A Background Modeling Scheme Based on High Efficiency Motion Classification for Surveillance Video Coding

Pei Liao[1], Xiaofeng Huang[2], Huizhu Jia[2,*], Kaijin Wei[2], Binbin Cai[2], Guoqing Xiang[1], and Don Xie[2]

[1] SECE of Shenzhen Graduate School, Peking University, 518055, Shenzhen, China
[2] National Engineering Laboratory for Video Technology, Peking University, Beijing, China
{pliao,xfhuang,hzjia,kjwei,gqxiang,xdxie}@jdl.ac.cn

Abstract. Recently, high-efficiency video coding becomes more and more demanded as the explosive requirements of network bandwidth and storage space for surveillance video applications. In this paper, we propose a background modeling scheme based on high efficiency motion classification. Firstly, pixels at each location are classified into three motion states, namely the static, the gentle motion and the severe motion states, according to the motion vectors of the corresponding current block and neighboring blocks. Then based on the classification and pixel differential value, the segmentation is performed for the co-located pixels in the training frames, and the mean pixel value of each segment can then be calculated. Finally, the background modeling frame can be obtained by an optimized weighted average of the segmented mean pixel values. Experimental results show that our proposed scheme achieves an average PSNR gain of 0.65dB than the AVS surveillance baseline video encoder, and it gets the best performance among several high efficiency background modeling methods in fast motion and large foreground sequences.

Keywords: surveillance video coding, motion classification, background modeling, weighted average.

1 Introduction

Surveillance video systems are widely used for safety and communication applications recently. The huge required network bandwidth and the increasing demands of storage space are two key challenges in its applications. The compression efficiency of existing video coding standards, like H.264 [1] and AVS [2], is usually not high enough because they are basically designed for general video applications. Consequently, it is necessary to take the specifics of the surveillance video, like static background, into account for high-efficiency surveillance video coding.

In most surveillance applications, cameras are usually fixed at a certain location and direction to capture the scene for a long time. And the background in these frames

* Corresponding author.

W.T. Ooi et al. (Eds.): PCM 2014, LNCS 8879, pp. 44–53, 2014.

is always the same except the noise generated by the camera or the slow change of the environment. Typically, a framework of background modeling based video encoder is used which is shown as Fig. 1. In this framework, a good background modeling frame which is long-term referenced by surveillance video sequences will improve the coding efficiency dramatically.

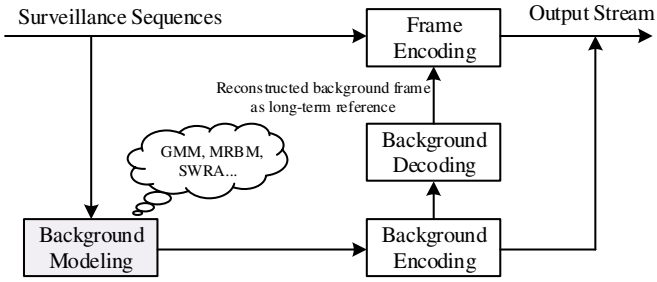

Fig. 1. Framework of background modeling based video encoder

Recently, many effective background modeling methods have been proposed, which can be classified into two categories named parametric and non-parametric methods. In parametric methods, an adaptive Gaussian mixture model (GMM) [3] is used for background subtraction and object detection, with better Gaussian mixture quality compared to earlier GMM methods. However, the huge bandwidth requirement limits its application in hardware realization. In non-parametric methods, an effective background generation method named most reliable background mode (MRBM) [4] is used, which generates clear background and is robust to noise and camera shaking. In paper [5], a segment-and-weight based running average (SWRA) method is proposed to alleviate the computational complexity of background modeling, and achieves comparable coding performance as GMM-5 method.

Actually, the background modeling methods in papers [3-4] are designed for object detection or background subtraction, while the main objective of background modeling in Fig. 1 is to save encoded bits for background frame and provide better long-term reference for surveillance video sequences. Although the algorithm in work [5] is oriented for video coding, besides the pixel level information, the global level information, such as motion vectors, can be adaptively used to improve surveillance video coding performance. In this paper, a motion classification based background modeling scheme is proposed. This scheme firstly classifies pixels at each location into three motion states, namely the static, the gentle motion and the severe motion states. Secondly, the segmentation of the co-located pixels in the training frames is performed based on both its motion state and pixel differential value, and the mean pixel value of each segment can be calculated. Finally, the background modeling frame can be obtained by an optimized weighted average of the segmented mean pixel values.

