# ADAPTIVE KEY FRAME EXTRACTION USING UNSUPERVISED CLUSTERING

*Yueting Zhuang†∗, Yong Rui, Thomas S. Huang and Sharad Mehrotra*

†Department of Computer Science
Zhejiang University, Hangzhou, 310027, P.R.China
Beckman Institute and Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
E-mail:{yzhuang,yrui,huang}@ifp.uiuc.edu and sharad@cs.uiuc.edu

## ABSTRACT

Key frame extraction has been recognized as one of the important research issues in video information retrieval. Although progress has been made in key frame extraction, the existing approaches are either computationally expensive or ineffective in capturing salient visual content. In this paper, we first discuss the importance of key frame selection; and then briefly review and evaluate the existing approaches. To overcome the shortcomings of the existing approaches, we introduce a new algorithm for key frame extraction based on unsupervised clustering. The proposed algorithm is both computationally simple and able to adapt to the visual content. The efficiency and effectiveness are validated by large amount of real-world videos.

## 1. INTRODUCTION

Recent years have seen a rapid increase in the usage of multimedia information. Of all the media types (text, image, graphic, audio and video), video is the most challenging one, as it combines all the other media information into a single data stream. Owing to the decreasing cost of storage devices, higher transmission rates, and improved compression techniques, digital video is becoming available at an ever increasing rate.

However, efficient access to video is not an easy task due to video's length and unstructured format. Video abstraction and summarization techniques are needed to solve this difficulty [1]. Shot boundary detection and key frame extraction are two bases for abstraction and summarization techniques.

A *shot* is defined as an unbroken sequence of frames recorded from a single camera, which forms the building block of a video. The purpose of shot boundary detection is to segment the video stream into multiple shots[2]. After shots are segmented, key frames can be extracted from each shot. *Key frame* is the frame which can represent the salient content of the shot. Depending on the content complexity of the shot, one or more key frames can be extracted from a single shot[3].

Since effective shot boundary detection techniques exist in the literature [4, 5], we will focus our attention on key frame extraction techniques in this paper. Key frames provide a suitable abstraction and framework for video indexing, browsing and retrieval [1]. They allow users to quickly browse over the video by viewing only a few high-lighted frames. The use of key-frames greatly reduces the amount of data required in video indexing and provides an organizational framework for dealing with video content [5].

Because of its importance, many research effort has been given in key frame extraction [6, 7, 8, 9]. Progress has been made in this area, however, the existing approaches either are computationally expensive or can not effectively capture the major visual content. In this paper we present a clustering based approach which is both efficient and effective.

The reminder of the paper is organized as follows. In section 2, representative related work in key frame extraction is reviewed and evaluated. The proposed clustering based approach is described in section 3. Experimental results over large data set and comparison with existing approaches are given in section 4. Concluding remarks are in section 5.

This work was supported in part by ARL Cooperative Agreement No. DAAL01-96-2-0003 and in part by NSF/DARPA/NASA Digital Library Initiave Program under Cooperative agreement No. 94-11318. ∗Also supported in part by 863 High-Tech No. 863-306-04-03-3 and NSF of China.

## 2. RELATED KEY FRAME EXTRACTION TECHNIQUES

### 2.1. Shot boundary based approach

After the video stream is segmented into shots, a natural and easy way of key frame extraction is to use the first frame of each shot as the shot's key frame [6]. Although simple, the number of key frames for each shot is limited to one, regardless of the shot's visual complexity. Furthermore, the first frame normally is not stable and does not capture the major visual content.

### 2.2. Visual content based approach

Zhang et. al. propose to use multiple visual criteria to extract key frames [7].

- *Shot based criteria:* The first frame will always be selected as the first key frame; but, whether more than one key frame need to be chosen depends on other criteria.
- *Color feature based criteria:* The current frame of the shot will be compared against the last key frame. If significant content change occurs, the current frame will be selected as a new key frame.
- *Motion based criteria:* For a *zooming-like* shot, at least two frames will be selected: the first and last frame, since one will represent a global, while the other will represent a more focused view. For a *panning-like* shot, frames have less than 30% overlap are selected as key frames.

