# Hysteroscopy Video Summarization and Browsing by Estimating the Physician's Attention on Video Segments

Wilson Gavião Neto[a,1], Jacob Scharcanski[a], Jan-Michael Frahm[b], Marc Pollefeys[c]

[a] *Instituto de Informática, Universidade Federal do Rio Grande do Sul.*
*Avenida Bento Gonçalves, 9500. Porto Alegre, RS, Brazil 91501-970*
[b] *Department of Computer Science, University of North Carolina at Chapel Hill.*
*Chapel Hill, NC 27599, USA*
[c] *Department of Computer Science, ETH Zurich.*
*CAB F66, Universitatstrasse 6, 8092 Zurich, Switzerland.*

## Abstract

Specialists often need to browse through libraries containing many diagnostic hysteroscopy videos searching for similar cases, or even to review the video of one particular case. Video searching and browsing can be used in many situations, like in case-based diagnosis when videos of previously diagnosed cases are compared, in case referrals, in reviewing the patient records, as well as for supporting medical research (e.g. in human reproduction). However, in terms of visual content, diagnostic hysteroscopy videos contain lots of information, but only a reduced number of frames are actually useful for diagnosis/prognosis purposes. In order to facilitate the browsing task,

*Email addresses:* `wgaviao@gmail.com` (Wilson Gavião Neto), `jacobs@inf.ufrgs.br` (Jacob Scharcanski), `jmf@cs.unc.edu` (Jan-Michael Frahm), `marc.pollefeys@inf.ethz.ch` (Marc Pollefeys)

[1] *Current Affiliation:* Universidade do Vale do Rio do Sinos (UNISINOS) and Centro Universitário Ritter dos Reis (UNIRITTER)

*Preprint submitted to Medical Image Analysis* *June 6, 2011*

we propose in this paper a technique for estimating the clinical relevance of video segments in diagnostic hysteroscopies. Basically, the proposed technique associates clinical relevance with the attention attracted by a diagnostic hysteroscopy video segment during the video acquisition (i.e. during the diagnostic hysteroscopy conducted by an specialist). We show that the resulting video summary allows specialists to browse the video contents nonlinearly, while avoiding spending time on spurious visual information. In this work, we review state-of-art methods for summarizing general videos and how they apply to diagnostic hysteroscopy videos (considering their specific characteristics), and conclude that our proposed method contributes to the field with a summarization and representation method specific for video hysteroscopies. The experimental results indicate that our method tends to produce compact video summaries without discarding clinically relevant information.

*Keywords:* Video Summarization, Video Indexing, Video Browsing, Hysteroscopy, Medical Video

## 1. Introduction

Diagnostic hysteroscopy is a popular method for assessing and visualizing regions of the female reproductive system like cervical channel, uterine cavity, tubal ostea and endometrial characteristics. A diagnostic hysteroscopy examination is performed by gynecologists with a small lighted telescopic instrument (hysteroscope). During the examination, the hysteroscope transmits an image sequence to a screen, while the gynecologist guides the instrument to assess and diagnose uterine disorders.

In practice, several diagnostic hysteroscopies are performed daily. Each

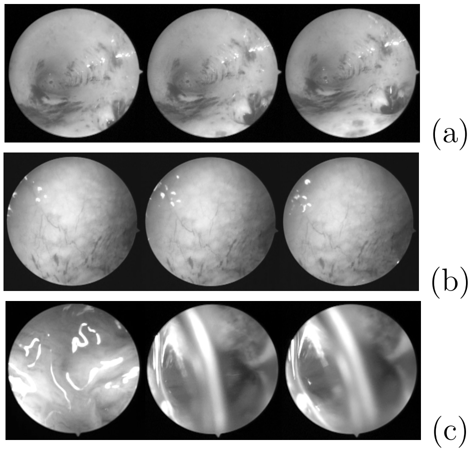


Figure 1: (a-b) Typical frames of a relevant hysteroscopy video segment, which is characterized by unobstructed views of uterus details. (a) Tubular orifice examination. (b) Uterine fundus analysis. (c) Some frames of an irrelevant video segment. These are characterized by regions with mucus (middle and right) and undesired lighting effects (left).

diagnostic hysteroscopy lasts 3 minutes in average, and generates a continuous (uninterrupted) video sequence. Hospitals and clinics often record the full video sequence for establishing video-based diagnosis comparisons, for reviewing or referring cases, as well as for supporting studies in medical research fields (like in human reproduction (Masamoto et al. (2000); Li et al. (2010); Cunha-Filho et al. (2004, 2008); Gavião et al. (2007)) or in early detection of cancer (Clark et al. (2002))). However, in practice only portions of the recorded video sequences actually are relevant from the diagnosis/prognosis point of view. Consequently, whenever a specialist needs to review a video recorded case, he/she have to browse linearly through the video sequence to find the desired contents. A diagnostic hysteroscopy video often contains thousands of frames, and the task of browsing can be time-consuming.

In this paper, we propose a video representation approach that allows fast nonlinear browsing of hysteroscopy video contents. A video summary is generated by our method, which helps guiding the specialists to/through the relevant video sequences. Using such an approach, hysteroscopy video libraries can be built, and the video summaries may be used for indexing and/or fast random access to contents of specific hysteroscopy video sequences.

In order to evaluate our approach, specialists were requested to indicate video segments where the visual content could be represented by a single frame. For a gynecologist, the interesting frames occur in video segments that provide unobstructed views of the female reproductive system, as shown in Figs. 1(a-b). The video segments with frames corrupted by lighting effects (e.g. highlights), or affected by biological features like mucus secretion (as exemplified in Fig. 1(c)), can not be used for diagnosis/prognosis, and do not need to be further retrieved. Section 1.2 provides more details about the characteristics of diagnostic hysteroscopy videos.

When the specialist is performing a diagnostic video hysteroscopy, he/she guides the hysteroscope seeking relevant clinical findings. Little time is spent observing clinically irrelevant areas, but most examination time is spent examining areas that may be relevant for the diagnosis/prognosis. When the relevant areas are found, the specialist focuses the micro camera on the region of interest, or moves it slowly to also examine its surroundings. Therefore, clinically relevant video segments tend to be redundant, since they contain similar frames due to the low camera activity (Gavião et al. (2007); Scharcanski and Gavião (2006); Scharcanski et al. (2005)).

We build our approach based on an analysis of camera motion, as outlined in section 2. In practice, estimating camera motion from 2-D image motion needs recognizing the scene configuration as a first step, and then choosing the appropriate motion model to estimate the camera motion (Irani and Anandan (1998); Torr et al. (1999)). Hysteroscopy videos are captured with a hand-held camera, where the camera motion, and consequently the scene configuration, constantly changes. Since 2-D and 3-D camera motion models suffer from distinct kinds of degeneracies (Torr et al. (1999)), different camera models could be selected for different scene configurations encountered along the image sequence. Therefore, we approach the task of estimating camera motion in hysteroscopy videos as a model selection problem, and propose to select the appropriate motion model adaptively along the image sequence before computing the motion analysis.

This paper is organized as follows. In the next subsection our approach is justified within the context of motion-based video indexing literature. Hysteroscopy videos are described in detail in subsection 1.2. Section 2 presents an outline of our method as well as our contributions. Section 3 discusses the adopted methodology to compute consistent image tracking points from real data. This methodology should be robust to deal with (i) noisy data, (ii) the presence of false point matches, as well as (iii) degenerate scene configurations. Section 4 presents the proposed hysteroscopy video representation. In Section 5 we report experiments and evaluate the effectiveness of the proposed method. We discuss our experimental results in Section 6, and Section 7 presents our conclusions. Appendix A presents a review of concepts in the context of parametric camera motion estimation.

### *1.1. Literature review and the summarization of hysteroscopy videos*

Motion analysis is largely used in video processing tasks, however it is not trivial to represent low-level motion features conveniently in terms of visual content (Lew (2001); Chang (2002); Ngo et al. (2001); Del Bimbo (1999)). Therefore, a large number of approaches have been proposed within this research context (Duan et al. (2006); Ngo et al. (2003); Zhu et al. (2005); Liu et al. (2003); Vasconcelos and Lippman (2000); Piriou et al. (2006); You et al. (2007); Ho et al. (2006); Ma et al. (2005)). As discussed above, camera motion is related to frame redundancy, and it is an useful feature in the context of content-based hysteroscopy video summarization. Our approach is based on camera motion quantification, therefore our review is oriented towards this context.

Since camera motion is an indicator of hysteroscopy operator intention, and it is often associated with visual changes, methods that characterize camera motion qualitatively have been proposed in the nonparametric video indexing literature. Duan et al. (2006) computed a motion feature space and used mean-shift algorithm to recognize camera motion patterns, like panning, tilting and zooming. Ngo et al. (2003) proposed classifying camera motions by analyzing temporal patterns formed from spatial image slices. Such patterns will delimit sub-shots, which are further grouped according to color similarities and temporal closeness. Zhu et al. (2005) also proposed classifying camera motion qualitatively. Histograms are computed from motion vectors and typical camera movements, like panning and zooming, are associated with distinct histogram shapes. However, the combination of different camera movements appears frequently in videos recorded in a hand-held camera fashion, like hysteroscopy videos. Consequently, it could be difficult to characterize camera motion patterns in terms of panning, rotating, tilting and zooming, as proposed by  Duan et al. (2006); Ngo et al. (2003) and Zhu et al. (2005).

In the context of video content representation, many approaches start by distinguishing camera motion (sometimes assumed as dominant motion) from independent object motion (assumed as residual motion). To achieve this, 2-D affine motion model has been widely used to explain the image motion induced by the 3-D camera movement (Bouthemy et al. (1999); You et al. (2007); Ho et al. (2006); Ma et al. (2005); Piriou et al. (2006); Tan et al. (2000); Peyrard and Bouthemy (2005)). In general, authors argue that an affine model can cope with most scene contexts and, even when it can not, it gives satisfactory results for motion representation, keeping the complexity at reasonable levels as well. However, when an affine camera model is adopted, some assumptions are made about the scene context (Longuet-Higgins and Prazdny (1980); Ma et al. (2003)). Considering the nature of hysteroscopy videos, we can not assume some constrained scene contexts as realistic, as discussed in section 1.2.3.

To overcome the limitations of affine motion models, a potential solution would be to incorporate 3-D camera models and deal with generic scene contexts (e.g. view constraints (Hartley and Zisserman (2000)). However, this idea has not attracted much attention in the CBVI (Content-Based Video Indexing) research community (Rothganger et al. (2007); Waizenegger et al. (2008)). The proposition of such methods can be justified by the following difficulties in estimating 3-D camera models: (i) dealing with noisy feature correspondences is challenging, specially in the case of small motions. State-of-art solutions are iterative and depend on nonlinear optimization of geometric constraints, making this process susceptible to local minima and, consequently, unreliable (Sim and Hartley (2006)); (ii) 3-D camera models usually are complex and require to make assumptions about the scene configuration to estimate reliably the camera motion (see Appendix A.4)(Torr et al. (1999); Hartley and Zisserman (2000); Ma et al. (2003)).

The 2-D and 3-D camera models complement each other: (a) 2-D models are not suitable for some 3-D scene configurations, that could be properly treated by 3-D camera models; and (b) some scene configurations can be modeled in 2-D, but 3-D camera models degenerate and can not deliver reliable solutions. As explained in section 1.2.3, both scene contexts appear in hysteroscopy videos. Therefore, we approach the task of estimating camera motion in hysteroscopy videos as a model selection problem and propose to use a state-of-art method (Frahm and Pollefeys (2006)) to select the appropriate motion model for different segments of a hysteroscopy video sequence. However, it is important to note that we do not wish to estimate 3-D camera motion precisely, as in 3-D scene reconstruction tasks (Wu et al. (2007); Hartley and Zisserman (2000)). We just take advantage of rigid 3-D scene constraints to quantify changes in the field of view associated with camera movements.

### *1.2. Specific characteristics of hysteroscopy videos and video summarization*

A video scene usually contains three fundamental information components (Irani et al. (1997)): spatial/appearance information, temporal information, and geometric (3D scene structure and camera motion) information.

Therefore, we characterize hysteroscopy videos according to these components:

### *1.2.1. Spatial information*

Hysteroscopy image intensities are determined by the orientation of the hysteroscope tip, which contains a point light source. Therefore, specular highlights are likely present, as illustrated in the frames of the Fig. 1.

Moreover, due to wide angle optics, most hysteroscopes produce images with noticeable distortion. Therefore, if a distortion-free pinhole camera model is assumed, errors will be introduced in the motion estimate.

