Real-time shot detection based on motion analysis and multiple low-level techniques

Carlos Cuevas and Narciso García
Grupo de Tratamiento de Imágenes - E.T.S. Ing. Telecomunicación
Universidad Politécnica de Madrid - Madrid - Spain
{ccr, narciso}@gti.ssr.upm.es

ABSTRACT
To index, search, browse and retrieve relevant material, indexes describing the video content are required. Here, a new and fast strategy which allows detecting abrupt and gradual transitions is proposed. A pixel-based analysis is applied to detect abrupt transitions and, in parallel, an edge-based analysis is used to detect gradual transitions. Both analysis are reinforced with a motion analysis in a second step, which significantly simplifies the threshold selection problem while preserving the computational requirements. The main advantage of the proposed system is its ability to work in real time and the experimental results show high recall and precision values.

Keywords: Temporal segmentation, shot detection, real-time, pixel-based analysis, edge-based analysis, motion analysis

1. INTRODUCTION
The exponential growth of applications dealing with large collections of video data (i.e. digital libraries, distance learning, video-on-demand, digital video broadcasting, interactive TV, multimedia information systems, etc.) has created a need for tools that can efficiently index, search, browse and retrieve relevant material. To do this, indexes describing the video content are required, that should be rich and as complete as possible.

A first step to obtain a video indexing and retrieval of visual data is to perform a temporal segmentation, that is, to divide the sequence into a set of segments (shots) that will be used as basic elements for indexing. A shot is defined as an unbroken sequence of frames taken from one camera. Therefore, it is required to find the location of camera shot transitions, which can be either abrupt or gradual.

Abrupt transitions (also called cuts) are simpler, and occur when the last frame of the first shot is directly followed by the first frame of the second shot. Gradual transitions are the outcome of many processing effects applied to artificially combine two consecutive shots. So, instead of cutting and pasting the second shot after the first one, it is possible to insert an effect: a wipe, a fade, or a dissolve. These transitions are more complex and difficult to detect than cuts.

More than fifteen years of temporal video segmentation research have produced a large variety of algorithms. Among others, several reviews describe and analyze these methods. So, it is possible to find different kinds of techniques that can be classified in one of the next groups: pixel-based methods, histogram-based methods, block-based methods, feature-based methods, motion-based methods, and combinations of different methods.

Methods based on pixel, histogram or block analysis, are usually easy to implement, but they have a number of potential problems, like deciding what threshold is the best. A lot of motion requires a higher threshold value than little motion. Moreover, thresholds should be different for different videos (no particular value may correctly identify all shot changes). A threshold value too low allows to identify shot changes that do not exist, while a threshold value too high results in not detected shot changes.

Methods based on motion analysis or methods that combine multiple strategies, obtain better results (less misclassifications and less false detections), but are significantly more complex and slower than the previous methods. For example: in the strategy proposed in all kind of transitions are detected combining three main techniques: a histogram analysis, object segmentation and a wavelet analysis; in dissolves are analyzed using a multi-pass approximation to the linear behavior of the mean and the semi-parabolic behavior of the variance;
the proposal in\textsuperscript{9} detects shot changes analyzing features extracted from motion vectors and using a block-based motion search; and in \textsuperscript{10} abrupt and gradual transitions are identified analyzing high-level features between consecutive images to solve fast motion and noise problems. These algorithms obtain good results over different kind of sequences while working in unfavorable situations (similar background in consecutive shots, significant content changes, camera motion, moving objects, or illumination changes). Nevertheless, some video editing systems\textsuperscript{11} provide more than 100 different types of edits and, for this reason, it is very difficult to correctly identify all types. Anyway, these algorithms are usually very complex, and this complexity increases the associated computational cost. In addition, some of these strategies require threshold selection and preserve the previously mentioned problem with thresholds.

Therefore, there is still a need for fast algorithms, where the requirements for fast execution (real time) determines the maximum number of allowed operations, thus limiting the application of many known techniques. So, in this paper, a new and fast shot detection strategy which allows the detection of both abrupt and gradual transitions is proposed. A pixel-based analysis is applied to detect cuts and, in parallel, an edge-based analysis is used to detect gradual transitions. Both processes are reinforced with a motion analysis in a second step, carried out only over the previously detected transitions.

Usually the threshold selection is an important problem in typical low level systems for temporal video segmentation. However, in the proposed system, the added motion analysis allows significant reductions in this dependence. Besides, while a motion analysis usually imply high computational requirements, the pre-selection of candidate segments, obtained from the pixel and edge analyses, results in a significant temporal reduction.

