Video Scene Detection by Link-Constrained Affinity-Propagation

Chun-Rong Huang and Chu-Song Chen
Institute of Information Science
Academia Sinica
Taipei, Taiwan, R.O.C.
Email: {nckuos, song}@iis.sinica.edu.tw

Abstract—**Video scenes provide semantic meanings for video content description and summarization. This paper explores the pair-wise visual cues of near-duplicate objects for link-constraint affinity-propagation without using keyframes. Experiments demonstrate that our method is more capable to identify scenes comparing with non-constrained clustering algorithms.**

I. INTRODUCTION

Video scene detection is important for video indexing, skimming, summarization and understanding. A video scene is defined as a collection of semantically related shots that describe a high-level concept or story [1]. Compared to shots, video scenes provide richer semantic meanings for video content description and serve as a compact representation for indexing and summarization.

Many scene detection methods have been proposed. Videos are usually segmented into shots at first. Keyframes are then extracted to represent each shot. By comparing the keyframes, visually similar shots can be clustered and integrated into scenes. For example, Yeung *et al.* [2] proposed a scene transition graph (STG) to represent the video structure. An STG is built by clustering the shots based on their visual similarities and temporal consistencies. Hanjalic *et al.* [3] used a similar approach to automatically segment movies into logical story units (LSU). An LSU is a series of temporally contiguous shots containing similar visual content identified by comparing the DCT images of the keyframes. Javed *et al.* [4] proposed a framework to remove commercials from talk show videos. This approach constructs a shot connectivity graph (SCG) that links similar shots over time by comparing HSI histograms between keyframes. Scenes belonging to a talk show are obtained by identifying strongly connected structures of the SCG.

Instead of using temporally close shot pairs, Rasheed and Shah [5] considered the similarities between every shot pair. They construct a fully connected shot similar graph (SSG) by using color and motion information of keyframes. The normalized cut technique is then applied to the SSG for graph partition. These partitions are used to represent individual scenes in the video. Ngo *et al.* [6] also represented a video as a weighted unidirectional graph, which is partitioned into sub-graphs (clusters) by using normalized cuts. They added the time order information to build a new temporal graph and then extracted the scene boundaries. Chasanis *et al.* [7] proposed a scene detection method that uses an improved spectral clustering method with a global *k*-means algorithm to automatically decide the number of clusters. During clustering, shot similarity is evaluated based on the visual features only, without imposing the time constraints.

We believe that scene detection should be based not only on the pair-wise shot similarities obtained by comparing low-level features, but also on matching similar objects or backgrounds between shots. If the same objects or backgrounds appear in two shots, we consider the possibility that the two shots may have been taken in the same scene. An object is called a near-duplicate of a reference object if it is visually similar to the reference object. If two shots are captured in the same scene, it is likely that many pairs of near-duplicate objects (NDOs) will be found. We propose using an invariant-local-feature-based method to represent the objects in a shot. NDO pairs can be found by comparing the objects between shots. We also employ the color ratio gradient (CRG) [8] that is insensitive to geometric and photometric transformation to evaluate the color distribution of each shot. The shot similarities are computed by comparing objects and CRG distributions between shots.

During clustering, we do not apply hard constraints, such as “must-link” [9], to force shots containing similar objects into the same group because the constraints may contain outliers and noises when performing NDO detection. We also relax the time constraint that a scene must consist of temporally connected shots. We deem that shots with the same NDO should have a high probability to be clustered together. These can be considered as “soft-link” constraints between shots. We modify the affinity propagation [10] by employing the “soft-link” constraints to cluster shots into scenes. Even when some NDO pairs are incorrect, our method is still able to group the shots of the same scene.

II. PRELIMINARY

Recently, invariant local features have been proposed to solve the image matching problem. One of the most famous invariant local features is SIFT [11]. However, it is not suitable for some real-time applications. Huang *et al.* [12] proposed the contrast context histogram (CCH) to find image correspondence. As shown in [12], CCH has comparable matching accuracy to that of the SIFT, but has much improvement in the computation efficiency.

This research was supported by the National Science Council of Taiwan under Grant No. NSC98-2631-001-013.

