audio sample;  $w(m)$  is the window function, and  $L$  is the window length.  $\delta$  is a very small value to avoid calculation overflow, and  $K$  is the order of DFT

---

```

Start from the beginning of the audio stream
for each frame in the audio stream do
  Compute the STE value of the frame
  If the STE value is at most the STE threshold
    return Silence
  Compute the ZCR value of the frame
  If the ZCR value is below the ZCR threshold
    return Marker
  Compute the R (Pitch) and SF values
  If the R value is larger than the R threshold and the SF value is below the SF threshold
    return Speech
  Else return Background

```

---

Figure 4.4 Audio frame classification.

Figure 4.4 presents the algorithm for audio frame classification. The algorithm is easy to implement and each phase has linear running time since it scans the input audio stream only once.

We determine the values of the thresholds for each type of audio frames by performing sensitivity analysis on the effect of each threshold. For silence detection, we chose STE threshold to be zero as it gives the highest accuracy as shown in Figure 4.5(a). To detect the marker type, we varied the ZCR threshold from 5 to 400 and measured the accuracy. We selected 150 as ZCR threshold since it offers the highest accuracy (see Figure 4.5(b)). To discriminate between speech and background, SF and pitch are used. For the same SF value, Figure 4.5(c) shows that the higher the pitch threshold, the lower the accuracy for background classification, but the higher the speech classification. So we chose the pitch threshold that offers the best accuracy for both categories (1000). For the same pitch threshold, the accuracy is good when

the SF threshold is set to 30. The effectiveness of our frame classification is over 97% based on our experiments. This is attributed to the choice of the audio features and the appropriate threshold values we use.

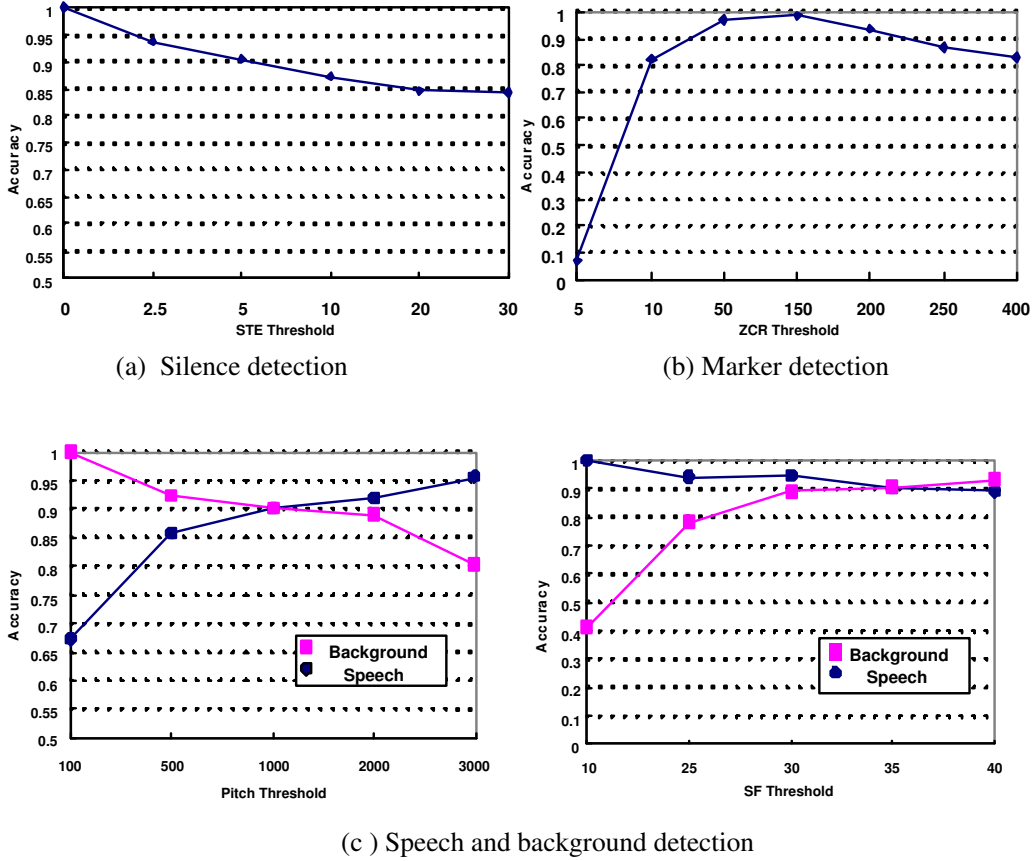


Figure 4.5 Sensitivity analysis for the selection of thresholds.

#### 4.2.2 Phase 2: Speech Segment Detection

This phase locates the speech segment based on frame tokens using a Finite State Automaton (FSA). The FSA recognizes the following regular expression of the frame tokens. This expression denotes a pattern that starts with one or more *Marker* tokens, followed by one or more *Speech* or *Background* tokens, and ended with one or more *Marker* tokens.



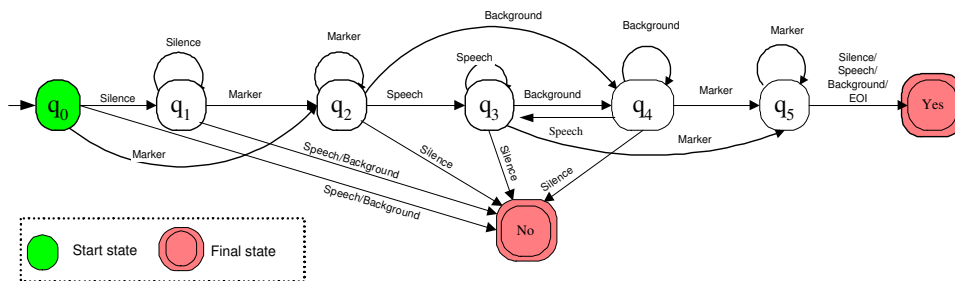
$$Marker^+ \cdot (Speech \vee Background)^+ \cdot Marker^+ \quad (4.6)$$

Formally, this FSA is defined as a quintuple as shown in Figure 4.6(a) and the finite state diagram of the FSA in Figure 4.6(b).

Given a sequence of types of audio frame tokens, the FSA records the beginning and the ending time of the speech segment if it ends in the “Yes” state. The FSA does not record any information if it ends in the “No” state.

$Q_1$	Set of 8 states: $q_0, q_1, q_2, q_3, q_4, q_5, Yes, No$				
$\Sigma_1$	Set of input symbols={Silence, Marker, Speech, Background}				
$q_0$	Start state				
$F_1$	Set of final state $F_1 = \{Yes, No\}$				
$\delta_1$	Transition function that maps $Q_1 \times \Sigma_1$ to $Q_1$ :				
		<b>Silence</b>	<b>Marker</b>	<b>Speech</b>	<b>Background</b>
	$q_0$	$q_1$	$q_2$	No	No
	$q_1$	$a_1$	$q_2$	No	No
	$q_2$	No	$q_3$	$q_3$	$q_4$
	$q_3$	No	$q_5$	$q_3$	$q_4$
	$q_4$	No	$q_5$	$q_3$	$q_4$
$q_5$	Yes	$q_5$	Yes	Yes	

(a)  $M_1 = (Q_1, \Sigma_1, q_0, F_1, \delta_1)$



(b) Finite state diagram

Figure 4.6 Speech segment detection using a finite state automaton.

### 4.2.3 Phase 3: Speech Recognition

This phase accepts the speech segment as input and outputs the corresponding text transcript. Existing speech recognition techniques provide satisfactory performance when dealing

with short speech segments from the same person. Since most of the speech segments generated from Phase 2 are less than ten seconds, it is sufficient to use existing recognition software to perform speech-to-text translation. In our implementation, we use Sphinx 2 recognition software (CarnegieMellonUniversity, 2003). Other speech recognition engines can be used. The output of this phase is the associated text transcript of the speech segment.

#### 4.2.4 Phase 4: Scene Identification

This is the last phase (Phase 4) of our method. This phase aims for an understanding of each speech segment using text transcript from Phase 3 and domain knowledge. We employ another Finite State Automaton (FSA) in this phase. Recognized words are categorized into six categories as follows.

- *Location category* includes the terms describing important anatomic landmarks of the colon such as “cecum”, “terminal ileum”, “ascending colon”, “transverse colon”, “descending colon”, “sigmoid”, “rectum”.
- *Action category* includes the terms indicating the action of the endoscopist such as “entering”, “leaving”, and “back in”.
- *Position category* consists of the terms indicating the position of the endoscope such as “begin”, “end”, “in the middle of”.
- *Abnormal category* includes the terms indicating abnormality such as “polyp” and “cancer”.
- *Error category* has the terms indicating errors that have been previously made such as “sorry” and “wrong”. The error category is used to handle the case that the endoscopist corrects his misunderstanding about the location of the camera during the insertion phase of the colonoscopic procedure.
- *Unused category* includes words that cannot be classified in other categories such as a, an, the, and non-communicative words such as uh. We use this last category to remove

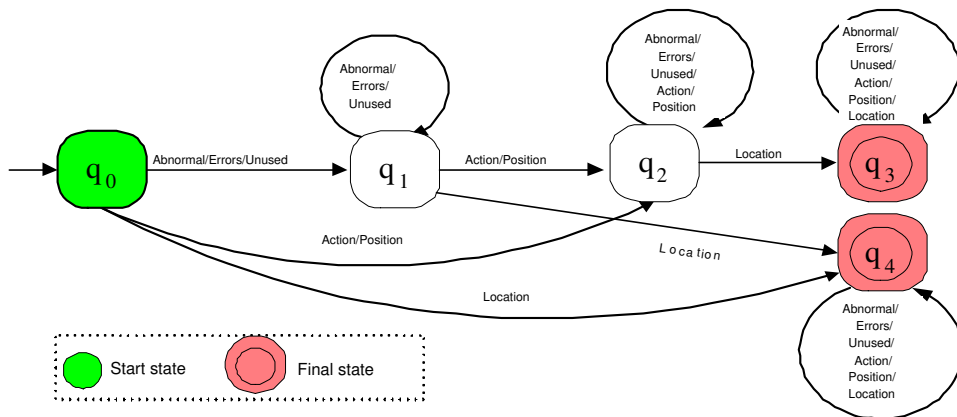
words that are not useful.

The finite state automaton (FSA) for this phase is formally defined in Figure 4.7(a). The corresponding finite state diagram is shown in Figure 4.7(b). The FSA recognizes the following regular expression. This expression represents a pattern that starts with zero or more *Action* or *Position*, followed by exactly one *Location*.

$$(Action \vee Position)^* \cdot Location \tag{4.7}$$

$Q_2$	Set of 5 states: $q_0, q_1, q_2, q_3, q_4$			
$\Sigma_2$	Set of input symbols={Location, Action, Position, Abnormal, Errors, Unused}			
$q_0$	Start state			
$F_2$	Set of final states $F_2 = \{q_3, q_4\}$			
$\delta_2$	Transition function that maps $Q_2 \times \Sigma_2$ to $Q_2$			
		Abnormal, Errors, Unused	Action, Position	Location
	$q_0$	$q_1$	$q_2$	$q_4$
	$q_1$	$q_1$	$q_2$	$q_4$
	$q_2$	$q_2$	$q_2$	$q_3$
	$q_3$	$q_3$	$q_3$	$q_3$
	$q_4$	$q_4$	$q_4$	$q_4$

(a)  $M_2 = (Q_2, \Sigma_2, q_0, F_2, \delta_2)$



(b) Finite state diagram

Figure 4.7 Finite state automaton for scene identification.

To understand the meaning of the sequence of words produced by Phase 2, each word is mapped into one of the six pre-defined types. The FSA processes the type of each word one by one and changes its state according to the input sequence. When our FSA falls in one of the final states, the FSA records the time of the beginning and the ending of the scene and the corresponding name. The final state  $q_4$  indicates the case that the endoscopist’s comment only has location information (e.g., “rectum”) whereas the final state  $q_3$  indicates that the endoscopist’s comment provides more information. Based on the transcript and the timestamp of each speech segment, we obtain the scene boundaries as follows. Starting from the first speech segment, we locate the nearest speech segment that has the same name (e.g., rectum in “entering rectum” and in “leaving rectum, entering sigmoid”). The starting time of the former speech segment and the ending time of the latter speech segment indicate the scene boundaries.

To extend this phase to process videos from other endoscopic procedures such as EGD, we only need to add appropriate terms in the location category: esophagus, stomach, and duodenum. Similarly, the endoscopist can use different terms for other categories such as the abnormal categories to reflect the standard of terms acceptable within that community.

### 4.3 Visual Model Approach to Refine the Scene Boundaries

By applying the audio analysis scene segmentation algorithm, we are able to identify the majority of the scene boundaries. However, some scenes may not be detected because the endoscopist’s speech is not recognized by the speech recognition software. To determine the missing scene boundaries, we apply our visual analysis method based on our new visual model as follows (Cao et al., 2004c,a).

#### 4.3.1 Visual Model for Scene Segmentation

Based on our observations and consultations with our endoscopist, we observe a specific pattern appearing around 60% of scene boundaries in colonoscopy videos. We call this pattern the *cornering pattern* as it corresponds to the endoscopist’s action of steering the endoscope

around the cornering parts of the colon (see Figure 4.8) (i.e., cecum and terminal ileum, ascending and transverse colons, transverse and descending colons, and descending and sigmoid colons). The cornering pattern consists of three sequences of images (see Figure 4.9). The first sequence is composed of images with recognized edges. The second sequence has all blurry images—images with unclear edges. The transition between these two sequences is quite abrupt like a hard cut in produced videos. The third image sequence is like a fade-in sequence with a gradual increase in pixel intensities/color and edges. This sequence happens as the endoscopist starts to recognize some part of an anatomic landmark and gradually adjusts the camera position to make the image clearer. Existing production models (Hampapur et al., 1995; Truong et al., 2000) cannot capture the cornering pattern. Hence, we propose a new visual model for this pattern. Let  $S_1(x, y, t)$ ,  $S_2(x, y, t)$ , and  $S_3(x, y, t)$  represent the first, the second, and the third image sequences, respectively. The spatial dimension is represented by  $x$  and  $y$  and the temporal dimension is represented by  $t$ . Hence, the cornering pattern  $S(x, y, t)$  is defined in Equation(4.8).

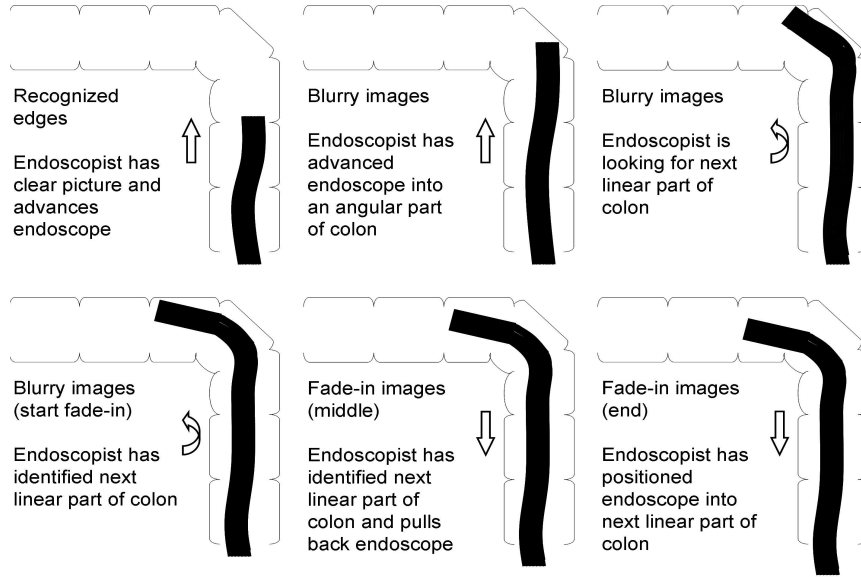


Figure 4.8 Examples of the cornering action.

$$S(x, y, t) = (1 - H(t - t_1)) \times S_1(x, y, t) +$$

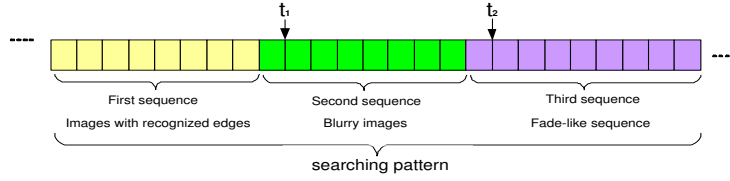


Figure 4.9 Cornering pattern around a scene boundary.

$$\begin{aligned}
 &H(t - t_1) \times (1 - H(t - t_2)) \times S_2(x, y, t) + \\
 &H(t - t_2) \times f(t - t_2) \times S_3(x, y, t)
 \end{aligned} \tag{4.8}$$

where  $t_1$  denotes the timestamp of the first frame after the first sequence and  $t_2$  is the timestamp of the first frame after the second sequence (see Fig 4.9).  $H(t)$  is a function that outputs 1 when  $t \geq 0$  and 0 otherwise. When  $t < 0$ ,  $f(t)$  produces zero; otherwise, the function is a temporally scaling function. This function is typically not a linear function as in the case of a production model for a typical fade sequence.

### 4.3.2 Feature Extraction and Analysis

Since our colonoscopy videos are already encoded in MPEG-2, we extract visual features directly from the compressed videos to reduce the segmentation time. We first obtain a DC-image from the Y-color plane (intensity) of each frame using the techniques in (Yeo and Liu, 1995). A DC-image is a spatially reduced version of the original image. We compute the standard deviation of DCT coefficients in each DC-image. This is based on our observation that the distribution of the standard deviations of the DC images in the cornering pattern often follows the pattern in Fig. 4.10. That is, the standard deviation of each DC-image in the second sequence is generally small and smaller than those of the frames in the other two sequences. We call the second sequence *monotone sequence*. We observe that the standard deviations of the frames in the fade-in sequence can be modeled using a curve fitting method. We choose a linear regression model to describe the standard deviations of the frames in the third sequence by one or more linear function. The challenge is to find the ending frame of each linear curve automatically. Hence, the scaling function  $f(t)$  in Equation(4.8) may be a

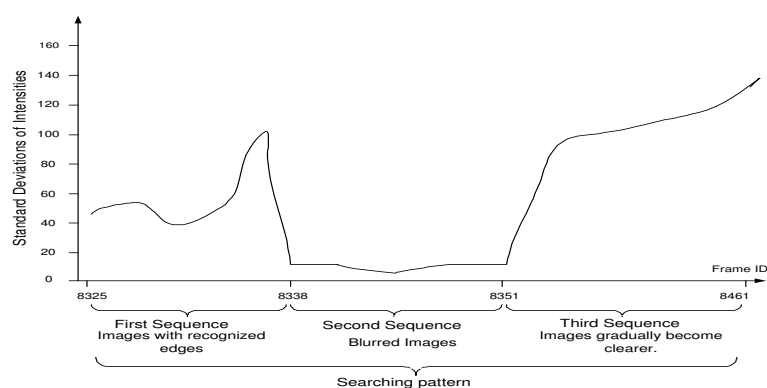


Figure 4.10 Pattern of standard deviations of DC images in the cornering pattern.

combination of one or more linear function.

### 4.3.3 Scene Boundary Detection Algorithm

**Step 1: Preprocessing:** Since more than 99% of the scene boundaries fall in the speech segments, we restrict visual analysis on the video segments corresponding to the endoscopist's speech segments excluding those that contain the keyword in the abnormal category. This is because the terms in this category are very specific and irrelevant to scene boundaries. Next, we apply the filter that removes the black area (the area with DC coefficients below a threshold) surrounding the useful portion of the image (see Fig. 4.11)

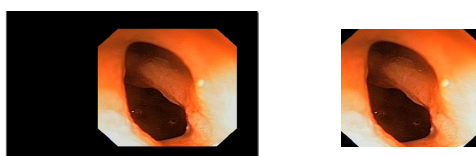


Figure 4.11 Original colon image and the image after the removal of the black surrounding region.

**Step 2: Detection of a monotone sequence:** A sequence of consecutive frames is declared as a monotone sequence if it has at least a pre-defined minimum number of consecutive frames with the standard deviation of each of these frames below a *monotone threshold*.

**Step 3: Hard Cut Detection:** To check a discontinuity between the first sequence and the monotone sequence, we use a sliding-window of size  $2w + 1$  consecutive frames. The parameter  $w$  is set to be smaller than the minimum duration between two sequence changes. For example, setting  $m = 30$  for 30 frames per second video means that there can not be two sequence changes within one second. We first position the center of the sliding window at the frame immediately before the first frame in the monotone sequence. We derive a sequence of bin-wise histogram differences between DC-images of two consecutive frames in the window. We declare a hard cut at the center of the sliding-window if the histogram difference of the two consecutive frames at the center is the largest within the window, and the ratio between the largest difference and the second largest difference in the window is larger than a predefined *hard-cut ratio*. If a cut is not found, we slide the window away from the monotone sequence by one frame. The same process is repeated until a cut is found or a given number of frames before the monotone sequence have been checked. In the latter case, no hard cut is detected.

**Step 4: Detection of a fade-in sequence:** We check whether two linear curves fit well with the standard deviations of the coefficients of DC-images after the monotone sequence using the algorithm in Fig. 4.12.

