國立交通大學

資訊科學與工程研究所

碩 士 論 文

建構於 H.264 無損幀內編碼的變動長度編碼法

A new VLC in H.264 lossless intra-coding for screen content

研 究 生：蕭成憲

指導教授：蔡文錦 教授

中 華 民 國 101 年 11 月

建構於 H.264 無損幀內編碼的變動長度編碼法
A new VLC in H.264 lossless intra-coding for screen content

研 究 生：蕭成憲　　Student：Cheng-Hsien Hsiao

指導教授：蔡文錦　　Advisor：Wen-Jiin Tsai

國 立 交 通 大 學
資 訊 科 學 與 工 程 研 究 所
碩 士 論 文

A Thesis

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

November 2012

Hsinchu, Taiwan, Republic of China

中華民國 101 年 11 月

中文摘要

視窗內容係指經由電腦或者其他數位裝置所產生之圖像與影音，其實際應用的例子有: 遠端桌面、桌面分享、視訊會議以及遠端教學等等。這些視窗內容通常都包含了大量的文字與圖示部分，然而這些部分並不適合利用當前的圖像壓縮規格以及 H.264/AVC 中的幀內編碼方式進行壓縮。

因此，於本篇論文中我們針對視窗內容中的文字部分，提出一個新的變動長度編碼用於 H.264/AVC 無失真幀內編碼中。我們利用文字區塊之特性，基於適應性變動長度編碼法而設計新的無失真編碼方法，並且利用位元率失真最佳化決定區塊壓縮的模式。實驗結果顯示提出之方法相較於 H.264/AVC 編碼方式不僅能夠降低編碼位元率，並且能夠增進編碼效率以及視覺品質。

關鍵字：視窗內容、無失真幀內編碼、變動長度編碼

ABSTRACT

Screen content image and video were generated by computer or other digital devices in the real applications such as remote desktop, desktop sharing, video conferencing, and remote education. However, screen content normally contains numerous texts and graphic part which are not appropriate to be encoded by the state-of-the-art image compression standards and intra frame coding in H.264/advanced video coding (AVC).

Therefore, in this paper we proposed a new variable length coding in H.264/AVC lossless intra-coding for text part in screen content images or videos. We utilize the characteristics of text blocks to design the new lossless coding method based on context adaptive variable length coding (CAVLC), and determine the block compression mode by rate-distortion optimization (RDO) selector. Experiment results show that, compared to H.264/AVC coding, the proposed method not only reduce the coding bitrates, but also enhance the coding efficiency and visual quality on text content.

Index Terms-- Screen content、 Lossless intra-coding、Variable length coding

誌謝

在這兩年多的研究所生涯中，能完成我的碩士論文，首先最要感謝的就是我的指導教授蔡文錦博士。在學業研究上，孜孜不倦地與我討論各種相關的議題，點出我研究上的盲點，引導我前往正確的方向；在日常生活中，不時的關心我並且給予我前進的力量。在此向我最敬愛的指導教授蔡文錦博士，致上最高的敬意。

我要感謝實驗室的學長姐，吳佳穎、呂威漢、張育誠、謝寧靜、孫域晨，謝謝你們指導我各種研究上的相關知識。另外要感謝我的同學們，黃致遠、楊巧安、林宗翰，謝謝你們陪伴我度過這段追求知識的過程，課程上的互相砥礪，生活中的互相打氣，讓我從你們身上獲益良多。還有謝謝學弟們，王敬嚴、林建儒、胡振達、高彬倬，謝謝你們讓我在研究所生涯中過得更加精彩，祝福你們順利畢業。

最重要的，感謝我的家人及女友，尤其是我的父母親，在背後默默支持我，讓我在疲憊的時候有一個溫暖的避風港，謝謝你們對我的期待及付出。

接下來就要告別學生生涯進入職場了，大家珍重。

謹以此論文獻給我的師長、家人及所有關心我的朋友們

CONTENT

LIST OF FIGURE

LIST OF TABLE

Chapter 1 Introduction

With the development of computer and network technologies, there are more and more images and videos which mainly include screen contents in real application scenarios such as remote desktop, desktop sharing, video conferencing, and remote education. Screen contents normally refer to image and video which are generated by computers or some other digital devices, and they are generally a combination of text, graphics and natural contents For example, web pages, slides, online games, captured screen and so on are kinds of screen contents.

Unlike natural contents, texts and graphics contents are much sharper, with high contrast as well as more sensitive to human eyes. The state-of-the-art image compression standards and intra frame coding in H.264/advanced video coding (AVC) are all designed for natural contents, and these compression standards typically utilize transform and quantization to achieve compression, but for text and graphics contents, quantization after the transform will result in unbearable edge noise on the decoded frames, so these compression standards are not appropriate for text and graphic contents.

Therefore, many approaches have been proposed in the recent years to resolve the compression of text and graphics together with natural contents. In block-based approaches, first segment the images or frames into non-overlapping blocks with certain size, and then for each block, the adopted compression schemes can be classified into two main categories as shown in Figure 1-1.

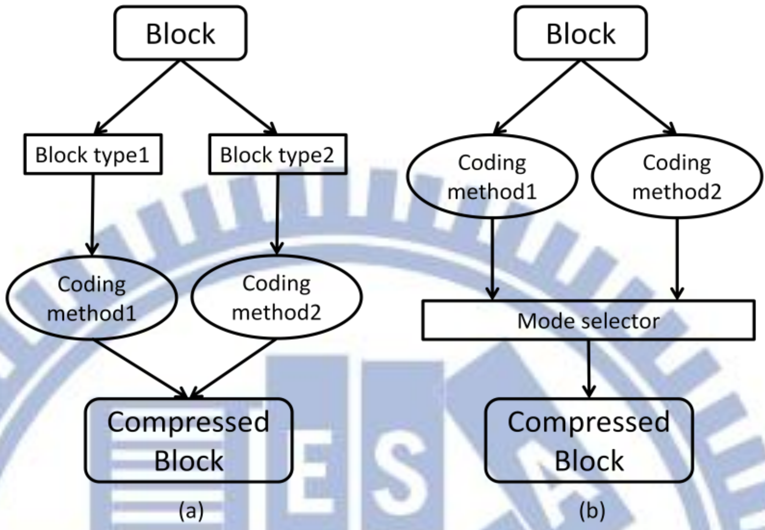


Figure 1-1 Two categories of block-based compression scheme

In Figure 1-1(a), first the block is classified into one of the two distinct block types: text blocks or natural blocks; then according to the block types, use corresponding coding method to encode the block. Thus, how to implement the block classification to distinguish text blocks from natural block is the key process in this compression scheme. In [1], a transform coefficient likelihood (TLC) scheme was proposed, which examines the DCT coefficient values of 8 × 8 blocks for separating the textural and graphical portions of a compound image, which. In [2], the authors analyzed the histograms and gradients of the blocks to classify each 16 × 16 block into one of four types: smooth, text, hybrid and picture blocks based on gradient-histogram distribution. The method in [3] segmented screen images into text/graphic, picture/background blocks by computing the statistical features based on discrete wavelet transform coefficients in the detail sub-bands of each 8 × 8 block.

In Figure 1-1(b), the block is first encoded by distinct coding methods, and then passes through mode selector such as rate-distortion optimization (RDO) in H.264 to choose the best result. In [4], two new lossy modes were proposed, which include residual scalar quantization (RSQ) mode and base colors and index map (BCIM) mode. The method in [5-6] combined gzip lossless coding technique into H.264 hybrid coding, and used macroblock as the basic coding unit. In [4-6], they all applied RDO criterion to select the optimum mode or coder.

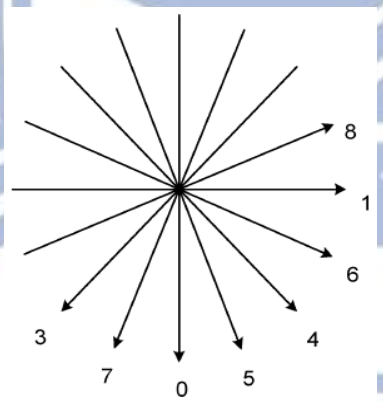
In fact, no matter which category the compression scheme is, the method to encode each kind of blocks properly is the most essential part, especially for text blocks. Typically, coding methods for text blocks can be divided into two categories: lossy [2, 4] and lossless [3, 5, 6] coding. For lossless coding, the quality of text block can be preserved, so there should be no noise on text edges. Therefore, how to reduce the coding bitrates becomes the major issue for lossless coding.

Consequently, in this thesis, a new variable length coding (VLC) is proposed, which is applied on H.264 lossless intra-coding for text blocks. To see the performance of the proposed method, experiments have been conducted to compare our method with the lossless intra-coding method in [7-8]. The rest sections of this paper are as follows: section 2 describes related works about lossless intra-coding method; section 3 describes our motivation and the proposed method which includes block classification, text block analysis and VLC table design; experimental results are provided in section 4 and the final section concludes this thesis.

Chapter 2 Related Works

2.1 Lossless Intra Coding

Traditional H.264/AVC is designed for lossy compression, the lossless coding capabilities are less well known. In the standard which included the so-called fidelity range extensions (FRExt) [9], added design improvements for more efficient lossless coding. The intra prediction utilized various predicted directions and selectable linear combinations of neighbor pixel values to form a prediction block which consisted of the difference data.



0	Vertical
1	Horizontal
2	DC
3	Diagonal Down Left
4	Diagonal Down Right
5	Vertical Right
6	Horizontal Down
7	Vertical Left
8	Horizontal Up

Figure 2-1 Nine prediction modes for the intra 4 × 4 prediction

When using lossy coding technique, the compression process consists of block transform followed by quantization to remove spatial redundancy, and entropy coding of the block of transformed coefficients. However, when using lossless coding technique, the quantization which mainly cause the data loss cannot be performed, therefore, bypassing transform and quantization in lossless coding, and the compression process only consists of entropy coding of the block of difference data.

2.2 Pixel-wise Intra Prediction [7]

The H.264/AVC intra prediction is performed on variable block sizes such as 4 × 4, 8 × 8 or 16 × 16, and the block-based intra prediction utilizes the boundary pixels of current block to be encoded and various predicted directions to form the residual data. However, the correlation between current predicted pixel and its neighbor pixels is closer than block boundary pixels. Thus, [7] proposed pixel-wise intra prediction based on pixel-by-pixel differential pulse code modulation (DPCM), performing intra prediction by neighbor pixel rather than farther block boundary pixels as shown in Figure 2-2.

