OPEN ACCESS

University of BRISTOL

Mygdalis, V., Iosifidis, A., Tefas, A., & Pitas, I. (2015). Video summarization based on Subclass Support Vector Data Description. In *2014 IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES 2014) : Proceedings of a meeting held 9-12 December 2014, Orlando, Florida, USA* (pp. 183-187). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/CIES.2014.7011849

Peer reviewed version

Link to published version (if available):
10.1109/CIES.2014.7011849

Link to publication record in Explore Bristol Research
PDF-document

This is the author accepted manuscript (AAM). The final published version (version of record) is available online via IEEE at http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7011849. Please refer to any applicable terms of use of the publisher.

University of Bristol - Explore Bristol Research
General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/

Video Summarization based on Subclass Support Vector Data Description

Vasileios Mygdalis, Alexandros Iosifidis, Anastasios Tefas and Ioannis Pitas
Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece
Email: {tefas,pitas}@aiia.csd.auth.gr

***Abstract*—In this paper, we describe a method for video summarization that operates on a video segment level. We formulate this problem as the one of automatic video segment selection based on a learning process that employs salient video segment paradigms. We design an hierarchical learning scheme that consists of two steps. At the first step, an unsupervised process is performed in order to determine salient video segment types. The second step is a supervised learning process that is performed for each of the salient video segment type independently. For the latter case, since only salient training examples are available, the problem is stated as an one-class classification problem. In order to take into account subclass information that may appear in the video segment types, we introduce a novel formulation of the Support Vector Data Description method that exploits subclass information in its optimization process. We evaluate the proposed approach in three Hollywood movies, where the performance of the proposed Subclass SVDD (SSVDD) algorithm is compared with that of related methods. Experimental results show that the adoption of both hierarchical learning and the proposed SSVDD method contribute to the final classification performance.**

I. Introduction

Video summarization techniques develop condensed versions of the original input stream through identification of the most important and pertinent content within the stream [1]. The derived video summaries can be exploited in various applications, like movie (post-)production interactive browsing and searching systems, offering the user the ability to efficiently access video content [2], [3]. Different techniques vary by the type of content used, the performed analysis and the type of video summary representation. Regarding the type the exploited information, it may belong in either generic or domain specific type (e.g., sports, news, movies etc.) as well as information besides the video stream (external information, provided by a user). Objects, events, perceptions and features are extracted by analysing the available modalities (image, sound or text) for abstracting intuitive semantics from the video stream [1]. The abstracted semantic content that needs to be included in the target summary, is commonly represented as a cue of still images (key frames) or a video skim. Key frames are sequences of still images presented in temporal order, that represent the input video stream [4]. This process may involve temporal video segmentation, so that the extracted key frames represent a video segment. The process of key frame extraction is also known as "key-framing", "story-boarding" or "static video summarization". A video skim is a video of shorter length than the input stream, which is known as "dynamic video summarization" [1], [5], [6], [7], [8], [9].

In video summarization techniques with applications to movie post-production, the state-of-the-art approach exploits key frame extraction and video skimming techniques. Usually, long videos containing multiple shots are temporally segmented, either manually or automatically by applying shot detection algorithms. A user attention model is proposed in [6], where visual, audio and textual features are extracted by applying multimodal analysis. A saliency score for each frame is computed and the most salient frames are selected to be the key frames. Video segments around each key frame are concatenated using a fade-in fade-out technique in order to form the summarized video skim. A different approach is proposed in [7]. The video stream is segmented into shots, then face detection and tracking are performed on the segmented video clips. Clustering is performed on the extracted facial images, in order to determine which images belong to the same character. The extracted characters are selected to form a character community network, which forms a graph of interactions between the movie characters. Redundant interactions are excluded from the video skim. Activity information has been exploited mainly in domain-specific applications, like in video surveillance where motion detection techniques are used, in order to create summaries that contain sets of subject actions, like pedestrian walking. Detected actions taking place in different directions and speeds, are fused in a single scene to form a short length video containing as many actions as possible [8], [9].

