# Segmentation of Recorded Endoscopic Videos by Detecting Significant Motion Changes

Manfred Jürgen Primus and Klaus Schoeffmann and Laszlo Böszörmenyi
Alpen-Adria-Universität Klagenfurt, Universitätsstr. 65-67, 9020 Klagenfurt, Austria
{mprimus, ks, laszlo}@itec.at

***Abstract*—In the medical domain it has become common to store recordings of endoscopic surgeries or procedures. The storage of these endoscopic videos provides not only evidence of the work of the surgeons but also facilitates research, the training of new surgeons and supports explanations to the patients. However, an endoscopic video archive, where tens or hundreds of new videos are added each day, needs content-based analysis in order to provide content-based search. A fundamental first step in content analysis is the segmentation of the video. We propose a new method for segmentation of endoscopic videos, based on spatial and temporal differences of motion in these videos. Through an evaluation with 20 videos we show that our approach provides reasonable performance.**

## I. Introduction

A specific form of minimally-invasive surgery is called endoscopy, where an endoscope is inserted into human bodies and organs and transmits images for the surgeon. As a side-product the video-stream can also be recorded for post-operative usage. The reasons why operations and procedures should be recorded for long-term storage are diversified. First, some countries (e.g., the Netherlands) require this by law. Further, in medical research and training the videos can be used as illustrative material. Before the actual surgery, similar videos – or parts of them – can be used to explain patients the operative treatment. Still images, showing major stages of the video, can be extracted post-operative and added to the patients dossier. Moreover, the videos can be used for quality assurance and improvement. Currently, only a small fraction of this fragmentary enumeration is used in practice, due to lack of appropriate software.

Content-based analysis is one of the techniques that can enable the above-mentioned use cases. Typically, the first step of any content analysis (e.g., concept detection) is the temporal segmentation of the video into shots [1]. However, the recordings of endoscopic procedures usually contain only one shot per video file and the visual differences between consecutive frames are small. Therefore, common methods for shot boundary detection [2], mostly focusing on already edited video material, do not help. Another domain of video, where shot boundary detection does not work reliably, is video surveillance. Whereas the camera used for surveillance is usually mounted fixed, the camera in endoscopic videos is moved freely by the surgeon. Beside the movement of the camera for inspection of the operation area and view point changes, a certain level of motion is always existent, even if the camera points to the same area for a while. These circumstances put a limit on the applicability of approaches that were published in the area of video surveillance. As a consequence, special segmentation methods are required.

In this paper we propose a video segmentation approach for endoscopic videos that finds relevant changes in motion. It uses temporal and spatial differences of motion patterns produced either by the camera movement or by the movement of different endoscopic instruments. The motion patterns are calculated with traces of feature points using Kanade-Lucas-Tomasi (KLT) tracking. In succession we group the traces spatially and calculate a singular motion value. Comparing the values of these groups let us segment the video as follows. When the groups provide similar but high values the underlying video frames show camera movement. If the values are similar but low there is no relevant movement at all. If an instrument is moved the values of the groups reveal different values. These approach allows for segmentation of endoscopic content of different kinds of endoscopy and is not limited to a specific one, as e.g., the methods proposed in [3], [4], [5], [6], [7], [8], [9]. Through an evaluation with 20 videos taken from different domains of endoscopy we show that the proposed approach achieves good coverage-overflow respectively precision-recall and F-measure results.

## II. Related Work

A broad overview on video segmentation methods that have been proposed over the last years is given by Del Fabro et al. [1]. They classify segmentation methods into seven classes, which are (1) visual-based, (2) audio-based, (3) text-based, (4) audio-visual-based, (5) visual-textual-based, (6) audio-textual-based and (7) hybrid approaches. Approaches (1) - (3) are basic approaches. The remaining approaches are combinations of them.

Typical endoscopy video files are provided without any audio data or meta-data except some basic classification data concerning the type of operation, data of the patient and the surgeon. These classification data are too simple and do not allow for any segmentation. Therefore, we are limited to visual-based methods. Most of the approaches described in [1] are using color information to segment the videos. Many segmentation methods perform clustering based on RGB- or HSV-color-histograms and on the identification of significant changes in the color distribution. These methods are not applicable for our purpose because most of the time, colors in an endoscopic video are very similar. A significant change happens, however, if an instrument is inserted. Unfortunately, the majority of the instruments have metallic surfaces and are mirroring the surrounding and show thus similar color as the background. On this account the difference between the color histogram of a frame where an instrument is visible and the color histogram of a frame where no instrument is

978-1-4799-0956-8/13/$31.00 ©2013 IEEE

visible is less significant than the difference between two color histograms, when the camera is moved.