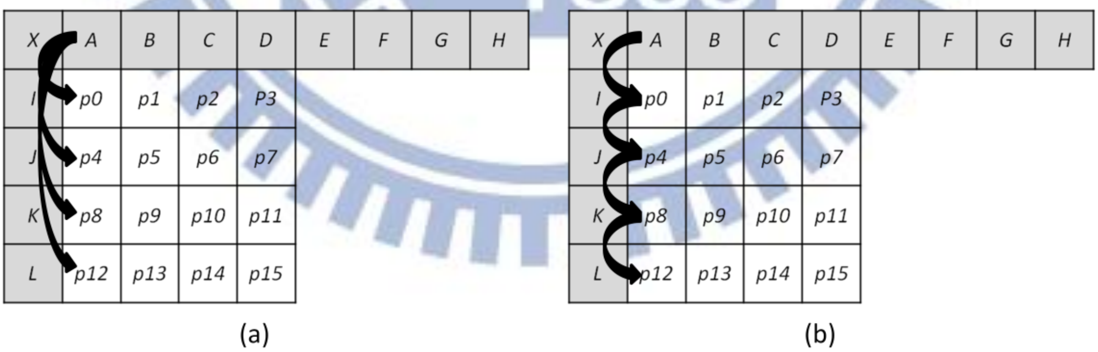


Figure 2-2 (a) Block-based (b) Pixel-wise intra 4 × 4 vertical prediction

In Figure 2-2, *p0* to *p15* are the block pixels to be encoded, as well as *X* and *A* to *L* are the block boundary pixels which used to perform the intra prediction. Performing vertical prediction on the first column pixels *p0*, *p4*, *p8* and *p12*, the block-based in Figure 2-2(a) uses boundary pixel *A* to calculate the residuals *R0*, *R4*, *R8* and *R12* as follows:

$$R0 = p0 - A \quad (1)$$
$$R4 = p4 - A \quad (2)$$
$$R8 = p8 - A \quad (3)$$
$$R12 = p12 - A \quad (4)$$

While the pixel-wise prediction in Figure 2-2(b) not uses boundary pixel *A* to predict all of the pixels in the column, but uses the upper pixel of each pixel to calculate the residual as follows:

$$R0 = p0 - A \quad (5)$$
$$R4 = p4 - p0 \quad (6)$$
$$R8 = p8 - p4 \quad (7)$$
$$R12 = p12 - p8 \quad (8)$$

Furthermore, the pixel-wise DPCM has better performance in reducing spatial redundancy that the purpose of intra prediction, and further reduces the bitrate in comparison with the lossless intra coding method previously included in the H.264/AVC standard.

2.3 Improved CAVLC (IMP_CAVLC) [8]

Context-based adaptive variable length coding (CAVLC) for the H.264/AVC standard was originally designed for lossy video coding, and as such does not yield adequate performance for lossless video coding. [8] proposed an improved CAVLC for lossless intra coding by considering the statistical differences in residual data between lossy and lossless coding.

In H.264/AVC, CAVLC was designed to take advantage of several characteristics of residual data in lossy coding:

1) After transform and quantization, sub-blocks typically contain many zeros, especially in high-frequency regions.

2) The level of the highest nonzero coefficients tends to be as small as one.

3) The level of nonzero coefficients tends to be larger toward the low-frequency regions.

Therefore, taking into consideration the above characteristics, CAVLC employs the syntax elements *coeff token, trailing ones sign flag*, *level prefix*, *level suffix*, *total zeros*, and *run before* to efficiently encode the residual data. The specific function of each syntax element is described in Table 2-1.

Syntax Elements	Description
coeff_token	Encodes the number of nonzero coefficients and trailing ones
trailing_ones_sign_flag	Sign of trailing one value
level_prefix	First part of code for nonzero coefficient
level_suffix	Second part of code for nonzero coefficient
total_zeros	Encodes the total number of zeros occurring after the first nonzero coefficient
run_before	Encodes the number of zeros preceding each nonzero coefficient

Table 2-1 CAVLC syntax elements for residual data

However, in lossless coding, residual data do not represent quantized transform coefficients, but rather the differential pixel values between the original and intra predicted pixel values. Therefore, the statistical characteristics of the residual data in lossless coding are as follows:

1) The probability of existence of a nonzero coefficient is independent of the scanning position, and the number of nonzero coefficients is generally large, compared with those in lossy coding.
2) The absolute value of a nonzero coefficient does not decrease as the scanning position increases and is independent of the scanning position.
3) The occurrence probability of a trailing one is not so high; therefore, the trailing one does not need to be treated as a special case of encoding.

Therefore, [8] modified CAVLC on the number of nonzero coefficients and level coding. For the coding of the nonzero coefficients, [8] encoded the total number of nonzero coefficients (*numcoeff*) but do not consider the number of trailing ones (*numtrailingones*), and only used one VLC table which is designed according to the statistics of number of nonzero coefficients in lossless coding as shown in Table 2-2.

numcoeff	Codeword	
	Check Bit	Bits for *numcoeff*
0	1	1111
1	1	0000
2	1	0001
3	1	0010
4	1	0011
5	1	0100
6	1	0101
7	1	0110
8	1	0111
9	1	1000
10	1	1001
11	1	1010
12	1	1011
13	0	00
14	0	01
15	0	10
16	0	11

Table 2-2 Codeword table for *numcoeff* in IMP_CAVLC

For the level coding, the absolute level value of each nonzero coefficient (*abs_level*) is adaptively encoded by a selected VLC table from among the 7 predefined VLC tables in reverse scanning order. However, *abs_level* in lossless coding is independent of the scanning position. Therefore, [8] designed an adaptive method for VLC table selection that can decrease or increase according to the weighted sum of previously encoded *abs_level*. The decision procedure for determining the VLC table is described as follows:

$$T(abs\_level_i) = \frac{1}{a_i + 1}\{a_i \cdot avg_i + abs\_level_i\} \tag{9}$$

$$a_i = \begin{cases} 0, & i = lastcoeff \\ 1, & i = lastcoeff - 1, lastcoeff - 2 \\ 2, & \text{otherwise} \end{cases} \tag{10}$$

$$avg_i = \frac{1}{(lastcoeff - i + 1)}\left\{\sum_{k=lastcoeff}^{i} abs\_level_k\right\} \tag{11}$$

Where a_i and abs_level_i are the weighting coefficient and *abs_level* value, respectively, where both values are related to the current scanning position i. In addition, $T(abs\_level_i)$ and *lastcoeff* represent the threshold value for selecting the corresponding VLC table used to encode the next *abs_level* and the scanning position number of the last nonzero coefficient, respectively.

Chapter 3 Proposed Method

There are lossless coding approaches in H.264, and some works, e.g., the method in [8], have been proposed to improve the efficiency of lossless coding. However, most of them are not specialized for text blocks because the characteristics of text blocks are not taken into considerations. Moreover, for lossless coding, since reducing coding bitrates is the most important issue, it is worth analyzing entropy coding thoroughly for text blocks, and design a new coding scheme to have a more efficient coding performance.

The proposed method is designed for encoding screen contents which include texts and natural images in the same frame. Our approach expects to encode the natural images by using traditional H.264 which is a lossy method, while encode the texts by using a lossless coding method which performs pixel-wise intra prediction and then encodes the residual directly using a proposed VLC without performing transform and quantization. The block diagram of proposed method is shown in Figure 3-1. Each 4×4 block is encoded respectively by a lossy H.264 encoder and the proposed lossless method. Then a rate-distortion optimization (RDO) technique is adopted for coding-mode (lossy or lossless) decision.

$$J = D(mode) + \lambda R(mode) \tag{12}$$

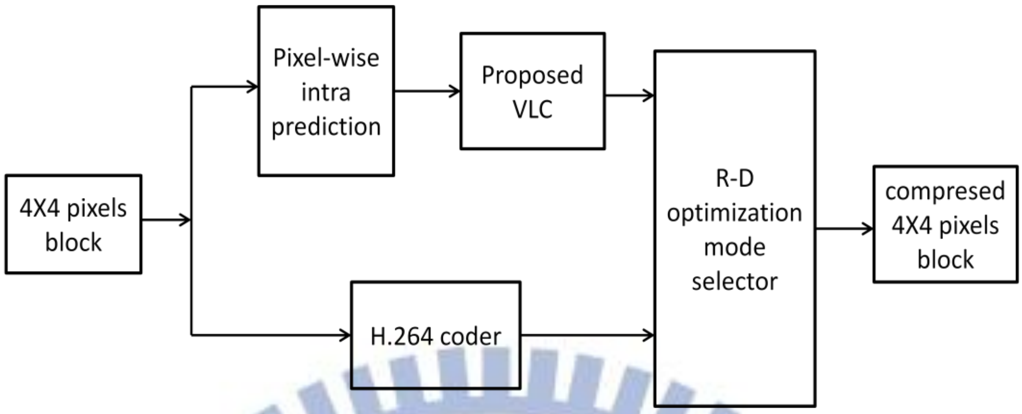


Figure 3-1 Block diagram of proposed method

3.1 Block Classification

There are various block classification methods based on numerous characteristics of texts. One of the characteristics is the distribution of DCT coefficients. DCT-based classification schemes utilize the statistics of DCT coefficients to train a classifier and get empirical thresholds [1].

Herein, we propose a new DCT-based block classification method. Considering that text blocks typically have many high-gradient pixels, which will result in many nonzero high-frequency coefficients. Namely, the more nonzero DCT coefficients a block has, the more likely it is a text block. Therefore, we utilize the number of nonzero DCT coefficients, denoted by *lsy_coef_num*, to perform block classification. For 4 × 4 blocks, the value of *lsy_coef_num* ranges from 0 to 16. To find out text blocks as much as possible, we heuristically choose those blocks with *lsy_coef_num*

$\geqq$ *10* for *QP=24* and the results are shown in Figure 3-2, where four screen-content sequences are adopted, one on each row.

