Automatic Video Caption Detection and Extraction in the DCT Compressed Domain

Chin-Fu Tsao¹, Yu-Hao Chen¹, Jin-Hau Kuo¹, Chia-wei Lin¹, and Ja-Ling Wu¹²
¹Communication and Multimedia Laboratory,
²Graduate Institute of Networking and Multimedia
National Taiwan University, Taipei, Taiwan, R.O.C.
{cftsao,ulyness,david,wjl}@cmlab.csie.ntu.edu.tw

Abstract
The text in a video frame can help us to understand the semantics of video content directly. Although there are many approaches that can automatically detect and localize text in a video, most of them use the original pixels of an image to find the text regions. In this paper, we present an approach to automatically localize captions in MPEG compressed videos. Caption regions are segmented from background by using their distinguishing texture characteristics. Unlike previously published ones which fully decompress the video sequence before extracting the caption regions or only extract text regions in Intra-(I-) frames, our approach detect and localize caption regions directly in the DCT compressed domain. Therefore, only very small amounts of decoding processes are required. Experiments show that a good caption detection rate can be obtained, and the average recalls of Intra- and Inter-frame detections are 97.77% and 97.84%, respectively.

1. Introduction
As digital technologies advance rapidly, digital video is becoming more and more popular in our daily life. In response to the urgent demands of browsing and retrieving from large video data volumes, various video content analysis techniques using visual, audio, and textual cues have been proposed. These techniques are capable of parsing, indexing, and abstracting massive amounts of video data. One of the key information for indexing video content is the presence of text captions in videos. Texts in a video can be classified as scene text and caption text. Scene text is text that occurs naturally in the 3-D scene and is distorted by perspective projection. Caption text comprises 2-D strings that are embedded in the video frame during the editing stage of content production. Captions in video frames provide highly condensed information about the contents of the video. Thus, they can be used for video skimming, browsing, and retrieval in large video databases. For example, captions in news broadcasts and documentaries usually annotate information on where, when, and who of the reported events.

In fact, it is not an easy task to detect and extract captions from images or videos reliably. The size of characters is variable, and the colors of characters in one frame may not be the same. Characters may change their sizes and locations, and may appear in a cluttered, moving, or changing background. In recent years, there are many algorithms proposed for detecting captions from still images and videos. All of the published algorithms can be categorized as the “uncompressed domain text detection” and the “compressed domain text detection”. As compression technique advances, most digitized images and videos are compressed in order to save storage space and transmission bandwidth. Therefore, there is an emerging trend to extract features directly in the compressed domain [1], [2], [3]. Extracting features directly in the compressed domain can save computation time for decompressing, storage for decompressed file, and processes to extract some characteristics embedded in the compressed domain.

Yeo and Liu [4] extract captions from partially decompressed MPEG videos, where video frames with reduced resolution were reconstructed from the original MPEG sequences by using either the DC components or the DC components plus two AC components. If there are large differences between two successive inter-frames, the appearance of text regions is detected. However, if the caption text appears gradually, the method does not work well. In [5], based on the assumption that caption text usually appears and disappears in Intra-coded blocks, Gargi et al. proposed an approach to filter the video stream on the basis of the numbers of Intra-coded blocks in P- and/or B-frames. However, since abrupt shot changes may also
cause blocks to be intra-coded, Gargi’s approach is applicable only to relatively static and smooth video segments between shot boundaries. In [6], Yu Zhong et al. proposed an algorithm to extract caption text from images by using DCT coefficient. Caption texts in JPEG or MPEG-1, 2 Intra-coded frames can be detected accurately. However, the algorithm lacks for the mechanism detecting caption text in Inter-coded frames.

In this paper, we modify the algorithm of [6] to localize captions in MPEG compressed videos (including both Intra- and Inter-coded frames). The algorithm uses texture information extracted directly from DCT coefficients and motion vectors to localize caption texts in Intra- and Inter-frames, respectively. This paper is organized as follows: In sections 2 and 3, details of the proposed methods to localized caption texts in I-, P- and B-frames of MPEG compressed videos are addressed, respectively. Experimental results are presented in section 4. Finally, in section 5, we conclude the paper and address the directions of our future work.

