Computers and Electrical Engineering 40 (2014) 247–256

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Hierarchical representation of video sequences for annotation ☆

Engin Mendi

Department of Computer Engineering, KTO Karatay University, Konya, Turkey

ARTICLE INFO

Article history:
Available online 6 April 2014

ABSTRACT

Video annotation is an important issue in video content management systems. Rapid growth of the digital video data has created a need for efficient and reasonable mechanisms that can ease the annotation process. In this paper, we propose a novel hierarchical clustering based system for video annotation. The proposed system generates a top-down hierarchy of the video streams using hierarchical *k*-means clustering. A tree-based structure is produced by dividing the video recursively into sub-groups, each of which consists of similar content. Based on the visual features, each node of the tree is partitioned into its children using *k*-means clustering. Each sub-group is then represented by its key frame, which is selected as the closest frame to the centroids of the corresponding cluster, and then can be displayed at the higher level of the hierarchy. The experiments show that very good hierarchical view of the video sequences can be created for annotation in terms of efficiency.

1. Introduction

In recent years, the proliferation of digital videos has led to a rapid expansion in the availability and the amount of video data. The growth of archived video material has made indexing and annotating the information crucial. Since manual indexing and annotating the video material are both computationally expensive and time consuming, automated systems that can efficiently perform these processes are needed [1].

Multimedia content classification refers to the computerized apprehension of the semantic meanings of a multimedia file or document. With the increase in digital video contents, efficient techniques for classification of videos according to their contents have become more important. Applications such as digital libraries, e-Learning, video-on demand, digital video broadcast and interactive TV generate and use large collections of video data. For an effective use of these video data, all digital contents must be classified based on their categories [2].

The traditional solution to the problem of searching and finding large numbers of images and videos from a database is to annotate each image and video manually with keywords or captions and then search on those captions or keywords using a conventional text search engine [3]. Semantic annotation of video content is an important step towards more efficient retrieval and browsing of visual media. The goal of automatic video annotation is to assign relevant captions or other descriptive text to the content of the shot or key frame that reflects its visual content. It helps to reduce the growing amount of cluttered video data by categorizing them. Automatic discovery and organization of descriptive captions can be used to build a story structure of a video stream, which brings ease and effectiveness to the data access by the user. Utilizing the content of key frames to label a set of semantic descriptions can guide the organization of video for a high-level representation. One of the

☆ Reviews processed and recommended for publication to Editor-in-Chief by Deputy Editor Dr. Ferat Sahin.

E-mail address: engin.mendi@karatay.edu.tr

http://dx.doi.org/10.1016/j.compeleceng.2014.03.001
0045-7906/

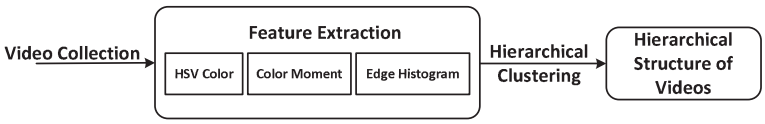


Fig. 1. Block diagram of proposed technique.

most typical approaches for automatic annotation is applying machine learning techniques [3–5]. The methodology of these techniques can be summarized as: (i) computing the feature vectors characterizing low level visual features of the key frames such as color, texture or oriented edges, (ii) learning a model and annotating a set of test key frames, and (iii) automatically applying annotations to new key frames. Although machine learning methods can be used in annotation of the videos, these techniques have some limitations due to the lack of large-scale labeled training data set.

The majority of existing approaches for image and video annotation use supervised learning techniques. Xiang et al. [6] presented a genarative model using Markov Random Fields. A new potential function is used for optimal parameter estimation and model inference. In [7], a contextual learning model is introduced for recognition and categorization at the object level using conditional random field. Qi et al. [8] proposed a Correlative Multi-Label framework that integrates modeling of individual concept and the conceptual correlations. In [9], a keypoint-based semantic concept detection framework based on bag-of-visual-words (BoW) is proposed to annotate images and video shots. Various representation choices with respect to dimension, weighting, selection, etc. are investigated in order to evaluate the performance of BoW in semantic annotation. Cusano et al. [10] extended Support Vector Machine to multi-class classification in image annotation. In [11], supervised multiclass labeling is introduced for image annotation and retrieval. Images are represented by localized features, and a Gaussian mixture model is employed for training input vector. Mixtures from the images annotated with a semantic label are then determined with expectation-maximization algorithm and pooled into a density estimate for the corresponding class. Finally, based on the class densities, semantic retrieval and annotation are implemented using minimum probability of error criterion. Another supervised machine learning approach to automatic annotation is mixture hierarchy model proposed by Carneiro and Vasconcelos [12]. This includes a multiple instance learning between captions and image features, allowing the estimation of probability distribution. Wang et al. [13] proposed a graph-based technique using bi-relational graph model that connects traditional data graph and the label graph with a bipartite graph. Each class and its labeled images are considered as a semantic group, and random walk is applied to the bi-relational graph to generate class-to-image and class-to-class relevances.

