2014 22nd International Conference on Pattern Recognition

Key-frame Extraction using Weighted Multi-View Convex Mixture Models and Spectral Clustering

Antonis I. Ioannidis, Vasileios T. Chasanis and Aristidis C. Likas
Department of Computer Science and Engineering
University of Ioannina
Ioannina, Greece
{aioannid, vchasani, arly}@cs.uoi.gr

***Abstract*—Reliable video summarization is one of the most important problems in digital video processing and analysis. The most common approach used for shot representation is the extraction of a set of key-frames sufficiently representing the total content of the shot. In such way, the whole video content can be represented using only a few, cautiously picked, non redundant key-frames maintaining at the same time a great percentage of information. A typical approach is to extract key-frames using clustering. However, using a single image descriptor to extract key-frames is not sufficient due to large variations in the visual content of videos. In our approach, a weighted multi-view clustering algorithm is employed to combine two different image descriptors into a single similarity matrix, that serves as an input to a spectral clustering algorithm. Each image descriptor (view) does not contribute equally to the similarity matrix, but the weighted multi-view clustering algorithm associates a weight with each view and learns these weights automatically. Numerical experiments using a variety of videos demonstrate that our method is capable of efficiently summarizing video shots regardless of the characteristics of the visual content of the video.**

I. Introduction

In recent years there has been a great interest in the field of digital video processing. High quality mobile devices like notepads and smartphones, providing video applications such as recording and playing video, along with the already existed digital cameras, have created an enormous amount of digital videos. As a result, in order to handle this huge amount of data, efficient methods are needed for better storing, indexing and video retrieval.

A video sequence can be decomposed into shots. A shot is defined as an unbroken sequence of video frames taken from a single camera. The efficient summarization of a video shot is needed for two reasons. Firstly, one can rapidly make an assessment about the video content by inspecting the key-frames of the shots of the video. Secondly, having extracted key-frames from video shots, one can proceed further with grouping shots into scenes.

Two key factors should be taken into consideration when key-frames are extracted. Firstly, the key-frames should represent the whole video content without missing important information and secondly, these key-frames should not be similar, in terms of video content information, thus containing redundant information. A major category of key-frame extraction algorithms detect abrupt changes in the similarity between successive frames [1], [2]. Another category of key-frame extraction algorithms perform clustering of shot frames into groups and select a representative frame of each group as key-frame. In [3], multiple frames are detected using unsupervised clustering based on the visual variations in shots. A variant of this algorithm is presented in [4], where the final number of key-frames depends on a threshold parameter which defines whether two frames are similar. In [5], the mutual information values of consecutive frames are clustered into groups using a split-merge approach. A different technique for the key-frame selection is described in [6], where the key-frames position in the video is taken into account.

A major difficulty concerning the key-frame extraction problem is the large variety in visual content. The use of a single image descriptor (e.g. color histograms, shape and texture descriptors) cannot guarantee the efficient summarization of all videos. To ameliorate this shortcoming we propose the fusion of two different image descriptors using a single clustering algorithm. More specifically, we propose to use two different descriptors to compute a single similarity matrix that will be used as input to a spectral clustering algorithm. Typical multiview approaches assign equal weights to different views (image descriptors). In the herein approach, the weight of each view is computed automatically using a weighted multi-view clustering algorithm [7]. The weights reflect the quality of each view and therefore affect its contribution to the final clustering solution accordingly.

The rest of the paper is organized as follows. In Section II we describe the image descriptors employed in our method. In Section III we present the key-frame extraction process that includes the weighted multi-view clustering algorithm implemented to compute the weights assigned to the different views and the spectral clustering algorithm used to cluster frames into groups. In Section IV we present numerical experiments. Finally, in Section V we provide conclusions and suggestions for further study.

II. Image descriptors

In our method we have employed several image descriptors to describe the content of shot frames.

- HSV Color Histograms: Two different HSV normalized color histograms are employed. The first one, presented as "HSV1D", results from the concatenation of 64 bins for hue and 16 bins for each of saturation and value. For the second one, presented as "HSV3D", we use 8 bins for hue and 4 bins for each of saturation and

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.596

value, resulting into a 128 (8×4×4) dimension feature vector. The main disadvantage of these descriptors is that they only represent the color distribution of an object ignoring its shape and texture. Color histograms are also sensitive to noise, such as lighting changes.