In this paper, we describe a method for video summarization that operates on a video segment level. Resulting video summaries can be described using the MPEG-7 AVDP profile [10], [11]. We focus our attention in video summarization under unconstrained environments, i.e., the summarization of Hollywood movie shots. Since in movies the activities performed are of great interest for the movie plot, we choose to employ a video segment representation that describes activity information. We employ one of the current state-of-the-art video representations to this end [12]. Related work in 'human action recognition in the wild' [12], [13], [14], [15], has shown that this task corresponds to a very challenging problem due to many reasons, like camera movement, illumination changes, different camera observation angles, etc. After obtaining the video segment representation, we formulate the video summarization problem as the one of automatic video segment selection based on a learning process. In order to learn what properties of a video segment are important for

video summarization, we employ salient video segments to learn the parameters of our learning scheme. Specifically, in the case of Hollywood movies, video shots appearing in movie trailers are characteristic salient video segments, since they have been specially edited in order to catch the viewer attention and, at the same time, to describe the movie plot. In order to appropriately describe the multi-modality of the various video trailer shots, e.g., such shots may depict action scenes, comedy scenes, etc., we design a hierarchical learning scheme. This has been motivated by related work in deep learning [16], where complex classification problems are analysed in multiple levels. The first levels of such schemes are unsupervised and are employed in order to extract semantic information related to the problem to be solved. Finally, a supervised process is employed in order to combine the semantics determined by the previous (unsupervised) levels, along with the labeling information that is available for the training data. In our case, the first step of the adopted learning process is employed in order to roughly determine different video segment types, e.g., action, comedy, etc., by clustering the representations of the movie trailer shots. Subsequently, a supervised learning process is employed.

Since our learning process involves only positive training data, i.e., salient video shots, the final step of our learning scheme is restricted in one-class classification. We employ the Support Vector Data Description (SVDD) approach [17] to this end, in order learn to multiple hyperspheres (one for each video segment type) enclosing the corresponding salient video segments. In order to increase performance, we extend the SVDD classification scheme, so that to incorporate subclass information to its optimization process. This has been motivated by the fact that each video segment type may consist of several subclasses, due to different observation angles, illumination changes, etc. We compare the performance of the proposed Subclass Support Vector Data Description (SSVDD) method with that of SVDD, as well as with that of other one-class classification methods, in video shot summarization of three Hollywood movies. Experimental results show that it is able to outperform other competing approaches.

The remainder of the paper is structured as follows: In section II we describe the video representation method. In sections III and IV we describe the proposed Subclass SVDD (SSVDD) method. Experimental results evaluating each performance on video summarization is described in section V. Finally, conclusions are drawn in Section VI.

II. Video Representation

Let us denote by $\mathcal{V} = \{V_1, \ldots, V_M\}$ a video database consisting of M video segments V_i. V_i may be various takes obtained during movie production, or different video shots appearing in a larger video (e.g., a movie). In the latter case, we automatically segment long videos containing multiple shots in shorter ones, each corresponding to a video shot. We employ the method in [18] to this end. We would like to employ V_i, $i = 1, \ldots, N$ in order to create a summary $\mathcal{S}$ of $\mathcal{V}$, where $\mathcal{S} \subseteq \mathcal{V}$, i.e., a video formed by the most salient video segments V_i. This process is usually noted as video skimming [1].

Let us denote by $\mathcal{X} = \{X_1, \ldots, X_N\}$ another video database that contains N salient video segments X_i. We would like to employ the video segments in $\mathcal{X}$ in order to train a classifier that can determine whether the video segments V_i are salient or not. To this end, we employ the Dense Trajectory-based video description [12] in order to describe the video segments in $\mathcal{X}$ and $\mathcal{V}$. This video description calculates five descriptor types on the trajectory of densely-sampled video frame interest points that are tracked for a number of consecutive video frames. The five descriptor types are: Histogram of Oriented Gradients (HOG), Histogram of Optical Flow (HOF), Motion Boundary Histogram along direction x (MBHx), Motion Boundary Histogram along direction y (MBHy) and the normalized trajectory coordinates (Traj). We employ these video segment descriptions in order to obtain five video segment representations by using the Bag-of-Words model. That is, the descriptors calculated for the training video segments X_i, $i = 1, \ldots, N$ are clustered in order to determine five sets of descriptor prototypes (each for a descriptor type). Subsequently, each of the video segments X_i and V_i are represented by five vectors $\mathbf{x}_i^v$, $\mathbf{v}_i^v$, $v = 1, \ldots, 5$, respectively. In order to fuse the information appearing in different video representations, we combine the video segment representations with kernel methods, as in [12]. That is, we employ the RBF-χ^2 kernel function, where different descriptor types are combined following a multi-channel approach [19]:

$$\mathbf{K}(\mathcal{X}_i, \mathcal{X}_j) = exp\Big(- \sum_v \frac{1}{4A^v} D\left(\mathbf{x}_i^v, \mathbf{x}_j^v\right) \Big), \qquad (1)$$

$D\left(\mathbf{x}_i^v, \mathbf{x}_j^v\right)$ is the χ^2 distance between the BoW-based video representation of $\mathbf{x}_i$ and $\mathbf{x}_j$ with respect to the v-th channel. A^v is the mean value of the χ^2 distances between the training samples for the v-th channel.