In this paper, we present a hierarchical clustering based system for annotation of video contents. Fig. 1 shows the block diagram of proposed technique. The system creates recursive hierarchy adopting partition clustering at each level of the hierarchy. With the clustering processes, the features of video frames are used to cluster the shots into classes, each of which consists of similar content. Using appropriate visual features, a tree-based story structure is built to organize the video data. Similar contents can be viewed easily at each level of the top-down hierarchy without any need of a priori models. As a result, user can browse all video sequences by moving to different levels of hierarchy, which can make the annotation process easy. The issue of accessibility is a great challenge in application of videos such as information systems and video libraries. Unfortunately the quality of accessibility service is not same with respect to the infrastructure around the world. Providing ease in accessing the certain relevant segments of the data would be very beneficial in supporting the knowledge enhancement needs of the user by allowing better understanding of video content. The technique presented in this paper include development of hierarchical clustering based framework that will ease video annotation process. This could help bringing quality service to efficiently manage video data in terms of cost, time, and reliability.

The rest of this paper is arranged as follows. In the next section, video content representation is discussed. In Section 3, we provide feature extraction details. In Section 4, hierarchical browsing of key frames is discussed. Experimental results are given in Section 5. Finally, the conclusions of this paper are summarized in Section 6.

2. Video content representation

Fig. 2 illustrates an overview of video content organization. Content-based video representation requires several video processing steps. Temporal video segmentation is the first step towards automatic indexing and annotation of video streams. It includes shot boundary detection and key frame extraction. Shot boundary detection aims partitioning a video into shorter segments. Key frame extraction provides a pictorial summarization and representation of a video sequence. In feature extraction process, several image features representing the visual content of the key frames are extracted. Finally, extracted features are organized automatically using clustering and annotated video sequence can be hierarchically viewed.

3. Feature extraction

In this section, we present visual features used in video representation. The feature vector consists of three components: color histogram, color moment and edge histogram.

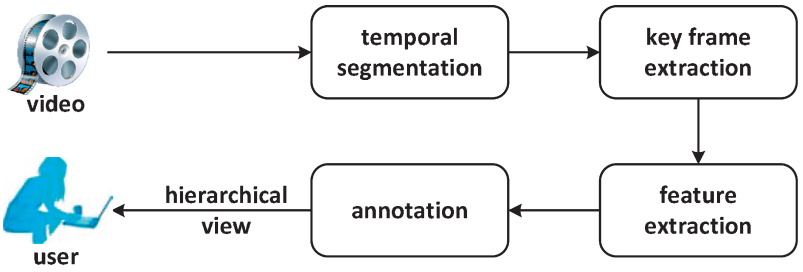


Fig. 2. An overview of the video content organization.

3.1. Color histogram

RGB color spaces are the additive combination of red, green and blue lights and usually used by most of the digital devices [14]. However, features extracted in hue-saturation value (HSV) space in which colors are described by hue, saturation and value can capture the distinct characteristics of computer graphics better than of RGB space [1]. In addition, HSV color space is more convenient for human perception. Therefore, RGB color space is converted to HSV space as follows [15]:

$$\begin{aligned} \mathrm{H} &= \begin{cases} 60\left[\frac{\mathrm{G}-\mathrm{B}}{\max(\mathrm{R},\mathrm{G},\mathrm{B})-\min(\mathrm{R},\mathrm{G},\mathrm{B})}\right] + 360 & \text{if } \max(\mathrm{R},\mathrm{G},\mathrm{B}) = \mathrm{R} \\ 60\left[\frac{\mathrm{B}-\mathrm{R}}{\max(\mathrm{R},\mathrm{G},\mathrm{B})-\min(\mathrm{R},\mathrm{G},\mathrm{B})}\right] + 120 & \text{if } \max(\mathrm{R},\mathrm{G},\mathrm{B}) = \mathrm{G} \\ 60\left[\frac{\mathrm{R}-\mathrm{G}}{\max(\mathrm{R},\mathrm{G},\mathrm{B})-\min(\mathrm{R},\mathrm{G},\mathrm{B})}\right] + 240 & \text{if } \max(\mathrm{R},\mathrm{G},\mathrm{B}) = \mathrm{B} \\ \text{not defined} & \text{if } \max(\mathrm{R},\mathrm{G},\mathrm{B}) = 0 \end{cases} \\ \mathrm{S} &= \begin{cases} \frac{\max(\mathrm{R},\mathrm{G},\mathrm{B})-\min(\mathrm{R},\mathrm{G},\mathrm{B})}{\max(\mathrm{R},\mathrm{G},\mathrm{B})} & \text{if } \max(\mathrm{R},\mathrm{G},\mathrm{B}) \neq 0 \\ 0 & \text{if } \max(\mathrm{R},\mathrm{G},\mathrm{B}) = 0 \end{cases} \\ \mathrm{V} &= \max(\mathrm{R},\mathrm{G},\mathrm{B}) \end{aligned} \tag{1}$$

