This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

A Real-Time State Dependent Region Estimator for Autonomous Endoscope Navigation

J. Micah Prendergast, Gregory A. Formosa, *Student Member, IEEE*, Mitchell J. Fulton, Christoffer R. Heckman, *Member, IEEE*, and Mark E. Rentschler, *Senior Member, IEEE*

***Abstract*—With significant progress being made toward improving endoscope technology such as capsule endoscopy and robotic endoscopy, the development of advanced strategies for manipulating, controlling, and more generally, easing the accessibility of these devices for physicians is an important next step. This article presents an autonomous navigation strategy for use in endoscopy, utilizing a state-dependent region estimation approach to allow for multimodal control design. This region estimator is evaluated for its accuracy in predicting yaw angle of the camera relative to the lumen center, and for estimating the location of the camera based on overall haustra morphology within the colon. To assess the utility of this region estimator, multimodal control is used to allow for autonomous navigation of the Endoculus, a robotic capsule endoscope, within a benchtop, to-scale, simulated colon. The estimation approach is presented and tested, demonstrating successful tracking of fixed velocity rotations at speeds up to 40°/s and allowing for curve anticipation approximately 10 cm before entering a curved section of the simulator. Finally, the multimodal control strategy utilizing this estimator is tested within the simulator over a variety of anatomic configurations. This strategy proves successful for navigation in both straight sections of this simulator and in tightly curved sections as small as 8 cm radius of curvature, with average velocities reaching 2.61 cm/s in straight sections and 0.99 cm/s in curved sections.**

***Index Terms*—Autonomous navigation, capsule endoscopy, computer vision.**

I. Introduction

TRADITIONAL endoscopes and colonoscopes are typically used to diagnose and treat a host of gastrointestinal (GI) diseases. While many attempts have been made toward developing less invasive alternatives to these devices such as wireless capsule endoscopes, these systems are diagnostically limited due to their passive nature and do not offer the intervention capabilities of traditional scopes. Recent work toward robotic capsule endoscopes with the full diagnostic and treatment potential of conventional scopes has shown great promise, however, the challenges of navigating these devices within the body remains. While substantial efforts have been devoted to achieving autonomous or assisted navigation of endoscopes to improve these procedures, these solutions have lacked the robustness and adaptability required of a medically adoptable autonomous or assisted navigation solution.

We present a novel visual navigation strategy for lumen center tracking comprising a high level state machine for gross (i.e., left/right/center) region prediction and curvature estimation and multiple state-dependent controllers for center tracking, wall avoidance, and curve following. This structure allows our navigation system to navigate even under the presence of significant occlusion that occurs during turn navigation and to robustly recover from mistakes and disturbances that may occur while attempting to track the lumen center. We evaluate this strategy onboard the Endoculus [1], a treaded and steerable robotic capsule endoscope with the full toolkit of a conventional colonoscope (see Fig. 1).

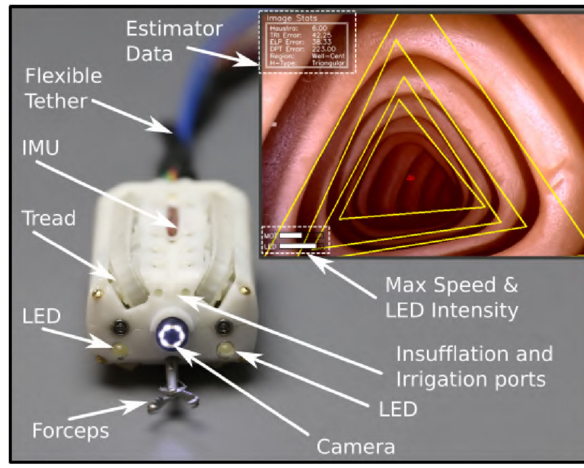


Fig. 1. Endoculus [1] shown with active forceps. The user interface (inset) shows a view of the lumen with four haustral folds identified and a lumen center estimated. Note that user settings are also shown in the interface as well as various image information (upper left of inset).

Manuscript received July 25, 2020; accepted October 13, 2020. This work was supported in part by the National Science Foundation (NSF) under Grant 1849357, in part by the CU-Boulder Innovative Seed Grant Program and in part by the CU-Boulder Economic Development and International Trade. This research was also conducted with government support under and awarded by the Department of Defense, Air Force Office of Scientific Research and National Defense Science and Engineering Graduate (NDSEG) Fellowship 32 CFR 168a. This article was recommended for publication by Associate Editor A. Krupa and Editor P. Dupont upon evaluation of the reviewers' comments. *(Corresponding author: J. Micah Prendergast.)*

J. Micah Prendergast, Gregory A. Formosa, Mitchell J. Fulton, and Mark E. Rentschler are with the Department of Mechanical Engineering, University of Colorado, Boulder, CO 80309 USA (e-mail: jopr1310@colorado.edu; gregory.formosa@colorado.edu; Mitchell.Fulton@colorado.edu; mark.rentschler@colorado.edu).

Christoffer R. Heckman is with the Department of Computer Science, University of Colorado, Boulder, CO 80309 USA (e-mail: christoffer.heckman@colorado.edu).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TRO.2020.3038709.

Digital Object Identifier 10.1109/TRO.2020.3038709

1552-3098 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

A. Motivation

Colorectal cancer (CRC) remains the third leading cause of cancer deaths in the U.S. for both men and women [2] despite being very responsive to treatment with a relative five year survival rate of 90% if detected in the early and localized stage. Regular CRC screening via conventional colonoscopy is the most effective method of early detection and serves not only to identify CRC, but also to prevent it, by removing potentially precancerous polyps using a semirigid scope inserted into a patient's anus, rectum, and colon [3]. A recent study [4] showing an increase in CRC in younger adults has recently prompted a revision to the CRC screening timeline by the American Cancer Society (ACS) which previously recommended that adults over 50 years of age receive a colonoscopy once every ten years. The ACS now recommends that adults of average risk for CRC begin screenings at 45 years of age, however, despite these recommendations, a 2010 study showed that only 56% of adults 50 years or older had actually followed the ACS guidelines.

One issue may lie in the perceived invasiveness of colonoscopy procedures which can be uncomfortable for both patients and physicians. It is estimated that up to 90% of patient pain during colonoscopies is caused by scope looping, where the colonoscope continues to advance into the colon without a simultaneous progression of the tip. Looping leads to colon distension and in severe cases, can cause tissue damage and even perforation [5]. Improvements to this procedure, however, may significantly reduce or even eliminate the potential for looping. Novel devices, such as robotic capsule endoscopes (RCEs), with the ability to pull themselves rather than push their way into the colon, may significantly decrease the invasiveness of colonoscopy procedures.

Novel methods for controlling robotic endoscopes (whether they be mobile platforms or more similar to conventional scopes) may also serve to advance these procedures as a whole; improving navigation tasks, while simultaneously searching for and detecting any potential diseases. In an attempt to revolutionize this important screening procedure, mobile robotic endoscopes along with the necessary strategies for navigating and controlling these devices have been under development. While these RCE's have boasted a diverse range of actuation techniques [1], [6]–[11], autonomous navigation for these devices within the complex and deformable *in vivo* environment still remains largely out of reach.

B. Endoscope Navigation

A wide range of attempts have been made to develop autonomous visual navigation solutions for flexible and capsule endoscopes, including lumen centralization methods and feature tracking methods. Unfortunately, many of these strategies have failed to perform in real time due to computational constraints, or struggle to perform when the lumen center is not immediately visible [12]. The most common methods for autonomous endoscope navigation have focused on darkest or deepest region techniques in which the darkest and/or deepest region in an image is identified and used as the goal for adjusting endoscope heading. One advantage to these techniques is that they are typically very simple to execute, and a number of studies have examined different techniques for both segmentation of these dark regions and 3-D depth estimation utilizing shape-from-shading or structured light approaches [13]–[21]. While success has varied widely, all of these approaches suffer from a fundamental flaw in making the assumption that the region of greatest depth within any given image represents a useful goal for immediate heading adjustment. While reaching the deepest point in an image may be a very reasonable goal for endoscope navigation on a larger scale, the lighting conditions, focal length of the endoscopic camera, and geometry of surrounding tissue all have a significant impact on how far away this maximum depth goal may be. As depicted by the solid colored RCE in Fig. 2, an algorithm solely based on perfect alignment of the camera toward the deepest point in every image ignores both the size of the endoscope itself and the 3-D structure of the surrounding anatomy, and thus, will inherently limit the ability of the scope to progress down the lumen. Ideally a heading control algorithm should attempt to align the front end of the scope parallel to the lumen walls at all times so as to allow for both mobility and visualization (as shown by the transparent RCE in Fig. 2), however, maximum depth estimates will only achieve this goal when there is no perceivable curvature in the upcoming tissue (i.e., long, straight sections). Lumen centralization methods using the edges or contours of structures surrounding the lumen have also been demonstrated [22]–[24]. These methods provide a more useful goal for immediate heading adjustment in that the surrounding structures are by definition much closer to the camera than the point of maximum depth, however, large structures like haustral folds may not always be present in images particularly under the presence of occlusion and during sharp turns when an endoscope faces the colon wall for much of the maneuver. Other methods utilizing feature tracking [25] and optical flow [26] also

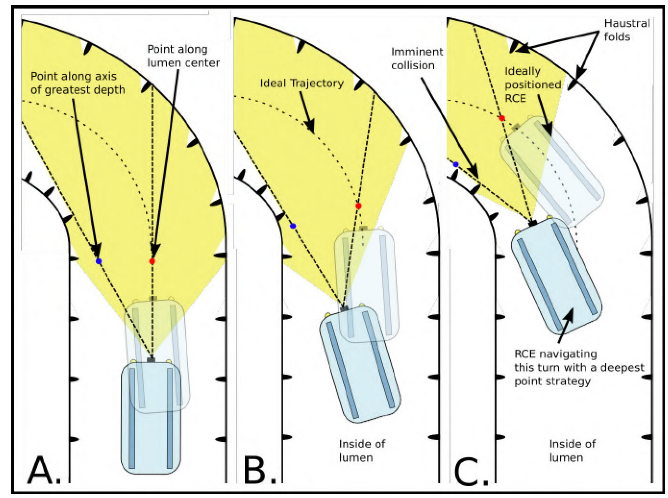


Fig. 2. RCE is shown in the lumen facing an upcoming turn. The point of greatest depth (darkest point) and the lumen center are both shown (A). If the RCE navigates toward the deepest point the device motion will inevitably be biased toward the inner wall of the turn (B), eventually occluding the camera and potentially colliding with the wall (C). Ideally the device will track the true lumen center as shown by the transparent RCE in each frame.

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

show promise, however, these risk being misled by occlusion, deformation, and adverse or changing lighting conditions.

A truly robust navigation algorithm must be able to handle a difficult and diverse set of structural environments within an anatomy that can vary widely between patients. While well-centered views of the colon may show significant structure, the tight turns and deformability of the colon essentially guarantee that a large percentage of images will not be well-centered in the lumen, but will instead show close views of the lumen wall, resulting in an occluded camera with no view of the larger structure. To account for these difficult cases, we present a state dependent region estimation method based on an explicit analysis of the large structures present in each image. This system comprises a high level state machine for gross region prediction, a turn estimator for anticipating sharp turns, and several lower level controllers for heading adjustment. From this structure, the system is able to utilize lumen centralization control approaches when the device has clear views of the lumen, while employing alternative estimation/control approaches around sharp curves. This structure allows our navigation system to operate even under the presence of significant occlusion from the lumen wall (when accurate lumen center estimates are much more difficult to make) and to anticipate and respond to upcoming turns without user intervention. We test this strategy onboard the Endoculus, [1] a treaded and steerable robotic capsule endoscope.