### 2.3. Motion analysis based approach

Wolf proposes a motion based approach to key frame extraction [8]. He first computes the optical flow for each frame [10], and then computes a simple motion metric based on the optical flow. Finally he analyzes the metric as a function of time to select key frames at the local minima of motion. The justification of this approach is that in many shots, the key frames are identified by *stillness* – either the camera stops on a new position or the characters hold gestures to emphasize their importance [8].

### 2.4. Shot activity based approach

Motivated by the same observation as Wolf's, Gresle and Huang propose a shot activity based approach [9]. They first compute the intra- and reference histograms and then compute an activity indicator. Based on the activity curve, the local minima are selected as the key frames.

### 2.5. Summary

The first two approaches to key frame extraction are relatively fast. However, they do not effectively capture the visual content of the video shot, since the first frame is not necessarily a key frame. The last two approaches are more sophisticated due to their analysis of motion and activity. However, they are computationally expensive and their underlying assumption of local minima is not necessarily correct.

Ideally, key frames should capture the semantics of a shot. However, at current stage, the Computer Vision techniques are not advanced enough to automatically generate such key frames. Instead, we have to base key frame selection on low level visual features, such as color, texture, shape of the salient object in a shot. It is obvious that if a frame is important, the camera will focus more on this frame. This is the basic assumption that we use in our clustering based key frame extraction technique. In next section, we will present our proposed approach which is both efficient and effective in key frame extraction.

## 3. CLUSTERING BASED APPROACH

Clustering is a powerful technique used in various disciplines such as Pattern Recognition [11], Speech Analysis [12], and Information Retrieval[13], etc. In [14], an unsupervised clustering based approach was introduced to determine key frames within a shot boundary. In this section, we introduce a different clustering approach to key frame extraction.

Given a video shot $s = \{f_1, f_2, ..., f_N\}$ obtained from a shot boundary detection algorithm [4], we cluster the $N$ frames into $M$ clusters, say, $\sigma_1$, $\sigma_2$, ..., $\sigma_M$. The similarity of two frames is defined as the similarity of their visual content, where the visual content could be color, texture, shape of the salient object of the frame, or the combination of the above. In this paper, we select the color histogram of a frame as our visual content, although other visual contents are readily integratable into the algorithm. The color histogram we used is a $16 \times 8$ 2D HS color histogram in the HSV color space. The similarity between frames $i$ and $j$ is thus defined as:

$$\sum_{h=1}^{16} \sum_{s=1}^{8} min(H_i(h,s), H_j(h,s)) \quad (1)$$

Any clustering algorithm has a threshold parameter $\delta$ which controls the density of clustering. The higher the $\delta$, the more the number of clusters. In human learning and recognition system we also have this threshold. For example, if the threshold is low, we will

classify cars, wagons, mini-vans as vehicles; however, if the threshold is high, we will classify them into different categories. The threshold parameter provides us a control over the density of classification. Before a new frame is classified into a certain cluster, the similarity between this node and the centroid of the cluster is computed first. If this value is less than $\delta$, it means this node is not close enough to be added into the cluster. The unsupervised clustering algorithm can be summarized as follows:

1. Initialization: $f_1 \rightarrow \sigma_1$, $f_1 \rightarrow$ the centroid of $\sigma_1$ (denoted as $c_{\sigma_1}$), $1 \rightarrow numCluster$;
2. Get the next frame $f_i$. If the frame pool is empty, Goto 6;
3. Calculate the similarities between $f_i$ and existing clusters $\sigma_k (k = 1, 2, \cdots, numCluster)$: $sim(f_i, \sigma_k)$, based on Equation(1);
4. Determine which cluster is the closest to $f_i$ by calculating $Maxsim$. Let
   $Maxsim = \max_{k=0}^{numCluster} sim(f_i, \sigma_k)$.
   If $Maxsim < \delta$, it means that $f_i$ is not close enough to be put in any of the clusters, goto 5; otherwise, put $f_i$ into the cluster which has $Maxsim$, and Goto 6.
5. $numCluster = numCluster + 1$. A new cluster is formed: $f_i \rightarrow \sigma_{numCluster}$.
6. Adjust the cluster centroid: Suppose the cluster $\sigma_k$'s old centroid is $c'_{\sigma_k}$, D is the number of frames in it, the new centroid is $c_{\sigma_k}$, thus $c_{\sigma_k} = D/(D+1)c'_{\sigma_k} + 1/(D+1)f_i$. Goto 2.