A color quantization is done using 128 colors (8 levels for hue channel, 4 levels for saturation channel and 4 levels for value channel) reduce the overall computation effort while preserving the colors. Since human color perception is more tolerant to saturation and value deviations, the quantization should preserve more hue levels when compared to saturation and value [16,1]. Finally, HSV histogram of a frame is computed as:

$$H(k) = \sum_{i=1}^{N} h_k(i) \tag{2}$$

where h_k is the color histogram of the kth frame of the video sequence with N bins.

3.2. Color moment

Color moments provide information about color distribution in video frames. In probability theory and statistics, it is known that a probability function can be characterized by its moments. For instance, the mean of a probability distribution function is simply the first moment of the probability function and the variance is related to the second moment. Similarly, if the color distribution of a video frame is interpreted as a probability function, then the moments characterized by the color distribution can be used as features to identify that video frame based on color.

Stricker and Orengo [17] proposed three moments for the color distribution. The first moment is the mean which is the average color value in the video frame. The second and the third moment are standard deviation and skewness that are drawn from the color values.

Assuming all video frames contain N pixels, if the value of i-th HSV color channel at the j-th video frame pixel is p_{ij}, then mean-E_i, standard deviation-σ_i and skewness-s_i can be defined as:

$$\begin{aligned} E_i &= \sum_{j=1}^{N} \frac{1}{N} p_{ij} \\ \sigma_i &= \sqrt{\left(\frac{1}{N}\sum_{j=1}^{N}(p_{ij} - E_i)^2\right)} \\ s_i &= \sqrt[3]{\left(\frac{1}{N}\sum_{j=1}^{N}(p_{ij} - E_i)^3\right)} \end{aligned} \tag{3}$$

3.3. Edge histogram

Edges play an important role in image perception. Edge histograms capture the shape information in the video frame on the basis of edge directions. Therefore, features of edge histograms are extracted to represent shape attribute. A global edge histogram is built based on the edge information and its orientations contained in the video frame using a Canny edge operator [18]. Then, the histogram is normalized with respect to the total number of edges. This gives a set of features, each of which represents the directional edges at a range of angles.

Fig. 3 shows an example image with its Canny edge image and edge histogram. Each edge of the original image is assigned to different histogram bin corresponding to its orientation.

4. Hierarchical browsing of key frames

Non-sequential video content browsing requires a proper organization. Annotated coherent content in a hierarchical tree-based story structure can be used for fast navigation, selective transmission and indexing of the video sequences. Similar contents can be viewed easily at each level of the top-down hierarchy.

A hierarchical viewing schema of video stream is developed for annotation using hierarchical k-means clustering. Tree-based structure of the video stream is generated grouping the shots into classes, each of which consists of similar content. Each class is represented by its key frame, which can then be displayed at the higher level of the hierarchy. The video frames closest to the cluster centroids are chosen as key frames.

We apply hierarchical k-means clustering to divide the video streams recursively into sub-groups. K-means [19] is one of the most commonly used iterative clustering algorithms. The objective of k-means is partitioning the data set into k clusters by minimizing the distance between the data point and cluster centroid. Our hierarchical k-means approach extends traditional k-means to provide more detail description about the relationship among the video content at different levels.

Fig. 4 illustrates a 3 level hierarchical k-means clustering with tree representation by setting $k = 2$. In the root level, the video data set is split into two child clusters. Then, k-means clustering is applied to each cluster based on the clustering results of the previous level. The process continues until the dataset is divided into single data points or a stopping criterion is satisfied.

(a) Original image. (b) Canny edge image. (c) Edge histogram.

Fig. 3. Example of an edge histogram.