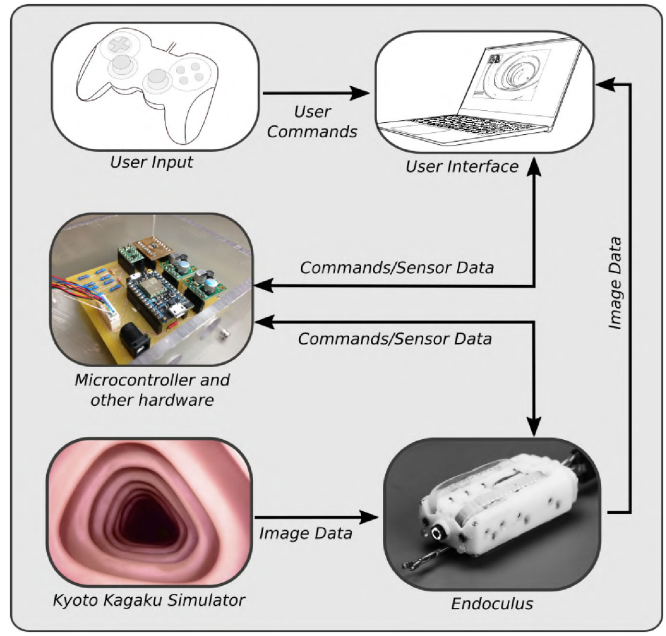


Fig. 3. Overview of the Endoculus system is shown including the joystick input controller, laptop, external hardware, and the Endoculus itself. A view of the Kyoto Kagaku colon simulator is also shown as well as data flow between each system component.

II. Materials and Methods

Section II is broken into five sections. In Section II-A, the Endoculus and supporting hardware/user interface is briefly described. Next, Section II-B presents the image preprocessing steps and the segmentation process and segment classification steps are described. This is followed in Section II-C by a detailed description of the lumen center estimation process for each segment. Section II-D describes the various control strategies used onboard the device. Finally, Section II-E details the experimental evaluation of the image processing algorithm and its subsequent efficacy in the overall navigation system.

A. Endoculus System

1) Endoculus: The Endoculus design incorporates Faulhaber Series 0615–4.5S DC Micromotors (Faulhaber, Croglio, Switzerland) to allow for left/right steering of the device. A small camera (Kzyee 5.5mm Wireless Endoscope, Wuzhou Jin Zhengyuan Technology Co. Wuzhou, China) and two LED's are housed in the front of the device, and an inertial measurement unit (LSM9DS1, STMicroelectronics, Geneva, Switzerland) is housed in the top of the chassis. Two onboard motor encoders (PA2-50, Faulhaber, Croglio, Switzerland) allow for measuring individual motor speed. The Endoculus system also incorporates tool ports for biopsy forceps/snares, irrigation and insufflation. Finally, the Endoculus itself is tested within the Kyoto Kagaku Colonoscope Training Model (Kyoto Kagaku Co. Ltd, Kyoto, Japan). This benchtop simulator has been clinically validated as an exceptional training module with excellent accuracy compared to the *in vivo* colon [9], [27]. The Endoculus, simulator, and additional system components are shown in Fig. 3.

2) Offboard Electronics: The Endoculus is controlled externally via a custom interface comprising a Wi-Fi enabled device (Photon, Particle Inc., San Francisco, CA) with an onboard ARM Cortex M3 microcontroller (STM32F205 120 MHz ARM Cortex M3), a dual motor driver (DRV8835, Texas Instruments, Dallas, TX) current sensors for estimating motor torque (LTC2991, Linear Technologies, Milpitas, CA), an offboard solenoid for controlling insufflation and two switching transistors for controlling LED brightness and the offboard solenoid (NITRA AVS-3111-24D, AutomationDirect, Cumming, GA). The electronics interface connects to a host computer via a USB serial connection. In addition, the Endoculus camera connects directly to the host computer via USB as shown by the data flow arrows in Fig. 3.

3) User Interface: A custom Python interface shown in Fig. 4B and 4C was built for sending and receiving user commands and/or autonomous control commands to/from the Endoculus and for collecting and visualizing raw/processed video as well as other data from the Endoculus system. The user can interact with this interface via an off-the-shelf gaming controller (Gamepad F310, Logitech International S.A., Lausanne, Switzerland). Once the system is running, the Endoculus system is solely controlled from this input device and the user is able to begin/stop a test run, steer each side of the device motors independently (tank steering), adjust max motor speed, adjust LED brightness, insufflate, and record snapshots and video all from the gamepad. In addition, for testing control strategies, the user is able to enter/exit autonomous control modes with the press of a button. Feedback to the user is accomplished via the Endoculus video display, which overlays user selectable information including: estimated viewing region/lumen center

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

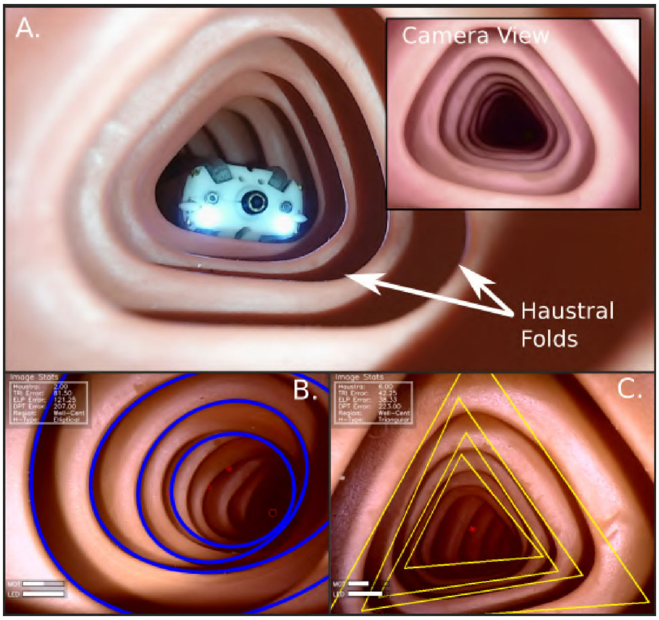


Fig. 4. Endoculus is shown within the Kyoto Kagaku colon simulator (A). The inset is the view from the Endoculus. The user interface is shown in both an elliptical (B) and triangular (C) region of the simulator.

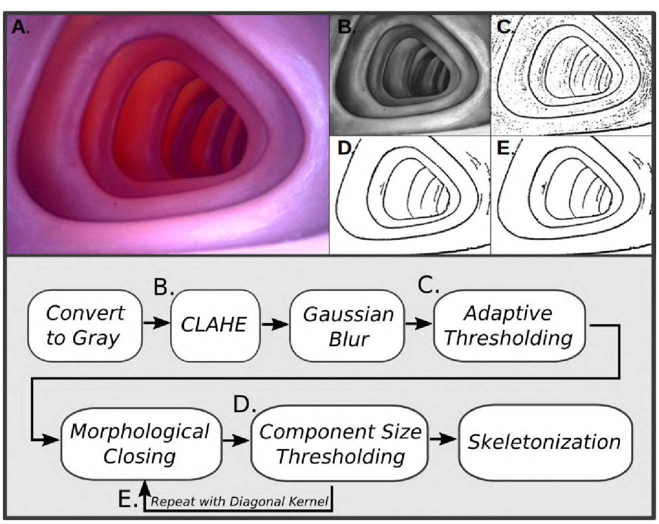


Fig. 5. Several of the preprocessing steps are shown, with the original image (A), including conversion to grayscale with contrast adjustment (B), adaptive thresholding (C), component size filtering (D) and morphological closing (E). Block diagram at the bottom shows these preprocessing steps. This process is completed for every RGB image before the final segments are extracted and characterized.

estimated position, the number of segmented haustral folds, predicted motion, and current operating mode/settings of the device (auto, manual, LED brightness, max motor speed, etc.).

B. Segmentation and Classification

1) Preprocessing and Segmentation: Before each image is segmented it undergoes several preprocessing steps shown in Fig. 5. Each 640 $\times$ 480 px color image is first converted to grayscale and Contrast Limited Adaptive Histogram Equalization (CLAHE) [28] is then applied over the image which is

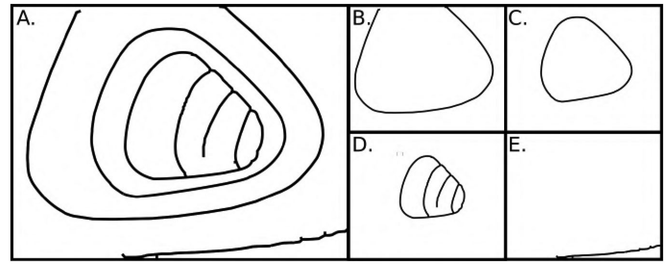


Fig. 6. Image segments are shown for a single segmented image (A). The outermost segment (B) is *b-closed*, the segment (C) has no endpoints and is *closed* while (D) is nearly closed, and is thus a good candidate for becoming *o-closed*, and thus, shape fitting, it does have an endpoint within it, and is thus, initially considered *open*. The last segment in the bottom right corner is *c-crossing* (E).

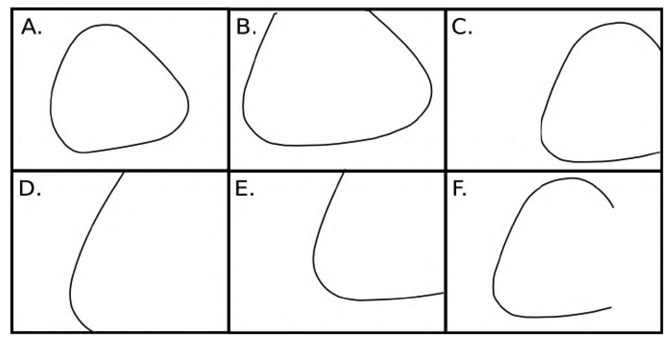


Fig. 7. Different segment types are shown including closed (A), b-closed (B) and (C), b-crossing (D), c-crossing (E), and open (F).

broken into 144 tiles (12 $\times$ 12) for this purpose (see Fig. 5B). A Gaussian blurring filter is then applied to this contrast adjusted image to complete the preprocessing steps. To segment the image, a mean adaptive thresholding method is utilized to determine the appropriate threshold at each pixel using a local neighborhood of size 9 $\times$ 9 px (see Fig. 5C). A morphological closing of the image is performed next using a small square kernel, and then connected components of less than 200 px are removed to reduce the noise in the image. This step is then repeated using a diagonal kernel to attempt to connect some of the remaining large components and remove any residual noise from the image (see Fig. 5D and E). This segmented image is than skeletonized to reduce each segment to 1px in width. Finally, a simple filter is used to identify junctions and endpoints in each skeletonized segment. All image segments are then pruned to remove branches as large as 8 pixels. An additional result of this pruning is that segments without endpoints are identified and flagged as closed curves. Junctions are not currently utilized, but may serve an important purpose in later work similar to [23].

2) Segment Classification: Once segmentation is complete, each image segment is considered independently before being combined with all other segments to finalize lumen center estimates. An example segmented image is shown in Fig. 6 and example segment classes are shown in Fig. 7. All segments are classified into one of several types of segments and based on that classification, different properties of each segment are used to estimate the lumen center. This classification process

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

is particularly important, not only for the different ways these segment types can be used, but also due to the accuracy different types provide. As there is no guarantee that a frame will contain the more accurate segment types (*closed*, *o-closed*, *b-closed*), it is still important to consider estimates from each of the less accurate segments (*b-crossing*, *open*). Thus the goal of classification is to ensure that the final estimate of the lumen center is based on the most useful segments in any given frame, while also ensuring that each segment type is used to the best of its ability. The following describes the process of classifying these segments and is followed by a description of how each segment type is utilized.

