Scalable comic-like video summaries.

Luis Herranz, Janko Ćalić, José M. Martínez, Marta Mrak

***Abstract*—This paper describes an efficient system for scalable video summarisation that exploits comic-like summary representations to facilitate interactivity and balance between content coverage and compactness. Due to visual disturbance induced by the transitions between scales, a new heuristic algorithm is proposed to restrict changes to bounded summary segments. Conducted user evaluations show that the proposed methodology improves usability while keeping the summaries compact and informative.**

***Index Terms*—video summarisation, scalability, comics, usability, user interface**

I. Introduction

THE ubiquitous proliferation of digital video has brought new challenges to the video management technology. The ever increasing demand for better user experience, coupled with plethora of delivery platforms and devices, highlighted the importance of effective interaction with large video collections. In order to enable this, video data needs to be abstracted in a user-friendly way. This process of abstraction is know as video summarisation and generates compact visual representations of content that can be intuitively comprehended in a fraction of time of the original. Video summaries play an important role in multimedia search and retrieval by making the browsing of large-scale video content easier and more efficient.

Traditionally, video summarisation methods have focused on selecting relevant pieces of content to create a useful representation [1], [2] either in the form of a storyboard or a video skim. Storyboards are compact fixed representations comprising a sequence of video stills (key-frames) extracted from the sequence, while the video skims comprise short video sequences assembled from the relevant sections of the source video. In terms of usability, comprehension of each video skim requires time, making them impractical in the large-scale video browsing context.

However, there has been an increasing interest in extending conventional storyboard video summaries with additional features and functionalities, such as customised and personalised summaries [3], [4] and multi-document summaries [5]. In addition, due to the user-centric nature of video summarisation, new types of storyboard presentations have been explored, such as comic-like summaries [6], [7], video posters [8] and video collages [9].

Having in mind the myriad of application contexts of video summarisation coupled with the ever-increasing demand for intuitive user experience, there is a growing interest in summarisation methodologies that support adaptation of summaries to dynamic contextual parameters. Due to the wide range of tasks, database sizes, sequence durations and end-user display sizes, one of the key functionalities of dynamic summarisation is to facilitate effortless choice of the presented level of detail, i.e. the *scale* of the summary.

L. Herranz and J.M. Martínez are with the Video Processing and Understanding Lab, Escuela Politécnica Superior, Universidad Autónoma de Madrid, 28049 Madrid, Spain.

J. Ćalić is with the I-Lab at the Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford GU2 7XH, UK

M. Mrak was with the University of Surrey, UK. She is now with the British Broadcasting Corporation, Research and Development (BBC R&D), UK

The notion of *scalability* has been extensively applied in video coding and adaptation providing multiple versions of the same video [10], and some features of scalable video streams have been utilised in video summarisation [11]. However, there has been very little focus on scalability of the summaries themselves [12], [13], where the scalable parameter is the length of the summaries.

This paper addresses the challenges of interactive scalable video summarisation presented in the comic-like form. The intuitive and familiar structure of comics is combined with the adaptation potential of scalable representations in order to achieve good coverage of content at arbitrary level of detail while maintaining intuitive interaction and adequate user experience. In addition to the issues of algorithmic efficiency and key-frame saliency that were previously investigated in detail [7], here we address the problem of multi-scale summaries and the closely related issue of visual disturbance. The visual disturbance is a distracting effect for human observers induced by discrete transitions between two scales in a comic-like or any other non-linear summary. The experimental results demonstrate that the proposed method for alleviation of the visual disturbance significantly improves usability of scalable comic-like video summaries.

The rest of the paper is organised as follows. Section II outlines related work on visual representation of summaries, followed by an introduction of scalable comic-like summaries and methods to generate single scale summaries in Section III. The proposed algorithms for generation of scalable summaries and methods for alleviation of the induced visual disturbance are given in Sections IV and V respectively. The experimental setup and the results of the system evaluation are presented in Section VI, followed by the conclusions and future work.

II. Related Work

Being one of the most appealing visual abstractions of video content, comic-like summaries have been proposed as user-friendly and easily-readable representations [7]. By exploiting the narrative structure of "spatially juxtaposed images in deliberate sequence intended to convey information" [14], comics are able to use spatial relations of their imagery to convey the notion of time. In contrast to the conventional storyboards, the narrative structure of comic-like video summaries is more complex and utilises images of different sizes, laid out so that its position and scale can convey an estimated frame importance.

A similar approach was introduced in the form of video posters [8], proposing a pictorial representation with variable image sizes to summarise the most dramatic incident taking place in a video sequence. However, video posters do not necessarily follow the temporal structure of the video, but the pattern of the video poster is selected among a few predefined patterns. In addition, it was reported that each video poster is limited to a maximum of 16 images.

Addressing this limitation, a number of methods [6], [7] have proposed more efficient algorithms capable of generating larger layouts by following the comic-like structure. The problem of optimal image layout in a comic-like visual structure is usually posed as an NP-hard combinatorial optimisation problem, making the full search methods impractical for large number of images. One way of addressing this problem is to apply suboptimal algorithms based on heuristic simplifications [6]. However, as proposed in [7], nearly

optimal performance can be achieved by utilising a fast suboptimal algorithm suitable for large layouts due to its linear complexity.