The paper is organized as follows: In Section 2 a block diagram describing the architecture of the system is presented. Section 3 details the pixel-based analysis for the abrupt transitions detections and the edge-based method to detect the gradual transitions. The motion-based strategy is described in Section 4. Finally, Sections 5 and 6 contain the results and the conclusions respectively.

2. ARCHITECTURE

The proposed strategy combines two low-level techniques and a motion-based analysis for the detection of both abrupt and gradual transitions. Figure 1 shows a block diagram containing a detailed description of the processing phases in the proposed system. The input is a set of images of a sequence \(\{I^1, I^2, \ldots\}\) that are fed to two parallel processing lines: one for abrupt transitions detection, and the other for gradual transitions detection.

To detect the abrupt transitions, first, a pixel-based analysis is carried out. Comparing the intensity level of the pixels for consecutive images, a set of possible cuts, \(\{C_r(I^{r_1}, I^{r_2})\}\), is obtained, where \(r\) is the number of the detected cuts which is delimited by the pair of images \(\{I^{r_1}, I^{r_2}\}\), where \(r_2 = r_1 + 1\). In a second step, motion analysis is applied to filter the correct ones, discarding false detections. The result of this motion analysis is the final set of abrupt transitions, \(\{C_r(I^{r_1}, I^{r_2})\}\).
For the gradual transitions detection, in a first stage an edge-based analysis is performed on every image as it is received. On the results of this analysis, an evaluation begins when a shot change is defined (i.e. a final abrupt transition has been identified). Possible candidates for gradual transitions are added to the set \( \{ G_s(I^{s_1}, I^{s_2}) \} \), where \( s \) is the number of the detected transition which is delimited by the pair of images \( (I^{s_1}, I^{s_2}) \), where \( s_2 = s_1 + N_s \), being \( N_s \) the number of transition images (typically higher than 10). To separate false detections from correct ones, a motion analysis (similar to that carried out for abrupt transitions) is applied. The result of this analysis is the final pruned set of gradual transitions, \( \{ G_s(I^{s_1}, I^{s_2}) \} \). The output of the system is the union of both abrupt and gradual final sets \( \{ C_r \} \cup \{ G_s \} \).

Although the proposed system is able to work in real time, it has latency as the beginning of the evaluation for the possible gradual transitions requires that the last frame of the cut-limited-shot has been identified. As this latency time depends on the maximum difference between consecutive cut detections, artificial cuts in the edge-based analysis block can be inserted to assure an upper bound of its value.

3. LOW LEVEL TECHNIQUES

3.1 Pixel-based analysis

The first step in the proposed strategy is the application of a fast and efficient pixel-based algorithm to detect abrupt transitions. Usually, the basic idea behind pixel-based approaches for abrupt shot changes is that frames belonging to a single shot are more similar than frames belonging to different shots. Typically, these approaches are based on the computation of the differences between pixel intensity values, placed at the same coordinates of consecutive images. Applying a fixed threshold, they decide whether these differences are high enough to fit to an abrupt transition. Nevertheless, these strategies show to be too sensitive to sudden illumination changes and fluctuations, resulting in false detections when a significant illumination change occurs between consecutive images.

To overcome this limitation, a metric is proposed here, which avoids illumination sensitive pixel to pixel differences. This metric is defined as

\[
M(I^k | I^{k-1}) = \frac{1}{HW} \sum_{i,j} \rho_{i,j}^k,
\]

where \( I^k \) and \( I^{k-1} \) are each pair of consecutive images to be compared; \( H \) and \( W \) are the image height and width; \( (i,j) \) are the coordinates of each one of the pixels in the image; and \( \rho \) can take the value of 1, -1 or 0 depending on the differences between the pixel values and the mean values of the images:

\[
\rho_{i,j}^k = \begin{cases} 
1 & \text{if } |I_{i,j}^k - \mu^k| > T_n \text{ and } \text{sign} (I_{i,j}^k - \mu^k) = \text{sign} (I_{i,j}^{k-1} - \mu^{k-1}) \\
-1 & \text{if } |I_{i,j}^k - \mu^k| > T_n \text{ and } \text{sign} (I_{i,j}^k - \mu^k) \neq \text{sign} (I_{i,j}^{k-1} - \mu^{k-1}) \\
0 & \text{otherwise}
\end{cases}
\]

where \( \mu^k \) is the mean value of the image \( I^k \), and \( T_n \) is a noise threshold (typically \( T_n = 2 \)).

