Multimed Tools Appl (2014) 70:1487–1502
DOI 10.1007/s11042-012-1168-z

Video text detection and localization in intra-frames of H.264/AVC compressed video

Xueming Qian · Huan Wang · Xingsong Hou

Published online: 20 July 2012

Abstract Video texts are closely related to the video content. The video text information can facilitate content based video analysis, indexing and retrieval. Video sequences are usually compressed before storage and transmission. A basic step of text-based applications is text detection and localization. In this paper, an overlaid text detection and localization method is proposed for H.264/AVC compressed videos by using the integer discrete cosine transform (DCT) coefficients of intra-frames. The main contributions of this paper are in the following two aspects: 1) coarse text blocks detection using block sizes and quantization parameters adaptive thresholds; 2) text line localization according to the characteristics of text in intra frames of H.264/AVC compressed domain. Comparisons are made with the pixel domain based text detection method for the H.264/AVC compressed video. Text detection results on five H.264/AVC video sequences under various qualities show the effectiveness of the proposed method.

Keywords Text detection · DCT coefficient · H.264/AVC · Integer DCT · AC coefficient · Intra prediction

1 Introduction

Video texts are rich of semantic information, which can be utilized in content based video indexing and retrieval [31, 34, 35, 38, 41]. Text detection is a fundamental step in the text related applications [5, 8, 13–15, 18, 20–28]. Three kinds of characteristics are often utilized in video text detection. The first is the connection characteristics of video texts [8, 10, 20]. The text detection methods based on this characteristic assume that text regions have uniform colors and satisfy certain constraints on size, shape, and spatial layout. The second is the texture alike characteristic of the text regions [13, 14, 23, 26, 28, 42]. The text detection methods based on texture information usually assume that the text regions have special texture patterns. And the third is the edge density information [18, 21]. These

X. Qian (✉) · H. Wang · X. Hou
Xi'an Jiaotong University, Xi'an, China
e-mail: qianxm@mail.xjtu.edu.cn

methods make full use of the fact that the edge densities of background are comparatively sparser than those of the text regions [32, 40]. Usually the corner point number in the text region is larger than that in the background regions.

Video texts can be classified into scene texts and overlaid texts. The scene texts are embedded in video frames. Usually this type of texts can be viewed as a part of a scene. They appear naturally in the scenes which are captured by cameras [3, 26]. The overlaid texts are added during video production [14, 23, 26]. Since this type of texts is purposefully overlaid, they provide important clues for video content analysis and retrieval [15, 28, 31, 34–36].

Video text detection and localization can be carried out both in pixel and compressed domains [1, 4, 5, 18, 21, 26, 27, 33, 35, 39]. DCT coefficients of compressed video can be utilized to represent the block texture and edge information [2, 25, 26, 29, 42]. Macro-block (MB) type information of P- (Predicative) and B- (Bi-directional) frames are also important for text and non-text discrimination [5, 15]. Zhong *et al.* utilized the horizontal DCT texture to detect the candidate text blocks. Then, the candidate text blocks are refined by morphological operations and verified by the vertical DCT texture information [42]. Both the DCT coefficients and the intra-coded MB numbers of B- and P- frames are fused for text detection [5]. In [17], Lu and Barner proposed a text detection method based on the texture information represented by weighted DCT coefficients. Compared to the traditional DCT coefficients based text detection methods [26, 42], the weighted DCT coefficients based text detection method further improves text detection performances. Compared to the DCT based text detection approaches, DWT based approaches are also popular. The DWT based approaches are efficient to transform images into different sub-bands. Texts with different scales are with different responses in different sub-bands. Multi-resolution based text detection methods are often adopted to detect texts with various sizes [4, 18, 39]. By integrating the detected texts in various sub-bands, better performance can be achieved [4, 18]. In order to eliminate false detections, the edge [40], texture [26], and shape information [16] are often utilized in text verifications. In addition, the available redundant temporal information is often used in candidate text region verification and falsely detected text region elimination [5, 18, 21, 26, 27, 33, 35]. Lyu et al. proposed a multi-resolution based text detection method [18]. Firstly, original edge map is generated for each of target video frames. Multi-resolution text maps of a target frame are generated by down-sampling the original text map. Then text detection, verification and localization are carried out on multi-resolution text maps. Finally, the text detection texts in various resolutions are integrated.

In our previous work, we proposed a text detection, localization and tracking approach by utilizing the DCT coefficients of intra frames of the MPEG-1/2 video sequences [26]. It consists of following steps: 1) candidate text blocks (with size 8×8) detection; 2) text line verification and localization; 3) text line matching and tracking [9, 26].

The existing compressed domain based text detection and localization methods are mainly proposed for video sequences compressed by MPEG-1/2, and MPEG-4 Part 2. However, till now, there is no related work reported on the text detection in compressed domain for H.264/AVC compressed video. There are several challenges to detect texts in compressed domain of H.264/AVC, which are summarized as follows: 1) the text detection approach [26, 42] is mainly used for MPEG-1/2 where the block sizes are all 8×8 and non-intra predication is utilized. Thus the DCT coefficients of a block can be utilized to approximate its texture information. However, intra predication is utilized in H.264/AVC which makes the existing works may not applicable for H.264/AVC. Does the texture information of a block can be revealed by the coefficients of residual component after intra prediction? This is the first problem we want to address in this paper. 2) The texture information is approximated by the DCT coefficients, which have some relationship with compressed video quality or bit-rates. As far as we known, existing approaches [26, 42] did not take the compressed video quality into account during text detection. How to make the compressed video text detection approach insensitive to compressed video quality is the second problem we want to address in this paper. 3) In H.264/AVC, the Integer DCT can be carried out with block sizes 4×4 and 8×8. How to represent texture for the blocks with various sizes and incorporate them in text detection, is the third problem we want to solve in this paper.

Thus in this paper, we focus our attention on text detection in H.264/AVC compressed domain by considering the above issues. The rest of this paper is organized as follows. In Section 2, a brief overview of the intra-frame coding and integer DCT in H.264/AVC is presented. In Section 3, a text detection and localization method for the H.264/AVC compressed domain is proposed. Experimental results and discussions are given in Section 4. Conclusions are finally drawn in Section 5.