After calculating the kernel matrices for the training and test video segments, we would like to calculate a vectorial representation for each of the video segments in $\mathcal{X}$ and $\mathcal{V}$. We apply the kernel Principal Component Analysis [20] to this end, in order to determine two vector sets $\tilde{\mathcal{V}} = \{\mathbf{v}_1, \ldots, \mathbf{v}_M\}$ and $\tilde{\mathcal{X}} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ that represent the video shots in $\mathcal{V}$ and $\mathcal{X}$, respectively.

III. Hierarchical learning scheme

In this Section, we describe in detail the proposed hierarchical learning scheme for video summarization. We assume that the video segments in $\mathcal{X}$ correspond to video shots belonging to K scene types, e.g., action scenes, comedy scenes, etc. Since we expect the actions appearing in different scene types to be different, we group X_i, $i = 1, \ldots, N$ in K groups based on their visual (activity) information. In order to do this automatically, we cluster the vectors $\mathbf{x}_i$ in K clusters without taking into account the movie type labels that are available for the movie trailer shots in the training phase. This is due to the fact that we expect a movie trailer to contain shots belonging to several scene types. We apply K-Means [21] to this end and

determine K video segment groups $\mathcal{C}_k$, where $\bigcup_{k=1}^{K} \mathcal{C}_k = \mathcal{X}$. After the determination of the K video segment groups $\mathcal{C}_k$, we train K one-class classifiers (one for each video segment group $\mathcal{C}_k$). We employ Support Vector Data Description [17] to this end, that aims at determining a hypersphere enclosing most of (possibly all) the vectors $\mathbf{x}_i$ belonging to video segment group $\mathcal{C}_k$. However, such an approach does not take into account the subclass information that may appear in video segment group $\mathcal{C}_k$. In video summarization in unconstrained environments it is expected that scene types will be multimodal, due to camera movement, illumination changes, different camera observation angles, etc. We will describe an extension of the SVDD method that is able to incorporate subclass information in its training process in the next Section.

After training the K one-class classifiers, each of the video segments in $\mathcal{V}$, represented by the corresponding vector $\mathbf{v}_i$, is introduced to the K one-class classifiers and K responses $o_i^k,\ k = 1, \ldots, K$. Finally, the video segment i is assigned to the classifier that provides the maximal response, i.e.:

$$o_i = \max_k o_i^k,\ \ k = 1, \ldots, K. \tag{2}$$

IV. Subclass Support Vector Data Description

In this Section, we describe the proposed Subclass SVDD method. For notation convenience, let us denote by $\mathbf{y}_i^k,\ i = 1, \ldots, |\mathcal{C}_k|$ the i-th training vector belonging to the k-th video segment group. Let us assume that the vectors $\mathbf{y}_i^k$ form c_k subclasses. The variance of $\mathcal{C}_k$ with respect to the corresponding subclass mean vectors $\mathbf{v}_j^k,\ j = 1, \ldots, c_k$ is given by:

$$\mathbf{S}_k = \sum_{j=1}^{c_k} \beta_{ij}^k \left(\mathbf{y}_i^k - \mathbf{v}_j^k\right)\left(\mathbf{y}_i^k - \mathbf{v}_j^k\right)^T, \tag{3}$$

where β_{ij}^k is an index denoting if $\mathbf{y}_i^k$ belongs to subclass j. The number of subclasses c_k can either be set manually based on the properties of the problem at hand, or be automatically determined by applying k-fold (e.g., 5-fold) cross-validation.

The minimum bounding hypersphere that encloses most of (possibly all) the vectors $\mathbf{y}_i^k$ and exploits subclass information encoded in $\mathbf{S}_k$ can be determined by the corresponding center $\mathbf{u}_k$ and radius R calculated by optimizing for:

$$\min_{R,\xi,\mathbf{a}} \quad R^2 + c\sum_{i}^{|\mathcal{C}_k|} \xi_i \tag{4}$$

$$s.t.: \quad \|\mathbf{S}_k^{-\frac{1}{2}}\mathbf{y}_i^k - \mathbf{u}_k\|_2^2 \leq R^2 + \xi_i, \tag{5}$$

$$\xi_i \geq 0, \quad i = 1, ..., |\mathcal{C}_k|, \tag{6}$$

where ξ_i are the slack variables and c is a parameter denoting the importance of the error in the optimization problem.

