AN EFFICIENT SURVEILLANCE CODING METHOD BASED ON A TIMELY AND BIT-SAVING BACKGROUND UPDATING MODEL

Wei Chen, Xianguo Zhang, Yonghong Tian, Tiejun Huang
National Engineer Laboratory for Video Technology, Peking University, Beijing 100871, China
{wchen, xgzhang, yhtian, tjhuang}@jdl.ac.cn

ABSTRACT

[1]Background modeling is an important pre-processing step for object detection in surveillance video analysis systems. Recently, it has been proved to be useful for high-efficiency surveillance video coding. In existing works, the modeling background frame often needs to be high-quality encoded so as to achieve a large bit-rate saving. However, the high-quality background frame requires lots of bits in the code stream, so it is infeasible to update the background frame too frequently. Therefore, a better bit-allocation method is desirable to facilitate in-time background updating and bit-saving background coding. In this paper, we firstly build up a background updating model from a detailed analysis of results on surveillance video. Following this, we propose a bit-saving and quality-maintaining background frame coding method. In our method, the background frame can be updated more timely, consequently leading to the better coding efficiency. Experimental results show that our method can achieve more than 15% bit-rate decrease compared with three state-of-art methods.

Index Terms— surveillance video coding, bit-saving background updating, bit-allocation, coding efficiency

1. INTRODUCTION

Nowadays, a large number of cameras are widely deployed in different surveillance systems, consequently generating huge video data every day. This triggers the increasing demands for high-efficiency coding methods. Often, surveillance cameras are mostly deployed on a fixed position to capture a certain range of scene for a long time. Therefore, it is reasonable to employ the background prediction in hybrid block-based coding framework to improve the coding efficiency.

Following this, some methods are proposed in recent years. In H.120 [1], the reconstructed frames are used as a training set to generate a background frame for prediction. In [2], M. Paul proposed to generate such a background frame by the Gaussian mixture based dynamic background modeling [3-4]. Despite the coding efficiency has been improved in these two methods, however, modeling the background frame with the reconstructed frames cannot guarantee its quality, especially in low-bit-rate video coding.

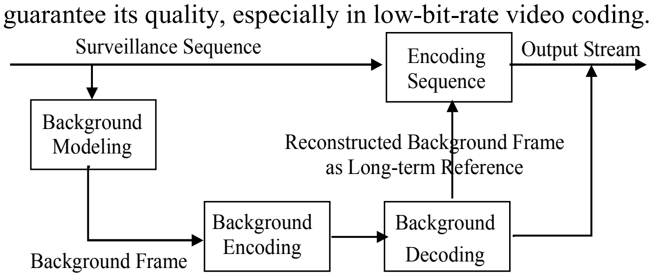

Fig. 1. Framework of BgModeling based encoders

In order to provide a better reference for the following frames in the hybrid block-based coding, Zhang et al. [5-7] proposed to utilize original input frame in the background modeling procedure so as to generate a more accurate background frame (as shown in Fig. 1). Although a remarkable bit-rate decrease has been achieved, there still exist several imperfect aspects. First and most importantly, to guarantee a decoding match, the background frame should be encoded into the code stream. This brings a problem that a large number of bits are paid to encode the background frame, consequently leading to an increase of the total bit-rate. Usually, the bits for a background frame may be several dozen times as large as a general frame. As a result, if the background frame is updated too frequently, the total bit-rate will be very large. On the contrary, if the background frame is updated after a long time, the prediction efficiency will certainly decrease. Thus it is highly necessary to develop a better trade-off between background coding and background updating.

Recently, some background updating strategies have been proposed. For example, a most common frame of a scene (McFIS) is generated as a background frame in [8] and then scene change detection (SCD) is used to judge whether the background frame should be updated. Although this method improves the encoding efficiency to some degree, the SCD-based background updating and the background frame bit-allocation strategy cannot meet the high-efficiency coding demand for surveillance video where the background often changes gradually.