2 Brief overview of the intra-frame coding and integer DCT of H.264/AVC

To improve coding efficiency, H.264/AVC uses more complicated intra prediction to remove spatial redundancy [6, 12, 19, 37] than the previous video coding standards. For a luma MB, its prediction block may be formed by blocks with sizes 4×4, 8×8, or 16×16. There are 9, 9 and 4 modes for luma blocks with sizes 4×4, 8×8 and 16×16 respectively. According to the relationship of rate and distortion, blocks with large sizes are appropriate for homogeneous areas and blocks with small sizes are beneficial for areas with more details during encoding [12]. Intra prediction of a block is formed by the previously reconstructed blocks in its neighbors. The residual component of a block is the difference of the original block and its optimal prediction. H.264/AVC utilizes integer DCT instead of float DCT. The integer DCT is deduced from float DCT. Now we describe the relationship of them in detial. The float DCT coefficients AC_{uv} of an $N\times N$ sized block $f(x, y)$ are defined as follows.

$$AC_{uv} = C_u C_v \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos\frac{(2x+1)\pi u}{2N} \cos\frac{(2y+1)\pi v}{2N} \tag{1}$$

where u and v (u, v=0, …, $N-1$) denote the horizontal and vertical coordinates respectively. $C_u, C_v = \begin{cases} \sqrt{1/N}, & \text{if} u, v = 0 \\ \sqrt{2/N}, & \text{others} \end{cases}$.

Equation 1 can be rewritten into the format of matrix transform

$$Y = AFA^T \tag{2}$$

where $\boldsymbol{A}$ is the DCT transform matrix, $\boldsymbol{F}$ is the image block and $\boldsymbol{Y}$ is the DCT coefficient matrix. A_{uv} is given by

$$A_{uv} = \begin{cases} \sqrt{\frac{1}{N}} & u = 0, 0 \le \upsilon \le N-1 \\ \sqrt{\frac{2}{N}} \cos\frac{(2\upsilon+1)u\pi}{2N} & 1 \le u \le N-1, 0 \le \upsilon \le N-1 \end{cases} \tag{3}$$

To simplify the implementation and ensure orthogonality of the integer DCT [12, 19, 37], H.264/AVC changes the float point DCT into the integer DCT as follows

$$Y = \left(C_I F C_I^T\right) \otimes E_I \tag{4}$$

where $\otimes$ denotes the operation of direct multiplication and C_I denotes the integer transform matrix. The entities in C_I are all integers. This is why it is called integer DCT. For a 4×4 block, $C_I = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}$ and $E_I = \begin{bmatrix} a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \\ a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \end{bmatrix}$ with a=0.5 and $b = \sqrt{2/5}$. Similarly, the matrices C_I and E_I of an 8×8 block sized integer DCT can be constructed [6, 19].

3 Text detection in H.264/AVC compressed domain

Usually, the overlaid texts have salient structure and high contrast. Most of the blocks in a text region can not be compensated well from their neighbors. Comparatively, the energies of the residual blocks in text region are higher than the energies of the blocks in the background. In this paper, the integer DCT coefficients of the intra-frames of H.264/AVC are utilized to represent block texture intensity and to carry out candidate text blocks detection. The block diagram of the proposed text detection and localization method is shown in Fig. 1, which consists of the following five steps. The first step is entropy decoding and inverse quantization for the luminance component of the intra-frames of the H.264/AVC bit-streams. The second step is block texture representation using the decoded DCT coefficients. The third step is candidate text block detection using block sizes and video quality adaptive thresholds. The fourth step is text block verification using the region related characteristics of video texts. The last step is text line localization using intra-prediction characteristics of the text lines in H.264/AVC intra-frames.

3.1 Block texture representation

In the high profiles of H.264/AVC, each MB of an intra-frame can be partitioned into blocks with sizes 4×4, 8×8 and 16×16 adaptively according to the residual information after intra-prediction. The 4×4 and 8×8 block-sized integer DCT is carried out on the residual of the luminance component. The energy of the residual component of a block can be utilized to measure the effectiveness of intra prediction. Let $Tcoef^N(p,o)$ denote the texture intensity of the (p,o)-th block carrying out $N\times N$ sized integer DCT. In this paper, $Tcoef^N(p,o)$ is represented by the absolute sum of the all the DCT coefficients of a block

$$Tcoef^N(p,o) = \sum_{\substack{0\le u,v\le N-1 \\ u+v\neq 0}} \left|Coef^N(u,v)\right| \tag{5}$$

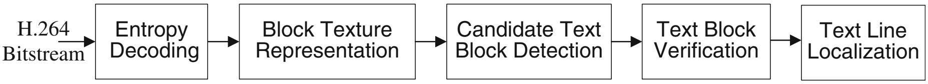

Fig. 1 Block diagram of text detection and localization in intra-frames of H.264/AVC compressed video

where $Coef^N(u,v)$ ($0 \le u$, $v \le N-1$) denotes the integer DCT coefficient at coordinate (u,v) of the residual component of intra-prediction. The quantization parameters influence the texture information very much. When the quantization parameter is larger, more DCT coefficients are likely to be quantized into zeros during encoding. Figure 2 shows the average texture intensity curves of all the non-zero-texture blocks (denoted ALL) and blocks with their texture intensities among the top 5 %, 10 %, and 20 % of each of the intra frames (denoted Top5%, Top10%, and Top20% respectively). The average block texture intensities of all the intra-frames of the blocks carrying out 4×4 sized DCT under quantization parameters QP=5, 15, 25, 35 and 45 of several video sequences are shown in Fig. 2 respectively. In Fig. 2 only several point pairs ($\overline{\mu_N(\alpha)}\big|_q$, q) are given (with q=5, 15, 25, 35 and 45). $\overline{\mu_N(\alpha)}\big|_q$ is the average texture intensity of the blocks (carrying out $N \times N$ sized integer DCT) with their texture intensities among top $\alpha \times 100$ % blocks under quantization parameter q. The value of $\overline{\mu_N(\alpha)}\big|_q$ under any q can be interpolated from the discrete points ($\overline{\mu_N(\alpha)}\big|_q$, q) with q=5, 15, 25, 35 and 45. The fitted curves $\overline{\mu_N(\alpha)}\big|_q$ against q using linear interpolations for α=5 %, 10 %, 20 % and 100 % (i.e. ALL) are shown in Fig. 2 respectively. From Fig. 2 it is obvious that the quantization parameters influence the DCT coefficients based texture intensities significantly. Thus, in compressed domain based video text detection, video qualities (in terms of the quantization parameters) should be taken into account.

