VSUMM: An Approach Based on Color Features for Automatic Summarization and a Subjective Evaluation Method[*][†]

Sandra Eliza Fontes de Avila, Arnaldo de Albuquerque Araújo (advisor)
Computer Science Department, Federal University of Minas Gerais – UFMG
31270–010, Belo Horizonte, Minas Gerais, Brazil
{sandra, arnaldo}@dcc.ufmg.br

***Abstract*—The fast evolution of digital video has brought many new multimedia applications and, as a consequence, research into new technologies that aim at improving the effectiveness and efficiency of video acquisition, archiving, cataloging and indexing, as well as increasing the usability of stored videos. Among all possible research areas, *video summarization* is one of the most important topics, which may enable a quick browsing of a large collection of video data and to achieve efficient content access and representation. Essentially, this research area consists of automatically generating a short summary of a video, which can either be a *static summary* or a *dynamic summary*. In this paper, we present VSUMM, a methodology for the development of static video summaries. The method is based on color feature extraction from video frames and unsupervised classification. We also develop a new subjective method to evaluate video static summaries. The video summaries are manually created by users and compared with different approaches found in the literature. Experimental results show – with a confidence level of 98% – that the proposed solution provided static video summaries with superior quality relative to the approaches to which it was compared.**

***Keywords*-Video summarization; Static video summary; Keyframes; Subjective evaluation; Clustering; Color histogram.**

I. Introduction

The recent advances in compression techniques, the decreasing cost of storage and the availability of high-speed connections have facilitated the creation, storage and distribution of videos. This leads to an increase in the amount of video data deployed and used in applications such as search engines and digital libraries, for example. This situation puts not only multimedia data types into evidence, but also leads to the requirement of efficient management of video data. Such requirements paved the way for new research areas, such as *video summarization*.

According to [1], there are two fundamental types of video summaries: *static video summary* – also called *representative frames*, *still-image abstracts* or *static storyboard* – and *dynamic video skimming* – also called *video skim*, *moving-image abstract* or *moving storyboard*. Static video summaries are composed of a set of keyframes extracted from the original video, while dynamic video summaries are composed of a set of shots[2] and are produced taking into account the similarity or domain-specific relationships among all video shots.

One advantage of a video skim over a keyframe set is the ability to include audio and motion elements that potentially enhance both the expressiveness and the amount of information in the summary. In addition, according to [2], it is often more entertaining and interesting to watch a skim than a slide show of keyframes. On the other hand, keyframe sets are not restricted by any timing or synchronization issues and, therefore, they offer much more flexibility in their organization for browsing and navigation purposes, in comparison with the strict sequential display of video skims, as demonstrated in [3], [4], [5], [6], [7]. In this paper, we focus on the production of static video summaries.

Recently, video summarization has attracted considerable interest from researchers and as a result, various algorithms and techniques have been proposed in the literature, most of them based on clustering techniques ([8], [9], [10], [11]). Comprehensive surveys of past video summarization results can be found in [1], [12], [13].

In the case of clustering-based techniques, the basic idea is to produce the summary by clustering together similar frames/shots and then showing a limited number of frames per cluster (usually, one frame per cluster). For such approaches, it is important to select the features upon which the frames can be considered similar (e.g., color distribution, luminance, motion vector) and also to establish different criteria that can be employed to measure the similarity.

Although there are some techniques that produce summaries of acceptable quality, they usually use intricate clustering algorithms that make the summarization process computationally expensive [11]. For example, in [9] the time needed for computing the summaries takes around 10 times

[*] This paper contains extracts of the M.Sc. dissertation named (in portuguese) “Uma Abordagem Baseada em Características de Cor para a Elaboração Automática e Avaliação Subjetiva de Resumos Estáticos de Vídeos” by the first author.

[†] The M.Sc. dissertation is available (in portuguese) at http://www.dcc.ufmg.br/pos/cursos/mestrado_defesas_detalhes.php?aluno=1111

[1] A *keyframe* is a frame that represents the content of a logical unit, as a shot or scene. This content must be the most representative as possible.

[2] A *shot* represents a spatio-temporally coherent frame sequence, which captures a continuous action from a single camera.

the video length. This means that a potential user would wait around 20 minutes to have a concise representation of a video that he/she could have watched in just two minutes.

In this paper, it is proposed a simple and effective approach for automatic video summarization, called *Video SUMMarization* (VSUMM). The method is based on the extraction of color features from video frames and unsupervised classification. In addition, a new subjective methodology to evaluate video summaries is developed, called *Comparison of User Summaries* (CUS). In this methodology, the video summaries are manually created by users and are compared with approaches found in the literature. The evaluation of VSUMM is done on 50 videos from the *Open Video Project* (OV) [14] and experimental results show that the VSUMM approach produces video summaries with superior quality relative to the approaches to which it was compared.

The main contributions of this paper are (1) a mechanism designed to produce static video summaries, which presents the advantages of the main concepts of related work in the video summarization; (2) a new evaluation method of video summaries, which reduces the subjectivity in the evaluation task, quantifies the summary quality and allows comparisons among different techniques quickly; and (3) an experimental research, since our conclusions were derived from statistical techniques, which has demonstrated – with 98% of confidence level – the superiority of the VSUMM approach.

The M.Sc. dissertation was concluded within 19 months. Some of our results were published in IWSSIP’08 (full paper) [15] and SIBGRAPI’08 (full paper) [16]. The last one deserve special attention, because it is the most important national conference in Image Processing area. Besides, our paper was selected as one of the best papers of SIBGRAPI’08 in the areas of image processing, computer vision and pattern recognition; and we were invited to submit an extended version of our paper to be published at the journal Pattern Recognition Letters.