**Step 5: Boundary Identification:** If both a monotone sequence and a fade-in sequence are detected, the scene boundary is declared at the first frame after the ending frame of the fade-in sequence. However, if a hard cut and a monotone sequence are detected without the fade-in sequence, we declare the scene boundary at the hard-cut location.

#### 4.4 Performance Study

As shown in the previous sections, our approach is divided into two steps. The first step (Section 4.2) uses the audio analysis approach to perform scene segmentation. The second step (Section 4.3) applies the visual model based method to refine the scene boundaries from the first step. In the following sections, we present our experimental results for each step.



---

```

/* Let  $\sigma_i$  be the standard deviation of the coefficients in the DC-image of frame  $i$  */
 $e :=$  frame ID of the last frame in the monotone sequence
 $i := 0; c := 0;$ 
repeat
   $n := 2;$  /* consider the ending frame of the previous sequence and  $n$ 
             subsequent frames */
  repeat /* correlation coefficient value is in the range  $[0, 1]$  */
     $r_1^2$  is a correlation coefficient of  $\sigma_e, \dots, \sigma_{e+n}$ 
     $r_2^2$  is a correlation coefficient of  $\sigma_e, \dots, \sigma_{e+n+1}$ 
     $n := n + 1;$ 
  while  $r_1^2 - r_2^2 < (0.05 \cdot r_1^2)$  /* the change in correlation values is small */
  if  $r_1^2 > 0.8$  then  $c := c + 1;$  /* a linear curve fits well with the values */
   $i := i + 1; e := e + n;$ 
while  $i < 2;$ 
if  $c = 2,$  a fade-in sequence is detected

```

---

Figure 4.12 Fade-in sequence detector for a cornering pattern.

All the videos in our experiments were captured by the same endoscopist. Our test data set consists of twenty colonoscopy videos captured during colonoscopic procedures. The video format is MPEG-2, which is composed of audio and video stream. In the audio analysis approach, we use Sphinx 2 recognition software (CarnegieMellonUniversity, 2003) for the phase 3 in section 4.2.3. Twenty videos are used in our experiments. We use the scene boundaries determined manually as the reference and gather the following performance metrics. Note that, the partially identified scenes, like a scene with a correct identified start frame yet an incorrect identified end frame, are not counted as Relevant Scene.

For each video, we measure the values of *Relevant*, *Irrelevant*, and *Missed*. *Relevant* indicates the number of correct scenes identified by the program whereas *Irrelevant* denotes the number of scenes incorrectly detected by the program. *Missed* shows the number of correct scenes that are undetected by the program. *Recall* is defined as the ratio of the value of *Relevant* to the sum of the corresponding *Relevant* and *Missed* values. *Precision* is defined as the ratio of the value of *Relevant* to the sum of the corresponding *Relevant* and *Irrelevant* values. High recall and precision are desirable.

#### 4.4.1 Performance Evaluation of Audio-based Scene Segmentation Method

In this section, we discuss our experimental results on audio analysis approach in Section 4.2. The two finite state automata, the terms, and the categories used in the scene identification phase are stored in a configuration file. The file is read by the segmentation software as an input. Hence, the segmentation software can be extended to recognize other videos of other endoscopic procedures by modifying the configuration file.

We tested the scene segmentation program on twenty colonoscopy videos. Table 4.2 illustrates the effectiveness of the program. The last row of Table 4.2 shows the average precision and average recall of 0.95 and 0.80 over twenty videos, respectively. The precision and recall quantitatively indicate that our segmentation algorithm performs well. Despite high precision and recall, the program did miss a few scenes. This is mostly because the speech recognition technique does not recognize the endoscopist’s voice indicating the location of the tip of the endoscope.

Table 4.2 Precision and recall on twenty colonoscopy videos.

<i>ID</i>	<i>Relevant</i>	<i>Irrelevant</i>	<i>Missed</i>	<i>Precision</i>	<i>Recall</i>
03001	8	1	5	0.89	0.62
03007	9	1	4	0.90	0.69
03009	10	1	3	0.91	0.77
03010	11	0	2	1.00	0.85
03014	10	1	3	0.91	0.77
03015	13	0	0	1.00	1.00
03017	9	1	4	0.90	0.69
03019	9	1	4	0.90	0.69
03020	13	0	0	1.00	1.00
03047	9	1	4	0.90	0.69
03062	10	2	3	0.83	0.77
03133	11	0	2	1.00	0.85
03148	12	0	1	1.00	0.92
03152	12	0	1	1.00	0.92
03163	12	0	1	1.00	0.92
03177	8	1	5	0.89	0.62
03179	9	0	4	1.00	0.69
03185	12	0	1	1.00	0.92
03190	10	1	3	0.91	0.77
03197	11	0	2	1.00	0.85
<i>Average</i>	10.40	0.55	2.60	0.95	0.80

#### 4.4.2 Performance Evaluation of Visual Model Scene Segmentation Method

In this section, we present our performance study by applying the visual model approach (Section 4.3) to refine the scene boundaries from the previous step. We show the performance of the fade-like detector using one linear curve (“Model 1”), two linear curves (“Model 2”), and three linear curves (“Model 3”) in Table 4.3. The fade-like detector with two linear curves (“Model 2”) produces the highest recall and precision.

Table 4.3 Effectiveness of fade-like detection models on ten colonoscopy videos.

	Model 1	Model 2	Model 3
<i>Relevant</i>	80	101	92
<i>Irrelevant</i>	8	6	15
<i>Missed</i>	50	29	38
<i>Precision</i>	0.91	0.94	0.86
<i>Recall</i>	0.62	0.78	0.71

Table 4.4 Precision and recall of three scene segmentation algorithms.

<i>ID</i>	<i>Length</i> (min)	<i>Precision</i>			<i>Recall</i>			<i>Time (sec.)</i>		
		A	C	U	A	C	U	C	U	$\frac{C}{U}$
03001	18:26	0.89	0.90	0.91	0.62	0.69	0.77	2597	7150	0.36
03007	25:08	0.90	0.91	0.91	0.69	0.77	0.77	3600	10588	0.34
03009	37:22	0.91	0.85	0.85	0.77	0.85	0.85	5310	13973	0.38
03010	34:24	1.00	1.00	1.00	0.85	0.85	0.85	4905	14420	0.34
03014	36:33	0.91	0.77	0.85	0.77	0.77	0.85	5199	14442	0.36
03015	23:00	1.00	1.00	1.00	1.00	1.00	1.00	3317	8965	0.37
03017	21:14	0.90	0.90	0.90	0.69	0.69	0.69	3029	8413	0.36
03019	24:05	0.90	0.92	1.00	0.69	0.85	0.92	3466	9903	0.35
03020	13:07	1.00	1.00	1.00	1.00	1.00	1.00	1800	4800	0.38
03047	28:29	0.90	0.82	0.82	0.69	0.69	0.69	4037	11214	0.37
03062	30:34	0.83	0.77	0.77	0.77	0.77	0.77	4328	11697	0.37
03133	33:02	1.00	1.00	1.00	0.85	1.00	1.00	4762	12806	0.37
03148	24:28	1.00	1.00	1.00	0.92	0.92	0.92	3460	9582	0.36
03152	11:55	1.00	1.00	1.00	0.92	1.00	1.00	1587	4376	0.36
03163	19:34	1.00	1.00	1.00	0.92	1.00	1.00	2742	7374	0.37
03177	21:29	0.89	0.89	0.89	0.62	0.62	0.62	3031	8156	0.37
03179	29:15	1.00	1.00	1.00	0.69	0.77	0.77	4184	11252	0.37
03185	21:34	1.00	1.00	1.00	0.92	0.92	0.92	3049	8168	0.37
03190	27:07	0.91	0.92	1.00	0.77	0.92	1.00	3896	10477	0.37
03197	14:54	1.00	1.00	1.00	0.85	0.85	0.85	2020	5437	0.37
Average	24:44	0.95	0.93	0.95	0.81	0.85	0.86	3516	9560	0.36

Given the best parameter values, we compare the performance of our audio-based scene segmentation (denoted as “A”), our model approach (denoted as “C”), and our model approach using features derived from pixel intensities of uncompressed videos (denoted as “U”). Table 4.4 shows that our model-based approach, both in compressed domain and uncompressed domain, outperforms the audio-based technique. Our method in uncompressed domain performs slightly better than the one in compressed domain. This is because it can better detect the boundaries of the terminal ileum scene. Hence, a hybrid approach that uses our method in uncompressed domain for detecting boundaries of the terminal ileum scene and our method in compressed domain for other scenes should give the best result. The average processing time using our approach in compressed domain is only about a third of the time taken using our approach in uncompressed domain on the same machine.

## 4.5 Summary

We have described our proposed scene segmentation techniques in this chapter. This new approach employs both audio and visual analysis techniques for parsing colonoscopy videos. Our approach is based on our new definition of semantic units (colonoscopic scenes). For audio analysis, new algorithm of audio frame classification and novel usage of finite state automata, combined with speech recognition techniques, produce satisfactory scene boundaries. A new visual model that encodes the domain knowledge is employed to refine the scene boundary results from audio analysis.

## CHAPTER 5. OPERATION SHOT DETECTION

In this chapter, we introduce our image/video analysis techniques for operation shot detection. We first define a new type of semantic unit called *operation shot*, and discuss the challenges of operation shot detection in Section 5.1. In Section 5.2, we present new techniques to detect operation shots based on the detection of the cables of the diagnostic or therapeutic instruments. We evaluate the effectiveness of the proposed techniques on colonoscopy videos in Section 5.3.

### 5.1 Challenges of Operation Shot Detection

An operation shot is a segment of visual and audio data that correspond to a diagnostic or therapeutic operation in a colonoscopy video. We map the problem of detecting operation shots to the problem of identifying instruments used in diagnostic or therapeutic operations since the operations cannot be performed without these instruments. Given a variety of instruments, we further map the problem of detecting instruments to the problem of detecting the cables of the instruments as the cable is frequently presented in an operation regardless of the types of the instruments. The remaining difficulties are as follows. First, the cables come in different directions, colors, and sizes. Second, the cable appears very bright in many frames; this is related to light required to illuminate the colon. The light beam exits the endoscope tip directly adjacent to the instrument channel opening. It causes any cable exiting this channel to be exposed to undispersed light at maximal intensity which may result in over-exposure of the camera's CCD chip. The same holds true for colon mucosa and contents that are in immediate proximity of the endoscope tip. The intense brightness and resulting over-exposure may mask the actual color information of the cable and adjacent colon wall, making it difficult

to utilize color features for operation shot detection. Last, the appearance of the cable in a frame varies from one frame to another during an operation. Depending on the location in the colon, the space between the endoscope tip and the lesion, and the position of the lesion within the colon, one may see only the head of the instrument (without the cable) or the head of the instrument with a segment of the cable.

## 5.2 Spatio-temporal Operation Shot Detection

We propose a new spatio-temporal segmentation approach for operation shot detection. Figure 5.1 depicts an overview of our algorithm. The first five steps (A-E in the figure) together identify the presence of the cable in each of the images extracted from the input colonoscopy video. The first step is image preprocessing. In this step, each selected image is first enhanced by our new light reflection filtering algorithm. The enhanced image is then segmented into a number of regions. Next, we identify the insertion direction of an instrument. This is useful for removing irrelevant regions (i.e., regions that are not part of the cable) in the region filtering step. To remove the case that the instrument is falsely segmented into several regions, we use the region merging step to combine these regions into one potential cable region. Next, the region matching step matches the candidate regions in the image with the pre-defined template of the cables. We use the terms cable image and non-cable image to refer to an image with the cable and without the cable, respectively. The region matching step outputs a 1 when the image has at least one region sufficiently similar to the cable templates. Otherwise, the image is considered a non-cable image and the region matching step outputs a 0. Based on temporal information, the shot segmentation step utilizes our pre-defined rules to determine the boundaries of operation shots given a series of binary numbers from the region matching step. The details of each step of our algorithm are discussed below.

### 5.2.1 Image Preprocessing

This step includes four stages. Figure 5.2 shows the image examples in each stage. In the first stage, we extract  $t$  image(s) per second to reduce the analysis time for the subsequent

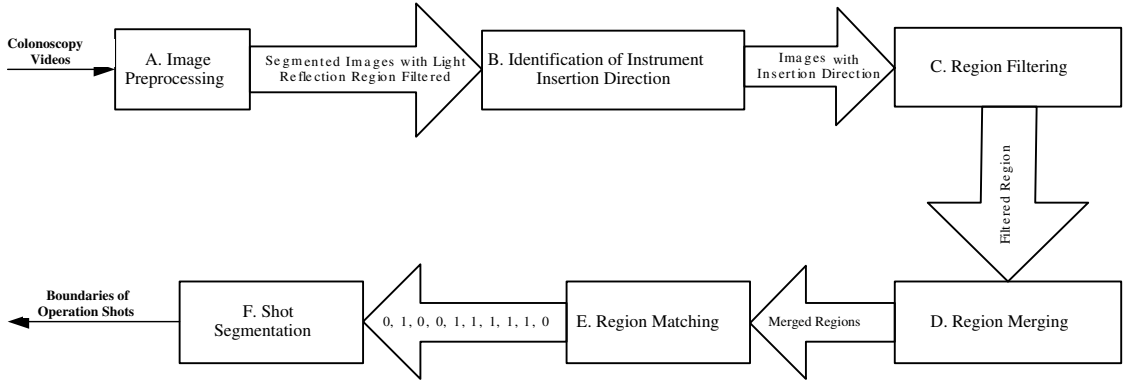


Figure 5.1 Overview of operation shot detection.

steps. Figure 5.2(a) is an example of the selected image. The second stage is called light reflection region filtering. In Figure 5.2(a), we can find many small over-bright white areas in the right part of the image, manually annotated with ellipse. These light reflected regions are generated due to foreign substances (i.e., stool, cleansing agent, air bubbles, etc.) covering the colon wall. They may considerably disturb the subsequent image processing techniques such as edge detection, texture analysis, and segmentation. We include our new light reflection filtering as the second stage of the image preprocessing to address this problem. We observe that many light-reflected areas are small. The majority of the pixels inside a light-reflected area can be identified as edge pixels by commonly used edge detectors. Based on these observations, we develop the following filtering procedure.

- Step 1: Using Sobel edge detector and the morphology closing operation with a flat, disk-shaped structuring element (Sonka et al., 2000) to extract the edge pixels from each image. This step generates a binary image where the white curvilinear structures represent the real edges and the small isolated white regions represent small over-bright areas in the original image, respectively.
- Step 2: Using a predefined  $w \times w$  sliding window to scan the entire image. If we find more than 85% of the pixels inside the window are edge pixels and more than 90% of the pixels in the boundary of the window are not edge pixels, we claim the area delineated

by the window is a real over-bright white area we desired. The percentage thresholds (85% and 90%) are derived from experiments on different colonoscopy videos. Our image enhancement technique is not very sensitive to these thresholds since the results do not vary much when we performed experiments with different threshold values between 80% and 95%.

- Step 3: Using the method in step 2, if we find the area covered by the sliding window is a real over-bright white area, we calculate the average pixel intensity  $I_{ave}$  for all the pixels in the boundary of this window. Update the image by changing the intensity value of the pixels inside the over-bright white area into value  $I_{ave}$ .

The generated image is illustrated in Figure 5.2(b). We can see the majority of the over-bright white areas circled by the red ellipse have been removed. Next, each enhanced image in the reduced colonoscopy video is segmented into a number of regions using JSEG (Deng and Manjunath, 2001). Figure 5.2(c) illustrates the segmentation results.

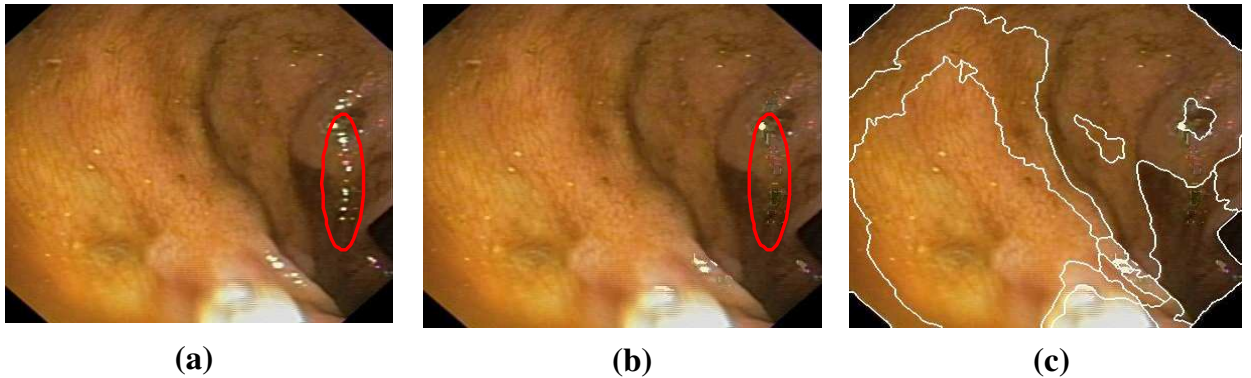


Figure 5.2 Image examples for image preprocessing step: (a) Original color image; (b) Image after removing the light reflected regions; (c) Segmented image using JSEG.

### 5.2.2 Identification of Instrument Insertion Direction

This step identifies the insertion direction of instruments. Only one endoscope is used per colonoscopic procedure and standard colonoscopy models have only one working channel,



in which instruments can be inserted. The insertion direction is determined by the location of the working channel in relation to the camera lens (see Figure 5.3(a)). Therefore, each colonoscopy video has one insertion direction. The instrument can appear in the field of view of the endoscope in any direction, depending on the model of the endoscope used in the procedure. We classify these directions into eight general directions as shown in Figure 5.3(b) and associate insertion direction  $i$  with a triangular “area  $i$ ” where  $1 \leq i \leq 8$  as shown in Figure 5.3(c). The ability to identify the correct triangular area can greatly improve the accuracy and decrease the processing time of subsequent steps.

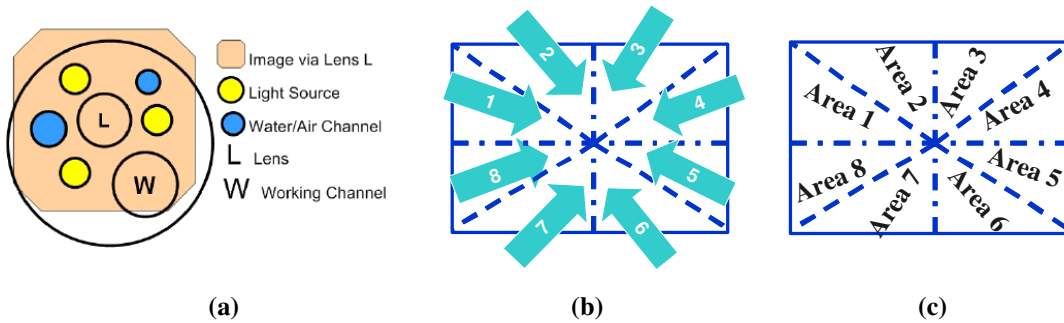


Figure 5.3 Possible triangular areas and insertion directions of instruments: (a) Various components of the tip of a current endoscope model projected on top of the image area; note the position of the working channel in relation to the lens; (b) Eight insertion directions that correspond to eight triangular areas; (c) Eight triangular areas that correspond to eight insertion directions.

We propose an algorithm to identify the insertion direction of the instrument for a video. The cable of the instrument has a tubular shape. The tubular shape has a strong curvilinear structure at the proximal end (most central in the image) with linear line shape of the longitudinal edges of the cable. If we can find this kind of shape in one of the eight triangular areas (for example, “Area 6”) and the orientation of the shape is close to the insertion direction of that triangular area (for example, the angle between the orientation of the object and the insertion direction 6 of triangular area 6 is very small), it is very likely that the insertion direction of this image is the same as the orientation of the shape. For each video, we perform the following algorithm which has two phases to identify the insertion direction.

• **Phase 1: Identification of the insertion direction of instruments for each clear image  $I$**

A Calculate the 2-D line filter using the Hessian Matrix (Sato et al., 1998). The Hessian Matrix of a pixel  $X$  of 2-D image  $I(X)$  (where  $X = (x, y)$ ) is given by

$$\begin{aligned} \nabla^2 I(X) = & \begin{bmatrix} I_{xx}(X) & I_{xy}(X) \\ I_{yx}(X) & I_{yy}(X) \end{bmatrix} \end{aligned} \quad (5.1)$$

where partial second derivatives of the image pixel  $I(X)$  are represented by expressions like  $I_{xx}(X) = \frac{\delta^2}{\delta x^2} I(X)$ ,  $I_{yx}(X) = \frac{\delta^2}{\delta y \delta x} I(X)$  and so on. Let the eigenvalues of  $\nabla^2 I(X)$  be  $\lambda_1(X)$  and  $\lambda_2(X)$  ( $|\lambda_1(X)| > |\lambda_2(X)|$ ), and their corresponding eigenvectors be  $e_1(X)$  and  $e_2(X)$  respectively. The eigenvector  $e_1(x)$ , corresponding to the largest eigenvalue  $\lambda_1(X)$ , represents the direction along which the second derivative is the maximum.