978-1-4244-3828-0/09/$25.00 ©2009 IEEE

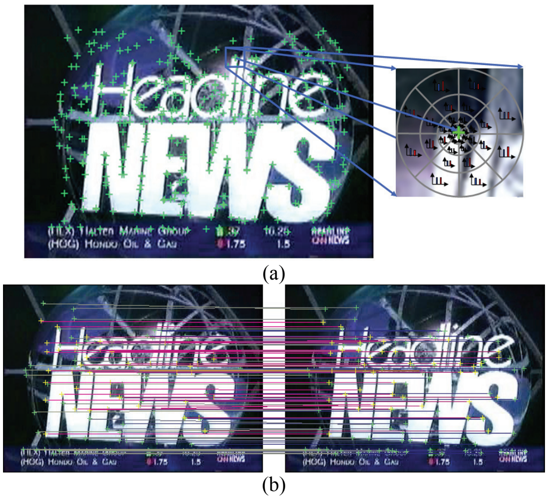

Fig. 1. (a)(Left) Keypoints (marked by green crosses) detected in the image. (Right) The log-polar coordinate system centered at a keypoint. (b) Matching results.

In CCH, Harris-Laplacian corners [13] (keypoints), which have been shown robust to image rotation, translation, scale changes, are extracted at first. For each keypoint p, a log-polar coordinate system centered at the keypoint is defined to divide the neighbor region into n non-overlapping sub-regions, as illustrated in Fig. 1(a). To construct a descriptor for each keypoint, CCH uses the contrast histogram, which is relatively insensitive to lighting changes and easy to compute. The contrast value is defined as the gray-value difference between the neighbor pixel and the keypoint. The positive and negative contrast histograms of a sub-region are defined as the averages of the positive and the negative contrast values in the sub-region, respectively. Concatenating the values of all the sub-regions forms a $2n$-dimensional vector, called the CCH descriptor of p, which reflects local image properties and is used to characterize the keypoint. To ensure that the descriptor is invariant to image rotations, the log-polar coordinate system is rotated to coincide with the edge orientation of the keypoint. To make the descriptor invariant to linear lighting changes, we normalize the CCH descriptor to a unit-length vector

Given two images I_1 and I_2, we detect the keypoints and compute the CCH descriptors for each image. The similarity of two keypoints is measured by the Euclidean distance of their CCH descriptors. We compare each keypoint in I_1 with all the keypoints in I_2 to identify its nearest neighbor and the second nearest neighbor. If the distance between the nearest neighbor and the second nearest neighbor is greater than a pre-defined threshold, we regard that the corresponding point in I_2 is found for the keypoint in I_1. Fig. 1(b) shows the image matching results obtained by using the CCH descriptors.

III. Method

A. Shot Change Detection

Since a video scene is a collection of semantically related shots, we first divide a video into shots by using the CCH feature based method [14]. The shot change detection algorithm is summarized as follows:

1) Construct the CCH keypoints of each frame.
2) Match the CCH keypoints between adjacent frames.
3) Obtain the candidate transitions from the local minima of the numbers of matched points.
4) Estimate the intervals of candidate transitions.
5) Perform fine selection of the scene transitions by matching the first and last frames of each interval and then eliminate the inappropriate ones.

Then, the i-th shot in the video can be expressed as a set of frames $S_i = \{f_a, f_{a+1}, \ldots, f_b\}$, where a and b are the indices of the first and the last frames of the shot, respectively.

B. Shot Representation

To find shot similarities, keyframes are usually extracted to represent a shot. Choosing a proper number of keyframes for comparison of shots can avoid the exhaustive matching of all the frames between shots. However, due to the variation of videos, few keyframes may not be sufficient to represent the shot taken under camera motion or containing fast moving objects. Thus, we propose to directly extract average space-time features from the entire shot instead of using keyframes.

1) Object key feature

Shots belong to one scene often contain similar objects or backgrounds. When two shots contain lots of NDO pairs, it is highly likely that they belong to the same scene. We propose the object key feature, which is an average space-time feature of a shot, for shot representation. NDO pairs are found according to the matching of object key features.