In a global illumination change, between two consecutive images belonging to the same shot, a large amount of pixels change their values. Hence, typical pixel-based methods compute a large amount of differences between the images, and erroneously classify the pair of images as a transition between shots. Nevertheless, this limitation disappears through the application of the proposed metric, which is not based in differences between pixels at the same position in consecutive images. In a illumination change most pixels change in a similar way, decreasing or increasing their values. Hence, if the difference between pixel values and the mean of the image is computed, for each one of the two consecutive images, no significant variations are achieved for pixels at the same spatial position. Therefore, through the proposed metric these situations are correctly treated, resulting in values of \( M \) similar to those obtained in situations where no significant changes occurs. Thus, applying this metric, robustness to global illumination changes is achieved and noisy fluctuations of the pixel values are avoided through the inclusion of \( T_n \).
So, the overall algorithm for abrupt transitions detection is:

- For each new image, $I^A$, compute the proposed metric.
- Comparing the metric values between the images $I^{A-T_w}$ and $I^A$, where $T_w$ determines the minimum number of images between abrupt transitions for a correct identification, check for the local minimum, $m_M(I^G|I^{G-1})$, where $G$ is the position of the minimum between $(A + T_w)$ and $A$, and $(I^G, I^{G-1})$ is the pair of images which difference provides this minimum. Taking into account that is not usual to find more than 1 cut per second, satisfactory results are obtained by selecting $T_w = 20$.
- The pair $(I^G, I^{G-1})$ is added to $\{C_r\}$ with $I^{r1} = I^{G-1}$ and $I^{r2} = I^G$ if:

$$|m_M(I^G|I^{G-1}) - \min \{M(I^i) / i \in [A - T_w, A] \text{ and } i \neq G\}| > T_p$$

(3)

where $T_p$ is a soft threshold that is explained in detail in Section 3.3.

However, situations such as a zoom, pan/tilt camera movements, or large objects moving within the scene could lead to false detections, showing in the evolution of $M$ similar situations to that for real cuts. Figure 2 depicts an example of the obtained results for a sequence with 2000 images and 9 cuts. This figure shows that the resulting values of $M$ are usually close to one, except for some pairs of images where significant minima appear. Most of these minima are caused by abrupt transitions in the sequence (marked with circles in the figure), but some of them (marked with arrows) result from undesirable situations such as abrupt camera movements or fast gradual transitions.

### 3.2 Edge-based analysis

The idea behind the proposed analysis for identifying potential gradual transitions is to compute the evolution of image features, particularly edges, along the video sequences.

In a gradual transition, the first shot gradually disappears while the second one gradually appears. Based on this fact, the evolution of image edges can give us a hint about the gradual transitions presence. Several experiments, over multiple video sequences containing multiple gradual transitions, have demonstrated that, along a gradual transition, the amount of edge-points evolves from those belonging to the previous shot to those in the following one. In between, edge images with strong contributions from both shots are expected to generate...
a local maximum during the transition. Hence, identifying these variations along the sequences, it is possible to locate the gradual transitions between shots.

In the proposed edge-based strategy, the used edge information is the percentage of edge pixels for each image of the sequence: \( N_E(I^k) \). The overall algorithm for gradual transitions detection can be given as follows:

- Compute the percentage of edge pixels, \( N_E(I^k) \), between a given pair of pre-detected cuts.
- Apply a low-pass filter to filter out the noise.
- Normalize \( N_E(I^k) \) to allow the use of a unique threshold.
- Find the set of local maxima, \( \{ M_{NE}(I^k) \} \); each maximum could determine a possible gradual transition.
- For each value (maximum) of \( \{ M_{NE}(I^k) \} \), find its closest previous (left hand) and posterior (right hand) local minima, \( \{ M_{NE}(I^{k-A}) \} \) and \( \{ M_{NE}(I^{k+B}) \} \).
- A new possible gradual transition, defined by \( I_{s1} = I^{k-A} \) and \( I_{s2} = I^{k+B} \), is added to \( \{ G_S \} \) if the following conditions are simultaneously verified:

\[
\begin{align*}
|M_{NE}(I^k) - M_{NE}(I^{k-A})| &> T_E \\
|M_{NE}(I^k) - M_{NE}(I^{k+B})| &> T_E
\end{align*}
\]  

where \( T_E \) is the selected threshold, which is described in Sec. 3.3.