- Census Transform Histogram (CENTRIST): Centrist descriptor has been proposed in [8]. Centrist is claimed to be a holistic representation that captures the structural properties of an image. However, it is not invariant to rotation.
- Wavelet texture : 9 Haar wavelet sub-bands are used on 3×3 grids to form a 81-d feature vector [9]. They are suitable for texture representation.
- Scale Invariant Feature Transform (SIFT): A well known descriptor that describes local features in images [10], based on storing the weighted edge orientation histograms of salient corners of an image.

The bag of visual words representation [11] was implemented to represent shot frames when SIFT descriptors were used. All the descriptors extracted from shot frames were clustered into 20 and 50 visual words (visual vocabulary). For each frame, its corresponding set of descriptors was mapped into these 20 or 50 visual words resulting into a vector containing the normalized count of each visual word in the frame (presented as "SIFT20" or "SIFT50", respectively). Furthermore, for each image descriptor we compute a kernel matrix which depicts the similarity between the frames of a video shot (see section IV-B). In the following section, we shall refer to the kernel matrix of an image descriptor as view.

III. Key-frame extraction

The aforementioned descriptors - views capture different aspects of an image. Due to large variations observed in the visual content of videos, a single descriptor cannot efficiently describe the content of a large video database. Thus, we propose the simultaneous use of two available views. Both views are weighted to form a single similarity matrix that will serve as an input to a spectral clustering algorithm. The weights that reflect the quality of each view are computed using an algorithm for training Weighted Multi-View Convex Mixture Models [7].

A. Weighted Multi-View Convex Mixture Models

Convex Mixture Models (CMMs) [12] are simplified mixture models built to assign data points into clusters and extract representative exemplars from the data set. These models are trained by maximizing the log-likelihood and all data points are considered as cluster representatives. The data points whose prior probability is the highest are chosen as exemplars (cluster representatives) and the rest points are grouped based on their most similar exemplar. Multi-view CMMs assume that the components of each view are generated from a CMM and they intend to extract representative exemplars taking into account all the available views equally, regardless their information quality. For example, a visual descriptor may contain irrelevant information or noise. The introduction of weights, which are computed automatically, bypasses this issue allowing each view to participate in the final result with different weight according to their information quality [7].

In more detail, suppose we are given a set of N instances and V views, $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$ where x_i is the representation of the i-th instance across the views, i.e., $x_i = \{x_i^1, x_i^2, \ldots, x_i^V\}, x_i^v \in \Re^{d^v}$. The CMM of each view v is given by:

$$Q^v(x^v) = \sum_{j=1}^{N} q_j f_j^v(x^v) = \\ = C_{\phi_v}(x^v) \sum_{j=1}^{N} q_j e^{-\beta^v d_{\phi_v}(x^v, x_j^v)}, x^v \in \Re^{d^v}, \quad (1)$$

where $q_j \geq 0$ is the prior probability of the j-th component, satisfying the constraint $\sum_{j=1}^{N} q_j = 1$, $f_j(x)$ is an exponential family distribution (see Eq. 3) with d_ϕ being the Bregman divergence corresponding to the components distribution, $C_\phi(x)$ is independent of x_j, and β^v is a constant affecting the obtained number of clusters [12].

The weighted multi-view CMM [7] is defined as follows:

$$F(x = \{x^1, x^2, ..., x^V\}) = \sum_{v=1}^{V} \pi^v Q^v(x^v) = \\ = \sum_{v=1}^{V} \pi^v \sum_{j=1}^{N} q_j f_j^v(x^v), x^v \in \Re^{d^v}, \quad (2)$$

where

$$f_j^v(x^v) = C_{\phi_v}(x^v) e^{-\beta^v d_{\phi_v}(x^v, s_j^v)}, p^v \geq 0, \\ \sum_{v=1}^{V} \pi^v = 1, q_j \geq 0, \sum_{j=1}^{N} q_j = 1. \quad (3)$$

From the above equations it can be observed that $F(x)$ is a mixture model whose number of components is equal to the number of the views and each component is a CMM $Q^v(x^v)$, corresponding to the v-th view. Each CMM is associated with a weight π^v which represents the prior probability of each view in the mixture model.