This paper is organized as follows: in Section II, some related work are described; our approach is presented in Section III; the experimental results are discussed in Section IV; finally, some concluding remarks and future lines of investigation are derived in Section V.

II. Related Work

Different approaches have been studied in order to elaborate our solution. Some of the main ideas that are related to the proposed solution are discussed next.

Zhuang et al. [17] proposed a method for keyframe extraction based on unsupervised clustering. In that work, the video is segmented into shots and then a color histogram (in the HSV color space) is computed from every frame. The clustering algorithm uses a threshold δ which controls the clustering density. Before a new frame is classified as pertaining to a certain cluster, the similarity between this node and the centroid of the cluster is computed first. If this value is less than δ, it means that this node is not close enough to be added into the cluster. The keyframes selection is employed only to the clusters which are big enough to be considered as keyclusters. In such case, a representative frame is extracted from this cluster as the keyframe. A keycluster is considered large enough if it is larger than the average cluster size. For each keycluster, the frame which is closest to the keycluster centroid is selected as the keyframe. According to [17], the proposed technique is efficient and effective, however, no comparative evaluation is performed for validating such assertions.

Hanjalic and Zhang [18] presented a method for automatically producing a summary of an arbitrary video sequence. The method is based on cluster-validity analysis and is designed to work without any human supervision. The entire video material is first grouped into clusters. Each frame is represented by color histograms in the YUV color space. A partitional clustering is applied n times to all frames of a video sequence. The pre-specified number of clusters starts at one and is increased by one each time the clustering is applied. Next, the system automatically finds the optimal combination(s) of clusters by applying the cluster-validity analysis. After the optimal number of clusters is found, each cluster is represented by one characteristic frame, which then becomes a new keyframe for that video sequence. [18] concentrated on the evaluation of the proposed procedure for cluster-validity analysis, instead of on evaluating the produced summaries.

Gong and Liu [19] proposed a technique for video summarization based on the Singular Value Decomposition (SVD). At first, a set of frames in the input video is selected (one from every ten frames) and then, color histograms in the RGB color space are used to represent video frames. To incorporate spatial information, each frame is divided into 3×3 blocks, and a 3D-histogram is created for each of the blocks. These nine histograms are then concatenated together to form a feature vector. Using this feature vector extracted from the frames, a feature-frame matrix A (usually sparse) is created for the video sequence. Therefore, the SVD is performed on A to obtain the matrix V, in which each column vector represents one frame in the refined feature space. Next, the cluster closest to the origin of the refined feature space is found, the content value of this cluster is computed and this value is used as the threshold for clustering the remaining frames. From each cluster, the system selects the frame that is closest to the cluster center as keyframe. This method is not compared with other techniques.

Mundur et al. [9] developed a method based on Delaunay Triangulation (DT), which is applied for clustering the video frames. The first step of their method is to obtain the video frames from the original video, pre-sampling the video frames. Each frame is represented by a color histogram in the HSV color space. This histogram is represented as a row vector and the vectors for each frame are concatenated into a matrix. To reduce the dimensions of this matrix, the Principal Components Analysis (PCA) is applied. After that, the Delaunay diagram is built. The clusters are obtained by separating edges in the Delaunay diagram. Finally, for each cluster, the frame that is nearest to its center is selected as the keyframe. To evaluate the summaries, [9] defined three objective metrics: significance factor, overlap factor and compression factor. In spite of the fact that the proposed method has been designed to be fully automatic (i.e., with no user-specified parameters and well suited for batch processing), it requires between 9 to 10 times the video length to produce the summary. Furthermore, the method does not preserve the video temporal order.

Furini et al. [10] introduced VISTO (VIdeo STOry-boards), a summarization technique designed to produce on-the-fly video storyboards. VISTO is composed of three phases. First, the video is analyzed in order to extract the HSV color description. For each input frame, a 256-dimension vector is extracted. Those vectors are then stored in a matrix and then, in the second phase, the clustering algorithm is applied to extracted data. The authors exploited the triangular inequality in order to filter out useless distance computations. To obtain the number of clusters, the pairwise distance of consecutive frames is computed. If the distance is greater than the threshold Γ, the number of clusters is incremented. The third and last phase aims at removing meaningless video frames from the produced summary. VISTO is evaluated through a comparison study with other approaches: the DT technique [9] and the Open Video storyboards [14]. [10] asked a group of 20 people to evaluate the produced summaries, using the following procedure: the video is presented to the user, and just after that, the corresponding summary is also shown. The users are asked whether the summary is a good representation of the original video. The quality of the video summary is scored on a scale going from 1 (bad) to 5 (excellent), and the mean opinion score is considered as an indication of the summary quality.

Guironnet et al. [20] proposed a method for video summarization based on camera motion. It consists in selecting frames according to the succession and the magnitude of camera motions. The method is based on rules to avoid temporal redundancy among the selected frames. The authors developed a subjective method to evaluate the proposed summary. In their experiments, 12 subjects are asked to watch a video and to create a summary manually. From the summaries of different subjects, an "optimal" one is built automatically. This "optimal" summary is then compared with the summaries obtained by different methods. The construction of an "optimal" summary is a difficult stage, which requires various parameters to be fixed.

According to the analysis of the approaches found in literature, it can be noticed that the keyframe selection techniques used several visual features and statistics. These features can affect both the computational complexity and the summary quality. Normally, the extraction of the video features may produce a high dimensional matrix. For this reason, dimensionality reduction techniques are used in order to reduce the size of those matrices, as it can be seen in [9], [19], for example. Needless to say, this additional step requires even more processing time. Another serious problem that can be observed is the lack of trustworthy comparisons among existing techniques. In other words, a consistent evaluation framework is seriously missing in video summarization research.