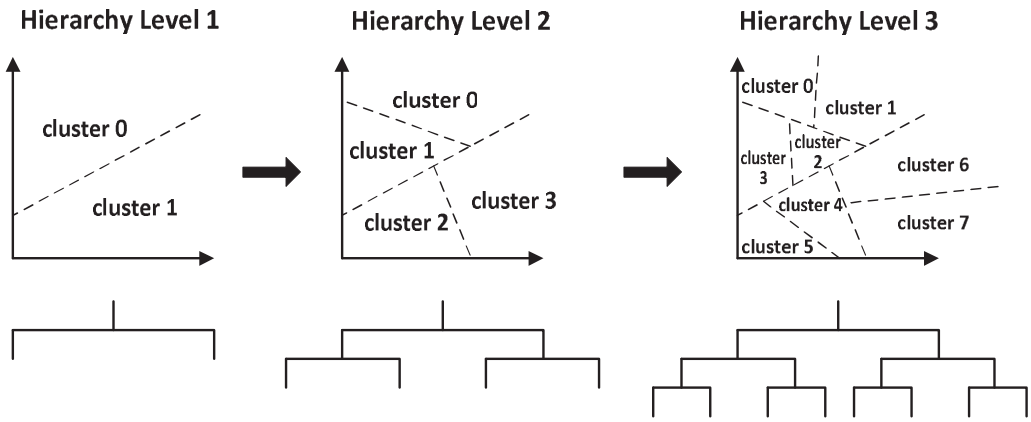


Fig. 4. Illustration of 3 level hierarchical k-means clustering with tree representation by setting $k = 2$.

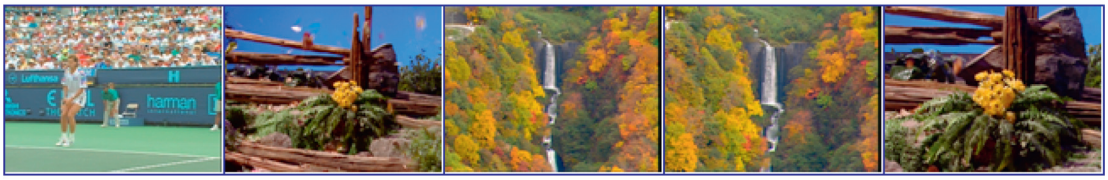
Our hierarchical *k*-means clustering algorithm is described as follows:

1. Set *X* as all input vectors of the video data.
2. Set *k* as the predefined number of clusters.
3. Perform *k*-means on the root level.
4. Record the results of clustering.
5. Choose a child cluster and re-set *k* as the predefined number of child clusters.
6. Perform *k*-means on the input vectors that belong to the chosen cluster.
7. Repeat from step 4 until the stopping criterion is reached, i.e., until the generated cluster comprises 5 or less video frames.

(a) Hierarchy level 1.

(b) Hierarchy level 2 from the third class at level 1 in (a).

(c) Hierarchy level 2 from the fourth class at level 1 in (a).



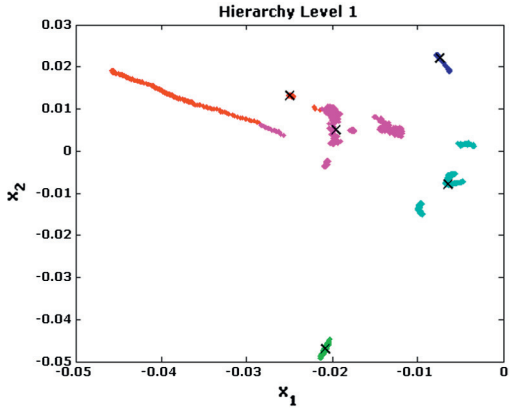
(d) Hierarchy level 2 from the fifth class at level 1 in (a).

(e) Hierarchy level 3 from the first class at level 2 in (d).

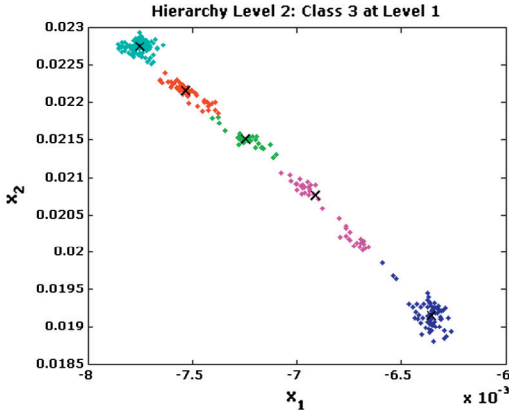
(f) Hierarchy level 3 from the second class at level 2 in (d).

Fig. 5. Hierarchical structure of YUV video sequences.