It must be pointed out that due to the different characteristics of each descriptor, different descriptors may have different distributions $f_j^v(x^v)$, different values β^v and different d_{ϕ_v}. All instances are considered as possible cluster representatives, since a CMM is used for each view. Moreover, the priors q_j are the same across all views, to allow the extraction of representative exemplars based on every view. Therefore, if a component has a high q_j, then probably it is a good exemplar for all the available views. Finally, a low weight π^v indicates that the view v contains non-valuable information for the problem at hand.

Since $F(x)$ is considered a mixture model, to cluster the dataset $\mathcal{X}$ into groups, the log-likelihood of the dataset must be maximized. The log-likelihood is defined as follows:

$$L(\mathcal{X}; \{\pi^v\}_{v=1}^V, \{q_j\}_{j=1}^N) =$$
$$= \sum_{i=1}^{N} \log \sum_{v=1}^{V} \pi^v Q^v(x_i^v) = \sum_{i=1}^{N} \log(\sum_{v=1}^{V} \pi^v \sum_{j=1}^{N} q_j f_j^v(x_i^v)). \tag{4}$$

It must be noted that in contrast to CMM, this maximization problem in not convex due to the addition of the weights π^v. Local maxima can be found by applying an EM algorithm [13]. The algorithm initially guesses values for the parameters and iteratively adjusts them in order to increase the likelihood and reach a local maximum. Since the only parameters of the weighted multi-view CMM are the prior probabilities π^v and q_j, by initializing this values uniformly ($\pi^{v^{(0)}} = 1/V, q_j^{v^{(0)}} = 1/N$), multiple executions can be avoided. The weights π^v may be initialized accordingly if there is prior knowledge concerning the quality of the available views. More information about the EM for the Weighted Multi-view CMMs can be found in [7].

In order to divide the set $\mathcal{X}$ into M different clusters $C_1, C_2, \ldots, C_M$, after the completion of EM, the M exemplars with the higher q_j priors are selected creating the set $\mathcal{X}^E = \{x_1^E, x_2^E, \ldots, x_M^E\} \subset \mathcal{X}$. The rest instances x_i are assigned to the cluster C_k, whose exemplar is x_k^E, with the greater posterior probability $P(C_k|x_i)$.

$$P(C_k|x_i) = \frac{q_k^E \sum_{v=1}^{V} \pi^v f_k^{v^E}(x_i^v)}{\sum_{v=1}^{V} \pi^v \sum_{j=1}^{N} q_j f_j^v(x_i^v)}. \tag{5}$$

The assignment to clusters is given by the following equation:

$$C_k = \{x_k^E\} \cup \{x_i | P(C_k|x_i) > P(X_l|x_i), \forall l \neq k, x_i \notin \mathcal{X}^E\}. \tag{6}$$

In Figure 1, we give an example of the extracted key-frames of a video shot using the weighted multi-view CMM algorithm plus HSV ("HSV3D"), Centrist and Wavelet descriptors. The first row depicts the ground truth of the video sequence. There are four key-frames where the same person holds square placards of different color and one key-frame that represents all the frames where the person changes placards. It is obvious that only the color descriptor is relevant, providing the best single view clustering solution. When HSV view is combined with one of the two other views, the clustering solution is still optimal. Surprisingly, even when centrist and wavelet views are combined, the clustering solution of the weighted multi-view CMM algorithm is also optimal, contrary to their single-view clustering solutions. In table I, the weights assigned to the views by the weighted multi-View CMM algorithm are presented. The weight assigned to HSV view is very large indicating that this is the best view.

B. Spectral clustering

The weights π^v computed from the weighted multi-view CMM algorithm are used to build a final kernel as a weighted sum of the individual view kernels. This kernel will serve as the input similarity matrix to a spectral clustering algorithm. Spectral clustering [14] is a well-known clustering algorithm. Assume we are given n data points $X = [x_1\ x_2 \ldots\ x_n]$ and a similarity matrix S $\in \mathbb{R}^{n \times n}$, where $Sij \geq 0$ reflects the similarity between x_i and x_j. Spectral clustering uses the eigenvalues of a modified similarity matrix to group X into k clusters. To perform key-frame extraction, the video frames of a shot are clustered into groups using an improved spectral clustering algorithm [15], that employs the global k-means algorithm [16] in the clustering stage after the eigenvector computation. Then, the medoid of each group, defined as the frame of a group whose average similarity to all other frames of this group is maximal, is characterized as a key-frame. As already mentioned, the similarity matrix employed in the spectral clustering algorithm is built as a weighted sum of the two individual view kernels.