After the clusters are formed, the next step is to select key frame(s). Here is our strategy: only those clusters which are big enough are considered as *key clusters*, and a representative frame is extracted from this cluster as the key frame. In this paper we say a cluster is big enough if its size is bigger than $N/M$, the average size of clusters. For each key cluster, the frame which is closest to the cluster centroid is selected as the key frame, which captures the salient visual content of the key cluster and thus that of the underlying shot.

## 4. EXPERIMENTAL RESULTS

In the experiments in this section, the video streams are MPEG compressed, with the digitization rate equal to 30 frames/sec. To validate the effectiveness of the proposed approach, representatives of different movie types are tested. In this section, we report the result on two movies: *movie-1*, an action movie and *movie-2*, a romantic movie.

As discussed in Section 3, the threshold parameter $\delta$ controls the density of clustering and thus the density of key frames. The user can therefore control the number of key frames he/she wishes to extract by adjusting the threshold parameter. Table 1 shows the key frame extraction from *movie-1* when $\delta = 0.80$, $\delta = 0.85$, and $\delta = 0.9$, while Table 2 shows that from *movie-2*. All the shots are randomly chosen. we list 8 of them in each example.

Table 1: Examples from *movie-1*

| shot-ID | $\delta = 0.80$ | $\delta = 0.85$ | $\delta = 0.90$ |
|---|---|---|---|
| | K-frames | K-frames | K-frames |
| 1(0-66) | 41 | 41 | 1 34 |
| 2(67-134) | 90 | 68 | 75 |
| 11(641-752) | 676 | 655 679 733 | 662 665 675 698 738 |
| 17(1072-1145) | 1101 | 1074 1102 | 1079 1097 1107 1133 1144 |

Table 2: Examples from *movie-2*

| shot-ID | $\delta = 0.80$ | $\delta = 0.85$ | $\delta = 0.90$ |
|---|---|---|---|
| | K-frames | K-frames | K-frames |
| 1(0-302) | 173 | 173 | 41 |
| 2(303-388) | 367 | 367 | 367 |
| 3(389-439) | 401 | 401 | 401 |
| 11(1113-1402) | 1219 | 1114 | 1117 1267 1348 |

The proposed clustering based key frame extraction approach is not only efficient to compute, it also effectively captures the salient visual content of the video shots. For low-activity shots, it will extract less key frames or one single key frame at most of the time(Table 2) while for high-activity shots, it will automatically extract multiple key frames depending on the visual complexity of the shot(Table 2). Examples of such cases are illustrated in Figure 1. Figure 1 (1)(2) shows the two key frames from shot-17 of *movie-1* because of its visual complexity, while Figure 1 (3)(4) shows the single key frame extraction from *movie-2*.

Figure 2 shows the total number of key frames within each shot in the video. The x-axis corresponds to the shot index, the y-axis to the number of key frames. Figure 2 (1) is for the *movie-1*, (2) for *movie-2*. In most cases, number of key frames within each shot is 1. Obviously the number of key frames in shots from

*movie-1* is bigger than that from *movie-2*. This Figure aslo informs us of which video is low-activity or high-activity. In Figure 2, there are some bins which reach keyframes number as high as 16 to 20. Referring to the inner video, it is found that these parts are the climax of story.