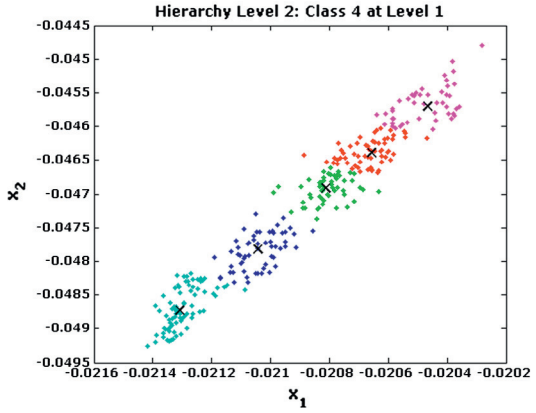
In contrast to *k*-means clustering algorithms depending on the choice for the number of clusters to be searched [20], our method is based on the measure of dissimilarity between groups of observations. The average dissimilarity between the objects to be clustered is:



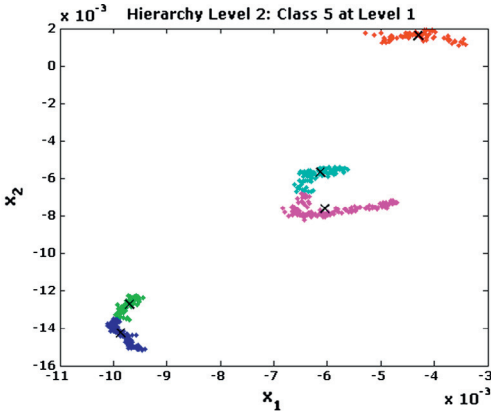
(a) Hierarchy level 1.



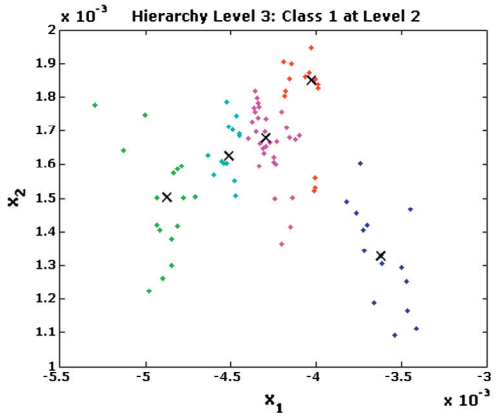
(b) Hierarchy level 2 from the third class at level 1 in (a).



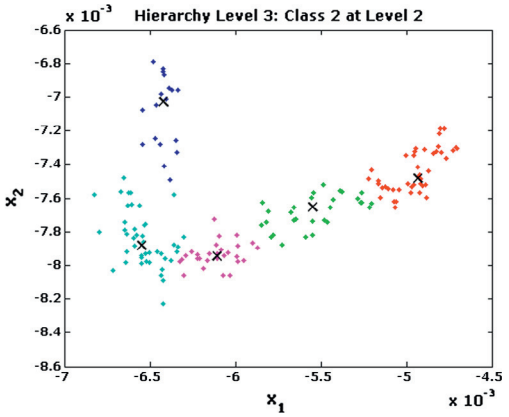
(c) Hierarchy level 2 from the fourth class at level 1 in (a).



(d) Hierarchy level 2 from the fifth class at level 1 in (a).



(e) Hierarchy level 3 from the first class at level 2 in (d).



(f) Hierarchy level 3 from the second class at level 2 in (d).

Fig. 6. Clustering plots of YUV video sequences.

$$d(Cluster_i, Cluster_j) = \frac{\sum_{O_i \in Cluster_i, O_j \in Cluster_j} d(O_i, O_j)}{|Cluster_i| . |Cluster_j|} \quad (4)$$

where O_i and O_j are two sets of feature vectors. Hierarchical representations are generated in which the clusters at each level of the hierarchy are created by merging clusters at the next lower level. Each cluster contains a single observation at the lowest level, only one cluster at the highest level. Proposed video annotation scheme using hierarchical clustering allows similar contents to be viewed easily at each level of the top-down hierarchy without any need of a priori models.

Both k-means and operations concerning tree have linear time complexity. Thus, the computational cost for building the tree structure using proposed hierarchical k-means clustering is $O(Lnkl)$ where L is the number of levels in the tree, n is the number of patterns, k is the number of clusters, and l is the number of iterations. Moreover, our approach is also low cost in terms of space complexity, that is $O((L + k)n)$. This makes the proposed technique quite efficient even for large problems.