In this paper, we firstly present an analysis of the three factors that influences the background updating time. The three factors are the gap between QPs of a background and general frames, the encoding gain with background frame as reference, and the bit-cost of encoding the background frame. Three conclusions can be drawn: (i) The best

[1] Prof. Tiejun Huang is the corresponding author.

encoding performance will be obtained with QP of the background frame equal to 0. (ii) The ratio of average encoding bits (shortly as bit ratio) for the frames before and after a background frame can represent the encoding gain to some degree due to using the background frame as reference. (iii) To save the bit cost meanwhile maintaining the quality of the background frame, some special encoding methods should be used for the corresponding Macroblocks (MB).

Taking these three factors into account, we propose an efficient surveillance coding method based on a timely and bit-saving background updating model. Firstly, we analyze and build up an adaptive updating model for different scenes and QPs. Generally, it updates the background frame when the bit ratio between the sum bits and the bit-cost of the encoded background frame is greater than the threshold calculated by our updating model. While coding the updated background frame, skip mode is used for the unchanged MBs. Secondly, following this timely and bit-saving background updating model, an efficient surveillance coding method is proposed. Besides, a background selection is used to choose a better reference between the initial background frame and the recently-updated one. Experimental results show that our method can averagely save 22.57% and 17.72% bit-rate on CIF and SD sequences over McFIS.

The rest of this paper is organized as follows. Sec.2 analyses the three referred factors. Sec. 3 presents the proposed background updating model. Sec. 4 describes our surveillance video coding method. Experiments and conclusion are given respectively in Sec. 5 and 6.

2. PROBLEM ANALYSIS

In traditional methods, updating time is always described by frame number. In this paper, from the view of bit-rate allocation, we propose to describe it with a scale factor δ calculated by

$$\delta_k = \left(\sum_{i=t+1}^{k} R(I_i) \right) \Big/ R(I_t), \text{ where } I_t = Bg \quad (1)$$

In Eq. 1, δ_k is the value of scale factor δ at the k^{th} frame I_k, R(A) means the length of encoding bits for an input frame A, and Bg is the current background frame. Eq. 1 describes the ratio between sum of the encoding bits for frames after a background frame and the bit-cost of the background frame. From the definition we can see, while encoding a sequence, each updating time corresponds to a value of δ uniquely. Therefore, we can use scale factor δ as a yardstick for judging whether the background frame should be updated.

In surveillance video coding, background encoding and the gap between QP of background and general frames determines their bit cost. The gain of encoding general frames with background frame as reference reflects the influence of the background frame. In order to build up an adaptive background updating model for different sequences at different encoding QPs, we analyze what input parameters should be used in the built up model to represent these three factors.

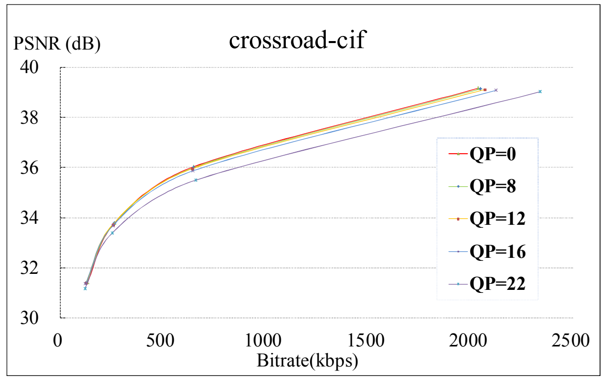

Fig. 2. RD-curve with different QP encoding background

2.1. Gap between QP of background and general frames

Before analyzing the QP gap, we must evaluate which QP should be used for the background frame. In the recent background modeling based surveillance video coding works [5-7], background frame is referenced for many following frames after it is encoded into code stream. Although a high-quality background increases the total bits, it can also save bits for the following frames and improve their coding efficiency. To evaluate which QP is better, different QPs for background frame have been used to encode the sequence *crossroad*. The experimental result is shown in Fig. 2.

From the RD-curve we can see, encoding background frame with QP equal to 0 facilitates an optimal performance. As the QP increases, the quality of background frame becomes worse and larger prediction error will be produced while coding with the background frame as reference. As for the searched vector in motion estimation, it is also less likely to find the best matched MB. Therefore, we can conclude that the background should be encoded with QP equal to 0.

Consequently, to achieve the best coding efficiency and build a better updating model, the QP of general frames can be utilized as input parameter for representing the gap between QP of background and general frames. The result and conclusion are convincing because scene change seldom happens in surveillance video, so one updated background frame can apply to dozens of following general frames to improve their coding efficiency.

