Scalable Video Summarization based on Visual Attention Model

Amr Abozeid
Mathematics department,
Computer Science Division
Faculty of Science,
Al-Azhar University, Cairo, Egypt
email: aabozeid@azhar.edu.eg

Hesham Farouk
Computers and Systems Department,
Electronics Research Institute (ERI),
Cairo, Egypt
email: hesham@eri.sci.eg

Kamal ElDahshan
Mathematics department,
Computer Science Division
Faculty of Science,
Al-Azhar University, Cairo, Egypt
email: dahshan@gmail.com

***Abstract*— Scalable video coding and summarization are becoming important research fields in many adaptive video applications. In this paper, we propose a scalable video summarization framework based on visual attention model. This framework utilizes the scalable video coding to produce scalable summaries. Experiments have been conducted to prove the concept and to measure the time performance of the proposed framework. The results show that the proposed framework is an efficient and promising solution.**

Keywords-Scalable Video Coding; Summarization; Visual Attention Model; Video Processing.

I. INTRODUCTION

As a result of the dramatic growth of video creation, research fields, such as video summarization, browsing, adaptation, indexing, and retrieval have been hot topics of recent research. Video Summarization (VS) can be defined as the creation of compact video representation. This new representation can provide the user with brief information about the video content. The advantages of VS include but are not limited to, enhancing browsing, streaming, storage and quick retrieval of videos.

Video summarization is very important nowadays, especially in the context of mobile computing and to ubiquitous accessing needs. Farouk et al. [4] presented a comparative study of mobile video summarization techniques. The comparative study showed that building an adaptive VS approach is required for many applications. VS adaptation can be defined as the ability of automatically producing a summary content that meets the user's preferences and device capabilities.

The main target of Scalable Video Coding (SVC) is to produce one bit-stream that contains multiple layers. Each layer has a specific resolution, quality and frame rate. In SVC, the encoding process is performed once, while many layers (versions) can be extracted from the bit-stream, according to the specific needs (adaptation needs) [5][6].

Although video summarization is an extensively studied topic in the literature [1][3][4][7][8], previous researchers focus mainly on a single scale summary (single output summary). In some cases, producing one scale summary may be insufficient. A scalable video summary has a number of applications. These applications include video summary adaptation, progressive video access, video visualization and interactive video browsing. The SVC concepts can also be applied in the VS context as an additional attributes of the generated summaries. The following Scalable Video Summarization (SVS) modalities are possible:

1. Temporal scalability (keyframes number, duration): adapt keyframes number or duration length to meet the user's request.
2. Spatial scalability (frame size): the video summary is coded at multiple spatial resolutions.
3. Quality scalability: video summary is coded at a single spatial resolution with different qualities.
4. Hybrid scalability: a combination of the three scalability modalities described above.

This paper proposes a scalable video summarization framework based on VAM. This framework is summarized as follows.

1. Extract and partially decoded the base layer of scalable video. The main goal of this step is to reduce the computational cost of the following steps.
2. Feature Extractions: the goal of this step is to extract features (e.g., color, motion, etc.) from the pre-sampled frames. Then build feature based curve for each feature (e.g., color curve).
3. Attention Curve Construction: after the feature based curves are obtained separately, these curves need to be merged in a meaningful way to construct the final attention curve. The attention curve peaks indicate the corresponding video frames or segments which most likely attract user's attention.
4. Scalable keyframe selection: the base layer and enhanced layers video summary are extracted from the scalable video based on the attention curve values.

Initial experiments are conducted to prove the concept of this framework. The results show that the proposed framework is an efficient and promising solution. The rest of this paper is organized as follows. Section II introduces some related work about the SVS. The proposed framework is discussed in Section III. Section IV presents the experiments and the results of the proposed framework. Finally, Section V concludes the paper and suggests future work.

II. RELATED WORK

The concept of SVS was first discussed in [9], where the scalable summary was introduced as a special set of embedded summaries. It presented a framework that consists of two main stages: analysis and generation, similar to SVC.