1) Closed segments: Segments that form a complete closed curve are designated as *closed* segments (see Fig. 7A). To determine when segments are *closed* an image convolution is done on the skeletonized image. This convolution flags pixels when they occur at a segment ends and thus any segments with no ends are considered to be closed while those that do have ends are considered to be some other segment type. While pruning of the skeletonized image during the preprocessing steps is important here as it allows the system to avoid incorrectly classifying closed segments which may initially have some residual branches and thus some end points, if a closed segment is missed due to a branch, it will typically be caught during the open segment process and recategorized as described below.
2) Boundary closed segments: In general, many segments tend to partially intersect the boundary of the image frame. When a segment intersects the same boundary (e.g., the right side of the frame) at two distinct points, we consider this segment to be a boundary closed or *b-closed segment* (see Fig. 7B and C). The distinct intersection points of these segments with the frame must be a minimal distance away to ensure that they are not simply branching or running along the boundary, thus a threshold distance of 100 pixels separation between intersections points must be met for segments to fall into this category.
3) Boundary crossing segments: When the camera of the Endoculus is very close to a haustral fold it is possible that the segmented fold will cross multiple boundaries of the image frame. We consider these to be boundary crossing or *b-crossing* segments when they run the full width or height of the image (i.e., crossing both the top and bottom or right and left boundaries), and corner crossing or *c-crossing* segments when they cross two adjacent boundaries such as the top and right boundary of the image (see Fig. 7D and E). This designation is applied after ensuring that segments could not otherwise be categorized as closed or b-closed, and thus, a more liberal requirement is placed on these segments such that they will be categorized as *b-crossing* or *c-crossing* if they are within 10 pixels of two distinct image boundaries.
4) Open segments: If a segment does not intersect multiple image boundaries and cannot be designated as *closed* or *b-closed* it is considered to be an *open* segment and some additional processing is required to determine how it will be used (see Fig. 7F). While many *open* segments may be very useful, segments left over from folds that were not well-segmented and thus mostly filtered out, are less useful and may offer poor or misleading center estimates if they are used incorrectly. It is therefore important to separate the more useful/trustworthy *open* segments from noisy/misleading segments. More trustworthy *open* segments come in several forms. If for instance, a segment is nearly *closed* but has been left open (perhaps due to a poor segmentation or because the segment is physically open at a point) it will still provide good estimates. Additionally, if a segment forms a closed shape but was left with residual branches (due to incomplete pruning or inner branching segments) it will also provide good estimates. Several characteristics are considered when deciding if an *open* segment should be considered trustworthy and thus effectively "closed" including the aspect ratio, size, and extrema points.

Aspect ratio is important to consider as it ensures that a segment is not simply a small residual piece of a bad segmentation, while the size (height and width) requirement helps to ensure that the segment is not the result of over segmentation resulting in some odd shaped multi-branching segment. Finally, the eight extreme points of the segment (top-left, top-right, right-top, right bottom etc.) are considered to ensure that a segment is complete enough to allow a useful shape fit. These extreme points (as determined by the segments intersection points with a bounding box) are generally unique for a smooth, nearly closed curve typically generated by the complete segmentation of a well centered haustral fold. For *open* segments, however, these extrema will not be unique when, for example, the segment terminates at the edge of its bounding box (e.g., a segment without a top and right side may only have 4 unique extreme points, left-top, left-bottom, bottom-left, bottom-right). Thus, the number of unique extrema present in the segment can rapidly provide a sense for how many sides of a segment exist, and thus, how closed the segment actually is. Those segments which meet the aspect ratio, size, and extrema requirements are considered to be good candidates for artificially closing via shape fitting (either with an ellipse or a triangle) and thus are designated as *o-closed* segments. If an *open* segment does not meet these requirements or if the shape fit results in a poor fit (as described below) it is simply considered to be *open*.

This process of attempting to close *open* segments is important for several reasons. While simulator images may show many closed curves, *in vivo*, these shapes are less common, and the ability to predict the lumen center without relying fully on *closed* segments is critical. In addition, the segmentation approach described earlier is intended to be conservative with the goal of selecting only the strongest edges in each image while filtering out any smaller/less prevalent segments. This can lead to unintentionally undersegmenting structures that would otherwise appear as closed in an image and results in opening segments that should be *closed*. This process is a way of quickly correcting for this issue without increasing the complexity and computation time of the segmentation.

C. Lumen Center Estimation

Once all segments have been classified, the specific properties of each segment type are used to form an estimate on the camera

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

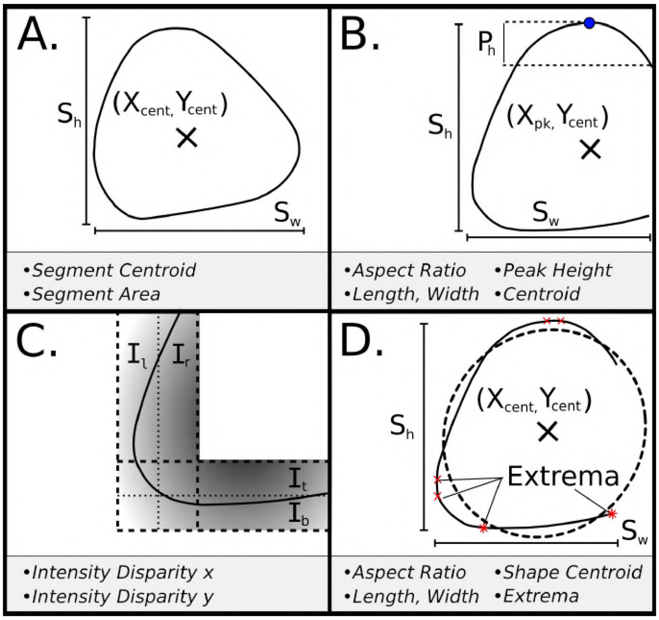


Fig. 8. Each segment class is used slightly differently for predicting the lumen center. An example of each major segment class/type and the respective properties of that segment type are shown here. (A) shows a closed segment, (B) shows a b-closed segment, (C) shows a b-cross segment, and (D) shows an open segment. Note that while a b-cross segment is not shown, the analysis is a simplified version of what is done for the c-cross segment in (C) with only the relevant right/left or top/bottom intensity values used.

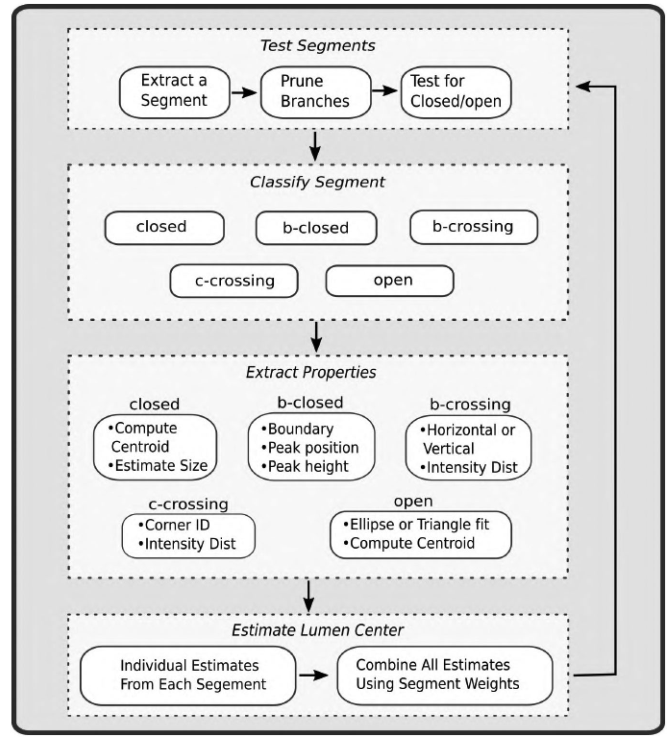


Fig. 9. Block diagram showing the complete region estimation algorithm following the preprocessing of the RGB image. Segmentation is performed on the full image and each segment is then independently classified and used to estimate the lumen center. All segment estimates are then weighted and combined to give the final estimate.

orientation with respect to the lumen. Thus each segments characteristics are used to estimate the camera's center offset X_{est} and Y_{est} from the lumen center whenever possible, (some segments may not be able to provide useful estimates on both degrees of freedom). While there may be some translational components to these offsets, due to the relative size of the Endoculus in the lumen, the ability to translate is considered marginal and thus we evaluate these offsets as purely angular rotation scaled based on image width W and height H and the camera's total field of view (1). These individual estimates will later be weighted and combined based on the relative accuracy of each segment type. Fig. 8 provides a summary of the various properties selected for each segment class and the overall segmentation, classification and center estimation strategy is summarized in Fig. 9. In addition, further descriptions of the algorithms used can be found in the Appendix

$$\alpha = \frac{FOV}{W}, \beta = \frac{FOV}{H}. \tag{1}$$

1) Closed Estimates: Due to the conservative nature of the segmentation process, closed segments are indicative of very well-defined haustral folds. These segments can thus provide very good information about the lumen center location by simply computing the centroid of the segment over all n segment points (x_i, y_i). As shown in (2) and (3), we compare the centroid of the segment to the center pixel of the image $(\frac{W}{2}, \frac{H}{2})$, and the magnitude and sign of this error estimates how far and in what direction the camera is off-center from the lumen

$$X_{cl} = \alpha\left(\frac{W}{2} - \frac{1}{n}\sum_{i=1}^{n} x_i\right) \tag{2}$$

$$Y_{cl} = \beta\left(\frac{H}{2} - \frac{1}{n}\sum_{i=1}^{n} y_i\right). \tag{3}$$

2) B-Closed Estimates: The quality and type of camera orientation estimates provided by boundary intersecting segments is heavily dependent on the specific class of these segments so we will consider them each separately. *B-closed* segments are similar to truly *closed* segments, and thus, also provide accurate information. Preliminary testing within the Kyoto Kagaku simulator showed that when the size, S_{sz} of a *b-closed* segment effectively spanned 80% or more of our frame area I_{sz} such as that shown in Fig. 7B, we could reliably fit a shape to this segment, compute the centroid of that shape and complete the analyzes using (4a), (5a). This, however, represents a minority of these segment types which typically do not fill most of a frame. A more typical *b-closed* segment is shown in Fig. 7C. This right *b-closed* segment (one that intersects the right boundary of the image) is generally indicative of a camera that is facing left of the lumen center, but without some other information it is not possible to say how far left of center the camera is. To improve this estimate we will consider the peak height P_h of the segment. For determining X_{bcl} for segments bounded by the right or left image boundary we consider large peaks to occur when enough of the top of the haustra is visible that the resulting segment begins it's downwards slope before hitting the image boundary. By considering the top most extreme points and the right boundary intersection point and computing the

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

pixel difference, we can determine if the segment reaches a peak and how large that peak is. If this peak height P_h is above a fixed threshold we consider it large enough to make a center prediction in the x direction and will estimate the lumen center to be the x-position of the peak, x_{pk} (considered to be the mean of the top most extrema) (4b). For these segments closed by a vertical boundary such as Fig. 7C, we will then use the y-centroid position to estimate our error in the y direction. For *b-closed* segments intersecting a horizontal, rather than a vertical boundary, we flip this method and estimate the x error from the x-centroid of the segment, and the y error via the peak identification approach described above (5b). Thus, we can apply this method to *b-closed* segments on the top, bottom, left or right of the image

$$X_{bcl} = \begin{cases} \alpha\left(\frac{W}{2} - \frac{1}{n}\sum_{i=1}^{n} x_i\right), & \text{for } \frac{S_{sz}}{I_{sz}} > 0.8 \quad (4a) \\ \alpha\left(\frac{W}{2} - x_{pk}\right), & \text{for } P_h > 20 \quad (4b) \\ 27.5 + \alpha \cdot S_h(0.5 - \frac{S_w}{S_h}), & \text{for } P_h < 20 \quad (4c) \end{cases}$$

$$Y_{bcl} = \begin{cases} \beta\left(\frac{H}{2} - \frac{1}{n}\sum_{i=1}^{n} y_i\right), & \text{for } \frac{S_{sz}}{I_{sz}} > 0.8 \quad (5a) \\ \beta\left(\frac{H}{2} - y_{pk}\right), & \text{for } P_h > 20 \quad (5b) \\ 27.5 + \beta \cdot S_w(0.5 - \frac{S_h}{S_w}), & \text{for } P_h < 20. \quad (5c) \end{cases}$$

If a large peak is *not* present in a *b-closed* segment we cannot use the previously described method to estimate the lumen center. We can, however, make the assumption that the center occurs near or past the intersection of the frame and the *b-closed* segment. This indicates that the camera orientation is offset from the lumen center by at least half of the camera's field of view (FOV). For the Encoculus camera this is an offset of 27.5° or 320 pixels in x and 240 pixels in y. This assumption is important in that while we can no longer estimate the lumen center with as much accuracy as before, we can assert a new boundary on our centeredness and expected error in our estimates and attempt to use additional information within the image to make our estimates. In the absence of a strong peak the aspect ratio of these *b-closed* segments is used to estimate our yaw and pitch angle. Thus for our X offset angle we use the difference of our aspect ratio S_w/S_h with an assumed ideal aspect ratio of 0.5 if the segment were to be perfectly centered on the frame boundary. We weight this aspect ratio error with our segment height and subtract this from our upper centeredness bound (4c), (5c). This estimate makes several unrealistic assumptions about the geometry of our haustral folds including that each fold has a 1:1 aspect ratio, that the height of each fold is constant across its width (i.e., rectangular), however, we accept this simplification as additional noise in our measurement, and rather than attempting to more accurately model the high level of variation which we anticipate, weight these estimates significantly less than our *closed* segments when all estimates are eventually combined.