However, the evolution of \( N_E(I^k) \) may exhibit a similar behavior in situations such as zooms, pan/tilt camera movements, camera travelings, or scenes with large moving objects. Figure 3 depicts the evolution of \( N_E(I^k) \) along 650 images between two previously detected cuts. Along this 650 images, 2 gradual transitions, 2 zooms, and 1 camera movement can be identified. The figure shows that important variations in the percentage of pixels have been obtained not only in the gradual transitions but also in the zooms and in the camera movement. Therefore, through the application of the proposed edge-based strategy, these complicated situations can be erroneously classified as gradual transitions. Nevertheless, these false detections will be identified and deleted from the set of gradual transitions, using the motion-based method presented in Section 4.
3.3 Threshold discussion

Threshold selection is an important problem in typically low level techniques, as the two methods previously presented for cuts and gradual transitions. A lot of motion require a higher threshold value than little motion, and should be different for different videos (no particular value may correctly identify all shot changes).

If the values assigned to the thresholds are too low, existing undesirable effects could be classified as transitions between shots. On the other hand, if the thresholds have large values, some existing transitions could be ignored.

For example, Figure 2 shows some possible false detections resulting from significant camera movements and a gradual transition. If the value assigned to the threshold $T_p$ is too low, these undesirable effects could be classified as abrupt transitions. On the other hand, if $T_p$ has a large value, some existing transitions could be ignored.

For the detections of gradual transitions, if $T_E$ is too low, numerous set of images where the percentage of edges changes will result in false detections and, if $T_E$ is too high, some gradual transitions could be ignored.

Other strategies based on low level techniques, assign values to the threshold depending on their objectives. High values allow to obtain most of the transitions, but the number of false detections is increased. Low threshold values result in a small number of false detections, but the number of misdetections grows.

To overcome these limitations, in this work it is proposed to select short enough threshold values for both, pixel and edge analysis ($T_P$ and $T_E$), so that it is ensured that all the existing transitions are detected, although together with a small set of false detections. These false detections will be identified and discarded through the motion analysis presented in Section 4, where the threshold selection is not so crucial.

According on this purpose, satisfactory results have been obtained with values between 0.2 and 0.3 for $T_P$, and values between 0.1 and 0.3 for $T_E$.

4. MOTION-BASED ANALYSIS

The analysis carried out insofar has obtained two sets of transitions (abrupt and gradual), looking for a minimum number of misdetections. On the other hand, many false detections, usually caused by the movement of large objects in the scene or by camera motion, can appear. So, it is required to rule out most of them through motion analysis on the transitions previously detected. This way, computationally expensive motion analysis is restricted to pre-selected pairs of images, i.e. those defining the cuts or the gradual transitions under evaluation.

Through this motion analysis, a set of motion vectors is obtained for each one of the pair of images under evaluation. If this set of vectors shows some spatial and orientation coherence, the vectors may correspond to the motion of the camera or the objects in the scene. Hence, it is possible to identify and separate the false detections from the correct ones.
Figure 5. Pairs of images ($I^A, I^B$) and its associated motion vectors ($V_{A-B}$). First row: correct cut with only one obtained motion vector; Second row: False detection with several short motion vectors; Third row: false abrupt detection with short and large motion vectors; Fourth row: false gradual detection with large low-deviation motions vectors; Fifth row: correct cut with large high-deviation motion vectors.
Figure 4 depicts a flow chart containing a detailed description of the proposed motion-based analysis. For each candidate pair of images, \((I^A, I^B)\), previously classified as a cut or a gradual transition, a motion analysis is carried out. Applying a Lucas-Kanade pyramidal algorithm,\(^{13}\) over this pair of images, a set of motion vectors, \(V_{A-B}\), is obtained. These vectors represent significant features (corners) which have been successfully matched between the two images. To determine if the pair of images denotes a real transition, three different characteristics of this set of vectors are analyzed: the number of vectors, \(N_V\), the amount of large vectors, \(S_V\), and the typical deviation of these large vectors, \(O_V\).

Firstly, the number of vectors, \(N_V\), is computed. If either no vectors are found or their number is very low (typically lower than \(T_V = 4\)), both images likely correspond to different shots as no common features have been identified. Then, the candidate transition is classified as correct. First row of images in Figure 5 contains an example where, from two images correctly classified as an abrupt transition, only one motion vector has been obtained. Thus, this pair of images is classified as a correct transition.

On the other hand, if \(N_V > T_V\), the size of the vectors, \(S_V\), is needed to be analyzed. When large moving objects appear in the pair \((I^A, I^B)\), \(V_{A-B}\) contains a large number of short vectors. Then, if the size of many vectors in the image is considered to be short enough (lower than \(T_V\)), the transition is considered to be a false detection. Images in the second and third rows of Figure 5 depict 2 examples where 2 images, with large moving objects, were erroneously classified as a cut. From the motion analysis between these images, a set of several shot vectors has been obtained. Thus, these images can be discarded from the detections.