In what concerns our key-frame extraction problem, suppose we are given a set of N frames and V views , the similarity matrix that serves as input to the spectral clustering algorithm is $S = \sum_{v=1}^{V} \pi^v K^v$, where K^v is the kernel matrix defined in Section IV-B and π^v the weights assigned to views from the weighted multi-view CMM algorithm. The number of clusters k is set equal to the number of key-frames in the ground-truth for each video. In the herein approach, we use pairs of image descriptors, thus V = 2.

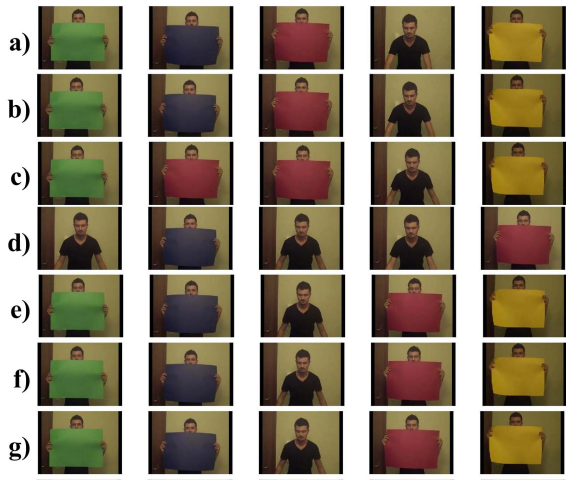

Fig. 1: Example shot sequence (Better seen in color). a) Ground truth. Clustering solution using: b) HSV, c) Centrist (CEN), d) Wavelets (WAV), e) HSV - CEN, f) HSV - WAV, g) CEN - WAV.

TABLE I: Weights assigned to the descriptors by the Weighted Multi-View CMM.

Descriptors	HSV	CEN	WAV
HSV - CEN	0.9880	0.0120	-
HSV - WAV	0.9449	-	0.0551
CEN - WAV	-	0.3286	0.6714

IV. EXPERIMENTS

In this Section we present the video sequences, kernel function and performance metrics that have been used in our experiments.

A. Datasets and ground truth

In our experiments we have processed 27 shot sequences with different visual content including car motion, construction demolition, car accidents, changing traffic lights, indoor and outdoor movement. Two different people were asked to visually extract key-frames as ground truth representing adequately the content of the shot sequences. Two different assessments were made, presented as "GT1" and "GT2" in the rest of the paper. In Figures 2 and 3, we present the number of key-frames per video sequence for each ground truth assessment and selected frames from the video sequences of the dataset, respectively.

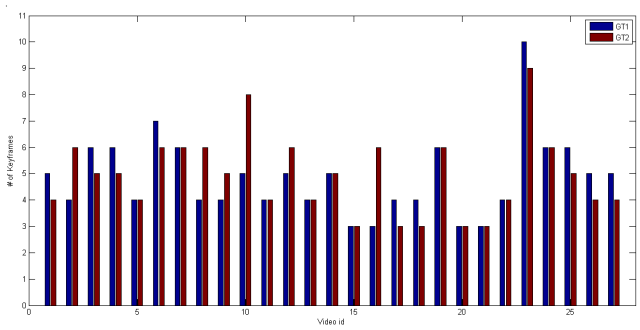

Fig. 2: Number of key-frames per video sequence for each ground truth assessment.

B. Kernel Function

In our approach, the Chi-Square kernel kernel function has been implemented to test the performance of the proposed method, due to its simplicity and effectiveness, especially in cases of histogram similarity. Given two image descriptor vectors x, y, the Chi-Square kernel kernel function is given from the following equation:

$$K_{\chi^2}(x,y) = \sum_i \frac{(x_i - y_i)^2}{\frac{1}{2}(x_i + y_i)}. \quad (7)$$