The majority of summarisation approaches create a single video summary of a certain length or duration. However, it is often desirable to generate a representation that provides different levels of detail for the same video content, balancing the presented amount of information and the summary length. This approach generates scalable video summaries, where complexity and/or length can be adjusted, without the need to rerun the entire summarisation algorithm again. The scalable summaries can be utilised in many applications, ranging from the customised adaptation of video summaries to a given length and progressive video access, to visualisation and interactive video browsing.

Hierarchical approaches to scalable summarisation proposed in [12] can provide some level of scalability, as they create summaries based on a narrative hierarchy (e.g. chapters, scenes, shots, frames), the number of scales is very limited. These scales provide a coarse scalability, which is mainly exploited in hierarchical browsing applications, where different levels of detail can be presented for the user-selected parts.

However, scalable summarisation aims at a finer scalability in order to deliver results in scenarios requiring fine adjustment of the summary length, such as interactive browsing and search. Having this in mind, a representation of video sequences based on a priority curve is proposed in [15]. When the priority curve is computed, a summary of any desired length can be easily created. However, the main drawback of this method is that it needs manual annotation of the sequence. An iterative growing algorithm [13] has been proposed to generate scalable storyboards and video skims with fine granularity, but these representation may be not suitable for visualising large-scale video summaries. Having this in mind, this paper proposes a summarisation framework for large-scale video data by combining efficient scalability and comic-like summaries in a user-friendly manner.

III. Scalable Comic-like Summaries

As mentioned above, a storyboard is defined as a sequence of images of the same size, displayed in a temporally ordered manner following the typical spatial layout from left to right and from top to bottom. The majority of storyboarding algorithms attempts to select as few images as possible, while covering the most of information present in the video sequence. This results in the removal of redundancy by minimising repetition of similar images.

However, repetition of similar images can provide extra information such as the duration or activity of a specific event, sequence structure or unexpected content. In some cases, this extra information is very useful and it is preferred to a more compact summary, providing more intuitive coverage of every part in the video sequence. From this perspective, comic-like summaries are very useful, as they can adapt the size of displayed images according to their relevance, i.e. a key-frame representing a shot may be surrounded by other smaller auxiliary frames that provide additional information about the temporal evolution of that shot.

In this paper, we propose an approach that utilises scalable comic-like summaries, providing arbitrary levels of detail and length, so that the users or applications can select their optimal scale: from the coarsest storyboard representation to the other extreme of detailed comic-like summary. There are two application contexts of scalable comic-like summaries: i) adaptation to specific constraints to the length of the summary by user preferences or usage contexts, and ii) progressive visualisation and interactive navigation, where users visualise multiple scales in a progressive manner, usually from coarser to finer scales. Following the definition of comics as a sequential art where space does the same as time does for film [14], this work intuitively transforms the temporal dimension of videos into the spatial dimension of the final summary by following the rules of comic narrative structure.

In the proposed video summarisation framework, depicted in Figure 1, we formulate the scalable comic-like summaries as an extension of conventional storyboards. The coarsest scale of the summary, with the lowest level of detail matches the conventional storyboard, i.e. it is a special case of comic-like summary with constant size and a trivial layout. As the new images are included at finer scales, the summary is enriched with new details yet maintaining the flow of temporal events. These images are conveniently scaled according to their importance and laid out into a spatial structure, which becomes more complex as the scale increases. The proposed layout algorithm considers only single row layouts to minimise complexity and to facilitate responsive interaction with summaries. In case the summary becomes too long, the browsing device splits it into several rows.

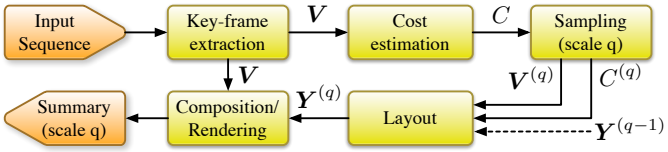

Figure 1. Architecture of the proposed framework.

The set of key-frames $\boldsymbol{V}$ is initially extracted from the input video sequence. The importance of each key-frame is estimated and stored in the cost function C, where the cost is proportional to the expected size of key-frame in the final summary. Based on the cost function and the scale q, the sequence of key-frames and corresponding cost values are sampled into subsets $\boldsymbol{V}^{(q)}$ and $C^{(q)}$, feeding the layout module that computes the summary layout $\boldsymbol{Y}^{(q)}$ for that scale, potentially relying on the layout at the previous scale $\boldsymbol{Y}^{(q-1)}$. Finally, with this information, the summary can be composed and rendered. We denote the comic-like summary $\mathcal{C} = \{\boldsymbol{Y}, \boldsymbol{V}\}$ as a pair of layout $\boldsymbol{Y}$ and key-frames $\boldsymbol{V} = (f_1, f_2, \ldots, f_N)$, with $I_{\boldsymbol{V}} = \{1, 2, \ldots, N\}$ representing the set of indices of $\boldsymbol{V}$.