Nevertheless, if the number of large vectors is higher than the number of short vectors, to classify the candidate transition as correct or false, the typical deviation of the large vectors, \(O_V\), is analyzed. When fast moving objects or quick camera movements appear between two images, the obtained vectors are large and show similar orientation (\(O_V < T_V\)). In these situations, the transitions are considered to be false detections (see fourth row of Figure 5). On the other hand, if large vectors show random orientations (fifth row of Figure 5), these vectors do not show spatial or orientation coherence (\(O_V > T_V\)) and the transitions are classified as correct.

5. RESULTS AND DISCUSSION

To test the temporal video segmentation system, an analysis of approximately 3.5 hours of video has been conducted. The test video sequences were cartoons sequences and musical movies, looking for the complexity of their gradual transitions and the high amount of camera movements and motion. The video duration, the number of cuts and the number of gradual transitions were annotated and are listed in Table 1.

<table>
<thead>
<tr>
<th>Video</th>
<th>Frames</th>
<th>Cuts</th>
<th>Gradual transitions</th>
<th>Total transitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cartoons</td>
<td>194400</td>
<td>512</td>
<td>28</td>
<td>540</td>
</tr>
<tr>
<td>Musicals</td>
<td>105210</td>
<td>125</td>
<td>44</td>
<td>169</td>
</tr>
<tr>
<td>Total</td>
<td>299610</td>
<td>637</td>
<td>72</td>
<td>709</td>
</tr>
</tbody>
</table>

Table 1. Overview of video durations, number of cuts and number of gradual transitions

To evaluate the performance of the algorithm the conventional \textit{recall} and \textit{precision} evaluation metrics have been used:

\[
\text{recall} = \frac{CD}{N_{\text{total}}}, \quad \text{precision} = \frac{CD}{CD + FD} \tag{5}
\]

where \(CD\) is the number of correct detections, \(N_{\text{total}}\) is the total number of real transitions and \(FD\) is the number of false detections. \textit{Recall} measures the capability in detecting correct shot transitions, while \textit{precision} measures the ability in preventing false alarms. Another measure, the \(Q\) value,\(^{10}\) has been used to evaluate jointly the \textit{recall} and the \textit{precision}:

\[
Q = \frac{\text{recall} \cdot \text{precision}}{100} \tag{6}
\]
As seen in Table 2, the recall and the precision rates are 96.33% and 94.33%, respectively. The results show a high recall and a high precision, for both kinds of videos. As this test material is unduly critical, the results on average material should give higher values, closer to 100%.

According to the architecture of the proposed system, the main reason for missed transitions is that the motion-based analysis has classified a correct transition as a false detection. This occurs when the set of vectors represent a motion that no exists. On the other hand, the reason of false detections comes from the fooling of the motion analysis due to motion vectors with wrong or insufficient information.

The added motion features bring in a considerable improvement to the performance of shot change detection. Moreover, the computational cost is acceptable as the motion-based strategy is applied only over some pairs of images that have been pre-selected through pixel and edge-based analysis.

In order to present the speed of the proposed system, some obtained frame rates, corresponding to sequences with different spatial resolutions, are shown in Table 3. The implementation of the system has been made on a 2.00 GHz Intel Pentium 2 processor with 2GB RAM. The third column in this table contains the frame rate without considering the proposed motion-based strategy. The last column presents the final frame rate, applying the motion analysis. This results demonstrate that the proposed system is able to provide satisfactory results in real-time.

6. CONCLUSIONS

A novel and fast shot detection strategy which allows detecting abrupt and gradual transitions has been presented. A pixel-based analysis is applied to detect cuts and, in parallel, an edge-based analysis is used to detect gradual transitions. Both analyses are reinforced with a motion analysis in a second step, which significantly simplifies the threshold selection problem while preserving the computational requirements.

The main advantage of the proposed system is its ability to work in real time, as the successive application of techniques restricts the use of the computationally expensive stages to few image pairs. Latency can be bounded by the insertion of artificial cuts in the edge-based analysis block.

Experimental results conducted on large data sets show that the proposed system is robust and effective over different kind of sequences and works in unfavorable situations (similar background in consecutive shots, significant content changes, camera motion, moving objects, and illumination changes), obtaining high quality results in real time.

ACKNOWLEDGMENTS

This work has been partially supported by the Ministerio de Ciencia e Innovaci´on of the Spanish Government under project TEC2007-67764 (SmartVision) and by the Comunidad de Madrid under project S-0505/TIC-0223 (Pro-Multidis).
REFERENCES