3.2 Candidate text block detection

Let *TMAP* denote the corresponding candidate text block map of an intra frame. *TMAP* is a binary matrix. *TMAP*(i,j)=1 expresses that the block (i,j) is a candidate text block, otherwise a candidate background block. In this paper, *TMAP*(i,j) is determined as follows

$$TMAP(i,j) = \begin{cases} 1, & Tcoef^N(i,j) \ge Tcoef^N_{th}(q) \\ 0, & \quad others \end{cases} ; N = 4, 8 \tag{6}$$

where $Tcoef^N_{th}(q)$ is a block sizes $N \times N$ and quantization parameter q adaptive threshold. $Tcoef^N_{th}(q)$ is related to the texture information of an intra-frame. It is calculated as follows

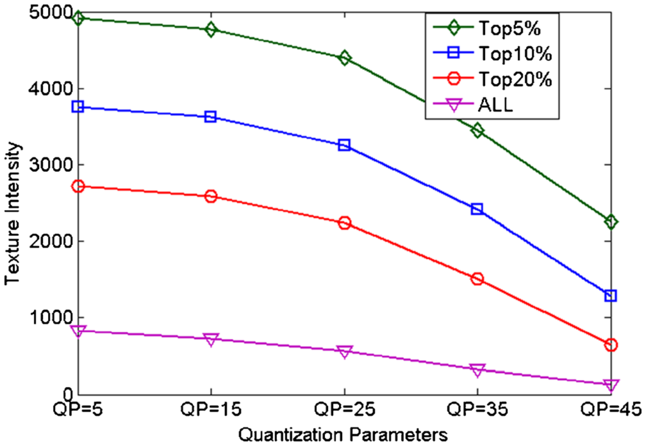

Fig. 2 Block texture intensity curves of all the intra-frames of several video sequences under quantization parameter QP =5, 15, 25, 35 and 45

$$\begin{cases} Tcoef_{th}^{N}(q) = \mu_N(\alpha) + T_N^q \\ T_N^q = \beta_N(q) \times T_N \end{cases} ; N = 4, 8 \tag{7}$$

where $\mu_N(\alpha)$ is the average texture intensity of the blocks carrying out $N \times N$ sized integer DCT with their texture intensities among the top $\alpha \times 100$ % of an intra-frame. T_N^q is a block sizes $N \times N$ and video quality q related threshold. It is utilized to remove the background blocks during coarse text detection. In this paper, T_N is a fixed threshold which is learned from several video sequences under q=25 and α=0.2. We set T_4=2000 and T_8=4000 according to the statistical results of the texture intensities of blocks undergoing 4×4 and 8×8 block sized integer DCT using the following rules:

$$T_N = T_N^q\big|_{q=25} \approx \overline{\mu_N(\alpha)}\big|_{q=25} \tag{8}$$

In Eq. 7, $\beta_N(q)$ under any q can be calculated from the fitted block texture intensity curves as follows

$$\beta_N(q) = \frac{\overline{\mu_N(\alpha)}\big|_q}{\overline{\mu_N(\alpha)}\big|_{q=25}} \times \beta_N(q)\big|_{q=25}; N = 4, 8 \tag{9}$$

Combining $\mu_N(\alpha)$, $\beta_N(q)$ and T_N, $Tcoef_{th}^{N}(q)$ can be determined adaptively.

$$Tcoef_{th}^{N}(q) = \mu_N(\alpha) + \frac{\overline{\mu_N(\alpha)}\big|_q}{\overline{\mu_N(\alpha)}\big|_{q=25}} \times T_N \times \beta_N(q)\big|_{q=25}; N = 4, 8 \tag{10}$$

According to Eqs. 8 and 10 can be rewritten as follows:

$$Tcoef_{th}^{N}(q) \approx \mu_N(\alpha) + \overline{\mu_N(\alpha)}\big|_q \times \beta_N(q)\big|_{q=25}; N = 4, 8 \tag{11}$$

From Eq. 11 we find that $Tcoef_{th}^{N}(q)$ is determined by taken the block sizes $N \times N$ and video quality q into account. The influences of α and $\beta_N(q)\big|_{q=25}$ to the text detection performances are further discussed (please turn to Section 4.4 for details).

3.3 Text blocks verification

Candidate text blocks of an intra frame can be determined according to the adaptive threshold $Tcoef_{th}^{N}(q)$ and the corresponding candidate text map can be obtained as shown in Fig. 3(a).

It is very hard to determine a block is a text block or a background block exactly from the texture information of itself. The region related characteristics of the text are valuable for text block determination [18, 26, 28, 42]. Usually, most of the blocks in text regions are detected as candidate text blocks according to their texture intensities. Then utilizing the region related characteristics, text blocks of a text line can be grouped into connected regions by morphological operations.