3) B-Crossing Estimates: Similarly to our peakless *b-closed* segments, *b-crossing* segments, and *c-crossing* segments (shown in Fig. 7D and E) cannot accurately predict our image center entirely on their own. However, with the additional information provided by an image intensity map we can still make useful predictions from these segments. In absence of complete occlusion, our intensity map is immediately useful in determining a very noisy estimate of our camera orientation within the lumen and can thus give us a rapid estimate concerning whether we are facing left or right and top or bottom. This can be quickly assessed by determining the darkest region within the image. We accomplish this by reducing the resolution of our Gaussian filtered image to 20% of full resolution and choose the darkest pixel in this smaller image. We then use this pixel position to determine the darkest region within our original image. If the camera is not occluded, the darkest point of the image will occur toward the lumen center and we can thus make a quick assessment of camera orientation. While this information is useful in biasing our orientation estimates toward one side or the other a more thorough analysis can be used to more finely determine how far off center our camera is oriented. To do this, we examine the intensity disparity that occurs on either side of *b-crossing* segments. Because a *b-crossing* segment may occur when we are very close to a haustra or significantly further away the position of these segments in the image tells us very little about how far off-center we are (i.e., we know nothing about the scale of the full haustra from this segment). Similarly, we cannot assess much from the curvature of these segments, as part of any given segment may have significantly more curvature than other parts and thus may not be a good indicator of how far we are from the segment. While the presence of multiple *b-crossing* segments may serve to provide some of this information we are never guaranteed these additional segments and would prefer to rely on the information each segment provides individually. The change of intensity around a segment ($I_R - I_L$), however, can be indicative of our camera angle with regard to the segment and the lumen center. If for example our camera is perfectly facing the lumen wall, the lighting provided by our device (LEDs on the left and right of the camera) will illuminate both sides of our image equally resulting in a relatively uniform intensity distribution across both sides of our image frame (equivalent to staring at a blank, uniformly lit wall). As the camera begins to turn toward the center, some of our light will now travel down the lumen resulting in a significant intensity disparity across each segment. By considering this intensity disparity we form a very rough estimate of our orientation offset from the lumen center. Although we use *b-crossing* segments as our example here, this method works equally well for *c-crossing* segments if we instead break our image into quadrants rather than halves. Thus, using this intensity disparity method we determine a useful estimate for our center even when no center is visible in our image frame. While these segments have the potential to appear at any camera orientation due to poor segmentation, in practice these segments rarely occur when the camera has a clear view of the lumen. We thus place an additional upper bound on our centeredness estimate of 115 pixels in the x direction or approximately 10° off-center, based on evaluative testing of this method (6), (7), and prescribe a simple linear equation to this intensity method based on initial evaluative testing in the simulated environment

$$X_{bcr} = \begin{cases} 10 + 40(0.5 - \frac{I_R - I_L}{255}), & \text{for left facing} \\ -10 - 40(0.5 - \frac{I_L - I_R}{255}), & \text{for right facing} \end{cases} \quad (6)$$

$$Y_{bcr} = \begin{cases} 10 + 40(0.5 - \frac{I_B - I_T}{255}), & \text{for top facing} \\ -10 - 40(0.5 - \frac{I_T - I_B}{255}), & \text{for bottom facing.} \end{cases} \quad (7)$$

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

4) Open Estimates: Open segments as shown in Fig. 7F can be problematic as they tend to result from a poor segmentation and thus may be inaccurate. To make use of our *open* segments we make the assumption that they could form a closed or nearly closed curve if properly segmented. As noted previously, we filter out any *open* segments with aspect ratios A_r, heights and widths outside of a specific threshold as well as any segments with fewer than six unique extreme points, and then attempt to fit a closed curve to these segments. We first fit an ellipse to our segment [29], however, while ellipse fitting is computationally inexpensive (relative to triangle fitting), for more triangulated haustra this can result in a good fit but an inaccurate reconstruction (an elongated ellipse for example). If our ellipse fit is poor, or results in an ellipse of aspect ratio beyond our threshold, we next fit a triangle to our segment [30]. If this fit results in a triangle where all angles are less than 90°, we consider this to be a good fit for this open segment, and use the centroid of this triangle to form our center estimate. Alternatively if our original ellipse fit is good, we use that centroid to estimate our center. As noted, we will refer to these formerly *open* segments that have been successfully closed by shape fitting as *o-closed* segments.

If our segment does not fit the requirements for shape fitting or shape fitting is unsuccessful, the segment is still considered *open*. Because these segments typically are very incomplete and have the potential to give misleading estimates, we only use the intensity disparity method previously used on *b-crossing* segments to estimate the lumen center from these open segments

$$X_{op} = \begin{cases} \alpha\left(\frac{W}{2} - \frac{1}{n}\sum_{i=1}^{n} x_i\right), & \text{for } o-closed \quad (8a) \\ 10 + 40(1 - \frac{I_R - I_L}{255}), & \text{for } open \quad (8b) \end{cases}$$

$$Y_{op} = \begin{cases} \beta\left(\frac{H}{2} - \frac{1}{n}\sum_{i=1}^{n} y_i\right), & \text{for } o-closed \quad (9a) \\ 10 + 40(1 - \frac{I_B - I_T}{255}), & \text{for } open. \quad (9b) \end{cases}$$

5) Combining Image Segments: Once all segment estimates are compiled, an overall orientation estimate must be calculated based on some combination of the individual estimates. To do this, we consider the general accuracy of each of the described methods as well as the overall size of our most accurate segment types. The presence of any *closed* segments significantly improves the quality of our estimate and thus the votes of *closed* segments when present should be weighted much more heavily than any other segment. In addition, *o-closed* segments while potentially less accurate than *closed* segments are also very trustworthy given the stringent constraints we ultimately place on them. We, thus, combine *closed* and *o-closed* segment estimates for our weighting. *B-closed* segments that have been successfully fitted to a shape or that have strong peaks are also very accurate since the lumen center is still estimated within the image itself so these will also be weighted at the same level as our *closed* segments. For these more accurate segments, we prescribe 90% of our overall estimate weight (if some of these segments are present). In addition, because our goal in navigation is to identify the lumen center as close to the Endoculus as possible (*i.e.*, *not* far down the lumen), we weight the estimates from those segments which have the greatest overall area and are thus assumed to be the closest to the Endoculus with 80% of their respective weight. While the final weighting will always be dependent on the number and existence of each segment type, this distribution ensures that when large, accurate segments are present they will have a disproportionate vote on the lumen center position

$$\begin{aligned} X_{est} &= \lambda_1 \left[0.8 \cdot X_{cl,1} + \frac{0.2}{n_{cl}-1}\sum_{i=2}^{n_{cl}} X_{cl,i}\right] \\ &+ \lambda_2 \left[0.8 \cdot X_{bcl,1} + \frac{0.2}{n_{bcl}-1}\sum_{i=2}^{n_{bcl}} X_{bcl,i}\right] \\ &+ \frac{\lambda_3}{n_{bcr}}\sum_{i=1}^{n_{bcr}} X_{bcr,i} + \frac{\lambda_4}{n_{op}}\sum_{i=1}^{n_{op}} X_{op,i} \end{aligned} \quad (10)$$

B-crossing, *c-crossing,* and *open* segments are our least accurate segment types, so when these are present along with other more accurate segments we will give them only 10% of the overall estimation power and in contrast to our more accurate segments, this distribution is not size dependent. Thus the presence of a single *closed*, *b-closed*, or *o-closed* segment can significantly bias our final estimate toward that curves independent estimate, however, our device will still be able to form useful predictions even when oriented significantly off of the lumen's center to the point of almost completely facing a wall. The overall weighting function is shown in (10). Note that each class weight, λ is dependent on the number of segments that occur.

The importance of the method described is that it provides the ability for our device to navigate and/or make decisions based on its estimated orientation even when it is not facing the center of the lumen. This is important for several reasons. When navigating tight turns, it is likely the device will always be primarily wall-facing. This system provides the ability to recognize when we are in such a state and navigate accordingly. Additionally, external disturbances such as a tether snag for example can easily result in large changes to the robots pose which we should be able to recover from, this system provides the ability to recognize when the device is no longer well-centered and to respond accordingly. Finally, endoscopy is not primarily about navigation but rather visualization/intervention. It is thus very important to provide the ability for any robotic endoscope to seamlessly transition from visualizing the tissue to navigating to the next section of tissue that much be visualized. This system provides this ability.

6) Finite State Machine: To more effectively utilize this region estimator and to avoid the problem of spurious estimates, we also incorporate state dependence into this system as shown in Fig. 10. To do this, we break our pose estimates into six discrete regions, well-centered, off-center, left-wall, left-occluded, right-wall, and right-occluded. Note that we do not include any pitching states in this current formulation even though we do include pitch in our region estimates. The Endoculus, our current robotic endoscope does not need to control pitch to maneuver successfully and thus we do not include these estimates in our relevant navigation states, however, these states would not be difficult to add for use on more conventional scopes. We define our inner four (nonoccluded) states based on the estimated angle

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

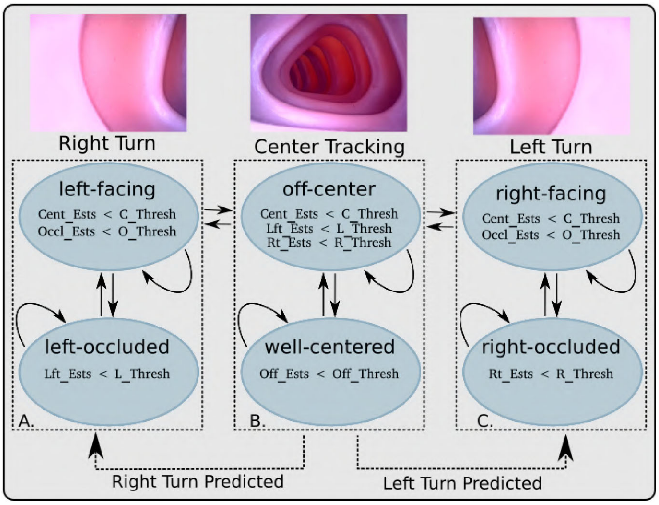


Fig. 10. Overview of the state machine. Each transition to a new state is triggered when a fixed threshold of images representing that new state are seen consecutively. The dotted arrows indicate state transitions induced by the imminent turn predictor, rather than than the region estimator.

from our region estimator and based on the previous frames processed. The device will begin in the well-centered state and remain in that state until its lumen center estimate error is greater than 10° for a fixed threshold of consecutive frames. Once this threshold is reached, the system moves to the off-center state where it will again remain until an error and frame threshold is reached toward either the left or right wall direction. While in some regard the state dependency of this system, amounts to low pass filtering our region estimates (as we require multiple adjacent votes in the same direction before a transition can be made) this state dependence also eliminates the possibility of erroneously moving between nonadjacent states (skipping from left to right for example), instead forcing the system to move only between states that are adjacent to one another. Once we have entered the right-wall or left-wall state it is possible that the camera will become so close to the wall of the lumen that useful estimates will not be possible. To manage these wall occlusions, we introduce left and right-occluded states. These states are reached if, while in the left or right-wall state, the system is unable to find any useful segments for a fixed threshold of frames.