B Generate a binary image  $I_B(X')$  and initialize all the pixel value as 0. For any pixel  $X'$  (where  $X' = (x, y)$ ) in the binary image  $I_B(x')$ , check the corresponding eigenvalue  $\lambda_1(X)$  of pixel  $X$  (where  $X = (x, y)$ ) in the original image  $I(X)$ . If the absolute value of  $\lambda_1(X)$  is larger than a predefined threshold value  $TH_\lambda$ , we treat the pixel  $X'$  as an edge pixel and set the value as 1.

C Perform a hierarchy clustering algorithm (Sonka et al., 2000) on the edge pixels of  $I_B$  and remove the clusters with a small number of pixels. For each cluster  $C_{cluster}$  whose number of pixels is large enough, we extract its skeleton  $C_{skeleton}$  and check the average curvature along  $C_{skeleton}$ . If the curvature is below some predefined threshold, it means the corresponding cluster  $C_{cluster}$  can be approximated as a linear line and it is a possible candidate for the boundary of the tubular shape object. Otherwise, we remove the cluster from the image. We name the cluster with linear line shape as  $C_{Linear}$ .

D For each possible insertion direction  $i$  ( $1 \leq i \leq 8$ ) in Figure 5.3, check the corresponding triangular area  $Area_i$  ( $1 \leq i \leq 8$ ). Specifically, we first check each  $C_{Linear}$  cluster in the entire image and select the cluster  $C_{LinearInAreaI}$  where more than 90% of the pixels belong to the  $Area_i$ . Then, for all the  $C_{LinearInAreaI}$  clusters in  $Area_i$ , we choose the cluster  $C_{MaxLinear}$  whose number of pixels is the largest among all the clusters  $C_{Linear}$ . Check the orientation of the  $C_{MaxLinear}$ . If the angle between the orientation of  $C_{MaxLinear}$  and the insertion direction  $i$  is less than  $22.5^\circ$  ( $90^\circ/4$ ), we claim  $i$  is a possible insertion direction. If there are multiple insertion direction candidates, for example  $i_1$ , whose corresponding cluster  $C_{MaxLinear}^{i_1}$  is in the triangular area  $i_1$ , and  $i_2$ , whose corresponding cluster  $C_{MaxLinear}^{i_2}$  is in the triangular area  $i_2$ , we set the insertion direction as  $i_1$  if the number of edge pixels in the corresponding cluster  $C_{MaxLinear}^{i_1}$  is more than  $C_{MaxLinear}^{i_2}$ . If we could not find any linear line shape cluster whose orientation is close to the insertion direction in any triangular areas, we do not consider this image.

• **Phase 2: Identification of insertion direction of instruments for the entire video:**

- A For each insertion direction  $i$  (where  $1 \leq i \leq 8$ ), calculate a value  $P_i$ , where  $1 \leq i \leq 8$ .  $P_i$  refers to the number of images determined as insertion direction  $i$  over the total number of images in this video.
- B Compare the eight value  $P_i$  ( $1 \leq i \leq 8$ ) and select  $j$ , where  $j = \underbrace{argmax}_{1 \leq i \leq 8}(P_i)$ , as the final insertion direction for this video.

Figure 5.4 shows two example images for this procedure. Figure 5.4(a) is the original input image and Figure 5.4(b) is the image after edge enhancement and clustering. In Figure 5.4(b), there is a cluster with small curvature change in the triangle area 6, which is circled by a triangular. The angle between the orientation of this cluster and the insertion direction 6 is less than  $22.5^\circ$ , which means the insertion direction of this image is 6.

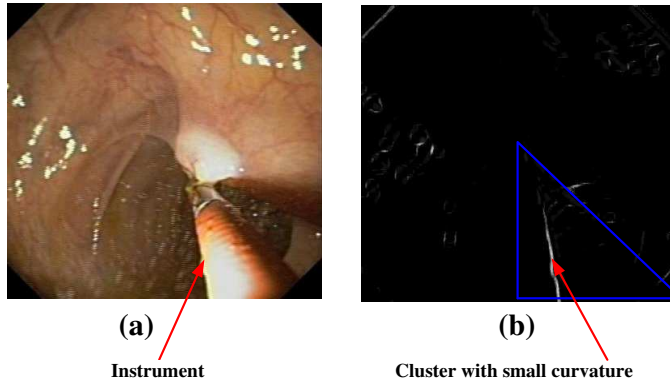


Figure 5.4 Image examples for insertion direction determination.

### 5.2.3 Region Filtering, Merging, and Matching

**Region Filtering** This step removes regions outside the triangular area of the corresponding insertion direction detected in the pervious step. We do this because the other regions are irrelevant to the cable of the instrument. Recall that in the image preprocessing step, an image is enhanced and segmented by using JSEG. Even with our careful selection of important parameters for JSEG, a segmented image still consists of roughly 30 regions on average and over 50 regions in extreme cases. These cases are caused by (1) various degrees of light reflection from the colon wall and (2) complex colon structure in some parts of the colon. Obviously, not all detected regions are part of the cable and therefore should be excluded. The following description of the filtering algorithm assumes that the detected instrument insertion direction is “Direction 6”. By examining a set of segmented cable images from several colonoscopy videos, we find that all the centroids of the desired regions (cable regions) fall in the triangular area shown in Figure 5.5(a). To remove irrelevant regions in our colonoscopy videos, we design the triangular filter as follows. Let  $w$  and  $h$  be the width and the height of the image in pixels, respectively. Given that the top-left corner of the image has the origin coordinate  $(0,0)$ , the filter  $F$  is a triangle with three vertices:  $f1 = (w/2, h/2)$ ,  $f2 = (w/2, h)$ , and  $f3 = (w, h)$ . Let  $R$  be a set of regions of a segmented image after the preprocessing step and let  $r.centroid$  represent the centroid of region  $r$ . The region filtering step identifies the result set  $C$

where

$$C = \{r | r \in R \wedge r.centroid \in F\} \quad (5.2)$$

In order to accommodate instrument detection in cases where the instrument appears in a different position in the field of view (e.g., different type or brand of endoscope), we define eight triangular filters as shown in Figure 5.5(b). Based on the triangular filter, we remove all the regions in which the centroid of the region falls outside the filter.

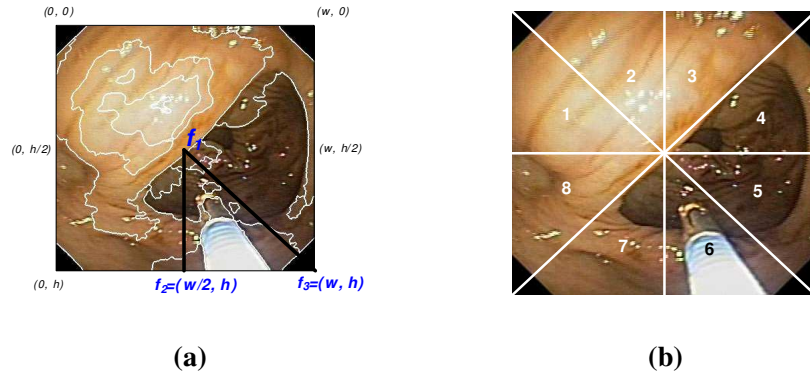


Figure 5.5 (a) The triangular filter in area 6; (b) Eight triangular filters.

**Region Merging** This step identifies the possible instrument regions calculated from the set of candidate regions from the region filtering step. Region merging is important since a whole instrument is often segmented into several regions. Our previous method (Cao et al., 2004a) generated all the combinations of regions if this combination contains at least one bottom region (a region with its smallest bounding rectangle touches the bottom of the image.). This method did not miss any combination of regions that represents the true instrument. However, it also generated many redundant region combinations. To overcome this problem without sacrificing performance, we propose a new region growing algorithm based on texture features as follows.

- **Feature Extraction:** Extract four texture features (Sonka et al., 2000) (Standard Deviation, Smoothness, Uniform, Entropy) for each region in the entire image.

- **Region Clustering:** Apply  $K$ -means clustering method for this image to classify each region into three categories: (a) Smooth region; (b) Periodic region; and (c) Coarse region.
- **Region Growing:** This step treats a bottom region as a seed region. For each seed, we perform region growing by adding a neighbor region if both of them belong to the same category.

We refer to the set of the combined regions after the merging step as  $Q$ . It is composed of merged regions used for the next stage.

**Region Matching** This step matches each of the regions in  $Q$  with a manually defined template set of the cable regions. The template set represents the different representative shapes of the cable found commonly in our colonoscopy videos. We manually selected representative cable images and extracted the corresponding cable regions. Instead of using Fourier descriptors as in (Cao et al., 2004a), we use moment invariants (Sonka et al., 2000) as our shape features. These features are not sensitive to linear transformation, making it suitable to handle different insertion directions of the instrument.

Let  $shape(q)$  return seven moment invariant features of region  $q$ . Let  $S = \{S_1, S_2, \dots, S_K\}$  be a set of  $K$  feature vectors where  $S_i = shape(i)$  for the template region  $i$ . Let  $dist(i, j)$  return the “city-block” distance between feature vectors  $i$  and  $j$  (Sonka et al., 2000). Given a similarity threshold  $d$ , the region matching step decides whether image  $I$  is a cable image or not as follows.

*$I$  is a cable image if there exists an  $s$  such that  $dist(s, Shape(q)) \leq d$  where  $s \in S \wedge q \in Q$*

In other words, the image is declared as a cable image if the dissimilarity between one of its regions in  $Q$  and one of the template regions is less than the threshold. Otherwise, the image is considered a non-cable image. The appropriate value of the threshold  $d$  is found to be 0.025 from experiments. The region matching step outputs a 1 for each detected cable image and a 0 otherwise. The “city-block” distance is used for similarity measure since it has been reported to perform slightly better than other distance metrics

for shape matching in (Eakins et al., 2003). Since the number of distinct cable shapes is small (about ten shapes), these shape feature vectors are loaded in memory once, and then are used during the matching process for the entire video.

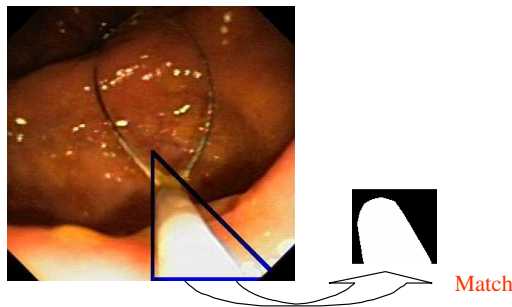


Figure 5.6 Example of region matching between an instrument image and a template region.

#### 5.2.4 Shot Segmentation

This step utilizes temporal information and domain knowledge to identify operation shots. This step addresses the fact that the appearances of a cable vary in the same operation shot and corrects the errors introduced by steps prior to this step. The shot segmentation step accepts  $L$ , a sequence of 0's and 1's from the region matching step, as an input and locates the boundaries of operation shots as follows.

- Step 1: This step aims to correct the misclassification results due to the region matching step. A misclassification result is the case where the image without a cable is classified as a cable image by the region matching step. We found that the detected cable image when surrounded by several non-cable images is very likely a misclassification. We use this observation to correct a misclassification. We first explain the algorithm when one frame per second ( $t = 1$ ) is used in the image preprocessing step. Let  $L'$  be the output sequence of binary numbers with the same length as the input sequence  $L$ . Starting from the beginning of the input sequence  $L$ , we slide a sliding window  $W$  (covering 5 binary numbers at a time) over the input  $L$  to find the correction pattern  $[0, 0, 1, 0, 0]$

in  $L$ . When such a pattern is found, we correct the misclassification result by changing the middle 1 to 0 in the corresponding position in the output sequence  $L'$ . Except this change, the corresponding position in the output  $L'$  has the same binary number as that in  $L$ . In other words, we have  $[0, 0, 0, 0, 0]$  in  $L'$  when  $[0, 0, 1, 0, 0]$  is under the current sliding window in  $L$ . Next, we slide the window one number to the right and repeat the same process until the end of the input sequence is reached. Note that we elected to use the pattern  $[0, 0, 1, 0, 0]$  since in our experiments this pattern removed errors better than other patterns with more zeros surrounding the middle one. We generalize the correction pattern for different values of  $t$  as a pattern that has  $2 * t$  of zeros followed by  $1 * t$  of one followed by  $2 * t$  of zeros. For instance, when  $t$  is 2 (2 frames per second are used), we use the sliding window of size  $5 * 2$  to search for the correction pattern of  $[0, 0, 0, 0, 1, 1, 0, 0, 0, 0]$ .

- Step 2: We scan  $L'$  from the beginning to the end. We declared an operation shot when we find a sequence  $O$  of consecutive frames in  $L'$  with all of the following properties
  - The sequence  $O$  starts with a 1 and ends with a 1 followed by at least  $8 * t$  consecutive zeros. In other words, the first frame and the last frame in the sequence  $O$  are cable images. A fixed number ( $8 * t$ ) of consecutive non-cable images following the last frame of the sequence  $O$  captures the withdrawal of the instrument quite well. We have experimented with larger or smaller numbers of trailing zeros. However, we found that the effectiveness of shot segmentation degrades with more or less trailing zeros.
  - The number sequence  $O$  has more 1's than 0's. That is, the sequence  $O$  has more cable images than non-cable images. This rule is developed based on the observation that an actual operation shot typically has more frames with a cable present than without one.
  - The sequence  $O$  lasts at least 4 seconds. Based on our experience and consultations with our endoscopist, operation shots typically last more than 4 seconds. Only



random biopsies (e.g., for tissue studies in patients with diarrhea or for dysplasia screening in patients with ulcerative colitis) may result in operation shots shorter than 4 seconds; sometimes these random, blind biopsies are even very difficult to be observed by the human eye. Hence, for the studies presented in this article we do not consider an operation shot shorter than 4 seconds.

### 5.3 Performance Study

This section presents experimental results to illustrate the effectiveness of our proposed techniques on three test data sets: **(1) Video Set I:** for identification of insertion direction of instruments. The accuracy of subsequent steps, such as region filtering and region merging, depends on this step. We used videos generated from multiple endoscope models in different endoscopic procedures, including colonoscopy and esophago-gastro-duodenoscopy (EGD), to determine the effectiveness of our technique. The videos and their properties are listed in Table 5.1. **(2) Image Set:** consists of about 1,000 cable and non-cable images extracted from six colonoscopy videos. We used this set to evaluate the effectiveness of the region filtering, region merging, and region matching steps of the operation shot detection technique. Details are listed in Table 5.2. **(3) Video Set II:** consists of twenty five colonoscopy videos with and without operation shots. This test set was used to evaluate the effectiveness of the operation shot detection technique. Table 5.3 shows the total number of operation shots in each video, the number of operation shots in each category, and the average length of the operation shots in the video. The average length of an operation shot in all test videos with diagnostic and therapeutic operations is about twenty two seconds.

#### 5.3.1 Determining Important Parameters for the Proposed Approach

The first step for operation shot detection is “Image Preprocessing”. In this step, we obtain the reduced colonoscopy video by extracting  $t$  frames per second from the input colonoscopy videos. In the experiments, we chose  $t$  equal to one, which implies that the maximum temporal distance between the actual boundary and the detected boundary due to temporal sampling is

Table 5.1 Characteristics of “Video Set I”.

Video Type	Video ID	Insertion Direction
Colonoscopy Video	010	Direction 6
Colonoscopy Video	015	Direction 6
Colonoscopy Video	019	Direction 6
Colonoscopy Video	024	Direction 6
Colonoscopy Video	044	Direction 6
EGD Video	137	Direction 8
EGD Video	139	Direction 8
EGD Video	001	Direction 8
EGD Video	003	Direction 8

Table 5.2 Characteristics of “Image set”.

Video ID	010	015	017	019	024	044	Total
Cable Image	93	11	25	14	123	163	429
Non-cable Image	149	92	60	21	142	194	658

one second. This temporal distance is considered very small compared to the average length of an operation shot (22 seconds). This distance can be made smaller with a higher value of  $t$ , however, with the expense of a significant increase in the analysis time for operation shot detection. Also in this step, we propose a non-linear filter to remove small over-bright white areas. In this method, we use a  $w \times w$  sliding window to scan the entire image to identify these areas. The value of  $w$  is mainly determined by the resolution of the colon image because the size of the white area is proportional to the size of the image. In our experiments, the resolution of our colon image is  $390 \times 370$  and we set the value  $w$  at 15. Recall that our algorithm to identify insertion direction extracts the strong curvilinear structure in the colon image and checks the orientation of the tubular shape object to determine the final direction. One important parameter of this method is the predefined threshold  $Th_\lambda$  used in the second step of this algorithm. We set this value relatively high in order to remove most false positive images (images that do not have any information about instrument insertion direction, but detected as images that contain the insertion direction). At the same time, because of the high threshold, our method generates more false negative images (images with a cable detected as images without cable insertion direction information). However, this does not affect the

Table 5.3 Characteristics of “Video Set II”.

<b>Video ID</b>	<b>Operation Shots</b>	<b>Forceps</b>	<b>Snare</b>	<b>Balloon</b>	<b>Average Length of Operation Shot (second)</b>
002	1	1	0	0	23.0
009	1	1	0	0	38.0
010	6	3	3	0	26.0
012	6	6	0	0	25.8
014	1	1	0	0	18.0
024	12	12	0	0	27.5
044	9	9	0	0	53.0
047	9	7	2	0	21.9
053	7	7	0	0	34.6
097	3	3	0	0	12.0
102	5	5	0	0	33.6
111	2	2	0	0	12.0
114	6	6	0	0	22.8
116	8	8	0	0	18.3
133	4	4	0	0	13.0
134	5	5	0	0	10.4
148	3	2	1	0	8.7
156	2	2	0	0	16.5
165	2	2	0	0	33.5
168	1	1	0	0	15.0
174	0	0	0	0	0
183	0	0	0	0	0
186	10	2	10	0	20.9
192	10	9	0	1	12.1
202	4	0	3	1	50.8
<b>Total</b>	<b>117</b>	<b>96</b>	<b>20</b>	<b>1</b>	-

final results since our detection algorithm selects the insertion direction  $i$  with the maximal  $D_i$  (where  $1 \leq i \leq 8$ ) value as our final insertion direction. We will discuss this issue in detail in the next section.

### 5.3.2 Effectiveness of Cable Detection

There are four important steps in detecting whether an image is a cable image or not. They are identification of instrument insertion direction, region filtering, region merging, and region matching steps. We quantify the effectiveness of each step as follows.

A Effectiveness of Identification of Instrument Insertion Direction: Table 5.4 illustrates the test results for Video Set I. The column labeled “Total” represents the total number of images extracted from the corresponding video. Recall that for each input image, our algorithm either assigns an insertion direction or skips the image. Each number in column “ $D_i$ ” is the number of images that are detected as images with “Insertion Direction  $i$ ”. We use column “Skip” to indicate the number of images that are detected as images without an insertion direction and are skipped by our algorithm. In the final stage of our method, we compare the eight  $D_i$  (where  $1 \leq i \leq 8$ ) values and select the direction  $i$  with the largest  $D_i$  value as the final insertion direction. Based on this method, values in column  $D_6$  for the first five videos and values in column  $D_8$  for the last four videos are selected. This indicates that the insertion directions for the first five videos and the last four videos are “Direction 6” and “Direction 8”, respectively. Hence, our algorithm gives the correct results for all tested videos.

B Effectiveness of Region Filtering: The purpose of the region filtering step is to remove irrelevant regions from further consideration. We use the image set for performance evaluation. For each image in the image set, we obtain the total number of original regions after image segmentation and the number of result regions—regions left after region filtering. Table 5.5 shows the ratio of the number of the result regions to the number of original regions gathered from selected images of each video. For the cable images, only 18% of the original regions remain. For the non-cable images, only 13% of

Table 5.4 Effectiveness of instrument insertion direction identification.

Video	ID	Insertion Direction	Total	D1	D2	D3	D4	D5	D6	D7	D8	Skip
Colonoscopy	010	Direction 6	1367	25	48	20	32	40	<b>147</b>	4	10	1041
Colonoscopy	015	Direction 6	791	12	15	13	9	8	<b>64</b>	14	9	647
Colonoscopy	019	Direction 6	691	19	15	9	11	10	<b>75</b>	12	7	533
Colonoscopy	024	Direction 6	1625	69	20	70	95	78	<b>382</b>	60	59	792
Colonoscopy	044	Direction 6	1044	36	30	19	78	18	<b>264</b>	20	29	550
EGD	137	Direction 8	283	10	9	10	9	8	15	17	<b>65</b>	143
EGD	139	Direction 8	446	23	22	18	19	17	25	26	<b>132</b>	163
EGD	001	Direction 8	57	0	0	0	0	0	0	1	<b>55</b>	1
EGD	003	Direction 8	28	0	0	0	0	0	0	0	<b>27</b>	1

the original regions remain. More regions are left in the cable images due to the presence of the cable. Although 82% of irrelevant regions are removed, no parts of the actual cable region are removed.

Table 5.5 Effectiveness of region filtering.

