Discriminative techniques for keyframe selection

Conference Paper · August 2005

DOI: 10.1109/ICME.2005.1521470 · Source: IEEE Xplore

Matthew Cooper
FX Palo Alto Laboratory

DISCRIMINATIVE TECHNIQUES FOR KEYFRAME SELECTION

Matthew Cooper and Jonathan Foote

FX Palo Alto Laboratory
Palo Alto, CA 94304
{cooper, foote}@fxpal.com

ABSTRACT

A convenient representation of a video segment is a single "keyframe." Keyframes are widely used in applications such as non-linear browsing and video editing. With existing methods of keyframe selection, similar video segments result in very similar keyframes, with the drawback that actual differences between the segments may be obscured. We present methods for keyframe selection based on two criteria: capturing the similarity to the represented segment, and preserving the differences from other segment keyframes, so that different segments will have visually distinct representations. We present two discriminative keyframe selection methods, and an example of experimental results.

1. INTRODUCTION

Video or motion pictures consist of a series of still images. Many applications extract one or more of these still images, termed keyframes, as useful graphical representations of the video data [1]. For example, in the file view interface of many operating systems, the first frame in a video file is often used to represent that data.

Users frequently need to manipulate a large number of video clips; for example, in "drag-and-drop" video editors like Apple's iMovie [1]. A drawback of previous keyframe selection techniques is that similar clips result in keyframes that are nearly identical. Thus different segments are indistinguishable in the interface. Many common video sources share this problem, such as short video clips from digital camera, or pre-segmented results from a video repository.

For practical browsing and manipulation of video media, keyframes must both represent the underlying video clip and distinguish that clip from the remainder of the collection. This is accomplished by measuring the similarity of the keyframe to both the shot it came from as well as other shots. Preferable keyframes must both resemble the shot they came from as well as differ from the other keyframes. In this paper, we present quantitative methods for selecting keyframes that are *both representative and discriminative*.

This approach resembles the familiar term-frequency-inverse-document-frequency (TF/IDF) keyword weighting developed for text document indexing and retrieval [2]. The ratio combines two factors: how well a keyword represents a particular document, and how well the keyword discriminates that document within the entire collection. We extend this idea to keyframe selection within video documents. Ideally, video source files have been segmented into shorter video shot segments or "shots." Because shots are locally homogeneous, a single keyframe is a reasonable representation. Any of the many existing approaches may be used for shot segmentation [3]. In the following discussion, we use the generic term "segment" to refer to any continuous video file, regardless of its source. In fact, this approach may also be used to select representative images from any image collection or set of collections.

2. RELATED WORK

In the Manga system [4], keyframes are selected using an importance score based on shot length and rarity within an agglomerative clustering framework. Like the methods presented here, this system trades off similarity and redundancy for keyframe selection. However, our system does not discard any (segment's) keyframes. More commonly, keyframe selection follows shot detection. In [5], video skims are produced by selecting short video shots and selecting keyframes based on detected object motion. There are also systems that select keyframes based on text, audio, or speech analysis, e.g. [6]. Aoki *et al.* [7] presented a rule-based approach for detecting scene structures such as dialogues ("patterns" and "acts") and to exploit redundancy in such scenes in keyframe selection. In that work, redundancy was defined in terms of specific rules (shot-level patterns) rather than frame-level feature similarity.

[8] presents a framework for video summarization also seeking to select keyframes that are "maximally distinct and individually carry the most information". Information is quantified using the color distribution entropy within each frame. The Bhattacharyya distance [9] between color-based histograms is combined with a temporal distance measure

[1] http://www.apple.com/ilife/imovie/

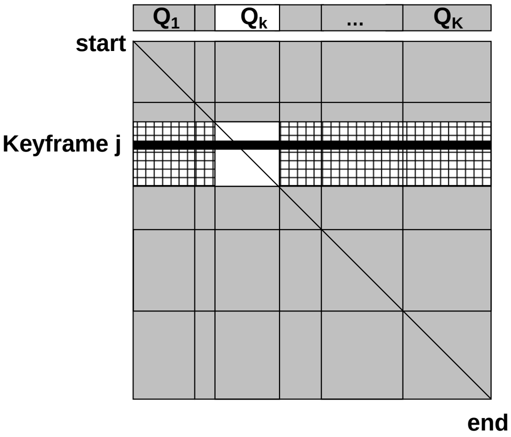

Fig. 1. Given video segments denoted $\mathbf{Q}_1, \cdots, \mathbf{Q}_K$, we can compute the average self-similarity $\mathcal{S}(j, \mathbf{Q}_k)$ as well as the average cross similarity $\mathcal{C}(j, \mathbf{Q}_k)$ using a similarity matrix.

to quantify inter-frame dissimilarity, and keyframes are selected using dynamic programming. In contrast to the approach here, the technique uses these frame based measures *within each shot* rather than across the entire source video (possibly for computational reasons). [10] presented an iterative approach to discriminative text document clustering based on mixture models that may also be applicable to keyframe selection. The approach is based on mixture modelling of co-occurrence data, in contrast to the non-iterative methods presented herein. Unlike TF/IDF [2], our approach is based on similarity rather than feature (keyword) occurrence frequencies.

