Video Summary Generation and Coding Using Temporal Scalability

Lino Ferreira[1], Luís Cruz[2] and Pedro A. Amado Assunção[3]

[1,2,3]Instituto de Telecomunicações, Coimbra, Portugal
[1,3]Instituto Politécnico de Leiria / ESTG, Leiria, Portugal
[2]Universidade de Coimbra / DEEC, Coimbra, Portugal
{[1]lino,[3]assuncao}@estg.ipleiria.pt, [2]lcruz@deec.uc.pt

Abstract — **In this paper two algorithms for video summary generation and coding are proposed. Two distortion metrics used in the video summary generation algorithm are compared and an algorithm with reduced computational complexity is presented. The paper also proposes two frame structures in the temporal domain suitable for coding using temporal scalability of the H.264/SVC.**

Keywords: SVC, Temporal Scalability, Video Summarization

I. Introduction

It is expected that in the near future, data networks will provide high-quality multimedia communication services in which the content quality is adapted to the processing/power of the terminal, the network conditions, and the user's preferences. In particular, in video communications, scalable coding is considered an important functional technology of video coding as it enables this type of adaptability to constrained terminals and network. Although it might not be suited to all communication scenarios, it proves to be useful in many video networking applications. For example, a method to cope with diverse user requirements in video communication is to use streams with spatial and temporal scalability, which allows signal resolution adaptation. The video coding standards, H.262|MPEG-2, H.263 and MPEG-4 Visual support temporal scalability, with particular relevance to H.264/AVC whose temporal scalability exhibits increased flexibility because of its reference picture memory control. For supporting temporal scalability with reasonable number of temporal layers, no changes to the design of H.264/AVC were required.[1].

The increasing availability of video and audio in personal computers, PDA and mobile phones creates a strong demand for short versions of the coded data either in spatial or temporal domain. Such short versions are useful to rapidly provide some information about the content of a long video or set of videos to users. Video summarization automatically creates a short version, or subset of key frames, which contains as much information as possible of the original video. From a video summary, the user should be able to evaluate if a video is interesting or not. For example, if a documentary contains a certain topic of his/her interest. Video summarization introduces distortions at the playback stage and this distortion is related to the conciseness of the summary whereby a more succinct summary implies higher distortion [2-5].

The regular scheme used in H.264/SVC to achieve temporal scalability is not directly applicable to the temporal scalable coding of video summaries as these are obtained by a non-uniform temporal sampling of the full time-resolution sequence. In this work we propose two methods of coding video summaries, using a modified scheme of the H.264/SVC temporal scalability. With these methods we can have a scalable bitstream with several temporal layers including the video summary of the original sequence. Using these methods the viewer can extract the video summary from the bitstream without having to decode the higher temporal layers and then quickly browse through the video. If the user wants to see the video (at full temporal resolution), it is necessary to decode all temporal layers (including the video summary layers). The various temporal layers allow frame rate adaptation to user's terminal and its preferences. Two algorithms of video summarization were implemented based on a suboptimal temporal partition of the original video sequence into windows of variable size. Both algorithms use temporal rate-distortion optimization, and use Dynamic Programming (DP) to find the optimal solution [6-8].

II. Definitions and Formulations

The temporal rate of a summary is defined as the ratio given by the number of frames selected to the video summary *m*, over the total number of frames of original sequence, *n*, that is $R(S)=m/n$.

Frame distortion between two frames *j* and *k* is denoted by $d(f_j,f_k)$. Different metrics can be used to calculate frame distortion. In this paper the mean squared error (MSE) and a metric based on a principal component analysis (PCA) are used. The MSE metric is given by

$$d(f_j,f_k)_{MSE}=\frac{1}{height*width}\sum_{y=0}^{height-1}\sum_{x=0}^{width-1}(f_j(x,y)-f_k(x,y))^2 \quad (1)$$

The PCA metric is the Euclidean distance between two frames in PCA space. The PCA metric is given by

$$d(f_j,f_k)_{PCA}=\sqrt{\left\|T(D(f_j))-T(D(f_k))\right\|^2} \quad (2)$$

where D denotes a down scaling process applied to the original frames and T is the PCA transform.