The VSUMM approach, proposed in present work, draws on the advantages of the existing techniques and concepts presented in related work. A fully reproducible evaluation framework is proposed and applied for comparisons among VSUMM and three other proposals, indicating that VSUMM is able to provide better summaries, according to the defined metrics.

III. VSUMM Approach

Figure 1 illustrates the steps of our method to produce static video summaries. Initially, the original video is split into frames (step 1). In next step (step 2), color features are extracted to form a color histogram in HSV color space. VSUMM does not consider all the video frames, but takes a sample. In addition, the meaningless frames found in the sample are removed. After that (step 3), the frames are grouped by k-means clustering algorithm. Then (step 4), one frame per cluster is selected (this selected frame is the keyframe). To refine the static video summary composed by the keyframes (step 5), the keyframes that are too similar are eliminated. Finally, the remaining keyframes are arranged in the original temporal order to facilitate the visual comprehension of result. Each step is detailed in next subsections.

A. Temporal Video Segmentation

Temporal video segmentation is the first step towards automatic video summarization. Its goal is to divide the video stream into a set of meaningful and manageable basic elements (e.g., shots, frames) [21]. In literature, the *shot boundary detection* [22] is widely used as first step to produce summaries (e.g., [8], [17], [18], [23], [24], [25], [26], [27]).

Another type of video segmentation is the *extraction of video frames*, where there is no temporal analysis of the video. Each frame is treated separately, the video sequence is split into images. Several authors have used this approach (e.g., [4], [9], [10], [19], [28]), and it is also used in this work. Moreover, VSUMM does not consider all the video frames, but takes only a subset taken at a pre-determined sampling rate. This is the so-called *pre-sampling* approach.

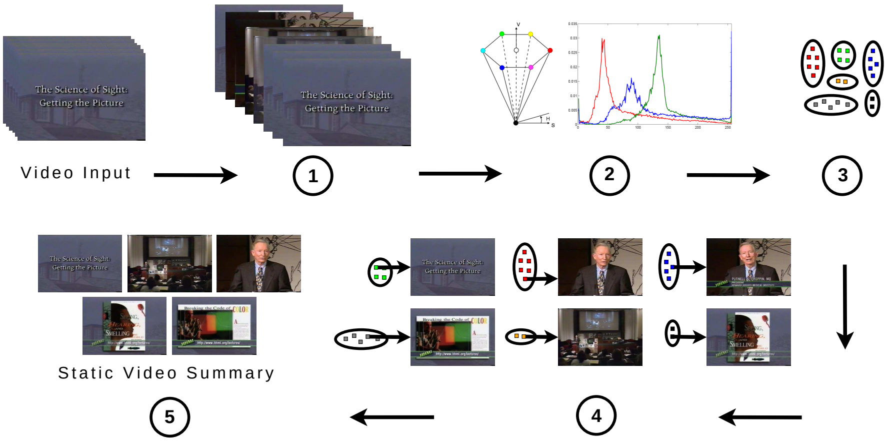


Figure 1: VSUMM approach.

By using a sampling rate, the number of video frames to be analyzed are reduced. The sampling rate assumes a fundamental importance, since the smaller the sampling rate, the shorter the video summarization time. Nevertheless, very low sampling rates can lead to poor quality summaries. Videos that have long shots tend to present an advantage with the pre-sampling approach, on the other hand, in those videos that present shorter shots, important parts of its content may not be represented. The relationship between the *loss of information* and the *shot size* is directly associated with the sample rates selected during the summarization process.

In VSUMM, the sampling rate is obtained by dividing the number of video frames by the video frame rate. For instance, for a two-minute-long video with a frame rate of 30 frames per second (i.e., 3600 frames), the total number of frames to be extracted is given by 120 $(3600/30)$ frames.

B. Color Feature Extraction

Color is perhaps the most expressive of all the visual features [29]. In VSUMM, color histogram [30] is applied to describe the visual content of video frames. This technique is computationally trivial to compute and is also robust to small changes of the camera position. Furthermore, color histograms tend to be unique for distinct objects. For these reasons, this technique is widely used in automatic video summarization ([9], [10], [17], [18], [19]).

Some key issues of histogram-based techniques are the selection of an appropriate color space and the quantization of that color space. In VSUMM, the color histogram algorithm is applied to the HSV color space, which is a popular choice for manipulating color. The HSV color space was developed to provide an intuitive representation of color and to be near to the way in which humans perceive and manipulate color. The VSUMM color histogram is computed only from the Hue component, which represents the dominant spectral component color in its pure form [31]. Moreover, the quantization of the color histogram is set to 16 color bins, aiming at reducing significantly the amount of data without loosing important information. The color bins value was established through experimental tests (see [16]).

C. Elimination of Meaningless Frames

A *meaningless frame* is a monochromatic frame due to fade-in/fade-out effects. To remove possible meaningless frames, VSUMM computes the standard deviation of the frame feature vector. As the standard deviation of monochromatic frames is equal to zero or a sufficiently small value close to zero[3], VSUMM just removes these frames.

This step is also employed by [10]. Unlike VSUMM, which removes meaningless frames as a pre-processing step, [10] apply it as a post-processing step, after an initial summary is produced. Nevertheless, there is no point in using meaningless frames in the clustering step and, hence, the removal of such frames is performed before clustering in VSUMM.

D. Clustering

The k-means clustering algorithm [32] is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem [33]. In this work, the k-means algorithm is applied to cluster similar frames, although slightly modified in how it initially distributes the video frames among the k clusters. This modification is applied to improve k-means performance while producing more effective results.