In group (1) two motion-based approaches are also available. The first one, introduced by Ngo et al. [10], is using spatio-temporal slices to produce a visual representation of the video. In the segmentation step of this approach the problem with the limited change in the color histogram prohibits the usage of this method as well. The second approach, described by del Fabro et al. [11] uses the motion vectors extracted from H.264/AVC videos. Similar and coherent motions are grouped together. A sliding window is used to detect and extract the most frequent patterns of motion. This approach can only be used for partial segmentation (finding repetitive patterns) and is therefore not applicable for our goal.

Cao et al. introduced a segmentation approach for colonoscopy in [9]. They divided the colon into six sections. Because they use the turnings of the colon to separate the sections it is only applicable to colonoscopy inspections. In this approach they used specific keywords spoken from the physician to trigger the begin of a new segment. In [8] they enhance their approach replacing the speech input by a visual model. When the endoscope is moved around a curve in the colon the images get blurry. Additionally, the physician has to do a new adjustment of the camera after passing the curves. These sequence of sharp images, followed by a set of blurry frames and continued with images with increasing sharpness trigger a scene transition every time this pattern appears.

During a colonoscopic inspection the medic also undertakes some therapeutic actions or removes some tissue for pathological investigations. Cao et al. extended their work in [7] to detect scenes where these additional tasks are done. They take advantage of the circumstance that the wire of the instrument, used for these removals, is very bright caused by reflections of the light source. To detect the bright wire they segment the frames spatially. The bright regions are compared to templates showing possible shapes of the wire. If a sequence of frames matches to these templates it is classified as an operation shot. A further improvement is proposed by Cao et al. in [5]. They improved their results by the use of techniques for image enhancement and by the detection of the insertion direction of the operation instrument.

Padoy et al. present an approach to segment a laparoscopic cholecystectomy into phases, based on temporal synchronization with respect to training data in [3]. A signal is sent each time an instrument is used. The usage of the instruments is analyzed with AdaBoost and weighted regarding to the significance in the current phase. The final segmentation is calculated with an adaptive dynamic time warping algorithm. In a subsequent paper Blum et al. improved the system by automatic instrument recognition [12]. In addition to dynamic time warping they use canonical correlation analysis, which shows better results than using the also tested Hidden Markov Model. Their approach can be used for standardized surgeries, where training data is available and no anomaly or variation during the interventional procedure happens.

We have not found scientific work concerning the partitioning of endoscopic videos into shot-like segments. These segments will be usable for video retrieval, video browsing or video summary approaches amongst others for endoscopic videos as shots are used for these tasks for common videos.


Fig. 1. The segmentation approach is divided into three main steps.

## III. Segmentation Approach

Our algorithm is separated into three steps as illustrated in Figure 1. In the first step (motion detection) we detect matching point features between consecutive frames and store these values in vectors. The point features in the first frame of each consecutive frame pair are selected so that they are more or less evenly spread in order to reflect all kind of motions. In a second step (area motion estimation) we divide the frames into smaller rectangular areas and subsume the distance of the coordinates of the matching point features to a single, aggregated motion value per area. In a third step (transition estimation), the aggregated motion value of each area is compared to the value of the corresponding area of the consecutive frame (except for the last frame). As a result, we can differentiate 3 main cases: (1) If no relevant motion happens then the aggregate motion values are similar to each other and near to zero. (2) In case of a camera movement, the motion values are similar to each other, but greater than a certain threshold. (3) If an object (e.g. an instrument) appears then the motion values of different areas are different. Areas where the object appears show high movement values, whereas areas where the moving object is not visible provide a motion vector with a length of about zero pixels.

We subsume these observations by calculating the standard deviation for every consecutive frame pairs. Based on this, the boundaries of the video segments are calculated.