Frame-by-frame distortion $d(f_k,f_{k-1})$ is a metric that reflects the "changes" of the video sequences, where f_k is the current frame and f_{k-1} is the previous one.

The first author has been supported by Fundação para Ciência e Tecnologia FCT, under grants SFRH/BD/37510/2007.

Temporal distortion D(S) is defined as the average frame distortion between the original and the reconstructed sequence and is given by

$$D(S)=\frac{1}{n}\sum_{k=0}^{n-1} d(f_k,f_k') \quad (3)$$

where f_k is current frame and f_k' is the reconstruct frame. If f_k' does not belong to the video summary then it is substituted by the most recent frame belonging to the video summary. The video summarization process can be framed as a temporal rate-distortion optimization problem[6] where the objective is to find the subset of images of the original video that provides its best representation within a given rate budget R_{max} (i.e. without using more than $m=R_{max}*n$ images). If a temporal rate constraint R_{max} is given, resulting from processing power of the terminal or the transmission rate and user's preferences, the optimal video summary S^* is the one that minimizes the summarization distortion, given by

$$S^*=\arg \min_{S} D(S), \quad \text{s.t.} \quad R(S)\le R_{max} \quad (4)$$

where R(S) is temporal rate and D(S) is the average frame distortion.

III. Distortion metrics comparison

In order to choose the most appropriate distortion metric to use in video summarization algorithm, the PCA and MSE metric are compared. Simulations were preformed on a PC with a 2.4GHz processor and 1.0 GB of RAM memory. In all simulations the temporal rate R(S) was 0.4 (good threshold between distortion and conciseness of video summary). The computational complexity is measured by processing time. The video sequences "foreman" and "mother daughter" were used with QCIF@30fps resolution to MSE metric and (8x6)@30fps resolution to PCA metric. The results of "mother daughter" are not presented in this paper, since similar performance and behavior was observed as for "foreman" sequence.

TABLE 1 - Computational complexity for "foreman" sequence

n	m	R(S)	MSE [s]	PCA [s]
20	8	0.4	0.75	3.78
40	16	0.4	12.16	6.48
60	24	0.4	61.73	42.01
80	32	0.4	194.61	117.36
100	40	0.4	478.00	273.60

TABLE 2 - Summary frames of "foreman" sequence

n	m	MSE	PCA
20	8	0,2,4,6,8,10,12,17	0,3,5,6,10,12,14,18
40	16	0,2,4,6,8,10,12,16,18,20,22, 25,29,32,35,37,	0,3,5,6,10,12,14,18,20,21,2 4,25,29,30,32,34
60	24	0,1,2,4,6,8,10,12,15,17,19,2 1,24,28,30,32,35,37,40,43,4 7,50,52,55,	0,3,6,10,12,14,18,20,24,25, 29,30,32,34,35,38,42,44,46, 47,49,51,54,57
80	32	0,2,4,6,8,10,12,15,17,19,21, 24,28,30,32,35,37,40,43,47, 50,52,55,63,65,67,69,71,72, 74,76,78	0,3,6,10,12,18,20,24,25,29, 30,32,34,35,38,42,44,46,47, 49,51,54,57,60,62,63,65,68, 73,74,76,79,
100	40	0,2,4,6,8,10,12,15,17,19,21, 24,28,30,32,35,37,40,43,47, 50,52,55,63,65,67,69,71,72, 73,74,76,78,80,84,87,89,92, 94,97	0,3,6,10,12,14,18,20,24,25, 29,30,32,34,35,38,42,44,46, 47,49,51,54,57,60,62,63,65, 68,73,74,76,77,78,80,86,88, 93,97,98

The computational complexity for the video sequence "foreman" is shown in Table 1. It is understandable that computation complexity increases when the relation *n-m* increases for both the MSE and PCA distortion metrics. Overall, the PCA metric results in lower-complexity than MSE, but for small values of *n* (e.g. 20) the MSE metric is faster. This difference is due to the scaling process used with the PCA metric and included in the computational complexity. The PCA metric is faster than MSE, because the resolution of sequence is different. When these metrics are computed with images of the same resolution, the processing time of the PCA is higher than MSE metric and the implementation of MSE metric is simpler than PCA metric.