3. KEYFRAME SELECTION

To measure keyframe similarity, we first compute feature vectors based on low-order discrete cosine transform (DCT) coefficients. We sample frames at 1 Hz and transform the individual RGB frames into the Ohta color space [11]. The DCT of each transformed channel is computed and a feature vector is formed by concatenating the resulting 25-49 low frequency coefficients of the three channels. Other parameterizations may of course be used, but we have not evaluated them here.

3.1. Similarity-based implementation

The result of the parametrization step is a vector of features for each frame. Given a source video with N frames, denote the frame-indexed feature vectors $\mathbf{V} = \{v_i : i = 1, \cdots, N\}$. The similarity between any two frames can be calculated using a distance measure; here we use the cosine similarity measure. The similarity between a candidate keyframe and a video segment is the average similarity between the given keyframe and all frames in the segment. Represent the segment comprised of frames $l, \cdots, r$ by $\mathbf{Q} = \{v_i : i = l, \cdots, r\} \subset \mathbf{V}$. Given the similarity measure $d(\cdot, \cdot)$, the average similarity $\mathcal{S}$ between any candidate keyframe j and the segment $\mathbf{Q}$ is

$$\mathcal{S}(j, \mathbf{Q}) = \frac{1}{|\mathbf{Q}|} \sum_{v_m \in \mathbf{Q}} d(v_j, v_m) \ , \quad l \leq j \leq r \ . \qquad (1)$$

$\mathcal{S}$ is then the average *self-similarity* of keyframe j. Let $\mathcal{C}$ be the average *cross-similarity*, or the similarity of keyframe j to all other segments in the collection: $\bar{\mathbf{Q}} \equiv \mathbf{V} \setminus \mathbf{Q}$. Define

$$\mathcal{C}(j, \mathbf{Q}) = \frac{1}{|\bar{\mathbf{Q}}|} \sum_{v_m \in \bar{\mathbf{Q}}} d(v_j, v_m) \ , \quad l \leq j \leq r \ . \qquad (2)$$

A similarity matrix $\mathbf{S}$ with elements $\mathbf{S}(i, j) = d(v_i, v_j)$, as depicted in Figure 1, can be used as a look-up table to accelerate these calculations.

A representative keyframe will have a high value of $\mathcal{S}$ – it will be very similar to the other frames within the same segment, on average. To be discriminative, the frame should also minimize $\mathcal{C}$ – it should not resemble frames (and hence the keyframes) in other segments. The difference or ratio of the two values indicate how well a given keyframe satisfies both criteria. Thus a subtractive figure of merit is

$$\mathcal{F}_S(j, \mathbf{Q}) = \mathcal{S}(j, \mathbf{Q}) - \mathcal{C}(j, \mathbf{Q}) \ , \qquad (3)$$

while a rational figure of merit is

$$\mathcal{F}_R(j, \mathbf{Q}) = \frac{\mathcal{S}(j, \mathbf{Q})}{\mathcal{C}(j, \mathbf{Q})} \ . \qquad (4)$$

To trade off the discrimination versus the self-similarity measures, we may construct weighted measures, using non-negative constants α and β as follows:

$$\mathcal{F}_S(j, \mathbf{Q}) = \alpha_S \mathcal{S}(j, \mathbf{Q}) - \beta_S \mathcal{C}(j, \mathbf{Q}) \ , \qquad (5)$$

and

$$\mathcal{F}_R(j, \mathbf{Q}) = \frac{(\mathcal{S}(j, \mathbf{Q}))^{\alpha_R}}{(\mathcal{C}(j, \mathbf{Q}))^{\beta_R}} \ . \qquad (6)$$

In both cases, increasing α relative to β will increase the importance of self-similarity relative to discrimination, and vice-versa.

To select the best representative keyframe v^* for a segment $\mathbf{Q}$, we maximize the goodness function $\mathcal{F}$ over all frames in $\mathbf{Q}$:

$$v^* = \operatorname*{argmax}_{v_j \in \mathbf{Q}} \mathcal{F}(j, \mathbf{Q}) \qquad (7)$$

For simplicity, we have considered here only single keyframes thus far. Note that it is straightforward to extend the above to select a best sub-segment (i.e. key-segment), or series of frames.

