ABSTRACT

This paper presents an edge-based semantic classification of sports video sequences. The paper presents an algorithm for edge detection, and illustrates the usage of edges for semantic analysis of video content. We first propose an algorithm for detecting edges within video frames directly on the MPEG format without a decompression process. The algorithm is based on a spatial-domain synthetic edge model, which is defined using interrelationship of two DCT edge features: horizontal and vertical. We then use a multi-step approach to classify video sequences into meaningful semantic segments such as “goal”, “foul”, and “crowd” in basketball games using the “edgeness” criteria. We then show how an audio feature (“whistles”) can be used as a filter to enhance edge-based semantic classification.

1. INTRODUCTION

MPEG video is now widely used in many multimedia application areas, since it offers an attractive low-cost possibility for the storage and transmission of digital video data. Conventional processing of digital video requires the video to be de-compressed which is an additional overhead. To avoid this problem, we operate directly on the compressed data. In this paper, we first present a fast algorithm for detecting edges of an object directly in MPEG compressed video. Our algorithm is motivated by the result of [1] and is based on pre-defined synthetic edge models [2]. The proposed algorithm provides better accuracy due to the pre-definition of all possible edges.

Edge information has been used for applications, such as face recognition, and content-based image and video retrieval [3]. In content-based video retrieval systems, edge information can be used as a feature for indexing and retrieval. The advantage of using low-level features is that they are applicable to generic video data. However, a major limitation of such approaches is that the features are not always adequate to retrieve semantically meaningful content. Therefore, it is important to have techniques for automatic extraction/classification of video sequences representing high-level semantics.

There have been some recent attempts to exploit domain knowledge and inherent properties of low-level features for automatic detection of high-level concepts in MPEG sports videos. Most of them attempt to relate or map the low-level information measured from video data to high-level concepts [4]-[6]. Yoshitaka et al. [4] use spatio-temporal correlations of objects to detect a certain semantic content, which are commonly performed by soccer players, namely “wall pass”, “overlap”, “though pass”, and “zone press”. Saur et al. [5] present a method that use the low-level information available directly from MPEG compressed video of basketball as well as prior knowledge of basketball video structure to provide high-level content analysis like “close-up views”, “fast breaks”, “steals”, etc. Nepal and Srinivasan [6] present temporal models for detecting goal segments in basketball videos. These approaches use motion-based visual features such as pan and zoom along with other audio-visual features (but not edge information). In this paper, we present how edge information, directly extracted from MPEG domain using our proposed algorithm, could be used to classify video sequences into high-level semantic categories.

The contributions of the paper can be summarized as follows:

• We propose an algorithm for edge detection directly from MPEG videos using pre-defined synthetic edge models in Section 2.
• In Section 3, we propose a technique of classifying video into sequences, representing high-level semantics, using detected edge information.

2. EDGE DETECTION ALGORITHM

MPEG video consists of three basic frame types: I- (intra-coded), P- (predictive coded), and B- (bi-directional predictive coded) frames. Each I-frame is divided into 16x16 macroblocks (MBs) and each MB consists of four 8x8 luminance (Y) blocks and two 8x8 chrominance (Cb and Cr) blocks. Each block is transformed into a DCT block, which consists of one DC and 63 AC coefficients. A P-frame is predictively coded with a past reference frame while a B-frame requires a future and past frames together for its prediction. We use only the AC
coefficients of Y blocks in I-frames for an edge detection algorithm.

We first define the two edge features and our spatial edge models, and then show how these two edge features are used to derive edges defined in the edge models. The horizontal and vertical edge features can be distinctly formed by the two-dimensional DCT of a block. The edges in an 8x8 block can be represented by two edge feature sets [2]:

\[
\begin{align*}
\text{Horizontal feature: } & H = \{ H_i : i = 1, 2, \ldots, 7 \} \\
\text{Vertical feature: } & V = \{ V_j : j = 1, 2, \ldots, 7 \}
\end{align*}
\]  

(1)

where \( H_i \) and \( V_j \) correspond to the DCT coefficients \( F_{u,0} \) and \( F_{0,v} \), for \( u, v = 1, 2, \ldots, 7 \), in Eq. (2), which describes the AC coefficients of DCT:

\[
F_{u,v} = \frac{2}{\sqrt{MN}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} x_{ij} \cos \left( \frac{(2i+1)\pi u}{2M} \right) \cos \left( \frac{(2j+1)\pi v}{2N} \right)
\]

(2)

where \( u = 1, 2, \ldots, M-1 \), and \( v = 1, 2, \ldots, N-1 \). For an 8x8 block, \( M = N = 8 \).