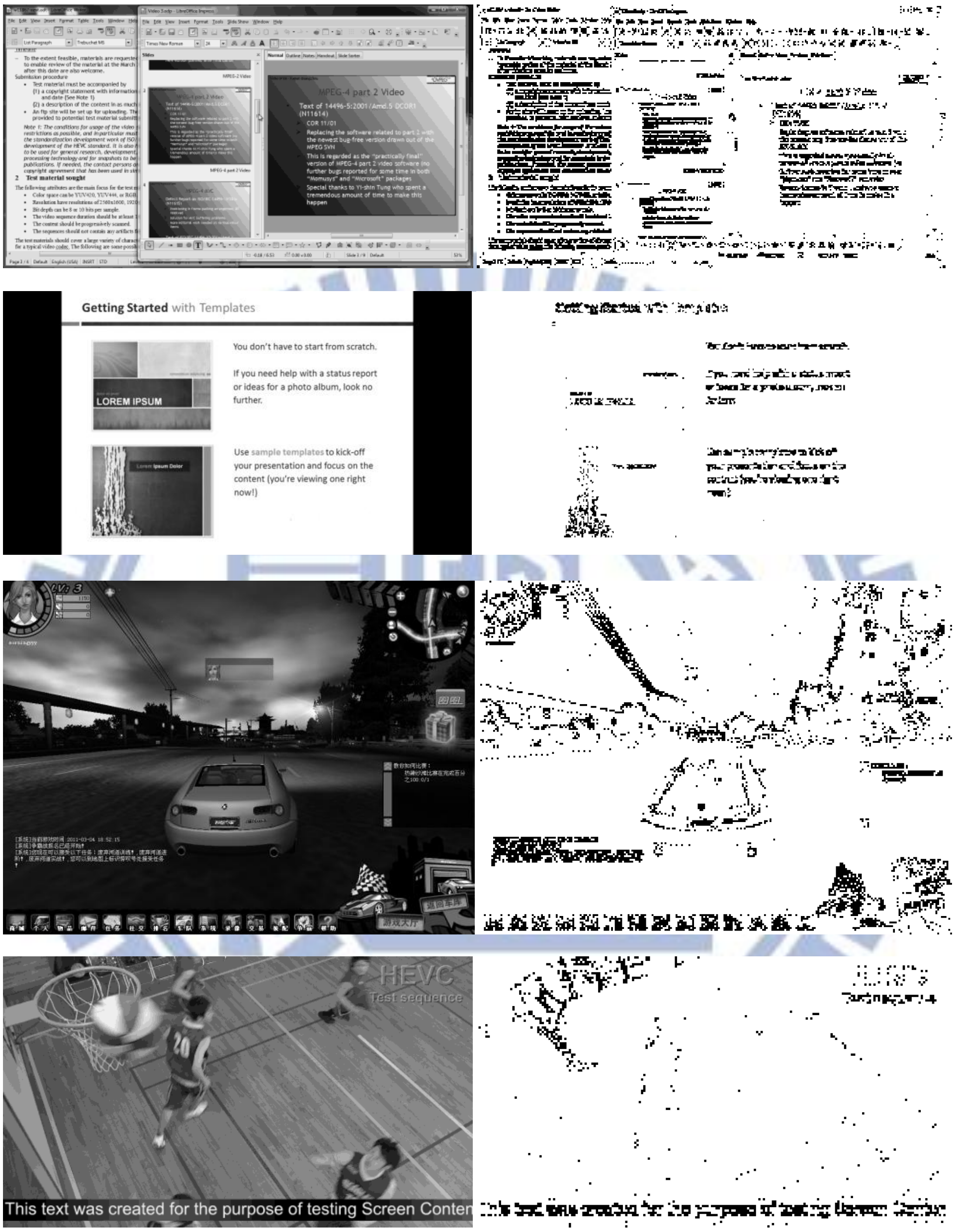


Figure 3-2 Block map of *lsy_coef_num* over 10

For each row in Figure 3-2, the left side shows the original frame and the right side shows the corresponding *block map* which includes all the blocks (marked as black color) with *lsy_coef_num* $\geqq$*10* in its original frame. It is observed that most texts in the frame can be found in this way. In the following subsections, we simply perform analysis on these blocks to design our VLC.

3.2 Text Block Analysis

For lossless coding, coding bitrate directly affects the compression performance. Figure 3-3 shows the curve of *lsy_coef_num* versus average bitrate for 4 test sequences in lossy mode.

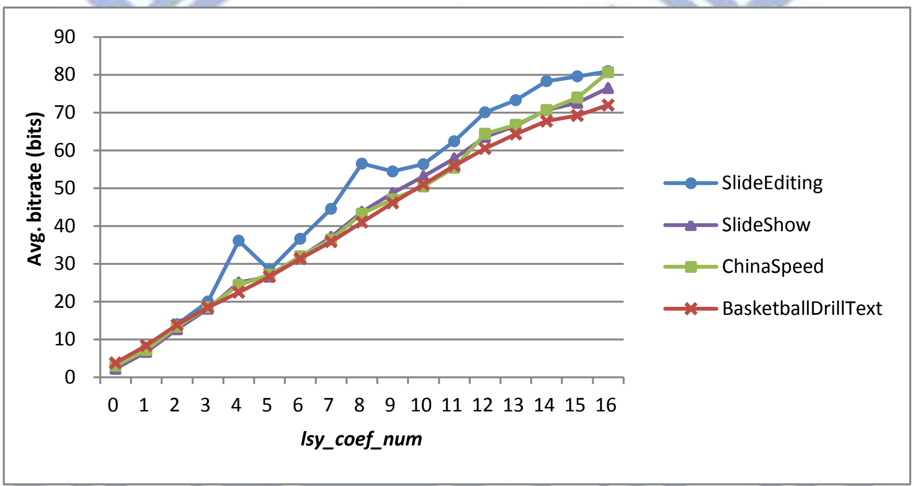
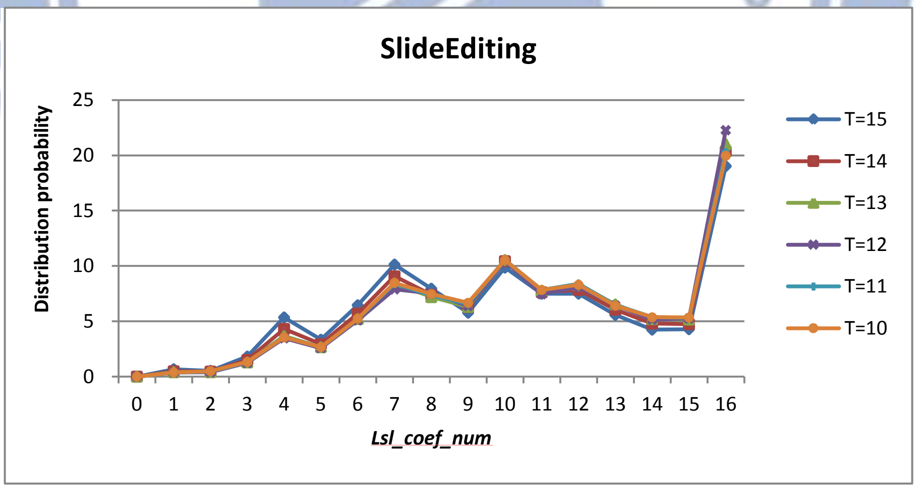


Figure 3-3 L*sy_coef_num* - Avg. bitrate curves for 4 test sequences

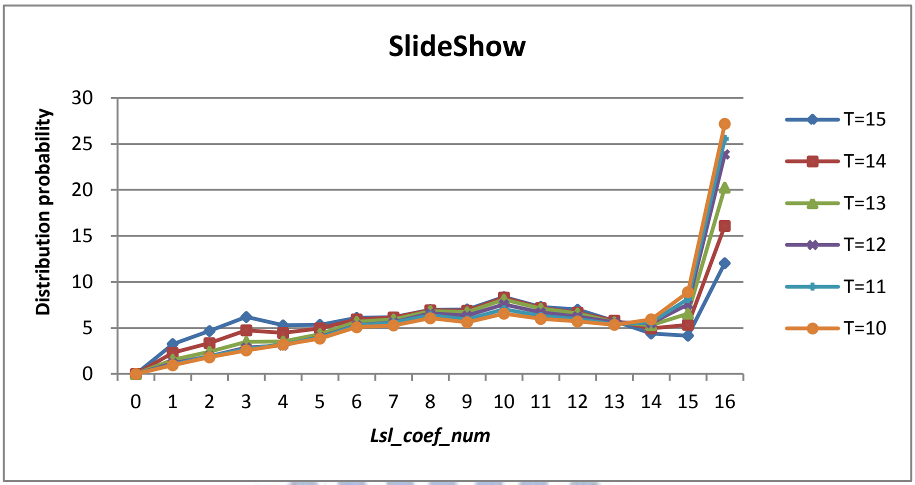
More *lsy_coef_num*, more bits for the block to be encoded. After previous block classification, the text block is distributed over the region of higher average bitrate, and our proposed method expects to design a lossless coding approach in order to reducing more bits for encoding text blocks. Due to our lossless coding is based on CAVLC and pixel-wise intra prediction, thus we analyze the statistics of coefficient and level information in lossless mode for text blocks which classified previously.

After performing pixel-wise intra prediction, the residual is directly encoded by entropy coding, herein, we call the residual as coefficient and the residual value as level. Then, we analyze the distribution of number of nonzero coefficients and level values in lossless mode, and we call them as *lsl_coef_num* and *lsl_level_cnt* respectively. Note that because of the coefficients is directly produced by pixel-wise intra prediction, the range of the absolute level value is 0 to 255.

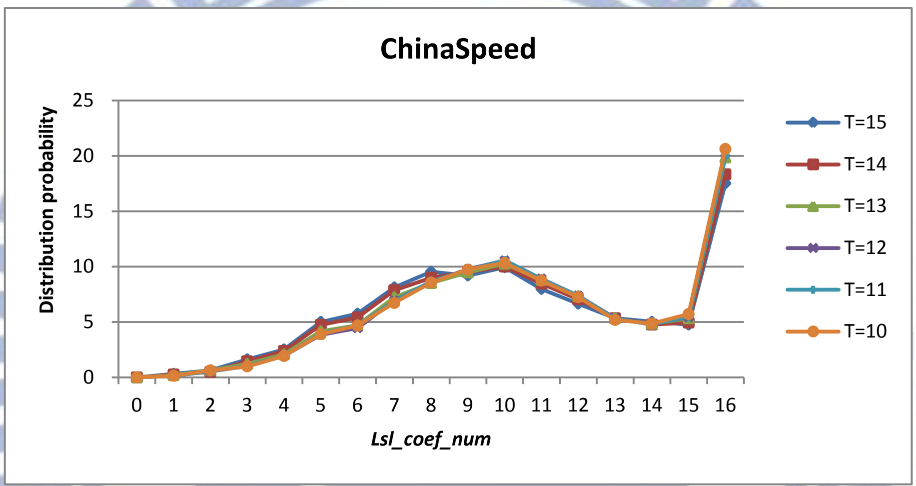
We observe the distribution probability of *lsl_coef_num* and *lsl_level_cnt* at threshold T from 10 to 15 for 4 sequences, and further find out some characteristics of coefficient block. Figure 3-4 shows the distribution probability of *lsl_coef_num* for each sequence.



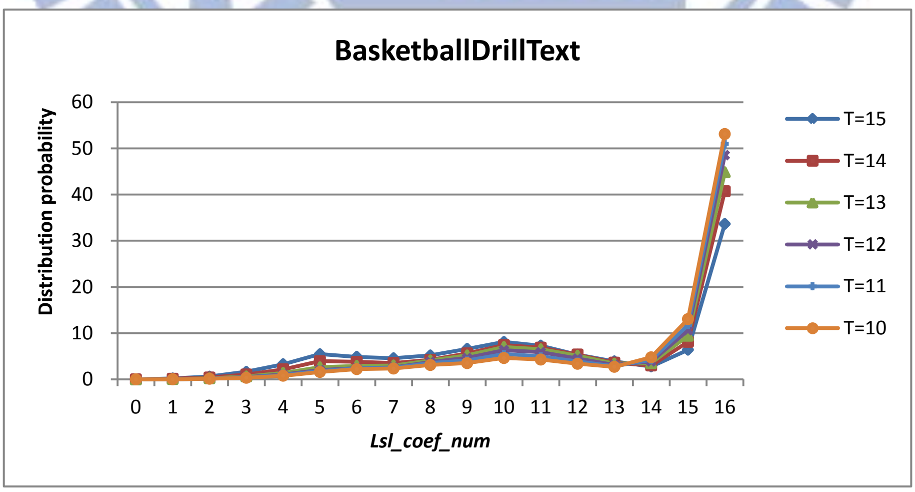
(a) Distribution probability of *lsl_coef_num* in *SlideEditing* sequence



