Universidad Zaragoza
1542

Master's Thesis

Automatic endoscopy video summarization

Author

Juan Plo Andrés

Supervisors

Ana Cristina Murillo Arnal

Oscar León Barbed Peréz

Master of Engineering in Robotics, Graphics and Computer Vision

ESCUELA DE INGENIERÍA Y ARQUITECTURA
2025

Abstract

Endoscopies are essential medical procedures for diagnosing, treating, and monitoring a wide range of conditions affecting the digestive system and other internal organs. Their importance lies in providing a visualization of the inside of the human body, allowing early detection of anomalies such as polyps, inflammation, bleeding, or tumors. However, endoscopies are large videos with lots of redundant data, resulting in a high processing cost.

This work has been developed as part of the Robotics, Computer Vision and Artificial Inteligence (RoPeRT) research group and is part of the EndoMapper project. The Endomapper project aims to develop advanced technologies for real-time localization and mapping within the human body using endoscopic video feeds.

Endoscopy videos, and colonoscopies in particular, obtained from real medical practice are videos about 10-30 minutes long. These videos frequently contain a substantial amount of redundant frames and are inherently noisy due to factors such as camera movements, lighting variations, which can obscure important visual information.

Addressing these challenges is crucial to improve the usability and effectiveness of endoscopy video analysis, enabling faster and more accurate insight for medical professionals. This work presents a new method for video summarization in the endoscopy domain. To generate an overview of an endoscopy procedure in a less overwhelming way, the developed method combines video summarization and video segmentation strategies to generate summaries.

The main tasks have been studying and applying some of the current state-of-the-art methods in video summarization and analyzing how they perform in the endoscopy domain. Also, we analyze how to generate summaries using video segmentation methods and how to take advantage of them, combining them with video summarization strategies.

The performed experiments involved comparing two current state-of-the-art methods in video summarization performance in endoscopy videos and evaluating the different methods developed in this work.

The project has been developed using very few annotated data, which prevented us from re-trainning the summarization model and made it difficult to perform exhaustive evaluations. However, the results obtained are satisfactory and serve as an starting point for endoscopy video summarization. Annotating more endoscopy videos, performing a re-training of the summarization model and investigating the applications of video summarization in other medical procedures are some of the possibilities of future work.

Acknowledgements

I would like to express my gratitude to two individuals for their invaluable support in completing this work. They are my two Master's Thesis advisors, Ana Cristina Murillo and Oscar Leon Barbed. I deeply appreciate all the help, guidance, and supervision they have provided throughout this long and challenging process, as well as for addressing all my questions and clarifying my doubts. Additionally, I am thankful for their guidance in the development of this work and for all the resources they have made available to me. Undoubtedly, I would not have been able to complete this project without their assistance.

I also want to thank my family, who have supported and encouraged me to continue developing this incredible work.

Finally, I wish to express my gratitude to the Universidad de Zaragoza and the Escuela de Ingenieria y Arquitectura for granting me the opportunity to receive a high-quality education.

Contents

Chapter 1

Introduction

1.1 Motivation

Artificial intelligence (AI) is revolutionizing medicine by transforming the way diseases are diagnosed, treated, and prevented. By analyzing vast amounts of data, AI enhances medical procedures such as the detection of conditions such as cancer and the development of accurate diagnoses and treatment plans. Additionally, AI plays a significant role in improving healthcare accessibility.

However, challenges such as data privacy limit access to the extensive datasets required to develop robust AI systems. Insufficient or unrepresentative data can introduce algorithmic bias, which may impair the performance and fairness of these technologies. Risks such as over-reliance on AI, potential misdiagnoses, and ethical concerns regarding patient data usage underscore the importance of careful oversight and transparency. Addressing these issues while leveraging the immense potential benefits of AI requires collaboration among healthcare providers and technologists to ensure its responsible, equitable, and effective application in medical practice.

Endoscopies are medical procedures that involve the use of a specialized instrument called an endoscope to examine the interior of a hollow organ or cavity within the body. The endoscope is a flexible or rigid tube equipped with a light source and a camera, which transmits images to a monitor, allowing healthcare providers to visualize areas that are otherwise inaccessible without surgery.

Endoscopies play a vital role in the early detection, diagnosis, and management of various medical conditions, including gastrointestinal disorders, cancers, and respiratory issues. They allow doctors to directly visualize internal organs and accurately identify abnormalities such as ulcers, tumors, and infections. Furthermore, endoscopies often enable therapeutic procedures, such as polyp removal or biopsies, to be performed during the same session, eliminating the need for additional surgeries. This approach minimizes patient discomfort and recovery time while reducing the risks associated with more invasive interventions.

To make a diagnosis, a professional must occasionally examine the entire endoscopy recording. These videos are lengthy and, while containing important information about the patient's health, they also include a significant amount of redundant content that can slow down the review and reporting process. This huge amount of data makes professionals rely on just a few images chosen during the procedure. This is where video summarization becomes useful. Video summarization consists in automatically extracting the most relevant parts of a video, either through textual descriptions or a sequence of keyframes. By summarizing endoscopies, we will reduce professionals' workload and improve efficiency, save time, enhance diagnostic

accuracy, facilitate data management, and support better patient care. Developing a method to perform this task is the main goal of this project, described in the next section.

1.2 Goal and tasks

The main goal of this project is to develop an **automatic video summarization** system for endoscopy videos that provides a complete **overview** of the recording in a less overwhelming manner. By adapting state-of-the-art methods in video summarization to the endoscopy domain, the project aims to facilitate the semantic understanding of the content of hours of medical videos.

To understand the exact goal of this work, an example is depicted in Figure 1.1. An endoscopy can be divided into different sections based on the information provided by the frames within each section. Our objective is to represent all sections using the fewest frames possible. As shown in the summary in Figure 1.1, the summary includes one frame per section. The sections vary in importance, and our goal is to enhance the representation of the more important sections while maintaining minimal representation for those that are less relevant.

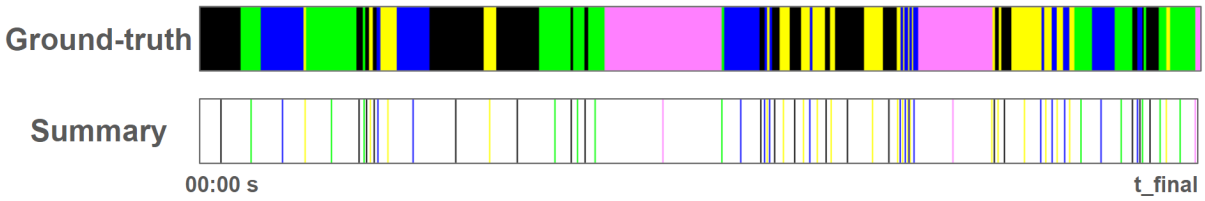


Figure 1.1: Visualization of a goal summary next to the ground truth sections manually annotated. Each bar represents the duration of all the video and the colored intervals the selected frames for the summary.

The tasks developed during this project are the following:

- **Literature review** on related works and the state-of-the art in video summarization.
- Select at least **one video summarization method** to implement or adapt to work with endoscopy videos.
- **Evaluate** the performance of the **selected method** and how well it covers the main goal of this work. The evaluation will use standard metrics commonly applied in these tasks, along with custom-developed evaluation metrics tailored to the endoscopy domain, and will be conducted on **real endoscopy data** for validation.
- **Combine** the video summarization method **with video segmentation** techniques in order to improve the results and make them fit better with our goal.
- **Evaluate** the benefits of **combining** both video processing techniques.

1.3 Context and tools

This work was developed in the Robotics, Perception and Real-time group at the University of Zaragoza, in the Institute of Engineering and Research of Aragon (i3A).

Endomapper: Real-time Mapping from Endoscopic Video. The Endomapper project aims to establish the foundations for real-time localization and mapping within the human body using only the video feed from a standard monocular endoscope. Its goal is to introduce live augmented reality to endoscopy, such as displaying the precise location of a tumor previously detected in a tomography or providing navigation guidance to help the surgeon reach the exact site for performing a biopsy. Endomapper explores the fundamentals of non-rigid geometry techniques to achieve, for the first time, mapping from gastrointestinal (GI) endoscopies. This is accomplished by leveraging existing state-of-the-art methods and enhancing them through the application of machine learning.

EndoMapper dataset. The EndoMapper dataset [1] contains 96 recordings of colonoscopy and gastroscopy procedures. The acquisition of the sequence in the dataset was performed at the Hospital Clinico Universitario Lozano Blesa, in Zaragoza (Spain). The dataset includes endoscopic videos of varying durations, totaling over 24 hours of footage.

Tools. Python is the most widely used programming language in deep learning systems, offering a rich ecosystem of libraries specifically designed for this field, including TensorFlow, PyTorch, Keras, and NumPy. Many state-of-the-art video segmentation methods are developed using these Python tools. Additionally, FFmpeg was considered as a tool for video segmentation.

1.4 Project structure

The master's thesis documentation is divided into the following chapters:

- Chapter 1 of the project, the introduction, provides an overview of the context, tools, objectives, contributions, and tasks of the project.
- Chapter 2 of the project focuses on the literature review of video summarization and the video segmentation methods relevant to this work. It begins with an overview of various approaches to video summarization developed over the years. Subsequently, it explores the framework chosen for this project, followed by an explanation of the video segmentation task and the tools utilized in the study.
- Chapter 3 focuses on the automatic analysis of endoscopic videos, presenting the proposed approach in detail. It begins with an overview explaining the system's structure, followed by a comprehensive description of the various modules involved in the workflow.
- Chapter 4 focuses on the experiments conducted to evaluate the functionality of the developed system through various tests and results.

  This chapter includes a comparison between two selected video summarization methods and the rationale for choosing one over the other. It also details the experiments designed to develop a strategy for creating summaries that combine video summarization and segmentation techniques. Finally, it provides an overview of the results obtained using the final approach on a set of videos from the EndoMapper dataset.
- Chapter 5 presents the conclusions drawn from the results obtained, as well as a discussion on the limitations of the work and possible future steps to be taken.

Chapter 2

Related work and Background

2.1 Video summarization

Various methods have been proposed over the past decade to automate video summarization in the field of computer vision. The most advanced techniques currently rely on deep neural network architectures.

The first approaches modeled temporal dependencies of frames and learned the importance of them using ground-truth annotations. For this, some methods used RNN's architecture [2], [3], or Fully Convolutional Sequence Networks [4]. Others tried to improve the lack of capacity these networks have by stacking multiple Long Short Term Memory and memory layers hierarchically [5]. Apostolidis *et al.* [6] tried to combine global and local attention mechanisms to give importance scores to the frames of the video. A recent approach by Mei *et al.* [7] presented a multi-modal video summarization including visual and textual video summarization techniques in order to provide both textual and visual summaries.

There are several approaches that have tried to tackle the video summarization task in the endoscopy domain. A popular approach is to perform clustering of the frames representing the frames in different ways. Li et al. [8] uses the color histogram of the frames as the encoding, then performs k-means to create the summary. Ismail et al. [9] tried to perform a clustering of the frames by encoding them using the MPEG-7 description. Recent studies employ more advanced deep learning methods. Sushma *et al.* [10] uses a CNN architecture to detect abnormal frames for summary creation. Chen *et al.*[11] utilizes a saliency encoding neural network (SNN) to extract the most relevant frames from recordings, showing promising results. However, these approaches do not fully align with the objectives of this work: some were tested on only short endoscopy videos, while others generated only a set of keyframes as the final summary.

We tested two of the previously mentioned methods: PGL-SUM [6] and UBISS [7]. PGL-SUM produces a final summary composed of a few selected shots from the original video. In contrast, UBISS yielded results more aligned with our objective, as its summary frames were more evenly distributed. We selected the UBISS visual summary framework as the baseline for our system because it provided a good baseline for our objective. Moreover, UBISS offered a simpler framework and reduced computational time compared to PGL-SUM.

2.2 Video segmentation

Video segmentation involves dividing a video V into k distinct sections s_i, where $\bigcup_{i=0}^{k} s_i = V$, based on spatial or temporal features, or by the semantic meaning conveyed in each shot. This process can help identify key regions, objects, or events within the video, making it easier to analyze, summarize, or retrieve relevant information. It plays a crucial role in applications such as video summarization, object tracking, and scene understanding.

There is an existing video segmentation tool called FFmpeg, which detects significant changes in video frames to identify segmentation points. An endoscopy video segmentation approach proposed in [12], utilizes the BYOL unsupervised representation learning framework to construct a video segmentation system.

Video segmentation and video summarization are two important tasks in video analysis and processing. While they have different objectives, they can be combined to improve each other. Our approach is to use summarization as a starting point and leverage video segmentation to enhance the summary.

Chapter 3

Automatic overview of endoscopic videos

The starting point of this work was the visual summary framework of UBISS [7], explained in more detail later in this section.

The framework consists of a saliency encoder and a *Summary Regressor*, which processes video features to assign an importance score to each frame of the original video. The output is then thresholded to generate an initial summary that highlights the most significant parts of the endoscopy.

On the other hand, we have the endoscopy video segmentation method proposed by Barbed *et al.* [12]. The segmentation method takes video features as input and assigns a label via clustering and temporal smoothing.

The initial summary and video segmentation are both used as inputs for a fusion algorithm. For this algorithm, we explore various fusion strategies to create an overview of the endoscopy, emphasizing the most important parts while preserving the procedure's structure in a clearer, less overwhelming format.

Figure 3.1 shows a diagram of the proposed approach, highlighting the different modules that form the architecture. The segmentation, summarization, and fusion modules are described in detail later.