Our approach is designed to analyze videos captured in minimally-invasive surgery in the first instance. These endoscopic videos usually do not use the whole rectangular video frame (except for videos where the surgeon performed a zoom-in operation during the procedure). Instead, the content is shown in a circular area in the center of the frame. The surrounding of this circular area is dark, sometimes with perceivable noise. If the whole image is used in the analyzing stage, the border of the circle induces many misinterpretations. Good trackable features are usually located at the border because of its contrast. The border is more or less without movement and provides no useful information about motion within the video. To overcome this problem we generate a mask using the algorithm proposed by Muenzer et al. [13] and exclude the area around the border from the tracking stage.

### A. Motion Detection

The aim of this step is to produce a set of matching point features between subsequent frames. Therefore, we select a number of point features $p$ within a video frame, locate the related point features $p'$ in the following frame and store these matched point feature pairs $(p, p')$.

For the selection of the point features we have to consider two principal issues. First, the point features must be well

distributed over the whole frame and second, the feature tracker must be able to find the point features in the second frame reliably. The second requirement is met if the point features are located at corners or within so-called salt-and-pepper textures [14]. These patterns can be found by calculating the eigenvalues $(\lambda_1, \lambda_2)$ of the structure tensor of a window around a pixel, with the size dependent on the magnitude of movement observed in the video, and a threshold $t$ using $min(\lambda_1, \lambda_2) > t$. The higher is $t$ the less but the more reliable point features are found. In order to avoid an unbalanced sampling of point features, in case of dense clusters, our method requires a minimal distance between point features.

For the tracking of the point features we use the well-known Kanade-Lucas-Tomasi (KLT) Feature Tracker. KLT is widely used in its original form but also in slight variations [15], [16]. It is faster than other techniques (e.g. Hessian-affine, Harris-affine, MSER) with the penalty of returning a higher amount of false positive matches. For our purpose this trade-off is negligible because it is absorbed by the amount of selected point features.

If a point feature is not found in the subsequent frame by KLT, it is dropped. For time performance issues it would be nice to reuse point features over a longer sequence of images and to recreate them only if they got lost. However, this is not possible for the following reasons. As endoscopic images are strongly enlarged, the movement distance of objects between two consecutive frames is enlarged as well. On this account small movements of instruments results in enlarged shifts between two neighboring frames. It is a weak point of KLT that it is not as reliable and accurate to track features over great distances [17]. This weakness can be observed if we reuse point features and fast movements of instruments occur over great distances. In this case mismatches happen and are reinforced from frame to frame as long as the strong movement is executed. In this case we have observed that feature points got crowded in some areas whereas in other areas only a few of them remain. This behavior prevents the detection of movement within areas, where only a few feature points remain. For these reasons we have to drop all point features and to select new ones after each matching phase.

### B. Area Motion Estimation

Based on the matching point features found with KLT the motion between two successive frames is examined. We use the motion in different areas of the frame in order to decide if the video should be segmented at the current frame. The frame is split horizontally $I$ times and vertically $J$ times into smaller areas $A$, as shown in Figure 2. In order to distinguish motion in the center of the frame and at the border of the frame we divide each frame into nine areas with $I = J = 3$. Every point feature $p$ found in the previous step is assigned to an area $A_{i,j}$ according to its coordinates. The motion value $M_{i,j}$ of each area is calculated as

$$M_{i,j} = \frac{\sum_{n=1}^{N} \sqrt{{dx_n}^2 + {dy_n}^2}}{N} \quad (1)$$

where $dx = p_x - p'_x$ and $dy = p_y - p'_y$ are the displacements in $x$- and $y$-direction of the point feature pairs. To get comparable results for each area $A_{i,j}$ we divide the sum of the displacements by $N$, which is the number of point features located in the current area. As shown in Equation 1, we use both absolute and signed values for computing the motion because we want to detect any kind of motion. If we would use signed values only, we could not distinguish between movement and no movement in some situations. For example, if there is a diagonal movement in the positive $x$- and negative $y$-direction, the sum of the signed values would result in zero. On the opposite, if we would use absolute values only, we would lose information about movement changes concerning the direction.