Fig. 1 shows the frame-by-frame distortion for the "foreman" sequence. In the plot is possible to single-out video shots with high activity, for example 270-330, and regions with low activity, see 350-400. In the figure we present the distortion values using the two metrics, where the values of the PCA based distortion were upward by a factor of 100. We observe identical activity profiles up to scale factor.

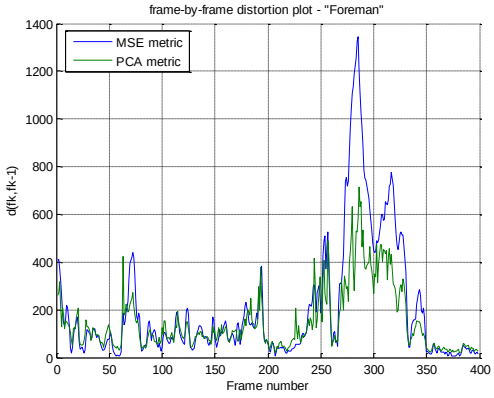

Fig. 1 - Frame-by-frame distortion for the "foreman" sequence

Based on the results of Fig. 1 a sub-optimal summarization algorithm was proposed, where a variable size search window is used. Its size is based on the activity level as defined for Fig. 1.

IV - A SUB-OPTIMAL FAST VIDEO SUMMARIZATION ALGORITHM

The sub-optimal fast video summarization algorithm is based on the reduction of search window of video sequence. The algorithm defines the distortion threshold, as given by

$$D(S)_{treshold} = (1/nseg)\sum_{k=1}^{n-1} d(f_k, f_{k-1}) \quad (5)$$

where *nseg* is number of segments. The original *n* frames are divided into segments whose number (*nseg*) is defined beforehand. The distortion threshold is used to determine the number of frames that belong to each segment, for a constant temporal rate R(S) in the segment. After determination of the number of frames in the segment, the algorithm searches the optimal video summarization for the segment. The proposed algorithm was implemented in C++. The pseudo-code of proposed algorithm is given by:

```
Definition the number_of_segments;
Find distortion_threshold;
For (segment i =1 till number_of_segments)
    While (total _distortion <= distortion_threshold )
        Frame x belongs to segment i
        Increment the Frame number
        Calculate total_distortion
    End
End
Definition the summary frames in segment i
Find the optimal video summarization in segment i
```

Fig. 2 represents the segmented windows for the "foreman" sequence of 100 frames for nseg=4 segments and distortion threshold computed as described previously. The number and the frames retained in each segment were found through a processing loop that compares the distortion threshold value and the successive addition of the frame-by-frame distortion. The processing stops when the latter becomes larger than the former. To ensure a constant R(S), the number of frames we keep in the summary for each segment is varied according to *m=R(S)*n* where *n* is the number of frames of the segment and *m* is the number of frames included in the summary (e.g. segment 1 has m=11 and n=27 and segment 2 has m=17 and n=42). Following the determination of *m* and *n* the optimal video summarization process is applied in each segment [6].

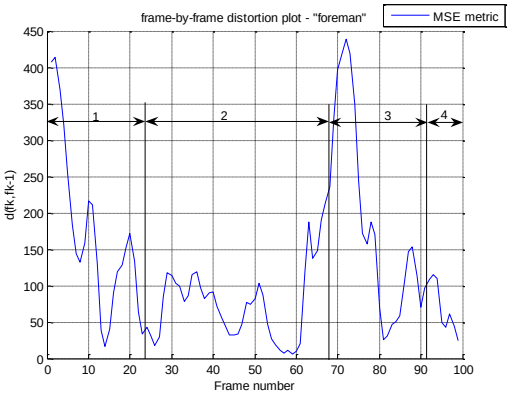

Fig. 2 - Frame segmentation of "foreman" sequence

In the tables 3-4 we show the results of sub-optimal fast video summarization algorithm for "foreman" sequence using the MSE and PCA metrics. Each table shows processing time, summary frames and average distortion for different *nseg* segments (0-original algorithm, 3, 4 and 5). When the original sequence is divided in three segments, the computational complexity is 14 of the complexity of summarizing the entire sequence as one segment and the average distortion is approximately the same. As it is shown in the tables the increase in the number of segments results in a decrease of computation complexity with a slight increase in average distortion. The decrease of computer complexity is independent of the sequence and distortion metric. These results were expected as the division of the video sequence into segments decreases the size of the search windows used by the algorithm and the optimal summarization is found faster.