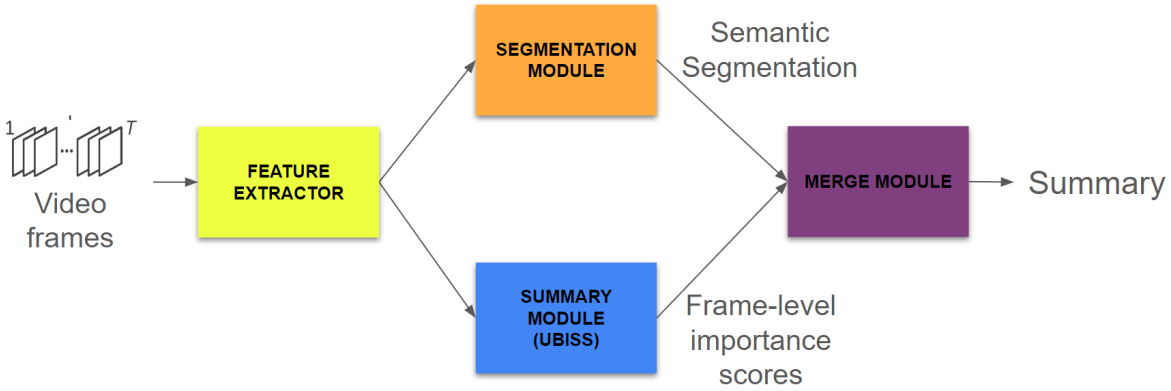


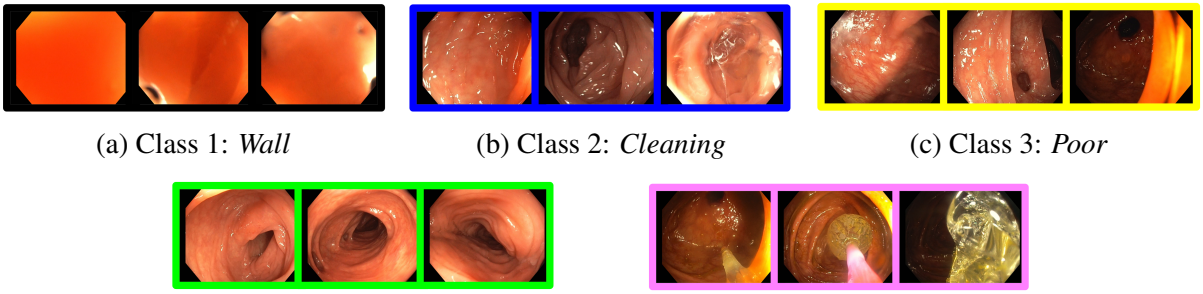
Figure 3.1: Given a video of T frames, our approach produces a set of deep feature representations using a pretrained CNN model. These representations are the input of both the segmentation module and the summarization module. The segmentation module produces a video segmentation with semantic labels. The summarization module generates frame-level importance scores. Both the segmentation and the scores are then used as the input of the fusion module, that generates the summary.

3.1 Segmentation Module

The segmentation module is composed by the endoscopy video segmentation method proposed in [12]. They run BYOL unsupervised representation learning on the training set of the same endoscopy dataset used in this work. With this, they obtain a way to describe endoscopies. Using these descriptors, they perform a clustering on another train set of the dataset. Then, they visually inspect the frames of each cluster to classify them into the following semantic classes:

- *Wall* represents frames in which the endoscope is facing a wall, losing all visibility.
- *Cleaning* are frames captured while the water pump is being used or other liquids are present, also losing visibility.
- *Poor* view are frames where, due to blur in the image or a bad endoscope position, it does not have full visibility.
- *Good* view refers to frames in which the endoscope is well positioned to examine and navigate the colon.
- *Tool* are frames in which a procedure is being performed and the tool used is represented in the image.

Some frame examples of the classes are shown in Figure 3.2.



(a) Class 1: *Wall* (b) Class 2: *Cleaning* (c) Class 3: *Poor*

(d) Class 4: *Good* (e) Class 5: *Tools*

Figure 3.2: Sample images from the 5 different classes

An example of video segmentation is depicted in Figure 3.3. This segmentation can be simplified in a binary segmentation depending on the importance of the classes: *Tools*, *Good* and *Poor* are considered as *Good* segments, depicted in green in Figure 3.3; *Cleaning* and *Wall* are considered as *Bad* segments, depicted in red in Figure 3.3. Both segmentations contribute to the elaboration of the summary.

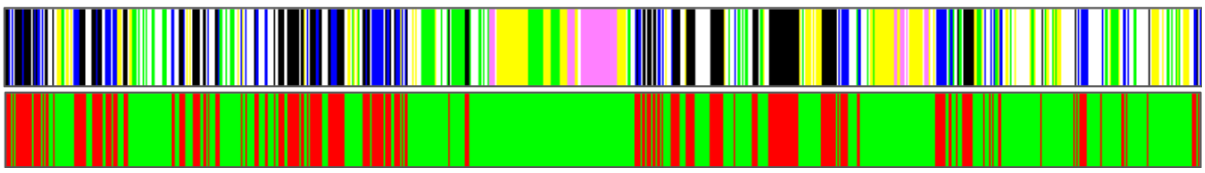


Figure 3.3: Example of endoscopy video segmentation. The top image shows the full-class segmentation, while the bottom image presents the binary simplification of the segmentation.

3.2 Summarization Module

The summarization module uses the visual summary framework from UBISS [7]. The model architecture is shown in Figure 3.4.

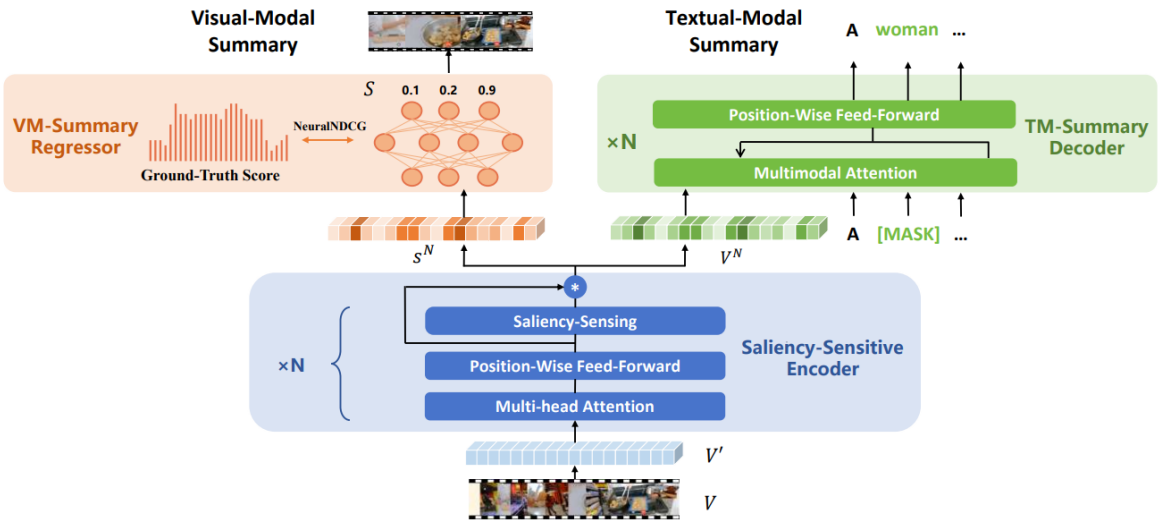


Figure 3.4: Model Architecture of UBISS. Figure from [7].

The input sequence V is first embedded as a feature sequence V' via a pre-trained model. Then, features are used as the input of the Saliency-Sensitive Encoder. Each encoder layer contains a saliency-sensing layer for learning temporal saliency information, except for the traditional multi-head attention layer followed by a position-wise feed-forward layer. The saliency-sensing layer first calculates the sigmoid function based on the output of the feedforward layer, obtaining an score S^N. S^N is then used as the input of the VM-Summary regressor, which is composed of two linear layers that transform the score S^N into the predicted saliency score S. TM-Summary decoder produces a textual summary using the score S^N. As it generates text, we do not use it in our system.

Finally, the saliency score S is thresholded to create a summary, highlighting the most important parts of the video (Figure 3.5).

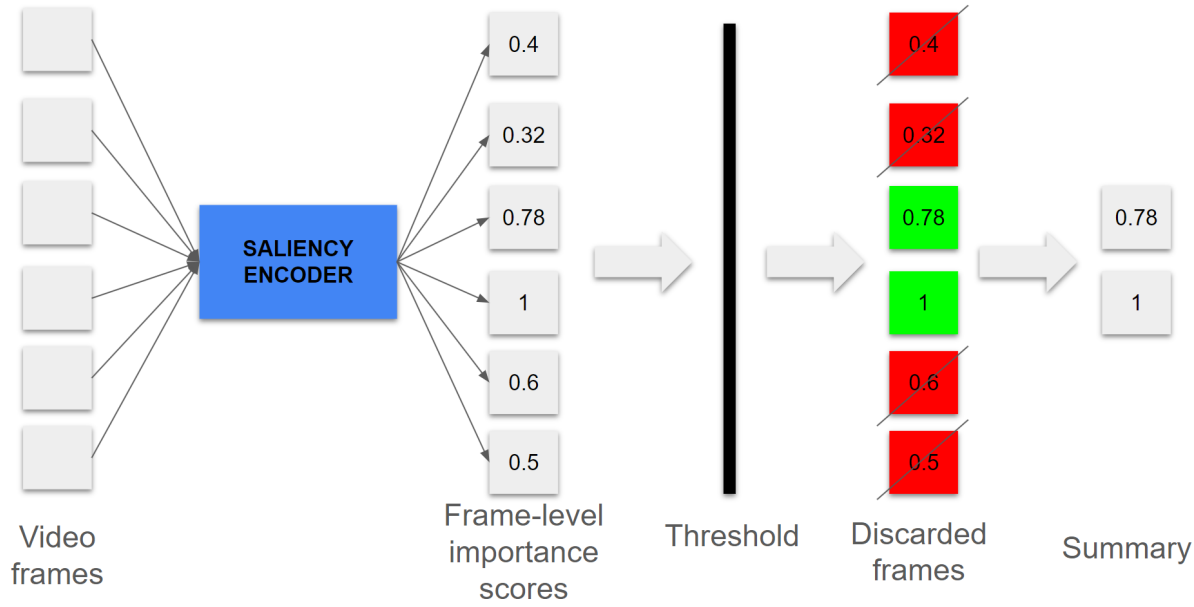


Figure 3.5: An example of the summarization process is shown where a frame-level score is assigned to the six initial frames. By applying a threshold of 0.75, four frames are discarded, leaving the two frames with the highest scores as the summary.

3.3 Fusion Module

In order to create the best summary possible, we developed different strategies to combine the results of the previous modules. It is important to note that state-of-the-art methods typically use a standard summary size of 15% of the video length. However, since our goal is to provide a quick overview of the entire endoscopy, we reduced this percentage to less than 2.5%.

3.3.1 Summarization only: UBISS-Uniform

The initial approach obtained frame-level importance scores and applied a threshold to generate the summary, as shown in Figure 3.5. Mei *et al.* feed the network using a video divided in small clips. The division consists on dividing the video in 2-second clips, resulting in frame segments with the same importance score. UBISS-Uniform simplifies the segments, selecting the middle frame from each to reduce the length of the final summary. This is taken into account when thresholding with the objective of the summary being a 2.5% of the original video length.

This is the most naive method, as it does not use the segmentation in any way. This method serves as a baseline to compare with our other strategies and to analyze the behavior of UBISS.

Figure 3.6: Example of a summary generated using the UBISS-Uniform strategy

3.3.2 Segmentation only

With the video segmentation from the method mentioned in [12], we explored how to generate a summary using this information.

SimpleSeg

The first idea selects a single frame from each segment of the segmentation, specifically the middle frame of each segment as depicted in Figure 3.7.

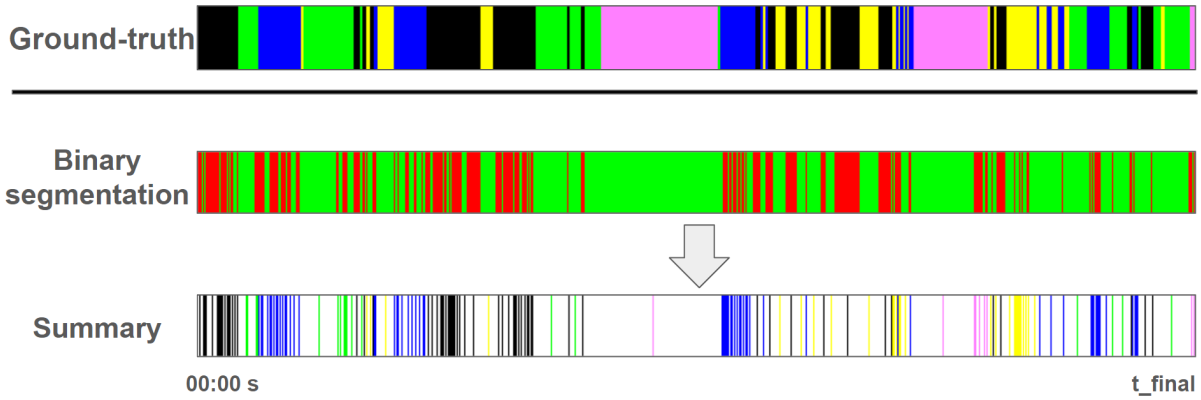


Figure 3.7: **SimpleSeg**: Summary generated using the binary segmentation of the video.

This is a naive approach on how to use the segmentation to generate a summary that does not take into account information like type of class or segment length.

PrioritySeg

In order to improve the results obtained using the segmentation, we made a distinction between *Good* segments and *Bad* segments. As mentioned in Section 3.1, *Tools*, *Good* and *Poor* are considered as *Good* segments; *Cleaning* and *Wall* are considered as *Bad* segments.

We created a class priority (*Tools* > *Good* > *Poor* > *None* > *Cleaning* > *Wall*) and we also prioritize longer segments over shorter ones. An additional restriction is that we discard the contribution of short-length segments. All the segments are sorted following these priorities and are iterated in order, counting the times a segment is visited until the length cap is reached. The number of times a segment is visited is the number of frames from that specific segment that will be added to the final summary. In order to give priority to *Good* segments, *Bad* segments are only visited once.

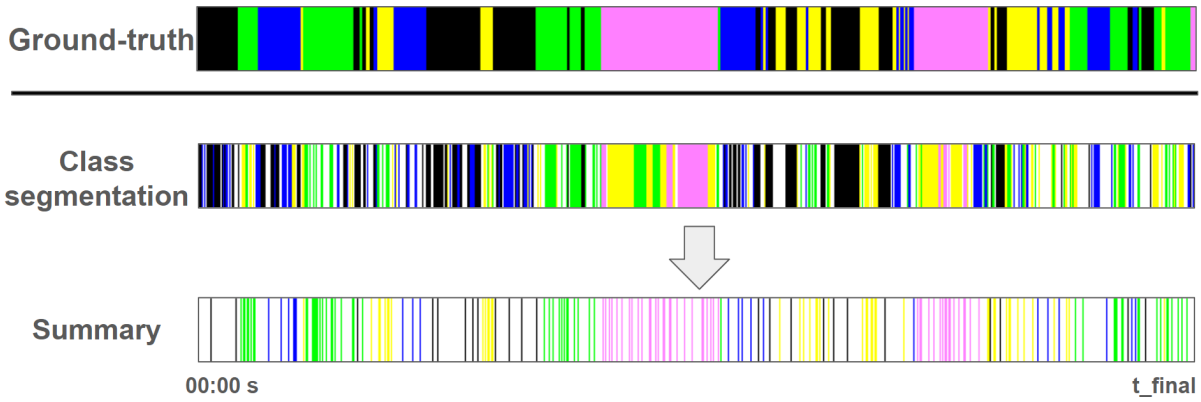


Figure 3.8: **PrioritySeg**: Summary generated using the class segmentation of the video, following class and size priorities.