Given the n-th shot, let us consider all the keypoints that have been extracted from the frames of the shot. Assume that there are a total of K keypoints in the n-th shot, where the k-th keypoint moves along a trajectory in the space-time volume of the shot. This trajectory can be represented by a time-ordered set of points. A keypoint k tracked in shot n is denoted as $O_{n,k}$ and the imaged location of $O_{n,k}$ at time t is $X_{n,k}(t) = [x_{n,k}(t), y_{n,k}(t)]^T$. The trajectory $T_{n,k}$ of $O_{n,k}$ is a sequence of points $X_{n,k} = \{X_{n,k}(i), X_{n,k}(i+1), \ldots, X_{n,k}(j)\}$, where i and j ($j > i$) are the indices of the first frame and the last frame of the object k. The object key feature $A_{n,k}$ of $O_{n,k}$ is represented by the CCH descriptors as follows:

$$A_{n,k} = \frac{\sum_{t=i}^{j} CCH(X_{n,k}(t))}{j-i}. \tag{1}$$

For two shots m and n, a NDO $O_{n,m,k,l}$ is a matched pair of $(O_{n,k}, O_{m,l})$, which indicates that the tracked keypoint k in shot n is compared with the tracked keypoint l in shot m. Given $O_{n,k}$, let $O_{m,l}$ and $O_{m,l'}$ be its most similar and second-most similar keypoints in shot m, respectively. The similarity between two shots is measured by the Euclidean distance between their object key features. We then identify whether $O_{n,k}$ and $O_{m,l}$ are matched by the following criterion:

$$Dist(A_{n,k}, A_{m,l}) < \alpha \times Dist(A_{n,k}, A_{m,l'}) , \tag{2}$$

where $Dist(\cdot)$ is the Euclidean distance between the appearance descriptors, and α=0.6 is a pre-defined constant. Compared with the keyframe based methods, object key features avoid the insufficient keyframe problems when videos are captured with a fast camera or object motion.

2) Shot color invariance feature

Beside keypoints, shots in the same scene often contain similar colors. However, the color content of shots may be affected by different capturing conditions. To overcome the problem, we use the color ratio gradient (CRG) introduced in [8] that is insensitive to object positions, shadows, and illuminations, to represent the color content of the shot. CRG is derived from the color constant color ratio. The color constant color ratio is defined as follows:

$$M(C_1^{\vec{x}_1}, C_1^{\vec{x}_2}, C_2^{\vec{x}_1}, C_2^{\vec{x}_2}) = \frac{C_1^{\vec{x}_1} C_2^{\vec{x}_2} - C_1^{\vec{x}_2} C_2^{\vec{x}_1}}{C_1^{\vec{x}_2} C_2^{\vec{x}_1} + C_1^{\vec{x}_1} C_2^{\vec{x}_2}}, C_1 \neq C_2, \tag{3}$$

where $C_1, C_2 \in \{R, G, B\}$ for a standard RGB camera, and $\vec{x}_1$ and $\vec{x}_2$ denote the image locations of two neighboring pixels. The CRG of an image point (x, y) is defined as

$$\begin{aligned} &\nabla M(C_1^{\vec{x}_1}, C_1^{\vec{x}_2}, C_2^{\vec{x}_1}, C_2^{\vec{x}_2}) = \\ &[M(C_1^{(x-1,y)}, C_1^{(x+1,y)}, C_2^{(x-1,y)}, C_2^{(x+1,y)})^2 + \\ &M(C_1^{(x,y-1)}, C_1^{(x,y+1)}, C_2^{(x,y-1)}, C_2^{(x,y+1)})^2]^{1/2} \end{aligned}, \tag{4}$$

where $C_1, C_2 \in \{R, G, B\}$.

To obtain a better representation of an image frame f_i in a shot, we split the frame into W sub-regions $\boldsymbol{R}_1$, $\boldsymbol{R}_2$, …, $\boldsymbol{R}_W$. Each region $\boldsymbol{R}_w$ contains three CRG histograms, namely, $H_{\boldsymbol{R}_w}^{\nabla M(R,G)}$, $H_{\boldsymbol{R}_w}^{\nabla M(R,B)}$, and $H_{\boldsymbol{R}_w}^{\nabla M(G,B)}$. The CRG distribution $H_{f_i}^{\nabla M}$ of the frame f_i is defined as the combination of the CRG histograms from all the sub-regions. We consider that each sub-region forms a space-time volume beginning from the starting and the ending frames of the shot. Then the shot color invariance feature of the shot n is defined as follows:

$$H^n = \frac{1}{b-a} \sum_{i=a}^{b} H_{f_i}^{\nabla M}. \tag{5}$$

3) Shot similarities

The object key feature similarity between the shots m and n is defined as the number of the tracked keypoints being matched as follows:

$$D_I(m,n) = card(Match(m,n)), \tag{6}$$

where

$$Match(m,n) = \{(O_{n,k}, O_{m,l}) \mid (A_{n,k}, A_{m,l}) \text{ satisfies } (2)\}, \tag{7}$$

and $card(\cdot)$ is the cardinality of a set. When $D_I(m, n) > \delta$, where δ is a positive integer, we assume that shot m and shot n contains NDOs.