2. Intra-Frame Caption Detection

In this section, the algorithm for Intra- (I-) frame caption detection is introduced, which follows closely to the approach given in [6]. The detection process comprises the following three stages: 1) block-based horizontal intensity variation testing, 2) block-based vertical intensity variation filtering, and 3) refinement by using binary morphological operations. Fig. 1 shows the adopted processing procedure for I-frame caption detection.

2.1. Overview of the MPEG Compression Standard and Characteristics of DCT Coefficients

First, we introduce the main conception of MPEG compression standard. Block-based motion compensated prediction and 8x8 discrete cosine transform (DCT) are basic techniques used in the MPEG standard. The choice of 16 by 16 macroblocks for motion compensation is a trade-off between the provided coding gain and the needed overhead. Each macroblock can be of 4-different types: Intra, Forward Motion Compensation, Backward Motion Compensation, and Bidirection Motion Compensation. Intra macroblocks are coded using only intra-frame information. Forward and Backward Motion Compensation macroblocks are temporally referred to by the previous and future pictures, respectively. Bidirection Motion Compensation macroblocks are coded with a reference that is an average of a previous and a future picture. The quantized DCT coefficients can be readily extracted from a video stream. Although they are quantized, the rank information is preserved and we can use them to compute texture features without involving any other decoding procedure.

2.2. Processing Procedures for Intra-Frame Caption Detection

I-frame caption detection method comes from the fact that a text line which consists of characters will have a high response to the horizontal and vertical harmonics, that is, characters in a text line will introduce the rapid changes in intensity. As shown in [6], the AC coefficients of the horizontal and vertical harmonics can be used to capture the horizontal and vertical intensity variations, caused by characters, within a text line.

Figure 1. The adopted processing procedure for I-frames caption detection.

Intra- (I-) frames, Predicted- (P-) frames and Bidirection- (B-) frames are all considered in the MPEG standard. The P-frame may contain Intra, Forward Motion Compensation, and skipped macroblocks. The B-frame may consist of all types of macroblocks. But there are only Intra macroblocks in the I-frame.

For the purpose of random accessing and editing, the structure of group-of-pictures (GOP) has been introduced. A GOP consists of several frames and the number of frames in a GOP is denoted by N, if this number is fixed in a bitstream. Also, if the number of frames between every I and P frames or P and P frames is fixed, the number is denoted by M.

As addressed in [6], values of DCT coefficients capture the local periodicity and directionality of the spatial image according to the following facts.

- The DCT coefficient values, which are computed based on an 8 x 8 spatial input, capture local image features.
- The values of DCT coefficients, which are amplitudes of harmonic waves, denote the relative amount of various 2D spatial frequencies contained in 8 x 8 blocks. Therefore, they can be used as measures of spatial periodicity and directionality.

The quantized DCT coefficients can be readily extracted from a video stream. Although they are quantized, the rank information is preserved and we can use them to compute texture features without involving any other decoding procedure.

2.3. Processing Procedures for Intra-Frame Caption Detection

I-frame caption detection method comes from the fact that a text line which consists of characters will have a high response to the horizontal and vertical harmonics, that is, characters in a text line will introduce the rapid changes in intensity. As shown in [6], the AC coefficients of the horizontal and vertical harmonics can be used to capture the horizontal and vertical intensity variations, caused by characters, within a text line.
2.2.1. High Horizontal Intensity Variation Testing.

For each 8 x 8 DCT block (i, j), the horizontal text energy (defined by the summation of the absolute amplitudes of the horizontal harmonics of the block) is computed as:

\[ E_{\text{hor}}(i, j) = \sum_{v_1=v_{1v}}^{v_2} |C_{v_1}(i, j)| \]  

(1)

where \( v_1 \) and \( v_2 \) are selected according to the size of the characters been located. In our experiments, we set \( v_1 = 1 \) and \( v_2 = 7 \). Those regions with \( E_{\text{hor}} \) larger than a given threshold are marked as candidate regions for captions. Figs. 2 and 3 show the origin frame and the frame after horizontal intensity variation testing, respectively.