The layout $\boldsymbol{Y}$ is defined as a sequence of indices of panels. A panel is the basic spatial unit of comics and it comprises an ordered pictorial sequence conveying information in the temporal order. The summary is composed by laying out the images following a sequence of panels, each of them based on a panel template. Given the height h of the row in a summary, there is a finite set of available panel templates that can be generated by the layout algorithm [7]. Let us denote a panel as a pair $P = \{p, I_P\}$, where p indicates the index of the panel template in the template set and $I_P \subseteq I_{\boldsymbol{V}}$ the sequence of indices of the key-frames in the panel. The panel template is represented as a sequence of frame sizes $\boldsymbol{T}_p = (\Omega_1, \Omega_2, \ldots)$, where $\Omega_n \in \{1, 2, \ldots, h\}$ is the relative size of the n-th key-frame of the panel, while $|\boldsymbol{T}_p|$ denotes the length of the panel template with index p, and $|\boldsymbol{T}|$ is the number of available panels in the template set.

A scalable comic-like summary is a set of comic-like summaries $\mathcal{CC} = \{\mathcal{C}^{(1)}, \cdots, \mathcal{C}^{(q)}, \cdots, \mathcal{C}^{(Q)}\}$, where the summary at the scale q is an ordered pair of layout and corresponding key-frames $\mathcal{C}^{(q)} = \{\mathbf{Y}^{(q)}, \mathbf{V}^{(q)}\}$ with indices $I_{\boldsymbol{V}^{(q)}} \subseteq I_{\boldsymbol{V}}$ arranged in temporal order. Due to the progressive nature of the summaries it can be observed that $\mathbf{V}^{(q)} \subset \mathbf{V}^{(q+1)}$ and $I_{\mathbf{V}}^{(q)} \subset I_{\mathbf{V}}^{(q+1)}$. In order to use the same set of key-frames $\boldsymbol{V}$ for all scales, in the mathematical expressions we will use $I_{\boldsymbol{V}^{(q)}}$ rather than $\boldsymbol{V}^{(q)}$.

Having in mind the requirement for invariance of the summarisation framework to the type of video content processed, the set of

key-frames used to generate the scalable summaries has been taken from different external sources. In the conducted experiments, two different methods for key-frame extraction were used: i) a camera-work based key-frame extractor described in detail in [16], and ii) benchmark algorithm to create the TRECVID video summarisation ground truth [17]. In order to generate an intuitive and easily readable summary, the significance of a key-frame in the final layout is conveyed by its size. The significance (i.e. the desired size) is estimated and stored in the cost function $C = (C_n | n \in I_{\boldsymbol{V}})$, where $C_n \in [0, 1]$. The method for calculation of the cost function C has been adopted from [7].

IV. Multiscale Layout

The main task of the layout algorithm is to find a layout that optimally follows the values of the cost function using only sizes available in panel templates, as depicted in Figure 2. Each panel template generates a vector of frame sizes that approximates the cost function values of corresponding frames. At any given scale q, a layout is a sequence of L panel templates $\boldsymbol{Y}^{(q)} = (p_1, p_2, \ldots, p_L)$ that follows the temporal structure of the video sequence. By unfolding the layout, the sequence of N frame sizes $\boldsymbol{\Omega}^{(q)} = \boldsymbol{\Omega}(\boldsymbol{Y}^{(q)})$ is:

$$\boldsymbol{\Omega}^{(q)} = \left(\overbrace{\Omega_1, \Omega_2, \ldots, \Omega_{|\boldsymbol{T}_{p_1}|}}^{\boldsymbol{T}_{p_1}}, \overbrace{\Omega_{|\boldsymbol{T}_{p_1}|+1}, \ldots,}^{\boldsymbol{T}_{p_2}} \Omega_n, \ldots, \overbrace{\Omega_N}^{\boldsymbol{T}_{p_L}} \right) \quad (1)$$

The indices of the keyframes $I_{P_l}^{(q)}$ of each panel $P_l, \forall l \in \{1, \ldots, L\}$ are also selected as a partition of the initial set of indices $I_{\boldsymbol{V}}^{(q)}$ according to panel lengths:

$$I_{\boldsymbol{V}}^{(q)} = \left(\overbrace{1, 2, \ldots, |\boldsymbol{T}_{p_1}|}^{I_{P_1}}, \overbrace{|\boldsymbol{T}_{p_1}| + 1, \ldots, n}^{I_{P_2}}, \ldots, \overbrace{N}^{I_{P_L}} \right) \quad (2)$$

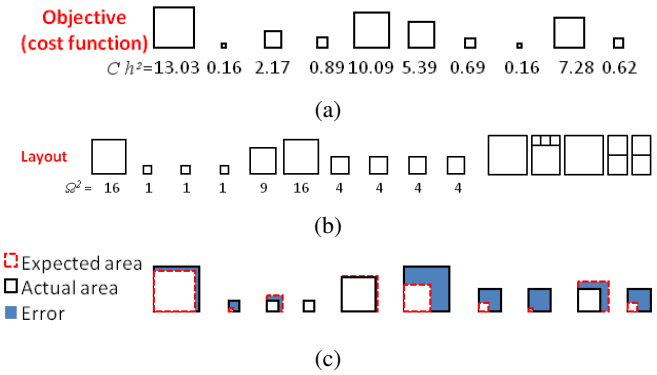

Figure 2. Cost-based approach to comic-like summaries: a) cost, b) solution (frame sizes and layout), and c) layout error.