Usually most of the background blocks can be effectively compensated from their neighbors during intra prediction. The corresponding texture intensities of the background blocks are comparatively small. It is likely that very small part of the blocks in background regions are classified into candidate text blocks according to Eq. 6. This makes the distribution of candidate text blocks in background region is sparser than the distributions of candidate text blocks in the real text regions. According to the sparse and dense distribution characteristics of candidate text blocks in background and text regions, morphological operations are effective to remove falsely detected background blocks.

$$FMAP(s,t) = CTL(s,t)\&CMAP(s,t); \quad s = 1,\cdots,Wb; \quad t = Ts,\cdots,Te$$

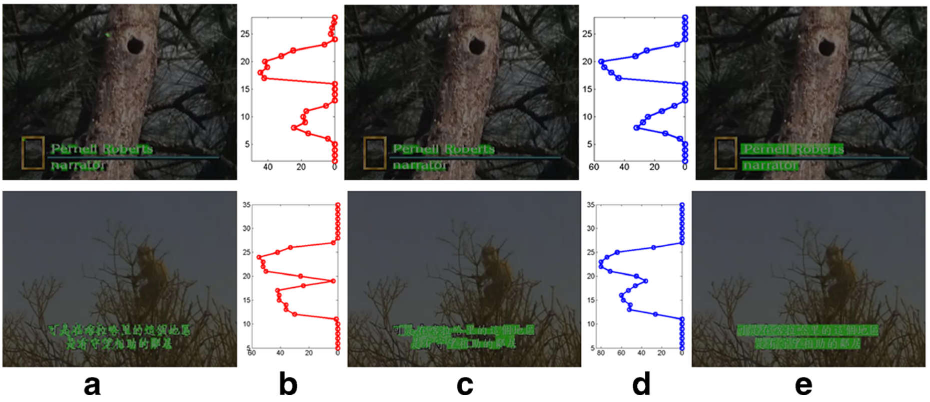

Fig. 3 Two examples of text line localization in intra-frames of H.264/AVC compressed domain. (**a**) candidate text blocks are labeled in green in the original images; (**b**) the texture projection curves of the local text block regions of the *TMAP*; (**c**) text blocks of *OMAP* are labeled in green in the original images; (**d**) the texture projection curves of the local text block regions of the *OMAP*; (**e**) text line localization results. Note that the horizontal and vertical axises of (**b**) and (**d**) are the block numbers and block indexes. This figure is best viewed in colors

Usually, text lines are horizontally and vertically overlaid [26]. In this paper, we focus on the horizontal text verifications. Hereinafter we introduce the corresponding text verification approach by examples as shown in Fig. 3. The detected candidate text blocks are labeled in green in the original intra-frames as shown in Fig. 3(a). The main steps of determining the horizontal text lines are as follows. Firstly, a closing operator with a structure element of sizes $r\times r$ is carried out on the candidate text map *TMAP*. Let *CMAP* denote the corresponding text map after closing operation. Large r is effective to connect the candidate text blocks of real text regions into coherent regions, while the candidate text blocks in background are likely to be grouped into a connected region at the same time. Generally speaking, the large connected regions in background are hard to be removed than the sparse blocks. The closing operation aims at connecting the candidate text blocks in text region and keeping the candidate text blocks in background isolated as much as possible. So, r can not set to be very large, we set $r=3$ in this paper.

After closing operation, the neighboring candidate text blocks in text region are grouped together. At the same time the candidate text blocks in background maybe also grouped into small regions. Then an opening operator with the structure element of sizes $1\times z$ is carried out on *CMAP* to remove the small regions in background and to detect horizontal text lines. To remove the small background regions, we must set $z>r$. if z is large enough, then the connected candidate text blocks in text region is eliminated. So, we set $z=5$ in our experiments. Let *OMAP* denote the corresponding map after this opening operation. This opening operation is effective in removing most noise blocks while keeping the text block regions stand out. Each of the remaining block regions is determined as candidate text region if its block number is large enough.

3.4 Text line localization

Due to the fact that texts are different from background, the blocks in the first row of a text line can not be effectively compensated from their neighbors as shown in the images of

Fig. 3(a). Usually the strokes of characters are irregular, this made blocks in text region can not be compensated very well using intra prediction. Thus the blocks in text region have high probability to be detected as candidate text blocks. This characteristic can be utilized in text line localization. The detailed steps of text line localization are as follows:

1) Determine the connected text region number (denoted *M*) in *OMAP* and get the vertical positions $V_T(i)$ and $V_B(i)$ of each text region, $i=1, \ldots, M$. For example, the text region number of two images in Fig. 3(a) are 3 and 1 respectively.
2) Carry out projections for the *i*-th text region in *TMAP* and *OMAP* respectively from $V_T(i)-k$ to $V_B(i)+k$. We get the corresponding texture projection curves $TP(t)$ and $OP(t)$ as follows:

$$TP(t) = \begin{cases} \sum\limits_{s=0}^{Wb} TMAP(s,t) & t \in [V_T(i)-k, V_B(i)+k] \\ 0 & others \end{cases} \tag{12}$$

$$OP(t) = \begin{cases} \sum\limits_{s=0}^{Wb} OMAP(s,t) & t \in [V_T(i)-k, V_B(i)+k] \\ 0 & others \end{cases} \tag{13}$$

where *Wb* is the block number (in terms of the blocks with sizes 4x4) in the width of an image ($Wb=W/4$). *k* is an offset which is utilized to get accurate starting and ending positions of a text line, *t* is block index. You known, the blocks are text region boundaries can not be compensated well from their neighbors in the background during intra-prediction. In this paper, $k=2$ is enough to get accurate text line positions. The projection curves of the local regions of *TMAP* and *OMAP* are shown in Fig. 3(b) and (d) respectively.

3) From $TP(t)$ and $OP(t)$, the candidate text map of the text lines is determined using minimum text block number constraints as follows

$$CTL(s,t) = \begin{cases} 1 & OP(t) \geq N_{th} \&\& TP(t) \geq N_{th} \\ 0 & others \end{cases}; s = 1, \cdots, Wb \tag{14}$$

If the block number in each row of a candidate text line is less than N_{th} then this row is considered to be background. In this paper we set =5. From Eq. 14 the neighboring text lines with large gaps can be separated. From *CTL* the starting point (denote *Ts*) and ending point (denote *Te*) of a candidate text can be determined.