D. Control

One advantage of the state-dependence of this system is that it enables the implementation of multiple state-dependent controllers to better accommodate the navigational needs of an endoscopic device. For the purposes of robotic capsule endoscopy the lumen region and center estimation method presented above is an important step toward quickly interpreting the *in vivo* scene for both low level control and higher level decision making. While lumen center tracking strategies such as [22] demonstrate success when the lumen is clearly seen, the tight turns of the GI tract guarantee that such clear views will not always be possible. To explore the potential to use this algorithm for control purposes a multi-part lumen center controller was designed. This controller combines lumen center tracking with a tight curve navigation system to allow the Endoculus to autonomously navigate straight paths, large curves and tight curves within the Kyoto Kagaku colon simulator. The state machine thus allows us to clearly delineate between these separate control modes and to serve as a means of determining when the controller must transition to each different mode

$$E = F_{\text{center}} - Est_{\text{center}} \tag{11}$$

$$\omega_r = \begin{cases} |E| \cdot D_{PID}, & \text{for } |E| > 30 \\ \omega_{max} - E \cdot K_P, & \text{for } 0 < E < 30 \\ \omega_{max}, & \text{for } -30 > E < 0 \\ \omega_{max}, & \text{for right facing state} \\ -\omega_{max}, & \text{for left facing state} \end{cases} \tag{12}$$

$$\omega_l = \begin{cases} |E| \cdot D_{PID}, & \text{for } |E| > 30 \\ \omega_{max}, & \text{for } 0 < E < 30 \\ \omega_{max} - E \cdot K_P, & \text{for } -30 > E < 0 \\ -\omega_{max}, & \text{for right facing state} \\ \omega_{max}, & \text{for left facing state} \end{cases} . \tag{13}$$

1) Center Tracking: In general, lumen center alignment of the Endoculus is an important goal. While we may not always wish to be in the center of the lumen (when closely inspecting a lesion on the colon wall for example), the ability to become centered in the lumen is critical because it enables both superior mobility (compared to rubbing against the wall, or driving directly into it) and superior visualization (enabling both better diagnostic capabilities and better estimation for future navigation goals). Lumen center tracking is thus an important component of any robotic endoscope control strategy. To control lumen center position, we feedback our lumen center estimate and our region estimate from the visual estimator to compute the control mode and our error (11). The region estimate serves as our outer loop, moving us between several possible states. When the Endoculus is already in the well-centered state the controller is set to align the onboard camera (and thus, the heading of the Endoculus) with the estimated center. This is accomplished with a simple dual mode PID controller similar to that used in [22] and [31] as shown by the first three cases in (12) and (13). Thus, this controller primarily operates in the two centered states as indicated in Fig. 10B. If the Endoculus transitions into one of the left or right-wall states our lumen center predictions are significantly noiser (due to lack of closed curves and our reliance on intensity disparity). Thus, rather than simply feeding back these noisy estimates, we instead allow the device to stop and attempt to turn toward center until reaching the well-centered state as indicated by the last two cases in (12) and (13). These different modes of operation all determine how the right and left motor speeds ω_r, ω_l of the Endoculus are set during the center tracking mode of operation.

2) Imminent Turn Prediction: Another useful component of this state-dependent architecture is that it enables the distinction of appropriate states in which to make predictions/estimates, providing greater confidence that these estimates are accurate.

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

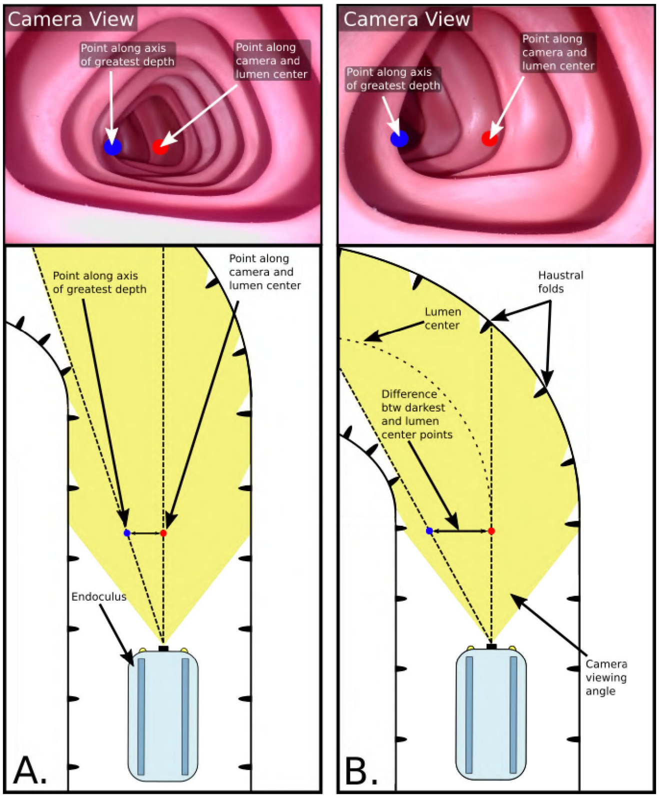


Fig. 11. As the onboard camera approaches a turn (A), the difference between the center as estimated by the haustra and the center as estimated by the deepest or darkest point will become more pronounced as shown in (B). This is used to trigger the turn control algorithm.

One application of this is in anticipating upcoming turns. While the camera is in the well-centered state, the expectation is that the camera has a good view of the upcoming lumen. As shown in Fig. 11, while long straight sections of the lumen will tend to show their darkest region P_{dark} centered with the upcoming haustral folds, as the camera approaches a turn, the dark region tends to move away from the estimated center (provided by the region estimator) Est_{cent} with this off-center distance increasing as the camera moves closer to the turn. Using the difference in these points, the Endoculus, while in the well-centered state, is able to predict when it is approaching an upcoming turn. As the Endoculus approaches a turn and this difference between P_{dark} and Est_{cent} passes a fixed pixel threshold for five consecutive frames (14), the controller switches to a turn anticipation mode in which it and will no longer attempt to track the center, but will rather begin to steer toward the outside of the upcoming turn to create more space for the device to maneuver. As shown by the dotted arrows transitioning to the outer states in Fig. 10A and C, the center tracking controller switches modes and imposes a temporary switch to our left or right facing state until the device has found the lumen wall (as indicated by successfully occluding the camera). This adjustment allows the controller to seamlessly switch into the tight turn control mode just before reaching the turn. This is necessary because once the device has reached the beginning of the turn it will no longer be able to reach the well-centered state and make useful predictions. In contrast to the center tracking control mode, this is a temporary open-loop control mode of operation for the device and the Endoculus will remain in this mode (driving slightly toward the outer wall) until it either reaches the wall occluded state and transitions into the turn navigation control mode that follows, or transitions to the well-centered state, indicating the turn prediction was incorrect. If this occurs the device will simply continue with the center-tracking control described previously

$$Turn = \begin{cases} Left, & \text{for } Est_{\text{cent}} - P_{\text{dark}} > 110\text{px} \\ Right, & \text{for } Est_{\text{cent}} - P_{\text{dark}} < -110\text{px} \end{cases}. \quad (14)$$

3) Turn Navigation: When the Endoculus is navigating tight turns, the length of the device, width of the lumen and the radius of the turn all limit the degree to which the device can face the center while still remaining mobile, and thus, mobility is more effectively served by driving toward the outside of the turn (toward the lumen wall) and pivoting the device right before reaching the wall. These tight turns require an alteration of the more intuitive center-tracking navigation strategy, where rather than attempting to center the camera in the lumen, the goal becomes one of tracking the outer wall of the lumen. To accomplish this, the region estimator along with the turn prediction mode described above is used to determine whether the device has entered the right or left turn mode. The Endoculus will then drive forward until occlusion occurs and will then attempt to turn left or right (dependent on the mode) until it reaches the right/left-wall state. Once this occurs the device will again attempt to drive forward until reaching the right/left-occluded state. This process will continue until the device successfully transitions into the off-center state at which point the controller mode will switch back to the center-tracking mode of operation. If at any point the device is unable to maneuver out of the occluded state (i.e., it cannot successfully turn right or left) for a set threshold of time, the Endoculus can automatically turn on insufflation (if enabled by the operator) for several seconds in an attempt to create more space as is currently done by endoscopists during conventional colonoscopy.

E. System Validation

To validate the utility of this visual estimation and control strategy, a series of benchtop experiments were conducted on the Endoculus as shown in Fig. 12. The state-dependent region estimator was first tested for its ability to successfully identify the current region in the camera view at various rotational speeds, as well as its ability to maintain correct predictions under the presence of large up and down vibration that could introduce noise into the initial image segmentation. In addition each of the control modes were tested independently to validate functionality. Finally, the Endoculus system was used in a series of navigational tasks combining all three control modes to demonstrate the effectiveness of the overall system. All of these tests were conducted within the Kyoto Kagaku Colonoscope Training Model.

To evaluate the center estimation strategy and the state-dependent region estimator overall, the Endoculus was tested for its ability to quickly and accurately estimate the lumen center

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

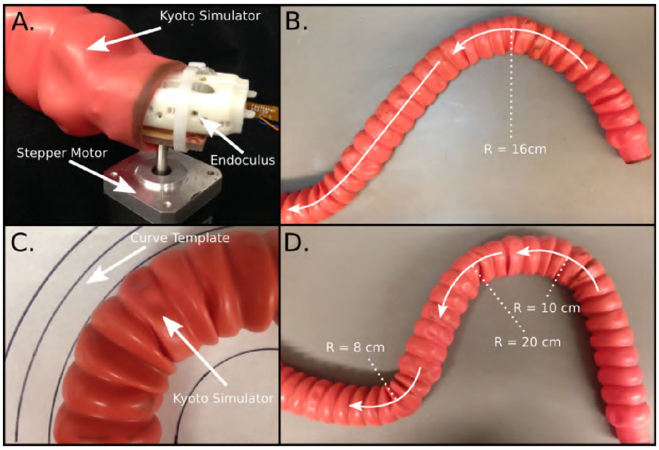


Fig. 12. Validating experiments are shown including the regions estimator test setup using the Endoculus mounted to a stepper motor while viewing the inside of the Kyoto Kagaku colon simulator (A), A curve test about a 12 cm radius within the simulator (C) and two combined tests showing both straight and curve sections (B), (D).

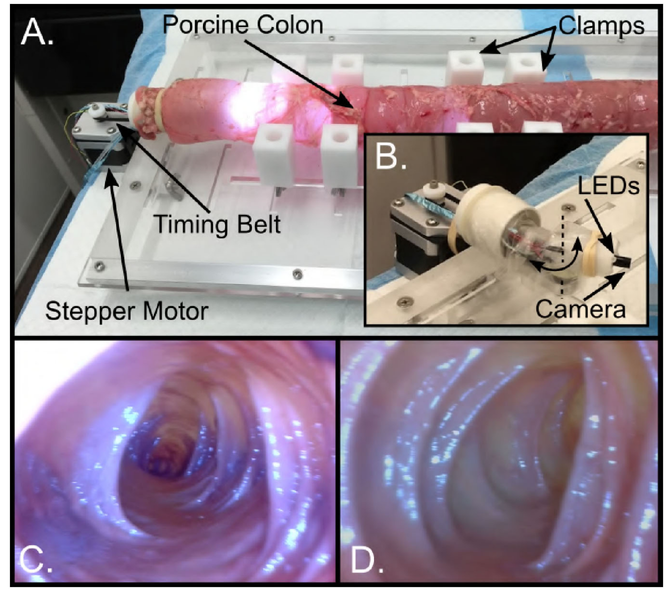


Fig. 13. *Ex vivo* test fixture is shown with excised tissue mounted (A). The rotating camera chassis with LED's, insufflation, and stepper motor can be seen in (B). Two images taken using this fixture within the tissue are shown in (C) and (D).

and to switch between states over a range of rotational speeds. The Endoculus was mounted in-line with the rotor of a stepper motor. This motor has a resolution of 200 steps/revolution, but was controlled with microstepping at 4 microsteps/step for a final resolution of 800 microsteps/revolution. The motor was used to yaw the Endoculus from left to right and right to left over a range of different speeds as the Endoculus viewed the inside of the Kyoto Kagaku simulator. The step position of the motor was directly controlled using the same system running the estimator and thus ground truth angular position, center estimates, and system states were all recorded in real-time. Using this data, the success of the region estimator's lumen center predictions and state transitions were evaluated under different amplitudes and angular velocities.

Center tracking was tested in straight sections of the simulator. The device was placed in the center of the elliptical and triangular sections of the Kyoto Kagaku simulator and allowed to drive the complete length of these respective sections (a distance of approximately 43 cm for the elliptical and 63 cm for the triangular, based on the initial starting/ending position of the device) while being timed and evaluated for wall collisions as determined by a wall-occluded camera. Ten trials were run in each section. During these experiments, the tether was managed by the experimenter to avoid snags that would disrupt the motion of the device.