3.2. Linear discriminant-based implementation

Linear discriminant analysis (LDA) is another approach to finding discriminative keyframes. Spectral methods have been used with considerable success for indexing text document collections for information retrieval [12]. Linear methods can additionally exploit labelled training data to "shape" the scatter in the reduced dimension space to enhance discrimination. Fisher's linear discriminant is an example of such a technique [9].

After segmentation, $\mathbf{V}$ is partitioned into K non-overlapping sets of contiguous frames, and hence features:

$$\mathbf{V} = \bigcup_{k=1,\cdots,K} \mathbf{Q}_k \ , \tag{8}$$

such that each feature vector v_i is an element of exactly one segment $\mathbf{Q}_k$. For each of the K segments, we compute the mean feature vector, μ_k and find N_k, the number of frames in segment $\mathbf{Q}_k$. Let μ denote the mean feature vector computed for the entire video. Then, define the within-class scatter matrix

$$\mathbf{S}_W = \sum_{k=1}^{K} \sum_{v_i \in \mathbf{Q}_k} (v_i - \mu_k)(v_i - \mu_k)^T \ , \tag{9}$$

and the between-class scatter matrix

$$\mathbf{S}_B = \sum_{k=1}^{K} N_k(\mu_k - \mu)(\mu_k - \mu)^T \ . \tag{10}$$

For a desired reduced rank $D \leq K - 1$, the optimal transformation is

$$\mathbf{W} = \underset{\mathbf{X}}{\operatorname{argmax}} \frac{|\mathbf{X}^T \mathbf{S}_B \mathbf{X}|}{|\mathbf{X}^T \mathbf{S}_W \mathbf{X}|} \ . \tag{11}$$

$\mathbf{W}$ is computed using standard eigenanalysis [9].

$\mathbf{W}$ projects the feature data to the $D \times N$ matrix $\tilde{\mathbf{V}} = \mathbf{W}^T\mathbf{V}$. This projection clusters features from the same segment, while simultaneously separating them from those of other segments. As a result, keyframe selection is as simple as determining the frame whose transformed feature vector is closest to each segment's mean feature vector. The optimal keyframe for each segment is then

$$v_k^* = \underset{v_j \in \mathbf{Q_k}}{\operatorname{argmin}} \|\mathbf{W}^T(v_j - \mu_k)\| \ . \tag{12}$$

LDA is well suited to our two objectives for keyframe selection. The dimension reduction from LDA emphasizes the representative features for each class to cluster frames within each segment. At the same time, the projection transforms the features to help distinguish between the classes. This provides a principled approach for simultaneous dimension reduction and keyframe selection.

3.3. Computational remarks

A key computational consideration is the cost of updating the keyframes as additional videos or images are added to a collection. If additional media is added to an existing collection, it could be desirable to update the keyframes to provide further discrimination. The similarity-based approach induces $O(N)$ complexity, where N is the total number of images (frames), to add an additional row and column to the similarity matrix. The linear discriminant technique is more costly. Because $\mathbf{W}$ is comprised of generalized eigenvectors as noted above, "folding-in" techniques [12] are applicable for updating the analysis. These costs are approximately $O(ND)$. Other computational enhancements are to consider only a subset of all video frames when computing or updating $\mathcal{C}$. An obvious approach is to only use the set of already-chosen keyframes $\{v_k^*\}$ to recalculate $\mathcal{C}$.

4. EXAMPLES

Figure 2 shows the results of discriminative keyframe selection for a collection of seven video segments, taken from a instructional golf video. The video contained several very similar shots, that differed only in slight details. Low-order DCT coefficients were used for the frame parameters, and the cosine distance measure was used. The unweighted measures of (5) and (6) were computed with $\alpha = 1, \beta = 0$ on the left and $\alpha = \beta = 1$ on the right. Thus the keyframes on the left were chosen non-discriminatively, and the keyframes on the right were chosen discriminatively from the same segments. The difference is apparent: the discriminatively-chosen keyframes are distinctly different in six of the seven segments, while the similarity-based method resulted in only four unique keyframes. Both the subtractive and rational measures resulted in identical output. Similar results were obtained using LDA.

5. CONCLUSION

We have presented methods for finding keyframes that are both relatively unique and reasonably similar to the video segments they represent. We expect these methods to be particularly useful for video editing or GUI applications where it is important that keyframes are distinct, so that different segments are not confused.

6. REFERENCES

[1] A. Girgensohn, J. Boreczky, and L. Wilcox, "Keyframe-based user interfaces for digital video," *IEEE Computer*, pp. 61–67, Sept. 2001.