From the clinical point of view, hysteroscopy images are spatially classified as relevant or not relevant. Relevant images are characterized by unobstructed views of the uterine wall and often show vascular details, as shown in Fig. 1(b). On the other hand, irrelevant images do not show uterine details clearly. Usually, such irrelevant images are generated in two contexts. Firstly, when the specialist moves the camera searching for uterine regions of interest, and tends to move it faster than when observing a region of interest. Therefore, the quality of the images is downgraded by the effects of fast camera motion. Secondly, mucus secretion along with the flow of insufflated gas produce bubbles, which can appear suddenly in the field of view, obstructing relevant information. Fig. 1(c) exemplifies these irrelevant images.

### *1.2.2. Temporal information*

Unlike comercial videos, diagnostic hysteroscopy is recorded as an uninterrupted image sequence in which the camera is manually guided through the uterine cavity. Usually, specialists tend to move the camera slowly when they are examining important uterine regions. Therefore, a reasonable question could be "why a specilist do not simply press a capture button when relevant images are being observed?". The answer is not so obvious. In clinical practice, the diagnostic procedure is based on the examination of different regions of the uterine wall. Therefore, the micro camera is relocated frequently, and often artifacts downgrade video frames (or parts of them), as illustrated in Fig. 1(c). Such undesired artifacts appear and disappear suddenly during the examination, as explained in the previous section 1.2.1. Thus, even by recording selectively with a record button, such artifacts would still affect the video quality. Therefore, it is more practical to record videos in full instead of using a record button.

### *1.2.3. Geometric information*

In general, there are four distinct phases in a video hysteroscopy: (i) initial *uterine cavity* examination, (ii) *left* and (iii) *right tubal orifice* examination and (iv) *uterine fundus* analysis. Specific examination goals are achieved in each phase (Hamou (1991)). However, in terms of camera motion analysis, we identified distinct scene configurations associated with these phases. Basically, the hysteroscopy video is obtained under three geometrical scene contexts:

a) A *panoramic examination*, involving camera translation and rotation in a 3-D environment (the uterine cavity) with clear depth variation, as shown in the images of Fig. 1(a). These images were taken during the *right tubal orifice* examination phase;

b) An approximate *2-D planar scene* context, which takes place when

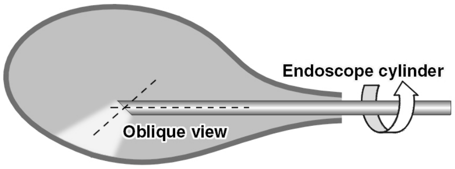


Figure 2: Oblique-viewing endoscope allows the specialist to change viewing direction by rotating the endoscope cylinder around its axis. Ideally, this is one of the degenerate scene configurations, which must be recognized to avoid meaningless results in terms of 3-D motion constraints.

the specialist approaches the uterine wall to examine its endometrial characteristics. This is typical of the *uterine fundus* phase, from which we took some frames shown in Fig. 1(b);

c) Oblique-viewing endoscopes are widely used in hysteroscopies because the viewing direction can be changed by simply rotating the endoscope cylinder around its axis, as illustrated in Fig 2. In practice, this resource is often used by the specialists and, as a result, scene configurations can appear with the camera undergoing only rotations. Mathematically, this movement can be modelled by two successive rotations, as proposed by Yamaguchi et al. (2004).

The presence of distinct scene configurations is an important issue that motivates our approach, since hysteroscopy videos are acquired from a hand-held camera. Some scene configurations need to be recognized to overcome limitations of the models used to estimate camera motion, as discussed in sections 1.1 and Appendix A.4.

## 2. Overview of our approach, contributions and paper outline

In this paper, we present an approach to summarize a hysteroscopy video and allow specialists to browse nonlinearly the content of the hysteroscopy video. As discussed in section 1, relevant frame sequences are associated with slow camera motion and, thus, we propose recognizing/summarizing relevant video segments by analysing the camera-induced motion. To achieve this, we assume the uterine cavity as a rigid environment and adopt a structure-from-motion approach (Ma et al. (2003)) to track and classify image point matches consistent with a parametric camera motion model. Non-rigid and independent motions are assumed to be associated with frames corrupted by undesired biological features (i.e., frames presenting an obstructed view of the uterine findings). Thus, a camera motion model is fitted to a set of image point matches which are then referred as inliers and, since they have been computed, we simply measure changes in the field of view by computing the loss of persistent inliers through the image sequence.

The main contributions of our work are the following:

- Most video summarization methods proposed in the literature focus on dynamic video content representations, which are constructed by exploiting the structure of comercial/edited videos, i. e. by recognizing transitions between semantic units like scene and shots (Ngo et al. (2005); Gao et al. (2009); Dimitrova et al. (2002); Bovik (2009)). However, diagnostic hysteroscopies are recorded as uninterrupted image sequences, where the camera is manually guided through the uterine cavity. Thus, our work contributes to better understanding of what could be considered a hysteroscopy video shot;

- Several summarization methods in the literature recognize semantic units in videos and use predetermined rules to select key-frames, and these key-frames eventually form the final video summary. Our method can estimate the content overlap between neighboring frames and, for this reason, can produce a video summary which is constructed adaptively with a reasonable balance between redundancy and information loss;

- In the context of endoscopic video content retrieval, an interesting approach is proposed by Oh et al. (2007). The authors propose to classify video frames into two classes, informative and non-informative frames. This approach, along with Scharcanski and Gavião (2006), may represent the state-of-art in endoscopic video retrieval. Oh et al. (2007) proposed to identify and discard non-informative frames in order to reduce the number of frames to be analyzed by a physician. We show that our method helps reducing the video contents browsing effort, since redundant information is summarized, and non-informative data is discarded. The approach proposed in this paper is computationally more intensive than the method proposed by Scharcanski and Gavião (2006), but our experiments indicate that our method potentially can outperform (significantly) the method in Scharcanski and Gavião (2006).

Additionally, our approach does not require any special device, like 3-D sensors, to quantify camera motion. These devices have been employed to compute the camera position precisely when the purpose is to recover 3-D scene layouts from endoscopic images (Wu et al. (2007)).

## 3. Robust camera model selection

Given a set of potential point correspondences between two views of a scene, the computation of a camera motion relation $T$ between these views is a widely studied problem in computer vision, as reviewed in Appendix A. In the estimation of $T$ it is not possible to provide a set of perfect correspondences due to noise in the image data and mismatches caused by ambiguities in the feature descriptions. In addition, some scene configurations can lead to incorrect estimates of $T$, as discussed in Appendix A.4. Therefore, a robust method is necessary, since it should be capable of detecting degenerate scene configurations from noisy data, and additionally deal with mismatches. In this section, we discuss an approach to estimate $T$ under the above mentioned limitations.

### *3.1. Dealing with noise and mismatches*

The most common technique to deal with mismatches in the set of correspondences is the *random sample consensus algorithm* (RANSAC) (Fischler and Bolles (1981)). It solves the two problems of computing a relation $T$, which best fits the data, and classifying the data as inliers (correct correspondences) and outliers (Hartley and Zisserman (2000)). The RANSAC method can be outlined in a few steps:

1. Select $m$ random correspondences from the set of all potential correspondences $\{p\}$, and compute a candidate relation $T_c$ from this random sample.

2. Apply $T_c$ to all potential correspondences $\{p\}$ and classify them by thresholding as inliers $\{in_c\}$ or outliers $\{out_c\}$.
3. The best candidate relation $T_c$ is that one which generates the largest consensus set of inliers.
4. The random sampling is repeated until a sufficient number of samples has been evaluated, or the desired probability $\rho$ that a good candidate relation has already been computed.

In the standard approach, RANSAC stops when the number of samples (trials) $S$ is at least

$$S = \log(1 - \rho)/\log(1 - \epsilon^m), \tag{1}$$

where, $\rho$ is the desired probability that at least one of the samples is free from coarse errors, and for each candidate relation $T_c$ the proportion of inliers $\epsilon$ in the data is computed as $|\{in_c\}|/|\{p\}|$. The number of trials $S$ increases with the outlier rate, but it is easy to see that it remains surprisingly small for reasonable values of $\rho$, $\epsilon$, and $m$. The minimal number $m = \lceil \frac{n}{r} \rceil$ of elements required to compute the relation $T$ depends on the number of constraints $r$ provided by each element and the number $n$ of parameters of the relation. For example, given at least $m = 8$ point correspondences it is possible to solve linearly for the essential matrix $\mathbf{E}$ up to a scale, since each point correspondence generate one linear equation (i.e. $r = 1$) in the entries of $\mathbf{E}$ ($n = 8$ parameters), as discussed in Appendix A.3.

In the context of essential matrix estimation, given an estimate of $\mathbf{E}^s$ (Equation A.4) from a RANSAC sample, the set of inliers can be determined in terms of the symmetric epipolar distance error (Hartley and Zisserman

(2000)), which measures how closely a pair of points $\mathbf{x} \rightarrow \mathbf{x}'$ satisfies the epipolar constraint (Equation A.1). The symmetric epipolar distance is based on Equation A.2 and considers the distance $d$ (in pixels) of a point $\mathbf{x}'$ to its projected epipolar line $l' = \mathbf{E}\mathbf{x}$ (see Fig.A.16). Thus, given an estimate of $\mathbf{E}^s$, a pair of corresponding points $\mathbf{x}_i \rightarrow \mathbf{x}'_i$ is classified as inlier if

$$d(\mathbf{x}'_i, \mathbf{E}\mathbf{x}_i) + d(\mathbf{x}_i, \mathbf{E}^{\mathsf{T}}\mathbf{x}'_i) < \tau, \tag{2}$$

where $\tau$ is an error threshold (in pixels).

### *3.2. Dealing with degenerate scene configurations*

Besides the problems involving noise and the presence of mismatches, if only degenerate correspondences are given it is not possible to compute the correct relation $T$. As discussed in Appendix A.4, degeneracy means that the data do not provide enough constraints to compute $T$ uniquely. In practice, it is hard to detect degenerate scene configurations from real data, since the remaining constraints can be determined by the noise or mismatches.

Some approaches have been proposed in the literature for estimating the essential matrix $\mathbf{E}$ (Equation A.1) (Torr (1997); Chum et al. (2005); Frahm and Pollefeys (2006)). Essentially, given a set of image point correspondences, these approaches choose between a homography $\mathbf{H}$ (degenerate data) and an essential matrix $\mathbf{E}$ (non-degenerate data) as the camera motion model that better explains the correspondences.

In this work, we follow the QDEGSAC method proposed by Frahm and Pollefeys (2006), since it does not require any specific knowledge about the degeneracies of the data. QDEGSAC employs RANSAC to compute the correct solution and exploits the number of constraints provided by the data. The algorithm was originally motivated by the limitations of RANSAC in estimating the correct relation for quasi-degenerate data, which means most data do not provide sufficient constraints to compute the relation uniquely (degenerate data), and only a small fraction of the data provides the remaining constraints. For quasi-degenerate data the relation can always be uniquely defined, but the RANSAC algorithm has a low probability of computing the correct relation in this case, as discussed by Frahm and Pollefeys (2006).

QDEGSAC can be applied to various estimation problems in computer vision, however we focus on the linear estimation of the essential matrix, hence the data matrix $\mathbf{A}$ is given in terms of the eight-point algorithm (see Appendix A.3). As mentioned in Appendix A.4, $\mathbf{A}$ should have a rank $r_{\mathbf{A}} = 8$ to obtain a non trivial solution for Equation A.4. In the absence of noise, $r_{\mathbf{A}} = 8$ means the data provides 8 linearly independent constraints and, consequently, the entries of $\mathbf{E}^s$ can be uniquely computed (up to scale) as the 1-dimensional null-space of $\mathbf{A}$ (Hartley and Zisserman (2000)). In this case, the data is *non-degenerate*.

On the other hand, if $r_{\mathbf{A}} < 8$, a smaller number of independent constraints is provided by the data and the solution $\mathbf{E}^s$ becomes ambiguous, which reflects a *degenerate* scene configuration. *Motion degeneracy* and *structure degeneracy* are common types of degeneracies (see Appendix A.4), and these are characterized by $r_{\mathbf{A}} = 6$. In this case, the relation $\mathbf{E}$ degenerates in a homography $\mathbf{H}$ (Torr et al. (1999)).

From above, the rank of the data matrix $\mathbf{A}$ can be used to detect degenerate data if data is noise free. In practice, the computation of the rank is inaccurate since it is sensitive to noise in the data. The noise disturbance causes small singular values to occur, hence it is still possible to estimate the rank by using an appropriate threshold on the singular values. However, if a sample contains degenerate correspondences and some mismatches, the rank $r_{\mathbf{A}}$ of the data matrix increases, and $r_{\mathbf{A}}$ appears to be similar to the expected rank of a non-degenerate case. For this reason, the ambiguity can not be detected by analyzing the singular values of the data matrix.