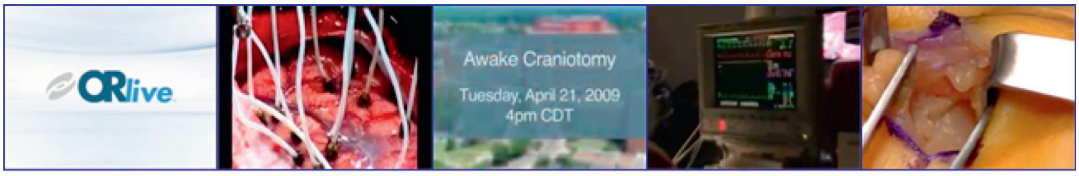
5. Experimental results

In this section, we present the experimental results using 2 different video sequences. The first is a collection of YUV videos with miscellaneous content and the second is medical videos.

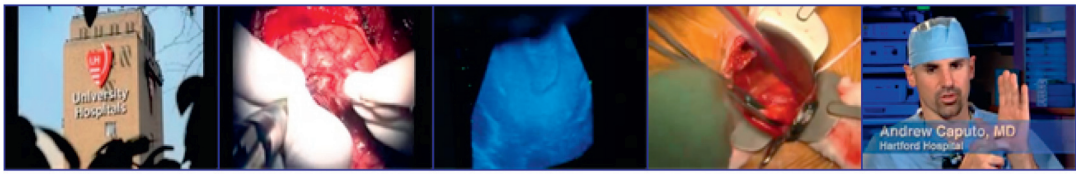
5.1. YUV videos

In this experiment, commonly used video test sequences in YUV format with miscellaneous content were used from [21]. Based on the color histograms, color moments and edge histograms, a hierarchical view with 5 classes at each level of the hierarchy is created using hierarchical k-means clustering.

Fig. 5 shows the hierarchical views of YUV video sequences. The 5 classes at the top level (Fig. 5(a)) is obtained by clustering the entire data of all videos. Each class is represented by an appropriate key frame which is selected as the closest frame to the centroids of the corresponding cluster. Successive clustering of lower level views is then performed based on the clustering results of the previous level classes.



(a) Hierarchy level 1.



(b) Hierarchy level 2 from the second class at level 1 in (a).

(c) Hierarchy level 3 from the second class at level 2 in (b).

(d) Hierarchy level 4 from the first class at level 3 in (c).

Fig. 7. Hierarchical structure of medical video sequences.

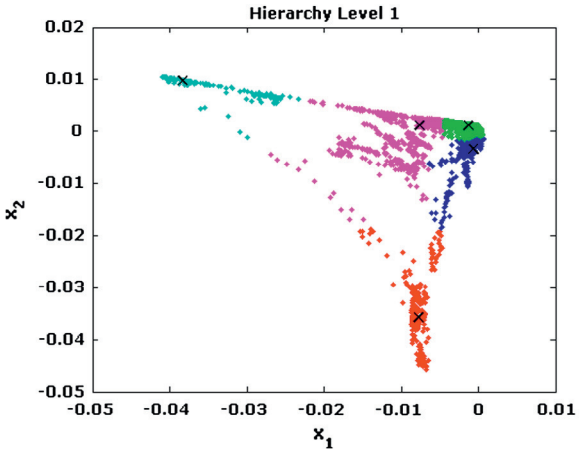
For instance, Fig. 5(b)–(d) depict the second level of hierarchy produced from the clustering of the third, fourth and fifth classes at level 1, respectively. As shown in Fig. 5(b) and (c), "container" and "mother and daughter" scenes are already generated at hierarchy level 2. On the other hand, since scenes of second level of hierarchy from the fifth class at level 1 (Fig. 5(d)) contain different content, i.e., "tennis", "tempete" and "waterfall", it needs to be further explored. Finally, clustering first, second and third classes at this level, "tennis", "tempete" and "waterfall" scenes are obtained at level 3 as shown in Fig. 5(e) and (f), respectively.

Fig. 6 shows the clustering plots of YUV video sequences. Fig. 6(a) presents the clustering of all the video data. Clustering of third, fourth and fifth classes at hierarchy level 1 are given in Fig. 6(b)–(d), respectively. Finally, Fig. 6(e) and (f) depict the clustering at second level of hierarchy: from the first and second classes, respectively. The x and y axes of clustering plots, respectively, represent the first and second principal components obtained by principal component analysis. Black cross markers ($\times$) on the plots indicate the centroids of the clusters which are used in selecting key frame representing the cluster.