 ISBN: 978-1-61208-611-8

The main objective is to analyze the video sequence once and generate many summaries with different lengths (analyzing once, generate many). During the analysis stage, the input video bitstream is divided into basic units called Group of Pictures (GoPs). A ranked list of these GoPs is built using the hierarchical clustering with average linkage and a ranking algorithm. Finally, the scalable summary is obtained at the generation stage depending upon the length (e.g., keyframes number, skim duration) requested by the user and/or the context. However, the analysis of the input video in [9] is not scaled to the size of the input. As a result, it fails to generate effective summaries for long duration videos due to the substantial increase in the computational cost associated with the analysis stage.

A framework based on sparse dictionary selection was proposed for scalable summarization of consumer (home) videos [10]. It formulates the video summarization problem as a dictionary selection problem. The video frames are considered as an original feature pool. An optimal subset is selected as "dictionary" from this pool under two constraints; sparsity and lower reconstruction error. Sparsity means the extracted dictionary should be as small as possible and selected from the original feature pool in a uniformly scattered way. Low reconstruction error means that the original video can be reconstructed with high accuracy using the selected dictionary (i.e., the selected dictionary is the most representative frame sets). This framework is designed to extract a scalable key frame and/or a video skimming. In contrast to most existing methods, this framework allows users to choose different numbers of keyframes without incurring an additional computational cost.

Etezadifar et al. [11] proposed a new method to improve the performance of the framework proposed in [10]. In this method, VS is performed as a selection and a training sparse dictionary problem simultaneously. Thus, the dictionary selection and learning were iteratively performed. Each iteration is performed independently and the obtained response is replaced by its previous value.

Panda et al. [12] introduced an SVS framework for both the analysis and the generation stages according to the summary length determined by the user. This framework consists of a 3-step analysis stage followed by a 1-step generation stage:

1. The Video Similarity Graph (VSG) is constructed from the input video frames based upon the color feature. Each frame is represented by a 256-dimensional feature vector obtained from the color histogram using the HSV color space. Then, VSG is constructed as a weighted complete graph, where each frame is represented as a vertex. Then the skeleton graph is extracted by choosing the vertices whose degrees are higher than a certain threshold from VSG (i.e., reduce the size of the VSG).
2. A Minimum Spanning Tree (MST) based clustering is used over the skeleton graph to obtain the initial clusters.
3. The initial clusters are propagated using a random walker algorithm [13] to obtain the final clusters.
4. The keyframes (frames that are closest to the centroids of each cluster) are extracted and arranged according to the cluster significance factor.

Perez et al. [14] proposed an SVS approach which provides different levels and views of summary details. This approach is based on the data cube On-Line Analytical Processing (OLAP) operations [15]. The data cube concept has been proposed to facilitate user's navigation through multidimensional space where each move corresponds to a query using some combination of the dimensions. In this work, different audio-visual descriptors are considered. This allows the data cube partitioning in a multimodal audio-visual descriptor space. This approach was designed to process the cultural video document only.

Based on the MPEG-DASH standard [16], a Context-aware Video Summarization and Streaming (CVSS) approach was proposed in [17][18]. The CVSS was proposed to provide an adaptable video streaming for mobile devices, especially, when there are limitations in the available time or mobile energy level. The CVSS consists of 3 phases:

1. The input video is converted into an MPEG-DASH compatible format.
2. A semantic attention value for each segment of the MPEG-DASH video is computed based on VAM. Then, based on the segment attention value, dynamic video summaries are generated with different durations to meet the user's request.
3. Finally, the video summary is adapted during the streaming session in order to be suitable for the available devices and network contexts.

The main observation is: scalability in video summarization is usually related to the summary length (temporal). However, the SVS concept should be extended to other scalability modalities (e.g., quality, spatial). This may be used to adapt the output summaries to targeted contexts (e.g., device context, network context).

III. SCALABLE VIDEO SUMMARIZATION FRAMEWORK

The proposed framwork consists of an (3-step) analysis stage followed by a (1-step) generation stage. Figure 1 is a block diagram shows all the four steps. The description of these steps are as follows:

A. Extract and partially decoded

In this step, the following tasks were implemented.

a. Extract the base layer (layer number 0) from the input scalable video by JVSM BitStreamExtractorStatic function. As shown in Table I, the base layer (layer number 0) has the minimum configurations and this will significantly reduce the computations.

b. Partially decode and extract the base layer frames.