Nevertheless, the QDEGSAC algorithm can be interpreted as a robust measurement of the rank $r_{\mathbf{A}}$ of the data matrix $\mathbf{A}$. Basically, the algorithm consists of three phases (see Frahm and Pollefeys (2006) for details):

1. Initially, a RANSAC process estimates the relation assuming that the data are not degenerate (i. e. assuming $r_{\mathbf{A}} = 8$). From this process, the data is classified into inliers $\{in_8\}$ and outliers $\{out_8\}$, according to the estimated relation $T_{RANSAC,8}$.
2. Afterwards, the rank of the data matrix is estimated robustly from the inliers $\{in_8\}$. This step is denoted as *model selection* and it determines a lower rank for degenerate data, even if mismatches generate free constraints. Basically, a series of RANSACs is performed over $\{in_8\}$. Inliers in the set $\{in_8\}$ are tested to be consistent with relations $T_{RANSAC,dim}$, which employ a smaller number of constraints ($dim < 8$).
3. Finally, the *model completion* inspects the outliers of the previous phases, attempting to find non-degenerate inliers that provide the remaining constraints to compute the 8 degrees of freedom of the relation

$\mathbf{E}^s$.

The algorithm produces two sets of inliers, and it can be qualitatively evaluated in terms of them:

- **Degenerate inliers** are those inliers that are in a degenerate configuration (e.g. coplanar points). These degenerate inliers are computed in the dimension 6, which means they are consistent with a relation $T_{RANSAC,6}$ ($r_{\mathbf{A}} = 6$). Ideally, a degenerate scene configuration is detected if all inliers computed in the dimension 8 are also obtained as inliers in dimension 6.

- **Non-degenerate inliers** are those inliers that are not in a degenerate configuration (e.g. are off-plane points). These inliers are consistent with a relation $T_{RANSAC,8}$, but not consistent with a relation $T_{RANSAC,6}$.

### *3.3. Computational cost*

When the data does not give support to a relation computed by employing only $8 - i$ constraints ($i > 0$), the QDEGSAC algorithm is more computationally expensive. It follows from Equation 1 that a RANSAC process needs a significant number of trials to prove that the data does not support a relation $T_{RANSAC,8-i}$. In order to reduce the unnecessary computational effort in degenerate scenes, the process of testing the inliers $\{in_8\}$ in lower dimensions (*model selection*) is stopped when the dimension 5 is achieved, since a motion relation estimated at dimension 6 (like a homography) is appropriate to explain data of practical degenerate scenes (Nistér (2000); Torr et al. (1999)). Note, however that, in the case of non-degenerate data ($dim = 8$), this approach does not avoid a computationally expensive RANSAC process, which tests the data in lower dimensions for degeneracies.

## 4. The proposed hysteroscopy video content representation

In this section, we present our hierarchical representation for hysteroscopy videos that allows the specialists to fast browse the video content. We exploit consistent image point tracks through the image sequence to quantify visual changes, and arrange frames hierarchically according to the number of corresponding points they share. The idea is that if a set of image points is observed through a set of frames, these frames hold an amount of content overlap, and can have some level of content similarity.

Our approach is presented as follows: In section 4.1, we present the notation for the initial set of point correspondences established through the frame sequence. In section 4.2, we validate geometrically the set of image point correspondences computed in the previous step. Section 4.3 presents our approach to represent hysteroscopy videos in terms of their content, and shows how this representation can be used for fast video browsing. Finally, section 4.7 presents the criterion to select key-frames, which will constitute the video summary, and will be used in the video browsing task.

### *4.1. Computing the set of potential point correspondences between frames*

Our approach starts by detecting and tracking points at every pair of consecutive frames $I^j$ and $I^{j+1}$ of the video sequence. KLT tracker is used for this purpose, since it has shown satisfactory results for tracking image points in endoscopic scenes (Rai et al. (2006); Wu et al. (2007)). Thus, KLT algorithm delivers the set of potential point correspondences $\{p^j\}_{j=1}^{N-1} = \{\mathbf{x}_i^j \rightarrow \mathbf{x}_i^{j+1}\}$ between consecutive frames $I^j$ and $I^{j+1}$, where $N$ is the number of frames in the video and $M$ is the number of points correspondences to be tracked from a given frame, $i = 1 \cdots M$. The value of $M$ is user defined to provide a high number of point correspondences that will be refined later, and we set $M = 650$ in our experiments. In order to preserve a number of $M$ point tracks in each frame, if $k$ points have been lost in frame $I^j$, they are replaced and $k$ new point tracks start at $I^j$, as implemented by Birchfield (2009).

### *4.2. Constraining point correspondences to be consistent with the camera motion*

In this section, we discuss our approach to integrate image point tracking and geometrical validation of the point tracks, since the image points $\{\mathbf{x}_i^j\}$ are tracked independently and a (simplified) local translational motion model is assumed, which is not general enough to explain the camera motion between frames.

We are interested in quantifying camera movements by analyzing the 2-D image motion induced by the camera itself and, for this reason, we start selecting point correspondences from $\{p^j\}$ that are consistent with a camera motion model. To achieve this, we essentially enforce the epipolar constraint (Equation A.1) over pairs of corresponding points $\{\mathbf{x}_i^j \rightarrow \mathbf{x}_i^{j+\Delta}\}$ in $\{p^j\}$. Due to noise, consistent point correspondences (inliers) are selected in terms of the symmetric epipolar distance error (Equation 2).

Hysteroscopy videos are acquired from a hand-held camera and distinct scene configurations become noticeable in some phases of a typical hysteroscopic examination, as discussed in section 1.2.3. However, some scene configurations are critical in the process of estimating the camera motion relation between a pair of frames, making a preliminary step to detect degenerated scene configurations necessary, as discussed in Appendix A.4. For example, in the context of typical hysteroscopy examination phases, the *uterine fundus* analysis characterizes a *structure degeneracy*, where the three dimensional layout of the scene is nearly planar (see Appendix A.4 for details). On the other hand, the *panoramic examination* phase characterizes a generic scene context, where an essential matrix $\mathbf{E}$ could be estimated and employed to compute the set of inliers.

Besides a certain degree of accuracy, we also wish to estimate the correct camera model to compute as many consistent point tracks (inliers) as possible, since further decisions in our approach will be made in terms of inliers consensus. As discussed in section 1.1, many approaches in the CBVI literature propose to use a simplified affine motion model (which is a particular case of a homography model $\mathbf{H}$) to explain camera motion. However, in some hysteroscopy scene configurations, this would restrict the number of inliers, keeping only those tracked points that are consistent with a portion of the scene.

To deal with the difficulties discussed above, we adopt the QDEGSAC algorithm as described in section 3. It detects degenerate scene configurations automatically, and chooses a potential camera model between pairs of frames. QDEGSAC is capable of dealing with mismatches in $\{p^j\}$, as well as finding inliers from noisy point correspondences. As output, QDEGSAC delivers the sets of inliers $\{in\}$ for every pair of frames considered in the video sequence. Only tracked points classified as inliers will be used in the next phase of our approach, which is detailed in the next section.

### *4.3. Hierarchical representation for hysteroscopy videos*

In this section, we propose a hierarchical representation for hysteroscopy videos so that specialists can browse the video contents non-linearly. We basically exploit the behavior of consistent point tracks through the video in order to group neighboring frames iteratively. In the previous step, consistent point tracks are computed and identified as inliers $\{in\}$ in each pair of frames (at regularly spaced intervals $\Delta$).

Using KLT algorithm, we compute image point correspondences $\{p^j\}_{j=1}^{N-1}$ for every pair of consecutive frames $I^j$ and $I^{j+1}$ in the sequence, since better matching results are achieved when the motion between frames is fairly small. However, in order to reduce the computational effort, we do not compute inliers between consecutive frames at the original frame rate. Videos are instead sampled at a $\Delta > 1$ frame rate, and little information is lost because relevant hysteroscopy video segments are usually acquired under slow or no camera motion, what produces an excessive number of frames (redundant views of the same uterine region).

On the other hand, image point tracks are eventually lost, or they move out of view as the video sequence evolves. Therefore, the number of point correspondences between widely separated views is not always sufficient to allow a reliable estimation of relations like the essential matrix $\mathbf{E}$ or the homography $\mathbf{H}$. For this reason, we set a conservative (small) value to $\Delta$

in our experiments, since we are interested in constructing a video representation that starts quantifying small view changes (between temporally close frames), but avoids computing relations between potentially redundant consecutive frames.

We assume the frames $I^j$ and $I^{j+\Delta}$ as irrelevant, and do not estimate the relation between them, if the number of point correspondences $M$ between them drops below a conservative threshold $\varphi$ of 50 point matches. Experimentally, we have verified that irrelevant frame sequences are associated with low rates of successfully matched points, and a rate of $\varphi < 100$ successfully matched points have implied in a high rate of failure when finding $M = 650$ point matches. Therefore, we set $M = 50$ in our experiments. In the following, we discuss some key concepts which are important for the comprehension of the proposed video representation.

### *4.4. Consistent points*

Let $\{\mathbf{x}^j_{con}\}$ be the set of *consistent points* associate with the frame $I^j$. A consistent point is defined as follows. Let $\{in\}^j_{j-\Delta}$ be the set of inliers computed by the QDEGSAC algorithm between the frame $I^j$ and its left neighbor $I^{j-\Delta}$. Let $\{in\}^{j+\Delta}_j$ be the set of inliers computed between the frame $I^j$ and its right neighbor $I^{j+\Delta}$. An image point $\mathbf{x}^j_i$ is defined as a *consistent point* associated with the frame $I^j$ if the correspondences $\mathbf{x}^{j-\Delta}_i \leftrightarrow \mathbf{x}^j_i$ and $\mathbf{x}^j_i \leftrightarrow \mathbf{x}^{j+\Delta}_i$ are validated as inliers by the QDEGSAC algorithm. In other words, $\mathbf{x}^j_i \in \{\mathbf{x}^j_{con}\}$ if

$$\begin{aligned} \mathbf{x}_i^{j-\Delta} \leftrightarrow \mathbf{x}_i^j &\in \{in\}_{j-\Delta}^{j} \text{ and} \\ \mathbf{x}_i^j \leftrightarrow \mathbf{x}_i^{j+\Delta} &\in \{in\}_{j}^{j+\Delta}. \end{aligned} \tag{3}$$

*4.5. Persistent points and content overlap*

Each frame $I^j$ is now represented by its set of consistent points $\{\mathbf{x}_{con}^j\}$. Given the frames $I^j$ and $I^{j+k\Delta}$, a point $\mathbf{x}_i^j$ is defined as a *persistent point* from $I^j$ to $I^{j+k\Delta}$ when

$$\mathbf{x}_i^{j+t\Delta} \in \{\mathbf{x}_{con}^{j+t\Delta}\} \quad \text{where} \quad t = 0 \dots k, \tag{4}$$

which means that $\mathbf{x}_i^j$ and their corresponding points from $I^{j+\Delta}$ to $I^{j+k\Delta}$ must be consistent points too.

Once we have defined the concept of persistent points, we introduce the notion of *content overlap* between frames. Given the frames $I^j$ and $I^{j+k\Delta}$, and their sets of consistent points $\{\mathbf{x}_{con}^j\}$ and $\{\mathbf{x}_{con}^{j+k\Delta}\}$, the *content overlap* $\theta_j^{j+k\Delta}$ between $I^j$ and $I^{j+k\Delta}$ is defined as the number of persistent points computed from $I^j$ to $I^{j+k\Delta}$.

Formally, let $per(\mathbf{x}_i^j, k)$ be a boolean function whose purpose is to verify the persistence of a point $\mathbf{x}_i^j$ from $I^j$ to $I^{j+k\Delta}$:

$$per(\mathbf{x}_i^j, k) = \begin{cases} 1, & \text{if } \mathbf{x}_i^{j+t\Delta} \in \{\mathbf{x}_{con}^{j+t\Delta}\} \quad \text{where} \quad t = 0 \dots k, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Thus, given the frames $I^j$ and $I^{j+k\Delta}$, the *content overlap* between them is defined as

$$\theta_j^{j+k\Delta} = \sum_{i=1}^{M} per(\mathbf{x}_i^j, k), \tag{6}$$

where $M$ is the number of point correspondences established by the KLT tracker.