Video ID	Cable images	Non-Cable Images
010	0.22	0.15
015	0.23	0.13
017	0.17	0.15
019	0.25	0.06
024	0.17	0.13
044	0.14	0.14
<b>Average</b>	<b>0.18</b>	<b>0.13</b>

C Effectiveness of Region Merging: To quantify the effectiveness of the region merging step, we evaluated the effectiveness of cable detection with and without region merging. Out of the 429 cable images in the image set, we manually identified all cable images whose cable is fragmented into more than one region by JSEG. We performed region matching with and without prior region merging on this sub-set of cable images. Table 5.6 shows the results. Region matching with region merging correctly identifies 96% of the images in the set as cable images. Without region merging, region matching only correctly identifies 69% of the images in the set as cable images. Therefore, region merging step

improves the accuracy for cable detection by 27%. Note that some fragmented cables can be detected even without region merging because the fragment of the cable happens to have a shape similar to that of the entire cable.

Table 5.6 Effectiveness of region merging.

<b>Fragmented images</b>	<b>Cable detection with region merging</b>		<b>Cable detection without region merging</b>	
	<b>Correctly detected</b>	<b>Percentage</b>	<b>Correctly detected</b>	<b>Percentage</b>
107	103	96 %	72	69%

D Effectiveness of Region Matching: The region matching step is the last step for cable detection. The effectiveness of this step is demonstrated via the effectiveness of the entire cable detection. First, given an actual cable image, the cable detection algorithm should indicate that the image is a cable image with a high accuracy. Second, given a non-cable image, the cable detection algorithm should determine that the image is a non-cable image with high accuracy. Table 5.7 shows the results of cable detection on the image set. The average accuracy of 92% for cable images and 95% for non-cable images in Table 5.7 are attributed by the effectiveness of (1) the region filtering step that removes irrelevant regions from the cable images; (2) the region merging step that combines fragmented regions that should have been detected as one region; and (3) the use of moment invariants as shape features for matching the candidate region with the template regions. Nevertheless, the cable detection algorithm still has 8% inaccuracy and we rely on the shot segmentation step to correct these small errors due to the following reasons: (1) The JSEG image segmentation algorithm sometimes merges parts of the colon wall and the cable together, which results in a shape different from the template cable shapes; (2) In a non-cable image, the shapes of one or more regions of the colon and the cable may be similar by chance. This error is inevitable in our detection method. In addition, the region merging step introduces the possibility that a combination of colon regions is similar to one of the template cable regions. However, this case happens rarely.

Table 5.7 Accuracy of cable detection.

Video ID	Cable Images	Non-Cable Images
010	0.96	0.89
015	0.75	0.92
017	0.99	0.92
019	1.00	0.95
024	0.90	0.96
044	0.94	0.95
<b>Average</b>	<b>0.92</b>	<b>0.95</b>

### 5.3.3 Effectiveness of Operation Shot Detection

To evaluate the effectiveness of the entire operation shot detection algorithm, we measured the number of false operation shots, the number of missed operation shots, the true positive fraction, the false positive fraction, and the boundary precision and recall. False operation shots are software detected shots that are not actual operation shots determined manually. A missed operation shot is an actual operation shot for which the software failed to detect both boundaries. Note that if one of the two boundaries of an operation shot is incorrect, the detected operation shot still captures part of the actual operation. In such cases we did not treat the detected operation shot as a false or a missed operation shot, but we quantified it using the following metrics. The True Positive Fraction (TPF) is the ratio of the total number of correctly detected images as part of actual operation shots (true positives) to the total number of images of actual operation shots. High TPF is desirable. The False Positive Fraction (FPF) is defined as the ratio of incorrectly detected images as part of operation shots (false positives) to the total number of images of actual operation shots. Low FPF indicates that a small fraction of a detected operation shot is not part of an actual operation shot. Note that our definition of FPF is different from the traditional FPF that uses the ratio of false positives to real negatives. We chose a different definition for FPF because the number of real negatives in general is much larger than the number of false positives our algorithm produces. Using the traditional definition, we have around 0.006 FPF on our test data set. To quantify the percentage of correctly detected boundaries, we use boundary precision and

recall. Boundary precision is the ratio of the number of correctly detected boundaries to the total number of detected boundaries. Boundary recall is the ratio of the number of correctly detected boundaries to the number of actual boundaries determined manually by humans. High boundary precision and recall are desirable.

We applied our method to 25 colonoscopy videos. Table 5.8 shows that only seven false shots are detected. In our opinion this number is a very small number given that any pair of frames in the videos can form a false operation shot. Averages of true positive and false positive fractions are 94% and 10%, respectively. Table 5.9 gives a more detail results for each video. The majority of the videos have a perfect true positive fraction of 1.0. For some videos, the detected operation shots are shorter than the actual operation shots by a couple of frames in the beginning or the end of an actual operation shot. The false positive fraction is due to the case that some detected shots are false; in addition, some detected shots are slightly longer than the actual operation shots. A boundary recall of 97% is very high. Only 3% of the actual boundaries are missed by the algorithm due to the following reasons. First, intense brightness causes JSEG to combine parts of the cable and colon wall. Second, in rare cases only the head of the biopsy forceps (without the cable) is presented in the video during the starting or the ending of an operation. The head of the forceps remains open for several seconds. Since the shape of the head of the open forceps is different from the shape of the cable, we cannot detect the correct boundary in this case. Note that the cable detection algorithm still declares a forceps head a cable image if the head of the forceps is closed since the closed head shape is very similar to the cable shape. The boundary precision is lower compared with the boundary recall. Our shot detection method introduces  $257-234-7=16$  false boundaries, of which 14 boundaries are due to 7 false shots. All the experiments were conducted on a PC with 3.40 GHz Pentium(R) 4 and 1GB of RAM. The processing time for each video frame once the insertion direction has been identified is about 7 seconds on average, of which 6 seconds are spent by JSEG to perform region segmentation. Better performance can be achieved with the more efficient implementation of JSEG.



Table 5.8 Effectiveness of operation shot detection: Overview results.

Number of false shots	7
Number of missed shots	0
Average true positive fraction	0.94
Average false positive fraction	0.10
Average boundary precision	0.88
Average boundary recall	0.97

Table 5.9 Effectiveness of operation shot detection: Detailed results for each video.

Video ID	# Actual Boundaries	#Detected Boundaries	#Correctly Detected Boundaries	True Positive Fraction	False Positive Fraction	Boundary Precision	Boundary Recall
002	2	4	2	1.00	0.38	0.50	1.00
009	2	4	2	1.00	0.45	0.50	1.00
010	12	14	11	0.71	0.04	0.79	0.92
012	12	12	12	1.00	0.00	1.00	1.00
014	2	2	2	1.00	0.00	1.00	1.00
024	24	26	24	1.00	0.11	0.92	1.00
044	18	20	17	0.72	0.08	0.85	0.94
047	18	19	16	0.86	0.17	0.84	0.89
053	14	15	13	0.82	0.09	0.80	0.93
097	6	8	6	1.00	0.31	0.75	1.00
102	10	10	10	1.00	0.00	1.00	1.00
111	4	4	4	1.00	0.00	1.00	1.00
114	12	14	12	1.00	0.03	0.86	1.00
116	16	16	16	1.00	0.00	1.00	1.00
133	8	8	8	1.00	0.00	1.00	1.00
134	10	10	8	0.85	0.08	0.80	0.80
148	6	7	6	1.00	0.10	0.86	1.00
156	4	6	4	1.00	0.16	0.67	1.00
165	4	6	4	1.00	0.14	0.67	1.00
168	2	2	2	1.00	0.00	1.00	1.00
174	0	0	0	-	-	-	-
183	0	0	0	-	-	-	-
186	20	22	20	1.00	0.17	0.91	1.00
192	20	20	19	0.82	0.00	1.00	1.00
202	8	8	8	1.00	0.00	1.00	1.00
<b>Total</b>	<b>234</b>	<b>257</b>	<b>226</b>	-	-	-	-
<b>Average</b>	-	-	-	<b>0.94</b>	<b>0.10</b>	<b>0.88</b>	<b>0.97</b>

## 5.4 Summary

In this chapter, we have introduced our new spatio-temporal algorithms for detecting the operation shot, which is defined as a segment of visual and audio data that correspond to a diagnostic or therapeutic operation in a colonoscopy video. Instead of detecting the operation shot directly, we converted this problem into the detection of the cable of the instrument used in the operation shot. Experiments on colonoscopy videos have showed the effectiveness of our proposed approach.

## CHAPTER 6. APPENDIX IMAGE CLASSIFICATION

In this chapter, we introduce our image analysis techniques for appendix image classification. We have developed two different approaches to solve this problem. The first technique we proposed is based on new intermediate features extracted from each image, followed with different classifiers for image classification. Our second approach is a new model based approach to capture both the local image parts and global spatial relations among the parts. We first present the challenges of appendix image classification in Section 6.1. Then we introduce the technique details of the two approaches in Section 6.2 and Section 6.3. Experimental results are given in Section 6.4.

### 6.1 Challenges of Appendix Image Classification

we define an image with a closely inspected appendiceal orifice an “appendix image”. Our purpose is to classify images in a colonoscopy video into two categories: appendix image class and non-appendix image class. The appendix image classification problem is very challenging because of the large intra variations in the appearance of the appendiceal orifice caused by the following reasons. First, because the appendiceal orifice may appear in different locations in the images, and may have different sizes, we must handle a wide range of transformation of object translation and scaling, changes of the viewing direction and distance. Secondly, there are large illumination and intensity variations among appendix images. For example, some part of the appendix appears very bright in many frames; this is related to light required to illuminate the colon. The light beam that exits the endoscope tip causes colon mucosa and contents that are in immediate proximity of the endoscope tip to be exposed to undispersed light at maximal intensity which may result in over-exposure of the camera’s CCD chip. The

same holds true for colon mucosa and contents that are in an immediate proximity of the endoscope tip. The intensive brightness and resulting over-exposure may mask the actual color information of the appendix and adjacent colon wall, making it difficult to utilize color features directly for appendix detection. Thirdly, depending on the location in the colon, the space between the endoscope tip and the lesion, and the position of the lesion within the colon, one may see only part of appendiceal orifice.

A direct application of existing object recognition techniques to detect the appendiceal orifice leads to unacceptable results. One of the reasons is the failure of the interesting point detector (or saliency operator). Many of the existing object recognition techniques are heavily relying on the interesting point detector to identify the regions on the object. If the object of interests (appendiceal orifice) always receive insufficient coverage from the detector, the object recognition task may fail no matter how accuracy of other steps (such as feature extraction and feature comparison) are. Figure 6.1 illustrates this case. The left image (Figure 6.1(a)) is the original appendix image and Figure 6.1(b) shows the results of applying the well-known “Scale Invariant Feature Transform” (SIFT) interesting point detector (Lowe, 2004). The location, orientation, and magnitude of each interesting point are annotated by the arrows. The majority of the appendiceal orifice (several elliptical shape muscles in the upper part of the image) is not identified by the feature detector. We also applied the eigenface based method (Turk and Pentland, 1991) to a set of appendix images and used the principle subspace to recognize new image. Due to the large variations of shape, color, and illumination, this method failed to detect the appendiceal orifice in many cases.

## 6.2 Feature-based Appendix Image Detection Approach

This is the first method to solve the appendix image classification problem. It consists of two steps. First, we obtain from each image intermediate features that we introduce in this section. Second, we use classification algorithms to group the images into two groups: appendix image class and non-appendix image class. Three classifiers, K-Means, Decision-Tree, and Support-Vector-Machine (SVM), are employed. The new intermediate features are



Figure 6.1 (a)Original appendix image; (b) Location, orientation, and magnitude of interesting points identified by SIFT feature detector.

derived based on the following two observations: 1) *When the appendix is closely inspected, a distant colon lumen is not visible ("no colon lumen");* 2) *The clearly seen appendix orifice has several curvilinear structures that are part of ellipses. These structures usually are located in the center of the image when the appendix is the focal point of inspection.*

The new features are as follows (1) Likelihood of no colon lumen; (2) Ratio of edge pixels that are part of curvilinear structures; (3) Coverage of the ellipse inside the image; (4) Ratio of edge pixels that are part of ideal ellipses. We describe the derivation of these features in more detail below.

- 1 Feature representing the likelihood of no colon lumen: Based on the first observation, this feature represents the possibility of the presence of distant lumen in the image. We first segment the image using JSEG (Deng and Manjunath, 2001). The segmented image contains multiple regions, as shown in Figure 6.2(b) and Figure 6.2(d). The likelihood of no distant colon lumen is computed as follows.

$$P_{NoLumen} = \frac{I_{DarkestRegion}}{I_{Max}} \quad (6.1)$$

$P_{NoLumen}$  is the likelihood of no distant colon lumen,  $I_{DarkestRegion}$  is the average intensity of the darkest region, and  $I_{Max}$  is the maximal intensity of the image. The darkest regions are pointed out by the arrows in Figure 6.2(b) and Figure 6.2(d). Generally, the value of  $P_{NoLumen}$  for an appendix image is larger than that of the image with distant colon lumen since the average intensity value of the darkest region of appendix image is larger than the value of images with "dark" distant lumen. Figure 6.2 shows the visual difference between the image with distant lumen and the appendix image. Note that the small size regions are removed by filtering out all the regions whose size is less than a pre-defined size threshold.

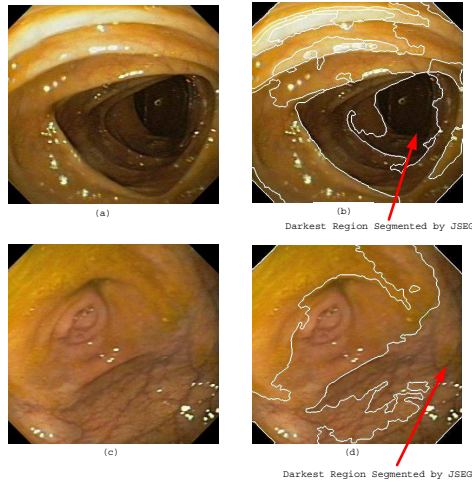


Figure 6.2 Image examples before and after segmentation: (a) Original lumen image; (b) Lumen image after segmentation; (c) Original appendix image; (d) Appendix image after segmentation. The average intensity of the darkest region for the lumen image is much smaller than the one for the appendix image, which is consistent with our observation (1).

2 Feature representing the ratio of edge pixels belonging to curvilinear structure: Based on the second observation, an appendix image has several curvilinear structures. We claim

that the possibility that the image contains the appendiceal orifice is high if the many edge pixels belong to curvilinear structures corresponding to the appendiceal orifice. To get the edge image that includes the curvilinear structures, we employ the same Hessian matrix-based technique used previously (Cao et al., 2007). In Figure 6.3, the binary image (Figure 6.3(b)) contains ellipse-shape curves that are part of the appendiceal orifice. But it also includes other curves, as shown in the bottom right part of Figure 6.3(b). We select the true appendix curve by checking the curvature change along the skeleton of each curve. The skeleton of the curve that is not a real appendiceal orifice has either a small curvature change (curve with linear shape, illustrated as the left curve of the two curves in the right bottom of Figure 6.3(b)) or a very large curvature change (curve with round shape, illustrated as the right curve of the two curves in the right bottom of Figure 6.3(b)).

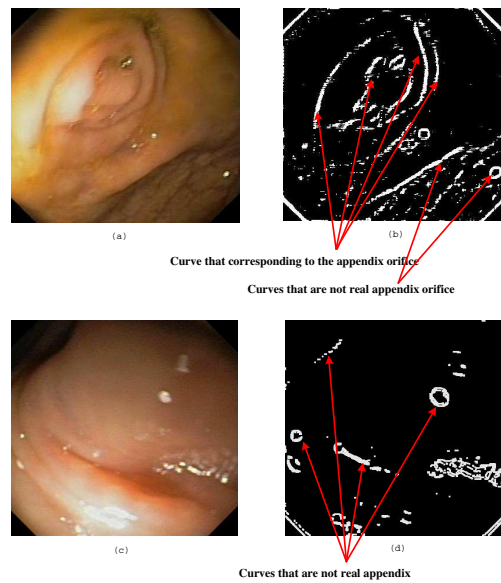


Figure 6.3 Image examples for image enhancement based on Hessian Matrix: (a) Original appendix image; (b) Edge image for image(a) after enhancement; (c) Original image without a clearly seen appendiceal orifice; (e) Edge image for image(b) after enhancement.

After the above preprocessing step, we compute the value of edge pixels in the curvilinear

structures corresponding to the appendiceal orifice over the total number of edge pixels in the binary image.

$$R_{Curve} = \frac{EdgeNum_{Curve}}{EdgeNum_{Img}} \quad (6.2)$$

$R_{Curve}$  indicates the ratio of edge pixels of the appendix curvilinear structures;  $EdgeNum_{Curve}$  is the number of edge pixels that belong to the curvilinear structures of the appendiceal orifice.  $EdgeNum_{Img}$  is the number of edge pixels in the entire image. In Figure 6.3(b), many edge pixels in the binary image (generated from the appendix image) are curves corresponding to the appendiceal orifice. But most edge pixels in Figure 6.3(d) are curves that are not part of the appendiceal orifice. A large value of  $R_{Curve}$  indicates a high probability that the appendiceal orifice is present in the image.

- 3 Features representing partial ellipses in an image: Recall our second observation that a closely inspected appendiceal orifice has several curvilinear structures that are part of ellipses. To derive the two features that reflect this observation, we only consider curves that are potential candidates for the true appendiceal orifice from the computation of the previous feature (e.g., curve  $ACu$  in Figure 6.4). We want to find an ideal ellipse with certain part fitting well with the candidate curve. For example, part of the ideal ellipse  $A$  fits well with curve  $ACu$  as shown in Figure 6.4). We introduce a new modification to the randomized Hough Transform (Xu et al., 1990) abbreviated to MHT. MHT works as follows. Given an edge image with a curve, say  $ACu$ , we expand the image and get the skeleton  $SkeACu$  of the curve. Then, we perform a boundary tracing algorithm on  $SkeACu$  to get a boundary represented by a sequence of points  $P_1, \dots, P_i, P_{i+1}, \dots, P_n$  where  $P_i$  and  $P_{i+1}$  are neighboring edge pixels. Next, we divide the sequence of points into three segments with equal length, illustrated in Figure 6.4. We generate an ideal ellipse as follows. We randomly pick one edge pixel from each segment and compute an ideal ellipse whose boundary passes the three chosen edge pixels in a polar coordinate. The ideal ellipse is represented by an ellipse tuple (x, y coordinates of the ellipse centroid,



the length of the major axis, the length of the minor axis, and the orientation of the major axis). If at least 90% of the edge pixels of the curve are on the boundary of the ideal ellipse, we consider this ellipse as a valid ideal ellipse for the curve and stop the MHT for this curve. Otherwise, we continue to select three random pixels from the three segments and repeat the above procedure. If we can not find any valid ellipse after  $N$  iterations where  $N$  is a predefined threshold, we skip this curve. Compared with other Hough Transform techniques, we do not perform coordinate transform operations. Instead, we randomly pick up three points from each segment to construct an ellipse based on ellipse geometry (Xu et al., 1998). Then we determine the degree of curve fitting by checking the ellipse coverage. This is different from many of existing curve fitting techniques, which attempt to formalize and solve an energy minimization problem. Our experiments have shown that the new techniques are simple yet effective in finding the desired curves.

As shown in Figure 6.5, the ideal ellipse (*Ellipse A* and *Ellipse B*) may or may not be enclosed entirely in the original image. The more curvilinear structures we find in the boundary of the ideal ellipse, the more likely the appendiceal orifice is present in the image.

After this step, we get a number of ideal ellipses. We eliminate any detected ellipse with less than 50% of its area inside the original image. We compute “Ellipse Coverage ( $COV_{Ellipse}$ )” to reflect the amount of the ellipse area inside the original image. Let  $k$  be the number of valid ellipses. Let  $PixelNumInsideOriImgEllipse_i$  be the number of pixels that are both inside the ellipse  $i$  and inside the original image and  $PixelNumInsideEllipse_i$  is the number of pixels inside the ellipse  $i$ , where  $i \in [1, k]$ . Combining this we have the following equation for the ellipse coverage.

$$COV_{Ellipse} = \frac{\sum_{i=1}^k PixelNumInsideOriImgEllipse_i}{\sum_{i=1}^k PixelNumInsideEllipse_i} \quad (6.3)$$

If this value is high, we can find many ideal ellipses with most of their area inside the original image. Hence, the possibility of the presence of an appendix is high.