4) From *CTL* and *CMAP* the final text map *FMAP* is determined as follows

$$FMAP(s,t) = CTL(s,t) \& CMAP(s,t); \quad s = 1, \cdots, Wb; \quad t = Ts, \cdots, Te \tag{15}$$

In order to connect the block regions of a horizontal text line into an integrated one, a closing operation is needed. In this paper, the structure element of the closing operator is set to be 1×11. The starting and ending points of a text line in the up- and bottom- directions can be determined from the text map after morphological operations. Let $Ss(t)$ and $Se(t)$ denote the starting point and ending point of the *t*-th row of a text line in the left- and right-directions. The starting point (denote *SsP*) and ending point (denote *SeP*) of a text line in left- and right- directions are determined as follows:

$$\begin{cases} SeP = \max\limits_{t}\{Se(t)\} \\ SsP = \max\limits_{t}\{Ss(t)\} \end{cases}; \quad t = Ts, \cdots, Te \tag{16}$$

Text line localization results of Fig. 3(a) are shown in Fig. 3(e). We find the text lines in the complex background are well localized. More experimental results are given in the following section.

4 Experimental results and discussions

Five video sequences are used as test set to evaluate the performance of the proposed text detection and localization method. The test set consists of a CCTV news video which is extracted from Chinese news channel (denoted CCTV), two famous document video sequences which are produced by BBC (denoted BBC1 and BBC2), a document video sequence *Piranha* which produced by National Geographic Channel (denoted Fish) and a video sequence download from FTP site of INRIA (denoted Movie [7]). The resolutions of these video sequences are given in the first column of Table 1. They are encoded into H.264/AVC bit-streams by the reference software JM10.2 under various quantization parameters [11]. In order to show the effectiveness of the proposed H.264/AVC compressed domain based text detection and localization method, we compared it with Lyu et al's method (denoted Lyu) [18] and, the DWT transform and K-means based method [30] (denoted DWT). Text detection algorithms of Lyu, DWT and ours are carried out at H.264/AVC decoder side by embedding the text detection algorithms in the reference software JM10.2 in Windows XP environment. The text detection algorithms are run on a PC with PIII 1.8GHz

Table 1 Text line detection and localization performances of Lyu, DWT and ours under α=0.2 and $\beta_N(q)\big|_{q=25} = 1$. The test video sequences are encoded by reference software JM 10.2 under q=25

Test video	Method	*NC*	*NM*	*NF*	*NR* (%)	*NP* (%)	*F*1(%)	spf
Fish (352*288)	Lyu	566	34	13	94.33	97.75	96.01	0.33
	DWT	559	41	38	93.17	93.63	93.40	9.82
	Ours	572	28	5	95.33	99.13	97.19	0.10
CCTV (352*288)	Lyu	619	128	87	82.86	87.68	85.20	0.32
	DWT	591	156	135	79.12	81.41	80.24	9.79
	Ours	604	143	94	80.86	86.53	83.60	0.10
BBC1 (576*432)	Lyu	882	65	18	93.14	98.00	95.51	0.72
	DWT	870	77	27	91.87	96.99	94.36	22.07
	Ours	879	68	0	92.82	100.0	96.28	0.24
BBC2 (576*432)	Lyu	162	13	43	92.57	79.02	85.26	0.69
	DWT	151	24	62	86.29	70.89	77.84	19.32
	Ours	166	9	2	94.86	98.81	96.79	0.22
Movie (352*288)	Lyu	1019	9	205	99.12	83.25	90.49	0.29
	DWT	966	62	276	93.97	77.78	85.11	8.94
	Ours	1018	10	198	99.03	83.72	90.73	0.08

CPU, and 1G RAM. In text detection performances evaluation, the decoding process for the P-, and B- frames and the corresponding chrominance are all skipped. Full decoding for the luminance component of the intra-frames is needed by Lyu and DWT, while ours only needs part decoding to carry out text detection in H.264/AVC compressed video streams. For objective computational costs comparison, the time utilized to decode bit-streams for carrying out text detection is considered to be a part of Lyu and ours.

The DWT based text detection approach extracts features on DWT sub-bands for K-means clustering to differentiate texts from background [30]. The DWT based method computed 21 statistical values on the high-frequency sub-bands, and employed the K-means clustering algorithm to partition all the pixels of an image into either text candidates or background.

4.1 Text detection and localization performance evaluation

We declare a text line is correctly detected if the minimum overlapping ratio (*MOLR*) of the detected text block region (*DTBR*) and the ground-truth text region (*GTR*) is greater than 80 %. *MOLR* is defined by

$$MOLR = \min\left\{\left|DTBR\bigcap GTR\right|/|GTR|, \left|DTBR\bigcap GTR\right|/|DTBR|\right\} \tag{17}$$

The recall *NR*, precision *NP*, and F1 are utilized to evaluate the objective text detection and localization performance as follows

$$\begin{cases} NR = \dfrac{NC}{NC+NM}\times 100\% \\ NP = \dfrac{NC}{NC+NF}\times 100\% \\ F1 = \dfrac{2\times NR\times NP}{NR+NP}\times 100\% \end{cases} \tag{18}$$

where *NF*, *NC* and *NM* denote the numbers of the falsely, correctly and missing detected texts respectively.

The text detection performances of Lyu and ours on the above five video sequences are shown in Table 1. The video sequences are encoded by reference software JM10.2 under $q=25$. We fix the two parameters $\alpha=0.2$ and $\beta_N(q)\big|_{q=25}=1$ in our experiment. The parameters T_4 and T_8 are learned from block texture intensities of the intra-frames of several video sequences excluded from the test set and we get $T_4=2000$ and $T_8=4000$ according to Fig. 2. In our experiments, we find that Lyu is good at detecting the text lines with large captions. This kind of text is often appeared in the news video sequences, such as CCTV. Usually text blocks in the inner region of large characters can be compensated very well from their neighbors during intra prediction. This makes the texture intensities of the blocks in the text region comparatively low. Hence only a small part of blocks in text region are detected as candidate text blocks and the distribution of the detected candidate text blocks in text region is sparse. These candidate text blocks have high probability to be eliminated during morphological operations. The text detection performance of ours is not as good as that of Lyu for the sequence CCTV. However ours is effective to detect the text lines consist of small characters. Since DWT requires the K-means to classify an image into the background and text regions, its performance depends on the similar background texture in the image. Thus, this method is sensitive to the density edges in the background regions. The average F1 values of the five test video sequences of Lyu, DWT and ours are 90.49 %, 86.19 % and 92.92 % respectively.