The frames are initially grouped in sequential order, instead of randomly as in the original k-means algorithm. As an example, suppose $k = 5$ and a set of 50 frames sampled from a video. In the original k-means, the frames would be initially allocated randomly among the 5 clusters in

[3] There are frames that are not completely homogeneous in color, but can be regarded as meaningless frames.

order to start their iterative refinement. In case of VSUMM, the initial allocation is going to be done by associating the first 10 frames to the first cluster, the next 10 frames to the second one, and so on. This procedure is adopted based on the fact that consecutive frames typically show some similarity among them already, making it faster for k-means to converge.

One drawback of the k-means clustering algorithm is that it demands the number of clusters k to be fixed *a priori*. Nevertheless, k is related to the summary size, which is going to depend both on video length and on its dynamics. This means that different videos require different values for k. To overcome this difficulty imposed by k-means, a fast procedure to make a reasonable estimate of the number of clusters is implemented. VSUMM computes the pairwise distance of consecutive frames in the extracted sample, according to Euclidean distance. Then, the value selected for k is based on a threshold τ, which measures the sufficient content change in the video sequence. Every time the distance between two consecutive frames is greater than τ, then k is incremented. The threshold value applied in this work, established through experimental tests, is equal to 0.5.

Figure 2 shows an example of how these distances are distributed along time. It is observed that there are points in time in which the distance between consecutive frames varies considerably (corresponding to peaks), whereas there are longer periods in which the variation is very small (corresponding to denser regions). Usually, peaks correspond to a sudden change in the video, while in dense regions frames are more similar to one another. Hence, frames between two peaks can be considered as a set of similar frames and therefore, the number of peaks provides a reasonable estimation to the number of clusters k.

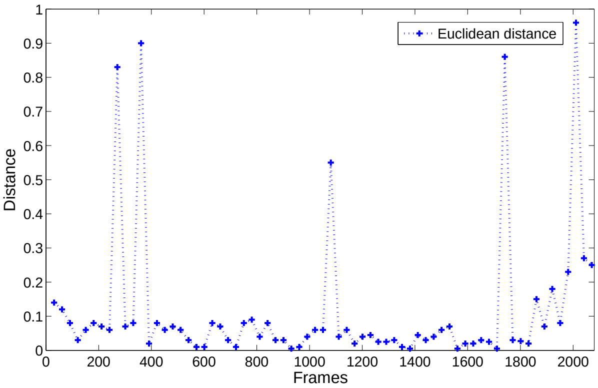


Figure 2: Pairwise distances of sampled frames of the video *The Great Web of Water, Segment 02* (available at [14]).

It is worth noticing that our method for the estimation of the number of clusters is based on a simple shot boundary detection method [34], whereas k is incremented for each sufficient content change in the video sequence.

E. Keyframe Extraction

Once the clusters are formed by k-means, they can be further analyzed for keycluster selection. The strategy applied for keyclusters selection is similar to the one proposed in [17], which is also applied in [27]. In VSUMM, a cluster is considered a *keycluster* if its size is larger than half the average cluster size (this value has shown to be more suitable as cut-off point than the average cluster size, as defined in [17]). For each keycluster, the frame which is closest to the keycluster centroid – measured by Euclidean distance – is selected as a keyframe. In the experiments described in Section IV, VSUMM$_1$ produces the summaries without performing keycluster selection and VSUMM$_2$ uses keycluster selection to produce its summaries.

F. Elimination of Similar Keyframes

The goal of this step is to avoid that keyframes too similar appear in the produced summaries. For this purpose, the keyframes are compared among themselves through color histogram. The similarity is based on a threshold τ, the same used to estimate the number of clusters. If the measured similarity is lower than τ, then the keyframe is removed from the summary.

In Figure 3, it is possible to see an example of similar keyframes ($\tau < 0.5$) and non-similar keyframes ($\tau \geq 0.5$). It is interesting to notice that the frames do not need to be identical to be considered too similar.

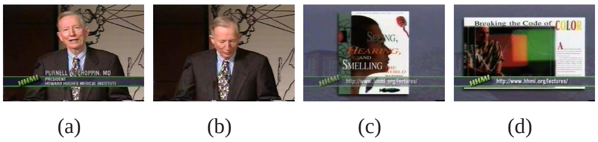

(a) (b) (c) (d)

Figure 3: Similar keyframes (a–b) and non-similar keyframes (c–d) of the video *Senses And Sensitivity, Introduction to Lecture 2* (available at [14]).

Finally, the remaining keyframes are arranged in temporal order to make the produced summary easier to understand.

G. Evaluation of Video Summary

In knowledge area, to advance effectiveness and/or efficiency of new solutions to a particular problem, these need to be evaluated, preferably against existing ones. However, a consistent evaluation framework is seriously missing for video summarization research. Presently, every work has its own evaluation methodology, often presented without any performance comparison with previously existing techniques. To some extent, this happens because, unlike other research areas, such as object detection and recognition, evaluating the correctness of a video summary is not a straightforward task, due to the lack of an objective ground-truth.

In this work, it is proposed a subjective evaluation method to evaluate video summaries. In this evaluation method, called *Comparison of User Summaries* (CUS), the video summary is built manually by a number of users from the sampled frames. The users summaries are then compared with the summaries obtained by different methods. In this way, the user summaries are the reference summaries, i.e., the ground-truth. The goals of this method are: (1) to reduce the subjectivity of the evaluation task; (2) to quantify the summary quality and; (3) to allow comparisons among different techniques to be done quickly.