The rest of the paper is organized as follows. Section 2 in detail describes the proposed method. Section 3 presents the experimental results. And conclusion of this paper is made in section 4.

2 Proposed Method

Background modeling increasingly plays an important role in surveillance video coding, which aims at segmenting foreground and background in object-oriented video encoder [6] and providing better long-term prediction efficiency in background-prediction-based video encoder as shown in Fig. 1. In this section, a motion classification based background modeling scheme is presented for background-prediction-based video encoder.

The overall framework of the proposed scheme is shown in Fig. 2. The motion classification is based on the motion vectors of the corresponding current block and the neighboring blocks. For the (x,y) pixels in the training frames, the segmentation is based on both its motion state and pixel differential value (SAD). The mean pixel value avg_i and the segment length L_i can be calculated for the i_{th} segment, respectively. The weight w_s_i for i_{th} segment is proportional to the L_i, and a weighted average of the previous segments and the current segment is calculated for background modeling pixel value BGV. The background modeling pixel value BGV is updated at the end of each segment. When BGV value equals to 0 at the last of the training frames, value 128 is directly assigned to BGV. The algorithm of our proposed scheme is shown in Fig. 3, which is a detailed description of Fig. 2.

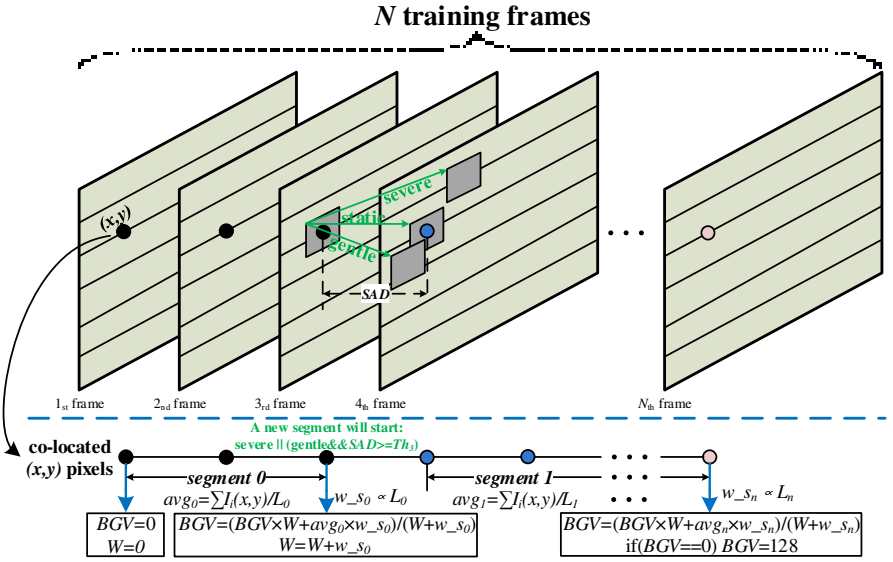

Fig. 2. Framework of the proposed scheme

(1) Initialization: For each (x,y) pixel location, the background modeling pixel value BGV and the total previous segment weight W are initialized to 0. The first segment length L and its mean pixel value avg are also initialized to 0.

(2) Motion classification: There are three motion states defined in our proposed method, namely the static, the gentle motion and the severe motion states. Each pixel

Input:
T = $\{I_i=f_i(x,y)|i=1\sim N\}$, where $f_i(x,y)$ is i_{th} frame in the N training frames.
Background Modeling Procedure:
For each pixel location (x,y), ***Begin***
$\{BGV, W, L, avg\}=0$; (1)
While $i=1\sim N$, ***Begin***
if$((\text{abs}(MV_x)+\text{abs}(MV_y))_c<\text{Th}_1$ && $\forall(\text{abs}(MV_x)+\text{abs}(MV_y))_n<\text{Th}_1)$ (2)
cs = static;
else if$(\text{abs}(MV_x)_c<\text{Th}_2$ && $\text{abs}(MV_y)_c<\text{Th}_2)$
cs = gentle motion;
else
cs = severe motion;
if((static || (gentle motion && $\text{abs}(I_i-I_{i+1})<Th_3$)) && $i<N$) ***Begin*** (3)
$avg = (avg\times L+I_i)/(L+1)$;
$L = L+1$;
End
else ***Begin*** /*a new segment*/
$w\_s \propto L$
$BGV = (BGV\times W+ avg\times w\_s)/(W+w\_s)$;
if($i==N$ && $BGV==0$)
$BGV = 128$;
$W = W+w\_s$;
L = avg = 0;
End
End
End
Output the background frame. (4)