(b) Distribution probability of *lsl_coef_num* in *SlideShow* sequence



(c) Distribution probability of *lsl_coef_num* in *ChinaSpeed* sequence

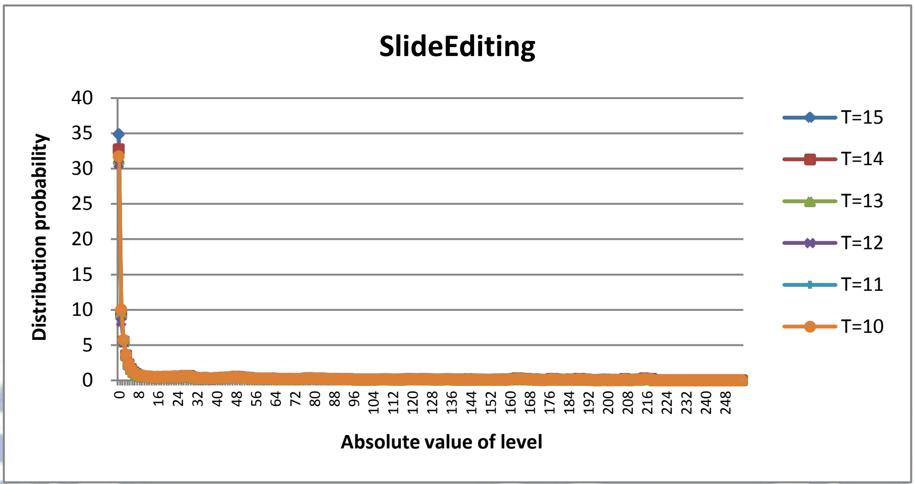


(d) Distribution probability of *lsl_coef_num* in *BasketballDrillText* sequence

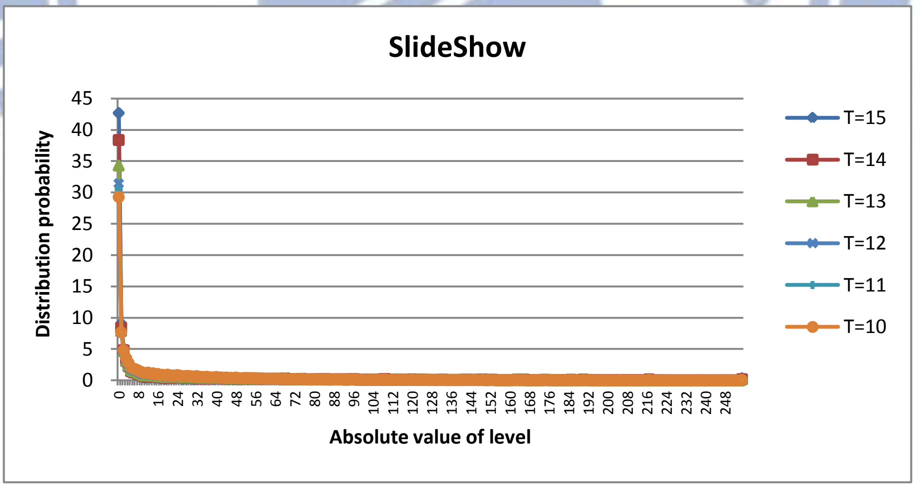
Figure 3-4 Distribution probability of *lsl_coef_num* at T from 10 to 15 for 4 sequences

As above shown, we can obtain two characteristics. First, the distribution probability of *lsl_coef_num* is similar at different threshold T for each sequence; second, the probability of *lsl_coef_num* equal to 16 is much higher than others.

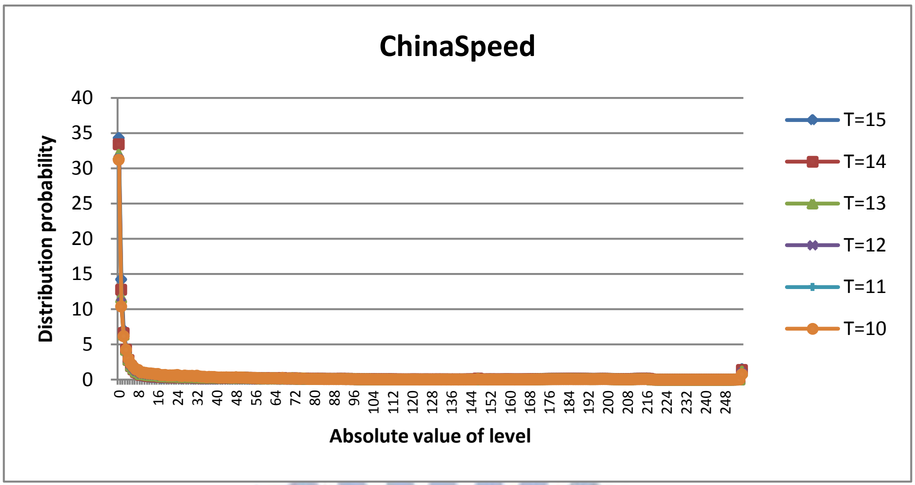
Following shows the distribution probability of *lsl_level_cnt* for each sequence in Figure 3-5.



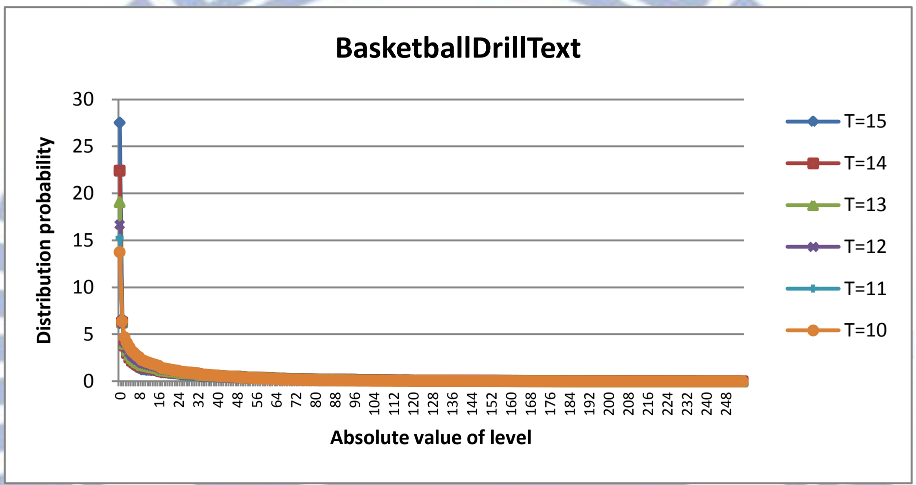
(a) Distribution probability of *lsl_level_cnt* in *SlideEditing* sequence



(b) Distribution probability of *lsl_level_cnt* in *SlideShow* sequence



(c) Distribution probability of *lsl_level_cnt* in *ChinaSpeed* sequence



(d) Distribution probability of *lsl_level_cnt* in *BasketballDrillText* sequence

Figure 3-5 Distribution probability of *lsl_level_cnt* at T from 10 to 15 for 4 sequences

According to Figure 3-5, the distribution probability of *lsl_level_cnt* is also similar at different threshold T for each sequence, and less the absolute level value, higher proportion in probability, especially 0 is most level value.

In CAVLC, zero information is encoded by run-length coding, due to the coefficient block in lossy mode contains many zero, especially in high-frequency regions, so the run-length coding is appropriate. However, the distribution of zero in lossless mode is dispersing. Figure 3-6 shows the occurrence probability of zero at each scanning position in text block. Note that the scanning order is raster scan.

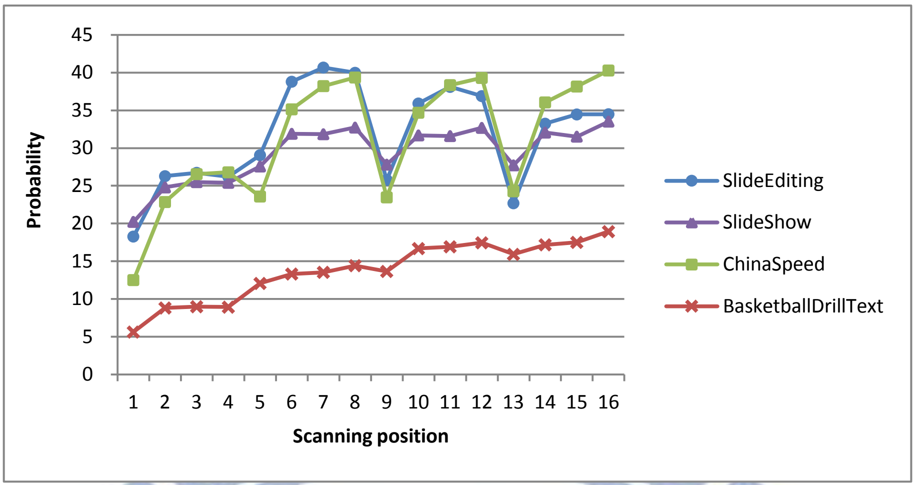


Figure 3-6 Distribution probability of zero in each scanning position

As Figure 3-6 shown, the occurrence probability of zero is independent of scanning position and obviously not centralized in high-frequency regions, thus the run-length of zero encoding in CAVLC is not appropriate for lossless coding.

3.3 VLC Table Design

According to the analysis of text block, we obtain some characteristics of text coefficient block to design the new entropy coding method.

Due to the occurrence probability of zero is independent of scanning position, we do not encode the syntax *total_zeros* and *run_before* which encoded in CAVLC, we regard zero as level coding. That is, the level normally refers to nonzero coefficients, but in our proposed method, our level refers to all coefficients including of zero and nonzero coefficients. Furthermore, because of no *total_zeros* and *run_before*, we also do not encode the *coef_token* syntax, as a result the proposed entropy coding only contain the level coding.

CAVLC has 7 selected VLC tables to encode the level, but these tables are designed for transform coefficients that the range of their values is extensive. However, the range of our absolute level values is only 0 to 255 in lossless mode, therefore the proposed method only utilize one VLC system to encode the level.

Owing to the distribution of *lsl_level_cnt* is similar at different threshold T, we adopt the average distribution probability of *lsl_level_cnt* for 4 sequences at T equals to 10 to design the VLC system. The principle of VLC design is that the level with higher probability then uses shorter bits to encode, and the bits for encode level can be divided into 3 parts: check bits (3 bits), level bits (0 to 7 bits) and sign bit (1 bit). Table 3-1 shows the proposed VLC table including of bitstream pattern, bit length and represented level values. Note the × in the table stands for the level bits and sign bit.

Bitstream pattern	Bit length	level
000x	4	+1,-1
001xx	5	(+2 to +3), (-2 to -3)
010xxx	6	(+4 to +7), (-4 to -7)
011xxxx	7	(+8 to +15), (-8 to -15)
100xxxxx	8	(+16 to +31), (-16 to -31)
101xxxxxx	9	(+32 to +63), (-32 to -63)
110xxxxxxx	10	(+64 to +127), (-64 to -127)
111xxxxxxxx	11	(+128 to +255), (-128 to -255)

Table 3-1 Proposed VLC table-1

According to the first 3 check bits, decoder following read different bit length. Longer bit length can represent more level values, but the distribution probability of these level values with longer bit length is lower than that with shorter bit length. But in Table 3-1, the zero still does not add in. Due to the zero is regarded as level coding in our proposed method and the distribution probability of zero is much higher than other levels, we modify the VLC table as shown in Table 3-2. We assign only 1 bit to encode zero, and others levels add 1 bit to distinguish with zero, so the bit length for each level is one more than in Table 3-1.