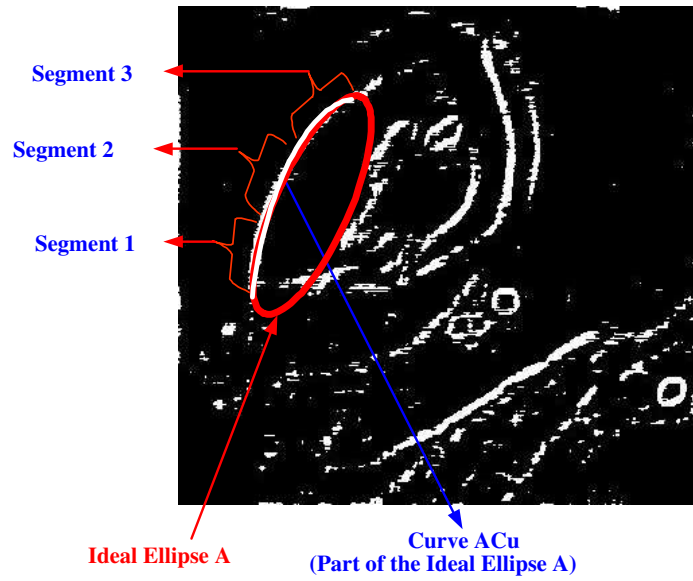


Figure 6.4 Derivation of the ideal ellipse A from curve ACu, which is part of the ideal ellipse A and resides in the boundary of ellipse A.

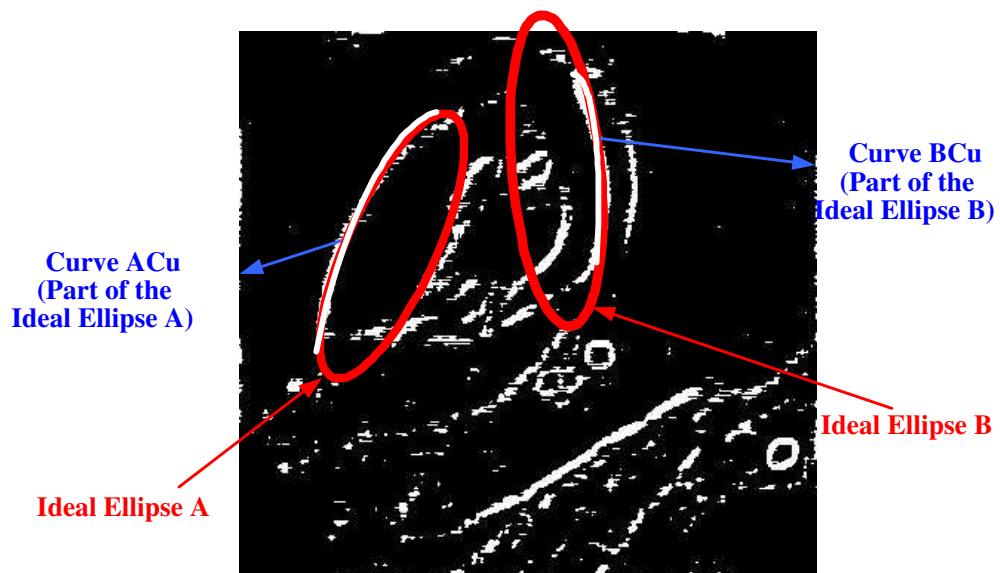


Figure 6.5 Relationship between the appendix curve and the ideal ellipse.

We introduce another feature called ‘‘Ellipse Edge Coverage ( $COV_{EllipseEdge}$ )’’ computed as follows.

$$COV_{EllipseEdge} = \frac{\sum_{i=1}^k EdgeNumOnEllipseBoundary_i}{\sum_{i=1}^k TotalPixelNumOnEllipseBoundary_i} \quad (6.4)$$

where  $EdgeNumOnEllipseBoundary_i$  is the number of pixels on the appendix curve that is on the boundary of ellipse  $i$ , and  $TotalPixelNumOnEllipseBoundary_i$  is the total number of pixels on the boundary of ellipse  $i$ , where  $i \in [1, k]$ . If this value is high, many ideal ellipses are present with the majority of their boundaries covered by the curvilinear structures of the appendiceal orifice. Thus, an appendix image has a high value of  $COV_{EllipseEdge}$ .

For each video, we extract the four intermediate features for each image and apply a classifier. We provide three classifiers: for K-means classifier, Euclidean distance function and equal weight for each feature are used; for the Decision-Tree classifier, we use C4.5 as our classifier; for the SVM classifier, we use the sequential minimal optimization algorithm with polynomial kernels. Any image in this video will be classified as either an appendix image or a non-appendix image. The details of the experimental results will be discussed in Section 6.4.

### 6.3 Model-based Appendix Image Detection Approach

The four intermediate features in the previous section can capture the global visual properties of the object (appendiceal orifice). However, it may not be effective if the intra-variations of the object are large. An alternative approach is to find some parts of the object that the intra-variations of the part are small. If we use these parts to represent the object and convert the problem of object recognition into the problem of parts recognition, we may achieve better performance. In this section, we propose a new model based approach to capture both the local image parts and global spatial relations among the parts. In our model, we decompose the appendiceal orifice into a set of parts that are assembled in a deformable configuration. A Principle Component Analysis (PCA) based appearance model and an N-star graph shape model

are used to capture the appearance of each part and the spatial relations among these parts. In the following sections, we first introduce the structure of the statistical model. Then we present our algorithms for learning the model parameters. Followed by a detailed description for recognizing an instance of the object (appendiceal orifice), given a new colon image.

### 6.3.1 Structure of the Statistical Model

We use a restricted pictorial structural model to encode the appearance and shape information. The pictorial structural model was first proposed by Fischler (M.A.Fischler and R.A.Elschlager, 1973) and improved by Felzenszwalb and Huttenlocher (Felzenszwalb and Huttenlocher, 2005). Motivated by their work, we represent the object by a set of parts  $\{p_0, p_1, \dots, p_P\}$  where the number of parts is  $(P + 1)$ . We define a P-star graph  $G = (V, E)$  where  $V = \{v_0, v_1, \dots, v_P\}$  corresponds to the  $(P + 1)$  parts; an edge  $(v_0, v_j)$  represents a pair of connected parts;  $v_0$  is the root node with vertex degree  $P$  and  $v_j$  ( $j \in [1, P]$ ) represents the leaf node with vertex degree 1. In addition, we define a configuration  $L = (l_0, l_1, l_2, \dots, l_P)$  to represent an instance of the object where  $l_i$  indicates the location of part  $v_i$  ( $i \in [0, P]$ ). An example of the model structure is illustrated in Figure 6.6. The left image shows a colon image that contains an object (appendiceal orifice) and the object is decomposed into four parts. Each of them is annotated by a dot rectangle. The object can be mapped into a 3-star graph. The top right part of the object corresponds to the root node  $v_0$  in the 3-star graph shown on the right. The other three parts are mapped into the three leaves in the graph. After we construct the P-star graph and establish the mapping between object parts and graph nodes, we consider the model parameters. Let  $\theta_f = (A, X)$  be a set of parameters that define a foreground object model where parameters  $A = \{a_0, a_1, a_2, \dots, a_P\}$  represent the appearance of the parts, and parameters  $X = \{x_{0j} | (v_0, v_j) \in E\} (j \in [1, P])$  characterize the spatial relationship between connected parts.

From the statistic point of view, our model can be best explained as follows. Suppose we have already learnt a set of parameters  $\theta_f$  for foreground objects and all non-object background images are modelled by a fixed set of parameters  $\theta_b$ . Given a new image  $I$ , we can determine

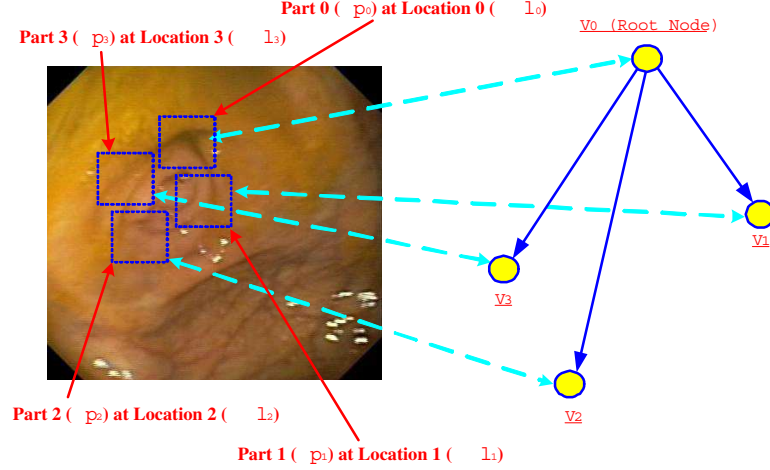


Figure 6.6 Illustration of the mapping between the image and the graph.

whether the image contains an instance of an object by considering posterior ratio  $R$  using Bayes' formulation:

$$R = \frac{p(h_1|I)}{p(h_0|I)} = \frac{p(I|h_1) \cdot p(h_1)}{p(I|h_0) \cdot p(h_0)} \approx \frac{p(I|\theta_f) \cdot p(h_1)}{p(I|\theta_b) \cdot p(h_0)} \quad (6.5)$$

where  $h_1$  represents the hypothesis that  $I$  contains an instance of the object and  $h_0$  represents the hypothesis that  $I$  contains background only. The right most expression in Equation 6.5 is an approximation because we represent the category with the imperfect model (Fergus et al., 2006). To compute the posterior ratio  $R$ , we should obtain the likelihood ratio and prior ratio. The prior ratio may be estimated from training or may be set to a constant value manually. To compute the likelihood ratio, we need to compute two likelihood probabilities:  $p(I|\theta_f)$  and  $p(I|\theta_b)$ . The denominator  $p(I|\theta_b)$  is the likelihood of seeing an image with background parameters. It can be considered as a constant for a given image (Fergus et al., 2006). The nominator  $p(I|\theta_f)$  indicates the likelihood of seeing an image with the foreground parameters. To obtain this value, we should sum over all possible object configurations (Crandall et al., 2005). Using conditional probability principles, we can get the following equation:

$$p(I|\theta_f) = \sum_{k=1}^K p(I|L_k, \theta_f) \cdot p(L_k|\theta_f) \quad (6.6)$$

where  $L_k$  represents a possible configuration and  $K$  is the number of total configurations. We define the score of the configuration  $L_k$  as the probability of this configuration occurs in the image  $I$  with foreground parameters  $\theta_f$ . We term this probability as  $p(L_k|I, \theta_f)$ . In addition, we name the configuration with the highest probability as the best configuration of this image and term the best configuration using symbol “ $L$ ”. In practice, we obtain the best results of  $p(I|\theta_f)$  by only selecting the best configuration instead of summing up all configurations, since usually, the background images contain many low-scoring configurations, which cause false positives (Fergus et al., 2003, 2006). Based on this assumption and Equation 6.6, we get the following formula:

$$p(I|\theta_f) = \sum_{k=1}^K p(I|L_k, \theta_f) \cdot p(L_k|\theta_f) \approx p(I|L, \theta_f) \cdot p(L, \theta_f) \quad (6.7)$$

where  $L$  represents the best configuration. Next we apply the Bayes’ formulation to the probability score of the best configuration  $L$  and get the following formula:

$$p(L|I, \theta_f) \propto p(I|L, \theta_f) \cdot p(L|\theta_f) \quad (6.8)$$

From Equation 6.7 and Equation 6.8, we can convert the problem of computing the likelihood of seeing an image with foreground parameters ( $p(I|\theta_f)$ , which is the left expression in Equation 6.7) into the problem of computing the probability of the best configuration given the image with foreground parameters ( $p(L|I, \theta_f)$ , which is the left expression of Equation 6.8). This conversion simplifies the recognition step: given a new image, we can determine the existence of the object by only considering the probability score of the best configuration.

The first term (likelihood probability  $p(I|L, \theta_f)$ ) in Equation 6.8 represents the likelihood of seeing an image given that the object is at a particular configuration and it only depends on the appearance of the parts. If the parts do not overlap (which is true in our case), we can assume that each part is independent. Hence, the likelihood probability can be factored as follows

$$p(I|L, \theta_f) = p(I|L, A) \propto \prod_{i=0}^P p(I|l_i, a_i) \quad (6.9)$$

where  $l_i$  represents the location of part  $v_i$  and  $a_i$  indicates the appearance parameters. The second term (prior probability  $p(L|\theta_f)$ ) in Equation 6.8 models the prior distribution over object configurations and it only relies on the spatial relations among the connected parts. It can be captured by a tree-structure Markov Random Field with edge set  $E$ , which is equal to the joint distribution for pairs of parts connected by edges divided by the joint distribution of each part (Poggi and Ragozini, 1999; Elia et al., 2003). Since we use a P-star graph to model the relative spatial difference between the root and the leaves, we can simplify the prior distribution with the following equation:

$$p(L|\theta_f) = \prod_{(v_0, v_j) \in E} p(l_0, l_j | c_{0,j}) \quad (6.10)$$

where part  $v_0$  and  $v_j$  are connected pairs;  $v_0$  is root node and  $v_j$  is leaf node.  $l_0$  and  $l_j$  are the locations for part  $v_0$  and  $v_j$ .  $c_{0,j}$  are the parameters for modelling the connection between part  $v_0$  and  $v_j$ . Using Equation 6.9 and Equation 6.10 to replace the two terms in the right expression of Equation 6.8, we can get the following formula

$$p(L|I, \theta_f) \propto \prod_{i=0}^P p(I|l_i, a_i) \cdot \prod_{(v_0, v_j) \in E} p(l_0, l_j | c_{0,j}) \quad (6.11)$$

In order to solve Equation 6.11, we need to obtain the parameters for both the appearance model and shape model by learning from training examples. For recognition purpose, we also need to assign the appearance probability score for each part candidate and the shape probability score for connected parts. In the next sections, we will introduce how we learn the model parameters from training examples and how to recognize an instance of the object for a new image by computing the probability score.

### 6.3.2 Learning

Suppose we have  $N$  training images; each of them contains an instance of the object and we annotate the location of each part in these images. The purpose of learning is to obtain the parameters  $\theta_f = (A, X)$  from the  $N$  training images  $\{I_1, I_2, \dots, I_N\}$  with  $N$  object configurations  $\{L_1, L_2, \dots, L_N\}$ . The object in each image is decomposed into  $P + 1$  parts and

we define the configuration of image  $I_k$  as  $L_k = (L_{k0}, L_{k1}, L_{k2}, \dots, L_{kP}, ) (k \in [1, N])$ . We solve this problem using maximal a posterior (MAP) method. We first define a function of  $\theta_f$  as below

$$f(\theta_f) = p(I_1, L_1, I_2, L_2, \dots, I_N, L_N | \theta_f) = \prod_{k=1}^N p(I_k, L_k | \theta_f) \quad (6.12)$$

This is the likelihood function of  $\theta_f$  with respect to a set of independent samples  $\{ \langle I_1, L_1 \rangle, \langle I_2, L_2 \rangle, \dots, \langle I_N, L_N \rangle \}$ . For the single term of the right expression in Equation 6.12, we obtain the following formula using conditional probability:

$$p(I_k, L_k | \theta_f) = p(I_k | L_k, \theta_f) \cdot p(L_k | \theta_f) \quad (6.13)$$

Based on Equations 6.12 and 6.13, we derive the maximal likelihood estimate for  $\theta_f$  as follows

$$\theta_f^* = \underbrace{\operatorname{argmax}_{\theta_f} p(I_1, I_2, \dots, I_N, L_1, L_2, \dots, L_N | \theta_f)}_{\theta_f} = \underbrace{\operatorname{argmax}_{\theta_f} \prod_{k=1}^N p(I_k | L_k, \theta_f)}_{\theta_f} \cdot \prod_{k=1}^N p(L_k, \theta_f) \quad (6.14)$$

The first part  $\prod_{k=1}^N p(I_k | L_k, \theta_f)$  of the last expression in Equation 6.14 relies only on the appearance, and the second part  $\prod_{k=1}^N p(L_k | \theta_f)$  depends only on the spatial relations among parts. We can independently obtain the parameters for the appearance model and for the shape model by two algorithms introduced in the next two sections. The input for these algorithms is a set of independent examples which is termed as “*IndependentSet*”. We define  $IndependentSet = \{S_0, S_1, \dots, S_P\}$  where  $S_i = \{ \langle I_1, L_{1i} \rangle, \langle I_2, L_{2i} \rangle, \dots, \langle I_N, L_{Ni} \rangle \}$  ( $i \in [0, P]$ ), and  $L_{ki}$  ( $k \in [1, N]$  and  $i \in [0, P]$ ) represents the location of part  $v_i$  of training image  $I_k$  ( $k \in [1, N]$ ).

### 6.3.2.1 Learning Parameters for Appearance Model

The purpose for this part is to estimate the parameters  $A$  for the appearance model. Since the first part of the last expression in Equation 6.14 is only related to the appearance of the



part, we can get

$$A^* = \underbrace{\operatorname{argmax}}_A \prod_{k=1}^N p(I_k | L_k, A) \quad (6.15)$$

Recall that Equation 6.9 gives the formula for computing the likelihood of seeing an image  $I_k$ , given that the image has a specific configuration  $L_k$  where  $L_k = (L_{k0}, L_{k1}, L_{k2}, \dots, L_{kP})$  and  $L_{ki}$  represents the location of part  $v_i$  ( $i \in [0, P]$ ). Combine Equation 6.9 and 6.15, we have the following maximal likelihood estimation.

$$A^* = \underbrace{\operatorname{argmax}}_A \prod_{k=1}^N \prod_{i=0}^P p(I_k | L_{ki}, a_i) \quad (6.16)$$

This is the maximal likelihood estimation of the appearance parameters. As we mentioned before, each part is assumed to be independent. Hence, we can solve Equation 6.16 by computing  $a_i$  independently as below

$$a_i^* = \underbrace{\operatorname{argmax}}_A \prod_{k=1}^N p(I_k | L_{ki}, a_i) \quad (6.17)$$

Hinted by the previous work in (Turk and Pentland, 1991; Moghaddam and Pentland, 1997), we introduce a new appearance model based on Principal Component Analysis (PCA). We first construct a principal subspace (also called "feature space" in (Turk and Pentland, 1991; Moghaddam and Pentland, 1997)). Then we compute the reconstruction error of the eigenspace decomposition (referred as "residual" or "Distance From Feature Space" in the work of (Turk and Pentland, 1991; Moghaddam and Pentland, 1997)) for each training example. In (Turk and Pentland, 1991), the authors use the reconstruction error as an indicator of image similarity directly. Unlike their work, we model the reconstruction error of the training data using density estimation techniques. In (Moghaddam and Pentland, 1997), the authors use two types of density estimation (a multivariate Gaussian and a Mixture-of-Gaussian) to estimate the complete probability distribution of the object's appearance. The target density is composed of two parts: the density in the principal subspace (referred as "Distance In Feature Space"), and the density in the orthogonal component of the principal subspace (referred as "Distance

From Feature Space” or “reconstruction error”). In our method, we model appearance of the part as a single Gaussian distribution and the problem of getting the parameter  $a_i$  is converted into the problem of deriving the values of mean and variance for the Gaussian distribution. The reason is because the transformation and view angle change are small for each part although the object in a whole has large variations. Hence, we can accurately model the samples using a single Gaussian distribution. For the same reason, we only apply the density estimation method to the reconstruction error (“residual” or “Distance From Feature Space”). For each part  $v_i (i \in [0, P])$ , we perform the following two stages’ algorithm to obtain the appearance parameter  $a_i$ . In the description of the two stages’ algorithm, the symbol “ $i$ ” is a fixed value and it represents the part number. The input of the algorithm is a set of independent examples “*IndependentSet*” where  $IndependentSet = \{S_0, S_1, \dots, S_P\}$  and  $S_j = \{ \langle I_1, L_{1j} \rangle, \langle I_2, L_{2j} \rangle, \dots, \langle I_N, L_{Nj} \rangle \} (j \in [0, P])$ .

- *Stage 1:* In this stage, we construct the principal subspace for part  $v_i$  by eigenvector extraction. Since our purpose in this stage is to obtain the principal subspace for part  $v_i$ , we only need to perform the eigenvector extraction to element  $S_i$  instead of all elements of “*IndependentSet*”. Specifically, we use the following steps to get the principal subspace:
  - Step 1: Based on the location  $L_{ki}$  of part  $v_i$  in each image  $L_k$  of  $S_i$ , we crop a sub image  $Sub_k$  with size  $m \times n$ . Then we transform  $Sub_k$  into a column vector with size  $mn \times 1$  and generate a sequence of training images  $S = SI_1, SI_2, \dots, SI_N$  for part  $v_i$ . How to set the appropriate values for  $m$  and  $n$  will be discussed in Section 6.4.
  - Step 2: Obtain the mean image  $M = \frac{1}{N} \sum_{k=1}^N SI_k$  and a set of difference images  $D = \{D_1, D_2, \dots, D_N\}$  where  $D_k = SI_k - M$ .
  - Step 3: Construct the covariance matrix  $CM$  from the set of difference images  $D$  (obtained in step 2):  $CM = A * A^T$  where  $A = \{D_1, D_2, \dots, D_N\} (k \in [1, N])$ , and  $*$  represents a matrix multiplication.
  - Step 4: Compute the eigenvalues and eigenvectors from  $CM$  (obtained in step 3).