CUS evaluation method is similar to that in [20]. In that evaluation method, an "optimal" summary is automatically built from user summaries. These summaries are then compared with the results of their summarization technique. Unlike that evaluation method, CUS compares each user summary directly with the automatic summaries, thus keeping the original user opinion. For comparing keyframes from different summaries, the same color histograms used in Section III-B are applied, whereas the distance among them is measured by Manhattan distance. Two keyframes are similar if the distance between them is less than a predetermined threshold δ. Once two frames are matched, they are removed from the next iteration of the comparing procedure. The threshold value applied, established through experimental tests, is equal to 0.5.

Figure 4 illustrates our evaluation method. Firstly, the users are asked to watch the video and then manually create a summary for it. For the users to produce their summaries, the sampled frames are displayed to them (step 1). They are oriented to select a set of frames that, in their opinion, is able to summarize the original video content. The users are free to select any number of frames to compose their summaries. Next (step 2), the user summaries are compared with the automatically generated summary. The quality of the automatically generated summary is assessed (step 3) by two metrics, called accuracy rate CUS_A and error rate CUS_E, which are defined as follows:

$$CUS_A = \frac{n_{mAS}}{n_{US}}, \tag{1}$$

$$CUS_E = \frac{n_{\overline{m}AS}}{n_{US}}, \tag{2}$$

where n_{mAS} is the number of matching keyframes from automatic summary (AS), $n_{\overline{m}AS}$ is the number of non-matching keyframes from AS and n_{US} is the number of keyframes from user summary (US).

The CUS_A values range from 0 (the worst case, when none of the keyframes from AS match with the keyframes from US, or vice-versa) to 1 (the best case, when all the keyframes from US match with the keyframes from AS). It is important to notice that $CUS_A = 1$ does not necessarily mean that all the keyframes from AS and US are matched. That is, if $n_{US} < n_{AS}$ (n_{AS} is the number of keyframes from AS) and $CUS_A = 1$, then some keyframes from AS did not match.

To CUS_E, the values range from to n_{AS}/n_{US} (the worst case, when none of the keyframes from AS match with the keyframes from US, or vice-versa) to 0 (the best case, when all the keyframes from AS match with the keyframes from US).

This means that the CUS_A and CUS_E metrics are complementary, the highest summary quality being when $CUS_A = 1$ and $CUS_E = 0$, meaning that all keyframes from AS and US are matched.

IV. Experimental Results

The experiments are performed into two parts: (1) preliminary experiments, aimed at analyzing the VSUMM parameters that have the strongest impact on results and to identify possible problems; and (2) refined experiments, aimed at improving those previous results. The preliminary results are published in [15], [16]. In this paper, only the refined results are presented.

The experiments were conducted on 50 videos selected from the *Open Video Project* [14]. All videos are in MPEG-1 format (30 fps, 352 $\times$ 240 pixels), in color and with sound. The selected videos are distributed among several genres (documentary, educational, ephemeral, historical, lecture) and their duration varies from 1 to 4 minutes. These 50 videos are the same used by [9] and [10] and were chosen to make possible a comparative evaluation.

The user summaries were created by 50 users, each one dealing with 5 videos, meaning that each video has 5 video summaries created by 5 different users. In other words, 250 video summaries were created manually.

As stated earlier, two slightly different approaches were applied to produce the automatic summaries: VSUMM$_1$ and VSUMM$_2$. The only difference between them is that in VSUMM$_1$, one keyframe is selected per cluster, and in VSUMM$_2$, one keyframe is selected per keycluster. These approaches were compared with two other approaches found in the literature for automatic summarization – DT [9] and VISTO [10]. Additionally, the summaries produced by VSUMM$_1$ and VSUMM$_2$ were compared with the OV summaries, which are generated using the algorithm from [35] added to some manual intervention to refine the produced summaries. All static video summaries for the aforesaid approaches (OV, DT, VISTO, VSUMM$_1$, VSUMM$_2$) can be seen at http://www.npdi.dcc.ufmg.br/VSUMM09.

The summaries quality is evaluated by the accuracy rate CUS_A (Equation 1) and error rate CUS_E (Equation 2). The results are shown in Table I.

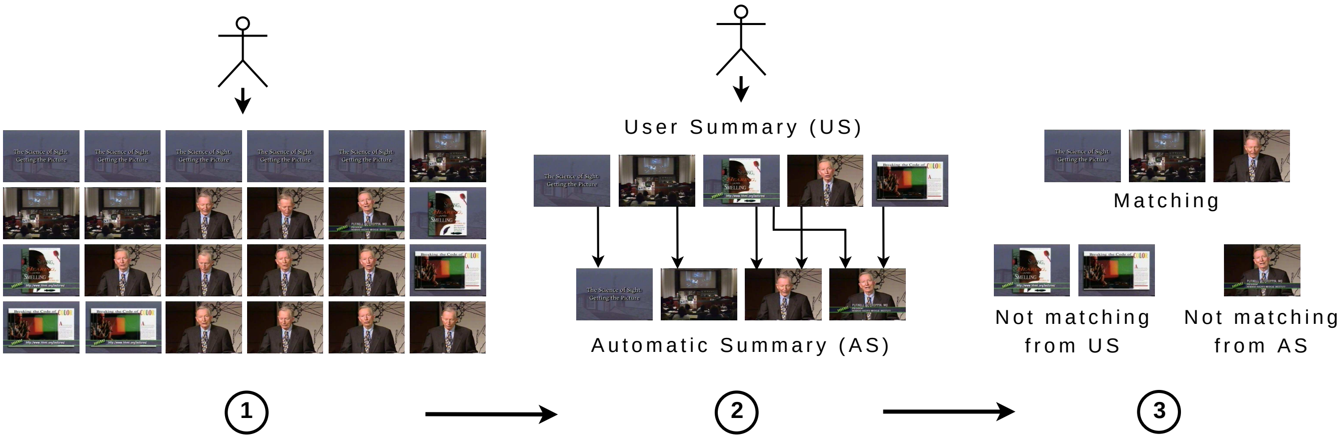


Figure 4: CUS evaluation method.