Bitstream pattern	Bit length	level
0	1	0
1000x	5	1,-1
1001xx	6	(+2 to +3), (-2 to -3)
1010xxx	7	(+4 to +7), (-4 to -7)
1011xxxx	8	(+8 to +15), (-8 to -15)
1100xxxxx	9	(+16 to +31), (-16 to -31)
1101xxxxxx	10	(+32 to +63), (-32 to -63)
1110xxxxxxx	11	(+64 to +127), (-64 to -127)
1111xxxxxxxx	12	(+128 to +255), (-128 to -255)

Table 3-2 Proposed VLC table-2

However, while *lsl_coef_num* is 16, there is no zero in the block, thus we can use Table 3-1 to encode the blocks as well as saving more bits. Furthermore, the highest distribution probability of *lsl_coef_num* is 16 and much higher than others, so we adopt Table 3-1 to encode the blocks with 16 *lsl_coef_num*, and other blocks that *lsl_coef_num* is less than 16 are adopt Table 3-2 to encode. Consequently, we have to transmit a 1-bit flag for each lossless coding block to decoder for distinguishing which table is used. According to extensive experiments, while the blocks with 16 *lsl_coef_num* encode by Table 3-1 rather than Table 3-2, the reducing bits are more than the extra bits for tag all lossless coding blocks.

Chapter 4 Experiment Results

In this paper, we proposed a new VLC in H.264 lossless intra-coding for encoding text blocks. In order to verify efficiency of the proposed method, we perform experiments on 4 test sequences of YUV420 and 8 bits per pixel format with high-definition resolutions as follows:

- *SlideEditing*, 1280 × 720, 300 frames.
- *SlideShow*, 1280 × 720, 500 frames.
- *ChinaSpeed*, 1024 × 768, 500 frames.
- *BasketballDrillText*, 832 × 480, 500 frames

We implement our proposed method in the H.264/AVC reference software version JM 18.2, and encode all frames as intra frame for verify the enhanced performance by our proposed lossless intra-coding with various OP which adopted 12 to 24. We compare our proposed method with two compression approaches. The first one is encoding all sequence by lossy H.264/AVC, and the second one is partial-lossless compression by improved CAVLC (IMP_CAVLC) [8] for lossless coding and H.264/AVC for lossy coding.

Experiment results firstly show the lossless block map to indicate the lossless block which selected by RDO selector, secondly show the *lsy_coef_num* versus average bitrate curve to verify whether achieving bits saving and finally show the R-D performance and visual quality.

4.1 Lossless Block Map

The purpose of showing lossless block map is verifying that our proposed entropy coding can precisely perform on the text blocks which classified by our block classification, and further be selected by RDO mode selector. We use the same frames of each sequence as in Figure 3-1 at QP is 24, and mark the lossless block as black color in the block map as shown in Figure 4-1.

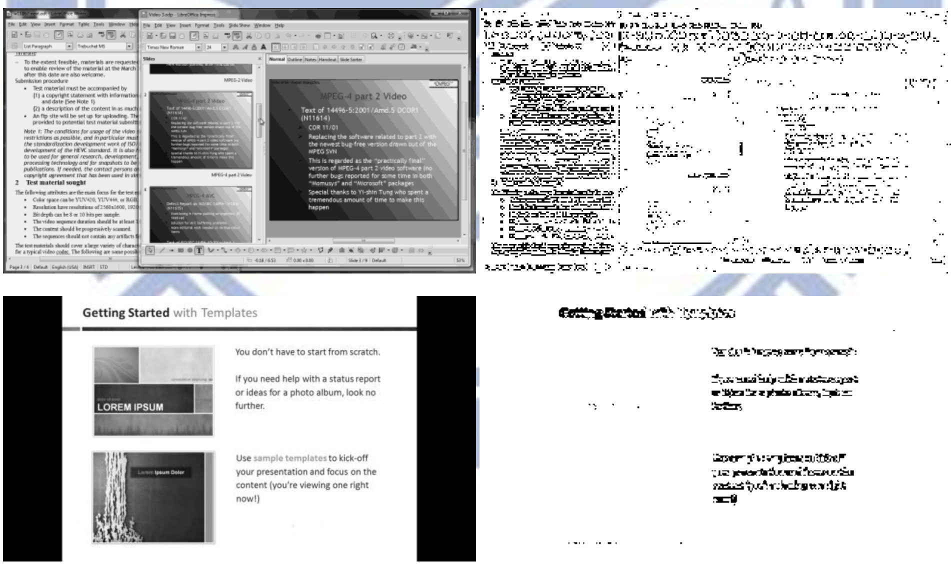
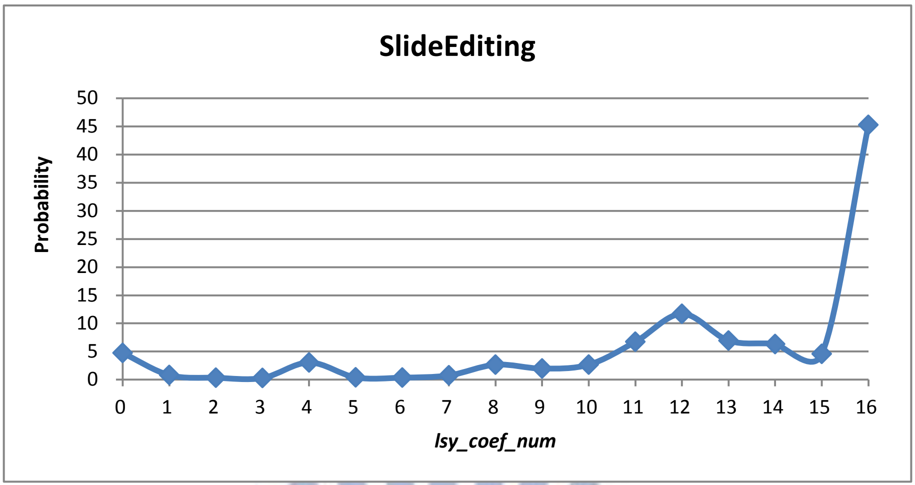
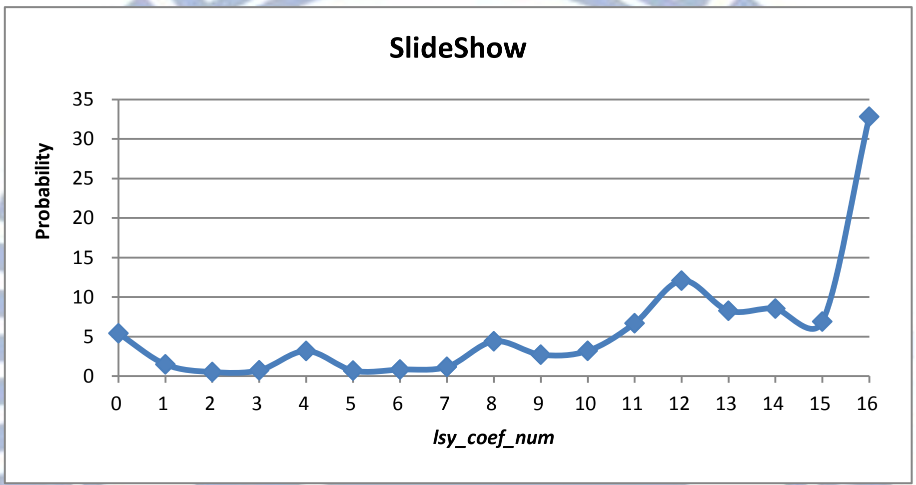


Figure 4-1 Lossless block map by RDO selector

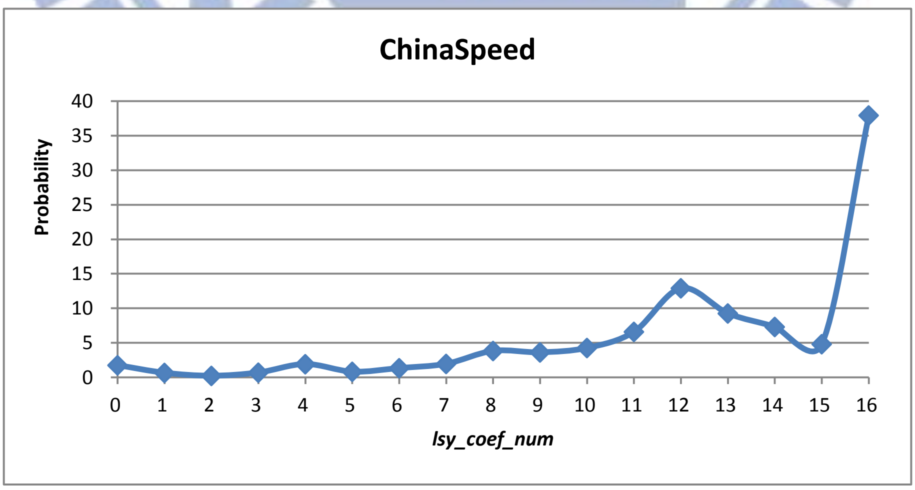
The black blocks in the block map are selected by RDO selector to be encoded by lossless coding. Our proposed method is designed according to the characteristics of text block, as Figure 4-1 shown, most text contents in the frames can be exactly selected. Due to our block classification is based on *lsy_coef_num*, we take extensive experiment to verify the distribution of *lsy_coef_num* in lossless blocks selected by RDO selector. Figure 4-2 shows the result for each sequence, the probability of *lsy_coef_num* over 11 average possesses 81 percent, especially at *lsy_coef_num* is 16 average possesses 38 percent.



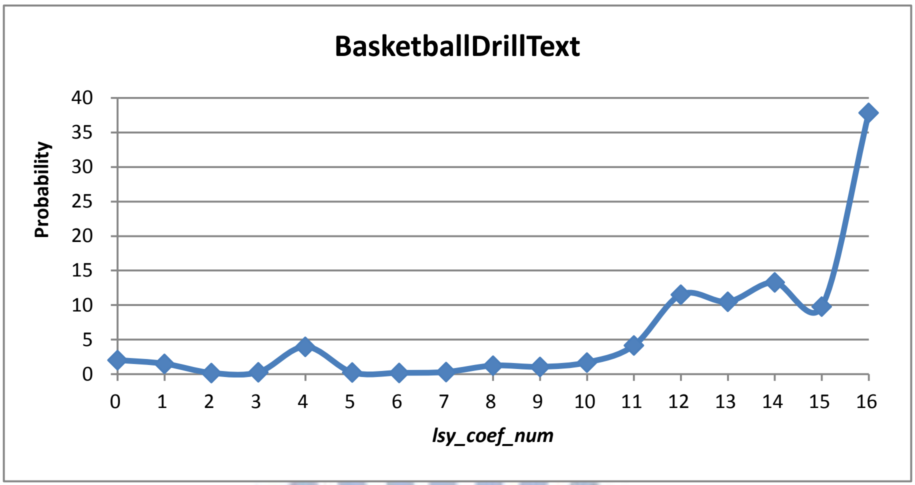
(a) Probability of each *lsy_coef_num* in *SlideEditing* sequence lossless blocks