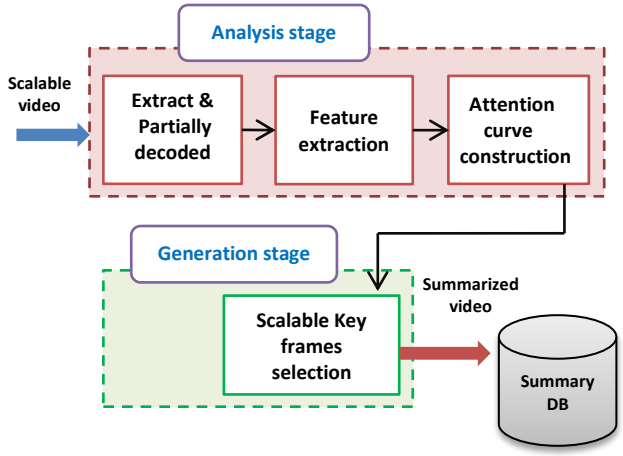

Figure 1. A block diagram of the proposed framework

TABLE I. SCALABLE VIDEO LAYERS

Layer	Resolution	Frame rate	Bitrate	MinBitrate
0	160x 96	1.8750	26.20	26.20
1	160x 96	3.7500	33.00	33.00
2	160x 96	7.5000	40.70	40.70
3	160x 96	15.0000	49.10	49.10
4	160x 96	30.0000	57.20	57.20
5	320x192	1.8750	129.00	129.00
6	320x192	3.7500	165.50	165.50
7	320x192	7.5000	208.10	208.10
8	320x192	15.0000	252.90	252.90
9	320x192	30.0000	288.90	288.90
10	640x368	1.8750	462.20	462.20
11	640x368	3.7500	625.90	625.90
12	640x368	7.5000	843.70	843.70
13	640x368	15.0000	1097.00	1097.00
14	640x368	30.0000	1303.00	1303.00

B. *Feature extractions*

In this step, color and motion features are extracted. Therefore, this step consists of two sub-steps: Static Attention Curve Extraction and Motion Attention Curve Extraction. These two sub-steps are adapted from [17] and briefly discussed in the next subsections.

1) Static Attention Curve Extraction

In this step, the static attention curve is extracted from the video frames based on the color feature. As shown in Figure 2, the curve describes the video contents by representing the important frames corresponding its peeks. The horizontal parts of the curve mean that the corresponding frames having the same attended areas probability and almost contain the same information. The gradual changes in the curve mean that there is a gradual difference in the content of the corresponding frames. On the other hand, sudden changes in the curve mean that there is a significant difference in the content of the corresponding frames.

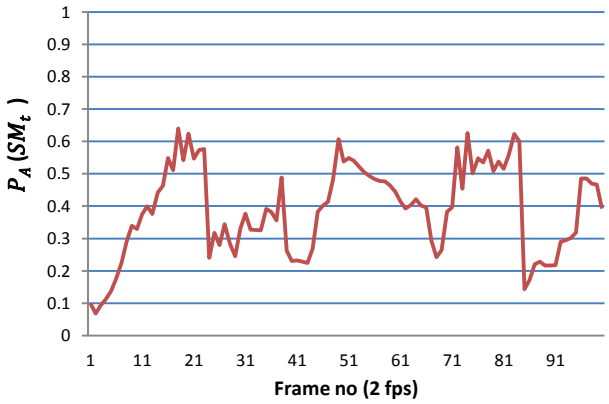

Figure 2. Static attention curve of "Big Buck Bunny" video

The Static Attention Detection Algorithm (see Figure 3) can be summarized as:

a. The Saliency Map (SM) is computed for each frame. A saliency map is a gray image, which contains attended/salient areas (bright areas) and unattended/non-salient areas (dark areas). The attended areas usually attract the user attention.
b. The attended areas of the saliency map are extracted as:
   i. Each SM is divided into non-overlapping Macro-blocks (MB).
   ii. Accordingly, each SM is represented by two sets (A and U). The set A is the set of all non-overlapping attended blocks (areas). Similarly, U is the set of all non-overlapping unattended blocks (areas).
c. After normalizing the value of the $P_A(SM_t)$ for each frame to [0, 1], a static attention curve (SC) is obtained.