Table I: Mean accuracy rate CUS_A and mean error rate CUS_E achieved by different approaches.

	OV	DT	VISTO	VSUMM$_1$	VSUMM$_2$
CUS_A	0.70	0.53	0.72	**0.85**	0.70
CUS_E	0.57	0.29	0.58	0.38	**0.27**

The results indicated that VSUMM$_1$ achieved the highest accuracy rate and VSUMM$_2$ achieved the lowest error rate. To verify the statistical significance of these results, the confidence intervals for the differences between paired means were computed to compare every pair of approaches. If the confidence interval includes zero, the difference is not significant at that confidence level. If the confidence interval does not include zero, then the sign of the mean difference indicates which alternative is better [36].

Tables II and III show the results of such comparisons between VSUMM$_1$ and the other approaches considered. These tables show the accuracy rates and the error rates, respectively.

Table II: Difference between mean accuracy rates CUS_A at a confidence of 98%.

Difference	Confidence Interval (98%) min.	 max.
VSUMM$_1$ – OV	0.08	0.22
VSUMM$_1$ – DT	0.26	0.38
VSUMM$_1$ – VISTO	0.07	0.20
VSUMM$_1$ – VSUMM$_2$	0.11	0.18

Since the confidence intervals – with a confidence of 98% – do not include zero in any case, the results presented in Tables II and III confirm that VSUMM$_1$ approach provides results with superior quality (highest accuracy rate) relative to the approaches to which it was compared. In addition, it is possible to say that the VSUMM$_1$ summaries are closer to the summaries created by users.

Table III: Difference between mean error rates CUS_E at a confidence of 98%.

Difference	Confidence Interval (98%) min.	 max.
VSUMM$_1$ – OV	–0.38	–0.01
VSUMM$_1$ – DT	0.01	0.17
VSUMM$_1$ – VISTO	–0.32	–0.09
VSUMM$_1$ – VSUMM$_2$	0.07	0.15

Moreover, also with 98% confidence, the results confirm that VSUMM$_1$ approach presents a lower error rate than OV and VISTO approaches. However, VSUMM$_1$ presents a higher error rate than DT and VSUMM$_2$ approaches.

In DT approach, this "positive" result was expected because the DT approach produces much smaller summaries than the summaries created by users. Consequently, the DT summaries present a low error rate at a cost of a low accuracy rate. So, this result can be disregarded, since the more interesting summaries are those that present low error rate and, at the same time, high accuracy rate.

In the case of VSUMM$_2$ approach, the analysis is similar to the DT approach. The VSUMM$_2$ summaries show at most the same size of the VSUMM$_1$ summaries, but eventually smaller, since some clusters are disregarded in the keycluster refinement step. As VSUMM$_2$ produces smaller summaries, it tends to miss less, but also to hit less frames, as can be seen in Table I, where the accuracy rate achieved by VSUMM$_2$ approach is significantly smaller than the accuracy rate achieved by VSUMM$_1$ approach.

Considering these observations, it is possible to conclude that VSUMM$_1$ approach provides better results relative to the approaches to which it was compared. Nevertheless, for applications which require lower error rate, the VSUMM$_2$ approach can be a better choice.

Figure 5 shows the video summaries produced by all different approaches considered for comparison (OV, DT, VISTO, VSUMM$_1$, VSUMM$_2$). The video under consideration is *Senses And Sensitivity, Introduction to Lecture 2* and Figure 6 displays the user summaries. As the CUS values reported, the OV and DT approaches exhibit identical low rates. Furthermore, it is possible to note that the VISTO summary contains keyframes that are very similar to each other, while VSUMM$_1$ provides a more concise summary for the video. Although VSUMM$_2$ achieves the lowest possible error rate ($CUS_E = 0$), the accuracy rate is also low. The highest summary quality ($CUS_A = 1$ and $CUS_E = 0$) is achieved by VSUMM$_1$ approach, which can be confirmed by a visual comparison with the user summaries that can be seen in Figure 6.

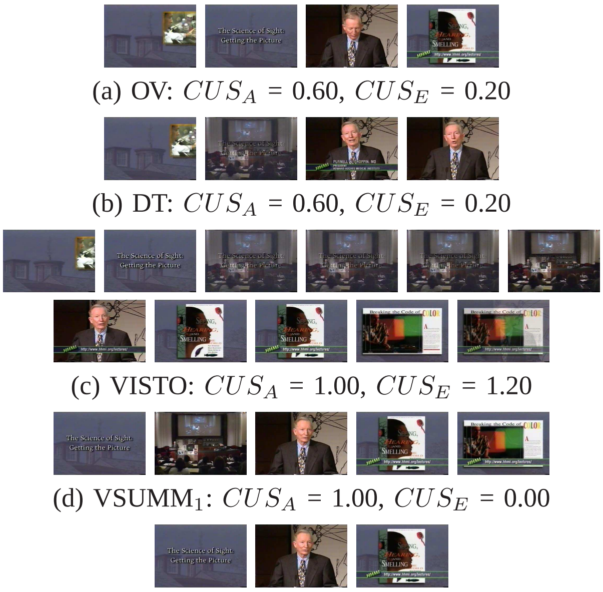
A. Discussion

We call the attention to the accuracy rates of VSUMM$_1$ and VSUMM$_2$ approaches. On the contrary to what could be expected at first sight, the VSUMM$_1$ approach provided results with superior quality relative to the VSUMM$_2$ approach. Since VSUMM$_2$ selects the keyframes from the keyclusters, eliminating the clusters that, in theory, would not be too important – because they are composed of a small number of frames –, then it was expected that the accuracy rate achieved by it would be higher than the accuracy rate achieved by VSUMM$_1$ approach.