Based on the Karush-Kuhn-Tucker (KKT) theorem [22], the above described optimization problem can be solved by finding the saddle point of the Lagrangian:

$$\begin{aligned}\mathcal{L}\left(R, \xi_i, \boldsymbol{\alpha}, \boldsymbol{\beta}\right) &= R^2 + c\sum_{i}^{|\mathcal{C}_k|} \xi_i - \sum_{i=1}^{|\mathcal{C}_k|} \beta_i \xi_i \\ &- \sum_{i=1}^{|\mathcal{C}_k|} \alpha_i \left(R^2 + \xi_i - \|\mathbf{S}_k^{-\frac{1}{2}}\mathbf{y}_i^k - \mathbf{u}_k\|_2^2\right)\end{aligned} \tag{7}$$

leading to the following optimality conditions:

$$\frac{\vartheta \mathcal{L}}{\vartheta \mathbf{u}_k} = 0 \ \ \Rightarrow \ \ \sum_{i=1}^{|\mathcal{C}_k|} \alpha_i \mathbf{u}_k = \sum_{i=1}^{|\mathcal{C}_k|} \alpha_i \mathbf{S}_k^{-\frac{1}{2}} \mathbf{y}_i^k, \tag{8}$$

$$\frac{\vartheta \mathcal{L}}{\vartheta R} = 0 \ \ \Rightarrow \ \ \sum_{i=1}^{|\mathcal{C}_k|} \alpha_i = 1, \tag{9}$$

$$\frac{\vartheta \mathcal{L}}{\vartheta \xi_i} = 0 \ \ \Rightarrow \ \ \alpha_i = c - \beta_i. \tag{10}$$

From (8), (9) the center $\mathbf{u}_k$ is given by:

$$\mathbf{u}_k = \sum_{i=1}^{|\mathcal{C}_k|} \alpha_i \mathbf{S}_k^{-\frac{1}{2}} \mathbf{y}_i^k \tag{11}$$

Replacing (8),(9) and (10) in $\mathcal{L}\left(R, \xi_i, \boldsymbol{\alpha}, \boldsymbol{\beta}\right)$ and using the KKT conditions, the optimization problem in (4) can be reformulated to its dual form:

$$\max_{\boldsymbol{\alpha}} \ \sum_{j=1}^{|\mathcal{C}_k|} \alpha_i \mathbf{y}_i^{k\,T} \mathbf{S}^{-1} \mathbf{y}_i^k - \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j \mathbf{y}_i^{k\,T} \mathbf{S}^{-1} \mathbf{y}_i^k, \tag{12}$$

subject to $0 \leq \alpha_i \leq c$ and $\sum_i \alpha_i = 1$.

After solving (12), the radius R is the one that:

$$R^2 = \{min\ \|\mathbf{S}_k^{-\frac{1}{2}}\mathbf{y}_i^k - \mathbf{u}_k\|_2^2,\ \ \mathbf{y}_i^k\ is\ a\ SV\}. \tag{13}$$

A test vector $\mathbf{v}_i$ can be introduced to the classifier and its response is given by:

$$f(\mathbf{v}_i) = R - \|\mathbf{S}_k^{-\frac{1}{2}}\mathbf{v}_i - \mathbf{u}_k\|_2. \tag{14}$$

By observing (12) it can be seen that the solution of the proposed SSVDD classifier is similar to that of SVDD. In order to exploit standard SVDD implementations [23], we can use an approach similar to the one proposed in [24]. We can apply eigenanalysis on the matrix $\mathbf{S}_k$ in order to decompose it to $\mathbf{S}_k = \mathbf{V}_k \boldsymbol{\Sigma}_k \mathbf{V}_k^T$, where $\mathbf{V}_k$ is an orthonormal matrix that contains the eigenvectors of $\mathbf{S}_k$ and $\boldsymbol{\Sigma}_k$ is a diagonal matrix containing the eigenvalues of $\mathbf{S}_k$. Then, we can employ the matrix $\mathbf{P}_k = \mathbf{V}_k \boldsymbol{\Sigma}_k^{-\frac{1}{2}}$ in order to map the original training vectors $\mathbf{y}_i^k,\ i = 1, \ldots, |\mathcal{C}_k|$ to vectors $\mathbf{z}_i^k$ by:

$$\mathbf{z}_i^k = \mathbf{P}^T \mathbf{y}_i^k. \tag{15}$$

It can be shown that:

$$\mathbf{z}_i^{k\,T} \mathbf{z}_j^k = \mathbf{y}_i^{k\,T} \mathbf{P}_k \mathbf{P}_k^T \mathbf{y}_j^k = \mathbf{y}_i \mathbf{S}_k^{-1} \mathbf{y}_j^k. \tag{16}$$