Fig. 3. Algorithm of the proposed scheme

in the training frames will be classified into these three states based on the motion vectors of the corresponding current block ($(MV_x,MV_y)_c$) and eight neighboring blocks ($(MV_x,MV_y)_n$) as shown in Fig. 4(a). The motion vector is searched by using the $(i+1)_{th}$ frame as reference for the current i_{th} frame as in Fig. 2. For simplicity, the motion vectors derived from motion estimation module in encoder can be directly used instead, which will decrease the complexity greatly. The static state is assigned when the addition of absolute motion vector values of current block and eight neighboring blocks are smaller than a threshold, simultaneously. In order to integrate our proposed scheme into existing hardware architecture [7] easily, only four neighboring searched blocks instead of total eight neighboring blocks are used for motion classification as shown in Fig. 4(b). The threshold Th1 is set to 1 for the tolerance of noise and small camera shaking.

In order to segment the co-located pixels in the training frames accurately, two motion states are distinguished in the proposed method as shown in Fig. 2. The partition of the motion states is based on the actual motion vector value of the current block,

when its absolute values of horizontal and vertical motion vector are both less than a threshold Th_2, the gentle motion is assigned, and otherwise the severe motion is assigned. The principle of the motion state partition is shown in Fig. 5. When the motion vector exceeds the block size (b_s), as MV_b in Fig. 5, there will be no overlap between the reference block and the current block. Accordingly, the current block is named as a "*scene change block*" and is set to the severe motion state. Otherwise, for the MV_a case as shown in Fig. 5, the current block is set to the gentle motion state. Thus, the threshold Th_2 is equal to the block size b_s.

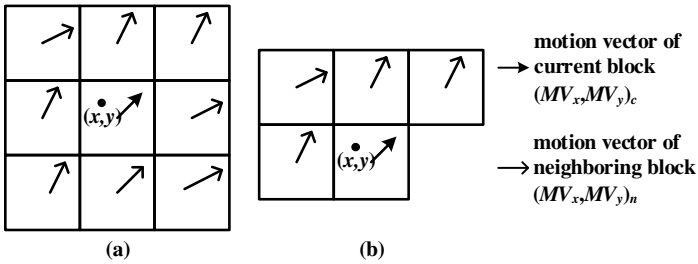

Fig. 4. (a) Motion vectors of current block and eight neighboring blocks for motion classification. (b) Motion vectors of current block and four neighboring blocks for motion classification.

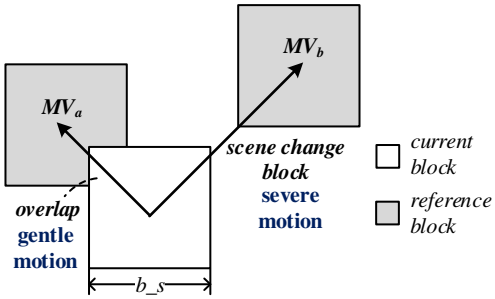

Fig. 5. Principle of motion state partition

(3) Segmentation and parameter calculation: When the current pixel is in a severe motion state or the pixel differential value is larger than the third threshold Th_3 in gentle motion state as in Fig. 2, a new segment will start. Otherwise, the segment is continued and its mean pixel value *avg* and segment length L are calculated, separately. The threshold value Th_3 is dynamically adjusted as Fig. 3 in [8], except that the $Diff(m,n)$ is the difference between $I_i(m,n)$ and $I_{i+1}(m,n)$.

The weight of the segment w_s is proportional to the segment length L. In order to evaluate the proportion accurately, six piecewise functions as shown in Fig. 6 are listed for testing. The test platform is illustrated as in Part 3.1. When the segment length L is smaller than a threshold Th_0 as in Fig. 6, the segment weight w_s is assigned to 0 in order to eliminate false segments. Typically, the value of Th_0 is set to $N/20$. For the segment length L larger than the Th_0, the six piecewise functions are listed based on the fact that the larger segment weight w_s is assigned with larger

segment length *L*. As shown in Table 1, compared to the AVS surveillance baseline video encoder, the quadratic function achieves the best performance among these six piecewise functions, which achieves an average PSNR gain of 0.645dB and an average bitrate decrease of 21.24%. In the piecewise linear functions as shown in Fig. 6(a), the linear function achieves the lowest performance (0.617dB PSNR gain in average), and the piecewise1 function achieves the highest performance (0.630dB PSNR gain in average). In the piecewise power functions as shown in Fig. 6(b), the performance is decreasing as the order of the power function increased. Table 1 illustrates that the quadratic function accords with the proportional relationship between the segment weight *w_s* and the segment length *L*. This conclusion will be useful for other background modeling methods.