Another point to be observed is the high number of keyframes of the video summaries created by users. Before performing the evaluation process, it was informed to the users that they should select the frames which, in their opinion, could represent the original video content in a concise way. Thus, it was expected to obtain user summaries consisting only of the most relevant frames (keyframes). Nevertheless, the user summaries showed the users preferred to create more extensive summaries that represent all the various video segments, regardless of the segment size.

V. Conclusions

Automatic video summarization has been receiving growing attention from the scientific community. This attention can be explained by several factors, for instance, (1) the advances in the computing and network infrastructure, (2) the growth of the number of videos published on the Internet, (3) scientific challenges, (4) practical applications as search engines and digital libraries, (5) inappropriate use of traditional video summarization techniques to describe, represent and perform search in large video collections. As examples,


(a) OV: CUS_A = 0.60, CUS_E = 0.20

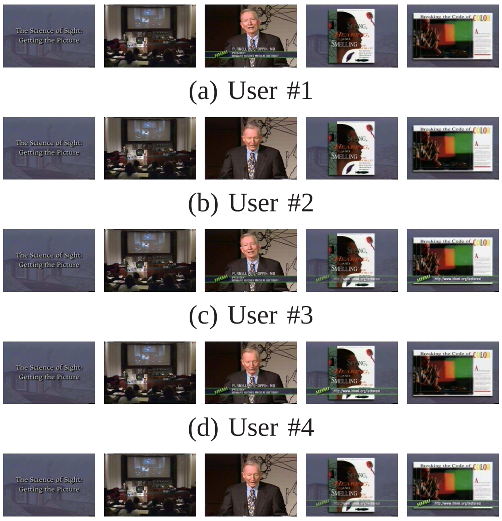
(b) DT: CUS_A = 0.60, CUS_E = 0.20

(c) VISTO: CUS_A = 1.00, CUS_E = 1.20

(d) VSUMM$_1$: CUS_A = 1.00, CUS_E = 0.00

(e) VSUMM$_2$: CUS_A = 0.60, CUS_E = 0.00

Figure 5: Video summaries of different approaches of the video *Senses And Sensitivity, Introduction to Lecture 2* (available at [14]).


(a) User #1

(b) User #2

(c) User #3

(d) User #4

(e) User #5

Figure 6: User summaries of the video *Senses And Sensitivity, Introduction to Lecture 2* (available at [14]).

the video search engines as Alta Vista[4], Google[5], and Yahoo[6] usually represent entire videos by a single keyframe.

In this paper, we presented VSUMM, a mechanism designed to produce static video summaries. It presents the advantages of the concepts of related work in the video summarization; on a single method, VSUMM includes the main contributions of previously proposed techniques. We also developed a new subjective method to evaluate the proposed summary, which (1) reduces the subjectivity of evaluation task, (2) quantifies the summary quality and (3) allows comparisons among different techniques quickly.

One of the future lines of investigation will be to test VSUMM on different genres of videos, such as cartoons, sports, tv-shows; and test VSUMM on long videos. In addition, other features will be investigated, for example, motion, shape, texture. The fusion of different features is also an interesting future direction. Furthermore, techniques to estimate the number of clusters will be exploited, for instance, Akaike's Information Criterion (AIC) [37] or Minimum Description Length (MDL) [38]. Moreover, other clustering algorithms will be investigated, for example, DBSCAN [39], a density-based clustering method. Finally, VSUMM can be extended to produce video skims. It can be created from keyframes by joining fixed-size segments, subshots, or the whole shots that enclose them, as employed in [18].

ACKNOWLEDGMENTS

The authors are grateful to CNPq, CAPES and FAPEMIG, Brazilian research funding agencies, for the financial support to this work.

The authors are also grateful to Ana Paula Brandão Lopes and Antonio da Luz Junior for the valuable suggestions and comments.

REFERENCES

[1] B. T. Truong and S. Venkatesh, "Video abstraction: A systematic review and classification," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 3, no. 1, 2007.

[2] Y. Li, T. Zhang, and D. Tretter, "An overview of video abstraction techniques," HP Laboratory, HP-2001-191, Tech. Rep., July 2001.

[3] M. M. Yeung and B.-L. Leo, "Video visualization for compact representation and fast browsing of pictorial content," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 5, pp. 771–785, 1997.

[4] S. Uchihashi, J. Foote, A. Girgensohn, and J. S. Boreczky, "Video manga: generating semantically meaningful video summaries," in *Proceedings of the ACM International Conference on Multimedia (Part 1)*, New York, NY, USA, 1999, pp. 383–392.

[5] A. Girgensohn, "A fast layout algorithm for visual video summaries," in *Proceedings of the International Conference on Multimedia and Expo (ICME)*. Washington, DC, USA: IEEE Computer Society, 2003, pp. 77–80.

[6] J. Ćalić, D. P. Gibson, and N. W. Campbell, "Efficient layout of comic-like video summaries," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 7, pp. 931–936, 2007.

[7] T. Wang, T. Mei, X.-S. Hua, X.-L. Liu, and H.-Q. Zhou, "Video collage: A novel presentation of video sequence," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2007, pp. 1479–1482.