Given three frames $I^{j-\Delta}$, $I^{j}$ and $I^{j+\Delta}$, the content overlap $\theta_{j-\Delta}^{j}$ between frames $I^{j-\Delta}$ and $I^{j}$ is larger than $\theta_{j}^{j+\Delta}$ between $I^{j}$ and $I^{j+\Delta}$ if

$$\theta_{j-\Delta}^{j} \; > \; \theta_{j}^{j+\Delta}, \tag{7}$$

which means that the frames $I^{j-\Delta}$ and $I^{j}$ contain more consistent points in common than the frames $I^{j}$ and $I^{j+\Delta}$.

### *4.6. Video segment tree*

An iterative process is employed to group frames into video segments. A video segment is represented as $\delta^{a \mapsto b}$, where $a$ and $b$ are the first and the last frame of the video segment, respectively. New segments are formed according to the content overlap test in Equation 7, and frames that share a high number of consistent point tracks are grouped first.

The building block of the iterative process consists in analyzing sequences of four frames. Given a sequence of four $\Delta$-spaced frames $I^{j-2\Delta}$, $I^{j-\Delta}$, $I^{j}$ and $I^{j+\Delta}$, and their respective sets of consistent points $\{\mathbf{x}_{con}^{j-2\Delta}\}$, $\{\mathbf{x}_{con}^{j-\Delta}\}$, $\{\mathbf{x}_{con}^{j}\}$ and $\{\mathbf{x}_{con}^{j+\Delta}\}$, the two central frames, $I^{j-\Delta}$ and $I^{j}$, will be grouped into a new video segment $\delta^{j-\Delta \mapsto j}$ if

$$\theta_{j-2\Delta}^{j-\Delta} \; \leq \; \theta_{j-\Delta}^{j} \; \geq \; \theta_{j}^{j+\Delta}. \tag{8}$$

Fig. 3 illustrates this idea in terms of four views and the view overlaps between them. The view overlaps are computed with our content overlap metric (Equation 6).

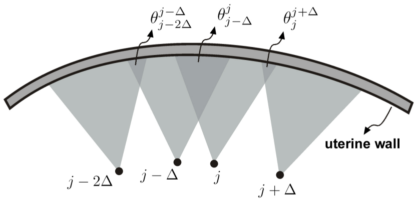


Figure 3: The overlap between views will determine which of the frames will be grouped first in our approach. The central frames $I^{j-\Delta}$ and $I^{j}$ will constitute a video segment $\delta^{j-\Delta \mapsto j}$ if the content overlap between them $\theta^{j}_{j-\Delta}$ is larger than the content overlaps $\theta^{j-\Delta}_{j-2\Delta}$ and $\theta^{j+\Delta}_{j}$, which are computed with relate to their neighboring frames $I^{j-2\Delta}$ and $I^{j+\Delta}$.

As single frames, video segments also are represented by sets of consistent points. However, in the case of video segments, such sets are constituted by consistent points that persist from the first to the last frame of the video segment, as defined in Equation 4.

As the iterative process goes on, new frames will aggregate to a video segment and the number of persistent points in the segment will decrease, since tracked points are eventually lost, or move out of view due to camera motion. Therefore, a video segment $\delta^{a \mapsto b}$ is considered stable, and it will not aggregate frames any more, when the overlap between the video segment and its $\Delta$-spaced neighbor frame $I^{a-\Delta}$, and the overlap between the video segment and its $\Delta$-spaced neighbor frame $I^{b+\Delta}$, drop below a threshold $\zeta$, i. e. when

$$\theta^{a \mapsto b}_{a-\Delta} < \zeta \quad \text{and} \quad \theta^{b+\Delta}_{a \mapsto b} < \zeta, \tag{9}$$

where $\theta^{a \mapsto b}_{a-\Delta}$ and $\theta^{b+\Delta}_{a \mapsto b}$ represent the content overlap between the video segment $\delta^{a \mapsto b}$ and its previous frame $I^{a-\Delta}$ and its next frame $I^{b+\Delta}$, respectively.

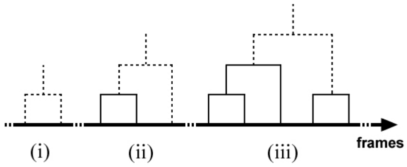


Figure 4: The iterative process groups neighboring frames to form video segments by (i) grouping single frames into new video segments, (ii) aggregating single frames to existing video segments or (iii) grouping neighboring video segments into larger video segments.

Actually, the iterative process develops in three ways: (i) grouping single frames into new video segments, (ii) aggregating single frames to existing video segments, or (iii) grouping neighboring video segments into larger video segments. This is illustrated in Fig. 4. The iterative process will stop when the video segments become stable (they stop to aggregate frames), i. e. when the content overlap between every neighboring video segment decreases below the threshold $\zeta$, as defined in Equation 9.

To each one of the video segments formed we associate a key-frame. Thus, at the end of the iterative process, a final set of video segments is provided and, from each of them, a key-frame is selected, which will finally constitute the video summary. This video summary will guide a specialist in the task of browsing the video content. Once a specialist has selected a key-frame $I_{kf}^{a \mapsto b}$, he/she can browse the related content in the corresponding video segment $\delta^{a \mapsto b}$. To achieve this, we propose to exploit the hierarchical structure produced by the iterative process when constructing each video segment. This structure produces a valuable information for video content browsing purposes, as we will discuss next.

Each final video segment can be represented by a hierarchical binary tree, which preserves the information about the pairs of video segments that were grouped iteratively until the final video segment is constituted. Fig. 4(iii) illustrates a binary tree, where the final step to constitute the video segment is represented by dotted lines, and the grouped video segments are represented in a solid line style. We call the binary tree associated with a video segment the *video segment tree*.

Such representation is helpful for organizing the hysteroscopy video content, since it allows the specialists to generate a video summary with more, or less, details/key-frames without introducing spurious frames into the summary. This is achieved by traversing the video segment tree through its levels: from the upper to the lower levels to increase the number of key-frames, and from the lower to the upper levels to generate a more compact video summary. This idea is illustrated in Fig. 5, which shows a particular video segment tree in which the task of browsing can be represented by an imaginary horizontal line across the tree (gray arrows), where more or less redundant video summaries (represented as a set of key-frames) are generated by sliding this line vertically. In each level of the tree, the set of subsegments (subtrees) are determined at the intersection of this line with the tree structure. A subsegment has an associated key-frame, the set of key-frames constitute the current level of the tree. Thus, once the specialist has selected a key-frame on the top of a video segment tree, he/she can browse the related video content without introducing frames which would be out of the video segment context.

Since a video segment tree delimits a sequence of content-related frames, the delimited video sequence can be analogous to a *shot*, which is a video

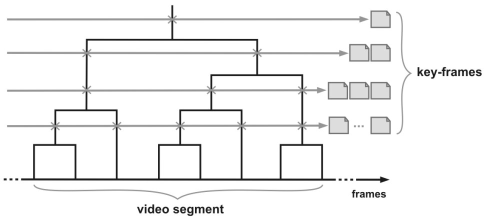


Figure 5: A particular video segment tree in which the task of browsing is represented as an imaginary horizontal line across the tree (gray arrows). As this line slides through the levels of the tree, more or less compact video summaries are produced in terms of the number of key-frames. The intersections × will determine a hierarchy of video subsegments (subtrees), which will constitute the final video segment at the top of the tree.

unit widely exploited in CBVI literature to represent the content of commercial/edited videos. Besides, an important advantage of the tree representation is that it allows a guided browsing of the video segment content as one navigates through the levels of the tree. The frame grouping process starts by grouping frames with a high degree of content overlap, frames with a low degree of content overlap are grouped at upper levels of the tree, hence redundant frames are progressively inserted into the video summary as the specialist traverses the segment tree from the upper to the lower levels. Thus, our guided video browsing is oriented towards less redundant summaries, which is a desirable feature in the context of video browsing applications.

### *4.7. Selection of key-frames*

Given a video segment $\delta^{a \mapsto b}$, or a video subsegment, and its associated set of consistent points $\{\mathbf{x}_{con}^{a \mapsto b}\}$, the selected key-frame $I_{kf}^{a \mapsto b}$ is the frame with the largest content overlap among the frames in the video segment. To quantify this criterion we take into account what we call *consistent point duration*, that is the number of frames in which an image point is tracked as a consistent point. Recall that each of the frames in a video segment $\delta^{a \mapsto b}$ has an associated set of consistent points $\{\mathbf{x}^{a}_{con}\}, \{\mathbf{x}^{a+\Delta}_{con}\}, \ldots, \{\mathbf{x}^{b-\Delta}_{con}\}, \{\mathbf{x}^{b}_{con}\}$ . For each set of consistent points, we compute the *mean duration*, and the frame with the highest mean duration is selected as the key-frame $I^{a \mapsto b}_{kf}$.

Let $con(\mathbf{x}^j_i)$ be a boolean function that verifies if $\mathbf{x}^j_i$ is a consistent point associated with frame $I^j$:

$$con(\mathbf{x}^j_i) = \begin{cases} 1, & \text{if } \mathbf{x}^j_i \in \{\mathbf{x}^j_{con}\}; \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

Thus, within a video segment $\delta^{a \mapsto b}$, we can define the *mean duration* of the points $\{\mathbf{x}^j_i\}^M_{i=1}$ associated with frame $I^j$ as

$$\widehat{dur}(I^j) = \frac{\sum_{i=1}^{M} \sum_{t=a}^{b} con(\mathbf{x}^t_i)}{M}, \tag{11}$$

where $M$ is the number of point tracks computed by the KLT tracker in each frame. Thus, the key-frame $I^{a \mapsto b}_{kf}$ associated with $\delta^{a \mapsto b}$ is the frame $I^j$ where

$$I^{a \mapsto b}_{kf} = \arg \max_{j=a \ldots b} \widehat{dur}(I^j). \tag{12}$$

In case of two or more frames of the same video segment having the same highest *mean duration*, the first frame in the sequence is selected as a key-frame.

## 5. Experimental results

Table 1: Description of the videos used in our experiments: ***hyst1***: traditional hysteroscopy examination with well-defined phases and abnormal findings; ***hyst2***: a short examination with relevant frames degraded by numerous small specular regions; ***hyst3***: many relevant frame sequences suddenly interrupted by undesired effects or fast camera motion; ***hyst4***: challenging image sequences from the persistence of the tracked points, since the frames show low texture content.

| Videos | Number of frames | Number of video segments selected by the specialists |
|---|---|---|
| *hyst1* | 3013 | 9 |
| *hyst2* | 2080 | 5 |
| *hyst3* | 5500 | 5 |
| *hyst4* | 7500 | 12 |

In this section, we present several experiments with synthetic and real sequences. We conducted experiments on four interpreted hysteroscopy videos, which were recorded at 30 frames per second. Table 1 shows a short description of the videos. These videos were selected based on their visual characteristics, which represent potential difficulties for the proposed approach. Fig. 6 shows the median frame of the video segments manually selected by specialists. We evaluate the performance of our approach in terms of the results of consistent image point tracking and video summarization.

### *5.1. Experiments on synthetic data*

We first test our summarization approach on synthetic data. Since the correct camera motion relation is known, the results estimated by QDEGSAC can be evaluated in comparison with the actual set of inliers and outliers (the ground truth). In the context of degenerate scenes, such results can be evaluated qualitatively in terms of the set of degenerate inliers, since the undesired detection of non-degenerate inliers would be caused by the limitations of QDEGSAC in dealing with noise and mismatches. The following notation is employed in this section:

- $T_{Actual}$ represents the correct camera motion relation between two points of view (frames);

- $\{in\}_{Actual}$ represents the *set of actual inliers*, which gives support to $T_{Actual}$;

- $\{out\}_{Actual}$ represents the *set of actual outliers* (mismatches), which does not give support to $T_{Actual}$;

- $\{in\}^{\tau}_{QDEGSAC}$ represents the set of inliers delivered by the QDEGSAC algorithm, where $\tau$ is the error threshold (in pixels) that allows to accept a pair of corresponding points as an inlier. $\{in\}^{\tau}_{QDEGSAC}$ gives support to the estimated camera motion relation $T_{QDEGSAC}$.

We randomly generate a set of 100 3D points with a depth variation of 10-50 units of focal length (u.f.l.). The interframe rotation is $\alpha \in \{0, 5\}$ degrees around a random axis of rotation, and the interframe translation is $|\mathbf{t}| \in \{0, 5\}$ u.f.l. with a random direction of translation. The views are obtained by perspective projection using an image size of $512 \times 512$ pixels. Zero-mean gaussian noise with a standard deviation of $\sigma = [0, 1]$ pixels is added to the point correspondences in two views. All parameters were set with the intention of approaching real data conditions. Figures 7-12 show measures that result from the average of 100 QDEGSAC runs.