The shot color invariance similarity between the shots m and n is defined as the histogram intersection of two histograms as follows:

$$D_C(m,n) = \frac{\sum_{k=1}^{N} \min\{H^m(k), H^n(k)\}}{\sum_{k=1}^{N} H^n(k)}. \tag{8}$$

The similarity between shots m and n is a function of object key feature and color invariance feature similarities, defined as follows:

$$s(m,n) = D_I(m,n) + D_C(m,n). \tag{9}$$

C. Scene Detection

NDOs between shots provide a strong constraint to identify whether the shots belong to the same scene. In [9], Wu *et al.* proposed to use near-duplicate keyframes (NDKs) to bridge news stories from various video sources over time. They treated such visual constraints as a "must-link" condition, i.e. two stories that share at least one pair of NDKs are always placed into the same cluster. In practice, although NDKs provide relatively credible evidences to link two shots, the constraints may also be outliers and noise due to errors in automatic NDKs detection. If we adopt the constraint that shots contains NDO pairs must to be clustered, the false NDO pairs will cause false clustering results. Thus, we apply "soft-link" constraints instead of "must-link" constraints. To cluster shots in the same scene by "soft-link" constraints, we introduce the affinity propagation (AP) [10] at first. The AP algorithm is an exemplar-based clustering algorithm that operates by exchanging messages between data points and learns a probability model of the data. Two kinds of messages, responsibility and availability, are used to take a different kind of competition into account. The responsibility r(m, n), sent from data point m to candidate exemplar point n, reflects how well-suited point n is to serve as the exemplar for point m. The availability a(m, n), sent from candidate exemplar point n to point m, reflects how appropriate it would be for point m to choose point n as its exemplar. The affinity between two data points gives a direct indication of whether they should be clustered together. In our approach, we use the shot similarity s(m, n) in (9) as the similarity of the affinity propagation.

In our approach, two "soft-link" constraints are conducted. The first one is that two shots contain NDO pairs should have a higher probability to be clustered because these shots are possibly captured in the same scene. During the message-passing procedure, we consider that NDO pairs provide a special message which links two shot m and n. Shot m then sends strong responsibility messages to shot n that shot m wants to choose shot n as its exemplar. The second one is that temporally connected shots may be captured in the same scene. During the message-passing procedure, we apply an additional temporal message t(m, n) to model the temporal distances between shots. The same as the original affinity propagation, the initial availabilities a(m, n) are set to zero. By apply the "soft-link" constraints, the new responsibility of our link-constrained affinity propagation is defined as follows:

$$r(m,n) = \begin{cases} s(m,n) + t(m,n), \text{if } s(m,n) > \delta \\ s(m,n) + t(m,n) - \\ \max\limits_{n' s.t. n' \neq n} \{a(m,n') + s(m,n') + t(m,n')\}, \text{otherwise} \end{cases} \tag{10}$$

, where δ = 5 and the temporal message t(m, n) is defined by the temporal distances between shots m and n as follows:

TABLE I. RESULTS

			Our method		AP	
Data	*Length*	*Scene*	*R*	*P*	*R*	*P*
1	5:59	24	79.2	86.4	87.5	87.5
2	17:21	29	82.8	80.0	69.0	66.7
3	31:15	36	83.3	76.9	75.0	69.2
4	27:34	60	88.3	89.8	75.0	77.6
Avg.	82:09	149	84.6	84.0	75.8	74.8

$$t(m,n) = e^{\frac{-(m-n)^2}{T}}, \tag{11}$$

where $T = 20$. If shots contain NDO pairs, data point *m* will send a strong message to candidate exemplar point *n* according to the similarity between shots *m* and *n*. If there are no NDO pairs between two shots, the shot *m* is then find the most suitable exemplar shot *n* based on the incoming availability, the shot similarity and the temporal message. The availability $a(m, n)$ is defined as follows:

$$a(m,n) = \min\left\{0, r(n,n) + \sum_{m'\text{s.t.}m' \notin \{m,n\}} \max\{0, r(m',n)\}\right\}, \tag{12}$$

which is used to identify which candidate exemplar would become a good exemplar. The self-availability is updated by

$$a(n,n) = \sum_{m' s.t. m' \notin \{m,n\}} \max\{0, r(m',n)\} \cdot \tag{13}$$

After the convergence, availabilities and responsibilities are combined to identify exemplars. For shot *m*, its corresponding exemplar is obtained as

$$n^* = \arg\max{}_n \{a(m,n) + r(m,n)\} \; . \tag{14}$$

As a result, the shots are clustered according the "soft-link" constraints.