Input: F_t // the input frame at a time t
Output: $P_A(SM_t)$// the probability of attended areas A in SM_t
Start
1. Initialize $A = U = \emptyset$
2. Compute SM_t for F_t
3. Loop for each $MB_{i,j}$ in the SM_t
4. If $(C(MB_{i,j}) \geq \varepsilon^{SM})$ then
5. Add $MB_{i,j}$ to the attended set A
6. Else
7. Add $MB_{i,j}$ to the unattended set U
8. End loop
9. $P_A(SM_t) = \frac{|A|}{|A|+|U|}$

End

Figure 3. Static Attention Detection Algorithm

2) Motion Attention Curve Extraction

Most of the video summarization approaches are based on motion feature in different ways [19][20][21][22]. In this step, we adopt the Fast Directional Motion Intensity Estimation (FDMIE) algorithm. FDMIE aims to detect the

Motion Intensity (MI) between the consecutive frames. The following two options decrease the complexity of FDMIE.

a. The motion intensity estimation has been applied to the regions in each frame that could potentially attract users' attention due to the motion (i.e., attended areas).
b. The Sum of Absolute Differences (SAD) is used to determine the matching between two blocks. The SAD is more used because it has a higher-quality precision and involves lower computational cost [23][24].

According to the FDMIE algorithm (Figure 4), the motion intensity between the saliency maps is computed.

a. For each block in SM_{t-1} (saliency map extracted from the t-1 frame), FDMIE computes the current minimum (C_{MIN}) distortion between this block and the corresponding block in SM_t by SAD.
b. The FDMIE searches the eight directions around the target block uses the One-at-a-Time Search (OTS) strategy. In OTS, the block-by-block search along a direction is continued if a newly searched block has lower distortion than the previously searched block. Otherwise, the search in that direction stops. The minimum distortion found in each directional search is set as a directional minimum (D_{MIN}) distortion.
c. Then, the Relative Distortion Ratio (RDR) is computed by dividing D_{MIN} by C_{MIN}.
d. If RDR between current D_{MIN} and C_{MIN} is lower than ε^D then other directional searches will be skipped and a final position of the block with C_{MIN} value become the position of the block with D_{MIN} value.
e. Otherwise, other directional searches will be started.
f. After a search round is completed, the lowest distortion among the D_{MIN}s (if found) is set as C_{MIN} and the next search round starts at the block with C_{MIN}.
g. Finally, the motion intensity $MI(P_{i,j})$ of the target block $P_{i,j}$is computed as the distance between the $P_{i,j}$position and the position of the block with final C_{MIN} value. Consequently, the motion intensity MI_{t-1} between the saliency SM_{t-1} and SM_t is computed as in (1).

$$MI_{t-1} = \sum_i^w \sum_j^h MI(P_{i,j}),\ P_{i,j} \in SM_{t-1} \qquad (1)$$

The range of threshold ε^D is [0, 1] and it used to control the FDMIE convergence speed of the algorithm. The higher threshold ε^D will speed up the convergence of the FDMIE, but it will also decrease the prediction quality. For example, if $\varepsilon^D$is set at 0.5 implies that the prediction quality is less than 50% and the number of search blocks is reduced to the half. Initially, the select value of $\varepsilon^D = 0.5$. In future, more experimental studies will be conducted o determine the best value of ε^D .

Input: $A_{t-1}, A_t, SM_{t-1}, SM_t$
Output: MI_{t-1}
Start
For each $P \in SM_{t-1},\ P \in A_{t-1}$
1. Initialize flag=false
2. Compute $C_{MIN} = SAD(P_{i,j}, Q_{i,j})$
3. For each 8 directions around the point with C_{MIN}
   a. Compute $D_{MIN} = SAD(P_{i,j}, Q_{i+di,j+dj})$
   b. If $D_{MIN} < C_{MIN}$
      If $RDR(D_{MIN}, C_{MIN}) < \varepsilon^D$
         Then $C_{MIN} = D_{MIN}$ and go to step 5.
      Else flag = true
      End for
4. If flag = true then D_{MIN}s are compared. The lowest one is set as C_{MIN} and update the corresponding position, go to step 1.
5. Compute $MI(P_{i,j})$ and it to MI_{t-1}

End For
Return MI_{t-1}
End

Figure 4. FDMIE Algorithm

The FDMIE output is a numeric value that represents the motion intensity of the frame F_{t-1} . After normalizing the motion intensity value for each frame to [0, 1] a motion attention curve (MC) is obtained. Figure 5 shows motion attention curve of "Big Buck Bunny" video.