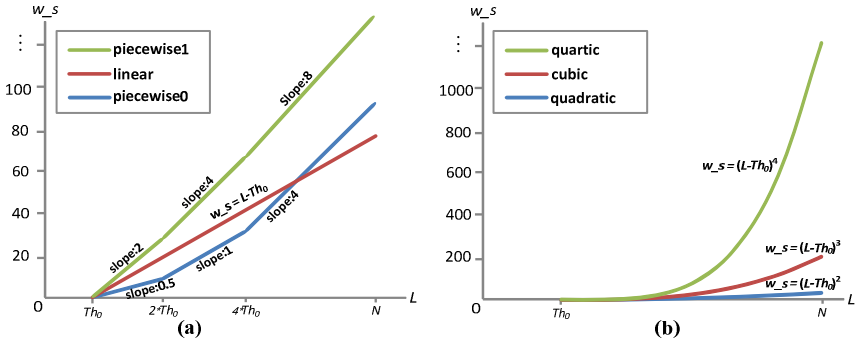

Fig. 6. The curves of six piecewise functions. (a) Piecewise linear functions. (b) Piecewise power functions.

Table 1. Performance comparisons of the six piecewise functions (VS SM-2)

Format	Sequence	piecewise0		linear		piecewise1		quadratic		cubic		quartic	
		PSNR (ΔdB)	Bitrate (Δ%)	PSNR (ΔdB)	Bitrate (Δ%)	PSNR (ΔdB)	Bitrate (Δ%)	PSNR (ΔdB)	Bitrate (Δ%)	PSNR (ΔdB)	Bitrate (Δ%)	PSNR (ΔdB)	Bitrate (Δ%)
CIF	*Crossroad*	0.533	-13.89	0.526	-13.71	0.550	-14.29	0.573	-14.86	0.550	-14.31	0.497	-13.05
	Overbridge	0.575	-17.91	0.567	-17.75	0.585	-18.24	0.596	-18.65	0.557	-17.46	0.496	-15.74
	Snowroad	0.736	-25.07	0.734	-25.04	0.740	-25.22	0.729	-24.78	0.702	-23.95	0.690	-23.56
	Snowway	0.467	-17.13	0.461	-16.97	0.477	-17.57	0.511	-18.88	0.485	-18.03	0.406	-15.17
SD	*Bank*	0.814	-31.77	0.818	-31.78	0.828	-32.13	0.846	-32.85	0.822	-31.92	0.792	-30.60
	Crossroad	0.547	-16.35	0.547	-16.35	0.558	-16.65	0.567	-17.05	0.541	-16.20	0.493	-14.89
	Office	0.403	-16.43	0.399	-16.31	0.412	-16.74	0.421	-17.05	0.403	-16.43	0.371	-15.36
	Overbridge	0.882	-24.86	0.880	-24.83	0.899	-25.31	0.918	-25.80	0.886	-24.97	0.832	-23.40
Avg.		0.620	-20.43	0.617	-20.34	0.630	-20.77	**0.645**	**-21.24**	0.618	-20.40	0.57	-18.97

Instead of buffering each segment weight such as w_s_0, w_s_1…, *W* is used to represent the total weights of previous segments in order to save memory as shown in Fig. 7. The background modeling pixel value *BGV* is updated at the end of each segment, by weighted average of total previous segments and current segment w_s_c.

The *BGV* value may be still 0 when it runs into the last training frame *N*. This is because the pixel is always in the severe motion state or its segment length *L* is too short, all of which indicate that the pixel is always in foreground. Pixel value 128 is assigned to that pixel as in [6], which will improve the compression efficiency.

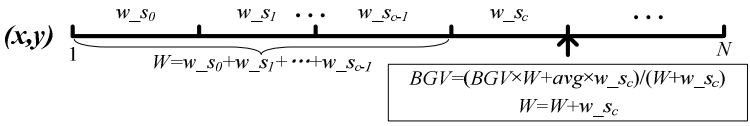

Fig. 7. The calculation of *BGV* and *W*

(4) Output: After the background modeling process at the last training frame, the background picture can be generated and output.