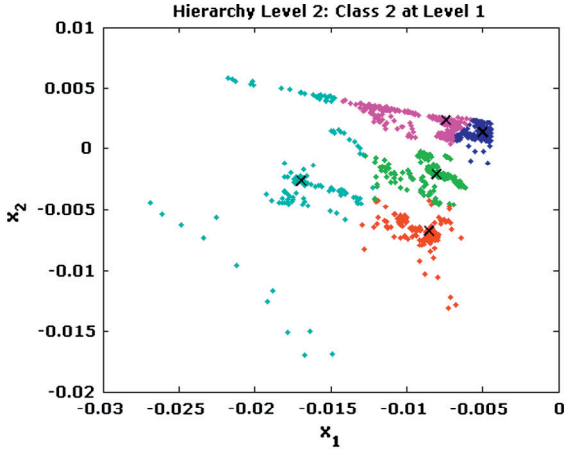
Such hierarchical structure can make annotation process very easy and fast. User can provide textual labels to the content when textual information is not available. Automatic annotation is also possible in the case of availability of specific features of desired scene.

5.2. Medical videos

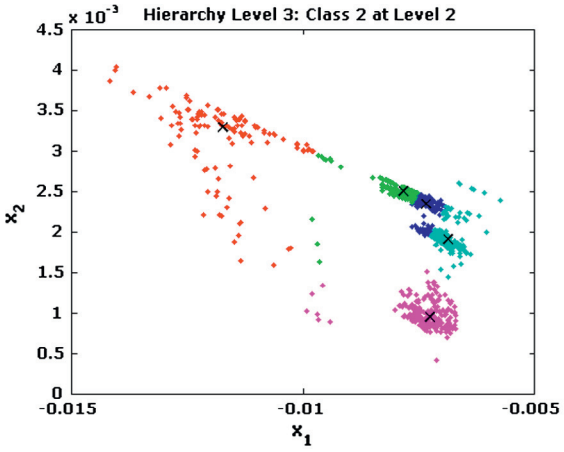
The proposed system was also experimented with a medical video sequence from [22]. The sequence is 8 min long and contains 60 shots.



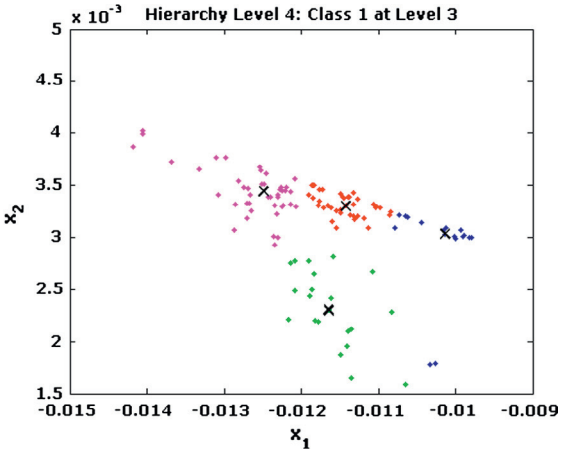
(a) Hierarchy level 1.



(b) Hierarchy level 2 from the second class at level 1 in (a).



(c) Hierarchy level 3 from the second class at level 2 in (b).



(d) Hierarchy level 4 from the first class at level 3 in (c).

Fig. 8. Clustering plots of medical video sequences.

Figs. 7 and 8 show the hierarchical views and corresponding clustering plots of medical video sequences, respectively. As shown in Fig. 7(a), an overview of the video content can be provided by looking at the top level. In this experiment, second level of hierarchy is generated from the second class at level 1 (Fig. 7(b)). At level 3, 5 subclasses obtained from the second class at level 2 are similar in color (i.e., white and red[1]) as shown in Fig. 7(c). Finally, scenes of same surgical operation are succesfully captured at level 4, which are under the first class at level 3 (Fig. 7(d)).

As in the previous experiment, our method can produce very good hierarchical view of the video sequences, which can make the annotation process very affordable. In addition, proposed approach can provide very efficient browsing facility. For instance, in order to find a specific part of a video stream, one can browse all video sequences by moving to different levels of hierarchy within the built story structure instead of browsing the entire videotape sequentially.

Proposed technique has several strengths. It has an advantage having linear computational complexity as compared to other methods which have non-linear run time. Therefore, the algorithm is efficient even for large data. In addition, as shown in the experiments, the method performs well for anisotropic video data with varying content. Another major advantage of the algorithm is that it generates nested partitioning structure rather than a single partition, which brings effectiveness in accessing a specific part of the video. On the other hand, the proposed technique may cause to outlier problem. However, such cases can be detected and clusters representing outliers can be regarded as miscellaneous content that does not fit into overall content pattern.

