On the Study of Shot Segmentation in Compressed Domain

Dr. Li-Chien Lin
Feng Chia University
Department of Communications Engineering
Outline

1. overview of the video shot detection
   spatial domain algorithms
   compressed domain algorithms
2. the proposed method
   object-based shot detection in compression domain.
3. results
4. conclusions
The hierarchical structure of a video:

1. video
2. scene
3. shot
4. frame
Why shot detection?

1. indexing : shot classification
2. browsing : library
3. retrieval : key-frame
4. multimedia application
Three kinds of shots: Cut, Fade, Dissolve
Methods of shot detection

Spatial domain algorithms

1. color histogram:
   - global histograms, regional histograms

2. edge change:
   - contrast of object boundaries
   - number of edge points

3. motion:
   - optical flow
Compressed domain

1. DCT values:
   - DC value, DCT vector

2. motion vectors:
   - motion vector, inter/intra
Compare the color difference:

- Color distances in RGB domain
  1. pixel by pixel
  2. Dominant colors or gray levels
  3. Dominant object: color-based correlation
\[ CHD_i = \frac{1}{N} \sqrt{\sum_{r=0}^{2^B-1} \sum_{g=0}^{2^B-1} \sum_{b=0}^{2^B-1} \left| p_i(r, g, b) - p_{i-1}(r, g, b) \right|} \]

if \( CHD_i \leq T_{th} \) no transition

else \( CHD_i > T_{th} \) the shot boundary detected

problem:

It is ineffective for the fade and dissolve
**Edge-based**

**Edge change ratio (ECR)**

\[
ECR_n = \max\left( \frac{\chi_n^{in}}{\sigma_n}, \frac{\chi_n^{out}}{\sigma_{n-1}} \right)
\]

\(\sigma_n\): the number of edge pixels in frame n

---

Figure 1: Typical ECR patterns for hard cuts, fades and dissolves
The comparison
in the spatial domain algorithms:

1. The most reliable method for cut detection is the color histogram. On contrary, the edge-based algorithm is more effective for the fade and dissolve.

2. Both methods suffice from the computation complexity.

3. The performance of Edge-based methods is greatly influenced by the global motion or large object motion.

4. Results of state-of-the-art algorithms:
   - cut: ~ 95%
   - gradual transition: ~ 70%
MPEG Algorithms

![MPEG Algorithm Diagram](image-url)
I frame:
  DCT coefficients

P frame:
  motion vectors
  residual values

B frame:
  forward and backward prediction
  vectors
  residual values
1. DCT coefficients in the I frame:
   dc value comparison.
   the comparison of DCT vector which is constructed by 16 DCT coefficients.

2. Motion vectors in B frame:
correlation of two frames is evaluated by the statistics of # of prediction forward or backward vector
\[
(N_{forw}, N_{back}) < T
\]
3. Motion-prediction statistic

\[
\min \left( \frac{1}{\min \left( \frac{N_{\text{forw}} + N_{\text{bidir}}}{N_{\text{total}}}, \frac{N_{\text{back}} + N_{\text{bidir}}}{N_{\text{total}}} \right)} \right)
\]

only use for B frame

4. Intra/Inter 比例法

\[
\frac{N_{\text{int ra}}}{N_{\text{int er}}} > T
\]

only use for P frame
The comparison in the compression domain algorithms:

1. DCT coefficients are still very effective for the detection of hard cut. However, one can’t access the DCT coefficients in P and B frame.

2. Motion vectors are very unreliable for the detection of shot detection.

3. Too small amount of data in the compressed domain for the shot detection, especially in the detection of gradual transition.

Solution: Partially restore the spatial information.

**dc coefficients of each block.**
System I

Video

static/dynamic

over ½ frame

Intra detect residual value

shot detection

不滿 1/3 frame

DC value residual value

static

motion detection

motion choose

background choose

dynamic
Background choose

Motion value: Camera motion, Object motion

Frame size 176*144 (QCIF)

Sample interval 30 frames

Sample range 1/2~1/3 block number
1. static/dynamic detection
2. motion detection
3. motion choose

<table>
<thead>
<tr>
<th></th>
<th>B</th>
<th>B</th>
<th>B</th>
<th>B</th>
<th>B</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>f</td>
<td>f</td>
<td>f</td>
<td>f</td>
<td>f</td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>f</td>
<td>F</td>
<td>F</td>
<td>f</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>f</td>
<td>f</td>
<td>F</td>
<td>F</td>
<td>f</td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>B</td>
<td>f</td>
<td>f</td>
<td>f</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>B</td>
</tr>
</tbody>
</table>
1. over 1/2 block number
   Intra block detection
   residual value choose

2. under 1/3 block number
   unstable motion
   residual value choose
   Intra block detection
Results of background block selection
1. cut detection:
   based on the number of intra-block for the selected background.