C. Performance Metrics

The evaluation of the results has been based on a visual comparison of the key-frames extracted from the experiments against the ones in the ground truth set. Two persons participated in the evaluation process and the cross-section of their evaluations was used. *Mean Value(m)* was used to evaluate the performance. Suppose we are given M video shots $VS = VS_1, ..., VS_M$ and let GT_i, F_i $(i = 1, \ldots, M)$ the number of ground truth key-frames and the number of successfully found ground truth key-frames of each video shot, respectively. Thus, F_i/GT_i is the percentage of successfully found ground truth key-frames per video shot. Mean Value is computed from the following equation:

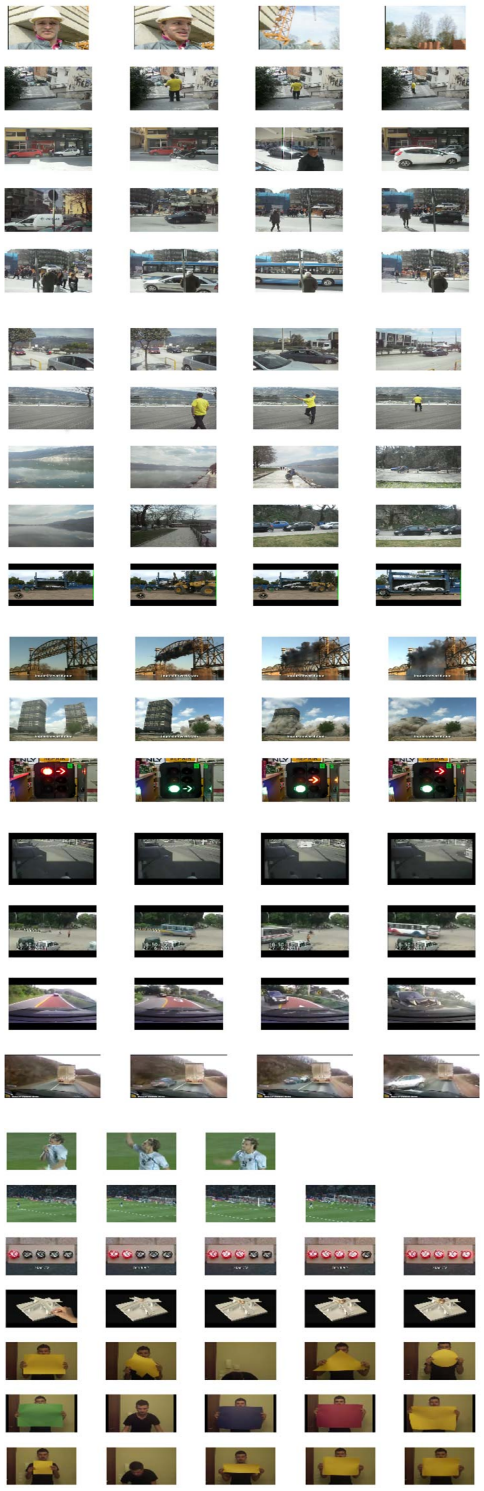

Fig. 3: Selected frames from the video sequences of the dataset.

$$m = \frac{1}{M}\sum_{i=1}^{M}\frac{F_i}{GT_i}. \quad (8)$$

D. Experimental Results

In tables II and III we present comparative results using Mean Value (Eq.8) on ground truth assessments "GT1" and 'GT2", respectively. Each of the six image descriptors (see section II) was also tested in a single-view experiment, using the methods proposed in [3] and [15]. Performance results of these experiments are presented as "HSV1D", "HSV3D", "CEN", "WAV", "SIFT20" and "SIFT50". Eight pairs of image descriptors were tested in the multi-view experiment. Experiment "SP-AVERAGE" corresponds to the solution using the spectral clustering algorithm, when the similarity matrix is built as an unweighted sum of the two individual view kernels. Experiment "SP-MULTIVIEW" corresponds to the solution using the spectral clustering algorithm, when the similarity matrix is built as a weighted sum of the individual view kernels, with weights determined by the weighted multi-view CMM algorithm. In another experiment we carried out, instead of employing all available frames of a video sequence, we sampled every five frames in order to reduce the implementation time.

In must be noted, that we use only two descriptors in each experiment. In future work, we plan to employ more descriptors at the same time. However, the use of many descriptors may increase the computational cost of the method, thus making it inefficient. In other words, a tradeoff between speed and accuracy must be set in such an approach.

TABLE II: Comparative results using Mean Value (m in (%)) on ground truth assessment "GT1", for both sampling rates (Sr = 1 and Sr = 5).