2.2. Encoding gain with background frame as reference

As different sequences own different background-pixel proportions, the encoding gain with modeling background frame as reference differs. Therefore, the practical encoding gain is an important factor that influences the sequence-adaptive background updating. In order to find a parameter to describe the encoding gain, we make an analysis through the following experiment. As Fig. 3 shows, F and B respectively denote the average bit-cost during a certain period of time before and after a background frame. The Y axis coordinate means the average bit-rate saving in the coding method using background as long-term reference. From the result we can see, the performance gain increases as the ratio of F/B grows up.

With the experimental results we can conclude that, the average bit ratio before and after a background frame F/B can represent the performance gain of using background as reference. This is also reasonable, because the more static background pixels a sequence owns, the more bit saving can be obtained after a background frame is inserted.

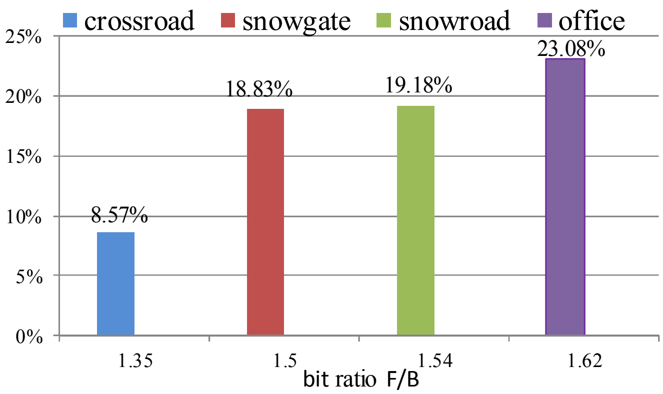

Fig. 3. The encoding gain with background as reference

2.3. Background Encoding

Usually, number the encoding bits of the background frame is very large, so the background updating time should be extended to avoid the burst bit-rate increase. However, this brings a problem that the background cannot be updated in time which significantly decreases the efficiency of frames using the background as encoding reference. Therefore, if we can encode the background using less bits, the background frame can be updated more timely. Actually, there are lots of similar parts in two consecutive background frames, and the similar blocks in the latter background is not certainly better than that in the previous one. With this feature, we conduct an experiment to compare two consecutive background frames using strict threshold judgment and the result is shown in Fig. 4. The unchanged MB compared with previous background is marked with red color, and changed MB is marked with green color. From the distribution of changed MBs and unchanged MBs we can see, most MBs are classified into unchanged MBs although strict threshold judgment was used.

This experimental result enlightens us to save the encoding bits of the background frame. While encoding the current background frame, the previous background frame can be used as reference based on the distribution matrix Ψ of the unchanged background blocks. Thus to shorten the updating time of the background frame, the encoding method Φ based on Ψ is also a necessary input parameter for our updating model.

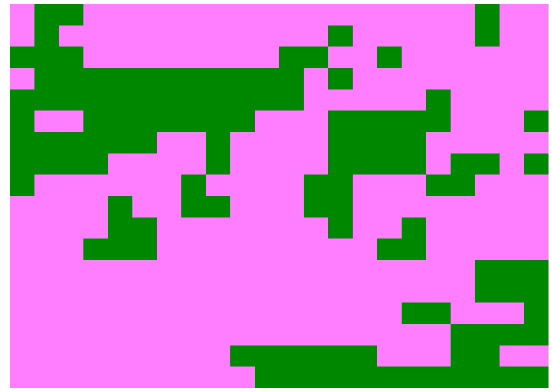

Fig. 4. The distribution of changed and unchanged MB

3. Background Updating Model

Through the analysis in section 2, we obtain the three input parameters for the sequence adaptive background updating model. These three input parameters are QP of the general frame, the bit ratio *F/B* and the distribution matrix Ψ of unchanged background MB. Taking them into consideration, we build a background updating model described as follows:

$$BG_i = \begin{cases} \Phi(BG_{i-1}, U_i, \Psi) & If\ \ \delta_i > Th = \Gamma(QP, \frac{F}{B}) \\ BG_{i-1} & Otherwise \end{cases} \quad (2)$$

In Eq. 2, BG_i and BG_{i-1} respectively stand for the current and previous background frame. U_i is the current original background frame needed to be encoded. *D* represents the encoding bits of previous background and *sum* represents the sum bits of general frame. *Φ* means the background encoding method based on the distribution matrix Ψ, and *Th* denotes the threshold calculated from F/B and QP by function Γ.