IV. EXPERIMENTS

To evaluate the proposed method, we use four video sequences in our experiments. The details of the videos are summarized in Table I. The news clip is from CNN. The movie "House of Flying Daggers" includes many complex dance and fight scenes, so it is good material for testing object tracking and the effect of motion blur. The animation sequence is from the 3D animation movie "Ratatouille". The documentary video is "The Story of Hoover Dam" from the open video project [15]. To evaluate the performance of the proposed algorithm, we use the recall (R) = $H/(H+M)$ and precision (P) = $H/(H+F)$ metrics, where H, M, and F are the numbers of hits, miss detects, and false detects, respectively. For each video, we manually identified the scene clusters as the ground truth.

As shown in Table I, the average recall and precision in our experiments are 84.6% and 84.0%, respectively. We compared our method with the standard affinity propagation algorithm [10]. Our approach performed better results in most cases. We owe the reasons to the following. The NDO constraints provide stronger links between shots, so that the shots in the same scene are easier to find the same exemplar. The temporal message helps to collect temporal connected shots into a scene; even they are possibly not visually similar. In contrast, the standard affinity propagation algorithm might keep these kinds of shots as exemplars themselves. As a result, with the "soft-link" constraints, our approach can achieve better performances.

V. CONCLUSIONS

We proposed a video scene detection method based on link-constrained affinity propagation. Compared with low level features, NDO pairs provide relatively reliable information to link shots. Using shot color invariant features to estimate the color attributes of shots is more robust against the variant capturing conditions. The experiments have shown that the link-constrained affinity propagation is useful to deal with the video scene detection problem.

REFERENCES

[1] Z. Xiong, X. S. Zhou, Q. Tian, Y. Rui, and T. S. Huang, "Semantic retrieval of video - review of research on video retrieval in meetings, movies and broadcast news, and sports," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 18–27, 2006.

[2] M. Yeungy, B.-L. Yeoz, and B. Liu, "Segmentation of video by clustering and graph analysis," *Computer Vision and Image Understanding*, vol. 71, no. 1, pp. 94–109, 1998.

[3] A. Hanjalic, R. Lagendijk, and J. Biemond, "Automated high-level movie segmentation for advanced video-retrieval systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 580–588, 1999.

[4] O. Javed, Z. Rasheed, and M. Shah, "A framework for segmentation of talk and game shows," *in Proc. of the International Conference on Computer Vision,* vol. 2, pp. 532–537, 2001.

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

[6] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, "Video summarization and scene detection by graph modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, pp. 296–305, Feb. 2005.

[7] V. Chasanis, A. Likas, and N. Galatsanos, "Scene detection in videos using shot clustering and symbolic sequence segmentation," *in Proc. of IEEE 9th Workshop on Multimedia Signal Processing*, pp. 187–190, Oct. 2007.

[8] T. Gevers, "Image segmentation and similarity of color-texture objects," *IEEE Transactions on Multimedia*, vol. 4, no. 4, pp. 509–516, Dec. 2002.

[9] X. Wu, C.-W. Ngo, and A.G. Hauptmann, "Multimodal news story clustering with pairwise visual near-duplicate constraint," *IEEE Transactions on Multimedia*, vol. 10, no. 2, pp. 188–199, 2008.

[10] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, p. 91V110, 2004.

[12] C.-R. Huang, C.-S. Chen, and P.-C. Chung, "Contrast context histogram – an efficient discriminating local descriptor for object recognition and image matching," *Pattern Recognition*, vol. 41, no. 10, pp. 3071–3077, 2008. [Online] Software available: http://imp.iis.sinica.edu.tw/CCH/CCH.htm

[13] K. Mikolajczyk and C. Schmid, "Indexing based on scale invariant interest points," *in Proc. of the International Conference on Computer Vision*, vol. 1, pp. 525–531, 2001.

[14] C.-R. Huang, H.-P. Lee, and C.-S. Chen, "Shot change detection via local keypoint matching," *IEEE Transactions on Multimedia*, vol. 10, no. 6, pp. 1097-1108, 2008.

[15] The Open Video Project, 2008. [Online] Available: http://www.open-video.org