Fig. 7 shows the performance of QDEGSAC in computing a relation with high support in the point correspondences for $(|\mathbf{t}|, \alpha) = (5, 5)$, as a function of noise $\sigma$. Given an error threshold of $\tau$ pixels to accept a point correspondence as an inlier, we compute the number of inliers delivered by the QDEGSAC algorithm (see Fig. 7, left). Results are shown for $\tau \in \{1, 2, 3\}$. Additionally, Fig. 8 shows the performance of QDEGSAC under degenerate scene configurations for $(|\mathbf{t}|, \alpha) = (5, 5)$ and coplanar points (planar scene). Fig. 8 also shows the proportion of degenerate inliers, which should be as high as possible, since the scene configuration is degenerate and the undesired presence of non-degenerate inliers can be explained by noise.

Given the correct camera motion relation $T_{Actual}$ between the views, we compute the mean error for the inliers $\{in\}^{\tau}_{QDEGSAC}$ delivered by QDEGSAC, where $\tau \in \{1, 2, 3\}$. We use a homography model $\mathbf{H}$ as the correct motion relation $T_{Actual}$ between degenerate views. In this case, we employ the symmetric transfer error (Hartley and Zisserman (2000)), which measures how closely a pair of points $\mathbf{x}_i \rightarrow \mathbf{x'}_i$ satisfies a relation $\mathbf{H}$. For non-degenerate scene configurations, we employ the symmetric epipolar error, as defined in Equation 2. Fig 9 shows the errors for both degenerate and non-degenerate scene configurations as a function of noise $\sigma$. Planar and pure rotation scene configurations has given almost the same errors. Note that data is corrupted only by noise in this stage, i. e. it does not contain mismatches. Thus, we observed the following:

1. The performance of the algorithm deteriorates with the amount of noise.
2. As expected, the computational effort increases with the amount of

noise. The noise makes the proportion of inliers to decrease, hence the required number of RANSAC trials $S$ increases (see Equation 1).

3. However, the required number of QDEGSAC trials in degenerate scenes is significantly smaller than the number of trials in generic scenes. In non-degenerate scenes, QDEGSAC needs a significant number of trials to verify if the inliers are not supported by a relation computed with less than 8 constraints, as discussed in section 3.3.
4. The best performance has been achieved by employing QDEGSAC with $\tau = 3$. It gives the lowest computational effort, as well as the highest number of inliers under noisy point correspondences (no mismatches). Besides, it gives almost the same symmetric error compared to QDEGSAC with $\tau = 1$ or 2.

Note that we wish to compute as many inliers as possible in the image sequence, hence our major concern is to avoid losing these consistent tracked points due to the presence of noise or mismatches. Mismatches can lead to a high number of inlier losses since an estimated motion relation $T_{mis}$, which is supported by mismatches, can stop the tracking process of actual inliers, since they could be considered as outliers in terms of $T_{mis}$. Therefore, we also evaluate how mismatches affect the performance of QDEGSAC in terms of the actual inlier losses, especially for higher values of $\tau$ like $\tau = 3$ pixels.

Please note that the higher the $\tau$ value is, the higher is the chance of classifying mismatches as inliers, and consequently, the higher is the risk of loosing consistently tracked points. Therefore, we have not considered values of $\tau$ higher than 3 pixels in our experiments.

Fig. 10 shows the number of actual inlier losses for $(|\mathbf{t}|, \alpha) = (5, 5)$ as a function of the number of mismatches. We generate a set of mismatches by selecting randomly a translation direction and a translation magnitude. Additionally, gaussian noise with $\sigma = 0.1$ pixels has added to the point correspondences in the two views. Fig. 11 shows the number of actual inlier losses under degenerate scene configurations for $(|\mathbf{t}|, \alpha) = (0, 5)$ (pure rotation). Fig. 11 also shows the proportion of degenerate inliers as well as the undesired presence of non-degenerate inliers.

We also evaluated the symmetric epipolar error for the inliers $\{in\}^{\tau}_{QDEGSAC}$ delivered by QDEGSAC in terms of the correct camera motion relation $T_{Actual}$. Fig 12 shows the errors for both degenerate and non-degenerate scene configurations. Note that we employ a homography $\mathbf{H}$ model as the correct motion relation $T_{Actual}$ for non-degenerate scenes. Fig 12 also shows the error computed over $\{in\}^{\tau}_{QDEGSAC}$ against the ground truth error, which is computed for the ground truth set of inliers $\{in\}_{Actual}$. In order to give an idea of the set of mismatches employed in these tests, we also compute the error over all point correspondences, which include all mismatches.

Considering the effects caused by mismatches, we observed the following:

1. The computational effort increases with the number of mismatches, since RANSAC requires more trials to reach a consensus in the set of correspondences, as discussed above in terms of noisy points.
2. In terms of actual inlier losses, better results has been achieved by employing QDEGSAC with $\tau = 3$. There is a clear trade off in choosing $\tau$: it should be sufficiently high to avoid actual inlier losses due to the effects of noise and, at the same time, it should be as low as possible

to avoid classifying mismatches as inliers. In our experiments, $\tau = 3$ gave the best results. For example, in degenerate scene configurations, as shown in Fig. 11, $\tau = 1$ results in false non-degenerate inliers, but $\tau = 3$ was less sensitive to this problem.

3. Qualitatively, the performance of QDEGSAC starts deteriorating around a mismatch rate of 10%. The transfer error computed for degenerate scenes is quite higher than the epipolar error computed for non-degenerate scenes. Since mismatches support the estimated motion relation, they are understood by the QDEGSAC algorithm as non-degenerate inliers. The homography, which is the correct motion relation, can not explain non-degenerate inliers, and gives a high transfer error for the mismatches that are incorrectly included in the estimated set of inliers.

### *5.2. Experiments with real sequences*

We evaluate the performance of QDEGSAC on real conditions of noise and mismatches, as well as the overall performance of our approach in selecting relevant information from hysteroscopy sequences.

#### *5.2.1. Performance of the QDEGSAC algorithm on real sequences*

We now test the performance of the QDEGSAC algorithm on frame pairs from several video sequences, e.g. the *Tubal Orifice* sequence, which is illustrated in Fig 13.

Due to the difficulty of determining a ground truth in real hysteroscopy sequences, we test QDEGSAC in terms of the number of successfully tracked points, and the corresponding symmetric epipolar error. Although this is an indirect measurement of the quality of the results, it gives a good indication of the quality of the estimated relation when mismatches exist in the data. We observed the following:

1. The error increases with the temporal distance between frames.
2. Although QDEGSAC delivers a relatively high number of inliers in dimension 6 with $\tau = 3$ for generic scenes, it gives the best performance in terms of the number of inliers and trials. Table 2 summarize the results for the *Tubal Orifice* sequence.
3. QDEGSAC with $\tau = 2$ gives the most coherent results in terms of scene configurations detection; for planar (degenerate) or almost planar sequences, it gives a high number of inliers in dimension 6; for non-planar scenes, like the *Tubal Orifice* sequence, a low number of inliers was detected in dimension 6. This was verified even for temporally close frames.

### *5.2.2. Experimental results on video summarization*

In this section, we evaluate the performance of our approach on video summarization. The experiments were conducted on four interpreted hysteroscopy videos: *hyst1*, *hyst2*, *hyst3* and *hyst4*. The videos were selected from a hysteroscopy library with more than 10.000 exams, which are usually recorded in a DVD format with interlaced frames. In order to minimize difficulties in the point tracking process, the videos were deinterlaced by employing the bob deinterlace method available in the VirtualDub software (www.virtualdub.org). The deinterlacing process splits frames in separate

Table 2: Results for sequence *Tubal Orifice* averaged over 100 QDEGSAC runs. Residual error is given in pixels.

| **Frames** | **1-10** | **10-20** | **1-20** |
|---|---|---|---|
| # KLT matches | 547 | 469 | 426 |
| # inliers for $\tau = 1$ | 359 | 333.10 | 266.31 |
| # inliers for $\tau = 2$ | 480.02 | 397.08 | 355.44 |
| # inliers for $\tau = 3$ | 507.01 | 428.67 | 389.23 |
| Res. error for $\tau = 1$ ($\sigma$) | 0.26 | 0.25 | 0.28 |
| Res. error for $\tau = 2$ ($\sigma$) | 0.42 | 0.44 | 0.45 |
| Res. error for $\tau = 3$ ($\sigma$) | 0.63 | 0.64 | 0.64 |
| # required trials for $\tau = 1$ | 786.77 | 694.27 | 2129.10 |
| # required trials for $\tau = 2$ | 192.63 | 206.42 | 295.80 |
| # required trials for $\tau = 3$ | 68.28 | 98.39 | 231.45 |

fields, hence the videos are processed at a double frame rate in our experiments.

Since $\tau = 3$ results in the lowest computational effort, as well as the highest number of inliers, without a significant impact on the quality of the results, we employed the QDEGSAC algorithm with $\tau = 3$ pixels. We have experimentally verified that $\Delta = 5$ satisfies the requirements discussed in section 4.3 and therefore the videos are processed at regularly spaced intervals of $\Delta = 5$ frames.

Two specialists assisted in the interpretation of the videos. In order to make the interpretation of the results easy, the specialists were requested to indicate important video segments, where the visual content could be represented by a single frame. Surprisingly, we verified that the specialists missed some relevant video segments (that were pointed out in our *ground truth* set). In diagnostic hysteroscopy, it is common to capture different relevant views of the uterine region in subsequent micro camera motions, i.e. an uterine region is captured by one view and then captured again in an even better view of the same uterine region. Specialists tend to consider both image sequences as relevant (see Section 1.2.1), and that generates redundancy because the video segments selected by the experts often are part of a subset of relevant video segments. Thus, in the remaining of this paper we will distinguish the set of *important* video segments selected by the specialists from the set of *relevant* video segments in a given hysteroscopy video. The *important* video segments is a reduced subset of the *relevant* video segments.

Fig 14 shows the entire set of video segments trees computed throughout the videos. Recall that frames that share a high number of consistent point tracks are grouped first in the iterative process, which stops when the overlap between neighboring frames (or neighboring video segments) drops below a threshold $\zeta$. We have experimentally found that $\zeta$ can be set in the range $[1, 10]$ without noticeable changes in the resulting set of video segments, therefore we set $\zeta = 10$ in our experiments. Thus, the tree structure results from frame sequences associated with slow camera motion.

Fig 15 shows in detail 8 segment trees, and the associated key-frames computed on a smaller sequence of 700 frames from the video *hyst1*. Note that this sequence starts with a *panoramic* examination, then the operator runs fast through irrelevant frames, until he reaches an uterine region in which he spends most of time diagnosing potential disorders.

Table 3 shows a summary of the results and a comparison between the proposed approach and the method proposed by Scharcanski and Gavião (2006). The Precision measure mentioned in Table 3 quantifies the performance of the methods in discarding irrelevant frames (as defined in section 1.2.1) and is defined as follows:

$$Precision = \frac{\#relevantframes}{\#relevantframes + \#irrelevantframes}. \tag{13}$$

Note that the proposed approach (M1) produces less *false positives* (i. e. higher precision) and a significant reduction in the number of key-frames (without discarding relevant information). Data-rate reduction quantifies the decrease in the number of frames in the final summary in relation to the number of frames $N$ in the video. As a refinement of our approach, we discard segment trees associated with video segments that last less than 1/3 seconds. Otherwise, the data-rate reduction values would be 89%, 87,7%, 83,6% and 83,4% for *hyst1*, *hyst2*, *hyst3* and *hyst4*, respectively.