This model indicates the following background updating strategy at the ith frame: if δ_i is not larger than threshold calcualted with F/B and QP by a function Γ, the background frame will not be updated. Otherwise, we will use original input frames to generate a new background frame and encode it utilizing the method *Φ* based on the distributon matrix Ψ. And then, we can reconstruct the background to obtain the updated BG_i.

From the expression we can see, the calculation method for adaptive threshold and the distribution matix Ψ play an important role in our updating model. Therefore, the derivation process of Γ for the adaptive threshold *Th* and the distribution matix Ψ are introduced in the following parts.

3.1 Derivation of the Adaptive Threshold

As referred in Eq. 2, F/B and QP are two key factors determining the optimal threshold for background updating. Thus to get the calculation model for *Th*, we have done large numbers of experiments to evaluate the relationship between optimal threshold and the two factors of F/B and QP.

In surveillance video coding, the background frame will be used for dozens of following frames. So in our experiments, we set the threshold *Th* with a range from 4 to 40. And for each QP from 22 to 37, we can get a performance curve (as shown in Fig. 5). In this figure, with the input QP of 27, the PSNR gain rises while *Th* is less than 15, and reaches to a peak value at a point where (*Th*, gain) equals to (18, 0.13dB). It means when the *Th* equals to 18, the encoder can get a max gain of 0.13dB. For QPs from 22 to 37, we use the above method to get the curve (as shown in Fig. 5) for each QP. Fig. 5 can be summarized by the curve of optimal *Th* and QP in Fig. 6, where the value of F/B is marked at each point.

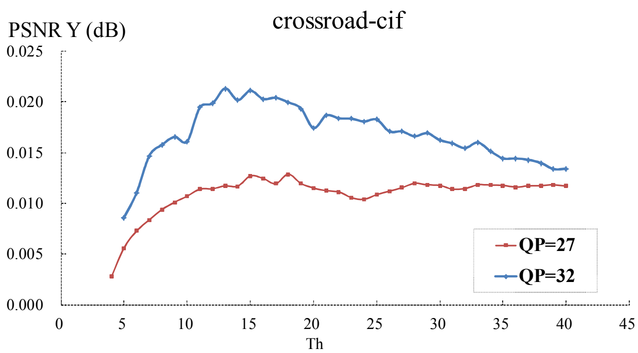

Fig. 5. The performance curve for different *Th*

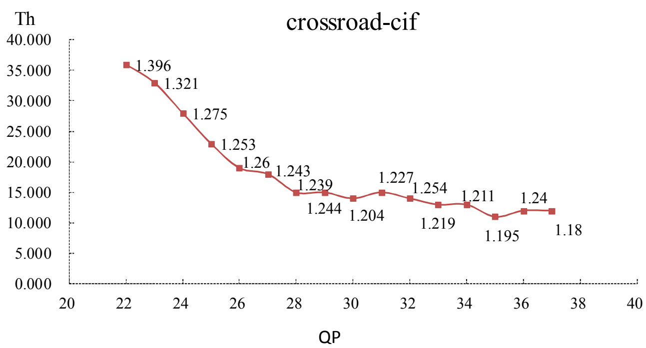

Fig. 6. The function mapping from QP to optimal *Th*

Through the experimental results in Fig 6, we engage to build up a function mapping from (QP, *F/B*) to *Th*. Through comparing different Pairs of (*Th*, QP, F/B) in different training sequences, the function can be summarized as follows:

$$Th = \Gamma(QP, \frac{F}{B}) = \left((\frac{\alpha}{QP} - \beta)^2 + \gamma \right) \times \frac{F}{B} . \qquad (3)$$