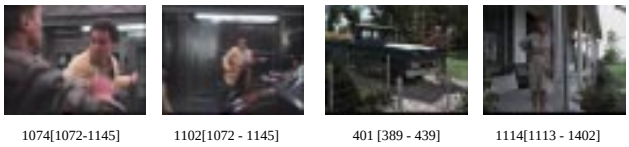


Figure 1: Examples of key frames extraction

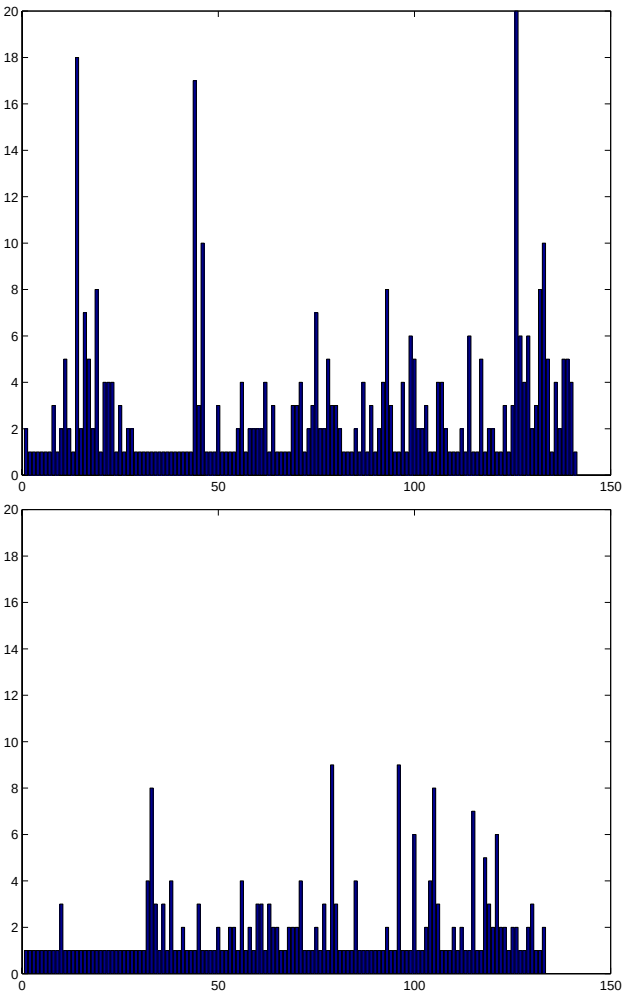

Figure 2: Statistics of two example video clips

## 5. CONCLUSIONS

This paper proposes a novel key frame extraction technique. The contributions and characteristics of the proposed approach are summarized below:

- *Efficiency*: Easy to implement and fast to compute. In [14], it is needed to compute the $\chi^2$ statistic for each frame histogram and the two representatives $H_{avg}$ and $H_{int}$. But in this algorithm, only the comparison between two histograms is necessary.
- *Effectiveness*: Able to capture the salient visual content of the key clusters and thus that of the underlying shot. The key frame selection is based on the number and sizes of clusters; and thus inherently depends on the visual content complexity of the shot. No questionable first frame is selected as the key frame[6]. Rather, multiple key frames will be selected from the complex shots(i.e. high activity), while only single key frame may be selected for the low-activity shots as shown in Figure 1.
- *On-line processing*: this algorithm is easy to be implemented on-line because it only depends on current and previous frames, which is not the case in many of the existing approaches. Figure 3 shows the interface and result of our video retrieval system in WEB-MARS where the key frame extraction is done by CGI program which actually uses this algorithm. Figure 3 (a) is the interface where a client user submits an MPEG file and a task to the server. (b) shows the resulting key frames.
- *Open framework*: Although we use color histogram as the similarity measure, any useful visual or semantic features can be readily integrated into this open framework. We are currently integrating texture feature and close-caption information into our system.

## 6. REFERENCES

[1] P. Aigrain, H. Zhang, and D. Petkovic, "Content-based representation and retrieval of visual media: A state-of-the-art review," *Multimedia Tools and Applications*, vol. 3, Nov 1996.