![Figure 2. The Original frame.](image)

![Figure 3. Caption detection results after horizontal intensity variation testing.](image)

2.2.2. Refining Text Blocks Using Vertical Filtering.

The resulting text candidate blocks are further refined using the vertical intensity variation filtering. This refinement is based on the observation that characters are arranged next to each other horizontally, and form more than one text line.

For each 8 x 8 DCT block (i, j), we compute the vertical text energy by summing up the absolute amplitudes of the vertical harmonics of the block:

\[ E_{\text{ver}}(i, j) = \sum_{u_1=u_{1v}}^{u_2} |C_{u_1}(i, j)| \]  

(2)

where \( u_1 \) and \( u_2 \) are empirically determined parameters. We use \( u_1 = 1 \) and \( u_2 = 7 \) in our algorithm. These vertical text energy values are then used to filter the candidate blocks obtained in section 2.2.1. If the block’s vertical text energy is greater than a threshold, we set it as a text block candidate. The threshold is set to the average texture energy of the corresponding DCT channel for all the blocks in the image. This further filtering procedure can remove most of the noise blocks. Unfortunately, after the vertical intensity variation filtering, some of the detected text candidate blocks will be detached or disconnected (due to their spacing between characters or words). Fig. 4 illustrates the frame after applying the vertical intensity variation filtering to Fig. 3.

![Figure 4. Caption detection results after vertical intensity variation filtering.](image)

2.2.3. Further Refinement.

In this stage, we apply morphological operations to remove the noise blocks which occurred in the last two stages, and also to merge the nearby disconnected text blocks into coherent regions. We apply a closing operation to fill the holes between detached text blocks. An opening operation is used afterward to smooth the image and remove the spot-like noise blocks caused by the horizontal and vertical tests. Currently, we use a structured element of size 1 x 3 pixels. The element allows the text blocks to be merged into horizontal lines. Most of the isolated noisy blocks will be removed, and the nearby disconnected text blocks will be merged into coherent regions in this stage. Fig. 5 shows the results after applying closing and opening operations to the image shown in Fig. 4. The results after dilating the output of closing and opening operations by one block are shown in Fig. 6. Some of the operational snapshots after applying the proposed processing procedure for the I-frames are given in Fig. 7.

![Figure 5. Caption detection results after applying the closing and opening operation.](image)

![Figure 6. Caption detection results after dilating the previous result by one block.](image)
3. Inter-Frame Caption Detection

In this section, the proposed P- and B-frames caption detection algorithm is addressed. In inter-frame caption detection, we sometimes need to reconstruct the DCT coefficients for the purpose of increasing the detection rate. Unlike the algorithm for I-frames, we use dequantized DCT coefficients (instead of quantized DCT ones) to discriminate the background and the caption texts in the P- and B-frames.

3.1. Selecting Texture Features

We use the DCT coefficients grouping scheme, as shown in Fig. 8, to conduct the inter-frame caption detection. For each 8 x 8 DCT block (i, j), we compute the text energy by summing up the absolute amplitudes of the vertical (V), horizontal (H) and diagonal (D) harmonics of the block, where V, H and D groups are illustrated in Fig. 8.

DCT blocks whose $E_{AC}^{ij}$ values are greater than a threshold $T$ are highlighted. By analyzing the test videos data, the $T$ value is empirically set to be 2000, in our experiments. Fig. 9 shows computed total AC energy, $E_{AC}^{ij}$, of two horizontal lines (i.e. Row1 and Row2) for 3 different test videos. From Fig. 9, it allows that the variations (peaks and valleys) of $E_{AC}^{ij}$ truly indicated the locations of the captions of the video.
3.2. P- and B-Frame Caption Detection