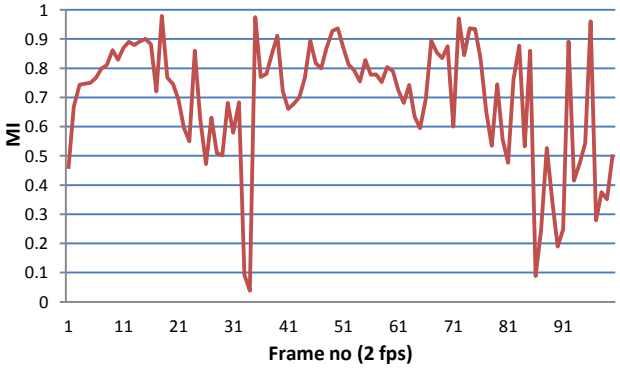

Figure 5. Motion attention curve of "Big Buck Bunny" video

C. Attention Curve Construction

After the static and the motion curves are obtained separately, these curves are merged to construct the final Attention Curve (AV). Figure 6 shows an example of the final attention curve that has been created by the proposed framework. The AV peaks indicate the corresponding video frames most important and usually attract user's attention. The final attention curve was constructed as in (2).

$$AC = w_s \times SC + w_m \times MC \qquad (2)$$

Where SC represents static attention curve and MC represents motion attention curve. The weight values w_s and w_m are used for a linear combination which satisfies the two conditions:

$$w_s , w_m \geq 0 \quad \text{and} \quad w_s + w_m = 1$$

In the experimental phase, we will conduct experiments to determine the best values for w_s and w_m. Initially, we determine $w_s = 0.4$ and $w_m = 0.6$.

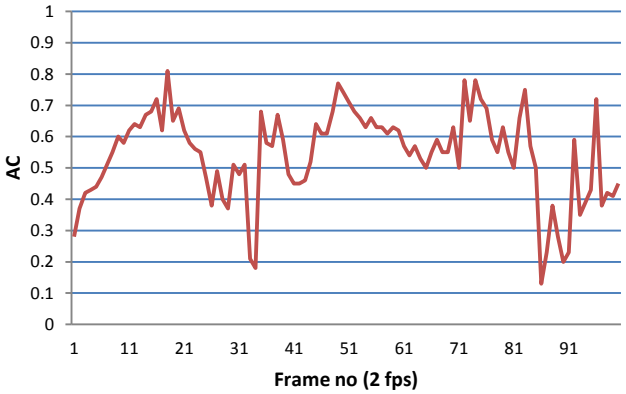

Figure 6. Attention curve of "Big Buck Bunny" video

D. *Scalable keyframes selection*

In the generation stage, the user can determine the number of keyframes to be generated (i.e., temporal scalable summary). Based on the attention curve, the frames with high attention values are selected to form the base-layer video summary (base-layer keyframes). The enhanced summary layers are constructed by selecting the corresponding base-layer keyframes form the enhanced scalable video layers. Finally, the selected scalable key-frames (scalable video summary) are saved in the summary folder.

IV. EXPERIMENTAL RESULTS

A prototype was implemented as web application using a J2EE (JSP and Servlet) technologies. Some third-party libraries are used during the implementations such as FFmpeg [25] and JVSM [26]. Most of the standard video formats (e.g. mp4, flv, avi, etc.) are supported by this prototype in addition to .264 (scalable format).

Table II describes the selected data set videos from the YouTube. All videos are transcoded from in H.264/AVC (.mp4) format to JVSM [26] scalable format (.264) with different layers as described in Table I. The efficiency of the proposed framework is evaluated by comparing the summarization time for both .mp4 and .264 formats.

The experiments were carried out on a PC equipped with an Intel Core i7 and 8 GB of RAM. The experiments are carried out on all videos mentioned in Table II. The results of these experiments are organized in Table III. For each video, we record the processed frames (total number of the processed frames) and Analysis Time (AT) of these frames. Then compute a number of Processed Frames Per Second (PFPS) by dividing the processed frames by AT.