Thus, by applying SVDD on the vectors $\mathbf{z}_i^k$ corresponds to applying the proposed SSVDD with the optimization problem given in (4). In the case where $\mathbf{S}_k$ is singular, one can choose to keep fewer eigenvectors (the ones corresponding to the non-zero eigenvalues) for $\mathbf{z}_i^k$ calculation.

V. Experimental Results

In this Section, we present experiments conducted in order to evaluate the proposed video summarization method and the performance of the proposed SSVDD classifier. We have employed three Hollywood movies of full length, belonging to action, adventure and drama categories, respectively. In order to train the one-class classifiers we have employed thirty Hollywood movie trailers belonging to action, adventure, comedy, thriller and drama categories. The trailers of the three (test) movies are not included in the training set.

Here we should note that usually, dynamic video summarization techniques are evaluated based on qualitative criteria [6], [7], e.g., by calculating criteria like the 'informativeness', or the 'enjoyability' based on the ratings provided by users for the entire video summary. However, such criteria are too subjective. In order to perform quantitative evaluation of the performance of each classifier in video summarization, we employ the trailers of the three (test) movies and manually create ground truth labels denoting whether a video segment (shot) of each movie has been employed in order to form the trailer, or not. We introduce the test vectors $\mathbf{v}_i$, $i = 1, \ldots, M$ to the classifiers trained on the video shots of the training movie trailers and obtain their responses. Subsequently, we keep the video shots corresponding to the $L = pN$ maximal response values, where $0 < p < 1$, and calculate the percentage of the movie trailer belonging to the created video summary.

We set the dimensionality of the BoW-based video segment representations $\mathbf{x}_i^v$, $\mathbf{v}_i^v$ equal to 4000 and test for a number of video segments $\mathcal{C}_k$, using values $K = 1, \ldots, 10$. We employ the training vectors belonging to each of the video segment groups $\mathcal{C}_k$ in order to train the proposed SSVDD classifier formed by c_k subclasses, using the values $c_k = 5, \ldots, 15$. For comparison reasons, we also train the standard SVDD classifier [17] and a variant of SVDD exploiting the variance of the training data [25] (noted as MCSVDD hereafter). In addition, we test the performance of each classifier in the case of a non-hierarchical learning.

The performance of each classifier for different values of p in the case of non-hierarchical learning (averaged over three movies) is illustrated Table I. As can be seen in this Table, by exploiting the variance of the training data, the MCVSVDD classifier outperforms SVDD in all the cases. The exploitation of subclass information further enhances classification performance and the proposed SSVDD algorithm provides the best performance in all the cases presented in Table I. This can be explained by the fact that the results reported in this Table have been obtained by using only one one-class classifier. Thus, by exploiting the subclass information appearing in different movie trailer shot types enhanced performance is obtained.

In our second set of experiments, we have employed the proposed hierarchical classification scheme. The performance of each classifier for different values of p in this case (averaged over three movies) is illustrated Table II. Similar to the non-hierarchical learning case, the proposed SSVDD classifier outperforms both SVDD and MCVSVDD choices in all the cases. In addition, by comparing the classification rates appearing in Tables I and II, it can be seen that the adoption of a hierarchical learning approach enhances performance in most cases.

TABLE I
Performance (%) for non-hierarchical learning

	0.1	0.2	0.3	0.4	0.5
SVDD [17]	13.69	25.48	33.91	44.33	52.97
MCSVDD [25]	18.38	30.79	42.33	52.21	57.86
SSVDD	18.66	32.03	44.68	56.62	67.06

TABLE II
Performance (%) for hierarchical learning

	0.1	0.2	0.3	0.4	0.5
SVDD [17]	15.61	26.73	34.86	45.22	52.19
MCVSVDD [25]	17.72	31.39	43.3	52.78	63.91
SSVDD	19.73	33.08	45.31	57.14	66.83

In an attempt to explain the classification rates obtained in the above described experiments, we have created the summaries of the three movies by exploiting the order of the video segments in $\mathcal{V}$. We have observed that video segments forming the different video summaries are quite similar to each other in terms of video saliency. This means that the proposed approach can be employed in order to assign saliency scores to the various video segments in terms of saliency and produce video segment suggestions that can be used in order to accelerate video post-processing, or to provide a good summarization of a video in terms of saliency.