From the above analysis, the proposed motion classification based background modeling method mainly depends on the motion vector, pixel differential value and piecewise weighting function. Besides, the proposed method updates the model value of each pixel at the end of a segment, and background frame is updated at the end of training frames. As a result, the proposed method satisfies the demand of periodical background updating and real-time requirement of hardware realization. In section 3, the performance of the proposed method is evaluated.

3 Experimental Results

3.1 Experiment Setup

In this section, AVS surveillance baseline video encoder (SM-2) is used to evaluate the efficiency of the proposed method. Besides our proposed method, other three background modeling methods, the SWRA method in [5], the GMM using 5 models for each pixel in [3], and the MRBR method in [4], are implemented and embedded into SM-2 encoder for comparison. Corresponding Encoders are named SM-MC, SM-SW, SM-GMM5 and SM-MR. The SM parameters configuration is listed in Table 2. In the table, profile ID 44 corresponds to the surveillance video coding.

Table 2. Parameters configuration.

Parameter	Vaule	Parmeter	Vaule	Parmeter	Vaule
Profile ID	44	Search Range	32	RD Optimization	Used
Rate Control	Disable	Reference Num.	2	Frame Structure	IPPP
Entropy Coding	CABAC	Motion Est.	UMHexagonS	QP of P frame	24,31,37,44
Intra period	25	Use Mode	ALL	Frame rate	25

For fair comparison, background frames in the encoders should be updated and encoded simultaneously. A sequence structure dividing input frames into super group of pictures (S-GOP) is used for the surveillance video encoders as shown in Fig. 8.

The background frame Bg_1 which is long-term referenced by sequences in S-GOP$_1$ is modeled by TrainSet$_0$, and encoded at the first frame of S-GOP$_1$. The same structure will be utilized for S-GOP$_n$, where Bg_n is modeled by TrainSet$_{n-1}$ in S-GOP$_{n-1}$. In our experiments, the number of training frames is set to 120 and the size of an S-GOP is set to 600. Besides, to simplify the bit-allocation of the background frame, the quantization parameter for background frame is equal to that of P frames minus 5, and only intra predictions are utilized.

Eight surveillance sequences captured by static camera are used in our experiments. For the data set, the first 900 frames of SD surveillance sequences of *Crossroad*, *Overbridge*, *Office*, *Bank* and CIF sequences of *Crossroad*, *Overbridge*, *Snowroad*, *Snowgate* [9] are used to evaluate the five encoders. The content of these sequences are shown in Fig. 9. In these sequences, *Crossroad* (CIF), *Overbridge* (CIF&SD) and *Office* (SD) have relatively large foreground regions and fast motion.

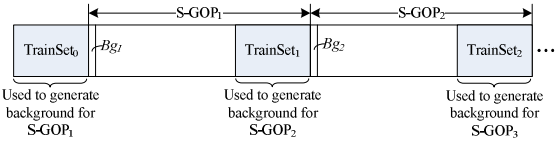

Fig. 8. Sequence structure for background modeling

Fig. 9. Test sequences and their content

3.2 Results

In this part, the encoding performance of the SM-MC, SM-GMM5, SM-SW, SM-MR and the SM-2 encoders are compared. Results show that the proposed motion classification based background modeling scheme encoder (SM-MC) achieves better performance than the other three background modeling based video encoders, especially in fast motion and large foreground sequences.

Compared to the SM-GMM5/SM-SW/SM-MR/SM-2 encoders, the SM-MC encoder achieves both PSNR gain and bitrate reduction. The SM-MC achieves an average gain of 0.02dB/0.06dB/0.12dB/0.65dB than the SM-GMM5/SM-SW/SM-MR/SM-2 encoders. For CIF sequences, our method achieves an average PSNR gain of 0.03dB/0.05dB/0.13dB gain than the SM-GMM5/SM-SW/SM-MR encoders. In

these four CIF sequences, the SM-MC encoder shows the highest gain for Crossroad and Overbridge sequences, and SM-GMM5 shows the best performance for Snowroad and Snowway sequences. And an average PSNR gain of 0.02dB/0.07dB/0.12dB is achieved than the SM-GMM5/SM-SW/SM-MR encoders for SD sequences. In SD sequences, the SM-MC encoder shows the best performance for Office and Overbridge sequences, and SM-GMM5 encoder gets the highest gain for Bank and Crossroad sequences. These all imply that our proposed method shows the highest performance in large foreground and fast motion sequences, and the existing GMM-5 method shows the highest gain in small foreground and slow motion sequences. The SM-M2 encoder shows the worst performance in these five encoders, and the second and the third worst are SM-MR and SM-SW encoders, respectively. An example rate-distortion curve of Office (SD) sequence is shown in Fig. 10.