[2] M. R. Naphade, A. M. Ferman, and et al, "A high performance algorithm for shot boundary detection using multiple cues," in *Proc ICIP*, (Chicago), Oct. 1998.

[3] Y. Rui, T. S. Huang, and S. Mehrotra, "Exploring video structures beyond the shots," in *Proc. of IEEE conf. Multimedia Computing and Systems*, (Austin, Texas USA), June 28-July 1 1998.

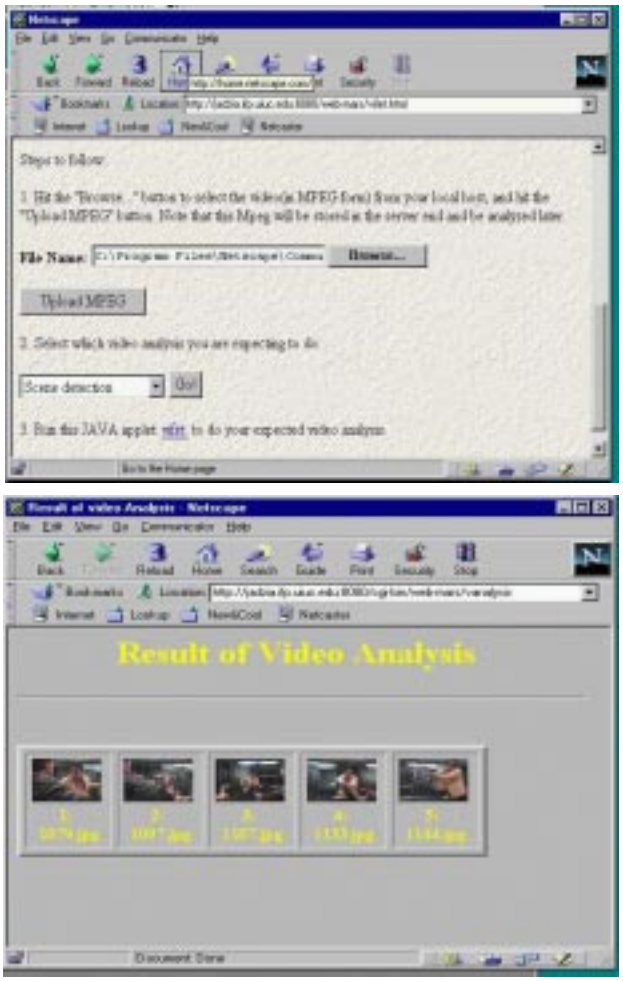


Figure 3: Key frames extraction on-line

[4] J. S. Boreczky and L. A. Rowe, "Comparison of video shot boundary detection techniques," in *Proc. SPIE Conf. on Vis. Commun. and Image Proc.*, 1996.

[5] H.J.Zhang, J. Y. A. Wang, and Y. Altunbasak, "Content-based video retrieval and compression: A unified solution," in *Proc. IEEE Int. Conf. on Image Proc.*, 1997.

[6] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," in *Visual Database Systems II*, 1992.

[7] H. Zhang, J. Wu, D. Zhong, and S. W. Smoliar, "An integrated system for content-based video retrieval and browsing," *Pattern Recognition*, vol. 30, no. 4, pp. 643–658, 1997.

[8] W. Wolf, "Key frame selection by motion analysis," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc.*, 1996.

[9] P. O. Gresle and T. S. Huang, "Gisting of video documents: A key frames selection algorithm using relative activity measure," in *The 2nd Int. Conf. on Visual Information Systems*, 1997.

[10] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[11] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, ch. 6, pp. 211–249. John Wiley and Sons, Inc.

[12] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, ch. 5. Englewood Cliffs, New Jersey: Prentice Hall, 1993.

[13] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company, 1983.

[14] A. M. Ferman and A. M. Tekalp, "Multiscale content extraction and representation for video indexing," in *Multimedia Storage and Archival Systems*, (Dallas, TX), Nov. 1997.