The scalable layout algorithm comprises two stages: *key-frame sampling* and *layout*. The key-frame sampling algorithm selects a subset of indices from $I_{\boldsymbol{V}}$ and their cost values according to some sampling strategy. It is assumed that each scale is independent and the only constraint is that all images at any given scale q are present in all more detailed scales $q' > q$. A simple cost-based sampling strategy is applied: those indices with the $N^{(q)}$ highest cost values are selected, where $N^{(q)}$ is the number of images in the scale q.

The layout optimisation problem consists of finding a layout minimising the layout error $\varepsilon(\boldsymbol{Y})$ for a given a cost function $C_n^{(q)}$ using the data resulting from key-frame sampling $I_{\boldsymbol{V}^{(q)}}$:

$$\boldsymbol{Y}^{(q)} = \arg\min_{\boldsymbol{Y}} \left(\varepsilon(\boldsymbol{Y}) \right) \quad (3)$$

$$\varepsilon(\boldsymbol{Y}) = \arg\min_{\boldsymbol{Y}} \left(\sum_{l=1}^{L} \varepsilon(P_l^{(q)}) \right) \quad (4)$$

$$\varepsilon\left(P^{(q)}\right) = \sum_{i=1}^{\left|\boldsymbol{T}_{p^{(q)}}\right|} \left(C_{n_0+i-1}^{(q)} - \frac{\Omega_i^2}{h^2} \right)^2, \; n_0 = \min_{n \in I_{P^{(q)}}}(n) \quad (5)$$

Since the layout problem must be solved for every scale q, it is essential to minimise its complexity. Therefore, a dynamic programming approach described in detail in [7] is utilised. It balances algorithm efficiency with suboptimal layout error to achieve linear complexity.

This method of generating layouts independently and *a-priori* can be utilised in a number of scenarios in where users interacts with a single scale of the summary. One example of application that uses independent scales is summary adaptation, as the user gets a scaled version of the summary according to user's preferences or constraints in the usage environment (e.g. limited display area in the screen).

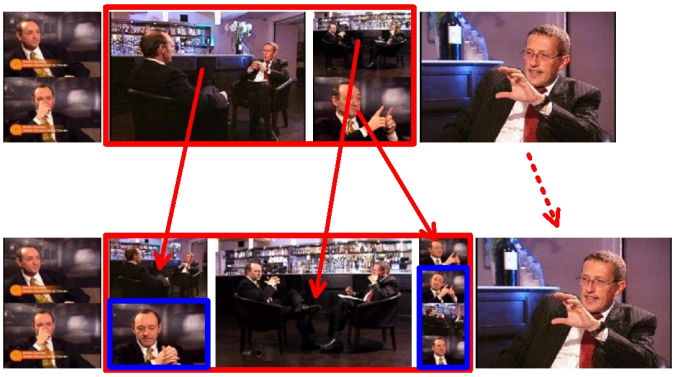

Figure 3. Example of transition between two consecutive scales.

However, in some application contexts (e.g. progressive visualisation or interactive browsing across scales), users have to visualise several scales interactively in relatively short time intervals. In our pilot trials, we observed that the main usability problem was to follow the changes during the transitions between scales. Due to the varied frame sizes in proposed summaries, transitions between consecutive scales become disturbing and uncomfortable, as some images may change their position and size in the new layout, and new panels may appear or disappear (see the example in Figure 3). Even if some panels are not modified, they can be pushed by others so they suffer a displacement, which may be also unpleasant if it is large or involves row changes. If these changes are scattered all over the summary and the delay between scales is too short, it becomes difficult to recover the structure of the summary. These undesirable effects hinder detection of new information (new images) added at the new scale, which should be the main objective of the scalable summarisation. Thus, minimising this problem becomes a dominant objective of our algorithm, as described in the following Section. We will refer to this effect as the *layout disturbance*.

V. Alleviating Layout Disturbance in Progressive Multiscale Summaries

In order to minimise the effect of layout disturbance we propose a heuristic algorithm, based on the concept of *anchor key-frames*, which can create more pleasant transitions between scales.

This heuristic algorithm is based on the idea of comic-like summaries as enhanced storyboards. From that point of view, the main and coarsest summary is the storyboard, representing the video only with the most representative key-frames (i.e. those with cost $C_n = 1$), entitled *anchor key-frames*. The remaining images are complementary, adding more information about the temporal evolution of the sequence and the duration of events. Therefore a set of heuristic rules is applied in the layout algorithm:

- Anchor key-frames are considered as the most relevant and must not change their size across scales, being always h and thus presented in a single panel, and
- The layout algorithm is not applied to the whole sequence of key-frames, but only to segments between those anchor key-frames with new key-frames in-between.

These conditions also help to limit the layout disturbance, as the number of changes between scales is restricted by design to only a fraction of the layout.