[8] Y. Hadi, F. Essannouni, and R. O. H. Thami, "Video summarization by k-medoid clustering," in *Proceedings of the ACM Symposium on Applied Computing (SAC)*, New York, NY, USA, 2006, pp. 1400–1401.

[9] P. Mundur, Y. Rao, and Y. Yesha, "Keyframe-based video summarization using Delaunay clustering," *International Journal on Digital Libraries*, vol. 6, no. 2, pp. 219–232, 2006.

[10] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini, "VISTO: visual storyboard for web video browsing," in *Proceedings of the ACM International Conference on Image and Video Retrieval (CIVR)*, 2007, pp. 635–642.

[11] ——, "On using clustering algorithms to produce video abstracts for the web scenario," in *Proceedings of the IEEE Consumer Communication and Networking (CCNC)*. IEEE Communication Society, January 2008, pp. 1112–1116.

[12] Y. Li, S.-H. Lee, C.-H. Yeh, and C.-C. Kuo, "Techniques for movie content analysis and skimming: tutorial and overview on video abstraction techniques," *Signal Processing Magazine, IEEE*, vol. 23, no. 2, pp. 79–89, March 2006.

[13] 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 (JVCIR)*, vol. 19, no. 2, pp. 121–143, February 2008.

[14] The Open Video Project. http://www.open-video.org.

[15] S. E. F. de Avila, A. da Luz Jr., and A. de A. Araújo, "VSUMM: A simple and efficient approach for automatic video summarization," in *15th International Conference on Systems, Signals and Image Processing (IWSSIP)*, Bratislava, Slovakia, June 2008, pp. 449–452.

[16] S. E. F. de Avila, A. da Luz Jr., A. de A. Araújo, and M. Cord, "VSUMM: An approach for automatic video summarization and quantitative evaluation," in *Proceedings of the XXI Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, 2008, pp. 103–110.

[17] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, vol. 1, 1998, pp. 866–870.

[4] http://video.altavista.com

[5] http://video.google.com

[6] http://video.search.yahoo.com

[18] A. Hanjalic and H. Zhang, "An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 8, pp. 1280–1289, 1999.

[19] Y. Gong and X. Liu, "Video summarization using singular value decomposition," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, 2000, pp. 2174–2180.

[20] M. Guironnet, D. Pellerin, N. Guyader, and P. Ladret, "Video summarization based on camera motion and a subjective evaluation method," *EURASIP Journal on Image and Video Processing*, pp. Article ID 60 245, 12 pages, April 2007.

[21] I. Koprinska and S. Carrato, "Temporal video segmentation: a survey," *Signal Processing: Image Communication*, vol. 16, no. 5, pp. 477–500, 2001.

[22] C. Cotsaces, N. Nikolaidis, and L. Pitas, "Video shot detection and condensed representation: A review," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 28–37, 2006.

[23] S. V. Porter, M. Mirmehdi, and B. T. Thomas, "A shortest path representation for video summarisation," in *Proceedings of the IEEE International Conference on Image Analysis and Processing (ICIAP)*, 2003, pp. 460–465.

[24] J. Rong, W. Jin, and L. Wu, "Key frame extraction using inter-shot information," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2004, pp. 571–574.

[25] Z. Li, G. M. Schuster, and A. K. Katsaggelos, "Minmax optimal video summarization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 10, pp. 1245–1256, 2005.

[26] I.-C. Chang and K.-Y. Chen, "Content-selection based video summarization," *Digest of Technical Papers International Conference on Consumer Electronics (ICCE)*, pp. 1–2, 2007.

[27] G. Cámara-Chávez, F. Precioso, M. Cord, S. Phillip-Foliguet, and A. de A. Araújo, "An interactive video content-based retrieval system," in *15th International Conference on Systems, Signals and Image Processing (IWSSIP)*, Bratislava, Slovakia, June 2008, pp. 133–136.

[28] I. Yahiaoui, B. Mérialdo, and B. Huet, "Automatic video summarization," in *Multimedia Content-Based Indexing and Retrieval (MCBIR)*, 2001.

[29] A. Trémeau, S. Tominaga, and K. N. Plataniotis, "Color in image and video processing: most recent trends and future research directions," *EURASIP Journal on Image and Video Processing*, vol. 2008, no. 3, pp. 1–26, 2008.

[30] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, November 1991.

[31] B. S. Manjunath, J. R. Ohm, V. V. Vinod, , and A. Yamada, "Color and texture descriptors," *IEEE Transactions Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 703–715, June 2001.

[32] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of The Berkeley Symposium on Mathematical Statistics and Probability*, L. M. L. Cam and J. Neyman, Eds., vol. 1. University of California Press, 1967, pp. 281–297.

[33] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Springer-Verlag New York, Inc., 2001, ch. Unsupervised Learning and Clustering, p. 654.

[34] S. J. F. Guimaraes, M. Couprie, A. de Albuquerque Araújo, and N. J. Leite, "Video segmentation based on 2D image analysis," *Pattern Recognition Letters*, vol. 24, no. 7, pp. 947–957, 2003.

[35] D. DeMenthon, V. Kobla, and D. Doermann, "Video summarization by curve simplification," in *Proceedings of the ACM International Conference on Multimedia*, NY, USA, 1998, pp. 211–218.

[36] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. John Wiley and Sons, Inc., 1992.

[37] H. Akaike, "A new look at statistical model identification," *IEEE Transactions on Automatic Control*, vol. 19, pp. 716–723, 1974.

[38] J. Rissanen, "Modelling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.

[39] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of International Conference on Knowledge Discovery and Data Mining (KDD)*, 1996, pp. 226–231.