3.2. The distribution matrix of unchanged MBs

To save the encoding bit of the background frame, the analysis in 2.3 enlightens us to encode the current background frame with the previous as reference based on the distribution matrix Ψ. So we firstly calculate the distribution matrix Ψ through classifying MBs into unchanged MB and changed MB as follows. Let $C_{i,j}$(k) and $BG_{i,j}$(k) denote the pixel value of position (i, j) in current and previous background frame, where k means the k^{th} pixel in one MB. And η is the threshold for judging whether the value of this pixel has changed. After that, we count the number of changed pixel in one MB. If the number is greater than a constant threshold ε, mark current MB as changed MB. The judgment of each MB can be described by following decision rule:

$$\Psi(i,j) = \begin{cases} 1, & \sum_{k=1}^{256} \left(\mathrm{sgn}\left(\left| C_{i,j}(k) - BG_{i,j}(k) \right| - \eta \right) \right) > \varepsilon \\ 0, & \sum_{k=1}^{256} \left(\mathrm{sgn}\left(\left| C_{i,j}(k) - BG_{i,j}(k) \right| - \eta \right) \right) <= \varepsilon \end{cases} \qquad (4)$$

4. THE PROPOSED CODING METHOD BASED ON THE UPDATING MODEL

4.1. Framework

Based on the derived background model referred in Sec. 3, we further propose an efficient surveillance coding method in this section. To make a clear description, as shown by the framework in Fig. 7, our proposed coding method based on the above analysis consists of the following five modules: the Background Modeling module which uses common background modeling method to generate a background frame, the Initial Background Encoding module that uses traditional intra-coding to encode the initialized background frame, the Updated Background Encoding module that uses our proposed coding method to encode the updated background frame, the Background Decision module for calculating the threshold *Th* to decide whether a background modeling procedure should be enabled, and the Background Selecting module used to select a better background among initial background and updated background. On the basis of this framework, our method includes the following steps:

For each frame, according to the embedded updating model, Background Decision module generates an enable control to decide whether a background frame should be inserted or updated. Only if the enable control is open, the following sub-steps will be operated: if the current background is the first one in a GOP, the Initial Background Encoding module employs intra mode to encode background frame. Otherwise, Updated Background Encoding module encodes the background with the previous background as reference according to the distribution of the unchanged MBs.

After that, the initial background and the recently updated background frame will be stored in their buffers. Before coding each general frame, the Background Selecting module chooses a better background from the buffered initial background frame and the updated background frame as long-term reference frame for the Frame Encoding module.

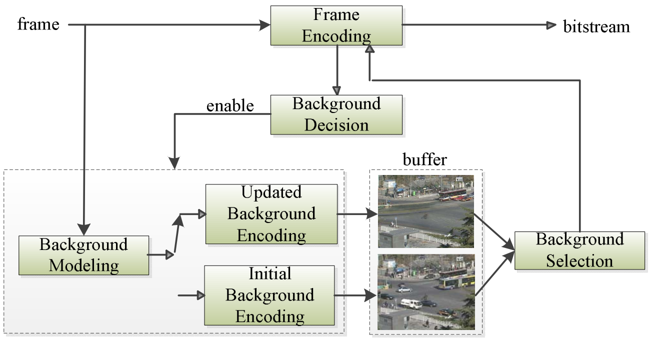

Fig. 7 The framework of the proposed method

4.2. Background Decision

In our summarized background updating model, the background is updated when δ is greater than *Th*. While realizing a practical threshold calculation as shown in Eq. 3., *F* is the average bit of 120 frames before a background frame, and *B* is that after a background frame. Note that 120 is used because an ideal background can be obtained with using 120 general frames as training set.

Our method works as follows: when a background is inserted, memorize the bit-cost of background as *D*. From next frame, accumulate each frame's bit-cost as *sum*, and δ equals to sum/D. After 120 frames, the value of *F/B* can be obtained by the way mentioned above. After that, Eq. 3 is used to get the value of *Th* from QP and F/B. Based on experience, when α equals to 300, β equals to 8 and γ equals to 15, we can get the best encoding performance. Finally, we use *Th* as a threshold. Once δ is greater than *Th*, Background Decision module enables background updating procedure composed of the Background Modeling, Initial Background Encoding and Updated Background Encoding..