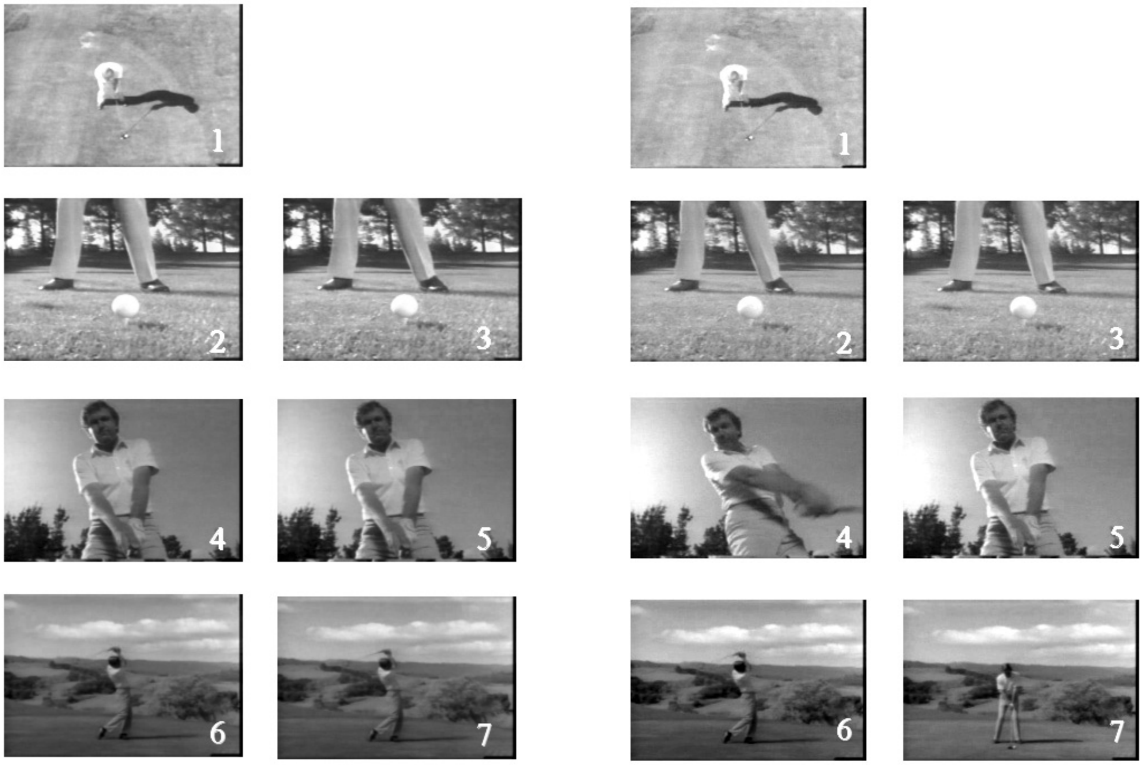

Fig. 2. Experimental results of discriminative keyframe selection. Left: non-discriminative: ($\alpha = 1, \beta = 0$ in (5) and (6)). Right: discriminative: ($\alpha = \beta = 1$)

[2] S. Robertson and K. Jones, “Simple proven approaches to text retrieval,” Tech. Rep. TR356, Cambridge University Computer Laboratory, 1997.

[3] R. Lienhart, “Reliable transition detection in videos: a survey and practitioner’s guide,” *Intl. J. of Image and Graphics*, vol. 1, no. 3, pp. 469–86, 2001.

[4] S. Uchihashi and J. Foote, “Summarizing video using a shot importance measure,” in *Proc. IEEE ICAASP Vol. 6*, 1999, pp. 3041–3044.

[5] M. Christel et al., “Evolving video skims into useful multimedia abstractions,” in *Proc. ACM CHI*, 1998, pp. 171–178.

[6] Q. Huang, Z. Liu, and A. Rosenberg, “Automated semantic structure reconstruction and representation generation for broadcast news,” in *Proc. SPIE Storage & Retrieval for Still Image and Video Databases*, 1994.

[7] H. Aoki, S. Shimotsuji, and O. Hori, “A shot classification method of selecting key-frames for video browsing,” in *Proc. ACM Multimedia*, 1996.

[8] J. Vermaak, P. Perez, M. Gangnet, and A. Blake, “Rapid summarization and browisng of video sequences,” in *Proc. British Machine Vision Conference*, 2002.

[9] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, Wiley-Interscience, 1973.

[10] J. Peltonen, J. Sinkkonen, and S. Kaski, “Discriminative clustering of text documents,” in *Proc. IEEE Conf. on Neural Information Processing*, 2002, pp. 1956–1960.

[11] Y.-I. Ohta, T Kanade, and T. Sakai, “Color information for region segmentation,” *Comp. Graphics & Image Processing*, vol. 13, pp. 222–41, 1980.

[12] M. W. Berry, S. Dumais, and G. W. O’Brien, “Using linear algebra for intelligent information retrieval,” *SIAM Review*, vol. 37, no. 4, pp. 573–595, 1995.