This method will be used in the fusion module in order to generate a better summary by combining it with the summarization strategy described before.

3.3.3 Fusion strategy: UBISS-Filtered

The first fusion attempt uses the binary segmentation of the video as a filter to discard all the frames that are in *Bad* segments. An example is shown in Figure 3.9.

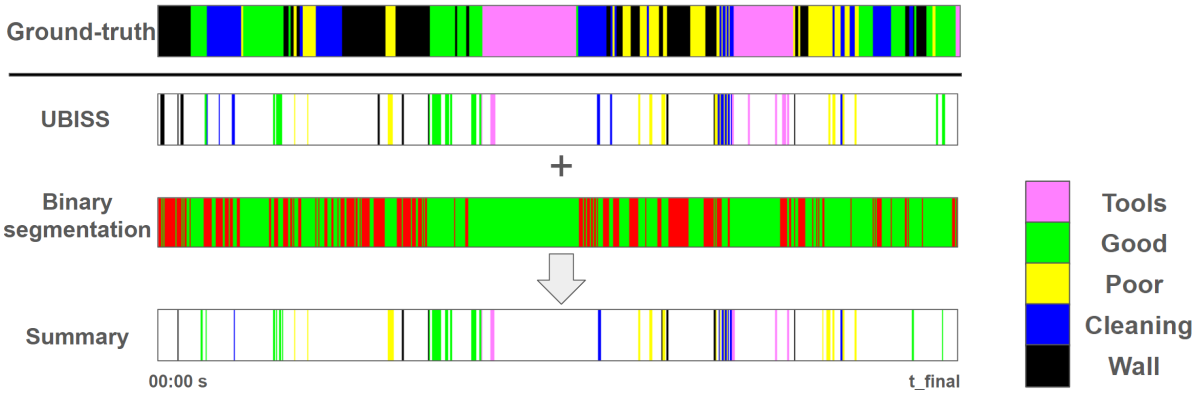


Figure 3.9: **UBISS-Filtered**: The inital summary obtained by the summarization module (UBISS) is filtered using the binary segmentation, discarding all the frames that are inside *Bad* segments.

3.3.4 Fusion strategy: SummSeg_v0

Using the summary obtained from **UBISS-Filtered** as an initial summary, we incorporated the **SimpleSeg** strategy method described in Section 3.3.2. We select the non-represented segments, i.e. segments whose frames are not in the initial summary. Then, we select one frame from each of the non-represented segments, specifically the middle frame of each one, and add those frames to the initial summary, building an enriched summary as depicted in Figure 3.10.

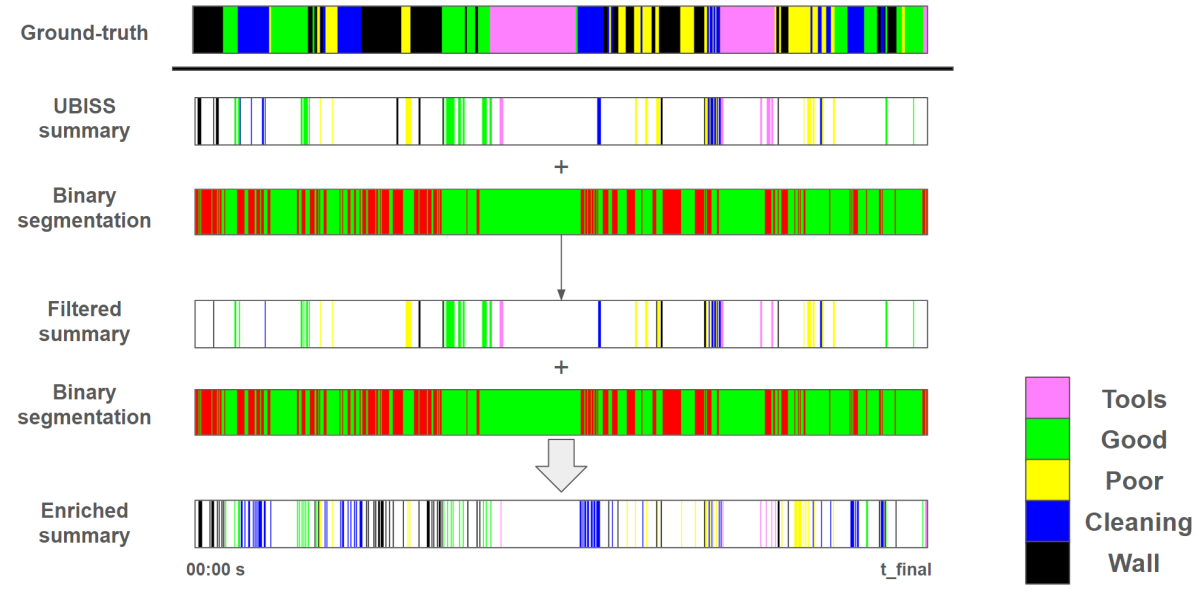


Figure 3.10: **SummSeg_v0**: The filtered summary obtained from **UBISS-Filtered** is enriched by adding frames from the non-represented segments of the binary segmentation.

This method generates a summary with a high amount of frames from *Bad* segments as we combined the naive segmentation strategy with our summarization strategy.

3.3.5 Fusion strategy: SummSeg

In order to improve the strategy described before, we upgraded the naive segmentation to the **PrioritySeg** strategy method described in Section 3.3.2. As mentioned before, **PrioritySeg** iterates the segments from the class segmentation and counts the number of times a segment is visited. Now that we have an initial summary as a base, every time a segment is visited we take into account the number of frames from this segment that are already in the summary. For instance, if a segment contains N frames in the summary, it must be visited at least $N+1$ times to contribute further to the summary, starting the count after the N-th visit.

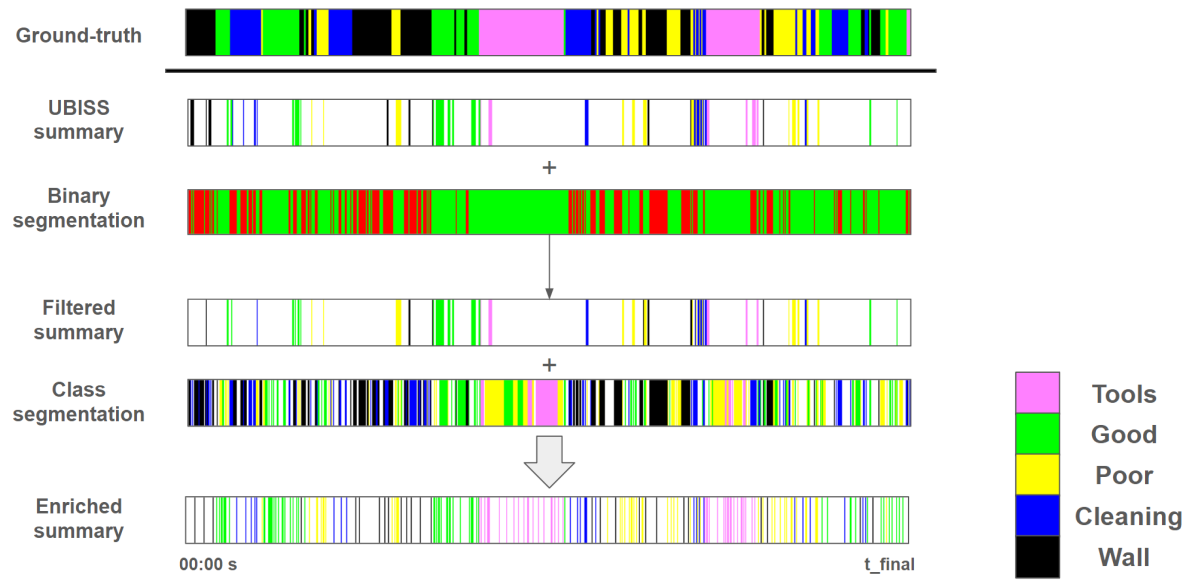


Figure 3.11: **SummSeg**: The filtered summary obtained from **UBISS-Filtered** is enriched by adding frames from the class segmentation following class and size priorities.

This is the final version of our approach, generating a summary that is closer to the stated goals of the project.

3.3.6 Fusion strategy: SummSegInv

We developed an alternative way to tackle the fusion of the segmentation and summarization strategies. We invert the process, starting from the binary segmentation, and we smooth the segmentation by eliminating segments with a legnth less than 10 frames. Once we have the smooth binary segmentation, we create a final summary by generating individual summaries for each segment fed into UBISS separately and then combining these segment summaries.

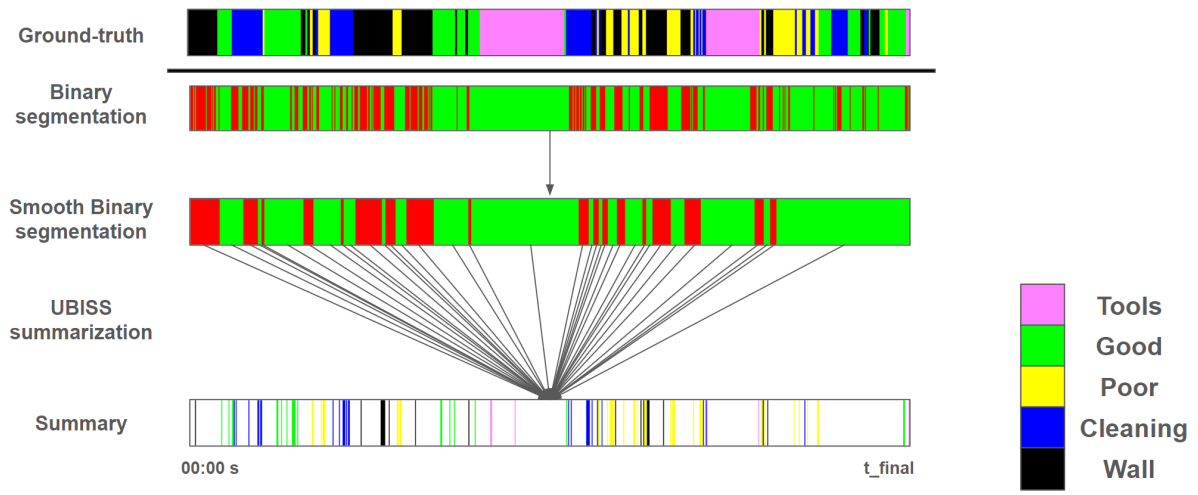


Figure 3.12: **SummSegInv**: The smooth segments are summarized using UBISS and then merged into the final summary.

Chapter 4

Experiments

4.1 Experimental setup

4.1.1 Dataset

The data used in the experiments of this work are from the EndoMapper dataset [1]. This public dataset contains 96 recordings of colonoscopy and gastroscopy procedures acquired during regular medical practice.

We selected 10 videos to perform the evaluation. With all of them we have observed good behaviour in our qualitative evaluation, and we have included a more thorough quantitative evaluation with videos *Seq_003_hd* and *Seq_036_hd* because they are provided with annotations for the different sections of the procedure as depicted in Figure 4.1. The other 8 videos that were selected to perform a qualitative evaluation are: *Seq_016_hd*, *Seq_022_hd*, *Seq_027_hd*, *Seq_034_hd*, *Seq_043_hd*, *Seq_076_hd*, *Seq_093_hd*, and *Seq_094_hd* .

Figure 4.1: Ground-truth annotations for the different sections of the endoscopy procedure *Seq_003_hd*.

4.1.2 Configuration of summarization models used

PGL-SUM. We run the PGL-SUM algorithm following the inference indications of their repository[1]. We selected the pretrained model *SumMe_table3*. For the Kernel Temporal Segmentation (KTS) we used the algorithm proposed by Ke *et al.* [13], using the code from their repository[2].

UBISS. We run the UBISS summarization algorithm following the inference indications of their repository[3]. We selected the pretrained model *UBiSS(NeuralNDCG, epoch=054)*.

[1] `https://github.com/e-apostolidis/PGL-SUM`
[2] `https://github.com/ChangPtR/D-KTS`
[3] `https://github.com/MeiYutingg/UBiSS`

4.1.3 Evaluation Metrics

In order to make a quantitative evaluation of the results of the experiments, we chose the following metrics:

- **Covered sections**: The number of sections from the ground truth represented in the summary, meaning that there is at least one frame from each segment included in the summary. This metric is computed separately for *Good* and *Bad* sections.
- **Class covering**: The count of ground truth sections of each class included in the summary.
- **Class compression rate**: Having the compression rate of a section s_i (CR_i) defined as:

$$CR_i = \frac{|s_i \cap summary|}{|s_i|}, \tag{4.1}$$

  where $|s_i|$ is the size in number of frames of a section s_i. The **Class compression rate** is the mean of the CR_i of the covered sections of a specific class.
- **Video compression rate**: The number of frames in the summary divided by the length of the original video.
- **Frame count**: Number of frames of a specific class represented in the summary.
- **Inference time**: The amount of time it takes for a trained model to process the input and produce an output, excluding the feature extraction part.

We also perform a qualitative evaluation by means of a compact visualization that gives a quick overview of the result, as shown in Figure 4.2. To clearly visualize the entire summary, this representation is condensed to one-tenth of the original video's length, meaning each colored line in the figure represents 10 frames selected for the summary.

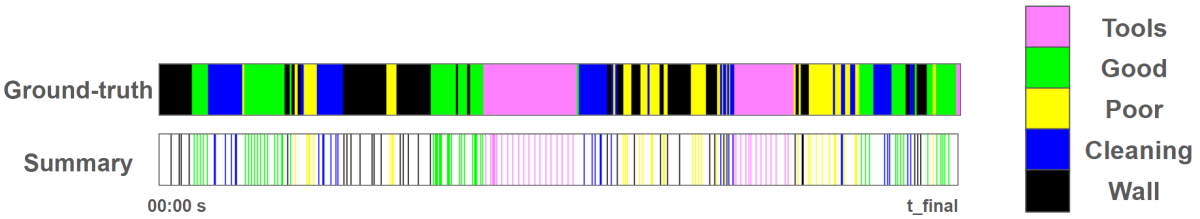


Figure 4.2: Example of the qualitative evaluation. Top row is the segmentation ground-truth of the video. Bottom row is a resulting summary. Each bar represents the duration of all the video and the colored intervals are the selected frames for the summary (white intervals represent segments that are not included in the summary).