In Section 1.1, we discussed some of the drawbacks of applying generic video summarization methods to diagnostic hysteroscopy videos. Also, we tested state-of-art methods designed for generic video summarization on hysteroscopy videos, illustrating that these methods are not adequate for hysteroscopies. In the Content-Based Visual Information Retrieval (CBVIR) literature we find several cluster analysis techniques that extract temporal-structural information from video sequences automatically. In these cases, similar adjacent frames are grouped in shot units, and such frame clusters are further grouped into scenes, and also in larger video structural components (Dimitrova et al. (2002)). Visual data clustering is a central concept in many recent state-of-art summarization methods (Ngo et al. (2005); Gao et al. (2009); Hanjalic and Zhang (1999); Rasheed and Shah (2005); Zhu et al. (2005)), and devising new methods for adjusting thresholds is an ongoing challenging research issue, since the efficiency and the results obtained by these methods often depend on the threshold settings. Therefore, we avoid critically tuning thresholds by adopting in our tests the adaptive unsupervised cluster analysis approach proposed by Hanjalic and Zhang (1999). We observed experimentally that such methods often provide : (a) good data-rate reduction, but the set of key-frames tend to not include representative frames from each important video segment for the specialists, i. e. *false negatives* are often found; and (b) frames of distinct hysteroscopy examination phases tend to be grouped together in the same cluster, which is not desirable (see Section 1.2.3). This is inconvenient in the daily medical practice because: (1) important information is omitted in video summary, and (2)

Table 3: Summary of results: A comparison between the proposed method (M1) and the method (M2) proposed by Scharcanski and Gavião (2006).

| Videos | *hyst1* | | *hyst2* | | *hyst3* | | *hyst4* | |
|---|---|---|---|---|---|---|---|---|
| Methods | M1 | M2 | M1 | M2 | M1 | M2 | M1 | M2 |
| Data-rate reduction (%) | 97.5 | 96.1 | 97.6 | 94.6 | 97.2 | 95.8 | 97.5 | 93.9 |
| Number of key-frames | 76 | 117 | 51 | 111 | 154 | 226 | 191 | 456 |
| *Precision* | 0.98 | 0.97 | 1 | 0.91 | 0.92 | 0.84 | 0.90 | 0.82 |

the contents of each hysteroscopy phase are semantically distinct and should not be mixed in the resulting video summary.

## 6. Discussion

Our main goal is to measure changes on the field of view that result from the camera motion. We do not treat occlusions, consequently point tracks that are lost due to, for example, specular highlights affect negatively the proposed content overlap metric (since the loss of those tracks were not actually caused by the camera motion). Thus, potentially better results could be achieved by considering only point tracks that moved out of the field of view, since neighboring frames which are very similar in terms of content (except for specular highlights) may be placed incorrectly in different video segments.

In some videos, we found few irrelevant video segments in which the camera remains static for few seconds. According to experienced gynecologists, it may be attributed to the inexperience of some specialists in the hysteroscopy procedure, or due to the existence of undesired obstacles that sometimes are difficult to overcome, such as mucus and bubbles (see section 1.2.1) stuck on the hysteroscope tip. Our approach can not deal with these situations very well, since a sufficient number of correspondences, which are consistent with a rigid motion model, are detected in an irrelevant video segment. This downgrades the precision of the summaries delivered by our method.

The most intensive computational effort of our approach is the point tracking process. It takes hours to process an entire hysteroscopy video on an desktop computer with Intel Core 2 Duo with 2 processors at 1.66GHz and 2GB of RAM. Our approach is not intended to deliver results in real time, however the performance of tracking process could be dramatically improved by employing the GPU-based KLT tracker proposed by Sinha et al. (2006).

An important issue which has been under investigation is the capability of KLT method in tracking image points through relevant hysteroscopy video segments, given the well-known difficulties involved in establishing correspondences among widely separated frames/views. At conventional frame rates, our experiments indicate that the specialist does not move the camera fast enough to make the KLT tracker lose point tracks due to the fast camera motion, regardless of irrelevant hysteroscopy video segments whose characteristics are presented in section 1.2.1.

From a methodological point of view, our approach operates locally in the video sequence, since only neighboring frames are considered in the process of summarization. Usually, with the purpose of producing compact summaries, generic video summarization methods estimate how informative individual frames are by analyzing their properties in the context of the entire video sequence. We have found that this strategy may lead to unacceptable results in hysteroscopy videos. It is not reasonable to compare frames from distinct phases of a diagnostic hysteroscopy examination video. Thus, our method is designed to deal specifically with the hysteroscopy videos, but it may not be the best choice for summarizing generic videos.

According to our preliminary experiments, the proposed approach can produce very compact video summaries, and these summaries include frames from every important video segment selected by specialists. In addition, in comparison with a state-of-art method, the proposed approach not only produces more compact summaries (in terms of number of key-frames) by an average of 44%, but also produces more precise summaries, which contain less *false positives* than comparable state-of-art methods (see Table 3).

## 7. Conclusion and future work

We have presented an approach for detecting the video segments that attracted the specialist attention during a diagnostic hysteroscopy video acquisition. The video segments relevance are estimated in proportion to the attention they attract (i.e. their redundancy), and these information are used to produce rich video summaries for fast browsing. Relevant video segments are detected by tracking and classifying point matches in frames along the video sequence that are consistent with a parametric camera motion model. The point tracking task is challenging, and we performed a detailed evaluation to show the limitations of the proposed methodology. Our experiments suggest that consistent point tracking along the video sequence can be used to assign relevance to diagnostic hysteroscopy video segments (i.e. to the hysteroscopy video contents).

In this paper we also proposed a hierarchical scheme for browsing diagnostic hysteroscopy videos. Although the proposed technique can be computationally complex, the experimental results suggests that compact video summaries can be produced without discarding important information as *false negatives.* Comparing our approach with other state-of-art methods, our experiments suggest that our method tends to produce hysteroscopy video summaries that are nearly 44% more compact, in average (i.e., our method tends to produce less *false positives* than comparable state-of-art methods). Besides, the proposed representation is flexible, and allows the user to organize the hysteroscopy video contents in a video summary with a desired level of details, minimizing the occurrence of spurious frames in the video summary. Future work will concentrate on improving our method by introducing a mosaic representation for key-frames within degenerate scene configurations.

## Acknowledgment

The authors thank Dr. Paulo Cará for providing access to the library of hysteroscopy videos and for his useful advice. We also thank CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) and CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), Brazil, for financial support.

## References

Birchfield, S., 2009. KLT: An implementation of the kanade-lucas-tomasi feature tracker. http://vision.stanford.edu/birch/klt/.

Bouguet, J.Y., 2009. Camera Calibration Toolbox for Matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/.

Bouthemy, P., Gelgon, M., Ganansia, F., 1999. A unified approach to shot change detection and camera motion characterization. IEEE Trans. on Circuits Systems for Video Technology 9, 1030–1044.

Bovik, A., 2009. The essential guide to video processing. Academic Press/Elsevier, Amsterdam ;;Boston. 2nd ed. edition.

Chang, S., 2002. The holy grail of content-based media analysis. IEEE Multimedia 9, 6–10.

Chum, O., Werner, T., Matas, J., 2005. Two-view geometry estimation unaffected by a dominant plane, in: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, pp. 772–779.

Clark, T.J., Voit, D., Gupta, J.K., Hyde, C., Song, F., Khan, K.S., 2002. Accuracy of hysteroscopy in the diagnosis of endometrial cancer and hyperplasia: a systematic quantitative review. The Journal of the American Medical Association 288, 1610–1621.

Cunha-Filho, J.S., Arbo, E., Sloczinski, C.R., Berton, G., Gavião Neto, W.P., Genro, V.K., 2008. Variability of endometrial glandular opening count in infertile patients prior to first ivf treatment. Reproductive BioMedicine Online 17, 564–568.

Cunha-Filho, J.S., Scharcanski, J., Gaviao, W., Passos, P.E., 2004. Digital hysteroscopy: a new diagnostic method for the mid-secretory endometrium., in: 60th Annual Meeting of the American Society for Reproductive Medicine, ASRM, Philadelphia, USA.

Del Bimbo, A., 1999. Visual Information Retrieval. Morgan Kaufmann, San Francisco, USA.

Dimitrova, N., Zhang, H.J., Shahraray, B., Sezan, I., Huang, T., Zakhor, A., 2002. Applications of video-content analysis and retrieval. IEEE MultiMedia 9, 42–55.

Duan, L.Y., Jin, J.S., Tian, Q., Xu, C.S., 2006. Nonparametric motion characterization for robust classification of camera motion patterns. IEEE Trans. on Multimedia 8, 323–340.

Fischler, M., Bolles, R., 1981. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. Communications of the ACM 24, 381–385.

Frahm, J.M., Pollefeys, M., 2006. Ransac for (quasi-)degenerate data (qdegsac), in: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, New York, USA. pp. 453–460.

Gao, Y., Wang, W.B., Yong, J.H., Gu, H.J., 2009. Dynamic video summarization using two-level redundancy detection. Multimedia Tools and Applications 42, 233–250.

Gavião, W., Scharcanski, J., Passos, E.P., Cunha-Filho, J.S., 2007. Evaluating the mid-secretory endometrium appearance using hysteroscopic digital video summarization. Image and Vision Computing 25, 70–77.

Golub, C., Loan, C., 1989. Matrix computations. John Hopkins University Press, Baltimore. 2nd edition.

Hamou, J., 1991. Hysteroscopy and Microcolpohysteroscopy, Text and Atlas. Appleton and Lange, USA.

Hanjalic, A., Zhang, H., 1999. An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis. IEEE Trans. on Circuits Systems for Video Technology 9, 1280–1289.

Harris, C., Stephens, M., 1988. A combined edge and corner detector, in: Proc. of 4th Alvey Vision Conference, pp. 147–151.

Hartley, R., 1997. In defense of the eight-point algorithm. IEEE Trans. on Pattern Analysis and Machine Intelligence 19, 580–593.

Hartley, R., Zisserman, A., 2000. Multiple View Geometry in Computer Vision. Cambridge University Press.

Heikkilä, J., Silvén, O., 1997. A four-step camera calibration procedure with implicit image correction, in: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition.

Ho, Y.H., Lin, C.W., Chen, J.F., Liao, H.Y.M., 2006. Fast coarse-to-fine video retrieval using shot-level spatio-temporal statistics. IEEE Trans. on Circuits Systems for Video Technology 16, 642–648.

Irani, M., Anandan, P., 1998. A unified approach to moving object detection in 2d and 3d scenes. IEEE Trans. on Pattern Analysis and Machine Intelligence 20, 577–589.

Irani, M., Sawhney, H.S., Kumar, R., Anandan, P., 1997. Interactive content-based video indexing and browsing, in: Proc. of IEEE Workshop on Multimedia Signal Processing, pp. 313–318.

Lew, M.S. (Ed.), 2001. Principles of Visual Information Retrieval. Springer-Verlag, London, UK.

Li, S.C., Pan, P., Yao, S.Z., Feng, M., Wu, J.H., Su, Y., Liu, B.Y., 2010. Hysteroscopic appearence of midsecretory endometrium in relation to pinopodes expression and the reproductive outcome in infertile women. Journal of Reproduction and Contraception 21, 17–26.

Liu, T., Zhang, H., Qi, F., 2003. A novel video key-frame-extraction algorithm based on perceived motion energy model. IEEE Trans. on Circuits Systems for Video Technology 13, 1006–1013.

Longuet-Higgins, H.C., 1981. A computer algorithm for reconstructing a scene from two projections. Nature 293, 133–135.

Longuet-Higgins, H.C., Prazdny, K., 1980. The interpretation of a moving retinal image, in: Proc. Royal Soc. London, pp. 385–397.

Lucas, B.D., Kanade, T., 1981. An iterative image registration technique with an application to stereo vision, in: Proc. of Int. Joint Conf. Artificial Intelligence, pp. 674–679.

Ma, Y., Soatto, S., Kosecka, J., Sastry, S.S., 2003. An Invitation to 3-D Vision: From Images to Geometric Models. Springer Verlag.

Ma, Y.F., Hua, X.S., Lu, L., Zhang, H.J., 2005. A generic framework of user attention model and its application in video summarization. IEEE Trans. on Multimedia 7, 907–919.

Masamoto, H., Nakama, K., Kanazawa, K., 2000. Hysteroscopic appearance of mid-secretory endometrium: relationship to early phase pregnancy outcome after implantation. Human Reproduction 15, 2112–2118.

Ngo, C.W., Pong, T.C., Zhang, H.J., 2001. Recent advances in content-based video analysis. Int. Journal of Image and Graphics 1, 445–468.

Ngo, C.W., Pong, T.C., Zhang, H.J., 2003. Motion analysis and segmentation through spatio-temporal slices processing. IEEE Trans. on Image Processing 12, 341–355.

Ngo, C.W., Pong, Y.F.M., Zhang, H.J., 2005. Video summarization and scene detection by graph modeling. IEEE Trans. on Circuits Systems for Video Technology 15, 296–305.

Nistér, D., 2000. Reconstruction from uncalibrated sequences with a hierarchy of trifocal tensors, in: Proc. of European Conf. on Computer Vision, pp. 649–663.

Oh, J., Hwang, S., Lee, J., Tavanapong, W., Wong, J., Groen, P.C., 2007. Informative frame classification for endoscopy video. Medical Image Analysis 11, 110–127.