The eigenvectors capture the source of the variance for part  $v_i$ . We only select  $M$  largest-eigenvalue eigenvectors  $\{\mu_1, \mu_2, \dots, \mu_M\}$ . Let  $\phi_i = \{\mu_1, \mu_2, \dots, \mu_M\}$  and this vector is called the principal subspace (or feature space) that contains the principal components of part  $v_i$ . How to set the appropriate values for  $M$  will be discussed in Section 6.4

- Step 5: Reconstruct each  $SI_k$  based on the linear combination of the  $M$  largest-eigenvalue eigenvectors from step 4. For each  $SI_k$ , we can get a sequence of reconstruction weights (coefficients) which is termed as “ $Wei_k$ ”. We define  $Wei_k = \{\omega_1, \omega_2, \dots, \omega_M\}$  where  $\omega_l$  is the coefficient for eigenvector  $\mu_l$  during the reconstruction and it is computed as  $\omega_l = \mu_l^T \cdot (SI_k - M)$  ( $l \in [1, M]$ ). Since we have  $N$  sub images, we can get  $N$  sequences of reconstruction weights, which form a set denoted as  $WeightSet4PartV_i = \{Wei_1, Wei_2, \dots, Wei_N\}$ .
- Stage 2: The second stage includes distance calculation and parameter estimation. For this stage, we use all elements from the “*IndependentSet*”. We project all the sub images from each element  $S_j (j \in [0, P])$  of “*IndependentSet*” to the principal subspace  $\phi_i$  of part  $v_i$  obtained from the step 4 of the first stage. The distance from the feature space for each sub image is computed as follows. Similar to the first step of Stage 1, for each element  $S_j (j \in [0, P])$ , we can get  $N$  instances (sub images) by cropping from the original image  $I_k (k \in [1, N])$  based on location  $L_{kj}$  of part  $v_j (k \in [1, N] \text{ and } j \in [0, P])$ . Hence, we have a total of  $N \cdot (P + 1)$  sub images. For each sub image, we reconstruct it with the linear combination of the eigenvectors obtained in step 4 of the first stage and get a sequence of reconstruction weights. Then we compute the Euclidean distance between the new reconstruction weights with each element from the “*WeightSet4PartV\_i*” obtained in step 5 of the first stage. The minimal Euclidean distance is selected. This distance determines the degree of similarity between the newly cropped sub image and the sub images of part  $v_i$  obtained in step 1 of Stage 1: the smaller the distance, the more likely the image belongs to part  $v_i$ . Altogether, there are  $N \cdot (P + 1)$  distance values since the number of sub images is  $N \cdot (P + 1)$ . These values form  $DistanceSet = \{D_0, D_1, \dots, D_P\}$ ,

$D_j = \{d_{1j}, d_{2j}, \dots, d_{Nj}\}$ , and  $d_{kj}$  ( $k \in [1, N]$ ) indicates the minimal distance value of the sub image cropped from location  $L_{kj}$  ( $k \in [1, N]$  and  $j \in [0, P]$ ). Among the elements in “*DistanceSet*”, the values for  $D_j$  ( $j \in [0, P]$ ) are all positive values except the values for  $D_i$  are zero because the values in  $D_j$  ( $j \in [0, P]$ ) are calculated based on the sub images from element  $S_i$ . Hence, we set 0 as the mean of the Gaussian distribution. For all the distance values that are larger than zero, we can treat them as the values in the right side of Gaussian distribution. We compute the value of variance  $\sigma$  of the Gaussian distribution using the formula:

$$\sigma^2 = \frac{1}{N \cdot (P + 1)} \cdot \sum_{j=0}^P \left( \sum_{k=1}^N d_{kj}^2 \right) \quad (6.18)$$

### 6.3.2.2 Learning Parameters for Shape Model

The purpose of this step is to obtain the shape parameters  $X$ . Since the second term of the last expression in Equation 6.14 only relies on the spatial relation among parts, we obtain the following equation

$$X^* = \underset{X}{\operatorname{argmax}} \prod_{k=1}^N p(L_k | X) \quad (6.19)$$

Recall that we use Equation 6.10 to model the prior distribution. Combine Equation 6.10 and Equation 6.19, we obtain the maximal likelihood estimation

$$X^* = \underset{X}{\operatorname{argmax}} \prod_{k=1}^N \prod_{v_0, v_j \in E} p(L_{k0}, L_{kj} | c_{0j}) \quad (6.20)$$

Recall that we use an  $N$ -star graph to represent the object and the  $N$ -star graph is a two level tree structure. Hence, all the connected edges are between root  $v_0$  and leaf  $v_j$ . The location of each node is represented by the  $X - Y$  coordinates of the node. We use two random variables  $\langle X, Y \rangle$  where  $X$  and  $Y$  are the relative spatial difference between part  $v_0$  and  $v_j$  along the coordinate of  $X$  and  $Y$  axes, respectively. We model both  $X$  and  $Y$  using a Gaussian distribution with their own mean and variance values as follows

$$p(x) \approx N(\mu_x, \sigma_x^2) = \frac{1}{\sqrt{2\pi}\sigma_x} \exp\left(-\frac{1}{2}\left(\frac{x - \mu_x}{\sigma_x}\right)^2\right) \quad (6.21)$$

$$p(y) \approx N(\mu_y, \sigma_y^2) = \frac{1}{\sqrt{2\pi}\sigma_y} \exp\left(-\frac{1}{2}\left(\frac{y - \mu_y}{\sigma_y}\right)^2\right) \quad (6.22)$$

Recall that the input for our algorithm is “*IndependentSet*” where *IndependentSet* =  $S_0, S_1, \dots, S_P$  and  $S_i = \{ \langle I_1, L_{1i} \rangle, \langle I_2, L_{2i} \rangle, \dots, \langle I_N, L_{Ni} \rangle \}$  ( $i \in [0, P]$ ). To obtain the mean and variance values of the shape model for part  $v_i$ , we only need the element  $S_i$  from the “*IndependentSet*”. In practice, we can use  $\langle x_{ki}, y_{ki} \rangle$  to represent the relative spatial difference between the centroids of part  $v_0$  and part  $v_i$  of image  $I_k$ . Then we compute parameters  $\mu_{xi}$  and  $\sigma_{xi}$  as follows:

$$\mu_{xi} = \frac{1}{N} \sum_{k=1}^N x_{ki} \quad (6.23)$$

$$\sigma_{xi}^2 = \frac{1}{N} \sum_{k=1}^N (x_{ki} - \mu_{xi})^2 \quad (6.24)$$

$\mu_{yi}$ , and  $\sigma_{yi}$  can be calculated similarly.

### 6.3.3 Recognition

In this section, we introduce our method on determining the existence of the object (appendiceal orifice) given a new image. As we have discussed in Section 6.3.1, we need to find the configurations with the maximal posterior probability in the image. Figure 6.7 shows three major components for recognition in our part-based technique. Given a new image, we first select parts candidates (*Step A*). Then, we calculate the appearance probability for each part (*Step B1*) and compute the shape probability between paired parts (*Step B2*). Finally, we compute the probability score of each configuration and select the best configuration based on the probability value (*Step C*). If the score of the best configuration is more than a predefined threshold, we claim that the image contains the object of interest. We describe each step in detail in the following sections:

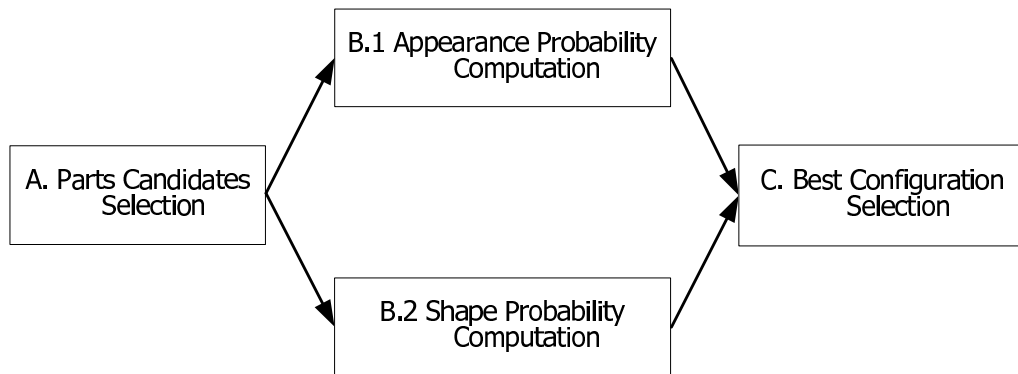


Figure 6.7 System overview of detecting the appendiceal orifice under the part-based statistical framework.

### 1 *Parts Candidate Section*

This is a preprocessing step to identify the candidate parts for computing the appearance and shape probability. The goal is to identify the pixels that likely belong to the object of interest and reduce the number of pixels used for next steps to make our algorithm feasible and efficient. We encode the following domain knowledge to filter out non-object pixels. Our observations and consultations with domain experts indicate that the clearly seen appendix orifice has several curvilinear structures that are part of ellipses. These structures usually are located in the center of the image when the appendix is the focal point of inspection. Based on this fact, we propose an algorithm as below to discard the pixels that are not likely to belong to the appendix orifice.

- Step 1: Curvilinear structure extraction. We obtain the edge image that includes the curvilinear structures from the original image using the Hessian matrix based techniques (Cao et al., 2006).
- Step 2: Non-appendix pixels removal. The edge image from step 1 contains ellipse-shape curves that are part of the appendiceal orifice. But it also includes curves that do not belong to the appendiceal orifice. We determine the curve that is likely the true appendix curve by checking the curvature change along the skeleton of each curve. The skeleton of the curve that is not part of the real appendiceal orifice has

either a small curvature change (curve with linear shape) or a very large curvature change (curve with round shape). Interested readers can refer to Section 6.2 for more details.

- Step 3: Cluster refinement. We refine the results from step 2 by selecting the curves that are more likely to be the real appendix curve by our new modification to the randomized Hough Transform (Xu et al., 1990), abbreviated to *MHT*. The details of *MHT* has been reported in Section 6.2

After performing the above algorithm, we can detect most of the pixels of interests and remove the majority of the pixels that are unlikely part of the appendix orifice. The output of this step is a set of pixels and each pixel will be the centroid of the possible part for next steps. We term the pixel candidate as *pc* in the next section. Our experiments indicate that the candidate selection algorithm can remove 60 to 70 percent of the pixels from the original image. Since the number of the pixels of the original image in our experiments is in the order of  $O(10^4)$ , the number of pixel candidates after part candidate selection is in the order of  $O(10^3)$ .

## 2 Probability Computation

There are two types of probability computation: appearance probability computation and shape probability computation. We can compute them separately since we assume the appearance and the location are independent as assumed in most previous work.

- *Appearance Probability Computation*

The appearance probability is computed based on our assumption that the appearance of the part follows the Gaussian distribution over the distance space. For each pixel candidate from "Parts Candidate Selection" step, we crop a sub image called "*SubImg*" whose centroid is *pc* and window size is  $m \times n$ . Since we decompose the object into  $(P + 1)$  parts, we should compute  $(P + 1)$  probability values for each sub image. Each of them indicates the appearance likelihood of this sub image belongs to the particular part. We give an example to illustrate the computation

of the appearance probability for part  $v_i$  ( $i \in [0, P]$ ). The input of this example are  $pc$  (the centroid of the part) and the corresponding “*SubImg*”,  $S_i$  from the “*IndependentSet*” (see Section 6.3.2 for the definition of “*IndependentSet*”). We first convert the “*SubImg*” into a column vector  $SubI$  with size  $mn \times 1$ . Then we obtain the Euclidean distance between this image and the principal subspace using the following steps:

- Step 1: Compare the sub image  $SubI$  with the mean image  $M = \frac{1}{N} \sum_{k=1}^N SI_k$  where  $SI_k$  is the cropped image from  $S_i$  (for the complete definitions of  $M$  and  $SI_k$ , please refer to step 1 and step 2 of stage 1 in the algorithm description in Section 6.3.2.1). Then we multiple this value with eigenvectors and obtain the corresponding reconstruction weight  $\omega_k = \mu_k^T \cdot (SubI - M)$  for the eigenvector  $\mu_k$ .
- Step 2: Form a new vector of weights  $\Omega^T = [\omega_1, \omega_2, \dots, \omega_M]$  where  $M$  is the number of the largest eigenvalues eigenvectors for part  $v_i$ . We can compute the Euclidean distance between this weight vector and the element from the “*WeightSet*” obtained in step 5 of the stage 1 in the algorithm description of Section 6.3.2.1. The minimal distance  $MinD$  is selected.
- Step 3: Compute the appearance probability score  $p$  of the sub image based on the minimal distance  $MinD$  we obtained from step 2. It is computed based on the Gaussian distribution assumption and the two parameters (mean  $\mu$  and deviation  $\sigma$ ) obtained in Section 6.3.2.1. Specifically, we define

$$p = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2} \cdot \frac{(MinD - \mu)^2}{\sigma^2}\right) \quad (6.25)$$

To further reduce the computation time in the following steps, we discard the sub image with low score since the part with low probability score are not likely to be the real part candidate. In our experiments, more than half of the part candidates are removed. Recall that the number of pixel candidates is in the order of  $O(10^3)$  after “part candidate selection”, we can reduce the number of possible location for



each part to the order of  $O(10^2)$  by this refinement.

- *Shape Probability Computation*

To compute the shape probability for each connected part  $v_0, v_j$ , we model the relative spatial distance between the two parts. The spatial differences of part  $v_j$  with respect to part  $v_0$  are recorded by two variables  $Diff_x$  and  $Diff_y$ . The shape probability is computed using the following joint distribution

$$p(x, y) = p(x) \cdot p(y) = \frac{1}{2\pi\sigma_{xj}\sigma_{yj}} \exp\left(-\frac{1}{2}\left(\left(\frac{Diff_x - \mu_{xj}}{\sigma_{xj}}\right)^2 + \left(\frac{Diff_y - \mu_{yj}}{\sigma_{yj}}\right)^2\right)\right) \quad (6.26)$$

where  $\mu_{xi}$ ,  $\mu_{yi}$ ,  $\sigma_{xi}$ , and  $\sigma_{yi}$  are parameters obtained in Section 6.3.2.2

### 3 Best Configuration Selection

Once we obtain the appearance and shape probability, we can compute the score for each possible configuration. We use a concrete example to illustrate how to get the score for one particular configuration. Suppose we decompose the appendix into four parts  $\{p_0, p_1, p_2, p_3\}$ . We obtain the appearance probability for each part as  $\{AP_0, AP_1, AP_2, AP_3\}$ . Since we are using an  $N$ -star graph, we get 3 shape probabilities,  $\{SP_{01}, SP_{02}, SP_{03}\}$  for the three edges between the root and the leaves. Based on Equation 6.11, we compute the log likelihood of this probability. The final score  $F$  is the sum up of the appearance probability and shape probability for each part. The formula for final score is  $F = \sum_{i=0}^4 \log(AP_i) + \sum_{i=0}^3 \log(SP_{0i})$ . Since we may have multiple candidates for each part, the total computation time is  $O(h^2n)$  where  $n$  is the number of parts and  $h$  is the number of possible locations for each part. This is computational feasible in our case since the number of parts for the object is usually small, about 4 to 7 parts per object. And the number of possible location for each part is usually in the order of  $O(10^2)$ .

## 6.4 Performance Study

In this section, we present the experimental results to illustrate the effectiveness of our proposed approach. We first describe the characteristics of our data sets. Then we discuss our

experiments on model training. Finally, we present our experimental results on ten colonoscopy videos.

#### 6.4.1 Datasets

We used three sets of data for experiments. **(1) Image Set I:** is used for training the decision tree model and SVM model for the first appendix detection technique (feature-based method). **(2) Image Set II:** is used for training the statistic model for the second appendix detection technique (model-based method) to achieve the best performance. **(3) Video Set:** is composed of images from ten colonoscopy videos. We apply both techniques to the video set. Table 6.1 shows the characteristics of “Image Set I”. For the second appendix detection technique (model-based method), we further classify the appendix images into two categories: appendix image class I and appendix image class II. Consequently, we need to construct two object models and the “*Image Set II*” consists of two sets of training images (training set I and training set II, one for each image class). Each training set is composed of both positive examples (images with appendiceal orifice) and negative examples (images without the appendiceal orifice). The location of the object (the appendiceal orifice) in each positive example is carefully annotated by the domain expert. For each run of the experiment, we randomly split the positive examples into two separate clusters with equal size. We then train the model using the first cluster and test the model using the combination of the second cluster and the negative examples. Figure 6.8 and Figure 6.9 illustrate the image examples from the two classes. Table 6.2 shows the number of positive examples and negative examples used in each training set. In our current model structure, we only use positive examples to construct the object model. More sophisticated model structures are needed in the future if we want to train the model using both positive and negative examples. The setting of the training images for the first method is different from the setting for the second technique. In the “*Image Set I*” which is used for model training of the first technique, we do not further categorize the appendix images into multiple classes. This is because the intermediate features for the first technique are global features and can handle the variance among different shapes of appendiceal orifice.

Our experiment also shows that the feature values from different shape of appendiceal orifice are close to each other.

The “*Videos Set*” is also very important to show the effectiveness of our approach. Table 6.3 shows the ten videos and the number of appendix images and non-appendix images in each video. Each video represents one colonoscopic procedure. Recall that we have two models (one for image class I and another for image class II) for the second appendix detection technique (model-based method). Hence, we compute two probability scores for each image. If any of the value is larger than the predefined threshold, we determine that the corresponding image belongs to the appendix image class.

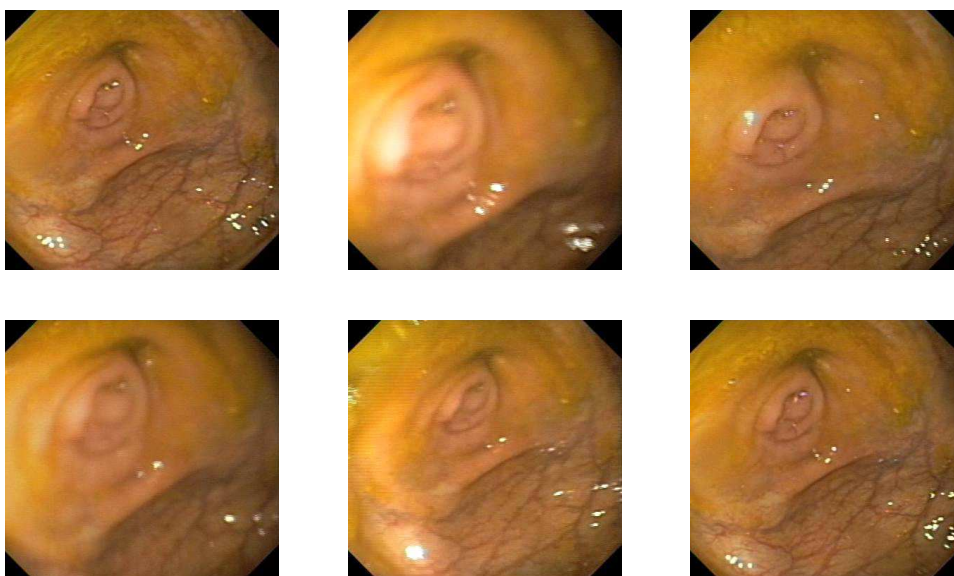


Figure 6.8 Positive training images for the appendix image class I.

Table 6.1 Characteristics of “Image Set I”.

# of Images with Appendiceal Orifice (Positive)	# of Images without Appendiceal Orifice (Negative)
500	500

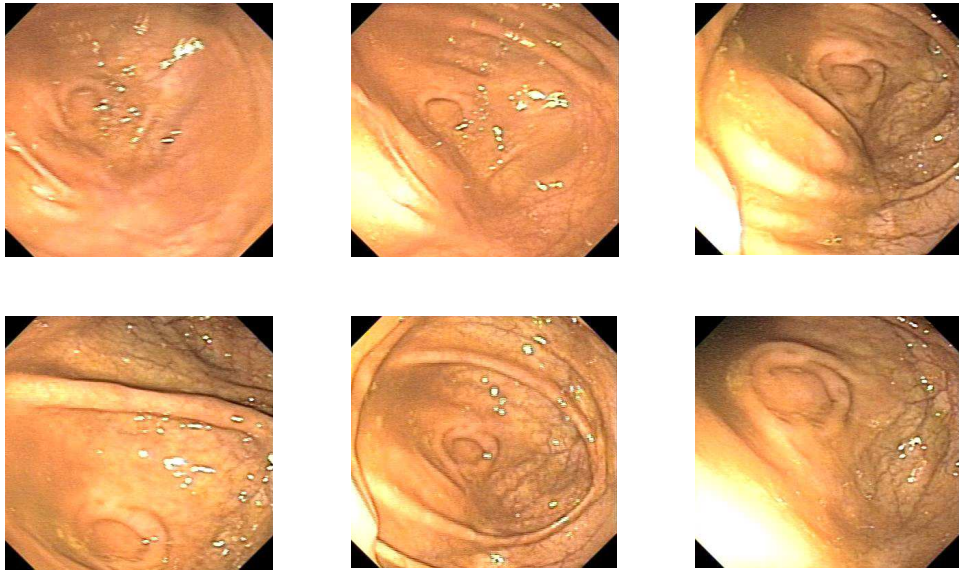


Figure 6.9 Positive training images for the appendix image class II.

Table 6.2 Characteristics of “Image Set II”.

Training Set	# of Images with Appendiceal Orifice (Positive)	# of Images without Appendiceal Orifice (Negative)
Training Set I for Appendix Image Class I	200	100
Training Set II for Appendix Image Class II	500	250

Table 6.3 Characteristics of “Video Set”.