(b) Probability of each *lsy_coef_num* in *SlideShow* sequence lossless blocks



(c) Probability of each *lsy_coef_num* in *ChinaSpeed* sequence lossless blocks

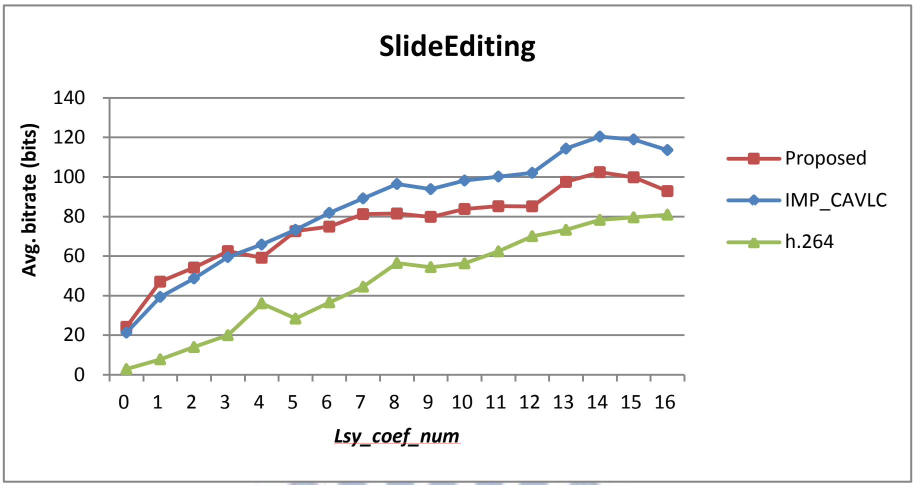


(d) Probability of each *lsy_coef_num* in *BasketballDrillText* sequence lossless blocks

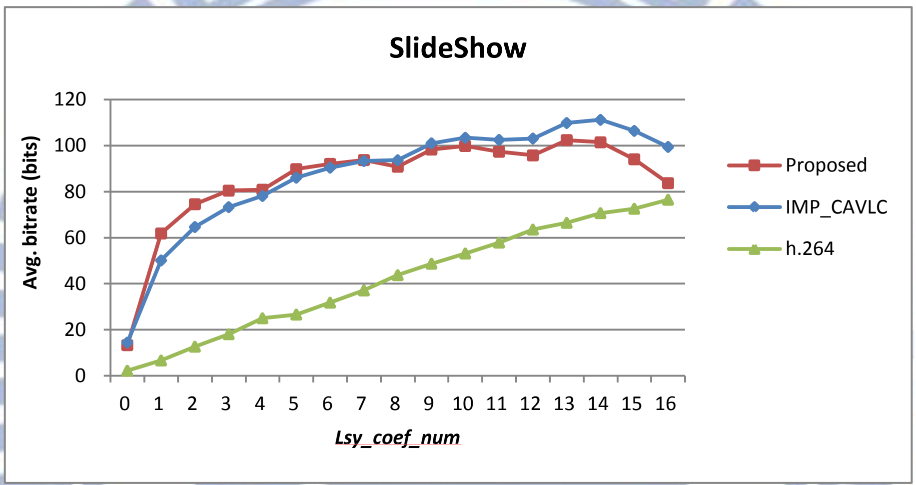
Figure 4-2 Probability of each *lsy_coef_num* in each sequence lossless blocks

4.2 Bitrate Performance

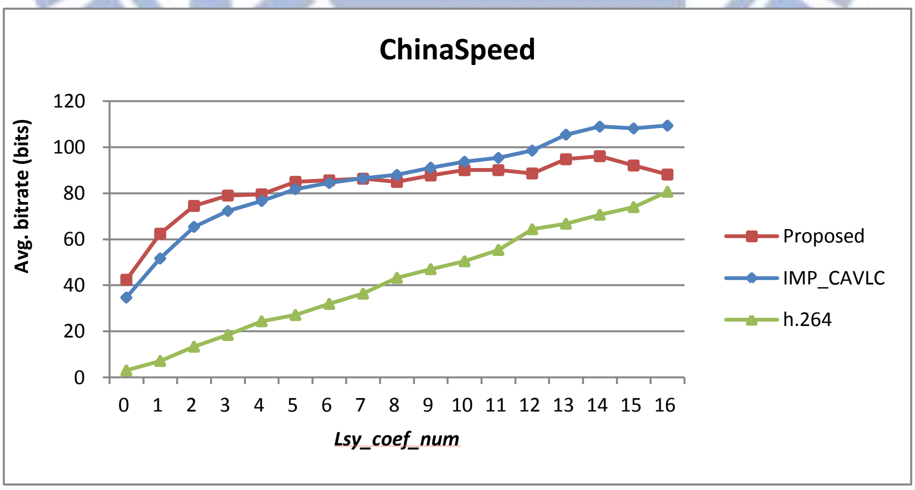
The purpose of our proposed method is designing a new lossless entropy coding to achieve more bits saving than IMP_CAVLC, and our target blocks are that with higher *lsy_coef_num* ones. Therefore, we take experiments to show the average bitrate of each *lsy_coef_num* to verify the performance of coding bitrate. Note the QP is 24 in here. The results of each sequence are shown in Figure 4-3.



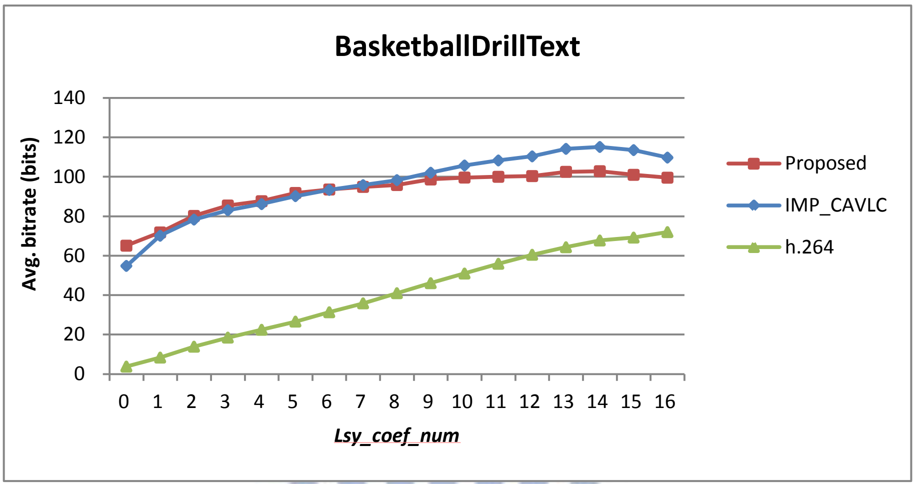
(a) ***Lsy_coef_num* versus Avg. bitrate curves in *SlideEditing* sequence**



(b) ***Lsy_coef_num* versus Avg. bitrate curves in *SlideShow* sequence**



(c) ***Lsy_coef_num* versus Avg. bitrate curves in *ChinaSpeed* sequence**



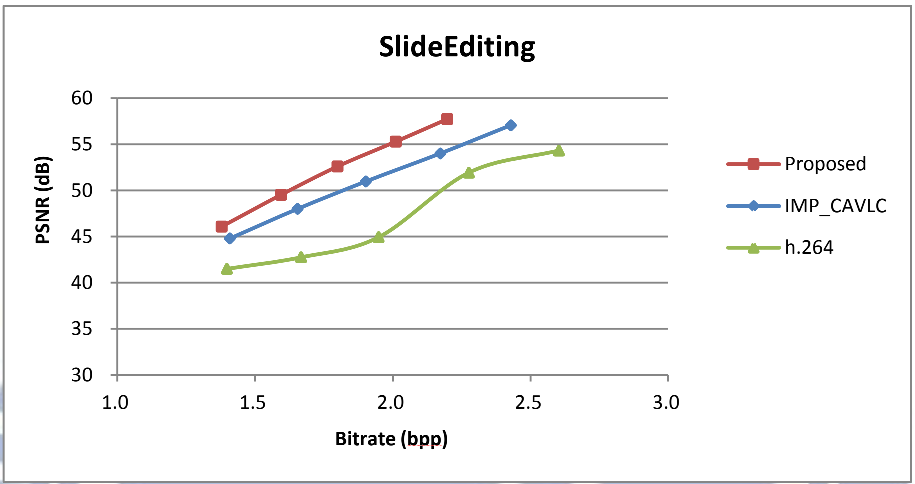
(d) *Lsy_coef_num* versus Avg. bitrate curves in *BasketballDrillText* sequence

Figure 4-3 *Lsy_coef_num* versus Avg. bitrate curves of each method for 4 sequences

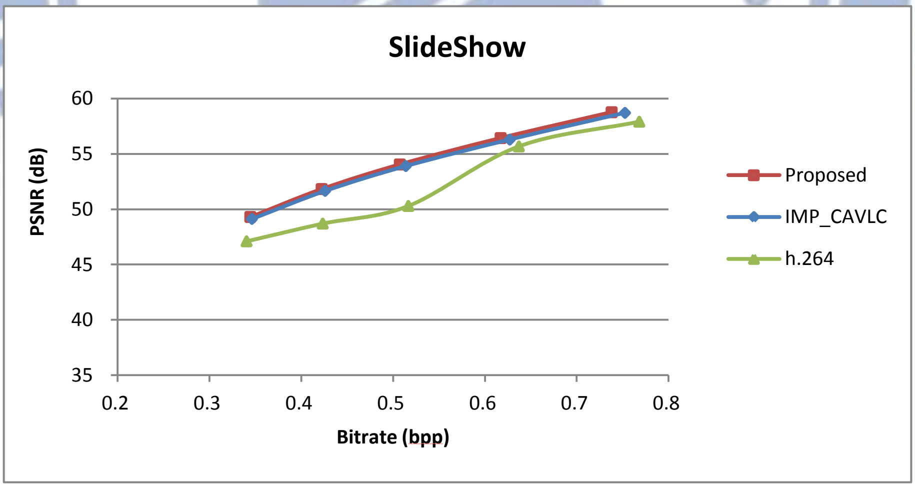
At higher *Lsy_coef_num* regions, we can reduce appropriately 10 to 20 average bits compared with IMP_CAVLC, especially at *Lsy_coef_num* is 16, the proposed average bitrate is close to H.264, that means our proposed method can achieve lossless coding with little average bits more than H.264 lossy coding at higher *Lsy_coef_num* regions, even at the smaller QP value, the proposed method can achieve fewer coding bitrate than lossy coding.

4.3 Rate-Distortion Performance

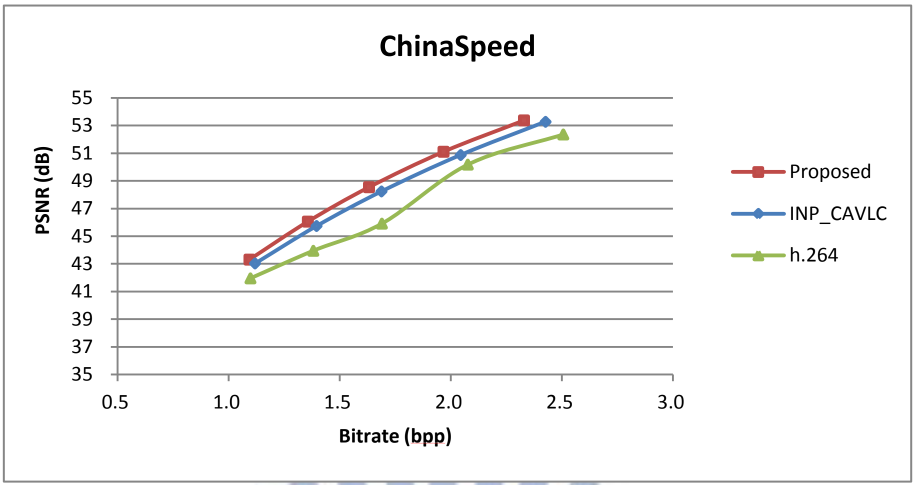
In this section, the R-D performance of each method is examined. Owing to our proposed method is lossless entropy coding, the distortion of the blocks encoded by our proposed method is 0, and this is also the reason why we only need to focus on reducing coding bitrate. According to section 4-2 bitrate performance experiment results, our proposed method exactly reduce more coding bitrate than IMP_CAVLC, and further result in more blocks encoded by lossless coding and enhance the PSNR performance. Therefore, following experiment results show the rate-distortion performance of each method for each sequence.