Peyrard, N., Bouthemy, P., 2005. Motion-based selection of relevant video segments for video summarization. Multimedia Tools and Applications , 259–276.

Piriou, G., Bouthemy, P., Yao, J.F., 2006. Recognition of dynamic video contents with global probabilistic models of visual motion. IEEE Trans. on Image Processing 15, 3418–3431.

Rai, L., Merritt, S.A., Higgins, W.E., 2006. Real-time image-based guidance method for lung-cancer assessment, in: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition.

Rasheed, Z., Shah, M., 2005. Detection and representation of scenes in videos. IEEE Transactions on Multimedia 7, 1097–1105.

Rothganger, F., Lazebnik, S., Schmid, C., Ponce, J., 2007. Segmenting, modeling, and matching video clips containing multiple moving objects. IEEE Trans. on Pattern Analysis and Machine Intelligence 29, 477–491.

Scharcanski, J., Gavião, W., 2006. Hierarchical summarization of diagnostic hysteroscopy videos, in: Proc. of IEEE Int. Conf. on Image Processing, Atlanta, EUA.

Scharcanski, J., Gavião, W., Cunha-Filho, J.S., 2005. Diagnostic hysteroscopy summarization and browsing, in: Proc. of Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society, Shanghai, China.

Shi, J., Tomasi, C., 1994. Good features to track, in: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition.

Sim, K., Hartley, R., 2006. Recovering camera motion using L infinity minimization, in: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1230–1237.

Sinha, S., Frahm, J.M., Pollefeys, M., Genc, Y., 2006. Gpu-based video feature tracking and matching, in: Proc. of Workshop on Edge Computing Using New Commodity Architectures.

Tan, Y.P., Saur, D.D., Kulkarni, S.R., Ramadge, P.J., 2000. Rapid estimation of camera motion from compressed video with application to video annotation. IEEE Trans. on Circuits Systems for Video Technology 10, 133–146.

Torr, P.H.S., 1997. An assessment of information criteria for motion model selection, in: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition.

Torr, P.H.S., Fitzgibbon, A., Zisserman, A., 1999. The problem of degeneracy in structure and motion recovery from uncalibrated image sequences. Int. Journal of Computer Vision 32, 27–44.

Tsai, R.Y., 1987. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. IEEE Journal of Robotics and Automation 3, 323–344.

Vasconcelos, N., Lippman, A., 2000. Statistical models of video structure for content analysis and characterization. IEEE Trans. on Image Processing 9, 3–19.

Waizenegger, W., Feldmann, I., Schreer, O., 2008. Semantic annotation and retrieval of unedited video based on extraction of 3d camera motion, in: Proc. of Int. Workshop on Content-Based Multimedia Indexing, London, UK. pp. 265–271.

Wu, C.H., Sun, Y.N., Chang, C.C., 2007. Three-dimensional modeling from endoscopic video using geometric constraints via feature positioning. IEEE Trans. on Biomedical Engineering 54, 1199–1211.

Yamaguchi, T., Nakamoto, M., Sato, Y., Konishi, K., Hashizume, M., Sugano, N., Yoshikawa, H., Tamura, S., 2004. Development of a camera model and calibration procedure for oblique-viewing endoscopes. Computer Aided Surgery 9, 203–214.

You, J., Liu, G., Sun, L., Li, H., 2007. A multiple visual models based perceptive analysis framework for multilevel video summarization. IEEE Trans. on Circuits Systems for Video Technology 17, 273–285.

Zhang, Z., 1999. Flexible camera calibration by viewing a plane from unknown orientations, in: Proc. of IEEE Int. Conf. on Computer Vision, pp. 666–673.

Zhu, X., Elmagarmid, A., Xue, X., Wu, L., Catlin, A., 2005. Insight video: Toward hierarchical video content organization for efficient browsing, summarization and retrieval. IEEE Trans. on Multimedia 7, 648–666.

## Appendix A. Camera motion from image point correspondences

We briefly review some concepts used in our approach, like geometrical issues involved in the representation of 3-D static scenes and the degeneracies that can appear in estimating camera motion. More details can be found in Ma et al. (2003); Hartley and Zisserman (2000); Torr et al. (1999).

### *Appendix A.1. Feature extraction and tracking*

Usually, a first step in estimating camera motion from a video sequence consists in establishing point correspondences between the images. Ideally, such corresponding points should be projections of the same point in space, as illustrated for two images in Fig. A.16 ($\mathbf{x}$ and $\mathbf{x}'$ are projections of $\mathbf{X}$).

The feature extraction step consists of selecting a number of interesting points, which can be detected with subpixel accuracy using the Harris corner detector (Harris and Stephens (1988)). Once a set of interesting points have been selected in an image, the Kanade-Lucas-Tomasi (KLT) (Lucas and Kanade (1981); Shi and Tomasi (1994)) tracker can be used to search for point correspondences in the next images. The KLT tracker delivers a set of potential point correspondences $\{\mathbf{x}_i \rightarrow \mathbf{x'}_i\}$ between pairs of images. Each correspondence is computed independently of the others, without considering a global motion consistency, which would involve all computed correspondences. It is important to note that most tracking methods, like the KLT tracker, often deliver ambiguities and false matches (Ma et al. (2003)), and therefore subsequent steps must be robust to overcome such limitations.

### *Appendix A.2. Camera motion*

Consider two images of the same scene taken from a camera which is moving relative to this scene, where the coordinates of a 3-D world point $\mathbf{X} = (X, Y, Z, 1)^{\mathsf{T}}$, and the corresponding 2-D image point $\mathbf{x} = (x, y, 1)^{\mathsf{T}}$, are represented in homogeneous coordinates. Following a basic pinhole camera model, each image is associated to a camera projection matrix $\mathbf{P}$, and a 3-D world point $\mathbf{X}$ is projected on the first image plane as $\mathbf{x} = \mathbf{PX}$ and as $\mathbf{x}' = \mathbf{P}'\mathbf{X}$ on the second image plane (Hartley and Zisserman (2000)). Since $\mathbf{x}$ and $\mathbf{x}'$ are the images of the same 3-D world point, they are understood as an image point correspondence $\mathbf{x} \rightarrow \mathbf{x}'$. Assuming the camera is calibrated, the corresponding image points $\mathbf{x}$ and $\mathbf{x}'$ satisfy the *epipolar constraint* (Longuet-Higgins (1981))

$$\mathbf{x}'^{\mathsf{T}}\mathbf{E}\mathbf{x} = \mathbf{0}, \tag{A.1}$$

where $\mathbf{E}$ is a $3 \times 3$ matrix which encodes the relative pose (rotation $\mathbf{R}$ and translation $\mathbf{t}$) between the two cameras $\mathbf{C}$ and $\mathbf{C}'$, as shown in Fig A.16. $\mathbf{E}$ is called the *essential matrix* and, given a number of image point correspondences, the entries of $\mathbf{E}$ can be recovered from the set of the generated epipolar equations, as explained in Appendix A.3.

As shown in Fig. A.16, a 3-D world point $\mathbf{X}$ and their projections, which are the image points $\mathbf{x}$ and $\mathbf{x}'$, lie on the epipolar plane. This is an important property that is exploited to search for image point correspondences, and these correspondences will support an estimate of $\mathbf{E}$ (the camera motion). For each point $\mathbf{x}$ in the first image, there will be a corresponding *epipolar line* $l'$ in the second image. The epipolar line $l'$ is determined by the intersection of the epipolar plane with the second image plane, as illustrated in Fig. A.16. Thus, there is a projective mapping $\{\mathbf{x} \mapsto l' \mid \mathbf{x}' \subset l'\}$ which is represented by the essential matrix $\mathbf{E}$, where

$$l' = \mathbf{E}\mathbf{x} \quad \text{and} \quad l = \mathbf{E}^{\mathsf{T}}\mathbf{x}'. \tag{A.2}$$

The points $e$ and $e'$ are known as *epipoles* and they are determined by the intersection of the line connecting the camera centers (the baseline) with the image planes.

*Appendix A.3. Estimating the essential matrix*

Given a number of image point correspondences $\mathbf{x}_i \rightarrow \mathbf{x}'_i$, the epipolar constraint (Equation A.1) is true for any pair of them. Thus, every pair of corresponding points gives one constraint on $\mathbf{E}$. Since $\mathbf{E}$ is a $3 \times 3$ matrix which is determined up to a scale factor, it has $3 \times 3 - 1$ unknowns. Therefore, 8 pairs of corresponding points are sufficient to compute the entries of $\mathbf{E}$ with a linear algorithm. This is the essence of the *Eight-Point Algorithm* (Longuet-Higgins (1981); Hartley (1997)). Basically, the Equation A.1 is rewritten in terms of the known coordinates of corresponding points $\mathbf{x} = [x\ y\ 1]^{\mathsf{T}}$ and $\mathbf{x}' = [x'\ y'\ 1]^{\mathsf{T}}$:

$$[\ xx'\ yx'\ x'\ xy'\ yy'\ y'\ x\ y\ 1\ ]\ \mathbf{E}^s = 0 \tag{A.3}$$

with $\mathbf{E}^s = [E_{11}\ E_{12}\ E_{13}\ E_{21}\ E_{22}\ E_{23}\ E_{31}\ E_{32}\ E_{33}]^{\mathsf{T}}$, which is obtained by stacking the entries of the matrix $\mathbf{E}$. Thus, from a set of eight image point correspondences we can stack their coordinates in a matrix $\mathbf{A}$ and, in the absence of noise, the vector $\mathbf{E}^s$ will satisfy

$$\begin{bmatrix} x_1x'_1 & y_1x'_1 & x'_1 & x_1y'_1 & y_1y'_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ x_8x'_8 & y_8x'_8 & x'_8 & x_8y'_8 & y_8y'_8 & y'_8 & x_8 & y_8 & 1 \end{bmatrix} \mathbf{E}^s = \mathbf{A}\mathbf{E}^s = 0. \tag{A.4}$$

In the presence of noise, the solution of this system of equations can be approximated by employing the Singular Value Decomposition (SVD) (Golub and Loan (1989)), as long as the eight equations are linearly independent.

### *Appendix A.4. Degenerate scene configurations*

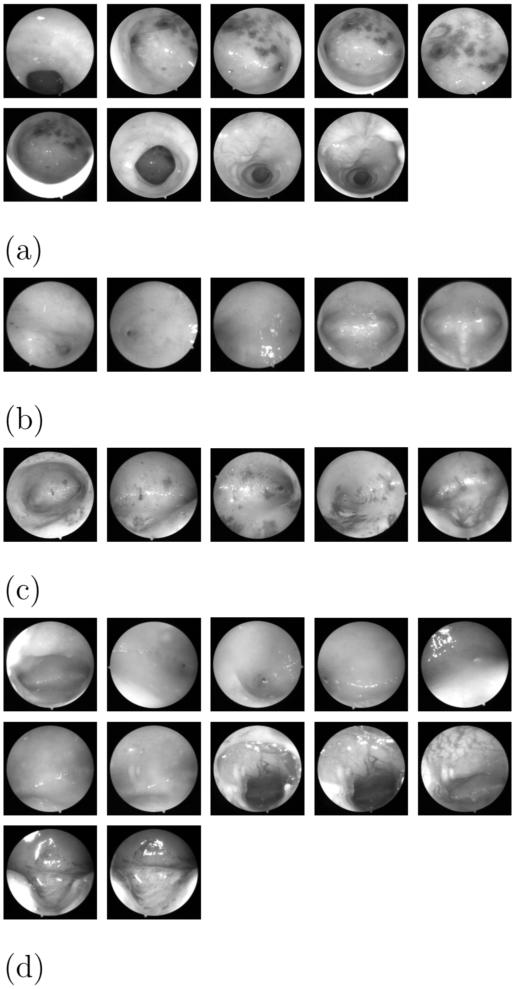
Successful methods for computing image point correspondences consistent with camera motion impose geometric constraints over putative image point matches, as introduced in Appendix A.2. However, there are some scene configurations in which geometric constraints fail to yield reliable results. Therefore, compute the essential matrix **E** from image correspondences requires some assumptions about the camera motion and scene structure. Scene configurations which do not conform to these assumptions are known as *degenerate configurations*.

Two types of degenerate scene configurations often appear in practice: *motion degeneracy*, when the camera undergoes a pure rotation around the optical center (there is no camera translation), and *structure degeneracy*, when the three-dimensional layout of the viewed scene is planar or, in a approximate way, the depth variation inside the scene is small compared to the distance between the camera and the scene. In both cases, it is not possible to determine the epipolar geometry, since the set of degenerate image correspondences does not provide enough constraints to compute **E** uniquely, and there will be a family of relations which will explain the data equally well (Hartley and Zisserman (2000)). Structure and motion degeneracies can be mathematically expressed in terms of the rank of the matrix **A** (Equation A.4). Under a noise-free context, the matrix **A** must have rank 8 to provide a unique solution, and it must have rank 6 when it is derived from structure or motion degenerate correspondences.