Video ID	Number of Appendix Images	Number of Non-appendix Images
03009	69	361
03010	135	608
03047	18	410
06047	57	473
06048	24	339
06036	153	419
06037	0	397
06038	0	425
06042	0	65
06043	39	257

### 6.4.2 Model Training

The purpose for this step is using the two image sets (“Image Set I” and “Image Set II”) to obtain the model parameters. We discuss model training for the two techniques as below.

We used “*Image Set I*” to train the two classifiers (Decision-Tree and SVM) used in the first technique. We used Weka software from (Witten and Frank, 2005) as our training and testing tool since this software provides the implementation of both Decision-Tree and SVM algorithms. C4.5 (Witten and Frank, 2005) is used as the decision-tree classifier and Sequential Minimal Optimization algorithm (SMO) algorithm with polynomial kernels (Platt, 1998; Witten and Frank, 2005) is used for SVM classifier. The optimal parameters for both classifiers were chosen by performing ten-fold cross validation on the feature sets extracted from “*Image Set I*”.

We used “*Image Set II*” to train the part-based model. First, we need to determine the number of parts used to represent the object, which part of the object should be used as the representative part, and what is the size of each part. These three issues are related with each other. Recall the computation time for the best configuration selection step in our proposed approach is  $O(h^2n)$  where  $n$  is the number of parts. Since  $h$  is in the order of  $O(10^2)$ , the computation time will reach the level of  $O(10^5)$  if the number of parts is equal or more than 10. In order to save the computation time, we should choose less than ten parts. At the same time, the object decomposition should capture the most discriminate part of the object to achieve optimal performance. Intuitively, more parts have more coverage of the object. However, more parts may increase the learning and recognition time and it may cause an over-fitting problem. Since usually the appendix has an ellipse shape, we choose to put parts in the arch of the ellipse. In another word, different arches of the same ellipse are used for part representation. Hence, different parts of the same object have the same scale. Based on this fact, we do not need to use different scales for different parts of the same object. This is different from other part-based approaches (Weber et al., 2000; Felzenszwalb and Huttenlocher, 2005). The size of the part is related with the number of parts for the object and the size of the object in the image. The more parts we decompose the object, the smaller the size of the part, and

vice versa; the larger the size of the object, the larger the size of the part, and vice versa. We ran our classifier using different combinations of the number of parts and the sizes of the parts. The results show that the number of parts does have impact on the performance of the classifier. However, our technique are not very sensitive to the change of the size since the results do not vary much when we performed experiments with different size values between 30 and 50 pixels. In subsequent performance evaluation, we use the size  $45 \times 45$  pixels for the part in class I and  $35 \times 35$  pixels for the part in class II. Figure 6.10 illustrates the error rates for both image classes when varying the number of parts, given the above sizes of the parts. The performance improvement is small if the number of parts is larger than four. Hence, we chose four as the number of parts in subsequent experiments. Another important parameter is the number of eigenvectors; our experiments suggest that while many eigenvectors are necessary for accurate reconstruction of the sub image, the recognition can still be performed correctly using fewer eigenvectors. Our result is similar to that obtained at (Turk and Pentland, 1991). We set this value to 15 based on experiments. Figure 6.11 shows six images from training set I. Each object in the image is decomposed into four parts, which covers part of the arch of the ellipse shape of the appendiceal orifice. We annotated each part of the object with rectangles in different color: red rectangle for part 0 (root), green rectangle for part 1, blue rectangle for part 2, and cyan rectangle for part 3. Cropped sub-images for each part are displayed in Figure 6.12. Figure 6.13 shows the first six largest-eigenvalue eigenvectors for each part. For better illustration of the effectiveness of eigenvectors, we convert each eigenvector into its corresponding eigenimage where the width and the height of the eigenimage are set to 45 for image class I and they are equal to 35 for image class II. From these figures, we can tell that the eigenvectors capture the variance of the original images and is a better representation for classification.

To illustrate the procedure of getting the shape parameters, we use Figure 6.14 to show our capacity of modeling the relative spatial relationship between parts. The top right red “+” sign represents the spatial location of part 0 (root of the N-star graph). Since we model the relative spatial location with respect to the location of the centroid of part 0, we can always put the

part 0 in the origin. Based on the relative spatial difference between the root and the leaves, we can plot the locations of all nodes in the graph in the same plot named the reference frame. Points belong to the same part form a cluster in the reference frame. The three “+” symbols represent the spatial centroid of the cluster of each part. Since we use Gaussian distribution to model the relative spatial relations between the root and the leaves, the points of each part except the first part (root) fall into a single cloud that is denoted by an ellipse. This result verifies the correctness of our assumption of Gaussian distribution. Another important parameter is the threshold to determine the existence of the object. Recall that our algorithms generate a probability score for each image and we select the image with the largest score. If this score is larger than a threshold, we claim the existence of the object in this image. We obtain the threshold value by averaging the best operating point from each ROC curve. The best operating point might be chosen so that the classifier gives the best trade off between the costs of false positive (false positive) against the costs of missing a positive (false negative). Figure 6.15 and Figure 6.16 show the ROC curves for the two models. Each figure contains 30 ROC curves; each of them corresponds to 30 runs of the training set (training set I or training set II) from the “*Image Set*”. We compute the critical ratio  $z$  (Hanley and McNeil, 1983) to determine whether the difference from the two ROC curves is statistically significant or not. It is defined as follows:

$$z = \frac{A_1 - A_2}{\sqrt{SE_1^2 + SE_2^2 - 2rSE_1SE_2}} \quad (6.27)$$

where  $A_1$  and  $SE_1$  indicate the area and the estimated standard error of the first ROC curve; where  $A_2$  and  $SE_2$  refer to corresponding quantities for the second ROC curve; and where  $r$  indicates the estimated correlation between  $A_1$  and  $A_2$ . The area under ROC curve is computed by the trapezoidal rule (Abramowitz and Stegun, 1972). The standard error is obtained by Dorfman and Alf maximum likelihood estimation algorithm (Dorfman and Maximum, 1968). The correlation value  $r$  is based on two values:  $r_{normal}$  (correlation in positive group) and  $r_{abnormal}$  (correlation in negative group). Both  $r_{normal}$  and  $r_{abnormal}$  can be obtained by the traditional Kendall tau rank correlation method (Kendall, 1938). For the 30 ROC curves from

training set I, we select 2 curves each time and the total number of different combinations is  $C_{30}^2$ , which is 435. For each combination, we compute the critical ratio value and we get 435  $z$  values altogether. We assume the  $z$  values are samples from a population that belongs to the normal distribution. Based on the central limit theorem, we estimate the mean and standard deviation of the  $z$  values as 1.166 and 0.012 respectively. We then compute the 95% confidence interval for the mean value and obtain the lower bound and upper bound of the confidence interval as 1.165 and 1.167, respectively. The standard error for the estimated mean is 0.00058. The above statistic values indicate that the difference among each ROC curves are very small (not statistically significant), which means the ROC curves from each run for the training set I are close to each other. Similar results are obtained for training set II. To compare the ROC curves from training set I and training set II, we assume the samples from both training sets belong to normal distribution. Based on this assumption, we perform the two samples two-tailed  $t$  test. Having calculated the  $t$ -statistic, we compare the  $t$ -value with a standard table of  $t$ -values to determine whether the  $t$ -statistic reaches the threshold of statistical significance. From the low  $t$  value (1.68 in our case) and corresponding high  $p$  value (0.1 in our case) we obtained, it is concluded that the difference between the areas under ROC curves from the two models are not statistically significant if we set the significant level as 0.05. In another word, the ROC curves of the two test sets are similar to each other. The average value of the area under ROC curve (0.907 for the first model and 0.894 for the second model) also indicate the good performance of our proposed models.

#### 6.4.3 Test Results on Images from Colonoscopy Videos

After we obtained all the model parameters, we tested our techniques using the images generated from colonoscopic procedures (“*Video Set*”). Since we define the appendix image class as positive results, we can get four values for each video: true positive (the number of real appendix images that are determined as appendix images by our algorithm), false negative (the number of real appendix images but detected as non-appendix images), true negative (the number of real non-appendix images that are detected as non-appendix images by our



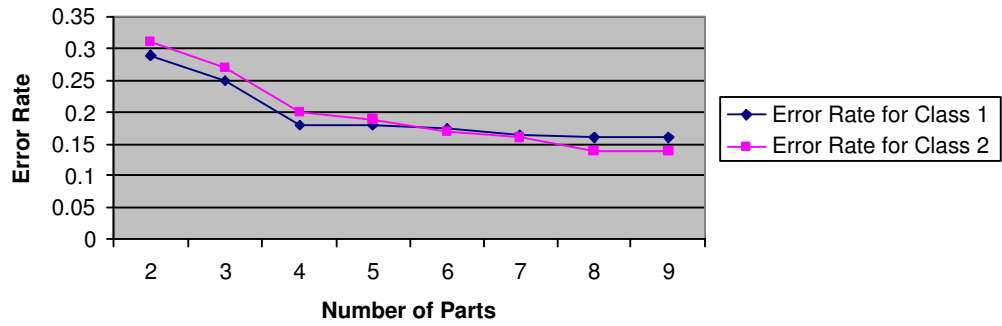


Figure 6.10 The relationship between the number of parts and the error rates for the two image classes.

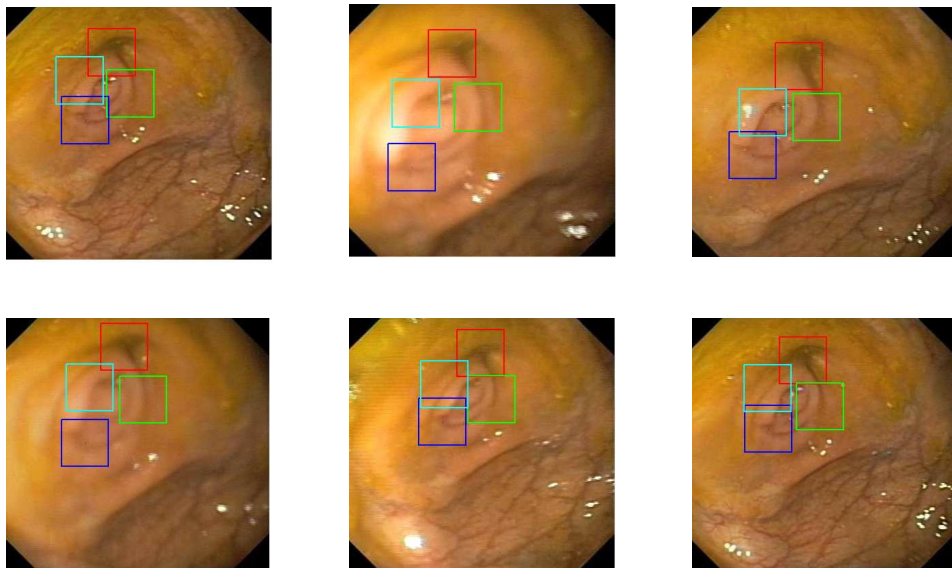


Figure 6.11 Positive training image examples with parts superimposed for training the model of image class I. (This figure is best viewed in color)

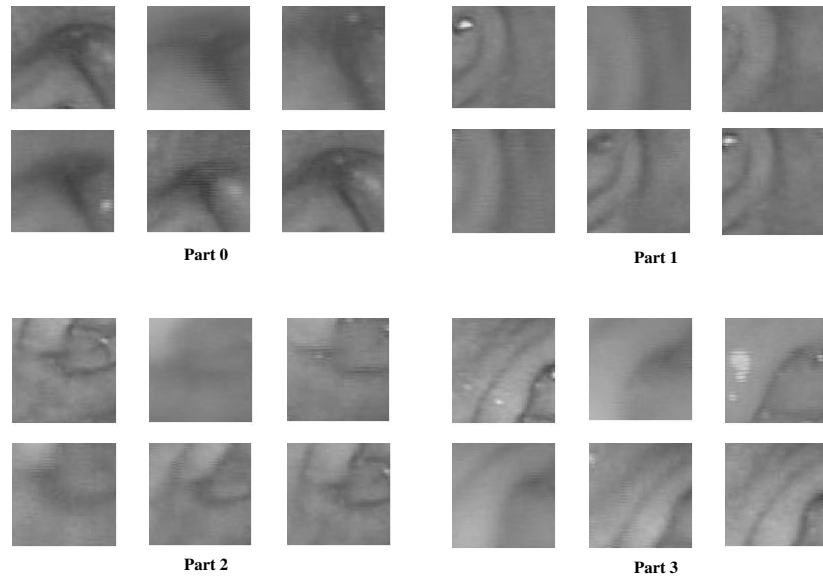


Figure 6.12 Six sub-images for part 0 (root), part 1, part 2, and part 3 cropped from images in Figure 6.11.

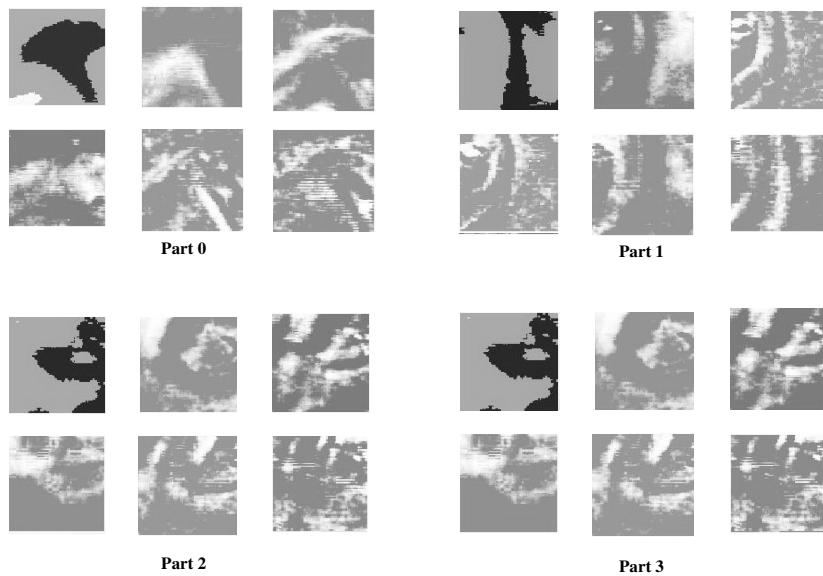


Figure 6.13 The first six largest-eigenvalue eigenvectors for part 0 (root), part 1, part 2, and part 3.

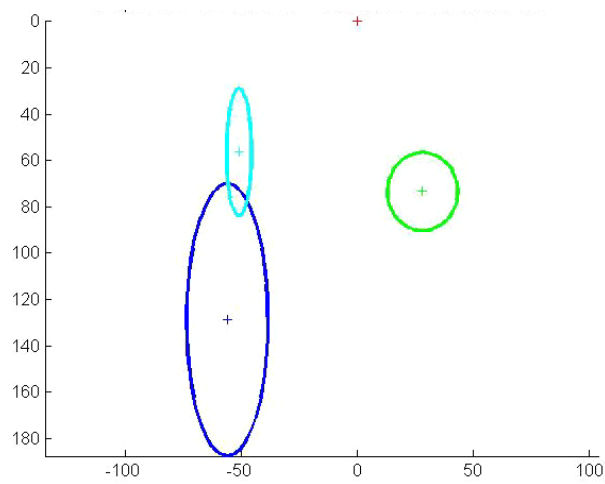


Figure 6.14 Illustration of the shape model for the spatial relations between root and leaves. (This figure is best viewed in color)

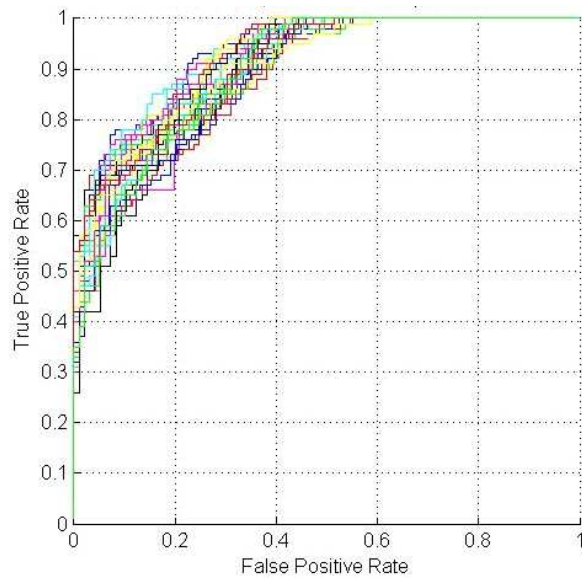


Figure 6.15 Thirty ROC curves of thirty runs on training set I of “Image Set II”.

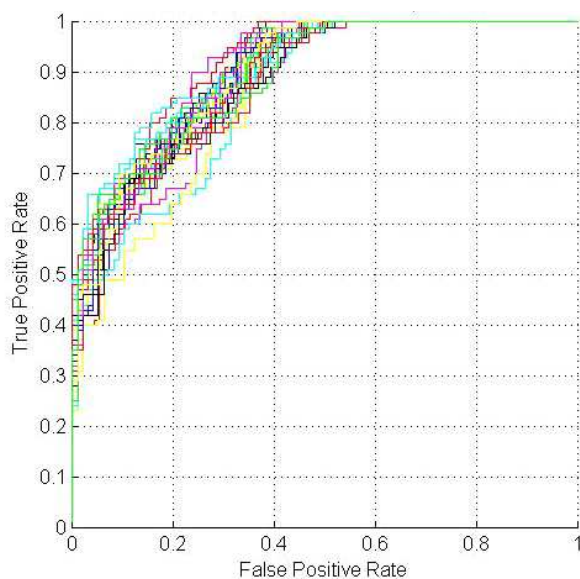


Figure 6.16 Thirty ROC curves of thirty runs on training set II of “Image Set II”.

algorithm), false positive (the number of real non-appendix images but detected as appendix images). We use two widely used performance measurements: sensitivity and specificity, where sensitivity is equal to true positive over by the number of real appendix images, and specificity is defined as true negative divided by the number of real non-appendix images. High sensitivity and specificity are desired.

Table 6.4 shows the results for the experiments on the colonoscopy videos. The first three columns (“KM”, “DS”, “SVM”) denote the three classifiers (“K-Means”, “Decision-Tree”, and “Support Vector Machine”) that belong to the feature-based approach. The column “PB” represents the part-based method that belongs to the model-based approach. Using the similar statistic methods we used in the previous section for comparing the ROC curves for the two models, we find that the average values of sensitivity and specificity for the three classifiers used in the feature-based technique are close to each other (not statistic significant). This indicates that the most critical part of the first appendix detection technique (feature-based technique) is the selection of the features. Once the features have been fixed, the performance difference is small among different classifiers. The average sensitivity and specificity for the

part-based technique are more than 90%. We perform two samples two-tailed  $t$ -test between the results from part-based model and the results from feature-based technique. The  $p$  value is between 0.1 and 0.05, which indicates that the performances for part-based technique are only slightly better than the feature-based technique. The feature-based technique perform poorly for video 06036 because many images in this video have weak edges and some parts of the appendiceal orifice are covered by external materials (for example, stool). However, the impact of the occlusion to the performance of the part-based technique is comparably smaller because the part-based technique is searching for some representative parts of the object instead of searching for the entire object. Even some parts of the object may be occluded, the part-based technique can still identify the object if the most distinctive parts are not being occluded. However, the four intermediate features extracted from these images may not be distinctive enough to differentiate the object from the background if occlusion happens in the images. In addition, the K-Means algorithm performs poorly on the videos without appendix images (video 06037, 06038, and 06042), whereas the other two feature-based technique (“DS” and “SVM”) perform better and the model-based technique can handle this case well. The reason for the failure of K-Means technique is because it produces a number of false positive images even there is no appendix image. The model-based approach does not perform well on video 06047 because some of the appendix images in this video have many strong light reflected areas spread over the colon wall. These light reflected areas may produce a high probability score for the appearance of the part. The problems are rooted from the PCA based appearance model. It has been reported that PCA method is sensitive to illumination change (Belhumeur et al., 1997). To address this issue using other dimension reduction methods, such as fisher linear discrimination techniques (Belhumeur et al., 1997; Martinez and Kak, 2001; Cooke, 2002) will be one of our future works.

## 6.5 Summary

In this chapter, we present two different approaches to solve the problem of appendix image classification. The first one is a feature-based technique and the second one is a model-

Table 6.4 Effectiveness of the appendiceal orifice detection on colonoscopy videos.