4.3. Initial and Updated Background Encoding

To preserve the background quality, the initial background should be high-quality encoded, so only intra prediction mode is used. As for the updated background frame, we employ different strategy for different kind of MBs according to the distribution Ψ. For changed MB, we enable all modes to guarantee its high quality and efficiency. And for unchanged MBs, as the data in previous frame is already high-quality encoded, the quality can be maintained without bit cost only using the SKIP mode. The decision rule can be described by

$$MODE = \begin{cases} SKIP & If\,(MB = unchanged \quad MB) \\ ALL & Otherwise \end{cases} \tag{5}$$

With this mechanism, we decrease the bit-cost of background encoding. Naturally the background updating time will be shortened. Besides, according to the analysis of background QP in 2.1, the QP used in encoding initial and updated background frame should equal to 0.

4.4. Background Selection

After a new background is updated, it can be very helpful for follow-up frames. Nevertheless, this new background may not fit for every frame. In order to decide which is a better reference for encoding the current frame between the initial and the updated background frames, we use the following algorithm for judgment.

Input value:
O(i, j), Bg(i, j), P(i ,j) and C(i ,j): the pixel value at position (i, j) of the original background, updated background, previous frame and the current frame;

Definition:
Function min(a, b): calculate the minimum value between a and b.

Calculation procedure:
For each MB in C, *Begin*
For each position (i, j) in the MB, *Begin*
d1 = d1 + |O(i, j) - C(i, j)|;
d2 = d2 + |Bg(i, j) - C(i, j)|;
d3 = d3 + |P(i, j) - C(i, j)|;
End
SAD1 = SAD1 + min(d1, d3);
SAD2 = SAD2 + min(d2, d3);
End

$$\text{Ref}=\begin{cases} Bg, & \text{SAD1>SAD2} \\ O, & \text{SAD1} \le \text{SAD2} \end{cases}$$

Output value:
Ref : The better long-term reference frame for C.

Fig. 8 Background selecting mechanism

In this mechanism, we directly calculate the SAD of each MB between two frames to judge their similarity. The optimal background frame can be selected according to such judgment: while summing up all the SADs, the minimum result indicates the better reference frame. With this low-complexity method, we can find a better long-term reference for each frame.

5. EXPERIMENTAL RESULTS

5.1. Experimental setup

Traditionally, peak signal to noise ratio (PSNR) and bit-rate are used as metrics for surveillance video coding efficiency. To evaluate our proposed method, H.264/AVC baseline profile encoder (BP), the recent McFIS and a recommend BP with background QP equal to 0 are used as anchors.

For an undisputed comparison, all the anchors and our algorithm are implemented on JM17.2 (very believable H.264/AVC reference software) with the common used Baseline Profile test conditions in the officially accepted proposal (as shown in Table 1). As a believable experiment for evaluating background updating requires long test sequences, eight sequences with more than 2500 frames from AVS workgroup [9] are used(as shown in Fig. 9). Resolution of these eight sequences is from CIF to SD, and the scenes include sunny and dusky (SU/DU), large and small foreground (LF/SF), fast and slow motion (FM/SM).

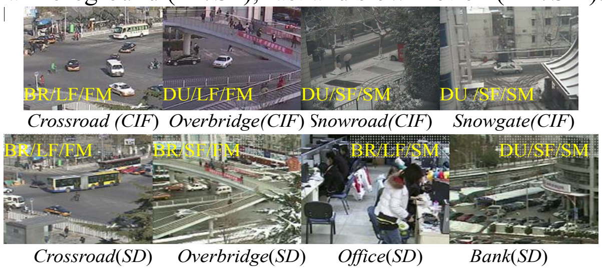

Fig. 9 Test sequences

Table 1. Parameters configuration

Parameter	Value	Parameter	Value
Porfile	Baseline	Used MODE	ALL
Rate Control	Disable	Framerate	25
Entropy Coding	UVLC	Frame Structure	IPPP
Search Range	32	IntraPeriod	0
RD Optimization	High	SAD Method	Hadamard
Motion Search	Fast Full	Reference Num	2