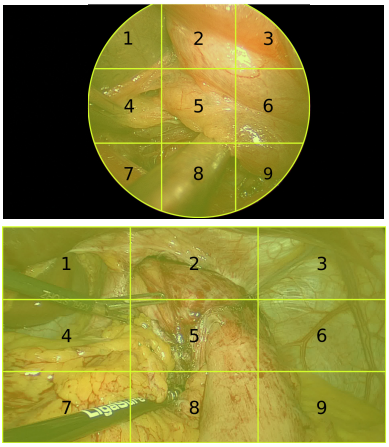


Fig. 2. Separation of a frame with circular (above) and zoomed-in content (below) into $I$ horizontal and $J$ vertical areas ($I = J = 3$).

The entire video is now represented by $K$ vectors of size $L = I \times J$, where $K$ is the number of frames of the video minus one. Each value in a vector represents the main movement within an area from a frame to a subsequent one.

### C. Transition Estimation

If the values of a vector are approximately zero, then there is no movement shown in the frames belonging to this vector. If all values of the vector are similar but significantly greater than zero then the movement is caused by a movement of the camera. If the values of a vector differ ocularly, there is a movement belonging to an active object in the scene. Depending on these observations we concentrate on the detection of these transitions between different movement behaviors.

Each value in the movement vectors reflects the instantaneous motion inside the area. Looking at the motion curve we do not see a smooth curve but a curve with positive and negative peaks. These peaks must be straightened out to prevent misinterpretation. Therefore, a temporal window is shifted along the motion values of the same areas. Inside this window of size $O$ the mean value $r$ is calculated. The bigger $O$, the more robust is response $r$. High frequent motion changes between motion and no motion should not influence the decision if there is a segment boundary or not.

The calculation of these $r$ values gives us a horizontal view to the movement changes inside an area. These views are represented by the nine diagrams at the head of Figure 3. Each diagram shows us the average movement within the corresponding area $A_{i,j}$. High motion is represented by large values in all areas (compare second 73 to 75 in Figure 3) and low motion by small values (second 60 to 64). Different values

are shown if an instrument is moved in the operation area (second 54 to 60). In the first half an instrument is inserted for a procedure (constant high values in some areas until second 58). In the second half the instrument is removed (decreasing values in area 1 and 2 and no movement at all in the other areas). Now we have to switch our view to a vertical one and compare what happens at a current time point of the video in each area. Therefore we calculate the standard deviation $\sigma^V$ with respect to the responses $r$ of each frame, where each response $r$ represents the motion in one of the $A$ areas of a frame and $\bar{r}$ is the mean value of these responses.

$$\sigma^V = \sqrt{(\sigma^V)^2} = \sqrt{\frac{1}{L-1}\sum_{i=1}^{L}(r_l - \bar{r})^2} \qquad (2)$$

The resulting $\sigma^V$ values correlate with movement changes in the video. If we also take into account the mean value we can distinguish between (1) no movement ($(\sigma^V < \epsilon) \wedge (\bar{r} < \epsilon)$), (2) camera movement ($(\sigma^V < \epsilon) \wedge (\bar{r} > \epsilon)$) and (3) object movement within a frame ($(\sigma^V > \epsilon)$). This is a very nice and simple result, nevertheless, for the segmentation we need further considerations. One is the determination of $\epsilon$. $\epsilon$ is related to the quality and the content of the video. Another consideration is that the aggregation of the values into a single one is not as smooth as needed. Therefore we reuse Equation 2 to overcome these problems in the following way:

Within a window of size $P$ we calculate the standard deviation $\sigma^H$ based on the $\sigma^V$ values, whose magnitude changes from low to high or the way around every time there is a change of movement within the video. To get the segmentation borders we iterate along the resulting $\sigma^H$ values and mark a frame as a border if there is a peak value. This leads to a segmentation defining segment borders at strong changes of the motion pattern. We assume that many of the resulting segments are semantically meaningful units - an exact investigation of this issue is topic of further research. The sensitivity to the motion changes is influenced by some thresholds and window sizes, which must be chosen carefully.

### D. Thresholds and Window Sizes

First, let us consider $O$, the size of the window used for smoothing and filtering noise. Noise can have multiple reasons. One reason is caused by mismatches caused by KLT, as already mentioned. Another reason is assignable to bumpy camera movements or to singular jerks. In our example videos a certain level of noise is caused by the beat of the heart. The bigger $O$ is chosen the less is the influence of noise. On the other side, if $O$ is too large, it may lead to undetected segment boundaries.