Turn prediction was evaluated by first measuring the distance at which the Endoculus would predict a turn both when static, manually driven and autonomously controlled via center tracking. Several turns of different radii of curvature and direction were tested. Following these preliminary tests, the turn prediction method was tested for its ability to transition into the turn control mode and successfully navigate turns. These tests were done first manually by driving the Endoculus up to an impending turn and then with the addition of the center-tracking algorithm to confirm that the Endoculus would properly transition from one control state to the other. Again, several different turn radii were tested to ensure proper functionality and the success or failure of this state was determined by whether or not the upcoming turn was anticipated correctly. In addition any false positives that occurred while navigating a straight section of the simulator were recorded.

The turn control modes of operation were evaluated on both right and left turns of varying radii. These were tested both manually by triggering the control mode and with the addition of the previous control modes. Success was determined by whether or not the device was able to successfully maneuver through the turn without intervention by the operator and reach the off-center and eventually well-centered control states. The time needed to navigate each turn and any instances of intervention were recorded.

Finally, the complete system was evaluated in several geometric configurations of the simulator for its ability to navigate between straight and curved sections. Each test was timed and the configuration geometry measured and any relevant observations or necessary user interventions were recorded.

F. Ex Vivo *Tissue Experimentation*

Although the Kyoto Kagaku Colon Simulator offers a convenient benchtop testing environment for estimator validation, it was important to evaluate the effectiveness of the yaw estimation strategy within actual tissue. A simple platform was designed and fabricated for this purpose, consisting of a tissue mounting plate, a stepper motor and driver, and a simple camera/insufflation/LED chassis (see Fig. 13). This platform allowed for airtight mounting of excised colon tissue around the rotating chassis such that the tissue could be insufflated. The camera chassis was designed to rotate about the same center position as the Endoculus and the LED's used for lighting were identical to those used on the Endoculus. A Tic T500 Stepper

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

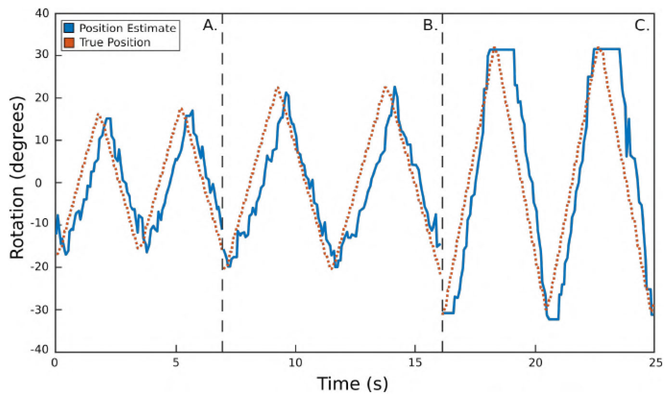


Fig. 14. Raw center estimates are compared to ground truth angular position as determined by stepper motor position during fixed velocity sweeps. Intermediate rotation speeds are possible with good tracking when the total sweep angle is less than 25° (A), (B). Over larger angular sweeps (30°) occlusion begins to occur and the system is unable to estimate angular position (C). Clipping in (C) is the result of using the last known good position when no estimate can be made.

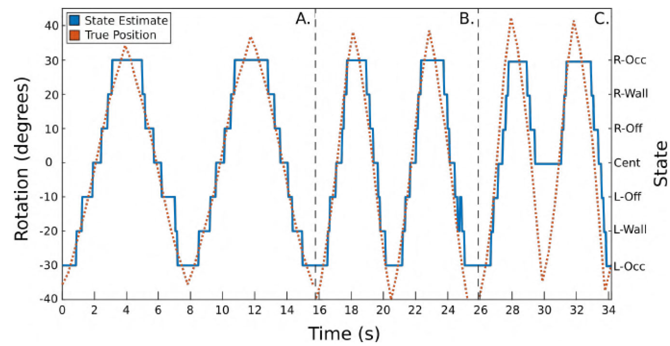


Fig. 15. State transitions are shown (right axis) and compared against ground truth position (left axis) for three real-time experiments. Transitions are accurate for intermediate frequency rotations such as sweep times of 2 seconds shown in (A), however, some errors do begin to occur at higher frequency sweep times such as the 1.1 second sweep shown in (B). Beyond this point, the state machine cannot properly transition and is unable to correctly track the motion as shown in the 1 second sweep (C).

motor controller was used (Tic T500, Pololu Inc, Las Vegas, NV) to control the yaw position of the chassis. The stepper motor itself has a step resolution of 200 steps/revolution and was operated using 1/8 microsteps for a total of 1600 steps/revolution and an approximate max resolution of $0.22°$. The motor itself was connected to a timing pulley/belt for rotating the camera chassis which is mounted directly to a timing pulley of the same size (1;1 gear ratio) about a shaft fixed within an aluminum cantilever. This cantilever design allows for the camera to be inserted into the tissue, while a thin and flexible latex boot seals around the camera chassis and cantilever such that proper insufflation can occur. The insufflation line itself runs directly through the camera chassis along with the LED's and camera. The camera used is identical to that used onboard the Endoculus.

Using this fixture, a series of experiments were performed to determine the effectiveness of the region estimator at accurately estimation the relative yaw of the camera chassis with regard to the external tissue. Porcine colon tissue was used in these experiments. The tissue was first mounted around the cantilever and secured about a mounting tube. The tissue was then oriented such that it aligned with the centerline of the platform. The stepper motor and camera were then controlled over a total sweep range of $60°$, while the region estimator operated in real-time and image and yaw position data were recorded. This was repeated in several positions throughout the tissue and with the tissue in several different orientations to determine any impact this might have on the region estimation strategy. While LED adjustment to ensure proper lighting was performed prior to each experiment, no attempts to alter or tune the region estimation algorithm were done, so as to truly assess the applicability of this method (as used within the simulated environment) for tissue navigation.

III. Results

A. Region Estimator Evaluation

It was found that while center estimates were accurate for offsets up to $25°$ (see Fig. 14A,B), right or left, wall occlusion would result in failed estimates at rotational angles greater than $30°$ (see Fig. 14C). Slow, lower velocity rotation also had a tendency to produce noisy estimates as shown in Fig. 14A,B. This appeared to be due to the ability of segments which had been classified as one of the more trustworthy types, but which existed on the margin of that type, to switch classes to a less trustworthy type and alter the overall estimate. Despite these errors, the Region Estimator performed well with rotations up to $40°$ offset from center, however, it was limited by rotational velocity with performance suffering somewhat for rotational velocities greater than $40°$/s (see Fig. 15). While slow and intermediate sweep speeds showed excellent tracking of the ground truth rotation angle, at higher rotational velocities greater than $45°$/s the system was unable to fully transition between all states (see Fig. 15C). While this limits the ability of this system to accurately track states at higher speeds, further evaluative testing of other components of the system did not appear to be impeded by this limitation. This result is likely due to the physical limitations of the Endoculus within the simulator which do not allow it to reach high yaw rotation speeds that might induce these errors during normal operation. Large vibrations also did not appear to impact the state transitions and the Endoculus successfully maintained correct region estimation even under heavy vibration within the simulator. It should be noted that while the necessity of the state machine to accumulate multiple frames toward one direction before transitioning to a new state certainly impacts the response time of this system, this filtering effect also allowed for accurate estimation of the device rotation far outside of the raw center estimates.

B. Center Tracking Controller

The center tracking system for the Endoculus proved very successful as shown in Table I. Over the course of ten tests in both elliptical and triangular sections of the simulator, no wall collisions occurred, no intervention was required and the system achieved a mean completion time of only 16.7 s and an average velocity of 2.61 cm/s in the elliptical section and 30.2 s with an average velocity of 2.16 cm/s in the longer triangular section.

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

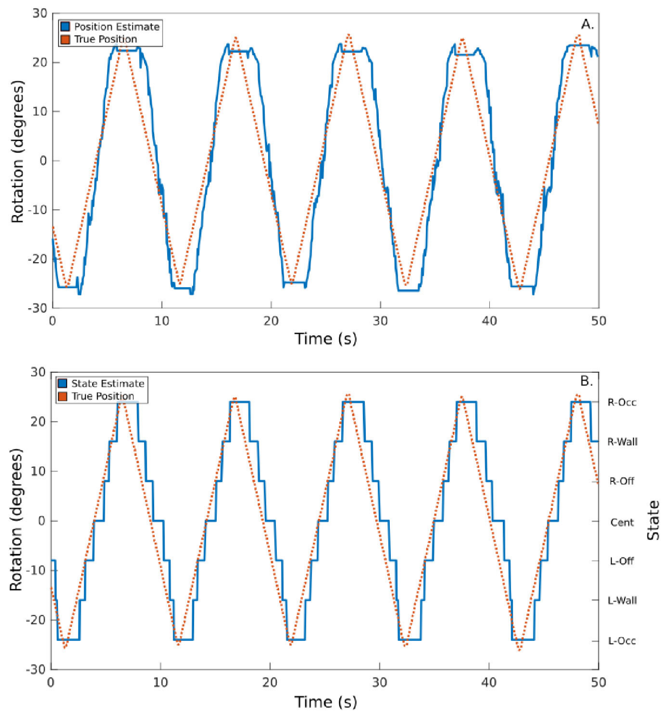


Fig. 16. Results from the *ex vivo* tissue experiments are shown including the raw yaw data and the region estimator data. In general this system showed acceptable tracking in the tissue environment for offsets of up to 25° from center, at which point the camera had been fully occluded by the tissue wall.

TABLE I
TIME TRIALS IN STRAIGHT SIMULATOR

Section	Time (s)	Std (s)	Speed (cm/s)	Std (cm/s)
Elliptical[a]	16.71	1.63	2.61	0.28
Triangular[b]	30.19	5.30	2.16	0.38

[a] Elliptical section length is 43 cm
[b] Triangular section length is 63 cm

C. Turn Prediction

The turn prediction strategy presented also demonstrated significant potential for autonomously predicting impending turns and their direction as well as for transitioning from manual or autonomous center-tracking into the turn control mode of operation. The turn prediction method successfully anticipated the direction of all turns of 5 cm $<$ radius $<$ 20 cm consistently at a distance of approximately 10 cm as measured from the front of the Endoculus to the start of the turn. For turns of $radius < 5$ cm the turn essentially appeared closed to the onboard camera. This presents a significantly noisier darkest/deepest point estimate and thus the Endoculus could not predict a turn below this radius. For turns of $radius > 20$ cm the tissue appears straight enough to the camera that the offset threshold is never met. In these instances the Endoculus is able to continue driving using center tracking without the need for turn navigation.

No false positive occurred during manual or autonomous center-tracking in any of the tests. In addition, the Endoculus was able to correctly transition to the turn control state in nearly all of the ten fully autonomous trials, save one attempt in which the device was unable to continue driving forward toward the wall and reach the occluded state (due to a snagging of the device's tether attachment).

TABLE II
TIME TRIALS IN CURVE SECTIONS

Turn Radius (cm)	Time (s)	Distance (cm)	Speed (cm/s)
18.0	62.05	31.4	0.51
16.0	148.49	28.3	0.19
14.0	98.02	25.1	0.26
12.0	22.27	22.0	0.99
10.0	49.98	18.8	0.38

TABLE III
FULL SYSTEM VALIDATION TESTS

Configuration radii (cm)	Time (s)	Notes
left 14, right 14	101.3	No intervention
left 10, right 12	122.4	No intervention
left 8, left 10	241.5	Tether snagging
right 14, left 10	145.4	Missed turn, snagging
right 8, right 10, left 14	198.2	Tether snagging
right 8, right 10	256.3	Tether snagging

D. Turn Navigation

The turn navigation approach used in this study proved to be very effective with the ability to navigate turns as tight as 10 cm radius of curvature as shown in Table II. The traverse times around these turns varied significantly and tether snagging had the potential to significantly impede device mobility around the tighter turns. At the 18 cm turn radius, the device primarily relied on center tracking throughout the length of the turn and at the 16 cm radius, the device struggled due to a constant switching back and forth between the two control modes. This particular case was unexpected and resulted in a much longer traverse time, however, future iterations of this method will account for this discontinuity more effectively. For the sharper turn validation tests this method proved to be very effective up to the smallest radius tested at 10 cm, and an even sharper 8 cm turn was navigated successfully during the full system validation test.