However, the main problem of cost-based sampling stems from the fact that the key-frames sampled for a new scale can be located at any position in the sequence. Consequently, the new images (and panels) can appear at any place in the layout, far from each other. That is the main source of layout disturbance, as it is more difficult to follow changes in the layout when they are spread across the summary. Taking into account the temporal order of key-frames in the sampling strategy is more suitable to avoid disturbance. The objective of this *temporally constrained sampling* is to include new batches of key-frames not only based on the cost function but also on its temporal location in the sequence. Thus, changes can be localised to a small area in the summary, i.e. a restricted temporal window of the sequence.

At each scale q, the sampling algorithm selects a batch of $M^{(q)}$ key-frames in a relatively short temporal interval, but all of them having a reasonably high cost. Anchor key-frames are the boundaries of these intervals. The first scale always returns the set of anchor key-frames. For the subsequent scales, the set of indices $I_{\boldsymbol{V}}$ is divided into L intervals, and a bin H_k is assigned to each interval k. Figure 4 depicts the sampling strategy, which comprises the following steps:

0) Input: $C, I_{\boldsymbol{V}}, I_{\boldsymbol{V}^{(q-1)}}, M^{(q)}$. Output: $I_{\boldsymbol{V}^{(q)}}, C^{(q)}$
1) Initialize histogram as $H_k = \emptyset$, $k = 1, \ldots, L$.
2) Sort the set of unselected key-frames at scale q by cost and let $I_{\boldsymbol{V}^*}$ be the sequence of their indices in decreasing cost order.
3) Loop over $I_{\boldsymbol{V}^*}$ until there are no more available indices. Let n^* be the first index in $I_{\boldsymbol{V}^*}$
   a) Find the interval k^* corresponding to n^* and set $H_{k^*} = H_{k^*} \bigcup n^*$
   b) If $|H_{k^*}| = M^{(q)}$ then go to step 6.
   c) If any available index in $I_{\boldsymbol{V}^*}$, let n^* be the next index in $I_{\boldsymbol{V}^*}$ and continue to step 3a.
4) Combine pairs of consecutive intervals so the histogram bins be $H'_k = H_k + H_{k+1}$
   a) If any $|H'_k| \geq M^{(q)}$ then $k^* = \arg\min_k |H'_k|$ and go to step 6.
5) If there is no interval satisfying $|H'_k| \geq M^{(q)}$, then continue combining intervals in increasing number (three intervals, then four, etc.)
6) Set $I_{\boldsymbol{V}^{(q)}} = I_{\boldsymbol{V}^{(q-1)}} \bigcup H_{k^*}$ and $C^{(q)} = (C_n | n \in I_{\boldsymbol{V}^{(q)}})$

Intuitively, the algorithm selects key-frames according to their cost in descending order, and tracks the number of key-frames selected from every interval. If one of the intervals reaches the number of required key-frames, the key-frames sampled in that interval are selected. If, after that first loop, there is not any interval with enough key-frames, adjacent intervals are combined and checked again.

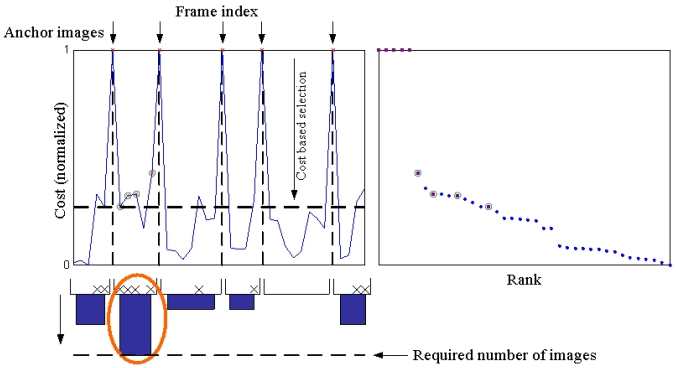

Figure 4. Illustration of the temporally constrained sampling algorithm.

In terms of the layout algorithm, the proposed heuristic rules localise the changes to a segment bounded by two consecutive anchor key-frames. Therefore the layout algorithm is applied only to that segment. The rest of the layout remains unchanged, and the only change the user perceives is the possible displacement due to other panels pushing them. Without loss of generality, the layout $\boldsymbol{Y}^{(q-1)}$ can be expressed as:

$$\boldsymbol{Y}^{(q-1)} = \left(\boldsymbol{Y_l}^{(q-1)}, \boldsymbol{T}_{anchor}, \boldsymbol{Y_m}^{(q-1)}, \boldsymbol{T}_{anchor}, \boldsymbol{Y_r}^{(q-1)}\right) \quad (6)$$

where $\boldsymbol{Y_l}^{(q-1)}$, $\boldsymbol{Y_m}^{(q-1)}$ and $\boldsymbol{Y_r}^{(q-1)}$ are the partial layouts at left, in-between and right of the anchor key-frames bounding the segment with new key-frames sampled at current scale q. These partial layouts are separated by two single key-frame panels $\boldsymbol{T}_{anchor}$:

$$\boldsymbol{Y}^{(q)} = \left(\boldsymbol{Y_l}^{(q-1)}, \boldsymbol{T}_{anchor}, \boldsymbol{Y_m}^{(q)}, \boldsymbol{T}_{anchor}, \boldsymbol{Y_r}^{(q-1)}\right) \quad (7)$$

where $\boldsymbol{Y_m}^{(q)}$ is the layout of newly sampled key-frames between the two anchor frames. Thus, a significant part of the summary is reused in the transition between scales $q-1$ and q. The previous formulation is only valid in the case of a single segment bounded by two consecutive anchor key-frames. If changes are spread in several segments, the layout algorithm is run independently for each of the segments bounded by consecutive anchor key-frames.