4.2 Results

4.2.1 Summarization methods: PGL-SUM vs UBISS

The goal in this experiment is to evaluate on our data the two selected summarization strategies in order to select which one is more convenient for our system. The methods we will evaluate are PGL-SUM [8] and UBISS [7].

We ran both algorithms using as input the video *Seq_003_hd*, configuring both approaches as explained in Section 4.1.2.

Figure 4.3 includes a compact visualization of the summary obtained on video *Seq_003_hd*, which has available manual annotations about all the sections in the video, also displayed in the figure. As observed, UBISS summary looks more representative of the video since it includes frames from more sections compared to PGL-SUM. Additionally, the frames in PGL-SUM are closely clustered, causing significant redundancy, whereas UBISS distributes the frames more evenly.

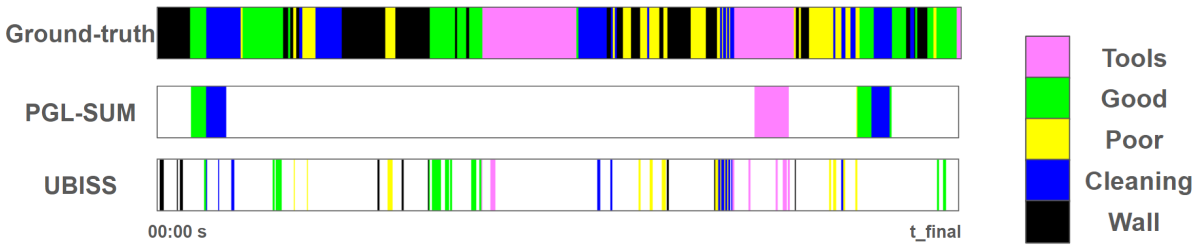


Figure 4.3: Visualization of resulting summary in video *Seq_003_hd* from PGL-SUM and UBISS next to the ground truth sections manually annotated. Each bar represents the duration of all the video and the colored intervals are the selected frames for the summaries.

Tables 4.1 and 4.2 and Figure 4.4 show more detailed results and analysis of these summaries, including the metrics described in Section 4.1.3. As we can observe in Table 4.1 UBISS summary covers more sections than the PGL-SUM summary. About the compression rates in Table 4.2, UBISS summary compresses more each section as its frames are more distributed and not as close as the frames from the PGL-SUM summary. A frame count distribution is shown in Figure 4.4 where we can tell that the distribution of UBISS is more uniform than PGL-SUM, which does not include any *Wall* frames at all. Finally, the inference time of UBISS is lower than that of PGL-SUM.

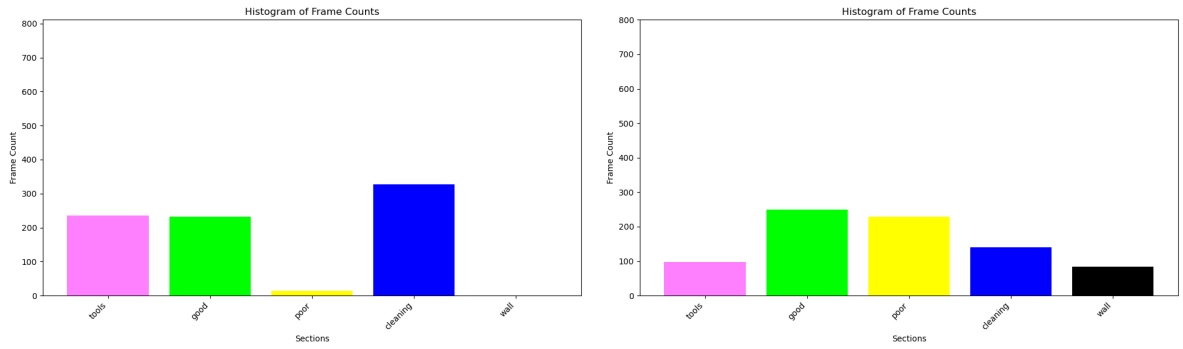
Following the results of this experiment, we decided to use UBISS for the summarization module. UBISS provides a result that aligns better with our goal, with a lower inference time and a simpler framework that does not require additional procedures to generate summaries.

Strategy	Covered sections		Class covering				
	Good	***Bad***	**Tools**	**Good**	**Poor**	**Cleaning**	**Wall**
PGL-SUM	5/36	2/35	1/3	3/12	1/21	2/16	0/19
UBISS	**22**/36	**16**/35	2/3	6/12	14/21	8/16	8/19

Table 4.1: **Covered sections** score for PGL-SUM and UBISS summaries of video *Seq_003_hd* regarding the *Good* and *Bad* sections covered, and the sections covered from each class.

Strategy	Class compression rate					Video	Inference
	Tools	**Good**	**Poor**	**Cleaning**	**Wall**	CR	Time
PGL-SUM	40.13%	53.93%	37.5%	71.34%	**0**%	15%	1.58 s
UBISS	7.36%	25.85%	48.05%	48.90%	7.94%	15%	0.34 s

Table 4.2: **Class compression rate**, **Inference time** and **Video compression rate** for the whole video applying PGL-SUM and UBISS on video *Seq_003_hd*.



PGL-SUM UBISS

Figure 4.4: Frame count distribution in the summary for video *Seq_003_hd* obtained with PGL-SUM and UBISS.

4.2.2 Fusion module evaluation

The objective of this experiment is to combine the summarization and segmentation strategies in order to build a summary that represents the highest amount of sections in a less overwhelming way, compressing the sections of the video without losing relevant information. We run this experiment on the video *Seq_003_hd* limiting the maximum summary size to 200 frames, in order to have fair comparisons. These methods leverage the output of the segmentation module, which takes 328 seconds to process the whole video (including the feature extraction).

The first approach is the **UBISS-Filtered** strategy described in Section 3.3.3, which filters the summary obtained from UBISS using the binary segmentation as depicted in Figure 4.5.

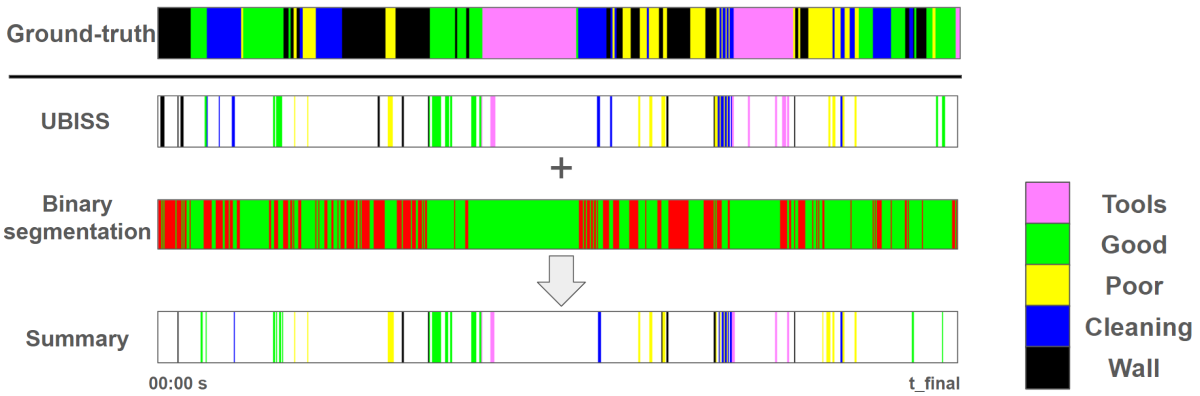


Figure 4.5: Fusion module from **UBISS-Filtered**. The inital summary obtained by the summarization module (UBISS) is filtered using the binary segmentation, discarding all the frames that are inside *Bad* segments.

This filter reduces the number of *Bad* frames. However, we also lose the representation of some *Good* sections.

In order to increase the representation of the segments that are not represented in the summary, we developed the **SummSeg_v0** strategy described in Section 3.10. This strategy adds one frame from every non-represented segment from the segmentation to the summary.

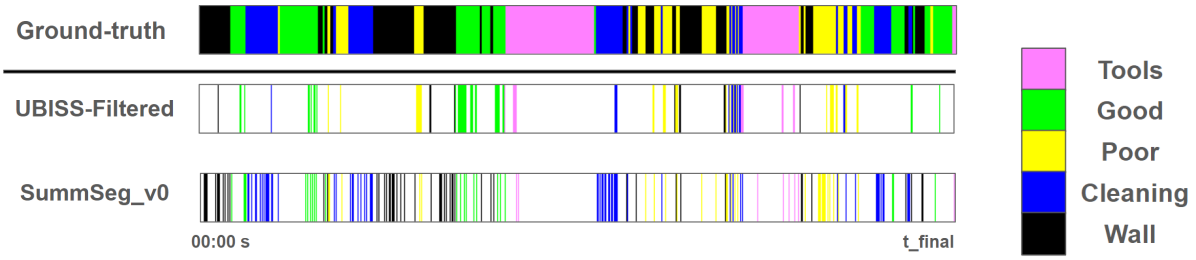
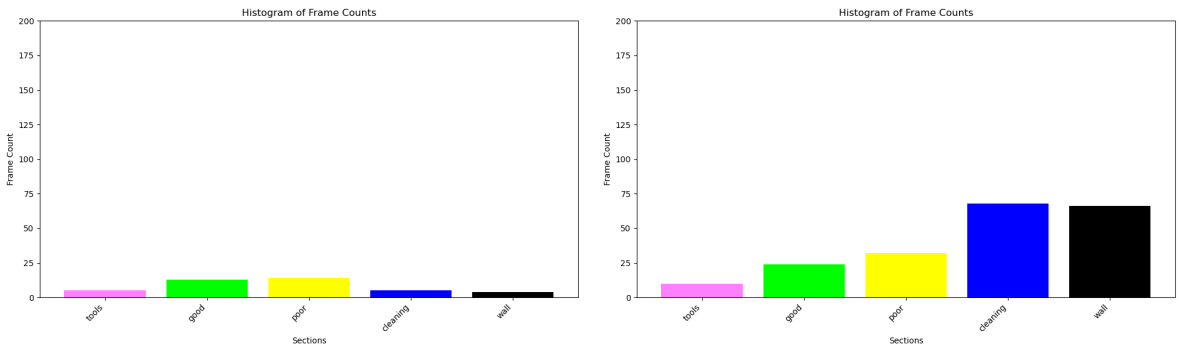


Figure 4.6: Summary results from the **UBISS-Filtered** and **SummSeg_v0** strategies.

Strategy	Covered sections		Class covering				
	Good	***Bad***	**Tools**	**Good**	**Poor**	**Cleaning**	**Wall**
UBISS-Filtered	21/36	9/35	2/3	5/12	13/21	5/16	4/19
SummSeg_v0	**26**/36	**33**/35	3/3	7/12	16/21	14/16	19/19

Table 4.3: Quantitative comparison of the **UBISS-Filtered** and **SummSeg_v0** summaries of video *Seq_003_hd* regarding the Good and Bad sections covered, and the sections covered per each class



UBISS-Filtered **SummSeg_v0**

Figure 4.7: Frame count distribution in the summary for video *Seq_003_hd* obtained with the **UBISS-Filtered** and **SummSeg_v0**

As we can observe in Table 4.3, **SummSeg_v0** covers more sections. However, as depicted in Figure 4.7, the number of *Bad* frames increased drastically, while the *Good* frames did not increase significantly. The reason for the high amount of *Bad* frames is because we only add one frame per segment in the segmentation regardless of the type of segment, not giving enough priority to the *Good* segments.

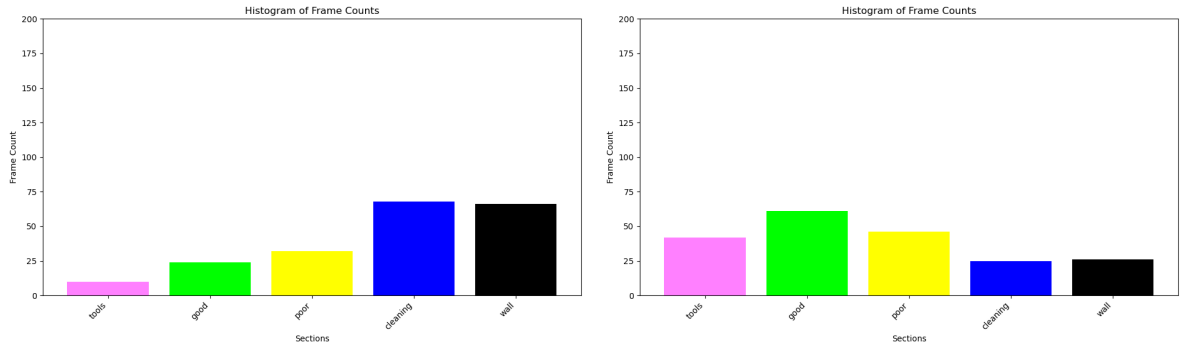
Our final version of the approach, **SummSeg** described in Section3.11 uses the class and size information in order to improve the results of the summary.

Strategy	Covered sections		Class covering				
	Good	***Bad***	**Tools**	**Good**	**Poor**	**Cleaning**	**Wall**
SummSeg_v0	26/36	33/35	3/3	7/12	16/21	14/16	19/19
SummSeg	**33**/36	27/35	2/3	12/12	19/21	13/16	14/19

Table 4.4: Quantitative comparison of **SummSeg_v0** and **SummSeg** summary of video *Seq_003_hd* regarding the Good and Bad sections covered, and the sections covered per each class

Strategy	Class compression rate					Video
	Tools	**Good**	**Poor**	**Cleaning**	**Wall**	compression rate
SummSeg_v0	1.17%	2.25%	4.90%	5.76%	3.93%	2.5%
SummSeg	**2.76**%	**3.79**%	4.54%	3.44%	1.8%	2.5%

Table 4.5: **Class compression rate** and **Video compression rate** applying **SummSeg_v0** and **SummSeg** on video *Seq_003_hd*.



SummSeg_v0 **SummSeg**

Figure 4.8: Frame count distribution in the summary for video *Seq_003_hd* obtained with the **SummSeg_v0** and **SummSeg**.

As we can observe in Table 4.4 **SummSeg** increases the number of *Good* sections. The number of sections covered is overall higher in **SummSeg**, and skewed towards *Good* segments. Regarding the compression rates seen in Table 4.5, the compression rates for the *Tools* and *Good* sections are higher in **SummSeg** while the compression rates of *Cleaning* and *Wall* are lower. This results in a summary that has more representation of *Good* frames overall and less representation for *Bad* frames, as we can observe in Figure 4.8.

4.2.3 Ablation

The goal of this experiment is to evaluate the contribution of each module of the approach.