4.2 Computational costs of Lyu, DWT and ours

Now we give an objective comparison for the computational costs of the Lyu, DWT and ours. Compared with Lyu, ours has low computational cost due to the following three aspects: 1) Inverse DCT and luminance component reconstruction are not required. 2) More computational reduction in text map determination. The text map of ours is generated by using the block texture information. While multi-resolution text maps must be generated by Lyu. 3) The size of our text map is only 1/16 of that of Lyu. The computational costs of ours are lower than that of Lyu in text verification and text line localization. The average computational costs of Lyu and ours for the five test video sequences are shown in the last column of Table 1 respectively. The average processing speeds of Lyu are 0.3 and 0.7 s per frame (spf) for the video sequences with resolutions 352*288 and 576*432 respectively. The average computational costs of DWT based text detection approach are 9.52 spf and 20.70 spf for the videos with resolutions 352*288 and 576*432 respectively. These of ours are 0.1 spf and 0.23 spf respectively. In all, the computational cost of ours is only about 1/3 of Lyu and 1 % of DWT.

4.3 Subjective results of text detection and localization

Figure 4 shows the subjective results of the proposed text detection and localization method. The detected text lines are labeled in green. Text lines in different backgrounds, colors, and sizes are correctly detected as shown in the images in the first and second rows. The images in the third row show some missing and falsely detected texts. The missing detected texts are those with very small characters and the distributions of text blocks are very sparse. Most of the candidate text blocks in text regions are removed during verification. The texts with very low contrasts with backgrounds are not likely to be detected. This can be shown from the second image of the bottom row of Fig. 4, where texts are undergoing fade-in/out [25]. The

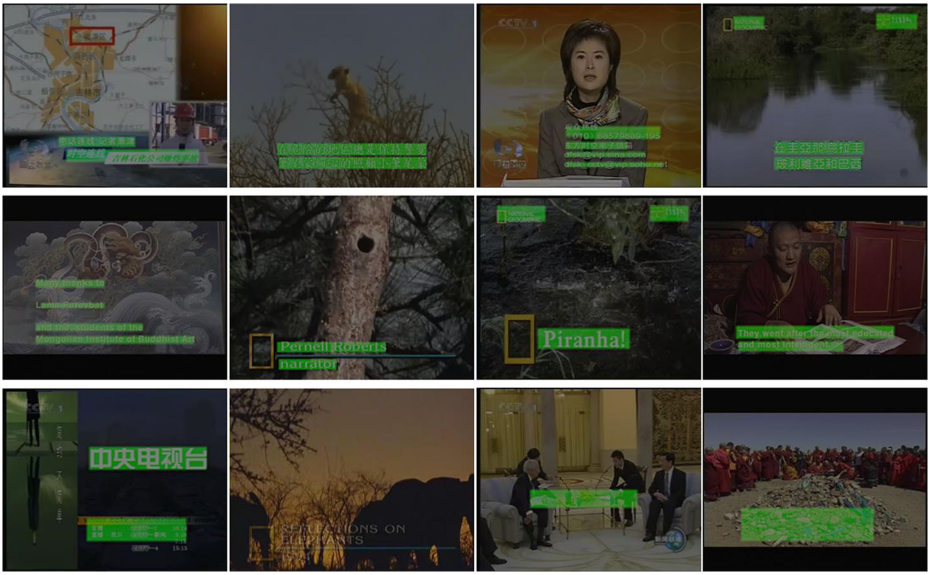

Fig. 4 Subjective results of text detection and localization of the proposed text detection method. This figure is best viewed in colors

falsely detected text lines usually have large texture intensities and most of the blocks have heavy residual during intra prediction.

4.4 Discussions on the selection of α and $\beta_N(q)$

In this section, the corresponding text detection performances of versus α and $\beta_N(q)$ under q=25 are discussed. The recall, precision, and F1 values of our text detection method under $\beta_N(q)\big|_{q=25} = 1$ and α takes five values in the range [0, 0.3] (α=0.05, 0.1, 0.15, 0.2 and 0.3) are shown in Table 2. From this table we find that more text lines are missing detected with a few false detections with the decrease of α. Better performance is achieved under α=0.3. The corresponding text detection performances under α=0.2 and $\beta_N(q)\big|_{q=25}$ takes four values in the range [0, 1] $\left(\beta_N(q)\big|_{q=25} = 1.0, 0.75, 0.5 \text{and} 0.25\right)$ are also shown in Table 2. We find that when α=0.2, the influences of $\beta_N(q)$ to the text detection performances are not very significant. Comparatively, better performances are achieved under $\alpha \in [0.15, 0.3]$ and $\beta_N(q)\big|_{q=25} \in [0.5, 1]$.

4.5 Text detection performances versus video qualities

Video qualities have significant impacts on the block texture information as shown in Fig. 2, thus they may influence the compressed domain based text detection performance. In order to show the effectiveness of the adaptive $\beta_N(q)$ determination method, two video sequences Movie and BBC1 are encoded into several bit-streams by setting the quantization parameters q=5, 15, 25, 35 and 45 respectively. The corresponding text detection performances of $\beta_N(q)$=1 (denoted Hard) and adaptive $\beta_N(q)$ (denoted Adaptive) which is determined by Eq. 9 are compared systematically. Let NR_Adaptive and NR_Hard denote the recall values of the Adaptive and Hard. And let F1_Adaptive and F1_Hard denote the F1 values of the Adaptive and Hard respectively. NR_Adaptive, NR_Hard, F1_Adaptive, F1_Hard versus quantization parameter q (with q=5, 15, 25, 35 and 45) are shown in Fig. 5 respectively. It is clear that better performances are achieved by Adaptive. Comparisons (in terms of NC, NM, NF, NR, NP, and F1) of Lyu, DWT and ours on Movie and BBC1 under q=45 are also shown in Table 3. The average F1 values of Lyu, DWT, Hard and Adaptive are 89.29 %, 86.95 %, 84.24 % and 94.76 % respectively. The F1 value of Adaptive outperforms Lyu and Hard by about 5.5 % and 10.5 % respectively. The average F1 values of Lyu, DWT and ours of the two video sequences under q=25 are 93.12 %, 81.48 % and 93.38 % respectively as