Window size $P$ is used in calculating $\sigma^H$, where $\sigma^H$ is used to detect movement changes between different areas caused by a moving object. If the $\sigma^V$ values are high then the correlating video scene is showing a movement of an object. If the $\sigma^V$ values are low the video is just showing a camera movement or a scene with no movement at all. $\sigma^H$ points to local changes at $\sigma^V$ values, hence the size of $P$ is small.

At last we need a threshold $\alpha$ to decide if a $\sigma^H$ value should be counted as high (indicating a moving object) or as low (indicating camera movement, or lack of movement).

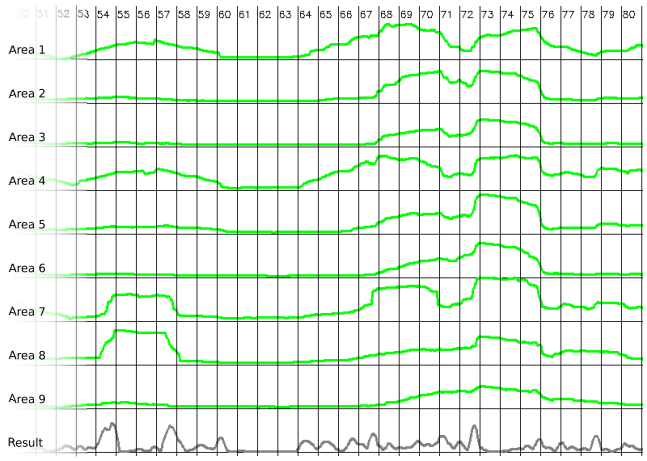


Fig. 3. This motion histogram shows the motion values of a part of an endoscopic video. The numbers at the top and the affiliate lines show the seconds of the video. The histograms for Area 1 to Area 9 show the average motion within these areas as green lines (values range is $[0, 107]$). The last diagram shows the resulting $\sigma^H$ values that are used to calculate the segmentation.

The values we have used for our tests have been found empirically by using combinations of different values for $O$, $P$ and $\alpha$ and applying them to three videos of our data set. The best window size $O$ to calculate the average movement in each area has been found to be in the range $[40, 80]$, the window size $P$ to calculate $\sigma^H$ is chosen to be in the range $[6, 10]$ and the threshold $\alpha$ to separate high and low $\sigma^H$ values is chosen to be in the range $[0.25, 0.45]$. Every combination provides a different number of true and false positive respective negative segmentations. To maximize the positive findings and minimize the negative ones we introduce a post-processing step to combine the different segmentations.

### E. Post-Processing Step

The post processing step is based on the outcome of several runs with combinations of the previously stated intervals. To reduce runtime we used sets of discrete values, namely $O \in \{40, 60, 80\}$, $O \in \{6, 8, 10\}$ and $\alpha \in \{0.25, 0.30, 0, 35, 0.40, 0.45\}$ and combined them to $O \times P \times \alpha$ different combinations. These combinations are used as parameters to get various segmentation candidates per video file. The candidates consist of sets of frame numbers. Each frame number shows a border of two segments. Comparing the borders at the result sets we see that the borders are spread over a small area of neighboring frame numbers.

Therefore, we have to find the best fitting border suggestion. For this we count, how often a certain frame is proposed as a border. The array $B$ contains this counter for each frame. Then we compute the best suggestion by the use of a weighting function to distribute borders close to $i$ with higher weight as follows:

$$B'[i] = \sum_{j=0}^{w} \frac{w}{j+1} * (B[i-j] + B[i+j]) \qquad (3)$$

where $w$ is a small window (actually 10) and $B'$ contains the likelihood of a frame being a border. Peak values, which are found iterating through $B'$, denote segmentation bounds.

## IV. Evaluation

The approach presented in this paper has been evaluated with the help of 20 distinct videos of laparoscopic and endoscopic thyroid surgeries, recorded in HD resolution. The overall length of the videos is 68 minutes, with an average length of about three minutes and a half, each. For the evaluation we manually segmented and annotated these 20 videos according to motion activity (described in the subsection below) with the accuracy of single frames.