The comparison is done between the **UBISS-Unifrom** summarization algorithm described in Section 3.3.1, the **PrioritySeg** algorithm described in 3.3.2, the **SummSeg** algorithm described in Section 3.11 and the **SummSegInv** algorithm described in Section 3.12. We ran these three algorithms on the video *Seq_003_hd* and obtained the following results. In order to have fair comparison, all the summaries generated have the same length, in this case 200 frames.

Strategy	Covered sections		Class covering				
	Good	***Bad***	**Tools**	**Good**	**Poor**	**Cleaning**	**Wall**
UBISS-Unifrom	27/36	24/35	2/3	7/12	18/21	13/16	11/19
PrioritySeg	30/36	24/35	2/3	12/12	16/21	10/16	14/19
SummSeg	**33**/36	**27**/35	2/3	12/12	19/21	13/16	14/19
SummSegInv	27/36	16/35	3/3	6/12	18/21	7/16	9/19

Table 4.6: Quantitative comparison of each module summary of video *Seq_003_hd* regarding the *Good* and *Bad* sections covered, and the sections covered from each class.

Strategy	Class compression rate					Video compression rate
	Tools	**Good**	**Poor**	**Cleaning**	**Wall**	
UBISS-Unifrom	2.35%	5.35%	6.03%	6.26%	1.96%	2.5%
PrioritySeg	2.10%	4.06%	4.04%	1.53%	1.22%	2.5%
SummSeg	**2.76**%	3.79%	4.54%	3.44%	1.8%	2.5%
SummSegInv	1.00%	3.10%	10.16%	10.0%	3.37%	2.5%

Table 4.7: **Class compression rate** and **Video compression rate** for the whole video applying each module strategy on video *Seq_003_hd*.

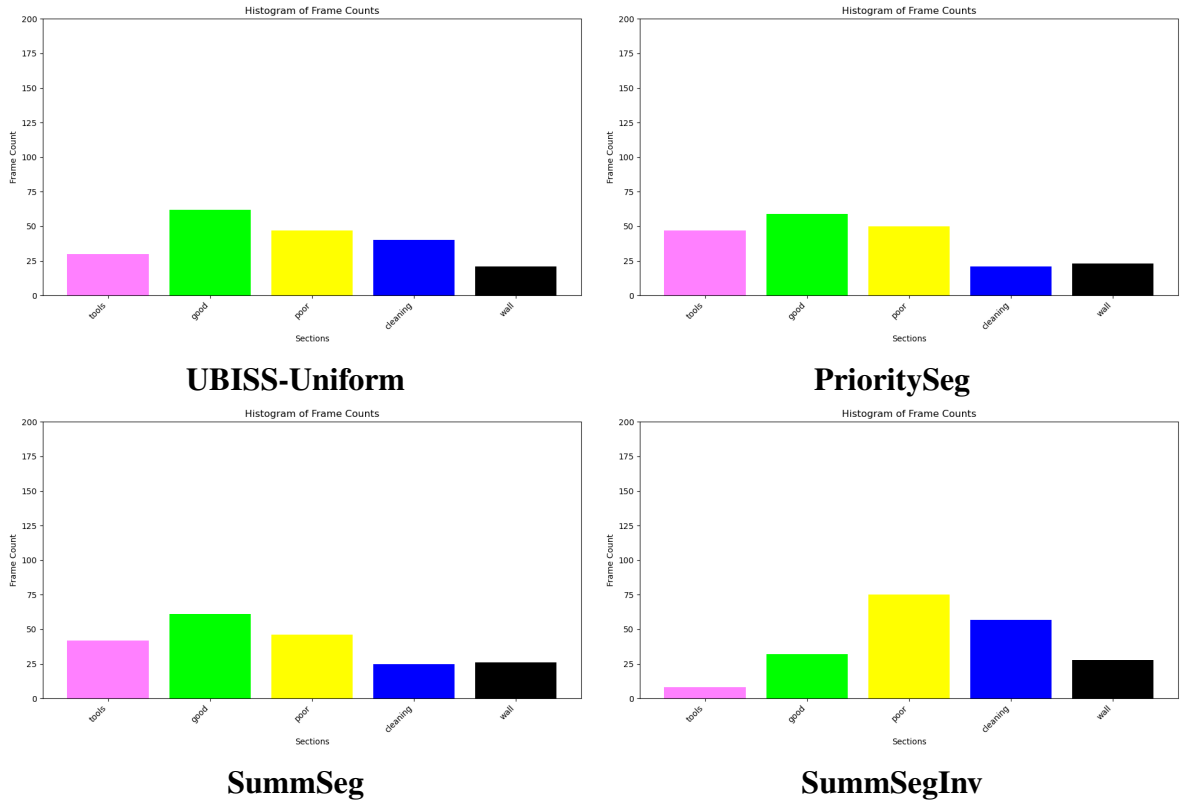


Figure 4.9: Frame count distribution in the summary for video *Seq_003_hd* obtained with each module strategy.

As we can observe in Table 4.6, **SummSeg** covers more sections than any other strategy. In Table 4.7 **SummSeg** compresses *Tools* sections less, which gives more representation to

the most important type of section. Regarding the frame distribution depicted in Figure 4.9, **UBISS-Uniform** and **SummSegInv** have the highest amount of *Bad* frames, with low *Good* frame representation. On the other hand, **SummSeg** and **PrioritySeg** have similar frame distributions, but, as commented before, **SummSeg** covers more sections with acceptable compression levels.

In conclusion, **SummSeg** achieves a more suitable solution for our goals combining the outputs from both the summarization and segmentation modules.

4.2.4 Approach evaluation

In this experiment we evaluate the approach developed and check if it works similarly in other test sequences compared to a uniform solution.

We compared our approach with a naive solution that consists on building a summary by sampling uniformly the frames from the video as depicted in Figure 4.10.

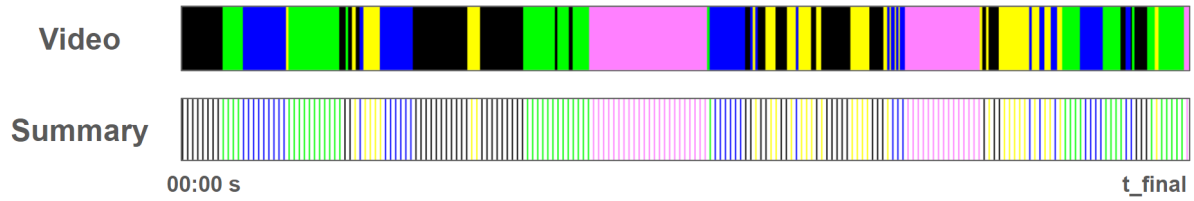


Figure 4.10: Example of the summary obtained by sampling uniformly the frames from the video

We ran **SummSeg** and the uniform sampling algorithm on videos *Seq_003_hd* and *Seq_036_hd*. A representation of the different sections of both videos is shown in Figure 4.11. Looking at the sizes and frequencies of the class labels, we observe that *Seq_036_hd* is very different from *Seq_003_hd*, adding diversity and robustness to our analysis.

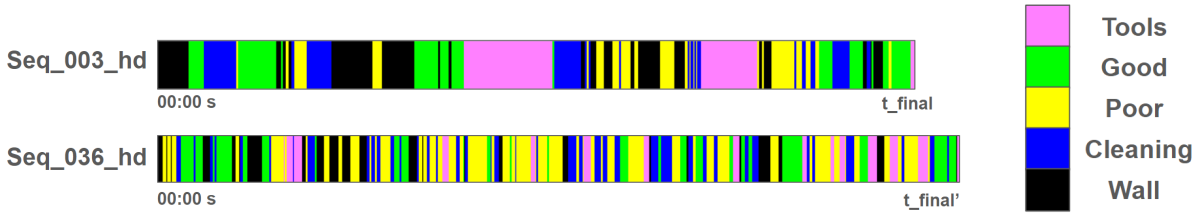


Figure 4.11: Representation of different sections of videos *Seq_003_hd* and *Seq_036_hd*

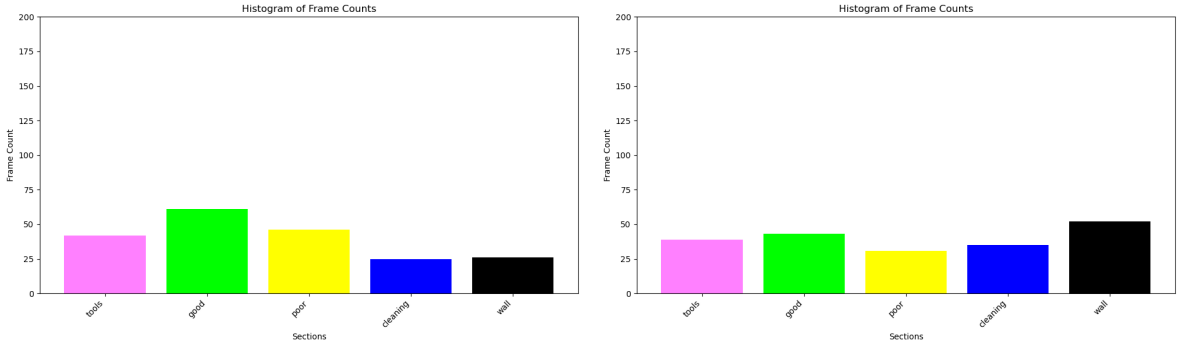
Seq_003_hd	Covered sections		Class covering				
	Good	***Bad***	**Tools**	**Good**	**Poor**	**Cleaning**	**Wall**
SummSeg	**33**/36	**27**/35	2/3	12/12	19/21	13/16	14/19
Uniform	29/36	**27**/35	3/3	10/12	16/21	13/16	14/19
Seq_036_hd	Covered sections		Class covering				
	Good	***Bad***	**Tools**	**Good**	**Poor**	**Cleaning**	**Wall**
SummSeg	66/87	39/65	13/15	19/25	34/47	24/40	15/24
Uniform	72/87	50/65	13/15	20/25	39/47	31/40	19/24

Table 4.8: Quantitative comparison of **SummSeg** and uniform sampling of videos *Seq_003_hd* and *Seq_036_hd* regarding the *Good* and *Bad* sections covered, and the sections covered from each class.

Seq_003_hd	Class compression rate					Video
	Tools	**Good**	**Poor**	**Cleaning**	**Wall**	compression rate
SummSeg	**2.76**%	**3.79**%	**4.54**%	**2.98**%	**1.80**%	2.5%
Uniform	2.10%	2.89%	2.89%	3.16%	2.94%	2.5%
Seq_036_hd	Class compression rate					Video
	Tools	**Good**	**Poor**	**Cleaning**	**Wall**	compression rate
SummSeg	**2.44**%	**3.38**%	**2.28**%	**2.01**%	**2.01**%	2.5%
Uniform	1.96%	2.25%	2.21%	2.31%	2.59%	2.5%

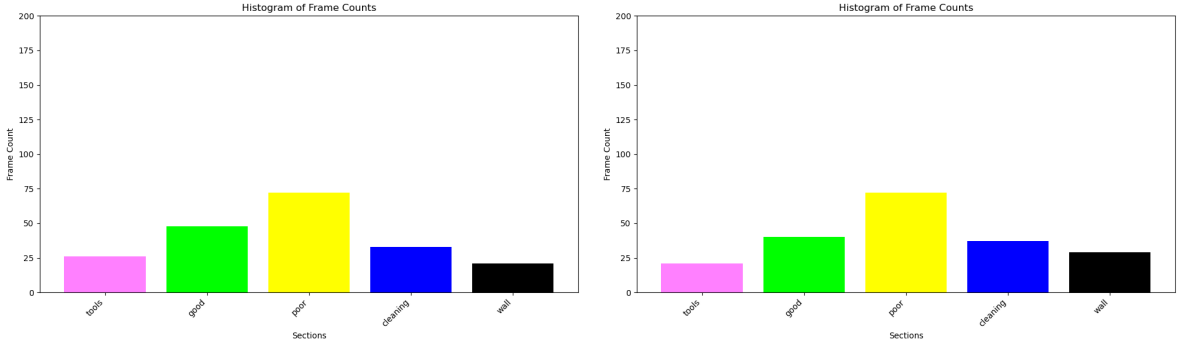
Table 4.9: **Class compression rate** and **Video compression rate** for the whole video applying **SummSeg** and uniform sampling on videos *Seq_003_hd* and *Seq_036_hd*.

Seq_003_hd



SummSeg **Uniform**

Seq_036_hd



SummSeg **Uniform**

Figure 4.12: Frame count distribution in the summary for videos *Seq_003_hd* and *Seq_036_hd* obtained with **SummSeg** and uniform sampling.

As we can observe in Table 4.8, **SummSeg** covers more sections in video *Seq_003_hd*, but covers a few less sections in video *Seq_036_hd*. However, as depicted in Table 4.9, **SummSeg** compresses the *Good* sections less, increasing their representation, whereas it compresses the *Bad* sections more, reducing their importance but maintaining their representation. Regarding the frame distribution, **SummSeg** has a higher amount of *Good* frames in both cases, while uniform sampling has more representation of *Bad* frames.

4.2.5 Qualitative evalutaion

The objective of this experiment is to evaluate the results of our approach in a qualitative way, by observing a mosaic with all the frames from the summaries.

We ran **SummSeg** on 10 videos of the EndoMapper dataset and generated an overview of the summaries. In this section, we will cover videos *Seq_003_hd* and *Seq_036_hd*. The rest of the videos can are included in Appendix A.