VI. Experimental Results

A. Experimental Setup

In order to evaluate the presentation and browsing of proposed summaries, a prototype of interface based on web technologies was developed. Instead of using images as main units for composition, the interface uses panels. Thus, it is easy to compose the summary and render it by laying out the panels from left to right and top to bottom (as a storyboard of panels), offering flexibility of the summary's shape and size.

The user interface was designed to be simple and intuitive. The pilot user tests showed that a suitable user interface was critical for the success of the proposed abstraction approach. In order highlight the changes that emerge between two scales, an option to enlarge newly added panels (by 130% in our experiments) and add a red frame around them was offered to the users (see Figure 5).

The proposed approach was tested for both independent summary and progressive summary scenarios, using the two algorithms described in this paper: basic algorithm (*basic*) and anchor based algorithm (*anchor*) with $h = 4$. The experiments were conducted using the key-frames from three different sequences sourced from

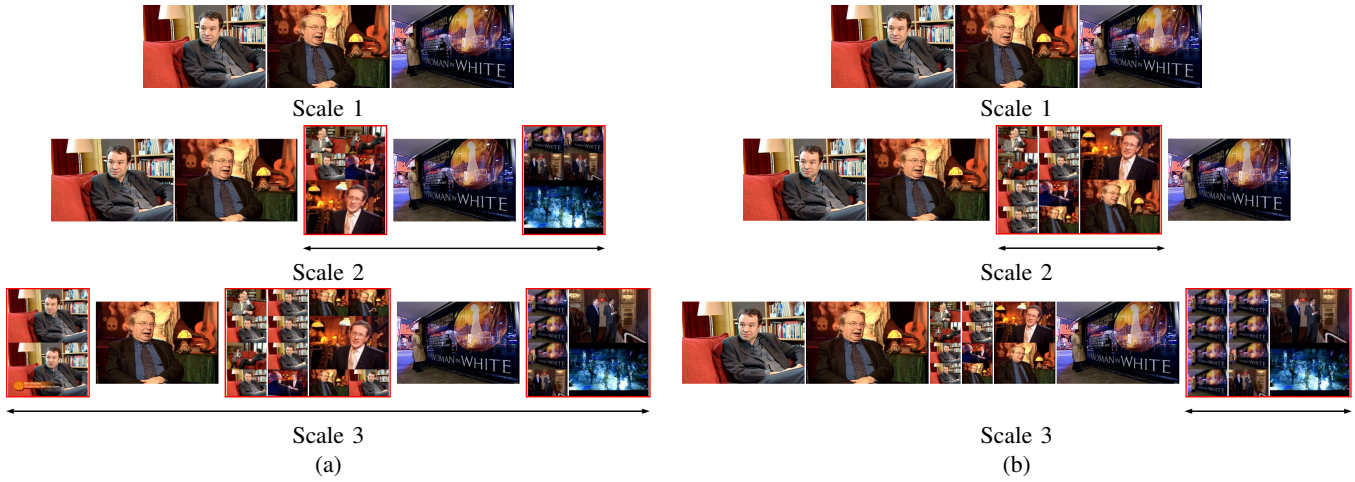

Figure 5. Example of summaries (scales 1-3, steps of 10 images): a) *basic*, b) *anchor*. New panels are highlighted.

two datasets (*Trec*: high redundancy clip from TRECVID BBC rushes corpus, *Franc*: medium redundancy clip from TRECVID BBC rushes corpus [18] and *Quest*: low redundancy clip from TURNER Broadcasting corpus [7]), in order to cover different levels of semantic redundancy and different number of keyframes. Table I shows the characteristics of the test sequences and the scalable summaries.

Table I
CHARACTERISTICS OF THE DATA SET AND SUMMARIES.

Sequence			Summary			
Name	Duration	Redundancy	#kf	#scales	Step	#clusters
Trec	34m17s	High	74	8	10	5
Franc	34m54s	Medium	255	13	20	10
Quest	20m14s	Low-medium	270	12	20	50

B. Objective Evaluation

As we discussed previously, the main motivation of the *anchor* algorithm was to reduce the effect of the visual disturbance, mainly due to uncontrolled changes between scales. We computed some measures related to how changes from consecutive scales are distributed over the layout. The first one is the span of the changes in the layout, measured as the distance between the first and the last image added in the new scale (see Figure 5 for an example). The results are shown in Figure 6a, where both methods are compared. Clearly, the temporally constrained sampling used in the *anchor* method helps to reduce this span for most scales. The last scale includes all the remaining images, so the span is considerably larger.