### A. Creating the Ground Truth

The scenes shown in the videos are a mixture of different activities during surgery. In some kind of the observations the camera is moved around to get an overview of the area where the surgery takes place and to inspect areas after important stages of the invasion. A camera movement is also done, if another point of view to the operation area is needed. An additional form of observation is done if the camera is held fixed and the surgeons observe a region of interest in detail (represented by keyframes of segment #1 in Figure 4 and segment #1, #3 and #6 in Figure 5). In all these cases instruments are either not visible, or if, they show no or only marginal movement.

The next group we identified within these videos is based on the presence of instruments. The usage of the instruments could be classified into some common kinds of segmentations. These are e.g. the insertion and removal of instruments into or from the operations area and movements within the operation area, with the purpose of positioning the instrument at a specific location. Beside these common types of movements we noticed that each instrument (scissors, needleholders, graspers, dissectors, retractors, mono- and bipolar instruments and so forth) has its own more or less typical set of course of motion. The scissor, for example, is not only used for cutting, but also used to push layers of tissues aside.

The segmentation of the videos for the ground truth is based on these observations. Every time, when a new course of motion started or ended, we set a segmentation bound. Most of the times there are a few frames, where nothing happens, until a new motion sequence starts. We set also segmentation boundaries at the begin and the end of an overview of the operation area and on the border of still image like scenes. Sometimes two instruments are used together for instance to pull apart adhered tissues. This combined motion has also been regarded as one segment.

In Figure 4 and Figure 5 we can see visualizations of the ground truth and the automatically found segmentations for two of the 20 video files. The selected segments show typical sequences during an operation, characterized by a frequent change of instruments and their usage. Each segment is represented by a keyframe, which is the center frame of a segment. The keyframes of the ground truth and the annotated segmentations show well that our approach is reliable with respect to the ground truth.

### B. Results

To measure the performance of our algorithm we have used Recall-Precision, F-Measure (harmonic mean of precision and recall) and Coverage-Overflow as proposed by Vendrig et al [18]. Coverage is the value to what extent each identified segment meets the equivalent segment of the ground truth. Overflow at the other hand shows how many frames are assigned to a segment, although they do not belong to this segment. The optimal value for coverage is 100% and for overflow 0%.

TABLE I. Coverage and overflow of the 20 videos

| Video Number | Coverage | Overflow | Precision | Recall | F-measure |
|---|---|---|---|---|---|
| 001 | 0.89 | 0.16 | 0.78 | 0.91 | 0.84 |
| 002 | 0.76 | 0.47 | 0.85 | 0.86 | 0.85 |
| 003 | 0.79 | 0.55 | 0.85 | 0.82 | 0.84 |
| 004 | 0.81 | 0.63 | 0.95 | 0.89 | 0.92 |
| 005 | 0.70 | 0.52 | 0.73 | 0.85 | 0.79 |
| 006 | 0.77 | 0.58 | 0.77 | 0.85 | 0.81 |
| 007 | 0.79 | 0.65 | 0.83 | 0.85 | 0.83 |
| 008 | 0.77 | 0.54 | 0.86 | 0.88 | 0.87 |
| 009 | 0.77 | 0.51 | 0.92 | 0.90 | 0.91 |
| 010 | 0.81 | 0.52 | 0.94 | 0.91 | 0.93 |
| 011 | 0.80 | 0.58 | 0.94 | 0.86 | 0.90 |
| 012 | 0.75 | 0.67 | 0.84 | 0.86 | 0.85 |
| 013 | 0.74 | 0.52 | 0.86 | 0.91 | 0.88 |
| 014 | 0.83 | 0.52 | 0.91 | 0.93 | 0.92 |
| 015 | 0.80 | 0.62 | 0.86 | 0.86 | 0.86 |
| 016 | 0.78 | 0.78 | 0.83 | 0.76 | 0.79 |
| 017 | 0.82 | 1.03 | 0.82 | 0.71 | 0.76 |
| 018 | 0.83 | 0.53 | 0.89 | 0.89 | 0.89 |
| 019 | 0.80 | 0.58 | 0.84 | 0.86 | 0.85 |
| 020 | 0.82 | 0.44 | 0.87 | 0.93 | 0.90 |
| Average | 0.79 | 0.57 | 0.86 | 0.86 | 0.86 |