TABLE II. DESCRIPTION OF THE DATASET

Video name	**Duration**	**FPS**	**Frames #**	**Resolution (W× H) (pixels)**
Big Buck Bunny	00:09:56	24	14304	640×360
Tears Of Steel	00:12:14	24	17616	640×360
Of Forests and Men	00:07:33	24	10872	640×360
Beautiful Birds	00:10:46	30	19380	640×360
Bird feeding babies	00:10:59	30	19770	640×360
Bird noises sounds	00:17:55	30	32250	640×360
Final Repechages	00:14:14	25	21350	640×360
High Jump	00:12:49	25	21350	640×360
Land Rover Discovery	00:07:14	29	12586	640×360
Sofia2	00:16:26	25	24650	640×360
Solar System	00:09:28	29	16472	640×360
Strawberry	00:20:59	25	31475	640×360

As shown in Table III, the AT of .264 videos is less than .mp4 videos. The AT includes the feature extraction and motion attention curve extraction. In case of scalable videos, the proposed framework can process an average of 325.7 fps. For other video formats, the proposed approach can process an average of 299.4 fps. It is important to note that those results depend on the computational power of the target environment.

TABLE III. THE PROPOSED FRAMEWORK EFFICIENCY EVALUATION

Video name	**MP4**			**Scalable videos (.264)**		
	Processed frames	***AT (s)***	***PFPS***	***Processed frames***	***AT (s)***	***PFPS***
Big Buck Bunny	1194	3.3	357.4	895	2.2	407.4
Tears Of Steel	2937	7.0	416.7	1102	4.0	274.9
Of Forests and Men	448	3.1	145.3	709	3.4	209.9
Beautiful Birds	1294	8.4	153.2	1213	3.2	376.4
Bird feeding babies	1320	3.5	375.4	1237	3.5	351.4
Bird noises sounds	1570	6.4	247.2	2018	7.3	276.4
Final Repechages	1710	6.6	257.3	1336	5.9	226.8
High Jump	1540	4.3	360.6	1203	3.2	373.7
Land Rover Discovery	870	2.3	379.7	814	1.9	418.3
Sofia2	1974	9.9	198.7	1542	5.3	289.1
Solar System	1138	2.8	413.4	1066	2.4	440.5
Strawberry	1857	6.5	287.4	1968	7.5	263.7
Average	**1487.7**	**5.3**	**299.4**	**1258.6**	**4.2**	**325.7**

 ISBN: 978-1-61208-611-8

V. CONCLUSIONS

In this paper, we propose a scalable video summarization framework based on Visual Attention Model (VAM). In this framework, the concept of SVC was extended to the video summarization context. Therefore, the input scalable video will be analyzed once and then generate many summaries with different scalability modalities (such as temporal, quality and/or spatial). VAM is applied to extract the semantic meaning of the low-level video features (color and motion). We carried out experiments to measure the efficiency of the proposed framework. The results show that the proposed framework is an efficient and promising solution. In the future, we intend to conduct more experiments and improve the proposed framework.

ACKNOWLEDGMENTS

This research is funded by the Science and Technology Development Fund (STDF), Egypt, project titled "Efficient Media Digital Library Design and Implementation of Summarized Video based on Scalable Video Coding for H.264 (MDLSS)", (project ID 15269).

REFERENCES

[1] M. Ajmal, M. H. Ashraf, M. Shakir, Y. Abbas, and F. A. Shah, "Video summarization: techniques and classification," Computer Vision and Graphics, Springer, vol. 7594, 2012, pp. 1-13, doi: 10.1007/978-3-642-33564-8_1.

[2] Z. Xiong, R. Radhakrishnan, A. Divakaran, Y. Rui, and T. S. Huang, *A unified framework for video summarization, browsing & retrieval: with applications to consumer and surveillance video*: Academic Press, 2006.

[3] B. T. Truong and S. Venkatesh, "Video abstraction: A systematic review and classification," ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP), vol. 3, no. 1, 2007, pp. 37, doi: 10.1145/1198302.1198305.

[4] H. Farouk, K. ElDahshan, and A. Abozeid, "The State of the Art of Video Summarization for Mobile Devices: Review Article," Graphics, Vision and Image Processing GVIP, vol. 14, no. 2, 2014, pp. 37-50.