E. Full System Validation

The results of the full system validation tests can be seen in Table III. The Endoculus navigation system as presented here proved successful at navigating a host of different geometric configurations. Several of these are shown with traverse times and any necessary interventions indicated. In general, center-tracking, turn-prediction and turn-control proved very successful during these full length multiturn tests, however, the device was sometimes impeded due to tether snagging as it progressed further and around multiple turns. One additional issue occurred multiple times in which immediately after exiting a turn the device would face a turn of the opposite direction. If the distance between these turns was not large enough ($>$ 15 cm) the Endoculus would not have enough time (due to its forward speed) to transition back to the center-tracking controller and

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

then into the opposite turn control mode before reaching the new turn. This is primarily due to the conservative threshold placed on the controller transitions which limits the speed at which they can occur (designed to prevent incorrect mode switching), however, one solution to this problem may simply be the addition of a turn exit mode at which point the device when reaching the well-centered state, pauses to assess any upcoming turns/straight sections before moving on. Despite these mobility issues, the system generally demonstrated successful autonomous lumen center estimation in real-time (frame rate of approximately 25 fps on 7th Gen Core I5 processor with 8 GB of RAM).

F. Ex Vivo Tissue Experimentation

Despite no attempts at tuning the region estimator for this somewhat new environment, the estimator itself demonstrated excellent tracking and yaw estimation within the *ex vivo* tissue as shown in Fig. 16. This was true across a range of illumination levels and tissue orientations (upcoming right/left turns) and demonstrates exceptional robustness of this strategy overall for right/left angular offsets of up to $25°$. In addition, the state transitions likewise demonstrated excellent tracking. While some saturation did occur as the camera entered into the occluded state on the far left or right positions, it must be noted that the pig colon being used was 5% to 10% (depending on the section being tested, smaller than the very uniform Kyoto Kagaku simulator. With this in mind, it is not surprising that the camera reached this occluded state slightly earlier than was observed within the benchtop simulator. Despite this saturation point, the state-dependence of the system allowed for uninterrupted and accurate state transitions even following these extreme occlusions, further demonstrating the utility of this system for future *in vivo* application.

IV. Discussion

While issues with tether snagging did hinder device navigation at times, the estimation and control strategy presented here proved successful at navigating the Endoculus through a host of different geometric configurations, including both tight turns and long straight sections. In addition, the yaw estimation strategy used proved successful both in the simulated benchtop colon, and in *ex vivo* tissue, demonstrating the potential of this system for future *in vivo* operation.

In general it is a tedious if not impossible process to explicitly describe the impact of specific visual cues in a way that enables decision making for a robotic device, however, while tissue properties, (size, color, health etc.) may vary significantly between patients, the very limited number of macro features within the colon significantly reduces the necessary complexity of such a system. While early work by Khan and Gillies[23] demonstrated the importance of utilizing these large scale features at the time this system was computationally prohibitive. In the experience of the authors, the use of the Haustral fold contours to improve immediate heading adjustment allows for a substantial improvement over darkest/deepest point techniques. This is particularly true when approaching turns, in which the difference between the darkest point and the estimated center from the contours has served as a very useful predictor for identifying both turn proximity and direction.

While it is likely that more data-driven approaches (e.g., those that rely on deep learning [32], [33]) to region classification will also be fruitful, the work presented here provides an explainable and intuitive framework for interpreting the major large scale visual features present within the colon. This may serve to inform a host of other approaches while also allowing for the inclusion of other visual cues that may further improve these estimates in the future.

In addition to the intuitive nature of the region estimation/classification strategy, the state dependency of this system provides a simple but important method for achieving multimodal estimation and control. In some ways this structure points to the difficult environment that the colon represents in that one must accept that camera occlusion occurs often and any system dependent on a clear view of the lumen will ultimately prove unsuccessful in many if not a majority of instances during a procedure. The architecture presented here has built this occlusion assumption into the overall strategy, only seeking to make lumen centering predictions when a clear view of the lumen is recognized, while also anticipating turn direction and haustra structure identification to improve future control decisions when clear views will almost certainly be unavailable. While this does not amount to true path planning, these predictions do allow us to operate even around tight turns under nearly complete occlusion by the lumen wall. This multimodal approach may prove to be critical for autonomous operation in the difficult *in vivo* environment, and while we have demonstrated only six device states and three tissue states (for imminent turn prediction), a host of other discrete states may also be utilized in future work. Distinguishing peristalsis and fluid bubbles (both also potential sources of occlusion) from wall occlusions will likely be a very important addition to this system. While these additional states were not explored in this work it is likely that additional functionality will be included in future work to enable more robust operation *in vivo*. The current state dependent system should allow for the easy addition of these states and/or substates to our system as we begin to implement methods for estimating when these different forms of occlusion may have occurred.

Additionally, although the current iteration of this system utilizes a simple finite state machine for combining multiple center estimates, it is likely that more sophisticated probabilistic estimation methods for combining the raw lumen center estimates may offer improved performance within this same general framework given here.

V. Conclusion

The navigation architecture presented in this work proved effective at autonomously navigating the Endoculus system through the Kyoto Kagaku simulator in real-time (25 fps) and showed exceptional ability in estimating the visual regions both in a simulated and *ex vivo* environment. The Endoculus was able to navigate straight sections of the simulated environment at speeds up to 2.61 cm/s and sharp turns even under significant wall-occlusion. A variety of control strategies could be built

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

on this same region estimator and it is expected that additional work will go toward designing more robust control strategies for handling the tight turns of the colon in the future.

While control/navigation experiments using the estimator have not yet been conducted *in vivo*, the ability for the yaw and region estimator to operate robustly even within the tissue environment indicates that given effective mobility, autonomous navigation within tissue utilizing this approach will be possible, likely only requiring simple changes to the controller gains without any significant changes to the more complicated estimation strategy.

It is expected that additional work toward mapping the GI environment will substantially aid this system in predicting upcoming tissue geometry and planning navigation and control strategies to respond to this geometry. Accurate turn anticipation was a difficult thing to achieve for this device, due to a lack of depth information and the visual similarity between a simple wall facing camera and an upcoming turn facing camera. Future work using simultaneous localization and mapping methods to provide 3-D tissue geometry information will significantly aid in addressing this challenge and may allow for the prediction of the distance to upcoming turns as well as the specific geometry of those turns, allowing for true path planning for a robotic endoscope.

APPENDIX

Algorithm 1: Algorithm for Lumen Center Prediction.

```
Input: Raw RGB Image
Output: Center Estimate
Initialization :
Preprocess Image
Segment Skeletonize
Find Endpoints
Loop through all remaining image segments
for i = 0 to Number of Segments do
  if (noEndpoints) then
    Process as closed segment
  else
    Find boundary intersections
    if Multiple Same Boundary Intersections then
      Process as b-closed segment
    else if Opposite Boundaries Intersections then
      Process as b-cross segment
    else if Adjacent Boundaries Intersections then
      Process as c-cross segment
    else
      Process as open segment
    end if
  end if
  Append to list of estimates and properties
end for
Combine all estimates
returnFinal estimate in x and y directions
```

Algorithm 2: Lumen Center Estimate From Closed Segment.

```
Input: Image Segment
Output: Center Estimate and Properties
Compute size of segment
Find x and y centroid of segment in pixels
Convert to angular offsets
returnAngular offset estimates and size
```

Algorithm 3: Lumen Center Estimate From b-closed Segment.

```
Input: Image Segment
Output: Center Estimate and Properties
Compute size of segment
if segment size > 80% of image then
  if Triangular region then
    Fit triangle to segment
    Find x and y centroid of segment in pixels
  else
    Fit an ellipse to segment
    Find x and y centroid of segment in pixels
  end if
else if Segment is closed on right or left then
  Compute peak height in y direction
  if Peak height > 20 pixels then
    x estimate is x position of peak
    y estimate is y centroid of segment
  else
    if Bounded by left side of frame then
      x estimate = 0 − S_h · (0.5 − S_w/S_h)
      y estimate is x centorid of segment
    else if Bounded by right side of frame then
      x estimate= I_w + S_h · (0.5 − S_w/S_h)
      y estimate is y centroid of segment
    end if
  end if
else if Segment is closed on top or bottom then
  Compute peak height in x direction
  if Peak height > 20 pixels then
    y estimate is y position of peak
    x position is x centroid of segment
  else
    if Bounded by left top of frame then
      y estimate = 0 − S_w · (0.5 − S_h/S_w)
      x estimate is x centroid of segment
    else if Bounded by bottom of frame then
      y estimate = I_h + S_w · (0.5 − S_h/S_w)
      x estimate is x centroid of segment
    end if
  end if
end if
Convert estimates to angular offsets
returnAngular Offset estimates, type, size, shapes fit
  (if applicable)
```

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Algorithm 4: Lumen Center Estimate From b-Crossing Segment.

Input: Image Segment, Intensity Image
Output: Center Estimate and Properties

if Vertical Segment **then**
  x estimate from intensity difference across segment
else if Horizontal Segment **then**
  y estimate from intensity difference across segment
end if
Convert estimate to angular offset
returnx OR y angular offset estimate

Algorithm 5: Lumen Center Estimate From c-Crossing Segment.

Input: Image Segment, Intensity Image
Output: Center Estimate and Properties

Find segment aspect ratio
if segment is 70% vertical **then**
  Classify as "Vertical"
else if segment is 70% horizontal **then**
  Classify as "Horizontal"
else
  Classify as both "Horizontal and Vertical"
end if
if Vertical Segment **then**
  x estimate from intensity difference across segment
else if Horizontal Segment **then**
  y estimate from intensity difference across segment
else if Horizontal and Vertical **then**
  x estimate from intensity difference across segment
  y estimate from intensity difference across segment
end if
Convert estimate(s) to angular offset(s)
returnx, y OR both angular offset estimates

Algorithm 6: Lumen Center Estimate From Open Segment.

Input: Image Segment, Intensity Image
Output: Center Estimate and Properties

Compute size of segment
Compute aspect ratio (A_r)
Determine number of unique extrema
if $|1 - A_r| > 0.25$ or size < 0.2 or # extrema < 6 **then**
  x estimate from intensity difference across segment
  y estimate from intensity difference across segment
else
  re-classify as o-closed
  if Triangular region **then**
    Fit triangle to segment
    Find x and y centroid of fitted shape in pixels
  else
    Fit an ellipse to segment
    Find x and y centroid of fitted shape in pixels
  end if
end if
Convert estimates to angular offsets
returnAngular Offset estimates, type, size, shapes fit (if applicable)

ACKNOWLEDGMENT

The authors would like to thank K. Johannes for his assistance in conducting the ex vivo tissue experiments presented in this work.

REFERENCES

[1] G. A. Formosa, J. M. Prendergast, S. A. Edmundowicz, and M. E. Rentschler, "Novel optimization-based design and surgical evaluation of a treaded robotic capsule colonoscope," *IEEE Trans. Robot.*, vol. 36, no. 2, pp. 545–552, Apr. 2020.

[2] R. L. Siegel, K. D. Miller, and A. Jemal, "Cancer statistics, 2016: Cancer statistics, 2016," *CA: Cancer J. Clin.*, vol. 66, no. 1, pp. 7–30, Jan. 2016. [Online]. Available: http://doi.wiley.com/10.3322/caac.21332

[3] W. Street, "Colorectal Cancer Facts & Figures 2017-2019," p. 40, 2017.

[4] A. M. Wolf, "Colorectal Cancer Facts & Figures 2017-2019," American Cancer Society, 2017. [Online]. Available: https://www.cancer.org/research/cancer-facts-statistics/colorectal-cancer-facts-figures.html

[5] S. G. Shah and Others, "Patient pain during colonoscopy: An analysis using realtime magnetic endoscope imaging," *Endoscopy*, vol. 34, no. 6, pp. 435–450, 2002.