In a transition previous images are rearranged and combined with the new images, resulting in new panels coming into view while some panels may dissapear. Both effects are the main sources of disturbance in the layout. For that reason, we also compared the number of inserted and removed panels for every scale. In the *basic* algorithm, the inclusion of new images (even only one) may cause changes in all the subsequent panels. However, the *anchor* algorithm restricts this effect only to a part of the layout, with a smaller number of new and removed panels, as shown in Figure 6b. However, the number of total panels in the layout is very similar for both methods and scales.

C. User Evaluation

The summaries were also evaluated by 18 assessors according to some subjective criteria, in two different scenarios. For the evaluation,

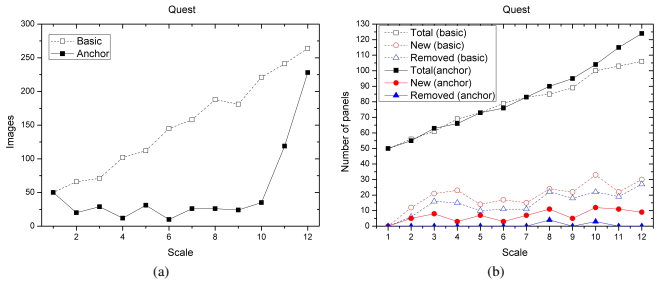

Figure 6. Comparison of *basic* and *anchor* methods for the *Quest* sequence: a) span of the layout change in panels, b) inserted and removed panels.

we used the web interface shown in Figure 5. The summaries were displayed on a large screen (1920x1200 pixels).

1) Scenario 1: Interactive summaries: In the first scenario, the assessors were free to interact with the interface and navigate across the scales of the summaries. In order to avoid biases, the name of the algorithm was hidden and the order of evaluation randomised. Results are shown in Table II. The satisfaction criterium was posed as an affirmative statement ("In general, the summary represents adequately the original content.") and evaluated using a typical Likert scale (1: Strongly disagree; 3: Nor agree nor disagree; 5: Strongly agree) [19]. In general, the results were very similar for both algorithms, and users were satisfied with the summaries. The assessors were also asked for their preference between both algorithms, with no clear preference except for a very slight preference for the *basic* algorithm for *Trec* and *Franc* sequences. Finally, the assessors were also asked about the utility of the interface ("The user interface helps to follow the changes across scales."), with a positive evaluation.

Table II
SUBJECTIVE RESULTS FOR THE INTERACTIVE SCENARIO.

		Trec	*Franc*	*Quest*
Satisfaction	Basic	4.4	4.0	3.9
	Anchor	4.4	3.9	4.0
Preference (5: Anchor - 1: Basic)		2.9	2.9	3.0
User interface		4.4	4.3	4.3

2) Scenario 2: Progressive summaries: In this second scenario, the summary progressively includes more frames, which consequently also changes the layout, and users are not allowed to interact with

the summary. In the evaluations, the assessors were presented with summaries in a progressive manner, from the coarsest to the finest scale at a fix rate of one scale per second. Three variations were evaluated: the *basic* method, the *anchor* method and the *anchor* method with new panels highlighted. The assessors were asked to sort them according to their preference (from higher to lower preference). Results (see Table III) show a clear preference for the *anchor* method with highlighting, and, in a second place, for the *anchor* method without highlighting. These results confirmed that the *anchor* algorithm can effectively reduce the disturbance, improving the utility of scalable comic-like summaries in this scenario, and also the importance of appropriate interface elements.

Table III
PREFERENCE OF THE ALGORITHMS IN THE PROGRESSIVE SCENARIO: BASIC (B), ANCHOR (A) AND ANCHOR WITH HIGHLIGHT (A+H).

	Trec			*Franc*			*Quest*		
%	1^{st}	2^{nd}	3^{rd}	1^{st}	2^{nd}	3^{rd}	1^{st}	2^{nd}	3^{rd}
B	28.6	14.3	**57.1**	14.3	28.6	**57.1**	21.4	14.3	**64.3**
A	14.3	**71.4**	14.3	14.3	**57.1**	28.6	21.4	**57.1**	21.4
A+H	**57.1**	14.3	28.6	**71.4**	14.3	14.3	**57.1**	28.6	14.3

3) Overall system evaluation: At the end of the evaluation, some general statements were posed to the assessors in order to evaluate the global opinion about the proposed summarisation approach. The criteria and the statements were the following:

- Utility of comic-like summaries ("Comic-like summaries are useful and effective representations of video content.")
- Utility of scalability ("Scalability, i.e. multiple levels of detail, is a useful feature in video summaries.")
- Browsing interface ("The interface provides a useful way to browse summaries of video content.")
- Utility of highlighting ("Highlighting feature is helpful in tracking changes across scales.")
- Overall system ("The proposed system is useful for browsing video content.")

Results of this last part of the subjective evaluation are shown in Figure 7. In general, most of the assessors agreed with these statements, supporting the proposed scalable comic-like summaries as an effective and flexible approach to video summarisation.