Table 2 Average text detection performance versus α and $\beta_N(q)$ for the five video sequences under q=25

		NC	*NM*	*NF*	*NR* (%)	*NP* (%)	*F1*(%)
$\beta_N(q)\big|_{q=25}$	α=0.3	3373	124	395	95.45	89.52	92.39
	α=0.2	3239	258	299	92.62	91.55	92.08
	α=0.15	3083	414	193	88.16	94.11	91.04
	α=0.1	2941	556	121	84.10	96.05	89.68
	α=0.05	1615	1882	8	46.18	99.51	63.08
α=0.2	$\beta_N(q)\big|_{q=25} = 1.0$	3239	258	299	92.62	91.55	92.08
	$\beta_N(q)\big|_{q=25} = 0.75$	3298	199	382	94.31	89.62	91.91
	$\beta_N(q)\big|_{q=25} = 0.5$	3371	126	506	96.40	86.95	91.43
	$\beta_N(q)\big|_{q=25} = 0.25$	3403	94	896	97.31	79.16	87.30

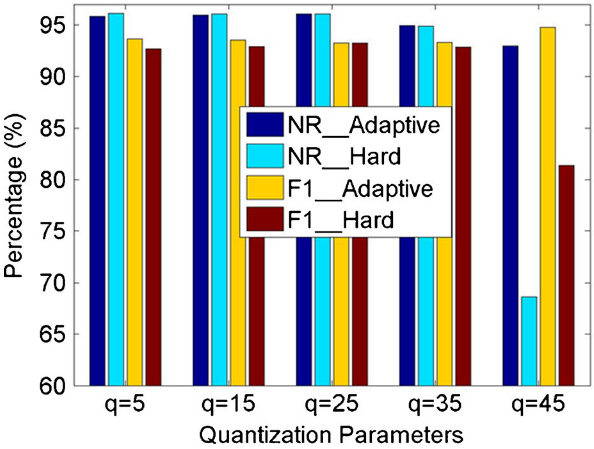

Fig. 5 Text detection and localization performances of Adaptive and Hard under q=5, 15, 25, 35, and 45. NR_Hard and F1_Hard denote the recall and F1 value of Hard. NR_Adaptive and F1_Adaptive denote the recall and F1 values of Adaptive

shown in Table 1. When the quantization parameter q changed from q=25 to q=45, the average F1 values of Lyu and DWT decrease 3.8 % and 5,47 % respectively, while that of ours increases 0.38 %. From the above comparisons, we find that the proposed text detection method is robust to video quality variations.

5 Conclusion

In this paper, an effective text detection and localization method is proposed for intra-frames of H.264/AVC compressed video by utilizing the block integer DCT coefficients. Candidate text blocks are detected by a compressed video quality and block sizes adaptive threshold. Characteristics of texts region in the intra frames of H.264/AVC are analyzed and utilized in text line localization. The H.264/AVC compressed domain based text detection method is robust to the variations of video qualities. Comparisons with pixel domain based method on H.264/AVC bit-streams with various qualities show the effectiveness and robustness of the proposed text detection and localization method.

Table 3 Text detection performances of Lyu, DWT and ours (including Adaptive and Hard) for the video sequences Movie and BBC1 with α=0.2 under q=45

		NC	*NM*	*NF*	*NR* (%)	*NP* (%)	*F1*(%)
Movie	Adaptive	980	48	64	95.33	93.87	94.59
	Hard	607	421	1	59.05	99.84	74.21
	Lyu	758	270	7	73.74	99.08	84.55
	DWT	724	304	28	70.43	96.28	81.35
BBC1	Adaptive	856	91	0	90.39	100	94.95
	Hard	748	199	0	78.99	100	88.26
	Lyu	843	104	3	89.02	99.65	94.04
	DWT	826	121	12	87.22	98.57	92.55

In this paper, experimental results are conducted on five video sequences with various qualities. We only study the cases that texts aligned in either horizontal or vertical direction. We do not address the problems for detecting text with various directions. Moreover, the proposed approach focus on detecting the overlaid text, which maybe not very effective for detecting scene texts. Each overlaid texts usually exist in multi-intra frames. Fusing the multi-frame information should improve text detection performances. Thus in our future work we will focus on above issues.

Acknowledgements This work is supported in part by the National Natural Science Foundation of China (NSFC) Project No.60903121, No.61173109, and Foundations of Microsoft Research Asia.

References

1. Chen D, Bourlard H, Thiran J (2001) Text identification in complex background using svm. In Proceedings of the International Conference on Computer Vision and Pattern Recognition, 2, 621-626
2. Crandall D, Kasturi R (2001) Robust detection of stylized text events in digital video. In Proceedings of the International Conference on Document Analysis and Recognition 865-869
3. Cui Y, Huang Q (1997) Character extraction of license plates from video. In Proceedings of the Conference on Computer Vision and Pattern Recognition 502-507
4. Ekin A (2006) Local information based overlaid text detection by classifier fusion. In *Proc. ICASSP2006*, 2, II753-II756.
5. Gargi U, Antani S, Kasturi R (1998) Indexing text events in digital video databases. In Proc. Int. Conf. Pattern Recognit., 1, 916-918
6. Gordon S (2003) Simplified Use of 8x8 Transform. Doc. JVT-I022, San Diego, Sept. 2003
7. INRIA FTP site. ftp://imedia-ftp.inria.fr//MUSCLE-VCD-2007//DB-MPEG1//Movie23.mpg
8. Jain A, Yu B (1998) Automatic text location in images and video frames. In *Proc. ICPR*, 1497-1499
9. Jiang H, Liu G, Qian X, et al. (2008) A fast and efficient text tracking in compressed video. in Proc ISM
10. Jung K, Kim K, Jain A (2004) Text information extraction in images and video: a survey. Pattern Recognition 37:977–997
11. JVT Reference Software version 10.2. ftp://ftp.imtc-files.org/jvt-experts/reference_software/
12. JVT-G050, 2003. Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264/ISO/IEC 14486-10 AVC. in Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VECG
13. Lee C, Jung K, Kim H (2003) Automatic text detection and removal in video sequences. Pattern Recogn Lett 24:2607–2623
14. Li H, Doermann D, Kia O (2000) Automatic text detection and tracking in digital video. IEEE Trans Image Process 9(1):147–156
15. Lim Y, Choi S, Lee S (2000) Text extraction in MPEG compressed video for content-based indexing. In Proc. Int. Conf. on Pattern Recognit., 4, 409-412
16. Liu Z, Sarkar S (2008) Robust outdoor text detection using text intensity and shape features. in Proc ICPR
17. Lu S, Barner K (2008) Weighted DCT coefficients based text detection. in Proc. ICASSP 1341-1344
18. Lyu M, Song J, Cai M (2005) A comprehensive method for multilingual video text detection, localization, and extraction. IEEE Trans Circuits and Systems for Video Technology 15(2):243–255
19. Malvar H et al (2003) Low-complexity transform and quantization in H.264/AVC. IEEE Trans CSVT 13:598–603
20. Mariano V, Kasturi R (2000) Locating uniform-colored text in video frames. in *Proc. 15th Int. Conf. Pattern Recognit.*, 4, 539-542
21. Ngo C, Chan C (2005) Video text detection and segmentation for optical character recognition. Multimedia Systems 10(3):261–272
22. Qi W, Gu L, Jiang H, Chen X, Zhang H (2000) Integrating visual, audio and text analysis for news video. in *Proc. Int. Conf. Image Process.*, 3, 520-523