Seq_003_hd

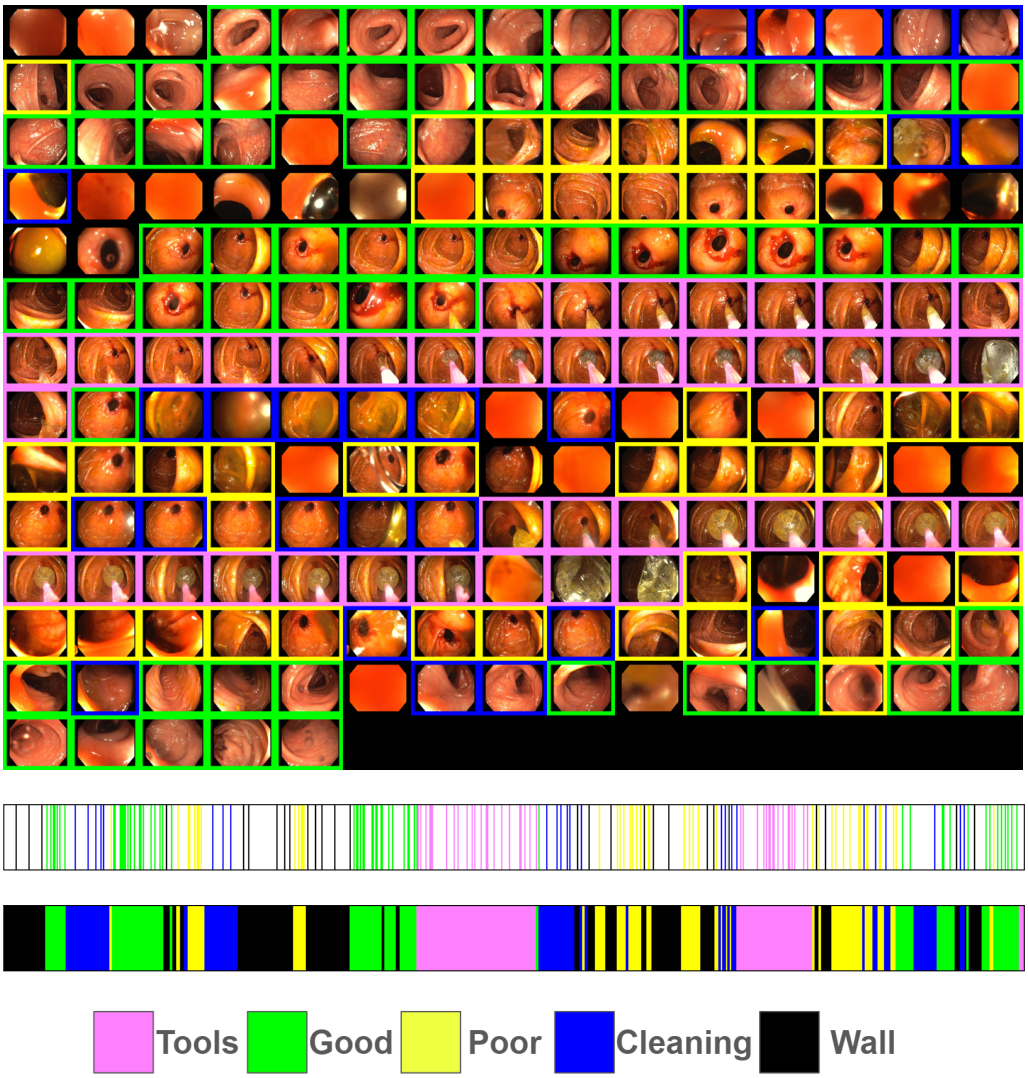


Figure 4.13: Visual representation of the summary obtained with **SummSeg** on video *Seq_003_hd*. Bottom row shows the ground-truth annotations with the class color legend.

As we can observe in Figure 4.13, all the *Wall* and *Cleaning* sections are covered with a few amount of frames, whereas *Tools*, *Good* and *Poor* sections are highly represented. The overview demonstrates minimal redundancy in the frames, effectively covering the sections with distinct frames.

Seq_036_hd

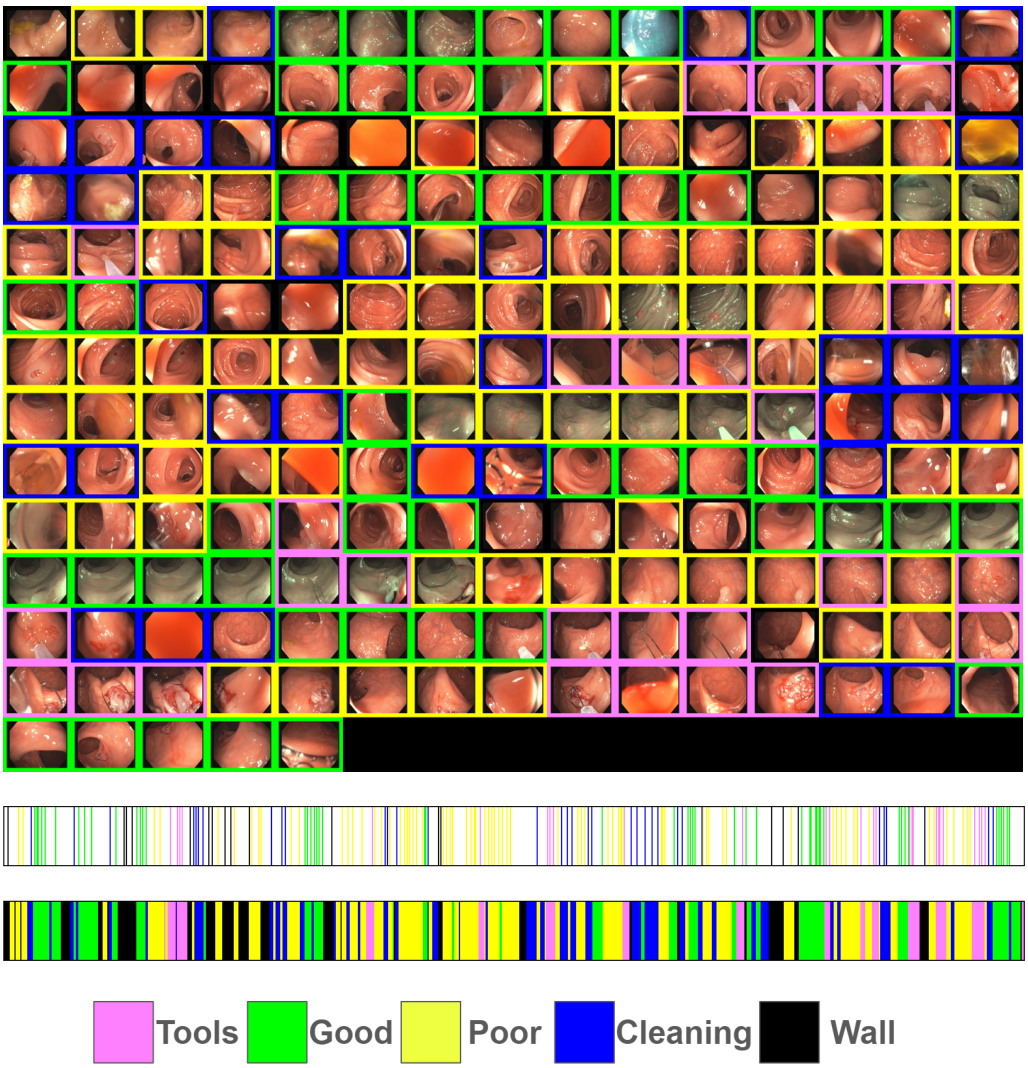


Figure 4.14: Visual representation of the summary obtained with **SummSeg** on video *Seq_036_hd*. Bottom row shows the ground-truth annotations with the class color legend.

Seq_036_hd is a procedure containing short sections with many medical interventions, shown as *Tools* sections. Like in the video *Seq_003_hd*, Figure 4.14 shows all the *Wall* and *Cleaning* sections are represented with a few amount of frames, whereas *Tools*, *Good* and *Poor* sections are highly represented.

Additional evaluations

Regarding the rest of the videos, since we do not have the annotations of the different sections for them, the colors of the labels are obtained using the automatic segmentation approach used in this work. It is important to take into account that there is an additional label *None*, depicted in gray, that refers to frames where the approach was not confident enough to clearly classify those segments.

The detailed results and figures are in Appendix A. In all these qualitative overview figures, we can observe the main limitations and positive aspects of our approach: the frames gathered

in the summaries are observing clearly the different parts of the colon and less than 10% of the frames have low visibility, i.e *Bad* frames.

Chapter 5

Conclusions, challenges and future work

5.1 Conclusions

We have shown that current state-of-the-art methods on video summarization are not directly valid to process endoscopies in a way that can be helpful for physicians. However, it is possible to adapt them in order to generate acceptable solutions. We have also shown that it is possible to combine video summarization and video segmentation strategies to tackle the goal of making an overview of endoscopy procedures in a less overwhelming manner.

We conducted experiments adapting and comparing state-of-the-art video summarization methods to the endoscopy domain. Our study focused on taking advantage of a semantic video segmentation approach to enhance video summarization results, and combining both strategies to provide a better overview of the procedure. We developed evaluation metrics adapted to our data in order to evaluate video summarization state-of-the-art methods in the endoscopy domain. Additionally, we adapted one state-of-the-art method and combined it with a video segmentation method in order to achieve a summary that provides an overview of the procedure in a less overwhelming manner.

All in all, this work serves as the starting point in a research project to automatically obtain an overview of an endoscopy procedure. Next, we discuss the main challenges found and possible new directions to continue this work.

5.2 Challenges and limitations

The challenges and limitations encountered during this work were the following ones. Firstly, this was the first time I tackled such a complex problem, which involved learning about endoscopy procedures, video summarization methods and video segmentation methods, and then integrating them into a summary generation method. Another challenge to consider was evaluating the results obtained throughout the work. Assessing such a specific task required considerable time before achieving a satisfactory evaluation method to compare our solutions. Additionally, understanding the pipeline code from the video summarization methods compared in this work.

I confronted the limitation of developing a machine learning framework with very few annotated data (2 videos). This restriction prevented us from re-training the summarization model and made it difficult to conduct more comprehensive quantitative experiments and evaluations.

5.3 Future work

Based on the limitations and challenges encountered in this work, there are several future directions that can be pursued. One possible direction is to train the summarization model to adapt it to the endoscopy domain, improving the results of the summary. This training would require manual annotation of many videos from the EndoMapper dataset.

Regarding the video segmentation part of this work, we used a video segmentation approach that used unsupervised learning. One possible direction could be studying the possibility of developing a supervised model using the annotations previously mentioned. Another potential direction is to leverage the approach's results to generate medical reports or assist professionals in creating their reports.

Finally, it may be worthwhile to investigate the applications of video summarization in other medical procedures such as biopsies or tomographies.

Bibliography

[1] P. Azagra et al. Endomapper dataset of complete calibrated endoscopy procedures. *Scientific Data*, 10(671), 2023.

[2] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pages 766–782. Springer, 2016.

[3] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Hsa-rnn: Hierarchical structure-adaptive rnn for video summarization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7405–7414, 2018.

[4] Mrigank Rochan, Linwei Ye, and Yang Wang. Video summarization using fully convolutional sequence networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 347–363, 2018.

[5] Junbo Wang, Wei Wang, Zhiyong Wang, Liang Wang, Dagan Feng, and Tieniu Tan. Stacked memory network for video summarization. In *Proceedings of the 27th ACM international conference on multimedia*, pages 836–844, 2019.

[6] Evlampios Apostolidis, Georgios Balaouras, Vasileios Mezaris, and Ioannis Patras. Combining global and local attention with positional encoding for video summarization. In *2021 IEEE International Symposium on Multimedia (ISM)*, pages 226–234, December 2021.

[7] Yuting Mei, Linli Yao, and Qin Jin. Ubiss: A unified framework for bimodal semantic summarization of videos. In *Proceedings of the 2024 International Conference on Multimedia Retrieval*, pages 1034–1042, 2024.

[8] Baopu Li, Max Q-H Meng, and Qian Zhao. Wireless capsule endoscopy video summary. In *2010 IEEE International Conference on Robotics and Biomimetics*, pages 454–459. IEEE, 2010.

[9] M. Maher Ben Ismail, Ouiem Bchir, and Ahmed Z. Emam. Endoscopy video summarization based on unsupervised learning and feature discrimination. In *2013 Visual Communications and Image Processing (VCIP)*, pages 1–6, 2013.

[10] B. Sushma and P. Aparna. Automated summarization of gastrointestinal endoscopy video. In Eunika Mercier-Laurent, Xavier Fernando, and Aravindan Chandrabose, editors, *Computer, Communication, and Signal Processing. AI, Knowledge Engineering and IoT for Smart Systems*, pages 27–35, Cham, 2023. Springer Nature Switzerland.

[11] Jin Chen, Yuexian Zou, and Yi Wang. Wireless capsule endoscopy video summarization: A learning approach based on siamese neural network and support vector machine. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 1303–1308, 2016.

[12] O León Barbed, Cristina Oriol, Pablo Azagra Millán, and Ana C Murillo. Semantic analysis of real endoscopies with unsupervised learned descriptors. In *Medical Imaging with Deep Learning*, 2022.

[13] Xiaopeng Ke, Boyu Chang, Hao Wu, Fengyuan Xu, and Sheng Zhong. Towards practical and efficient long video summary. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1770–1774, 2022.

Appendix A

Additional Results

A.1 EndoMapper videos

A.1.1 Seq_016_hd

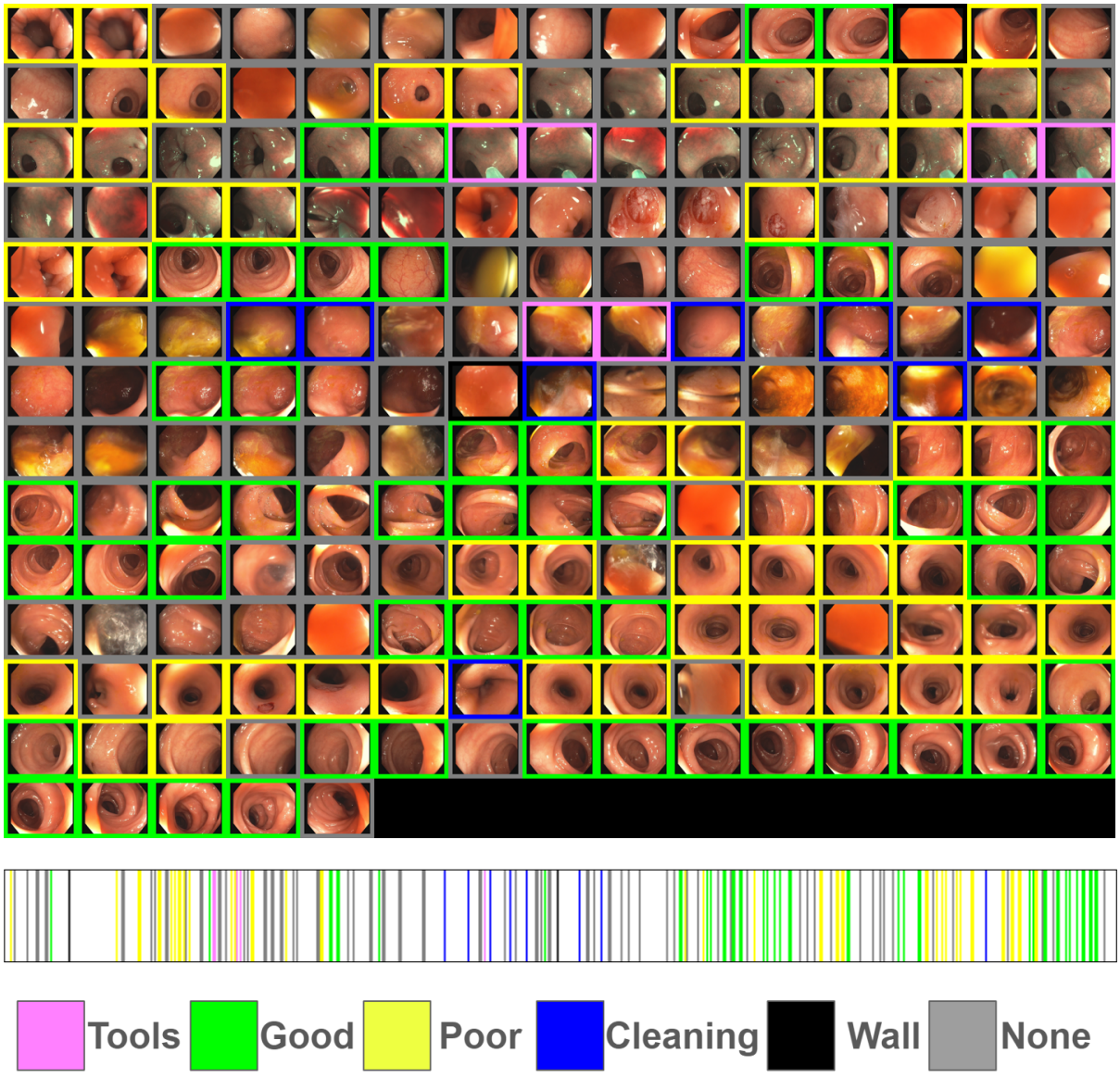


Figure A.1: Qualitative representation of the summary obtained with **SummSeg** on video *Seq_016_hd*.