Video ID	Sensitivity				Specificity			
	Feature-based			Model-based	Feature-based			Model-based
	KM	DS	SVM	PB	KM	DS	SVM	PB
03009	0.85	0.86	0.87	0.87	0.92	0.92	0.93	0.89
03010	0.91	0.91	0.91	0.92	0.85	0.87	0.87	0.91
03047	0.97	0.97	0.98	0.95	0.86	0.86	0.86	0.92
06047	0.85	0.85	0.86	0.89	0.89	0.90	0.90	0.82
06048	0.84	0.85	0.85	0.92	0.88	0.90	0.90	0.91
06036	0.73	0.77	0.77	0.85	0.72	0.75	0.75	0.93
06037	–	–	–	–	0.85	0.92	0.93	0.98
06038	–	–	–	–	0.87	0.90	0.91	0.98
06042	–	–	–	–	0.83	0.86	0.87	0.94
06043	0.85	0.86	0.86	0.92	0.78	0.82	0.81	0.88
<b>Average</b>	0.85	0.86	0.87	0.90	0.86	0.87	0.87	0.91

based technique. Both of them are applied to the colonoscopy videos and the experimental results show the effectiveness of our proposed approach. In the first approach (feature-based technique), our experiments indicate that the intermediate features we used are effective to capture the global visual properties of some viewpoints of the appendiceal orifice. However, if the variance of the visual property of the same appendiceal orifice is large (for instance, part of the appendiceal orifice is occluded by other objects), the performance of the feature-based technique is poor. The second approach (model-based technique) is a possible alternate technique to address these problems. In this method, the object is represented by some parts of the object. A new Principle Component Analysis (PCA) based appearance model and an N-star graph shape model are used to capture the appearance of each part and the spatial relations among these parts.

## CHAPTER 7. CONCLUSION AND FUTURE WORK

Currently, colonoscopy videos are not captured and maintained in such a way that easily allows post-procedure review or analysis. However, important medical knowledge may be presented in these videos. For instance, important statistics obtained from an endoscopic segment, such as the number of polyps appearing in a segment, and various therapeutic operations performed in a segment, are valuable for diagnosis of colonic diseases. Identifying a video segment with therapeutic or diagnostic operation is useful for reviewing causes of complications due to biopsy or therapeutic operations. The presence of a sufficient number of images showing a closely inspected appendiceal orifice indicates that most distal end of the colon has been reached during the procedure, which is one of the factors used to assess endoscopists' procedural skills. Automatic discovery of the medical knowledge by parsing the colonoscopy videos into semantic units is highly desirable and very useful for educational activities (presentations, teaching of fellows, manuscripts, etc.), for supporting GI endoscopic research, and for mining unknown patterns that may lead to diseases and cancers, and for providing platforms for improving and assessing endoscopists' procedural skills.

In this dissertation, we have defined three semantic units: a colonoscopic scene (a segment of visual and audio data that correspond to an endoscopic segment of the colon), an operation shot (a segment of visual and audio data that correspond to a diagnostic or therapeutic operation of a colonoscopic procedure), and an appendix image (a colon image that contains appendiceal orifice). Because of the unique characteristics of colonoscopy videos, novel algorithms for determining boundaries of colonoscopic scenes and operation shots, and identifying the appendix images, are presented. For scene segmentation, there are two major steps. The first step is audio analysis based parsing algorithm; the second step is the algorithm that

employs a visual analysis method based on a new visual model for colonoscopy videos. For operation shot detection, we convert the problem of detecting operation shots to the problem of identifying the cables of the instruments that are used during the operation. New spatio-temporal segmentation approach are introduced. Finally, we gave two techniques for appendix image classification. The first one is a feature-based method while the other is a model-based technique. The model-based technique performs better than the feature-based technique in our test sets. We have tested our algorithms using the colonoscopy videos captured during colonoscopy procedures. Experimental results show the effectiveness of our algorithms. The frameworks and algorithms presented in this dissertation can be extended to other important endoscopic procedures. In addition, many of the techniques have the potential to be used in medical information system to assist physicians in providing better health care. For example, we are integrating part of the algorithms in this dissertation into a novel quality control system to be used in a clinical trial at Mayo Clinic Rochester.

Our future work include (1) improve the performance of the detection algorithms for appendiceal orifice. Our current model-based method does not perform well when the illumination variations among the objects of interest are large. We plan to integrate other appearance models, such as an appearance model based on fisher linear discriminative method, into our existing statistic framework. Another weak point of our method is that we rely on manual annotation of the object for the positive examples in the training sets. Weakly supervised training or unsupervised training are desired and will be investigated as future work. Other potential improvements include improving the existing model structure by learning from both positive and negative examples; (2) propose new detection algorithms of more types of semantic units. Currently, we use different methods to detect different semantic units. However, a generic model that can take advantage of multiple knowledge sources and parse multiple semantic units is desirable; (3) representation of the detected semantic units in a manner that is easy for retrieval and mining. Ontology related technique provides a possible solution for this problem.



**BIBLIOGRAPHY**

- Abramowitz, M. and Stegun, I. A. E. (1972). *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover Publications, NY.
- Adams, B., Dorai, C., and Venkatesh, S. (2000). Novel approach to determining tempo and drama story sections in motion pictures. In *Proc. of IEEE ICME 2000*, pages 283–286, New York, NY, USA.
- Aguado, A. S., Montiel, M. E., and Nixon, M. S. (1996). Extracting arbitrary geometric primitives represented by fourier descriptors. In *Proc. of IEEE International Conference on Pattern Recognition*, pages 547–551, Vienna, Austria.
- Aigrain, P. and Joly, P. (1994). The automatic real-time analysis of file editing and transition effects and its applications. *Computer and Graphics*, 18(1):93–103.
- Ayache, N. and Faugeras, O. D. (1986). Hyper: a new approach for the recognition and positioning to two-dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:44–54.
- Belhumeur, P. N., Hespanha, J. P., and Kriegman, D. J. (1997). Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:711–720.
- Bimbo, A. D. (1999). *Content-based Video Retrieval*. Morgan Kaufmann Publishers, Inc., San Francisco, CA, USA.
- Briechle, K. and Hanebeck, U. D. (2001). Template matching using fast normalized cross

- correlation. In *Proc. of SPIE: Optical Pattern Recognition XII*, pages 95–102, Orlando, FL USA.
- Cao, Y., Li, D., Tavanapong, W., Oh, J.-H., Wong, J., and deGroen, P.-C. (2004a). Parsing and browsing tools for colonoscopy videos. In *Proc. of ACM Multimedia*, pages 844–851, New York, NY, USA.
- Cao, Y., Liu, D., Tavanapong, W., Oh, J.-H., Wong, J., and deGroen, P.-C. (2006). Automatic classification of image with appendiceal orifice in colonoscopy videos. In *Proc. of IEEE International Conference of the Engineering in Medicine and Biology Society*, pages 2349–2352, New York City, NY, USA.
- Cao, Y., Liu, D., Tavanapong, W., Oh, J.-H., Wong, J., and deGroen, P.-C. (2007). Computer-aided detection of diagnostic and therapeutic operations in colonoscopy videos. *To appear in IEEE Transactions on Biomedical Engineering*.
- Cao, Y., Tavanapong, W., Kim, K.-H., and Oh, J.-H. (2003). Audio-assisted scene segmentation for story browsing. In *Proc. of International Conference on Image and Video Retrieval*, pages 446 – 455, Urbana, IL, USA.
- Cao, Y., Tavanapong, W., Kim, K.-H., Wong, J., Oh, J.-H., and deGroen, P.-C. (2004b). A framework for parsing colonoscopy videos for semantic units. In *Proc. of IEEE International Conference on Multimedia and Expo*, pages 1879–1882, Taipei, Taiwan.
- Cao, Y., Tavanapong, W., Li, D., and Oh, J.-H. (2004c). A visual model approach for parsing colonoscopy videos. In *Proc. of International Conference on Image and Video Retrieval*, pages 160 – 169, Dublin, Ireland.
- CarnegieMellonUniversity (2003). A large vocabulary, speaker independent speech recognition codebase and suite of tools. In *www.speech.cs.cmu.edu/sphinx/*, Retrieved on April 2003, Pittsburgh, PA, USA.

- Chen, D., Wax, M., Li, L., Liang, Z., Li, B., and Kaufman, A. (2000). Novel approach to extract colon lumen from images for virtual colonoscopy. *IEEE Transactions on Medical Imaging*, 19:1220–1226.
- ColumbiaUniversity (2004). *Columbia Electronic Encyclopedia*. Columbia University Press, New York, NY.
- Cooke, T. (2002). Two variations on fisher’s linear discriminant for pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:268–273.
- Corridoni, J. M. and Bimbo, A. D. (1998). Structured representation and automatic indexing of movie information content. *Pattern Recognition*, 31(12):2027–2045.
- Crandall, D., Felzenszwalb, P., and Huttenlocher, D. (2005). Spatial priors for part-based recognition using statistical models. In *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 10–17, San Diego, CA, USA.
- Dario, P. and Lencioni., M. (1997). A microrobotic system for colonoscopy. In *Proc. of the IEEE International Conference on Robotic and Automation*, pages 1567–1572, Florence, Italy.
- Dawood, A. and Ghanbari, M. (1999). Clear scene cut detection directly from MPEG bit streams. In *Proc. of IEEE Internation Conference on Image Processing and Its Applications*, volume 1, pages 285–289, Manchester, UK.
- Deng, Y. and Manjunath, B. S. (2001). Unsupervised segmentation of color-texture regions in images and video. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23:800–810.
- Dorfman, D. D. and Maximum, A. E. (1968). Maximum likelihood estimation of parameters of signal detection theory—a direct solution. *Psychometrika*, 33:117–124.
- Eakins, J. P., Riley, K. J., and Edwards, J. D. (2003). Shape feature matching for trademark image retrieval. In *Proc. of IEEE International Conference on Image and Video Retrieval*, pages 28–38, Urbana-Champaign, IL, USA.

- Elia, C. D., Poggi, G., and Scarpa, G. (2003). A tree-structured markov random field model for bayesian image segmentation. *IEEE Transactions on Image Processing*, 12:1259–1273.
- Esgiar, A., R.N.G Naguid, B. S., Bennett, M., and Murray, A. (1998). Microscopic image analysis for quantitative measurement and feature identification of normal and cancerous colonic mucosa. *IEEE Transactions on Information Technology in Biomedicine*, 2:197–203.
- Felzenszwalb, P. F. and Huttenlocher, D. P. (2005). Pictorial structures for object recognition. *International Journal of Computer Vision*, 61:55–79.
- Fergus, R., Perona, P., and Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 264–271, Madison, WI, USA.
- Fergus, R., Perona, P., and Zisserman, A. (2006). Weakly supervised scale-invariant learning of models for visual recognition. *Accepted by the International Journal of Computer Vision*.
- Fischler, M. A. and Bolles, R. C. (1997). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395.
- Gamaz, N., Huang, X., and Panchanathan, S. (1998). Scene change detection in MPEG domain. In *Proc. of IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 12–17, Tucson, AZ, USA.
- Grimson, W. E. L. and Lozano-Prez, T. (1987). Localizing overlapping parts by searching the interpretation tree. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9:469–482.
- Haker, S., Angenent, S., Tannenbaurn, A., and Kikinis, R. (2000). Nondistorting flattening maps and the 3 visualization of colon images. *IEEE Transactions on Medical Imaging*, 19:665–670.

- Hamilton, P.W., Bartels, P., Thompson, D., Anderson, N., Montironi, R., and Sloan, J. (1997). Automated location of dysplastic fields on colorectal history using image texture analysis. In *J. Pathol*, pages 68–75.
- Hampapur, A., Jain, R., and Weymouth, T. (1995). Production model based digital video segmentation. *Multimedia Tools and Applications*, 1(1):9–46.
- Hanjalic, A., Lagendijk, R. L., and Biemond, J. (1999). Automated high-level movie segmentation for advanced video-retrieval systems. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(4):580–588.
- Hanjalic, A. and Zhang, H.-J. (1999). Optimal shot boundary detection based on robust statistical models. In *Proc. of IEEE International Conference Multimedia Computing and Systems*, pages 710–714, Florence, Italy.
- Hanley, J. and McNeil, B. (1983). A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology*, 148:839–843.
- Hietala, R. and Oikarinen, J. (2000). A visibility determination algorithm for interactive virtual endoscopy. In *Proc. of the conference on Visualization '00*, pages 29–36, IEEE Computer Society Press.
- Huttenlocher, D. P., Klanderman, G. A., and Rucklidge, W. J. (1993). Comparing images using the hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:850–863.
- Huttenlocher, D. P. and Ullman, S. (1990). Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5:195–212.
- Jemal, A., Siegel, R., Ward, E., Murray, T., Xu, J., and Thun, M. J. (2007). Cancer statistics, 2007. *CA Cancer J Clin*, 57:43–66.
- Karkanis, S., Iakovidis, D., Maroulis, D., Karras, D., and Tzivras, M. (2003). Computer-aided

- tumor detection in endoscopic video using color wavelet features. *IEEE Transactions on Information Technology in Biomedicine*, 7:141–152.
- Kendall, M. (1938). A new measure of rank correlation. *Biometrika*, 30:81–93.
- Khessal, N. and Hwa., T. (2000). The development of an automatic robotic colonoscope. In *Proc. of IEEE International Region 10 Conference 2000*, pages 71–76, Kuala Lumpur, Malaysia.
- Koh, C. and Gillies., D. (1994). Using fourier information for the detection of the lumen in endoscopy images. In *Proc. of IEEE International Region 10 Conference*, pages 981–985, Singapore.
- Lakare, S., Chen, D., Li, L., Kaufman, A., Wax, M., and Liang, Z. (2002). Electronic colon cleansing using segmentation rays for virtual colonoscopy. In *Proc. of SPIE 2002 Symposium on Medical Imaging*, San Diego, CA, USA.
- Lee, S.-W., Kim, Y.-M., and Choi, S. W. (2000). Fast scene change detection using direct feature extraction from mpeg compressed video. *IEEE Transactions on Multimedia*, 2:240–254.
- Lieberman, D. (2005). Quality and colonoscopy: a new imperative. *Gastrointestinal Endoscopy*, 61:385–91.
- Lienhart, R. (1999). Comparison of automatic shot boundary detection algorithms. In *Proc. of SPIE Storage and Retrieval for Image and Video Databases VII*, pages 290–301, Boston, MA, USA.
- Lim, Y. and Lee., J. (2001). A self-propelling endoscopic system. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1117–1122, Maui, HI, USA.
- Lin, T. and Zhang, H.-J. (2000). Automatic video scene extraction by shot grouping. In

- Proc. of the 15h International Conference on Pattern Recognition*, volume 4, pages 39–42, Barcelona, Spain.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110.
- Lu, L., Jiang, H., and Zhang, H. (2001). A robust audio classification and segmentation method. In *Proc. of ACM Multimedia*, pages 203–211, Ottawa, Ontario, Canada.
- M.A.Fischler and R.A.Elschlager (1973). Comparing images using the hausdorff distance. *IEEE Transactions on Computer*, 22:67–92.
- Martinez, A. M. and Kak, A. C. (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:228–233.
- Moghaddam, B. and Pentland, A. (1997). Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:696–710.
- Murase, H. and Nayar, S. K. (1995). Visual learning and recognition of 3-d objects from appearance. *International Journal of Computer Vision*, 14:5–24.
- Nam, J. and Tewfik, A. (2000). Dissolve transition detection using b-splines interpolation. In *Proc. of IEEE International Conference on Multimedia and Expo*, pages 1349–1352, New York City, NY, USA.
- Nang, J., Hong, S., and Ihm, Y. (1999). An efficient video segmentation scheme for MPEG video stream using macroblock information. In *Proc. of ACM Multimedia'99*, pages 23–26, Orlando, FL, USA.
- Naphade, M. R., Mehrotra, R., Ferman, A. M., Warnick, J., Huang, T. S., and Tekalp, A. M. (1998). A high-performance shot boundary detection algorithm using multiple cues. In *Proc. of IEEE International Conference on Image Processing*, pages 884 – 887, Chicago, Illinois, USA.

- NationalCancerInstitute (2007). Anatomy of colon and rectum. In [http://training.seer.cancer.gov/ss\\_module04\\_colon/unit02\\_sec01\\_anatomy.html](http://training.seer.cancer.gov/ss_module04_colon/unit02_sec01_anatomy.html), Retrieved on Feb 2007.
- Phee, S. and Ng., W. (1998). Automatic of colonoscopy: Visual control aspects. *IEEE Eng. in Medicine and Biology Magazine*, 17:81–88.
- Platt, J. (1998). *Fast Training of Support Vector Machines using Sequential Minimal Optimization*. MIT Press, MA.
- Poggi, G. and Ragozini, R. P. (1999). Image segmentation by tree-structured markov random fields. *IEEE Signal Processing Letters*, 6:155–157.
- Rui, Y., Huang, T. S., and Mehrotra, S. (1999). Constructing table-of-content for videos. *ACM Multimedia Systems*, 7(5):359–368.
- R.Zabih, J.Miller, K. (1999). A feature-based algorithm for detecting and classification production effects. *Multimedia Systems*, 7:119–128.
- Sato, Y., Nakajima, S., Shiraga, N., Atsumi, H., and Yoshida, S. (1998). Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. *Medical Image Analysis*, 2:299–305.
- Sharghi, M. and I.W, R. (2001). A novel method for accelerating the visualization process used in virtual colonoscopy. In *Proc. of International Conference on Information Visualization*, pages 167–172, London, England, UK.
- Shin, T., Kim, J.-G., Lee, H., and Kim, J. (1998). A hierarchical scene change detection in an MPEG-2 compressed video sequence. In *Proc. of IEEE International Symposium on Circuits and Systems*, volume 4, pages 253–256, Monterey, CA, USA.
- Shuttleworth, J., A.T., R. N. G. N., Newman, R. M., and Bennett, M. (2002). Color texture analysis using co-occurrence matrices for classification of colon cancer images. In *IEEE Canadian Conference on Electrical and Computer Engineering*, pages 12–15, Winnipeg, Canada.



- Sonka, M., Hlavac, V., and Boyle, R. (2000). *Image Processing, Analysis, and Machine Vision*. Thomson-Engineering, New York, NY.
- Sucar, L. and Gillies., D. (1990). Knowledge-based assistant for colonoscopy. In *Proc. of the 3rd International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, pages 665–672, New York, NY, USA.
- Sundaram, H. and Chang, S. F. (2000a). Determining computable scenes in films and their structures using audio-visual memory models. In *Proc. of ACM Multimedia'00*, pages 95–104, Los Angeles, CA, USA.
- Sundaram, H. and Chang, S. F. (2000b). Video scene segmentation using audio and video features. In *Proc. of IEEE ICME*, pages 1145–1148, New York, NY, USA.
- Tan, T. N. (1998). Rotation invariant texture features and their use in automatic script identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:751–756.
- Todman, A., Naguib, R., and Bennett., M. (2000). Visual characterization of colon images. In *Proc. of Medical Image Understanding and Analysis*, Birmingham U.K.
- Toews, M., Collins, D. L., and Arbel, T. (2006). A statistical parts-based appearance model of inter-subject variability. In *Proc. of the International Conference on Medical Image Computing and Computer Assisted Intervention*, pages 232–240, Copenhagen, Denmark.
- Truong, B. T., Dorai, C., and Venkatesh, S. (2000). New enhancements to cut, fade, and dissolve detection processes in video segmentation. In *Proc. of ACM Multimedia*, pages 219–227, Los Angeles, CA, USA.
- Turk, M. A. and Pentland, A. P. (1991). Face recognition using eigenfaces. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–591, Maui, HI, USA.
- U.Gargi, Kasturi, R., and S.H.Strayer (2000). Performance characterization of video-shot-change detection methods. *IEEE Transaction on Circuits and Systems for Video Technology*, 10:1–13.

- Veneau, E., Ronfard, R., and Bouthemy, P. (2000). From video shot clustering to sequence segmentation. In *Proc. of International Conference on Pattern Recognition*, pages 254–257, Barcelona, Spain.
- Weber, M., Welling, M., and Perona, P. (2000). Towards automatic discovery of object categories. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 101–108, Hilton Head Island, SC, USA.
- Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufuman, New York, NY.
- Xu, L., Oja, E., and Kultanen, P. (1990). A new curve detection method: Randomized hough transform (rht). *Pattern Recognition Letters*, 11:331–338.
- Xu, L., Oja, E., and Kultanen, P. (1998). Randomized hough transform: improved ellipse detection with comparison. *Pattern Recognition Letters*, 19:299–305.
- Yeo, B. L. and Liu, B. (1995). Rapid scene analysis on compressed video. *IEEE Transactions on Circuits and Systems for Video Technology*, 5(6):533–544.
- Yeung, M. M. and Liu, B. (1995). Efficient matching and clustering of video shots. In *Proc. of IEEE International Conference on Images Processing*, volume 1, pages 338–341, Washington, DC, USA.
- Yusoff, Y. and Kittler, J. (2000). Video shot cut detection using adaptive thresholding. In *Proc. of the British Machine Vision Conference*, pages 362–371, Bristol, UK.
- Zhang, H. J., Kankanhalli, A., and Smoliar, S. W. (1993). Automatic partitioning of full-motion video. *ACM Multimedia Systems*, 1(1):10–28.
- Zhang, H. J., Wu, J. H., Zhong, D., and Smoliar, S. (1997). Video parsing, retrieval and browsing: An integrated and content-based solution. *Pattern Recognition (Special Issue on Image Databases)*, 30(4):643–658.

Zhuang, Y., Rui, Y., Huang, T. S., and Mehrotra, S. (1998). Adaptive key frame extraction using unsupervised clustering. In *Proc. of International Conference on Image Processing*, volume 1, pages 866–870, Chicago, IL, USA.