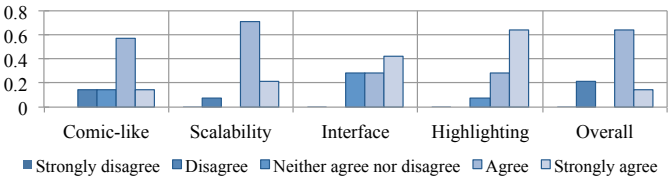

Figure 7. Overall system assessment results.

VII. CONCLUSIONS

In this paper, a novel video summarisation method is proposed, using the notion of scalability in the context of comic-like summaries, which offers a flexible and intuitive abstraction format based on the narrative structure of comics. In contrast to scalable storyboards, the non-trivial visual structure makes comic-like summaries more complex to compute, yet maintaining intuitive comprehension.

In the case of summaries with independent scales, suitable for applications that require the adaptation of the summary to a target length or size, user evaluation shows very good results. On the other hand, in case of progressive summarisation, required for interactive visualisation, the problem of visual disturbance was identified. Induced by the abrupt and complex transitions between scales due to many scattered changes in the layout during a short amount of time, the disturbance hindered usability and the users feet uncomfortable and confused. Based on previous experiments with scalable summaries and initial user feedback, a heuristic algorithm is developed to localise these changes to a limited area. In addition, user tests demonstrated that a carefully designed interface can to minimise disturbing effects and make the proposed abstraction approach appealing and pleasant. Elements which drive user's attention to the main changes are found to be particularly useful (e.g. feature of highlighting the changes in progressive summarisation mode).

The conducted experimental evaluation confirmed the value of scalable comic-like summaries for a wide range of applications in video retrieval and browsing. The proposed solution to alleviate the effects of layout disturbance, as well as improved user interface, has been proven to be helpful in providing more appealing and user-friendly video summaries.

REFERENCES

[1] B. T. Truong and S. Venkatesh, "Video abstraction: A systematic review and classification," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 3, no. 1, p. 3, 2007.
[2] A. G. Money and H. Agius, "Video summarisation: A conceptual framework and survey of the state of the art," *Journal of Visual Communication and Image Representation*, vol. 19, no. 2, pp. 121–143, Feb. 2008.
[3] B. Tseng, C.-Y. Lin, and J. Smith, "Using MPEG-7 and MPEG-21 for personalizing video," *IEEE Multimedia*, vol. 11, no. 1, pp. 42–52, 2004.
[4] P. M. Fonseca and F. Pereira, "Automatic video summarization based on MPEG-7 descriptions," *Signal Processing: Image Communication*, vol. 19, no. 8, pp. 685–699, Sep. 2004.
[5] F. Wang and B. Merialdo, "Multi-document video summarization," in *Proc. of IEEE ICME*, 28 2009-July 3 2009, pp. 1326–1329.
[6] A. Girgensohn, "A fast layout algorithm for visual video summaries," in *Proc. of IEEE ICME*, vol. 2, 2003, pp. II–77–80 vol.2.
[7] J. Calic, D. Gibson, and N. Campbell, "Efficient layout of comic-like video summaries," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 7, pp. 931–936, 2007.
[8] M. Yeung and B.-L. Yeo, "Video visualization for compact presentation and fast browsing of pictorial content," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 5, pp. 771–785, Oct. 1997.
[9] T. Mei, B. Yang, S.-Q. Yang, and X.-S. Hua, "Video collage: presenting a video sequence using a single image," *The Visual Computer*, vol. 25, no. 1, pp. 39–51, Jan. 2009.
[10] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, 2007.
[11] L. Herranz and J. M. Martínez, "An integrated approach to summarization and adaptation using H.264/MPEG-4 SVC," *Signal Processing: Image Communication*, vol. 24, no. 6, pp. 499–509, 2009, scalable Coded Media beyond Compression.
[12] X. Q. Zhu, X. D. Wu, J. P. Fan, A. K. Elmagarmid, and W. F. Aref, "Exploring video content structure for hierarchical summarization," *Multimedia Systems*, vol. 10, no. 2, pp. 98–115, Aug. 2004.
[13] L. Herranz and J. M. Martínez, "A framework for scalable summarization of video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 9, pp. 1265–1270, 2010.
[14] S. Mccloud, *Understanding Comics: The Invisible Art*. HarperCollins, 1994.
[15] M. Albanese, M. Fayzullin, A. Picariello, and V. Subrahmanian, "The priority curve algorithm for video summarization," *Information Systems*, vol. 31, no. 7, pp. 679–695, Nov. 2006.
[16] J. Calic and N. W. Campbell, "Compact visualisation of video summaries," *EURASIP Journal on Adv. in Sig. Proc.*, no. 2, p. 14, 2007.
[17] C. Petersohn, "Fraunhofer HHI at TRECVID 2004: Shot boundary detection system," in *Proc. of TRECVID*, 2004.
[18] W. Kraaij, P. Over, T. Ianeva, and A. Smeaton, "TRECVID 2006 - an overview," in *Proc. of TRECVID*, 2006.
[19] R. Likert, "A technique for the measurement of attitudes." *Archives of Psychology*, vol. 22, no. 140, pp. 1–55, 1932.