A.1.2 Seq_022_hd

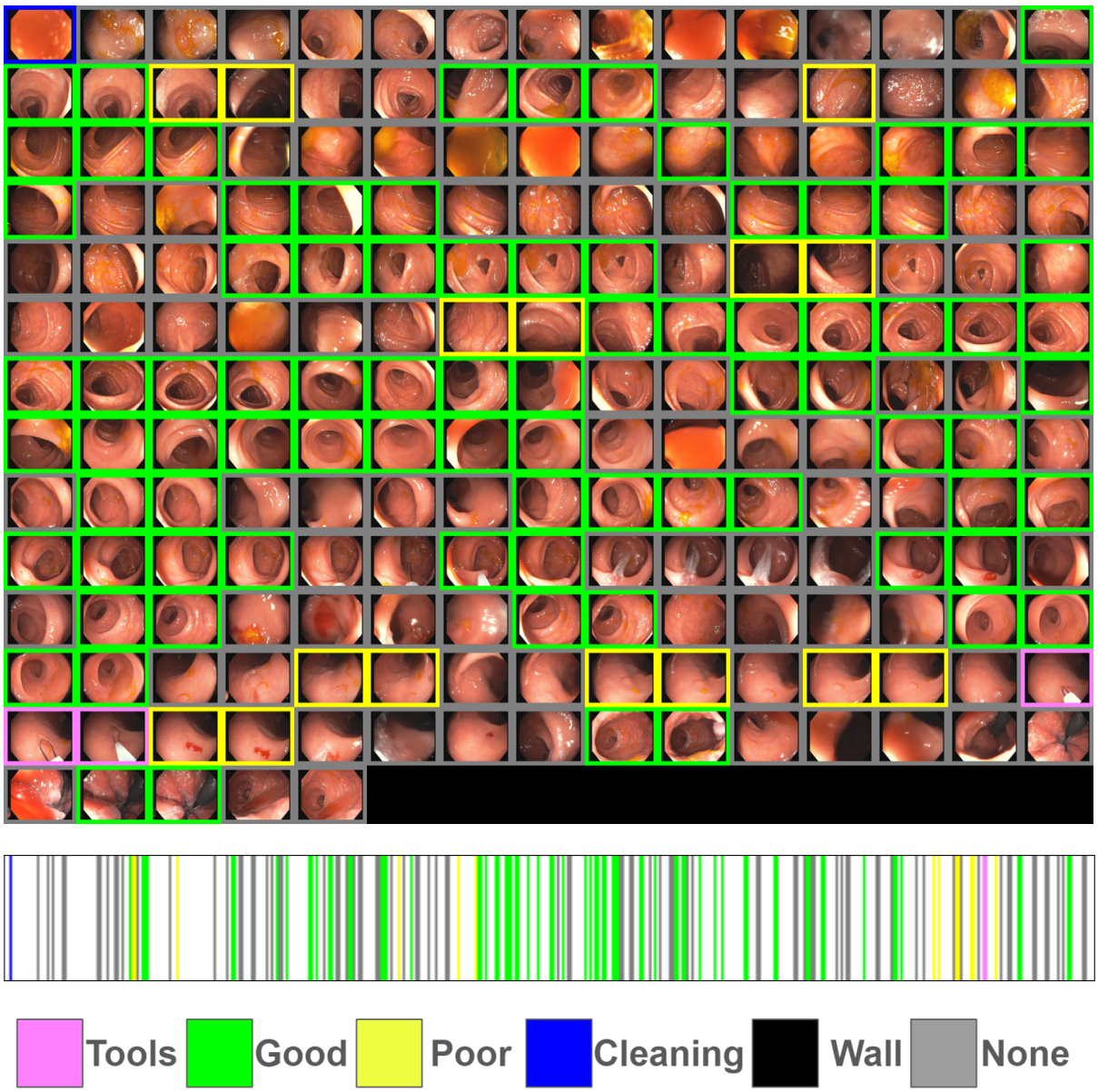


Figure A.2: Qualitative representation of the summary obtained with **SummSeg** on video *Seq_022_hd*.

A.1.3 Seq_027_hd

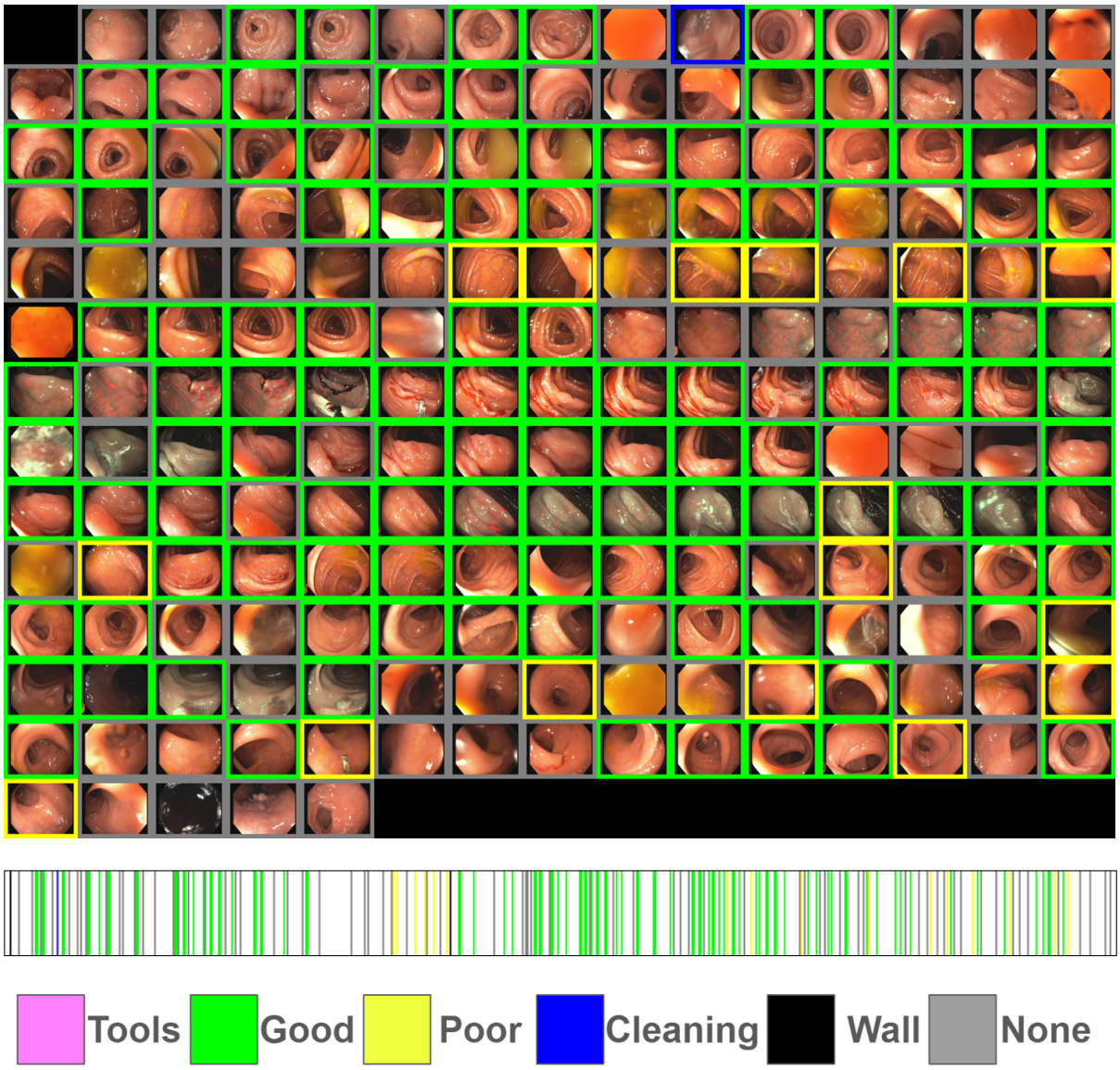


Figure A.3: Qualitative representation of the summary obtained with **SummSeg** on video *Seq_027_hd*.

A.1.4 Seq_034_hd

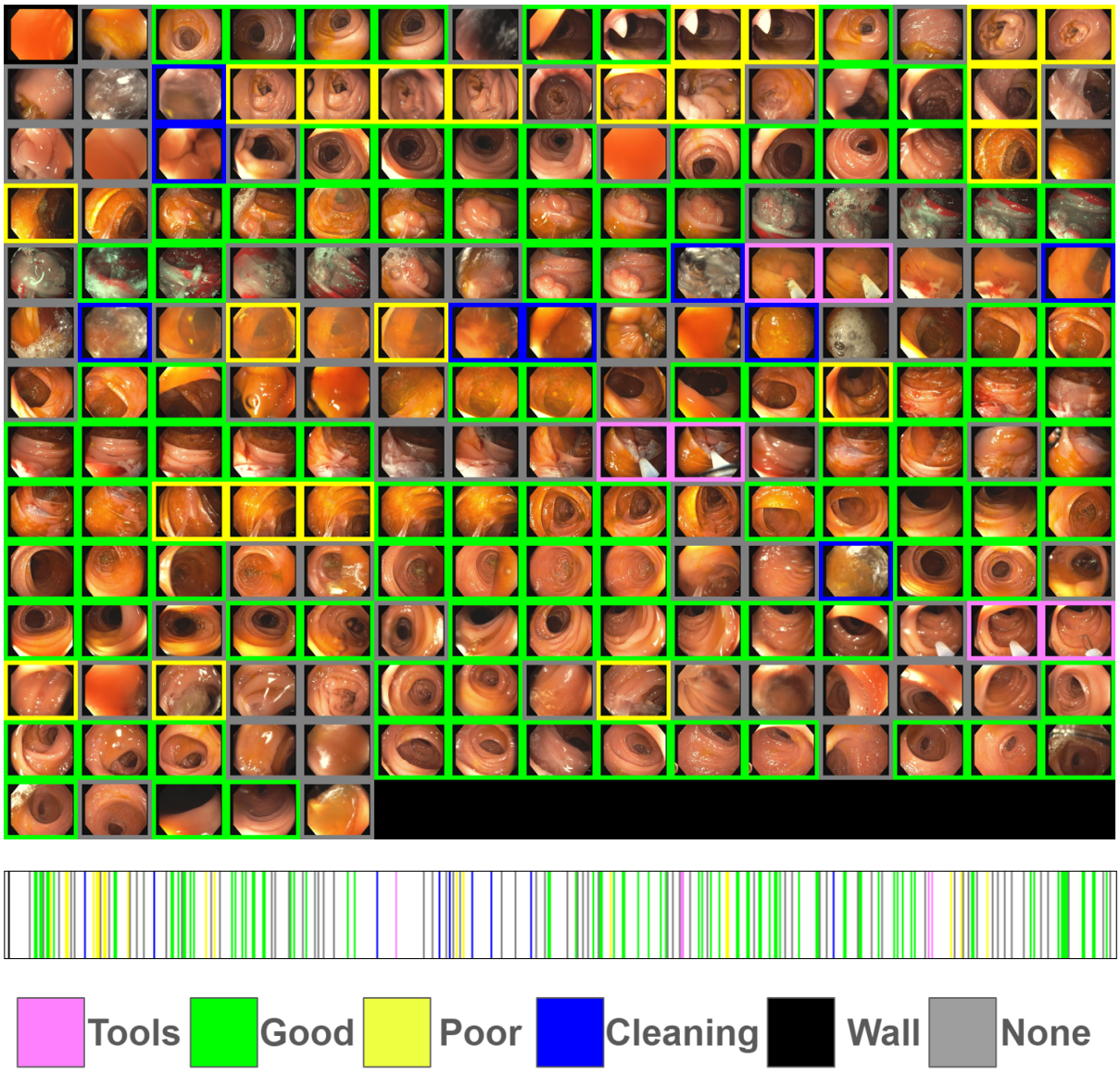


Figure A.4: Qualitative representation of the summary obtained with **SummSeg** on video *Seq_034_hd*.