VI. Conclusion

In this paper, we described a method for video summarization by exploiting an activity-based video segment description. We have formulated the problem as the one of automatic video segment selection based on a hierarchical learning process exploiting salient video segment paradigms. In order to take into account subclass information that may appear in different video segment types, e.g., shots depicting action schemes, we have proposed a Support Vector Data Description method that exploits subclass information in its optimization process. We evaluated the proposed approach in three Hollywood movies, where it has been shown that the adoption of a hierarchical learning approach enhances performance. In addition, experimental results show that the proposed SSVDD method is able to outperform other related methods. Future work could include parallel implementation of this method, since the multiple classifiers used are independent with each other.

Acknowledgement

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement number 316564 (IMPART).

REFERENCES

[1] A. G. Money and H. Agius, "Video summarisation: A conceptual framework and survey of the state of the art," *Journal of Visual Communication & Image Representation*, vol. 19, pp. 121–143, 2008.

[2] Y. Li, S. Lee, C. Yeh, and C. Kuo, "Semantic retrieval of multimedia," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 79–89, 2006.

[3] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: state of the art and challenges," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 2, no. 1, pp. 1–19, 2006.

[4] P. Kelm, S. Schmiedeke, and T. Sikora, "Feature-based video key frame extraction for low quality video sequences," in *Image Analysis for Multimedia Interactive Services, 2009. WIAMIS'09. 10th Workshop on*. IEEE, 2009, pp. 25–28.

[5] H. Katti, K. Yadati, M. Kankanhalli, and C. Tat-Seng, "Affective video summarization and story board generation using pupillary dilation and eye gaze," in *Multimedia (ISM), 2011 IEEE International Symposium on*. IEEE, 2011, pp. 319–326.

[6] G. Evangelopoulos, A. Zlatintsi, A. Potamianos, P. Maragos, K. Rapantzikos, G. Skoumas, and Y. Avrithis, "Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention," *IEEE Transactions on Multimedia*, vol. 15, no. 7, pp. 1553–1568, 2013.

[7] C. M. Tsai, L. W. Kang, C. W. Lin, and W. Lin, "Scene-based movie summarization via role-community networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 11, pp. 1927–1940, 2013.

[8] W. Fu, J. Wang, L. Gui, H. Lu, and S. Ma, "Online video synopsis of structured motion," *Neurocomputing*, vol. 135, pp. 155–162, 2014.

[9] K. Streib and J. Davis, "Summarizing high-level scene behavior," *Machine Vision and Applications*, vol. 25, no. 1, pp. 229–244, 2014.

[10] B. S. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7: multimedia content description interface*. John Wiley & Sons, 2002, vol. 1.

[11] I. T. .-A. 1:2012, "Audiovisual description profile (avdp) schema," 2012.

[12] H. Wang, A. Klaser, C. Schmid, and C. L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, 2012.

[13] A. Iosifidis, A. Tefas, and I. Pitas, "Discriminant bag of words based representation for human action recognition," *Pattern Recognition Letters*, 2014.

[14] ——, "Regularized extreme learning machine for multi-view semi-supervised action recognition," *Neurocomputing*, 2014.

[15] ——, "Minimum class variance extreme learning machine for human action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 11, pp. 1968–1979, 2013.

[16] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *arXiv:1206.5538v3*, 2014.

[17] D. M. Tax and P. Duin, "Support vector data description," *Machine Learning*, vol. 54, pp. 45–66, 2004.

[18] 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.

[19] J. Zhang, M. Marszalek, M. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *International Journal of Computer Vision*, vol. 73, no. 2, pp. 213–238, 2007.

[20] B. Schlkopf, A. Smola, and K. R. Moller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, pp. 1299–1319, 1998.

[21] R. O. Duda, P. E. Hart, and D. G. Stork, "Pattern classification, 2nd ed," 2000, wiley-Interscience.

[22] R. Fletcher, "Practical methods of optimization," *Wiley*, 1987.

[23] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, 2011.

[24] T. S. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," *Advances in Neural Information Processing*, 1999.

[25] S. Zafeiriou and N. Laskaris, "On the improvement of support vector techniques for clustering by means of whitening transform," *IEEE Signal Processing Letters*, vol. 15, pp. 198–201, 2008.