23. Qian X, Liu G (2006) Text detection, localization and segmentation in compressed videos. in *Proc. ICASSP2006.*, 2, II385-II388
24. Qian X, Liu G (2007) Global motion estimation from randomly selected motion vector groups and GM/LM based applications. Signal, Image and Video Processing 4:179–189
25. Qian X, Liu G, Su R (2006) Effective fades and flashlight detection based on accumulating histogram difference. IEEE Trans Circuits and Systems for Video Technology 16(11):1245–1258
26. Qian X, Liu G, Wang H, Su R (2007) Text detection, localization and tracking in compressed videos. Signal Processing: Image Communication 22(9):752–768
27. Rainer L, Axel W (2002) Localizing and segmenting text in images and videos. IEEE Trans Circuits and Systems for Video Technology 12(4):256–267
28. Sato T, Kanade T (1998) Video OCR: Indexing digital news libraries by recognition of superimposed caption. *ICCV Workshop on Image and Video retrieval*
29. Shen B, Sethi I (1996) Direct feature extraction from compressed images. in *IS&T SPIE: Storage and Retrieval for Image and Video Databases IV*, 2607, 404-417
30. Shivakumara P, Phan TQ, Tan CL (2009) A robust wavelet transform based technique for video text detection. Int Conf Document Analysis and Recognition, 1285-1289
31. Snoek C, Worring M (2005) Multimedia event-based video indexing using time intervals. IEEE Trans Multimedia 7(4):638–647
32. Sun L, Liu G, Qian X, Guo D (2009) A novel text detection and localization method based on corner response. in Proc ICME
33. Tang X, Gao B, Liu J, Zhang H (2002) A spatial-temporal approach for video caption detection and recognition. IEEE Trans Neural Networks 13(4):961–971
34. Wang P, Cai R, Yang S (2003) A hybrid approach to news video classification with multimodal features. *in Proc. Int. Conf. on Information, Communication and Signal Processing*, 2, 787-791
35. Wang R, Jin W, Wu L (2004) A novel video caption detection approach using multi-frame integration. *ICPR 2004. Proceedings of the 17th International Conference*, 1, 449-52
36. Wang F, Ma Y, Zhang H, Li J (2005) A generic framework for semantic sports video analysis using dynamic bayesian networks. *in Proc. Int. Conf. on Multimedia Modeling*, 115-121
37. Wiegand T, Sullivan G, Bjontegaard G, Luthra A (2003) Overview of the H.264/AVC video coding standard. IEEE Tans Circuits Syst Video Technol 13:560–576
38. Wu W, Chen D, Yang J (2005) Integrating co-training and recognition for text detection. In Proceedings of the International Conference on Multimedia Expo
39. Wu V, Manmatha R, Riseman E (1999) Textfinder: an automatic system to detect and recognize text in images. IEEE Trans Pattern Anal Mach Intell 21(11):1224–229
40. Zhang J, Goldgof D, Kasturi R (2008) A new edge-based text verification approach for video. in Proc. ICPR
41. Zhang H, Wu J, Zhong D, Smoliar S (1997) An integrated system for content-based video retrieval and browsing. Pattern Recognit 30:643–658
42. Zhong Y, Zhang H, Jain A (2000) Automatic caption localization in compressed video. IEEE Trans Pattern Analysis and Machine Intelligence 22(4):385–392

Xueming Qian (M’10) received the B.S. and M.S. degrees in Xi’an University of Technology, Xi’an, China, in 1999 and 2004, respectively, and the Ph.D. degree in the School of Electronics and Information Engineering, Xi’an Jiaotong University, Xi’an, China, in 2008. He was awarded Microsoft fellowship in 2006. From 1999 to 2001, he was an Assistant Engineer at Shannxi Daily. From 2008 till now, he is a faculty member of the School of Electronics and Information Engineering, Xi'an Jiaotong University. He was a visit scholar at Microsoft research Asia from Aug. 2010 to March 2011. His research interests include video/image analysis, indexing, and retrieval.

Huan Wang received the B.S. in Xi'an University of Technology, in 2004, and M.S. degrees in Xi'an Jiaotong University in 2010 respectively. Her research interests include video/image coding, communication and transmission.

Xingsong Hou received the B.S. degree in electronic engineering from North China Institute of Technology, Taiyuan, China, in 1995, and the M.S. degree and Ph.D. degree in the School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an, China in 2000 and 2005, respectively. From 1995 to 1997, he was an Engineer with the Xi'an Electronic Engineering Institute in the field of radar signal processing. Now he is an associate professor of the School of Electronics and Information Engineering, Xi'an Jiaotong University. His research interests include video/image coding, wavelet analysis, sparse representation, sparse representation and compressive sensing, and radar signal processing.