A.1.5 Seq_043_hd

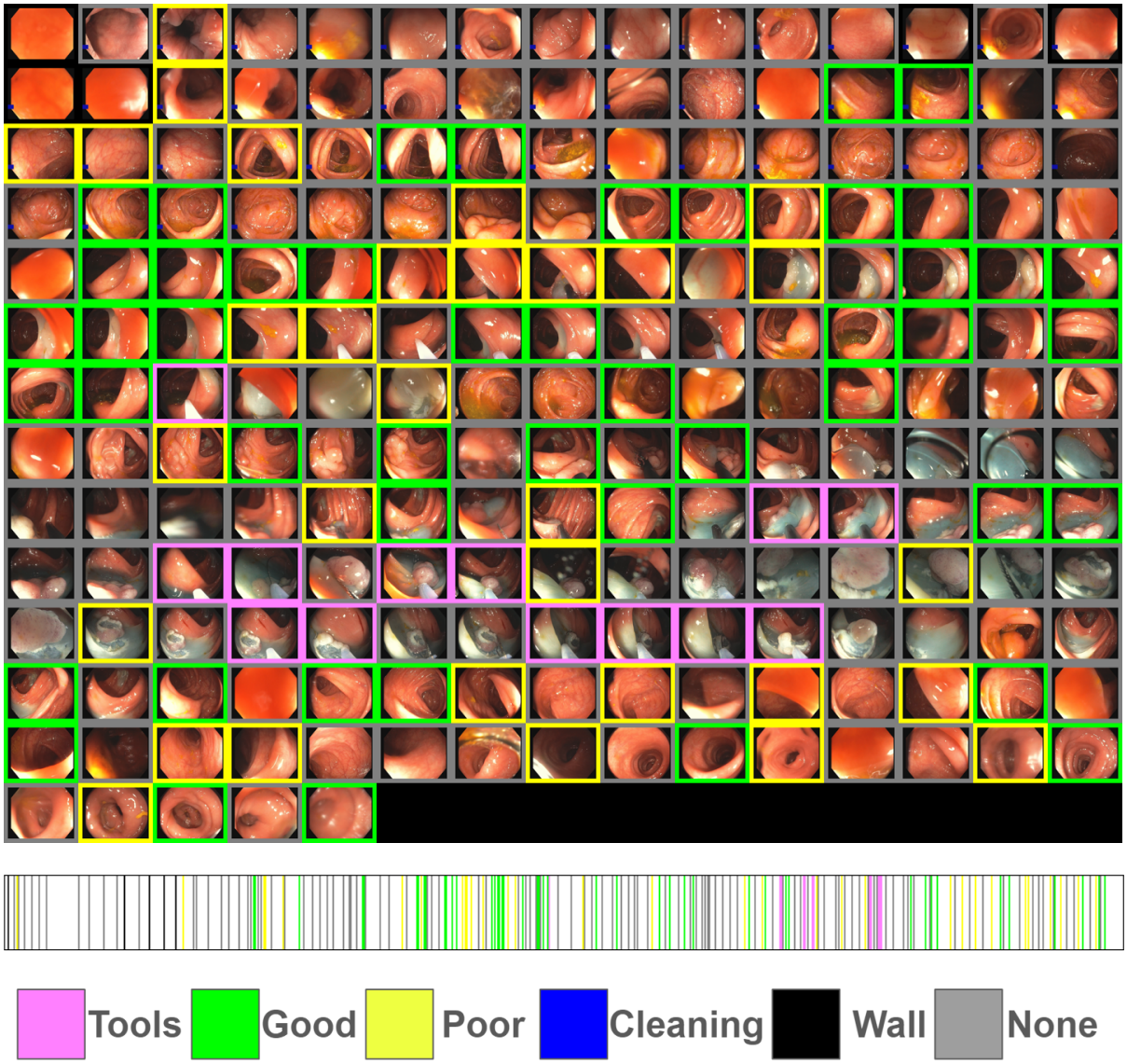


Figure A.5: Qualitative representation of the summary obtained with **SummSeg** on video *Seq_043_hd*.

A.1.6 Seq_076_hd

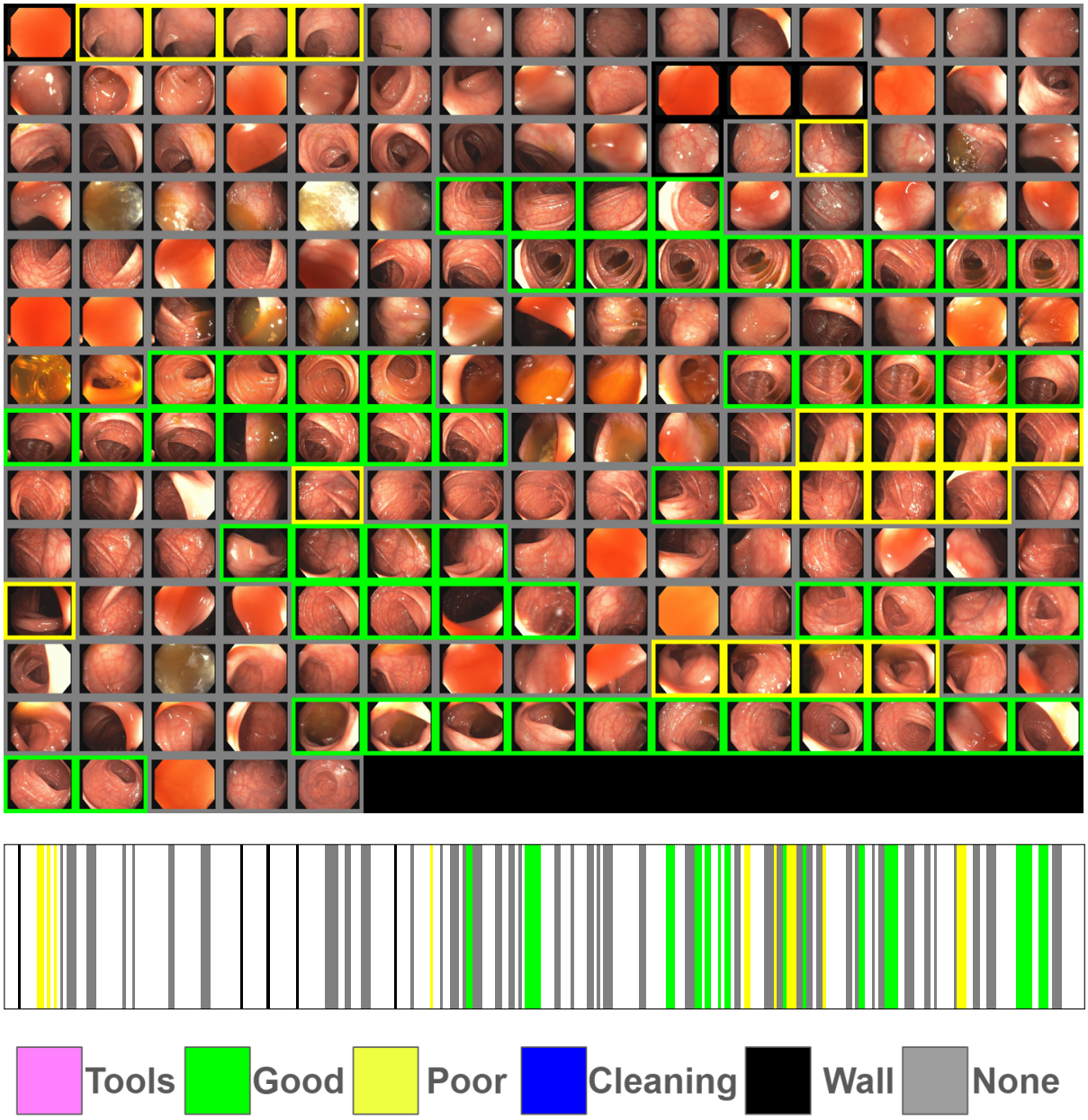


Figure A.6: Qualitative representation of the summary obtained with **SummSeg** on video *Seq_076_hd*.

A.1.7 Seq_093_hd

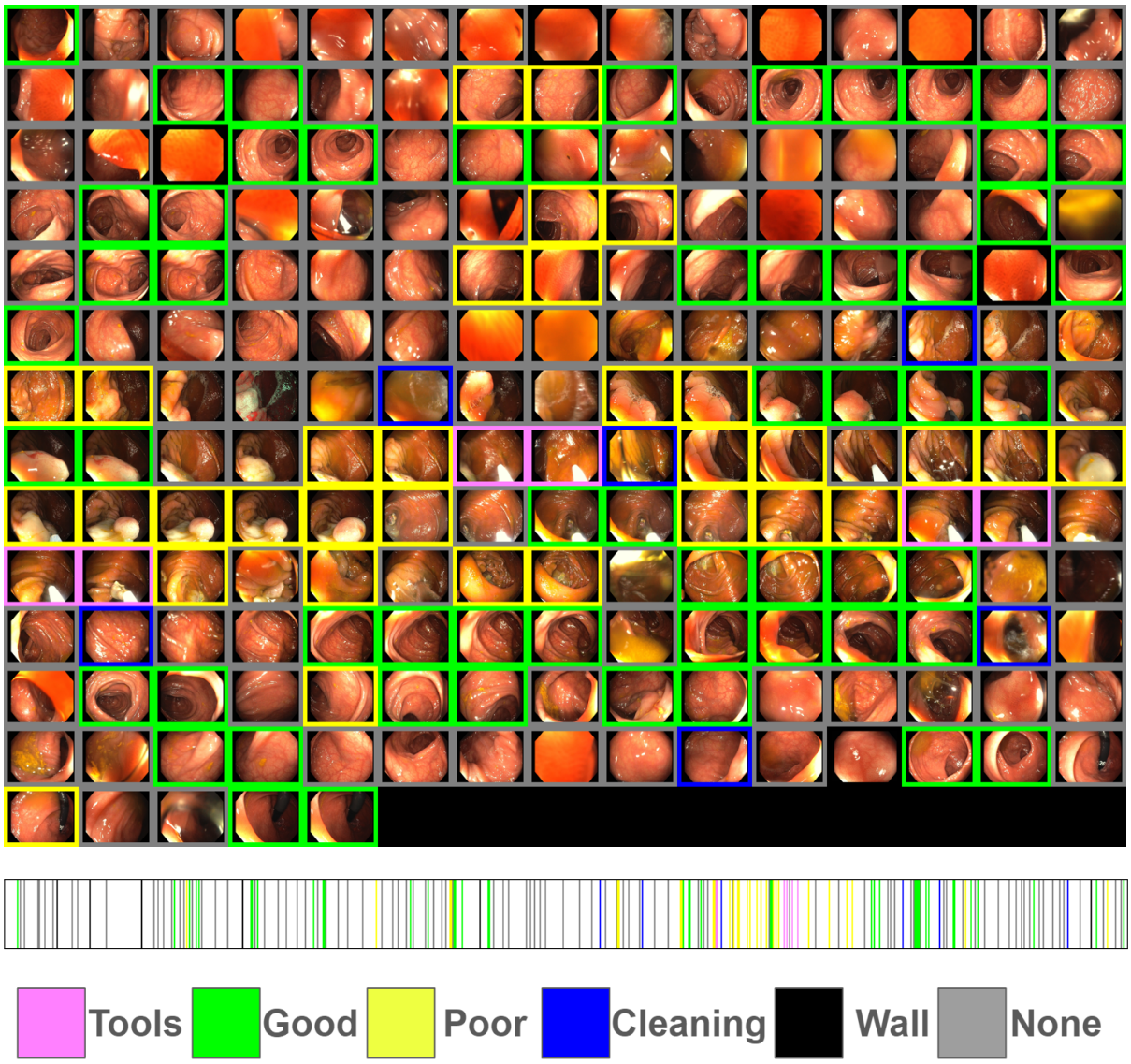


Figure A.7: Qualitative representation of the summary obtained with **SummSeg** on video *Seq_093_hd*.

A.1.8 Seq_094_hd

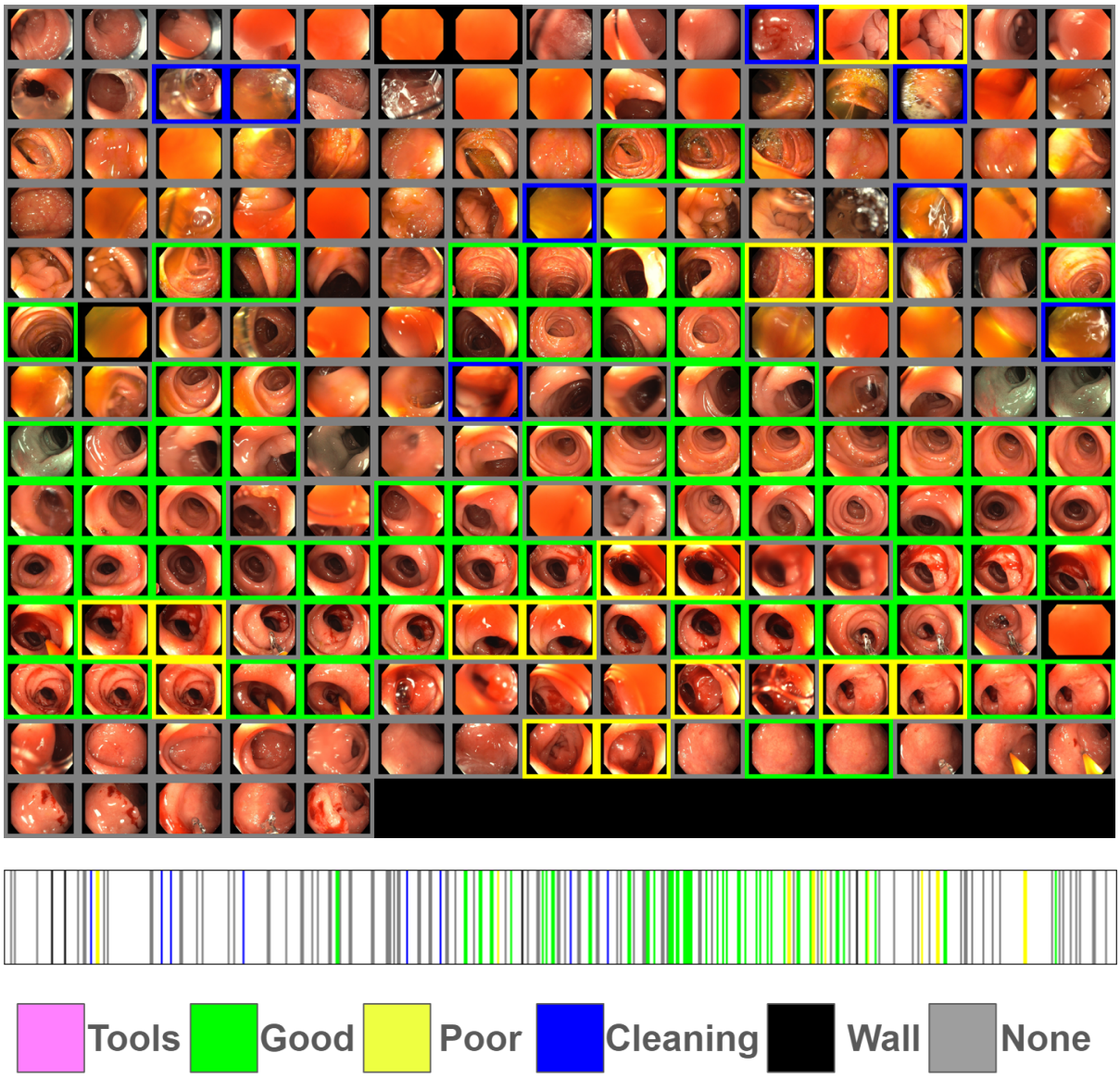


Figure A.8: Qualitative representation of the summary obtained with **SummSeg** on video *Seq_094_hd*.