Table 2. Performance comparison

Proposed VS	PSNR(□dB)			Bit-rate(□%)		
	McFIS	BG=0	BP	McFIS	BG=0	BP
CIF(352×288)						
Crossroad	0.48	0.39	1.69	-15.33%	-13.23%	-40.13%
Overbridge	0.50	0.47	1.05	-19.6%	-19.18%	-30.78%
Snowgate	0.57	0.35	2.42	-24.24%	-16.36%	-69.96%
Snowroad	0.72	0.58	1.88	-31.49%	-27.58%	-57.16%
Avg.	**0.57**	**0.45**	**1.76**	**-22.67%**	**-19.09%**	**-49.51%**
SD(720×576)						
Bank	0.46	0.25	1.90	-16.54%	-8.82%	-65.77%
Crossroad	0.46	0.39	1.54	-16.52%	-14.65%	-39.16%
Office	0.41	0.38	1.28	-17.9%	-17.15%	-45.22%
Overbridge	0.48	0.22	2.32	-19.9%	-13.39%	-65.24%
Avg.	**0.45**	**0.31**	**1.75**	**-17.72%**	**13.5%**	**53.85%**

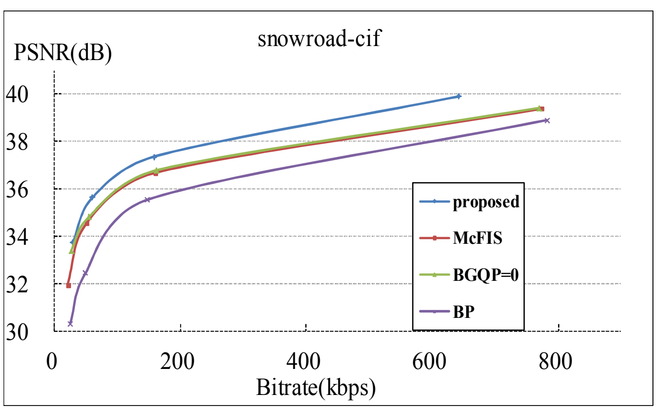

Fig. 10 The RD-curve of snowroad

5.2. Experimental results

As shown in Table 2, due to our timely and bit-saving background updating model, our method realizes high efficiency coding performance on surveillance video. Compared with H.264/AVC baseline profile encoder, our proposed method achieves an average PSNR gain of 1.76dB on CIF and 1.75dB on SD. And compared to McFIS, our proposed method can also achieve an average PSNR gain of 0.57dB on CIF and 0.45dB on SD surveillance stream, with 22.67% and 17.72% bit-rate saved meanwhile.

6. CONCLUSION

In this paper, we propose an efficient surveillance coding method based on a timely and bit-saving background updating model. One significant advantage of our method is that our background updating model is proposed from the view of bit-rate allocation between background frames and general frames. Besides, with a bit-saving background encoding method based on the distribution of unchanged MBs, background frame is updated more in-timely and general frames are encoded more efficiently. Results show that compared to McFIS, our method achieves an average PSNR gain of 0.57dB and 0.45dB on SD surveillance stream, with 22.67% and 17.72% bit-rate saved meanwhile. In the future, we will concentrate on accurate model parameters to further improve the encoding efficiency.

ACKNOWLEDGMENT

This work is partially supported by grants from National Basic Research Program of China under contract No. 2009CB320906, the Chinese National Natural Science Foundation under contract No. 61035001, No. 61171139and No. 61072095, and Fok Ying Dong Education Foundation under contract No. 122008.

7. REFERENCES

[1] ITU-T Recommendation H.120, Codec for videoconferencing using primary digital group transmission, 1984.
[2] M. Paul, W. Lin, C. T. Lau, et al., “Video coding using the most common frame in scene,” ICASSP, 2010.
[3] D. -S. Lee, “Effective Gaussian mixture learning for video background subtraction,” TPAMI, 2005.
[4] M. Haque, M. Murshed and M. Paul, “Improved Gaussian mixtures for robust object detection by adaptive multi-background generation,” ICPR, 2008.
[5] X. Zhang, L. Liang, Q. Huang et al., “An Efficient Coding Scheme for Surveillance Videos Captured by Stationary Cameras,” VCIP, 2010.
[6] X. Zhang, L. Liang, T. Huang et al., “A background model based method for transcoding surveillance videos captured by stationary camera,” PCS, 2010.
[7] D. Gutchess, M. Trajkovic, E. Cohen-Solal, et al., “A Background Model Initialization Algorithm for Video Surveillance,” ICCV, 2001.
[8] Manoranjan Paul, Weisi Lin, Chiew Tong Lau, et al., “MCFIS: better I-frame for video coding,” ISCAS 2010.
[9] ftp://124.207.250.92/public/seqs/video.