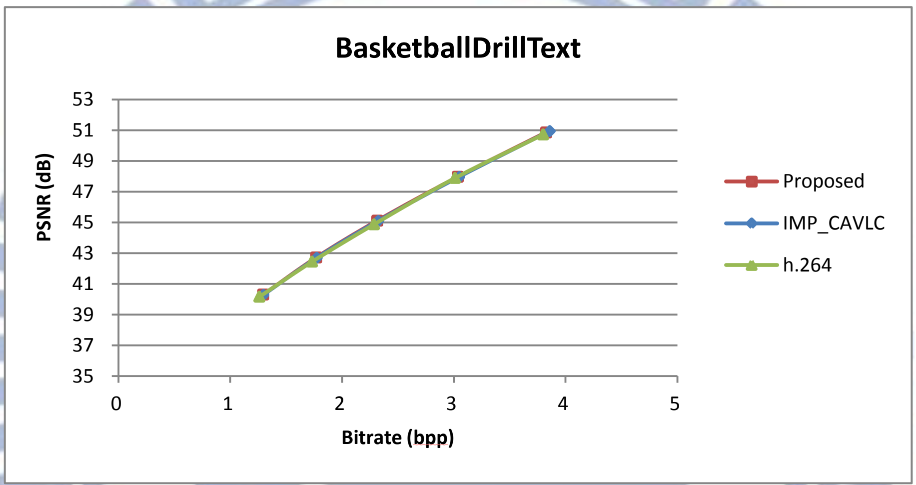
(a) R-D curves in *SlideEditing* sequence



(b) R-D curves in *SlideShow* sequence



(c) R-D curves in *ChinaSpeed* sequence

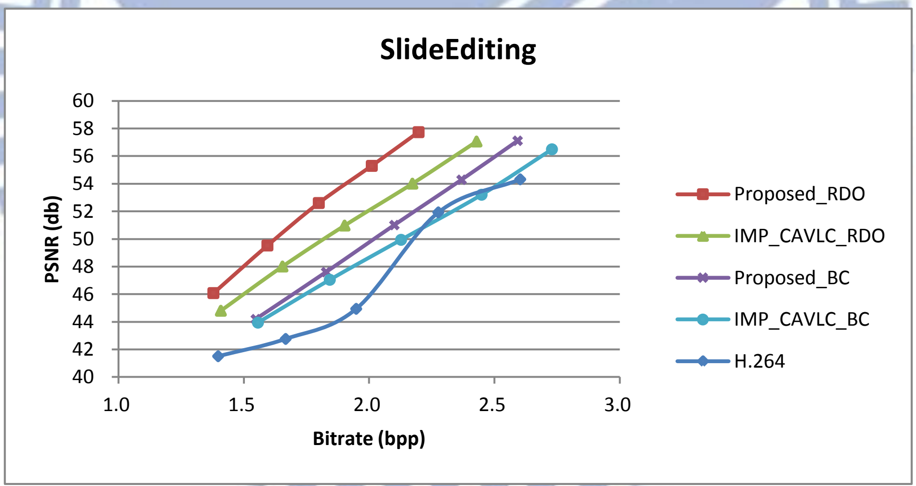


(d) R-D curves in *BasketballDrillText* sequence

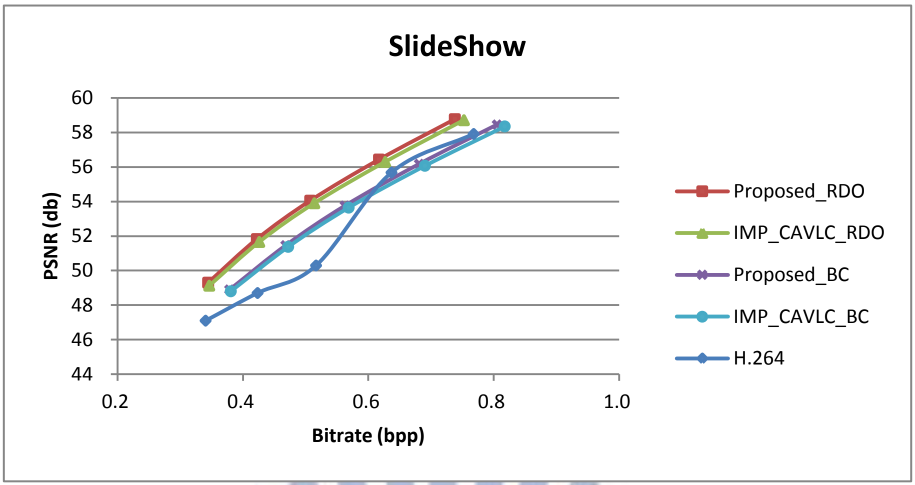
Figure 4-4 R-D curves of each method for 4 sequences

The *SlideEditing* sequence is a typical screen content sequence, it contains numerous text content in the frame, and resulting in at the same bitrate our proposed method can achieve more than 6 to 8 dB performance improvement than H.264 as well as 2 to 3 dB than IMP_CAVLC. For the *SlideShow* sequence, due to it contains less text content and more smooth background natural content, the PSNR performance is similar with IMP_CAVLC but 2 to 3 dB more than H.264. For the *ChinaSpeed* sequence, proposed method can achieve 1 dB more than IMP_CAVLC and 2 dB more than H.264. The last *BasketballDrillText* sequence is close to totally natural content sequence, the text content only appears in the bottom of frame with small proportion, hence our proposed method and IMP_CAVLC have almost the same performance with H.264, owing to the most blocks are selected to encode by lossy coding.

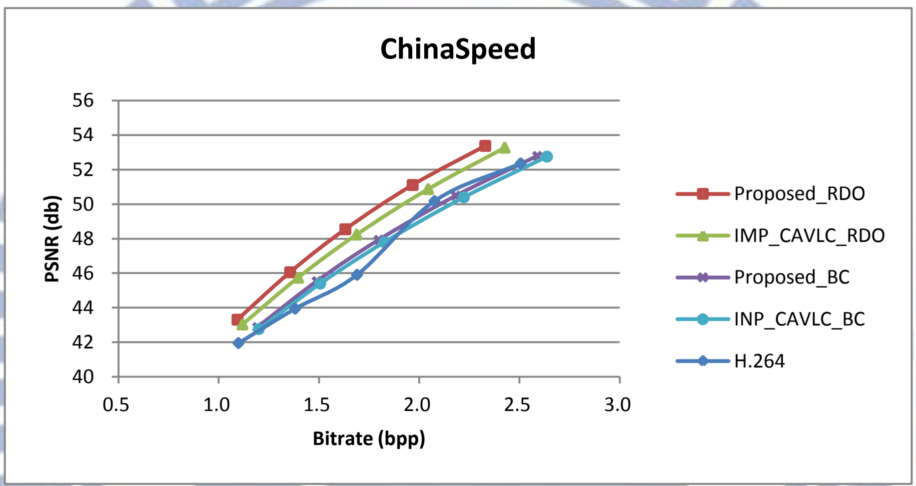
We further experiment the R-D performance by our block classification not RDO mode selector, if the block with *lsy_coef_num* over 10 then using lossless coding. By this experiment we can clearly verify the coding performance between proposed method and IMP_CAVLC for the same lossless blocks, and further compare the performance between compressed by RDO selector and our block classification (BC).



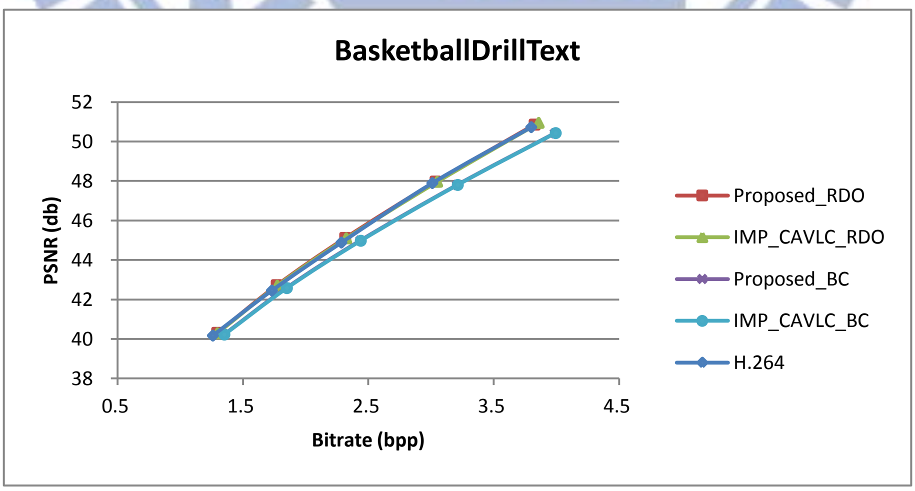
(a) R-D curves in *SlideEditing* sequence



(b) R-D curves in *SlideShow* sequence



(c) R-D curves in *ChinaSpeed* sequence



(d) R-D curves in *BasketballDrillText* sequence

Figure 4-5 R-D curves of each method compressed by RDO and BC for 4 sequences

As Figure 4-5 shown, no matter compressed by RDO or our block classification, the coding efficiency of proposed method is little better than IMP_CAVLC. And due to the RDO selector can choose more lossless block than block classification by R-D cost, the enhancement of coding performance compressed by RDO is better than by block classification.

According to the experiment results, proposed method has great enhancement on intra frame (I frame) coding, then we further take extensive experiment by join the temporal correlation inter frame (P frame) coding. For the *SlideEditing* sequence 30 frames, the GOP of our experiment is IPPPP, and the R-D curves of each method as shown in Figure 4-6.