Since the P-frame may contain Intra, Forward Motion Compensation, and skipped macroblocks, in our P-frame caption detection algorithm, we first check if there is any 8x8 block inside the macroblocks that was intra-coded or not. For each 8x8 intra-coded block, we apply the same algorithm used in the I-frame to detect captions. For all non-intra-coded 8x8 blocks, we will make use of the motion vector information, which can be directly extracted from a compressed video stream (c.f. Fig. 10), to do caption detection. The motion vector information is used to check whether the reference block is a text block or not. If the reference block is a text block, we treat the current block as a text block. Otherwise, the current block is claimed to be a non-text block. From preliminary experimental results, this scheme leads to high miss rate. In order to get higher precision rate, further modification is required. First, we use DCT to reconstruct the dequantized DCT coefficients. That means we transform 8x8 pixels of the inter-coded blocks to the DCT domain, directly. Reference [7] addressed an alternative way to calculate DCT coefficients of inter-coded blocks directly in the compressed domain. This approach owns the advantage of low computation complexity, and therefore, is adopted in our work. After the reconstruction, we conduct the prescribed block-based intensity variation tests on the reconstructed DCT coefficients (c.f. Fig. 10). In the case of skipped blocks, we just check whether the reference block is a text block or not. If the reference block is a text block, we then treat that the current block as a text block (c.f. Fig. 10).

After performing these steps, we applied the prescribed binary morphological operations to refine the result. Fig. 10 shows the flow chart of the proposed algorithm. Some of the experimental results, by applying the proposed caption detection algorithm, are shown in Fig.11 and Fig.12.

3.3 Character Extraction

It is difficult to directly extract characters from a DCT compressed image, thus, the decompression processing is required. The decompression is done only for the localized text regions. This helps to reduce processing time in comparison with fully decompression. We first applied the contrast adjusting operation to whole image. This adjusting operation makes the boundaries and backgrounds more clear. A 3x3 median filter is then used to remove noises. Coarse characters are extracted by using the thresholding technique, detailed in the next sub-section.
threshold due to the fact that texts always have high contrast with respect to its backgrounds. Fig.13 shows the analysis results of threshold value decision and the corresponding extraction results.

Figure 12. Snapshots of B-frame results after applying the proposed caption detection algorithm.

4. Experimental Results and Discussions

To examine the effectiveness of the proposed algorithm, an experiment was conducted on a corpus of TV news programs captured from local TV channels directly, in which a video is with moving credits and static background (MC_SB), a video with moving credits and moving background and two movie videos are with other variations (c.f. Table 1). The total length of the test videos is about 3 hours. We adopt precision and recall to measure the efficiency of our algorithm. Where

\[
\text{precision} = \frac{\text{number of hits}}{\text{number of hits} + \text{number of false alarms}}
\]

and

\[
\text{recall} = \frac{\text{number of hits}}{\text{number of hits} + \text{number of misses}}
\]

Table 1. The test video data set

<table>
<thead>
<tr>
<th>Sequence Name</th>
<th>Total Frames (L, P, B)</th>
<th>Sequence Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movie title and credits_1</td>
<td>(20, 20, 97)</td>
<td>Moving Credits with Static Background</td>
</tr>
<tr>
<td>Movie title and credits_2</td>
<td>(29, 29, 142)</td>
<td>Moving Credits with Moving Background</td>
</tr>
<tr>
<td>Movie title and credits_3</td>
<td>(74, 376, 000)</td>
<td>Static Credits with Static Background</td>
</tr>
<tr>
<td>Movie title and credits_4</td>
<td>(50, 285, 080)</td>
<td>Static Credits with Moving Background</td>
</tr>
<tr>
<td>TVBS-N news</td>
<td>(83, 340, 850)</td>
<td>News Clips</td>
</tr>
<tr>
<td>FTV news</td>
<td>(83, 352, 839)</td>
<td>News Clips</td>
</tr>
</tbody>
</table>

Two different experiments are conducted separately, including Intra- and Inter-frame caption detections. The average recalls of Intra- and Inter-frame caption detections are 97.77% and 97.84%, respectively. Tables 2 and 3 show the results in more detail. Fig. 14 demonstrates some of our caption detection results.