Based on horizontal and vertical edge features, we have defined some typical spatial-domain synthetic edge models, shown in Fig. 1.

![Fig. 1. Spatial-domain synthetic edge models for the 8x8 block size (From the top line: Horizontal, vertical, 45-degree diagonal (HIGH-LOW), 45-degree diagonal (LOW-HIGH), 135-degree diagonal (HIGH-LOW), and 135-degree diagonal (LOW-HIGH) edges).](image)

A total of 84 edge models are defined: 14 horizontal, 14 vertical, 28 45-degree diagonal, and 28 135-degree diagonal. We have used these models to analyse their corresponding DCT coefficients whose values are given by Eq. (2). Fig. 2 shows DCT coefficients of the first column (1-6) of edge models shown in Fig. 1. Only those DCT coefficients of Fig. 2 that correspond to horizontal \((H_i)\) and vertical \((V_j)\) features in Eq. (1) are used in our edge detection algorithm. In Fig. 2, the boxed DCT coefficients indicate the vertical and horizontal edge features.

To determine edge features defined in the edge models, the following tests are performed on horizontal and vertical features:

\[
\sum_{i=1}^{7} |H_i| \geq \text{Thr}_{\text{horizontal}}
\]

(3a)

\[
\sum_{j=1}^{7} |V_j| \geq \text{Thr}_{\text{vertical}}
\]

(3b)

If the tests in Eqs. (3a) and (3b) are “true” and “false”, respectively, it is defined that the block contains a vertical edge. For a horizontal edge in the block, the tests have to be “false” and “true”, respectively. If both tests are “true”, the block contains a diagonal edge and it is further tested to determine its orientation using the polarities of the first coefficients: \(H_i\) and \(V_j\). The detail of diagonal edge tests is described in reference [7].

![Fig. 2. Corresponding DCT coefficients for the first column (1-6) of the spatial-domain edge models in Fig. 1.](image)

The result of our edge detection algorithm on a test frame of a basketball video is shown in Fig. 3. For further semantic analysis, we use the “edgeness” criteria, based on edge images, such as the example shown in Fig. 3(b).

### 3. SEMANTIC ANALYSIS

This section describes a technique for identifying high-level semantic content, using edge information detected in Section 2. Our semantic analysis is based on the edge complexity, which we call “edgeness” that is defined as the total length of edges in the frame. We then define a
semantic $s$ on a video sequence as a function of edgeness and duration as follows.

$$s = f(\text{edgeness}, \text{duration})$$

where duration is the number of frames where the edgeness remains within a certain threshold.

We use a two-step approach for classifying video sequences into meaningful segments. The first step is a coarse-level semantic classification based on “edgeness”. The second step is a fine-level semantic classification based on duration. Let

- $e_i$ = edgeness of the $i^{th}$ I-frame,
- $N$ = number of I frames in the video,
- $\mu = \text{mean edgeness over the video} = \frac{\sum_{i=1}^{N} e_i}{N}$,
- $\sigma_{\text{avg}} = \text{average deviation from the mean} = \frac{\sum_{i=1}^{N} \text{abs}(e_i - \mu)}{N}$.

We first define three coarse-level semantics of frames based on edgeness as follows.

- High: if $e_i > \mu + \sigma_{\text{avg}}$
- Medium: if $\mu + \sigma_{\text{avg}} >= e_i >= \mu - \sigma_{\text{avg}}$
- Low: if $e_i < \mu - \sigma_{\text{avg}}$

Fig. 4 shows a course-level classification of a 24-second basketball video. The first step segments the video into groups of three different edgeness sequences: Low, Medium and High. We define the following rules to provide meaning to different types of “edgeness” based on empirical investigations on sports videos:

- If the “edgeness” is high, then the frame represents a “crowded scene”
- If the “edgeness” is medium, then the frame represents a “normal play scene”
- If the “edgeness” is low, then the frame represents a “close-up scene”.

Our empirical investigations consisted of observing basketball video clips, automatically categorizing the “edgeness” into three levels, “Low”, “Medium”, and “High”. Video sequences of each edgeness level were then reviewed and annotated with appropriate textual descriptions to describe events in the video clip. The terms used are subjective words to describe visual events. The observation results are shown in Table 1.

![Fig. 3. (a) Original frame of a basketball video, (b) Result of edge detection on it](image)

![Fig. 4. The result of coarse-level classification in a basketball video](image)

Table 1. Results of applying coarse-level classifications on a basketball video.