2. Fade and dissolve detection:
   based on the curve of average gradient value for the selected background along time axis:
   
   - Fade: a consecutive number of frame with zero value.
   - Dissolve: a concave hyperbolic curve along time axis.
Sobel mask

\[
\begin{array}{ccc}
Z_1 & Z_2 & Z_3 \\
Z_4 & Z_5 & Z_6 \\
Z_7 & Z_8 & Z_9 \\
\end{array}
\hspace{1cm}
\begin{array}{ccc}
-1 & -2 & -1 \\
0 & 0 & 0 \\
1 & 2 & 1 \\
\end{array}
\hspace{1cm}
\begin{array}{ccc}
-1 & 0 & 1 \\
-2 & 0 & 2 \\
-1 & 0 & 1 \\
\end{array}
\]

\[
G_x = (Z_7 + 2Z_8 + Z_9) - (Z_1 + 2Z_2 + Z_3)
\]

\[
G_y = (Z_3 + 2Z_6 + Z_9) - (Z_1 + 2Z_4 + Z_9)
\]

Gradient value = \[|G_x| + |G_y|\]
Average gradient for Fade

We can detect it by $a=0$
Average gradient for Dissolve

\[ \nabla I = y = ax^2 + bx + c \]

\[ \nabla I = y = a(x + b')^2 + c' \]

we can detected it by \( a > 0 \)
Average gradient within a same shot
## Experimental Results

Recall \( \frac{\text{Detects}}{\text{Detects} + \text{Miss}} \)  
Precision \( \frac{\text{Detects}}{\text{Detects} + \text{False}} \)

<table>
<thead>
<tr>
<th>Video</th>
<th>Recall (cut)</th>
<th>Precision (cut)</th>
<th>Recall (fade)</th>
<th>Precision (fade)</th>
<th>Recall (dissolve)</th>
<th>Precision (dissolve)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test data</td>
<td>93.3% (28/30)</td>
<td>96.6% (28/29)</td>
<td>95.6% (22/23)</td>
<td>100% (23/23)</td>
<td>83.3% (25/30)</td>
<td>80.6% (25/32)</td>
</tr>
<tr>
<td>동영상 데이터</td>
<td>92.9% (78/84)</td>
<td>94.0% (78/83)</td>
<td>88.9% (8/9)</td>
<td>72.7% (8/11)</td>
<td>82.3% (14/17)</td>
<td>70% (14/20)</td>
</tr>
<tr>
<td>Average</td>
<td>93.0%</td>
<td>95.5%</td>
<td>93.75%</td>
<td>91.1%</td>
<td>83.0%</td>
<td>75%</td>
</tr>
</tbody>
</table>

Precision : dissolve (fade)
Comparison with compressed domain algorithms

Algorithm A: Use dc coefficient to evaluate the frame

different proposed by: B. L. Yeo and B. Liu, “A Unified Approach to Temporal Segmentation of Motion JPEG and MPEG Compressed Video”

Algorithm B: Use dc coefficient histogram to evaluate the frame


<table>
<thead>
<tr>
<th></th>
<th>Recall (cut)</th>
<th>Precision (cut)</th>
<th>Recall (dissolve)</th>
<th>Precision (dissolve)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithm_A</td>
<td>82.4%</td>
<td>81.2%</td>
<td>32%</td>
<td>26%</td>
</tr>
<tr>
<td>Algorithm_B</td>
<td>85.6%</td>
<td>43.2%</td>
<td>16.7%</td>
<td>29.4%</td>
</tr>
<tr>
<td>Proposed method</td>
<td>93.1%</td>
<td>97.1%</td>
<td>83.0%</td>
<td>75%</td>
</tr>
</tbody>
</table>
Some problems of the proposed method

1. Illumination problem:
   Nonuniform illumination will cause the large difference in the DC and gradient value.

2. Fast motion:
   We have the problem to select the background block when the large motion occurs such as zooming or panning.

3. The similar background:
   When the consecutive shots has the similar background, then the proposed method has the difficulty to detect the shot boundary.