TABLE 3 – Proposed computer complexity reduction algorithm for "foreman" sequence with MSE metric

Number of segments	Processing time [s]	Summary frames	Average Distortion (MSE)
0	478 100%	0,2,4,6,8,10,12,15,17,19,21,24,28,30,32,35,37,40,43,47,50,52,55,63,65,67,69,71,72,73,74,76,78,80,84,87,89,92,94,97	81.72
3	62 14%	0,2,4,6,8,10,12,16,18,20,22,25,29,31,33,36,39,41,44,48,51,53 56,63,65,67,69,71,72,73,74,76,78,80,84,87,89,92,94,98	82.68
4	18 3.8%	0,2,4,6,8,10,12,16,19,21,24,27,29,31,33,35,37,39,41,43,45,48,50,52,55,63,65,67,69,71,73,75,77,79,83,87,89,92,94,97	86.35
5	11 2.3%	0,2,4,7,10,12,15,17,19,21,24,28,30,32,35,37,39,41,44,48,50,52,55,63,65,67,69,70,71,72,74,76,78,80,84,87,89,92,95,97	86.97

TABLE 4 - Proposed computer complexity reduction algorithm for "foreman" sequence with PCA metric

Number of segments	Processing time [s]	Summary frames	Average Distortion (PCA)
0	272.6 100%	0,3,6,10,12,14,18,20,24,25,29,30,32,34,35,38,42,44,46,47,49,51,54,57,60,62,63,65,68,73,74,76,77,78,80,86,88,93,97,98	1.43
3	41.7 15%	0,3,6,10,12,18,20,24,25,29,30,32,34,35,38,42,44,46,47,49,51,54,57,60,62,63,65,69,70,71,73,74,76,79,80,86,88,93,97,99	1.44
4	20.3 7.45%	0,3,6,10,12,14,18,20,21,24,25,29,30,32,34,35,38,42,44,46,47,49,53,57,60,62,63,66,69,70,71,73,74,76,79,80,86,88,93,98	1.46
5	15.2 5.56%	0,3,6,10,12,14,18,20,21,23,24,25,30,34,35,38,42,44,46,47,49,51,54,57,60,62,63,65,68,72,73,74,76,77,78,80,86,88,93,96,98	1.49

V. CODING WITH TEMPORAL SCALABILITY

In the previous sections (III and IV) we presented and examined the performance of two video summarization algorithms, published in[6] and our own fast solution. This section presents two temporally scalable video coding schemes, where the layers have non-uniform temporal sampling, in order to accommodate the coding of the

sequence that resulted from the process of temporal summarization. The schemes are based on the H.264/SVC coding tools and are able to produce a scalable bitstream which includes a video summary at different temporal layers. A normal scalable video bitstream comprises layers which allow extraction and rendering of video with different temporal, spatial and quality levels but so far no solution has been presented which includes summarization functionality in the scalable coding arrangement. We now suggest two approaches to achieve this goal.

In Fig. 3, a scheme with two layers is hinted, where the video summary corresponds to the base layer and an enhancement layer (layer1) is added to permit decoding of the full temporal resolution. These two temporal layers have non-regular frame rate. The base layer can be independently decoded but layer 1 cannot be decoded by itself because it depends on the lower layer. A group of pictures (GOP) is defined in which the first and the first frames of the next GOP are coded as intra pictures, and the frames between the first and the first of the next GOP are coded as B pictures, with reference to the previous and next closer frames. In Fig. 3 and Fig. 4 the GOP is regular, but it can be made to vary dynamically with time, in order to match the video summary frame distribution.