Table I shows the results for each endoscopic video in detail. The average coverage value of 79% shows good performance but leaves room for further tunings and improvements. The moderate results concerning the overflow are justifiable. If the border that is found by the algorithm is only one frame aside the frame number noted in the ground truth, the result gets penalized. This strictness is not meaningful for our approach. The frame where a separation of two segments can be stated cannot be fixed to a single frame typically. Mostly there is an overlapping area between two segments where a border can be stated. The size of this area can be up to 25 frames long, sometimes even more. For this reason we evaluate our results with Precision, Recall and F-Measure as well. The corresponding values show that our approach achieves reasonable segmentation performance with Recall and Precision higher than 90%, for several videos. The area where a border is measured as true positive is within 25 frames. The results confirm that our novel motion-based segmentation approach is applicable as a basis for further content-based analyses.

## V. Conclusion and Further Work

We have proposed a novel approach to group frames of endoscopic videos into segments based on changes of the motion pattern, such as (1) no movement, (2) motion caused by camera movement and (3) movements of endoscopic instruments. These segments could be used for further content-based analysis. We have shown that our approach is robust and accurate.

Future work will cover the problems we have identified during the evaluation of our approach and to overcome them by the use of additional low level features. We also intend

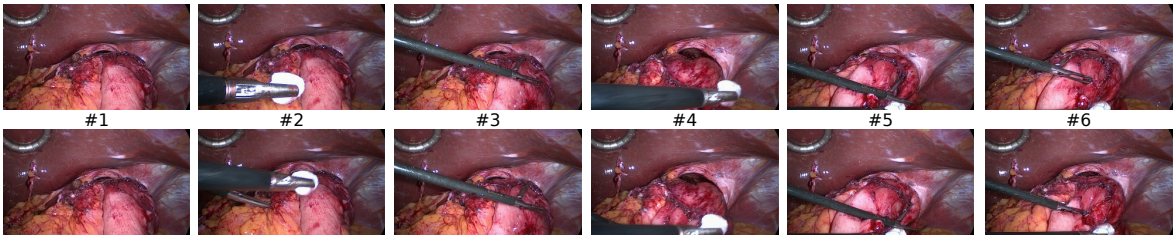


Fig. 4. The center frame of a segment is used to represent ground truth segments in the first line and annotated segments in the second line. The first represents a segment where no movement happens, the following five keyframes show segments where instruments are used for different purposes. Although the last two images are similar the instrument is used for different purposes in these segments.

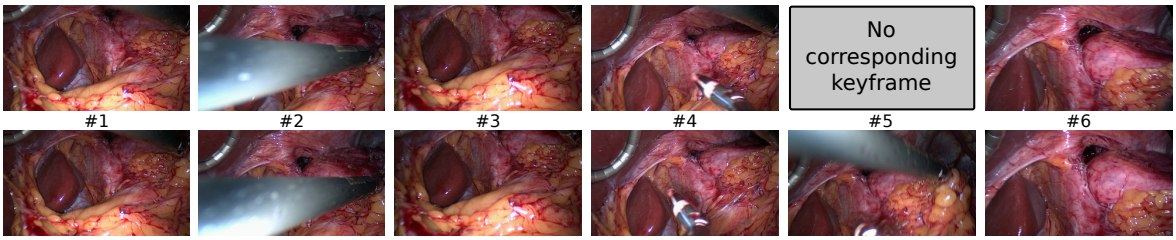


Fig. 5. This sequence of frames shows a scene where tissues are cauterized. Segment number five is not identified as a segment in the ground truth but it is identified by our algorithm. The reason is that the second instrument was removed at the same time as the cauter and was therefore not considered in the ground truth.

to work – in cooperation with medical experts – on the grouping of the small segments to semantically meaningful scenes. Furthermore, we plan to improve the reliability of the segmentation process and to investigate, how to use it for similarity search in endoscopic videos.

## ACKNOWLEDGMENT

This work was supported by Lakeside Labs GmbH, Klagenfurt, Austria and funding from the European Regional Development Fund and the Carinthian Economic Promotion Fund (KWF) under grant KWF - 20214 22573 33955.

## REFERENCES

[1] M. Del Fabro and L. Böszörmenyi, "State-of-the-art and future challenges in video scene detection: a survey," *Multimedia Systems*, pp. 1–28, 2013.

