A NEW ALGORITHM FOR SHOT BOUNDARY DETECTION

Yousri Abdeljaoued1, Touradj Ebrahimi1, Charilaos Christopoulos2 and Ignacio Mas Ivars2

1 EPFL Signal Processing Laboratory CH-1015 Lausanne, Switzerland
2 MediaLab, Ericsson Research, Ericsson radio Systems AB, S-164 80 Stockholm, Sweden

e-mail: Yousri.Abdeljaoued@epfl.ch, Touradj.Ebrahimi@epfl.ch, Charilaos.Christopoulos@era.ericsson.se

ABSTRACT
This paper presents a new approach to the detection of shot boundaries. The video is characterized through the tracking of feature points, extracted from the video sequence. The rate of the feature points that are lost or initiated is used as a criterion for shot boundary detection.

1 INTRODUCTION
Recent advances in storage, acquisition, and networking technologies are driving the creation of large amounts of rich multimedia content. However, browsing and retrieval of audiovisual content are still difficult. We focus on video in this paper, as this type of data is one of the richest but also most resource consuming modalities in multimedia content.

The detection of shot boundaries is one of the fundamental tasks in video analysis. After the video is segmented into shots, the extraction of key-frames allows a suitable representation for browsing and retrieval. Many attributes of the frames such as color and motion have been used for shot boundary detection. Histogram-based techniques are shown to be robust and effective [1]. The color histograms of two images are computed. If the Euclidean distance between the two histograms is above a certain threshold, a shot boundary is assumed. However, no information about motion is used. Therefore this technique has drawbacks in scenes with camera and object motion.

The detection of shot boundaries is one of the fundamental tasks in video analysis. After the video is segmented into shots, the extraction of key-frames allows a suitable representation for browsing and retrieval. Many attributes of the frames such as color and motion have been used for shot boundary detection. Histogram-based techniques are shown to be robust and effective [1]. The color histograms of two images are computed. If the Euclidean distance between the two histograms is above a certain threshold, a shot boundary is assumed. However, no information about motion is used. Therefore this technique has drawbacks in scenes with camera and object motion.

Different types of boundaries between shots exist:

- A cut is an abrupt shot change that occurs in a single frame.
- A fade-in starts with a black frame; gradually the image of the next shot appears, brightening to full strength.
- A fade-out is the opposite of a fade-in.
- A dissolve consists in the superimposition of a fade-out over a fade-in.

Early techniques were limited to cut detection

In this paper, a new algorithm for shot boundary detection is presented. It exploits photometric information (texture descriptors), as well as motion information (tracking). This algorithm is able to detect a variety of shot boundaries, including cuts, dissolves, and fades.

The remainder of the paper is organized as follows. Section 2 introduces the new algorithm for shot boundary detection. Section 3 describes the experimental results and Section 4 concludes the paper.

2 SHOT BOUNDARY DETECTION ALGORITHM

The shot boundary detection algorithm consists mainly of three steps (see diagram in figure 1). First, feature points are extracted from each frame [2]. Feature points correspond to points that contain a significant amount of texture, such as corner points. Such points are good candidates for tracking. Then, Kalman filtering is used in order to estimate the locations of the feature points in the current frame. The tracks up to the previous frame are used as input for the Kalman filtering step. A track at time k is defined as a sequence of feature points up to time k that have been associated with the same target.

Since many feature points (multi-target) have to be tracked, a data association filter is required [3]. The nearest neighbor filter is used within this algorithm. In order to validate the association, a set of Gaussian-derivative filters, characterizing the neighborhood of the feature point, is used. The developed tracking algorithm integrates the following capabilities:

- Track initiation: Creation of a new track as a new feature point is extracted.
- Track termination: Removal of a track when its corresponding feature point is no longer extracted.
- Track continuation: Update of a track when its corresponding feature point is extracted.

When many tracks are terminated (for instance in cut, fade-in, dissolve) or initiated (for instance in cut, fade-out, dissolve), the frame is a good candidate for a shot boundary (see figure 2).

We define an activity measure for the rate of change in tracks in order to detect shot boundaries. This activity measure depends on the number of terminated or initiated tracks. We define it as the maximum between terminated and initiated tracks calculated as a percentage. The percentage of initiated tracks is the number
A video sequence consists of a set of successive stationary and nonstationary states of activity. The significant events correspond to the stationary states, which are characterized by a constant or slowly time-varying activity change. On the other hand, shot boundaries correspond to an abrupt change (cut) or fast change (dissolve, fade-in, fade-out). According to these statements, the temporal segmentation algorithm should fulfill the following requirements:

- Detection of abrupt or fast changes.
- Detection of stationary segments.

For this purpose, we make use of a temporal segmentation algorithm, which models the data as a succession of states represented as a Gaussian process. A change in the state corresponds to a change in the parameters of the process (mean $\mu$ and variance $\sigma^2$). The following equations are used in order to update the process parameters:

$$
\mu_i = (1 - \alpha) \mu_{i-1} + a_i \tag{1}
$$

$$
\sigma_i^2 = (1 - \alpha) \sigma_{i-1}^2 + \alpha (a_i - \mu_i)^2, \tag{2}
$$

where $a_i$ is the current activity change, and $\alpha$ is a coefficient acting as a forgetting factor. If $|a_i - \mu_{i-1}| > \sigma_{i-1}$ a new Gaussian process is initialized with the mean equal to the current activity change and a large variance.
Figure 3 shows the activity change and its representation as a model of succession of Gaussian processes. Short impulses correspond to short processes with high activity change (shot boundaries), while longer slow moving segments correspond to stationary states.

3 EXPERIMENTAL RESULTS

The video sequences used for the evaluation of the shot boundary detection algorithm are MPEG compressed movies. This will allow us to test our algorithm on MPEG artifacts due to the compression. The three video sequences (3 x 3000 frames) contain several scene breaks ranging from cuts, which are easy to detect, to more sophisticated video editing effects such as dissolves. The test sequences also contain object and camera motion. To provide a comparison, we have implemented another algorithm for shot boundary detection. It is based on RGB color histogram[5]. Histogram based techniques are widely used for shot boundary detection. The results obtained by the evaluation of both shot boundary detection algorithms are reported in table 1.

The feature points based algorithm performs better in detecting shot boundaries. However, slightly more false alarms are obtained. This is due to the fact that one of the test sequences does not contain a lot of texture, which makes the extraction and the tracking of the corners difficult. The evaluation shows that object or camera motion is a major limitation of the histogram-based approach.

The feature points based algorithm is currently used within the MPEG-7 standardization effort in order to provide an automatic tool for the extraction of the Description Schemes (DS)[6].

4 CONCLUSIONS

A new algorithm for shot boundary detection was presented. It combines both photometric and motion information. The evaluation of the feature points based method proved that this approach outperforms a histogram-based method, especially in a scene characterized by large motion originating from objects as well as from the camera. Video editing effects, such as dissolves, are also detected.

The performance of the presented algorithm depends on the stability of the feature points extraction and tracking steps under different transformations. Therefore we are working on the improvement of these two steps.

References