In the absence of noise the detection of these degenerate cases would not be too hard. However, starting from real (noisy) data, the problem is much harder since the remaining constraints in the equations can be determined by the noise. The effects of assuming non-degenerate scene configurations in image sequences include the loss of consistent image point tracks, as well as the inclusion of wrong image point matches due to over-fitting of the estimated geometric model, as reported by Torr et al. (1999).

### *Appendix A.5. Lens distortion correction and camera calibration*

Usually, the field of view of endoscopes is narrow and wide-angle lens are employed to overcome this limitation. However, wide-angle lenses cause radial distortion in the images captured through them. Therefore, a linear pinhole camera model can not be directly applied, since the image $\mathbf{x} = (x, y)$ of a 3-D world point $\mathbf{X}$ is projected away from the position determined by a perfect perspective projection, like that induced by a pinhole camera. The distortion coefficients and the calibration matrix $\mathbf{K}$ can be computed using well established camera calibration techniques (Zhang (1999); Tsai (1987); Heikkilä and Silvén (1997)). In this work, we use a calibration package (Bouguet (2009)) which is based on the work of Heikkilä and Silvén (1997).



(a)

(b)

(c)

(d)

Figure 6: Median frame of the video segments selected by the specialists in each video. (a) *hyst1*. (b) *hyst2*. (c) *hyst3*. (d) *hyst4*.

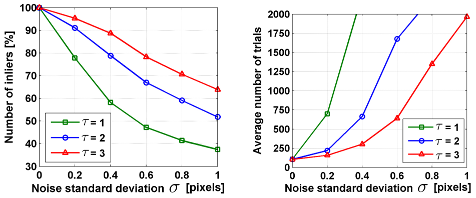


Figure 7: Number of computed inliers and the average number of QDEGSAC trials required to compute them as a function of noise for $|\mathbf{t}| = 5$ u.f.l. and $\alpha = 5$ degree (a generic scene configuration).

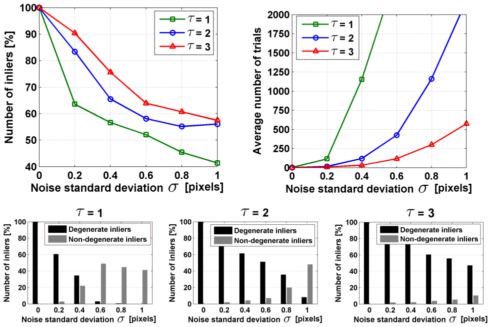


Figure 8: Performance of QDEGSAC for a planar (degenerate) scene configuration as a function of noise, where $(|\mathbf{t}|, \alpha) = (5, 5)$. (Top) Number of computed inliers and the average number of QDEGSAC trials required to compute them. (Bottom) Proportion of degenerate inliers computed for $\tau \in \{1, 2, 3\}$.

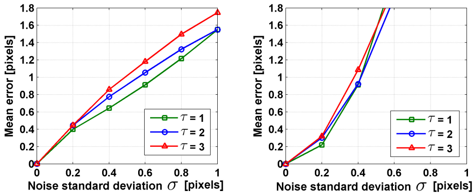


Figure 9: Mean errors computed over the sets of inliers delivered by QDEGSAC $\{in\}^{\tau}_{QDEGSAC}$ ($\tau \in \{1, 2, 3\}$) as a function of noise levels, where $(|\mathbf{t}|, \alpha) = (5, 5)$. (Left) Error computed under generic scenes. (Right) Error computed under planar (degenerate) scene configurations.

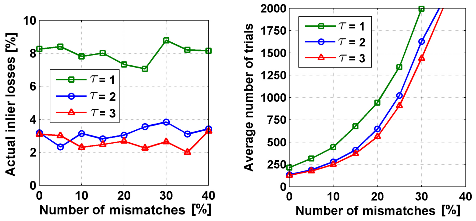


Figure 10: Performance of QDEGSAC as a function of the number of mismatches for generic scene configurations, where $|\mathbf{t}| = 5$ u.f.l., $\alpha = 5$ degree and $\sigma = 0.1$ pixels. (Left) Number of actual inlier losses and (right) the average number of required QDEGSAC trials.

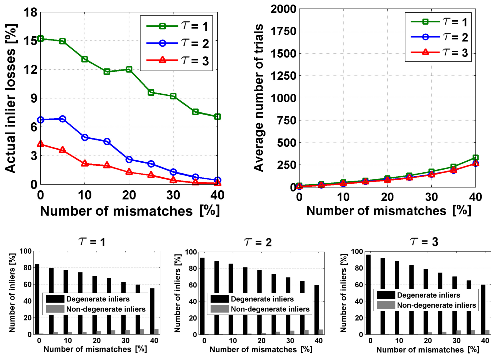


Figure 11: Performance of QDEGSAC as a function of the number of mismatches for a degenerate scene configuration, where $|\mathbf{t}| = 0$ u.f.l., $\alpha = 5$ degrees and $\sigma = 0.1$ pixels. (Top) Number of actual inlier losses and the average number of required QDEGSAC trials. (Bottom) Proportion of degenerate inliers for $\tau \in \{1, 2, 3\}$.

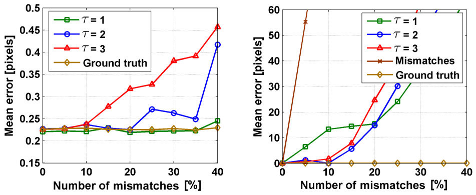


Figure 12: Mean errors computed over the sets of inliers delivered by QDEGSAC $\{in\}^{\tau}_{QDEGSAC}$ ($\tau \in \{1, 2, 3\}$) and the ground truth set of inliers $\{in\}_{Actual}$, as a function of the number of mismatches. (Left) Error computed under generic scene configurations with $(|\mathbf{t}|, \alpha) = (5, 5)$. (Right) Error computed under degenerate scenes, including the error computed over all point correspondences (including mismatches).

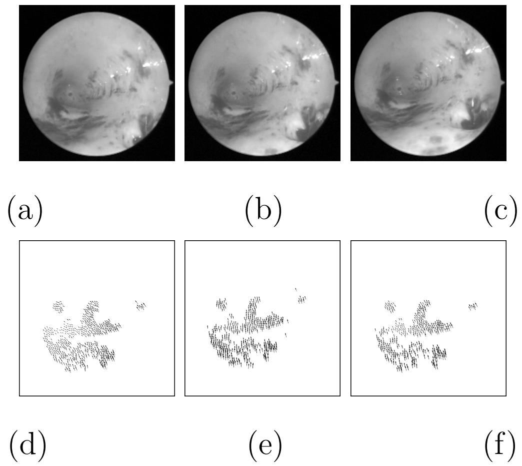


Figure 13: **Top**: frames 1, 10 e 20 of the *Tubal Orifice* sequence. **Bottom**: 2D displacements of the tentative point correspondences from the current view to the next ("→"). (a) Frame 1. (b) Frame 7. (c) Frame 13. (d) Two-dimensional displacements from frame 1 to 7. (e) Two-dimensional displacements from frame 7 to 13. (f) Two-dimensional displacements from frame 1 to 13.

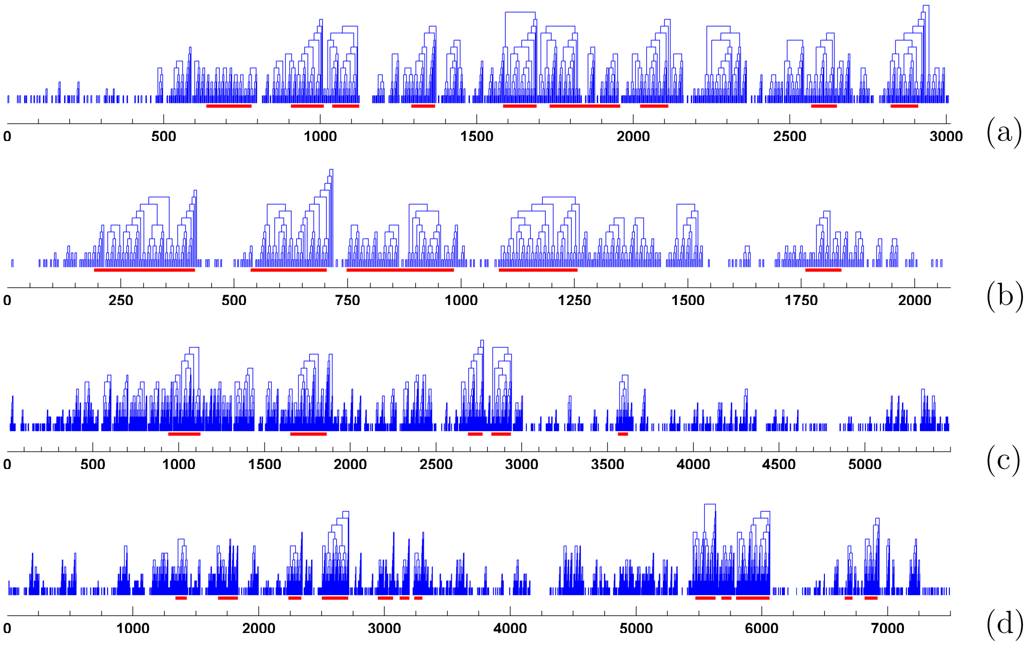


Figure 14: Segment trees computed throughout the videos. X-axis represents the frames in the video sequence. Horizontal line segments represent the important video segments manually selected by specialists. (a) *hyst1*. (b) *hyst2*. (c) *hyst3*. (d) *hyst4*. Notice the important video segments appear associated with noticeable trees.

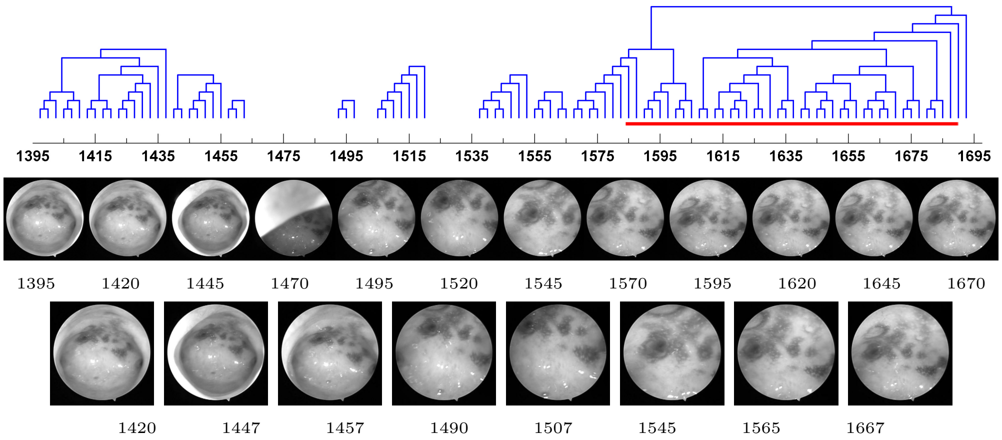


Figure 15: Segment trees and the associated key-frames computed over a smaller sequence of frames from the video *hyst1*. **Top** 8 segment trees as a function of the frame sequence. The solid horizontal line (red) represent the important video segment manually selected by the specialists. **Middle** The video sequence sampled at regularly spaced intervals of 25 frames. **Bottom** The 8 key-frames computed for each of the 8 segment trees. Note that all key-frames are clinically relevant, since they provide unobstructed views of uterus details, however only the 1667 is contained in the important video segment indicated by the specialists. We discard segment trees associated with video segments that lasts less than 1/3 seconds.

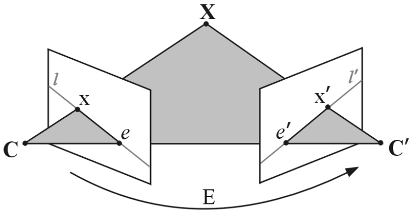


Figure A.16: Projections $\mathbf{x}$ and $\mathbf{x}'$ of a 3-D world point $\mathbf{X}$ lie on the same plane, which is known as epipolar plane. This is an important property that is widely exploited to search for correspondences, which will support the camera motion computation between $\mathbf{C}$ and $\mathbf{C}'$ in terms of $\mathbf{E}$.