[2] A. F. Smeaton, P. Over, and A. R. Doherty, "Video shot boundary detection: Seven years of trecvid activity," *Comput. Vis. Image Underst.*, vol. 114, no. 4, pp. 411–418, Apr. 2010.

[3] N. Padoy, T. Blum, I. Essa, H. Feussner, M.-O. Berger, and N. Navab, "A boosted segmentation method for surgical workflow analysis," in *Proceedings of the 10th international conference on Medical image computing and computer-assisted intervention - Volume Part I*, ser. MICCAI'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 102–109.

[4] M. Mackiewicz, J. Berens, and M. Fisher, "Wireless capsule endoscopy color video segmentation," *Medical Imaging, IEEE Transactions on*, vol. 27, no. 12, pp. 1769–1781, Dec. 2008.

[5] Y. Cao, D. Liu, W. Tavanapong, J. Wong, J. Oh, and P. de Groen, "Computer-aided detection of diagnostic and therapeutic operations in colonoscopy videos," *Biomedical Engineering, IEEE Transactions on*, vol. 54, no. 7, pp. 1268–1279, July 2007.

[6] M. Mackiewicz, J. Berens, and M. Fisher, "Wireless capsule endoscopy video segmentation using support vector classifiers and hidden markov models," in *Proceedings of the International Conference on Medical Image Understanding and Analyses*, 2006.

[7] Y. Cao, D. Li, W. Tavanapong, J. Oh, J. Wong, and P. C. de Groen, "Parsing and browsing tools for colonoscopy videos," in *Proceedings of the 12th annual ACM international conference on Multimedia*, ser. MULTIMEDIA '04. New York, NY, USA: ACM, 2004, pp. 844–851.

[8] Y. Cao, W. Tavanapong, D. Li, J. Oh, P. Groen, and J. Wong, "A visual model approach for parsing colonoscopy videos," in *Image and Video Retrieval*, ser. Lecture Notes in Computer Science, P. Enser, Y. Kompatsiaris, N. OConnor, A. Smeaton, and A. Smeulders, Eds. Springer Berlin Heidelberg, 2004, vol. 3115, pp. 160–169.

[9] Y. Cao, W. Tavanapong, K. Kim, J. Wong, J. Oh, and P. de Groen, "A framework for parsing colonoscopy videos for semantic units," in *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, vol. 3, June, pp. 1879–1882 Vol.3.

[10] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang, "Motion-based video representation for scene change detection," *International Journal of Computer Vision*, vol. 50, no. 2, pp. 127–142, 2002.

[11] M. del Fabro and L. Böszörmenyi, "Video scene detection based on recurring motion patterns," in *Advances in Multimedia (MMEDIA), 2010 Second International Conferences on*. IEEE, 2010, pp. 113–118.

[12] T. Blum, H. Feuner, and N. Navab, "Modeling and segmentation of surgical workflow from laparoscopic video." in *MICCAI (3)*, ser. Lecture Notes in Computer Science, T. Jiang, N. Navab, J. P. W. Pluim, and M. A. Viergever, Eds., vol. 6363. Springer, 2010, pp. 400–407.

[13] B. Muenzer, K. Schoeffmann, and L. Boeszoermenyi, "Detection of circular content area in endoscopic videos," in *Proc. of the 26th IEEE Int. Symp. on Computer-Based Medical Systems*, 2013.

[14] J. Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, jun 1994, pp. 593 –600.

[15] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. of the 7. Int. Joint Conf. on Artificial Intelligence (IJCAI '81)*, 1981, pp. 674–679.

[16] C. Tomasi and T. Kanade, "Detection and tracking of point features," Tech. Report CMU-CS-91-132, Carnegie Mellon University, Tech. Rep., Apr. 1991. [Online]. Available: http://www.ces.clemson.edu/~stb/klt/tomasi-kanade-techreport-1991.pdf

[17] C.-H. Ko, Y.-P. Tsai, and Y.-P. Hung, "Tracking features with large motion," *18th IPPR Conference on Computer Vision, Graphics and Image Processing*, pp. 1725–1730, 2005.

[18] J. Vendrig and M. Worring, "Systematic evaluation of logical story unit segmentation," *IEEE Transactions on Multimedia*, vol. 4, no. 4, pp. 492–499, 2002.