[6] J. O. Alcaide, L. Pearson, and M. E. Rentschler, "Design, modeling and control of a SMA-actuated biomimetic robot with novel functional skin," in *Proc. IEEE Int. Conf. Autom.*. Singapore, May 2017, pp. 4338–4345. [Online]. Available: http://ieeexplore.ieee.org/document/7989500/

[7] G. Pittiglio *et al.*, "Magnetic levitation for soft-tethered capsule colonoscopy actuated with a single permanent magnet: A dynamic control approach," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1224–1231, Apr. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8624310/

[8] L. J. Sliker, M. D. Kern, J. A. Schoen, and M. E. Rentschler, "Surgical evaluation of a novel tethered robotic capsule endoscope using micro-patterned treads," *Surg. Endoscopy*, vol. 26, no. 10, pp. 2862–2869, Oct. 2012. [Online]. Available: http://link.springer.com/10.1007/s00464-012-2271-y

[9] K. Noda, T. Kitada, Y. Suzuki, H. S. Colvin, T. Hata, and T. Mizushima, "A novel physical colonoscopy simulator based on analysis of data from computed tomography colonography," *Surg. Today*, vol. 47, no. 9, pp. 1153–1162, Sep. 2017. [Online]. Available: http://link.springer.com/10.1007/s00595-017-1517-7

[10] L. Manfredi, E. Capoccia, G. Ciuti, and A. Cuschieri, "A soft pneumatic inchworm double balloon (SPID) for colonoscopy," *Sci. Rep.*, vol. 9, no. 1, Dec. 2019, Art. no. 11109. [Online]. Available: http://www.nature.com/articles/s41598-019-47320-3

[11] C. C. Y. Poon, B. Leung, C. K. W. Chan, J. Y. W. Lau, and P. W. Y. Chiu, "Design of wormlike automated robotic endoscope: Dynamic interaction between endoscopic balloon and surrounding tissues," *Surg. Endoscopy*, vol. 30, no. 2, pp. 772–778, Feb. 2016. [Online]. Available: http://link.springer.com/10.1007/s00464-015-4224-8

[12] N. van der Stap, F. van der Heijden, and I. A. M. J. Broeders, "Towards automated visual flexible endoscope navigation," *Surg. Endoscopy*, vol. 27, no. 10, pp. 3539–3547, Oct. 2013. [Online]. Available: http://link.springer.com/10.1007/s00464-013-3003-7

[13] R. Reilink, S. Stramigioli, A. M. Kappers, and S. Misra, "Evaluation of flexible endoscope steering using haptic guidance," *Int. J. Med. Robot. Comput. Assist. Surg.*, vol. 7, no. 2, pp. 178–186, Jun. 2011. [Online]. Available: http://doi.wiley.com/10.1002/rcs.386

[14] S. Xia, S. M. Krishnan, M. P. Tjoa, and P. M. Y. Goh, "A novel methodology for extracting colon's lumen from colonoscopic images," *Systemics, Cybern. Inform.*, vol. 1, no. 2, pp. 7–12, Apr. 2003.

[15] R. Reilink, S. Stramigioli, and S. Misra, "Image-based flexible endoscope steering," in *Proc. IEEE/RSJ Int. . Intell. Robots Syst.* , Oct. 2010, pp. 2339–2344. [Online]. Available: http://ieeexplore.ieee.org/document/5652248/

[16] Z. Zhen, Q. Jinwu, Z. Yanan, and S. Linyong, "An intelligent endoscopic navigation system," in *Proc. Int. Conf. Mechatronics Autom..* , Jun. 2006, pp. 1653–1657. [Online]. Available: http://ieeexplore.ieee.org/document/4026339/

[17] H. Chettaoui, G. Thomann, C. B. Amar, and T. Redarce, "Extracting and tracking colon's "Pattern" from colonoscopic images," in *Proc. IEEE 3rd Can. Conf. Comput. Robot Vis.*, 2006, pp. 65–65.

[18] M. Tjoa, S. Krishnan, and M. Zheng, "A novel endoscopic image analysis approach using deformable region model to aid in clinical diagnosis," in *Proc. 25th Annu. Int. Conf. IEEE Eng. . Med. Biol. Soc.*, 2003, pp. 710–713. [Online]. Available: http://ieeexplore.ieee.org/document/1279862/

[19] I. Bricault, G. Ferretti, and P. Cinquin, "Registration of real and CT-derived virtual bronchoscopic images to assist transbronchial biopsy," *IEEE Trans.*

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Med. Imag., vol. 17, no. 5, pp. 703–714, Oct. 1998. [Online]. Available: http://ieeexplore.ieee.org/document/736022/

[20] N. van der Stap, C. H. Slump, I. A. Broeders, and F. van der Heijden, "Image-based navigation for a robotized flexible endoscope," in *Computer-Assisted and Robotic Endoscopy*, 1st ed., (Lecture notes in computer science). New York, NY, USA: Springer, 2014, no. 8899.

[21] C. K. Kwoh, G. N. Khan, and D. F. Gillies, "Automated endoscopic navigation and advisory system from medical image," *Proc. SPIE* . , vol. 3660, pp. 214–224, May 20, 1999. [Online]. Available: http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=980 428

[22] J. M. Prendergast, G. A. Formosa, C. R. Heckman, and M. E. Rentschler, "Autonomous localization, navigation and Haustral fold detection for robotic endoscopy," in *Proc. IEEE/RSJ Int. . Intell. Robots* , Oct. 2018, pp. 783–790. [Online]. Available: https://ieeexplore.ieee.org/document/8594106/

[23] G. N. Khan and D. F. Gillies, "Vision based navigation system for an endoscope," *Image Vis. Comput.*, vol. 14, no. 10, pp. 763–772, Dec. 1996. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0262885696010852

[24] S. Krishnan, C. Tan, and K. Chan, "Closed-boundary extraction of large intestinal lumen," in *Proc. 16th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Baltimore, MD, USA, 1994, pp. 610–611. [Online]. Available: http://ieeexplore.ieee.org/document/411878/

[25] N. Masson, F. Nageotte, P. Zanne, and M. de Mathelin, "In vivo comparison of real-time tracking algorithms for interventional flexible endoscopy," in *Proc. IEEE Int. Symp. on. Biomed. Imag.: From Nano to Macro*, Boston, MA, USA, Jun. 2009, pp. 1350–1353. [Online]. Available: http://ieeexplore.ieee.org/document/5193315/

[26] N. van der Stap, R. Reilink, S. Misra, I. A. M. J. Broeders, and F. van der Heijden, "The use of the focus of expansion for automated steering of flexible endoscopes," in *Proc. 4th IEEE RAS & EMBS Int. Biomed. Robot. Biomechatronics,* Rome, Italy, Jun. 2012, pp. 13–18. [Online]. Available: http://ieeexplore.ieee.org/document/6290804/

[27] A. M. Plooy *et al.*, "Construct validation of a physical model colonoscopy simulator," *Gastrointestinal Endoscopy*, vol. 76, no. 1, pp. 144–150, Jul. 2012. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0016510712005317

[28] S. M. Pizer *et al.*, "Adaptive histogram equalization and its variations," *Comput. Vis., Graph. Image Process.*, vol. 39, pp. 355–368, Sep. 1987.

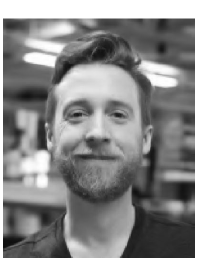
[29] g A. Fitzgibbon, M. Pilu, and R. Fisher, "Direct least squares fitting of ellipses," in *Proc. 13th Int. Conf. Pattern Recognit.* Vienna, Austria, 1996, vol. 1, pp. 253–257. [Online]. Available: http://ieeexplore.ieee.org/document/546029/

[30] J. O'Rourke, A. Aggarwal, S. Maddila, and M. Baldwin, "An optimal algorithm for finding minimal enclosing triangles," *J. Algorithms*, vol. 7, no. 2, pp. 258–269, Jun. 1986. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/0196677486900076

[31] J. M. Prendergast, G. A. Formosa, and M. E. Rentschler, "A platform for developing robotic navigation strategies in a deformable, dynamic environment," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 2670–2677, Jul. 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8338143/

[32] M. Turan, Y. Almalioglu, H. Gilbert, A. E. Sari, U. Soylu, and M. Sitti, "Endo-VMFuseNet: Deep visual-magnetic sensor fusion approach for uncalibrated, unsynchronized and asymmetric endoscopic capsule robot localization data," Sep. 2017, *arXiv: 1709.06041*. [Online]. Available: http://arxiv.org/abs/1709.06041

[33] M. Turan, Y. Almalioglu, H. B. Gilbert, and M. Sitti, "Learning to navigate endoscopic capsule robots," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 3075–3082, Jul. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8744622/

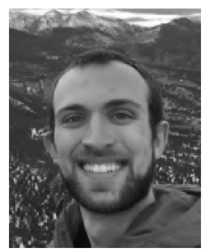


J. Micah Prendergast received the B.S. degree in mechatronic engineering from North Carolina State University, Raleigh, NC, USA, and the University of North Carolina, Asheville, NC, USA, in 2013, and the M.S. and Ph.D. degrees in mechanical engineering from the University of Colorado, Boulder, CO, USA, 2015 and 2019 respectively.

He was a Defense Science and Engineering Graduate (NDSEG) Fellow, focused on the development and validation of robotic endoscopy systems, with the University of Colorado. Currently, he is a Postdoctoral Researcher with the Technische Universiteit Delft, Delft, the Netherlands, where he is working on robotic rehabilitation systems for improved shoulder physiotherapy. His research interests include computer vision, medical device design and robotic sensing, perception and controls.

Gregory A. Formosa (Student Member, IEEE) received the B.S. degree in mechanical engineering from the University of Illinois, Urbana-Champaign, IL, USA, in 2010, and the M.S. and Ph.D. degrees in mechanical engineering from the University of Colorado, Boulder, CO, USA, in 2017 and 2020, respectively.

He was a Researcher with the Advanced Medical Technologies Laboratory (AMTL), University of Colorado, focusing on medical robotics design, dynamics, and control for autonomous endoscopic surgery. Currently, he is completing Postdoctoral work as an Engineering/Science Analyst with the National Science Foundation (NSF) in the Civil, Mechanical, and Manufacturing Innovation (CMMI) Division of the Engineering (ENG) Directorate. Previously, he has been a Design Engineer for on-highway and well-servicing truck transmissions with Caterpillar, Inc., Peoria, IL, USA. His research interests include robotics and mechatronics, medical device design, vehicular dynamics, robust control, and autonomy.



Mitchell J. Fulton received the B.S. and B.A. degrees in mechanical engineering, mathematics, and computational physics from Anderson University, Anderson, IN, USA, in 2017. He is currently working toward the Ph.D. degree in mechanical engineering with the University of Colorado, Boulder, CO, USA.

He is currently a National Science Foundation Graduate Research Fellow and Researcher with the Advanced Medical Technologies Laboratory, University of Colorado, where his research focuses on medical imaging, machine learning, and computer vision.

Christoffer R. Heckman (Member, IEEE) received the B.S. degree in mechanical engineering from UC Berkeley, Berkeley, CA, USA, in 2008 and the Ph.D. degree in theoretical and applied mechanics from Cornell University, Ithaca, NY, USA, in 2012.

He is an Assistant Professor with the Department of Computer Science, University of Colorado at Boulder, Boulder, CO, USA, and the Jacques I. Pankove Faculty Fellow with the College of Engineering & Applied Science. He was an NSF Graduate Research Fellow with Cornell University. He had Postdoctoral appointments with the Naval Research Laboratory, Washington, DC, USA, as an NRC Research Associate, and at CU Boulder as a Research Scientist, before joining the faculty there in 2016.

Mark E. Rentschler (Senior Member, IEEE) received the B.S. degree in mechanical engineering from the University of Nebraska, Lincoln, NE, USA, the M.S. degree in mechanical engineering from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, and the Ph.D. degree in biomedical engineering from the University of Nebraska.

He was a National Defense Science and Engineering Graduate (NDSEG) Fellow with MIT. He is currently a Professor and Sylvia Norviel Cancer Research Faculty Fellow in Mechanical Engineering with the University of Colorado, Boulder, CO, USA. He also holds a secondary appointment with the Department of Surgery, University of Colorado Anschutz Medical Campus, Aurora, CO, USA. His research interests include medical devices and surgical robotics.

Dr. Rentschler is also a Fellow of ASME.

Authorized licensed use limited to: UNIVERSITY OF COLORADO. Downloaded on April 02,2021 at 20:16:55 UTC from IEEE Xplore. Restrictions apply.