6. Conclusion

In this paper, we presented a novel hierarchical clustering based schema for video annotation. The proposed system generates a top-down hierarchy by successively partitioning each node of the tree into its children using *k*-means clustering based on the visual features. Similar contents can be then grouped at each level of the hierarchy. Thus, the content of the video can be roughly known without the need of sequential looking up manner. The experiments conducted showed that the system can produce very good hierarchical view of the video sequences, which can ease the annotation as well as browsing operations.

References

[1] Mendi E, Bayrak C. A web-based medical video indexing environment. In: Proceedings of the 2010 IEEE fourth international conference on semantic computing; September 2010. p. 172–5.
[2] Hanna J, Patlar F, Akbulut A, Mendi E, Bayrak C. HMM based classification of sports videos using color feature. In: Proceedings of the 6th IEEE international conference on intelligent systems (IS '12); September 2012. p. 388–90.
[3] Feng SL, Manmatha R, Lavrenko V. Multiple bernoulli relevance models for image and video annotation. In: Proceedings of the 2004 IEEE computer society conference on computer vision and pattern recognition (CVPR '04); 2004. p. 1002–9.
[4] Jeon J, Lavrenko V, Manmatha R. Automatic image annotation and retrieval using cross-media relevance models. In: Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval; 2003. p. 119–26.
[5] Lavrenko V, Manmatha R, Jeon J. A model for learning the semantics of pictures. In: NIPS; 2003.
[6] Xiang Y, Zhou X, Chua T-S, Ngo C-W. A revisit of generative model for automatic image annotation using markov random fields. In: IEEE conference on computer vision and pattern recognition, 2009. CVPR 2009; 2009. p. 1153–60.
[7] Rabinovich A, Vedaldi A, Galleguillos C, Wiewiora E, Belongie S. Objects in context. In: ICCV; 2007. p. 1–8.
[8] Qi G-J, Hua X-S, Rui Y, Tang J, Mei T, Zhang H-J. Correlative multi-label video annotation. In: Proceedings of the 15th international conference on multimedia, ser. MULTIMEDIA '07; 2007. p. 17–26.
[9] Jiang Y-G, Yang J, Ngo C-W, Hauptmann AG. Representations of keypoint-based semantic concept detection: a comprehensive study. IEEE Trans Multimedia 2010;12(1):42–53.
[10] Cusano C, Ciocca G, Schettini R. Image annotation using SVM. Proc SPIE 2004;5304:330–8.
[11] Carneiro G, Chan AB, Moreno PJ, Vasconcelos N. Supervised learning of semantic classes for image annotation and retrieval. IEEE Trans Pattern Anal Mach Intell 2007;29(3):394–410.
[12] Carneiro G, Vasconcelos N. Formulating semantic image annotation as a supervised learning problem. In: Proceedings of the 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), ser. CVPR '05, Washington, DC, USA, vol. 2; 2005. p. 163–8.
[13] Wang H, Huang H, Ding C. Image annotation using bi-relational graph of images and semantic labels. In: Proceedings of the 2011 IEEE conference on computer vision and pattern recognition, ser. CVPR '11; 2011. p. 793–800.
[14] Mendi E, Bayrak C. Shot boundary detection and key-frame extraction from neurosurgical video sequences. Imaging Sci J 2012;60(2):90–6.
[15] Chen W, Shi YQ, Xuan G. Identifying computer graphics using HSV color model and statistical moments of characteristic functions. In: IEEE international conference on multimedia and expo (ICME 2007); 2007. p. 1123–6.
[16] Yu L, Gimel'farb G. Image retrieval using color co-occurrence histograms. In: Proc of the image & vision computing New Zealand 2002 (IVCNZ2003); November 2003. p. 42–7.
[17] Stricker M, Orengo M. Similarity of color images. In: Proc SPIE storage and retrieval for image and video databases, vol. 2420; February 1995. p. 381–92.
[18] Canny J. A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 1986;8(6):679–98.
[19] Jain AK, Murty MN, Flynn PJ. Data clustering: a review. ACM Comput Surv 1999;31(3):264–323.
[20] Hastie T, Tibshirani R, Friedman JH. The elements of statistical learning. Springer; 2009.
[21] YUV video sequences. <http://trace.eas.asu.edu/yuv/>.
[22] ORLive, Inc. Online surgical and healthcare video and webcasts. <http://www.orlive.com/>.

[1] For interpretation of color in Fig. 7, the reader is referred to the web version of this article.

Engin Mendi is an assistant professor in computer engineering department at KTO Karatay University, Turkey. He received PhD degree from Integrated Computing at the University of Arkansas at Little Rock (UALR); two MS degrees, one in Applied Science from UALR and the other in Computational Engineering from Technical University of Munich and BS degree from Middle East Technical University.