Table 2. Experimental results of I-frame caption detection (SC_SB stands for static credits with static background).

<table>
<thead>
<tr>
<th>Sequence Name</th>
<th>Number of Total Characters</th>
<th>Correctly Detected Characters</th>
<th>False detected Characters</th>
<th>Missed Characters</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC_SB</td>
<td>3121</td>
<td>1194</td>
<td>86</td>
<td>17</td>
<td>98.4%</td>
<td>92.7%</td>
</tr>
<tr>
<td>MC_MB</td>
<td>2804</td>
<td>2788</td>
<td>204</td>
<td>16</td>
<td>99.4%</td>
<td>90.1%</td>
</tr>
<tr>
<td>SC_SB</td>
<td>5358</td>
<td>5192</td>
<td>387</td>
<td>166</td>
<td>96.9%</td>
<td>93.0%</td>
</tr>
<tr>
<td>SC_MB</td>
<td>5751</td>
<td>5735</td>
<td>342</td>
<td>16</td>
<td>99.7%</td>
<td>94.2%</td>
</tr>
<tr>
<td>FTV news</td>
<td>3494</td>
<td>3389</td>
<td>230</td>
<td>135</td>
<td>96.1%</td>
<td>93.6%</td>
</tr>
<tr>
<td>TVBS-N news</td>
<td>2851</td>
<td>3609</td>
<td>241</td>
<td>152</td>
<td>96.3%</td>
<td>93.8%</td>
</tr>
<tr>
<td>Total</td>
<td>22379</td>
<td>21877</td>
<td>1490</td>
<td>592</td>
<td>97.8%</td>
<td>93.6%</td>
</tr>
</tbody>
</table>

Table 3. Experimental results of P- and B-frame caption detections.

<table>
<thead>
<tr>
<th>Sequence Name</th>
<th>Number of Total Characters</th>
<th>Correctly Detected Characters</th>
<th>False detected Characters</th>
<th>Missed Characters</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC_SB</td>
<td>6609</td>
<td>6522</td>
<td>645</td>
<td>117</td>
<td>98.2%</td>
<td>91.1%</td>
</tr>
<tr>
<td>MC_MB</td>
<td>15042</td>
<td>14540</td>
<td>1343</td>
<td>232</td>
<td>98.5%</td>
<td>92.2%</td>
</tr>
<tr>
<td>SC_SB</td>
<td>55628</td>
<td>54936</td>
<td>5231</td>
<td>782</td>
<td>98.7%</td>
<td>91.3%</td>
</tr>
<tr>
<td>SC_MB</td>
<td>62386</td>
<td>62063</td>
<td>4013</td>
<td>363</td>
<td>99.4%</td>
<td>93.9%</td>
</tr>
<tr>
<td>FTV news</td>
<td>35663</td>
<td>34227</td>
<td>2559</td>
<td>1436</td>
<td>95.9%</td>
<td>93.1%</td>
</tr>
<tr>
<td>TVBS-N news</td>
<td>40196</td>
<td>38807</td>
<td>2477</td>
<td>1809</td>
<td>95.4%</td>
<td>93.9%</td>
</tr>
<tr>
<td>Total</td>
<td>216214</td>
<td>211555</td>
<td>16218</td>
<td>4659</td>
<td>97.8%</td>
<td>92.8%</td>
</tr>
</tbody>
</table>
Figure 14. The caption detect results of the sequences (a) MC_SB, (b) MC_MB, (c) SC_SB, (d) SC_MB, and (e) News video, respectively (where the first column are the original sequences and the second column are the corresponding caption detection results).

5. Conclusions and Future Work

An automatic approach for extracting texts from MPEG compressed video streams are presented in this paper. The method has been tested on a large set of videos. The experimental results show quite satisfactory performance. Because of the size limitation (an 8x8 DCT block is too small to represent large characters), we will try to combine neighboring DCT blocks (e.g. combining DC coefficients of neighboring DCT blocks) to compute texture features, and to segment text with a larger font size, in the near future.

6. References