[5] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H. 264/AVC standard," IEEE Transactions on circuits and systems for video technology, vol. 17, no. 9, 2007, pp. 1103-1120.

[6] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of SHVC: Scalable Extensions of the High Efficiency Video Coding Standard," IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 1, 2016, pp. 20-34.

[7] 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, 2008, pp. 121-143, doi: 10.1016/j.jvcir.2007.04.002.

[8] R. Pal, A. Ghosh, and S. K. Pal, "Video Summarization and Significance of Content: A Review," *Handbook on Soft Computing for Video Surveillance*, pp. 79-102: CRC Press, 2012.

[9] L. Herranz and J. M. Martinez, "A framework for scalable summarization of video," IEEE Transactions on Circuits and Systems for Video Technology, vol. 20, no. 9, 2010, pp. 1265-1270.

[10] Y. Cong, J. Yuan, and J. Luo, "Towards scalable summarization of consumer videos via sparse dictionary selection," IEEE Transactions on Multimedia, vol. 14, no. 1, 2012, pp. 66-75.

[11] P. Etezadifar and H. Farsi, "Scalable video summarization via sparse dictionary learning and selection simultaneously," Multimedia Tools and Applications, 2016, pp. 1-25.

[12] R. Panda, S. K. Kuanar, and A. S. Chowdhury, "Scalable Video Summarization Using Skeleton Graph and Random Walk." Year, pp. 3481-3486, doi.

[13] N. Paragios, Y. Chen, and O. Faugeras, *Handbook of mathematical models in computer vision*: Springer Science & Business Media, 2006.

[14] K. R. Perez-Daniel, M. N. Miyatake, J. Benois-Pineau, S. Maabout, and G. Sargent, "Scalable video summarization of cultural video documents in cross-media space based on data cube approach." Year, pp. 1-6, doi.

[15] J. Gray *et al.*, "Data cube: A relational aggregation operator generalizing group-by, cross-tab, and sub-totals," Data mining and knowledge discovery, vol. 1, no. 1, 1997, pp. 29-53.

[16] "MPEG-DASH Standard," [retrieved: 12, 2017], http://mpeg.chiariglione.org/standards/mpeg-dash.

[17] H. Farouk, Kamal A. ElDahshan, and A. Abozeid, "Effective and Efficient Video Summarization Approach for Mobile Devices," International Journal of Interactive Mobile Technologies (iJIM), vol. 10, no. 1, 2016, pp. 19-26.

[18] H. Farouk, K. El Dahshan, and A. Abozeid, "Context-Aware Joint Video Summarization and Streaming (CVSS) Approach," IEEE International Symposium on Multimedia (ISM), 2016, pp. 597-602.

[19] N. Ejaz, I. Mehmood, and S. W. Baik, "Efficient visual attention based framework for extracting keyframes from videos," Signal Processing: Image Communication, vol. 28, no. 1, 2013, pp. 34-44.

[20] A. B. Mejía-Ocaña *et al.*, "Low-complexity motion-based saliency map estimation for perceptual video coding," in 2nd National Conference on Telecommunications (CONATEL), 2011, pp. 1-6.

[21] J.-L. Lai and Y. Yi, "Key frame extraction based on visual attention model," Journal of Visual Communication and Image Representation, vol. 23, no. 1, 2012, pp. 114-125.

[22] N. Ejaz, I. Mehmood, and S. W. Baik, "Feature aggregation based visual attention model for video summarization," Computers & Electrical Engineering, vol. 40, no. 3, 2014, pp. 993-1005.

[23] Q. Yang, C. LI, and Z. LI, "Motion Navigation System Estimation Algorithm in Mobile Phone Video Learning System," Journal of Computational Information Systems, vol. 10, no. 16, 2014, pp. 7187-7194.

[24] M. Santamaria and M. Trujillo, "A comparison of block-matching motion estimation algorithms," in 7th Colombian Computing Congress (CCC), 2012, pp. 1-6.

[25] "FFmpeg," [retrieved: 12, 2017], https://www.ffmpeg.org/.

[26] "JSVM Reference Software," [retrieved: 12, 2017], https://www.hhi.fraunhofer.de/en/departments/vca/research-groups/image-video-coding/research-topics/svc-extension-of-h264avc/jsvm-reference-software.html.

 ISBN: 978-1-61208-611-8