<table>
<thead>
<tr>
<th>Sports</th>
<th>Coarse Level Semantic</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basketball</td>
<td>Low</td>
<td>Close-up views of players or coaches or umpires</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td>Normal play</td>
</tr>
<tr>
<td></td>
<td>High</td>
<td>Crowd, players with crowd on background</td>
</tr>
</tbody>
</table>

In the second step, we incorporate temporal domain knowledge to create semantic associations. Let us consider an example of a “Low” edgeness segment. In a basketball video, the low edgeness indicates “close-up views”. Our observation shows that there are different types of “close-up views” depending on the events that occur. These close-up views can be categorized based on the temporal duration of the event:

- **Goal Close-up Views**: A basketball game gains momentum when there is a successful field goal. Such successful goals are marked by “close-up view” of the scoring player. Such close-up views last for a short duration as another player immediately takes control of the ball and resumes normal play. This leads to a heuristic that a close-up view with short duration indicates a “goal close-up view”.

- **Foul Close-up Views**: In basketball, the play halts when a player commits a foul. Such fouls are marked by “close-up views” of the player who commits the foul. Such close-up views last for a longer duration. The heuristic here is that a close-up view with long duration indicates a “foul close-up view”.

- **Other Close-up Views**: Apart from above two types of close-up views, the other common close-up views are close-up views of coaches, spectators (known public figures such as Bill Clinton), etc. Duration of such close-up views are hard to classify. However, we notice that due to the fast nature of the game such close-up views usually occur during the breaks.
We classify close-up views into “Goal close-up view” and “Foul close-up view” using temporal rules as follows:

1. If the duration of the edgeness remains “low” for less than 2 seconds, then the video sequence is classified as a “goal close-up view”.
2. If the duration of the edgeness is greater than 2 seconds, then the video sequence is classified as “foul close-up view”.

Fig. 5 shows the low edgeness sequences of the same basketball video (shown in Figure 4). As can be seen, most of the low edgeness segments are of longer duration, which indicates “foul close-ups”. Our automatic analysis shows that there are a total of 19 low edgeness sequences: 10 fouls, 1 goal and 8 others. Out of 19 low edgeness sequences 11 have duration greater than 2 seconds, of which 7 are foul close-up views and 4 are others. The correctness of the edgeness and its duration-based classification is about 65%.

Fig. 5. A video segment of Fig. 4 from frame 10000 to 20000 with only low edgeness sequences.

In order to further improve the classification of the longer duration close-up views we use an audio feature, “whistle”.

In a basketball game, the umpire blows a whistle when a player commits a foul. We use an audio analysis tool, called MPEG Maate [8], to determine whistles from an audio segment of the MPEG videos. Our whistle detection algorithm uses subband energy. The accuracy of the whistle detection algorithm is about 95%. Identifying “whistles” is outside the scope of this paper and is described in reference [8]. This algorithm detects some non-whistle segments as whistle segments due to the shrill sounds in crowd cheers. Such video sequences are detected as “crowded scene” using the edgeness criteria. Here, the whistle feature can be used to filter out “foul close-up views” from “other close-up views”.

In Fig. 5, the arrows indicates the “close-up views” that have duration greater than 2 seconds followed by a “whistle” within 2 seconds. This filters out all 7 foul close-up views out of 11 close-up views with duration greater than 2 seconds. Thus, introduction of an additional feature, such as a whistle, as a filter increases the foul close-up classification rate from 65% to 100%. Thus, we conclude that edgeness feature, when used in conjunction with other multi-modal features, is useful to classify video sequences into meaningful semantic segments.

4. CONCLUSIONS

We have described two ideas in this paper: an efficient edge detection technique for MPEG videos, and the use of edgeness in identifying semantically meaningful segments.

We have defined two feature sets of DCT coefficients and used them to detect edges in MPEG video. Our algorithm relies on simple tests using DCT edge features, and is consequently very fast while it offers some visual accuracy. Rather than rely entirely on the accuracy a fully automated edgeness detection system, we have demonstrated a two-level classification and filtering approach to classify video sequences into high-level concepts using presence of edge information for certain temporal interval. In the first step, we have used the edgeness criteria and broadly classified video sequences into three different semantic segments. We have then used the duration of edgeness and domain knowledge to further classify each segment into more concrete semantic segments such as “foul” and “goal”. We also show how we can improve the classification rate by using other multimodal features in conjunction with edge information.

5. REFERENCES