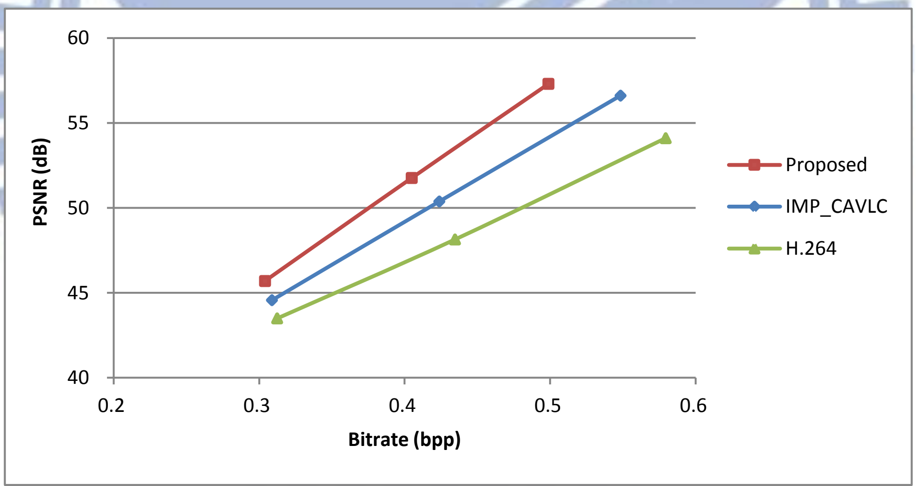


Figure 4-6 R-D curves of each method with inter coding for *SlideEditing* sequence 30 frames

Proposed method can achieve average more than 2 to 3 dB than IMP_CAVLC and 4 to 5 dB than H.264, that means our proposed method not only improve the performance of intra frame coding, but influence the inter frame coding. Figure 4-7 shows the PSNR frame-by-frame, further verifying the correlation of intra frames and inter frames

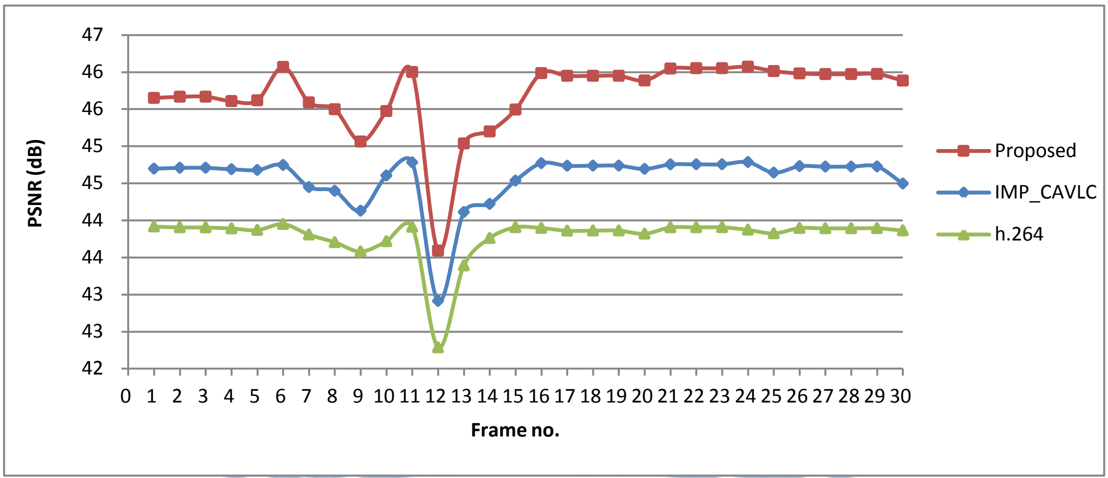
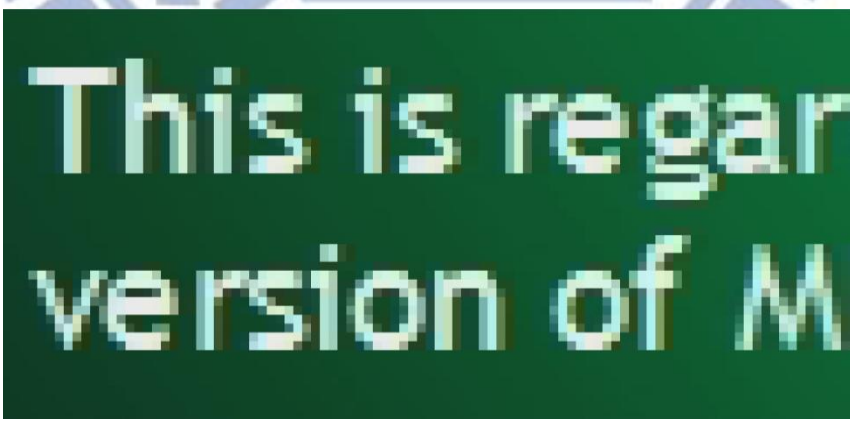
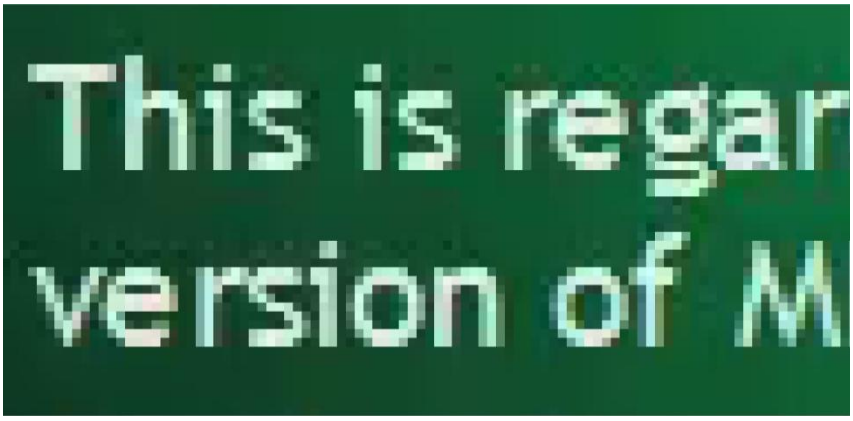


Figure 4-7 Frame-by-frame PSNR comparison of each method

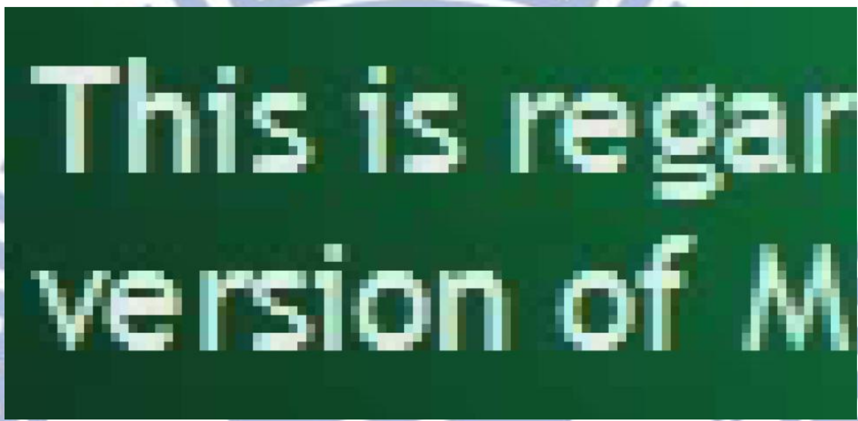
Finally we show the visual quality of each method in Figure 4-8 and Figure 4-9, due to our proposed method is lossless coding, thus we can modify the text edge noise in H.264 lossy coding and own the original quality. The IMP_CAVLC still has several lossy blocks around the text edge, although it is also lossless coding, but the coding bitrate is too large to be selected by RDO mode selector.



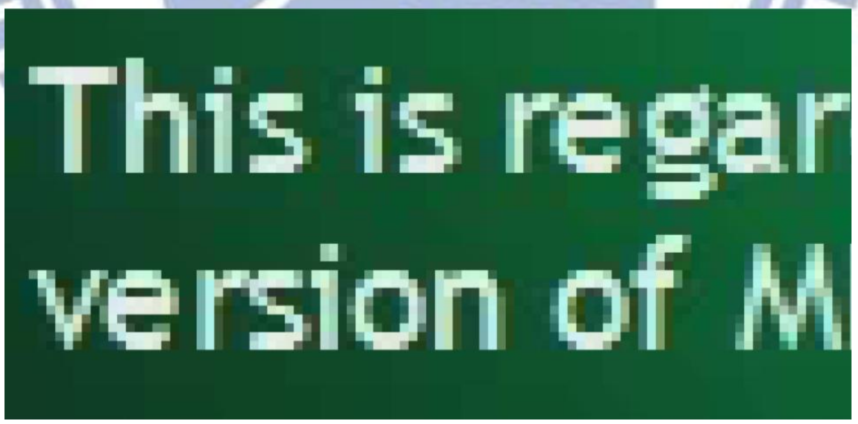
(a) Original



(b) H.264



(c) IMP_CAVLC



(d) Proposed

Figure 4-8 Visual quality comparison for part of *SlideEditing* sequence frame

(a) H.264

(b) IMP_CAVLC

(c) Proposed

Figure 4-9 Visual quality comparison for part of *ChinaSpeed* sequence frame

Chapter 5 Conclusion

In this paper, we proposed a new VLC in lossless intra-coding for screen contents. We utilize the number of nonzero coefficients in lossy mode to perform the block classification in order to finding out the text block, and then we analyze the characteristics of text coefficient block in distribution of number of nonzero coefficients and level values, moreover, the distribution probability of zero in the text block is also in our statistic. According to the text block analysis, we design a new VLC table to perform lossless entropy coding, and finally utilize the RDO mode selector in H.264 to determine the compression method.

Experiment results show that the proposed method provide an appropriate block classification method, it can exactly find out numerous text blocks in the frame. Furthermore, proposed method achieves more bits saving for text block coding and better PSNR improvement performance than other methods, as well as enhancement on the visual quality.

Reference

[1] I. Keslassy, M. Kalman, D. Wang, and B. Girod, "Classification of compound images based on transform coefficient likelihood," in *Proc. Int. Conf. Image Processing*, Oct 2001, vol. 1, pp. 750–753.

[2] W. Ding, D. Liu, Y. He, and F. Wu, "Block-based Fast Compression for Compound Images," in *IEEE Int. Conf. Multimedia and Expo*, July 2006, pp. 809–812.

[3] S. Ebenezer Juliet, D. Jemi Florinabel, "Efficient block prediction-based coding of computer screen images with precise block classification," in *IET Image Processing*, June 2011, vol. 5, no. 4, pp. 306–314.

[4] C. Lan, F. Wu, G. Shi, "Compress Compound Images in H.264/MPEG-4 AVC by Fully Exploiting Spatial Correlation," in *ISCAS IEEE Int. Symp. Circuits and Systems*, May 2009, pp. 2818–2821.

[5] S. Wang, T. Lin, "A Unified LZ and Hybrid Coding for Compound Image Partial-Lossless Compression," in *CISP 2nd Int. Congress Image and Signal Processing*, Oct. 2009, pp. 1–5.

[6] S. Wang, T. Lin, "United Coding for Compound Image Compression," in *CISP 3rd Int. Congress Image and Signal Processing*, Oct 2010, vol. 2, pp. 566–570.

[7] Y. Lee, K. Han, S. G.J., "Improved Lossless Intra Coding for H.264/MPEG-4 AVC," in *IEEE Transactions Image Processing*, Sep 2006, vol. 15, no. 9, pp. 2610–2615.

[8] J. Heo, S. Kim, and Y. Ho, "Improved CAVLC for H.264/AVC Lossless Intra-Coding," in *IEEE Transactions Circuits and Systems for Video Technology*, Feb. 2010, vol. 20, no. 2, pp. 213–222.

[9] G. J. Sullivan, T. McMahon, T. Wiegand, and A. Luthra, Eds., Draft Text of H.264/AVC Fidelity Range Extensions Amendment to ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC, ISO/IEC JTC1/SC29/WG11 and ITU-T Q6/SG16 Joint Video Team document JVT-L047, Jul. 2004.

[10] X. Li and S. Lei, "Block-based segmentation and adaptive coding for visually lossless compression of scanned documents," in *Proc. Int. Conf. Image Processing*, Oct. 1999, vol. I, pp. 219–223.

[11] W. Ding, Y. Lu, and F. Wu, "Enable efficient compound image compression in H.264/AVC intra coding," in *Proc. Int. Conf. Image Processing*, Oct. 2007, vol. 2, pp. 337–340.

[12] Reference software of H.264/AVC, version jm18.2: downloadable at http://iphome.hhi.de/suehring/tml/download/old_jm/.