Sampling Rate	Sr = 1	Sr = 5	Sr = 1	Sr = 5
Descriptors	**Method in [3]**		**Method in [15]**	
HSV1D	63.08	65.30	61.90	68.99
HSV3D	64.52	65.88	69.28	70.08
SIFT20	61.96	62.57	71.28	68.29
SIFT50	61.16	65.78	69.65	67.86
CEN	62.98	62.55	68.41	68.26
WAV	66.03	64.30	66.47	69.49
	SP-AVERAGE		**SP-MULTIVIEW**	
HSV1D - SIFT20	73.17	70.76	**78.82**	**80.26**
HSV1D - SIFT50	72.70	72.61	**79.44**	**79.65**
HSV3D - SIFT20	74.25	72.92	**83.26**	**79.09**
HSV3D - SIFT50	73.94	74.03	**83.88**	**82.18**
HSV1D - CEN	70.98	68.91	**83.26**	**81.50**
HSV1D - WAV	69.74	71.25	**81.72**	**82.36**
HSV3D - CEN	69.74	68.91	**80.79**	**80.45**
HSV3D - WAV	70.42	72.92	**82.03**	**80.76**

It can be observed that the proposed method achieves the best performance compared to all single-view methods and the method with equal weights regardless of the pair of descriptors employed. Moreover, even for different ground truth assessments, the proposed method provides the best results, indicating that the number of clusters (key-frames) does not affect the performance. Furthermore, testing the proposed method on the 20% of each video sequence (sampling every five frames) yields very good results.

Note that there is no single-view descriptor surpassing the other single-view descriptors. This is also supported by the results in tables IV and V where we present the average of the weights assigned to the descriptors by the weighted multi-view CMM algorithm for both ground truth assessments and for both sampling rates.

TABLE III: Comparative results using Mean Value (m in (%)) on ground truth assessment "GT2", for both sampling rates (Sr = 1 and Sr = 5).

Sampling Rate	Sr = 1	Sr = 5	Sr = 1	Sr = 5
Descriptors	**Method in [3]**		**Method in [15]**	
HSV1D	60.43	57.41	68.66	66.48
HSV3D	61.08	62.66	69.40	69.04
SIFT20	57.24	61.71	64.47	64.75
SIFT50	56.51	61.18	64.27	65.52
CEN	57.94	56.92	68.86	68.46
WAV	59.26	58.13	65.99	64.97
	SP-AVERAGE		**SP-MULTIVIEW**	
HSV1D - SIFT20	67.40	68.46	**76.72**	**76.02**
HSV1D - SIFT50	71.17	68.52	**77.83**	**76.17**
HSV3D - SIFT20	70.21	67.10	**78.94**	**79.38**
HSV3D - SIFT50	69.59	71.23	**81.41**	**80.06**
HSV1D - CEN	68.91	69.44	**78.91**	**79.10**
HSV1D - WAV	69.32	68.73	**80.34**	**77.44**
HSV3D - CEN	67.43	69.51	**78.76**	**80.49**
HSV3D - WAV	64.32	70.43	**79.37**	**79.07**

TABLE IV: Average values of weights (π^1 and π^2) assigned to the views (V_1 and V_2) by the Weighted Multi-View CMM, for both sampling rates (Sr = 1 and Sr = 5) on ground truth assessment "GT1".

Sampling Rate		Sr = 1		Sr = 5	
V_1	V_2	π^1	π^2	π^1	π^2
HSV1D	**SIFT20**	0.90	0.10	0.79	0.21
HSV1D	**SIFT50**	0.93	0.07	0.85	0.15
HSV3D	**SIFT20**	0.88	0.12	0.80	0.20
HSV3D	**SIFT50**	0.93	0.07	0.84	0.16
HSV1D	**CEN**	0.58	0.42	0.52	0.48
HSV1D	**WAV**	0.72	0.28	0.69	0.31
HSV3D	**CEN**	0.53	0.47	0.49	0.51
HSV3D	**WAV**	0.72	0.28	0.65	0.35

TABLE V: Average values of weights (π^1 and π^2) assigned to the descriptors (V_1 and V_2) by the Weighted Multi-View CMM, for both sampling rates (Sr = 1 and Sr = 5) on ground truth assessment "GT2".