A somewhat similar method proposal with three temporal layers is presented in Fig. 4. Here the full resolution corresponds to all temporal layers and the video summary is divided into two layers, namely base layer and layer 1. The base layer is composed of key frames used in the coding process as reference frames. The enhancement layer pictures of the two schemes are coded as B pictures, where the reference pictures are restricted to previous and following pictures. If the previous or next reference picture is not in the same layer, the algorithm uses the reference pictures of inferior layers. These schemes are compatible with the syntax of H.264/SVC which supports the dynamic GOPs. The JSVM (H.264/SVC reference software) decoder should also be capable of decoding bitstreams with any temporal coding structure. However, the JSVM encoder can only be configured to use a fixed GOP size.

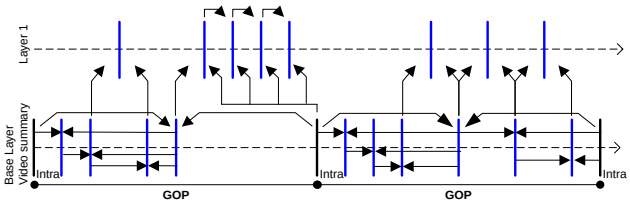

Fig. 3 - Temporal scalability of VS with two layers

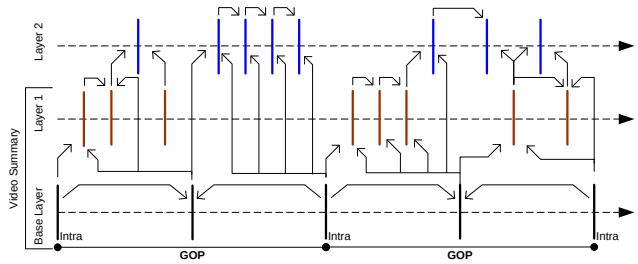

Fig. 4 - Temporal scalability of VS with three layers

VI. Conclusions and Future Work

In this work a new algorithm was presented that achieves reduction of the computational complexity of an optimal video summarization algorithm, published in [6], with gains of nearly 90% at about the same average distortion. On the one hand, by increasing the number of segments, our sub-optimal fast video summarization algorithm achieves lower computational complexity at the expense of a small increase in summary distortion. On the other hand as the number of segments decrease, the computational complexity increases and the distortion approaches that of the Zhu Li summarization algorithm's.

Two distortion metrics (MSE and PCA) were compared in the summarization algorithm. It was found that PCA metric is better in terms of computational complexity than the MSE metric. We have also proposed two temporal scalable examples to code video with inclusion of one or more summary layers. For future work these methods will be implemented in the SVC reference codec in order to evaluate their performance. The proposed methods are suitable for a wide range of applications where user's time, power and bandwidth are limited. Another application is on video-on-demand systems where they will provide the user with a tool to review summarized versions of the videos to make it easier selecting the one(s) he wants to see.

References

[1] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, 2007.

[2] W. Yao, L. Zhu, and H. Jin-Cheng, "Multimedia content analysis-using both audio and visual clues," *Signal Processing Magazine, IEEE*, vol. 17, pp. 12-36, 2000.

[3] D. Daniel, K. Vikrant, and D. David, "Video summarization by curve simplification," in *Proceedings of the sixth ACM international conference on Multimedia* Bristol, United Kingdom: ACM, 1998.

[4] Y. Gong and X. Liu, "Video summarization with minimal visual content redundancies," in *Image Processing, 2001. Proceedings. 2001 International Conference on*, 2001, pp. 362-365 vol.3.

[5] P. M. Fonseca and F. Pereira, "Automatic video summarization based on MPEG-7 descriptions," *Signal Processing: Image Communication*, vol. 19, pp. 685-699, 2004.

[6] Z. Li, G. M. Schuster, A. K. Katsaggelos, and B. Gandhi, "Rate-distortion optimal video summary generation," *Image Processing, IEEE Transactions on*, vol. 14, pp. 1550-1560, Oct. 2005.

[7] D. P. Bertsekas, *Dynamic Programming: Deterministic and stochastic models*: Prentice-Hall, 1987.

[8] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *EEE Trans. Inf. Theory*, vol. IT-13, pp. 260-269, April 1967.