Sampling Rate		Sr = 1		Sr = 5	
V_1	V_2	π^1	π^2	π^1	π^2
HSV1D	**SIFT20**	0.90	0.10	0.79	0.21
HSV1D	**SIFT50**	0.93	0.07	0.85	0.15
HSV3D	**SIFT20**	0.88	0.12	0.80	0.20
HSV3D	**SIFT50**	0.93	0.07	0.83	0.17
HSV1D	**CEN**	0.57	0.43	0.52	0.48
HSV1D	**WAV**	0.71	0.29	0.69	0.31
HSV3D	**CEN**	0.53	0.47	0.47	0.53
HSV3D	**WAV**	0.71	0.29	0.64	0.36

V. Conclusions

In this paper an efficient method for key-frame extraction is proposed. Pairs of different image descriptors (views) are employed to capture different aspects of video frames, due to large variations observed in the visual content of videos. A weighted multi-view CMM clustering algorithm is employed to assign weights to each view. Then, a similarity matrix is built using these weights, as a weighted sum of the individual

view kernels. This similarity matrix serves as an input to a spectral clustering algorithm that clusters the frames of a shot into groups, from which the key-frames are extracted. Performance results on several video sequences indicate that our method efficiently summarizes video shots regardless of the visual content of the video and the pairs of image descriptors employed. In future work, we plan to employ more descriptors at the same time by setting a tradeoff between speed and accuracy, since the use of many descriptors, as already mentioned, may increase the computational cost of the method. Moreover, we plan to experiment using additional videos and image descriptors, and we will try to improve the performance of our method using visual vocabularies of different size as well as other types of kernels.

ACKNOWLEDGMENT

The work described in this paper is co-financed by the European Regional Development Fund (ERDF) (2007-2013) of the European Union and National Funds (Operational Programme Competitiveness and Entrepreneurship (OPCE II), ROP ATTICA), under the Action "SYNERGASIA (COOPERATION) 2009".

REFERENCES

[1] W. Wolf, "Key frame selection by motion analysis," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, ser. ICASSP '96, 1996, pp. 1228–1231.

[2] G. Ciocca and R. Schettini, "An innovative algorithm for key frame extraction in video summarization." *Journal of Real-Time Image Processing*, vol. 1, no. 1, pp. 69–88, 2006.

[3] Y. Zhuang, Y. Rui, T. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," in *International Conference on Image Processing*, vol. 1, oct 1998, pp. 866 –870.

[4] Z. Rasheed and M. Shah, "Detection and representation of scenes in videos." *IEEE Transactions on Multimedia*, vol. 7, no. 6, pp. 1097–1105, 2005.

[5] Z. Cernekova, I. Pitas, and C. Nikou, "Information theory-based shot cut/fade detection and video summarization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 1, pp. 82–91, 2006.

[6] A. Girgensohn and J. S. Boreczky, "Time-constrained keyframe selection technique." *Multimedia Tools Appl.*, vol. 11, no. 3, pp. 347–358, 2000.

[7] G. Tzortzis and C. L. Likas, "Multiple view clustering using a weighted combination of exemplar-based mixture models." *IEEE Transactions on Neural Networks*, vol. 21, no. 12, pp. 1925–1938, 2010.

[8] J. Wu and J. Rehg, "Centrist: A visual descriptor for scene categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1489 –1501, 2011.

[9] E. J. Stollnitz, T. D. DeRose, and D. H. Salesin, "Wavelets for computer graphics: A primer, part 1," *IEEE Computer Graphics and Applications*, vol. 15, no. 3, pp. 76–84, 1995.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[11] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo, "Evaluating bag-of-visual-words representations in scene classification," in *Proceedings of the international workshop on Workshop on multimedia information retrieval*, ser. MIR '07. New York, NY, USA: ACM, 2007, pp. 197–206.

[12] D. Lashkari and P. Golland, "Convex clustering with exemplar-based models," in *Advances in Neural Information Processing Systems*, 2007.

[13] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[14] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems*, 2001, pp. 849–856.

[15] V. Chasanis, A. Likas, and N. Galatsanos, "Scene detection in videos using shot clustering and sequence alignment," *IEEE Transactions on Multimedia*, vol. 11, no. 1, pp. 89–100, January 2009.

[16] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognition*, vol. 36, no. 2, pp. 451 – 461, 2003.