Table 3. Performance Comparisons

Format	Sequence	SM-MC VS SM-GMM5		SM-MC VS SM-SW		SM-MC VS SM-MR		SM-MC VS SM-2	
		PSNR (ΔdB)	Bitrate (Δ%)	PSNR (ΔdB)	Bitrate (Δ%)	PSNR (ΔdB)	Bitrate (Δ%)	PSNR (ΔdB)	Bitrate (Δ%)
CIF	*Crossroad*	0.08	-1.99	0.08	-2.05	0.21	-5.19	0.57	-14.86
	Overbridge	0.04	-1.01	0.06	-1.88	0.16	-4.35	0.60	-18.65
	Snowroad	0	0.26	0.02	-0.64	0.04	-1.22	0.73	-24.78
	Snowway	-0.02	0.93	0.03	-0.79	0.09	-3.27	0.51	-18.88
CIF Avg.		0.03	-0.45	0.05	-1.34	0.13	-3.51	0.60	-19.29
SD	*Bank*	0	0.11	0.06	-2.15	0.07	-2.94	0.85	-32.85
	Crossroad	-0.01	0.25	0	-0.06	0.09	-2.44	0.57	-17.05
	Office	0.05	-2.01	0.16	-4.79	0.15	-4.75	0.42	-17.05
	Overbridge	0.02	-0.6	0.06	-1.46	0.15	-4.58	0.92	-25.80
SD Avg.		0.02	-0.56	0.07	-2.12	0.12	-3.68	0.69	-23.19
Avg.		**0.02**	**-0.51**	**0.06**	**-1.46**	**0.12**	**-3.59**	**0.65**	**-21.24**

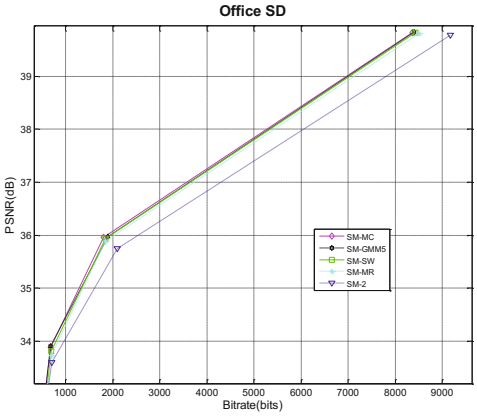

Fig. 10. An example rate-distortion curve of *Office* (SD) sequence

4 Conclusion

In this paper, a high-efficiency motion classification based background modeling scheme is proposed. Firstly, pixels at each location are classified into three motion states, namely the static, the gentle motion and the severe motion states. Then based on the classification and pixel differential value, the segmentation is performed for the co-located pixels in the training frames, and the mean pixel value of each segment can be calculated. Finally, the background modeling frame can be obtained by an optimized weighted average of the segment mean pixel values. In order to resolve the invalid background modeling in fast motion sequences in existing methods, our method takes the block level motion status into consideration for more accurate background modeling. Experimental results show that our proposed scheme achieves an average PSNR gain of 0.65dB than the AVS surveillance baseline video encoder, and gets the best performance among several high efficiency background modeling methods in fast motion and large foreground sequences.

Acknowledgements. This work is partially supported by grants from the Chinese National Natural Science Foundation under contract No.61171139, and National High Technology Research and Development Program of China (863 Program) under contract No.2012AA011703.

References

1. Wiegand, T., Sullivan, G., Bjøntegaard, G., Luthra, A.: Overview of the H.264/AVC Video Coding Standard. IEEE Transactions on Circuits and Systems for Video Technology 13(7), 560–576 (2003)
2. Gao, W., Ma, S., Zhang, L., Su, L., Zhao, D.: AVS Video Coding Standard. Intelligent Multimedia Communication: Techniques and Applications, 125–166 (2010)
3. Haque, M., Murshed, M., Paul, M.: Improved Gaussian mixtures for robust object detection by adaptive multi-background generation. In: IEEE International Conference on Pattern Recognition, pp. 1–4 (2008)
4. Liu, Y., Yao, H., Gao, W., Chen, X., Zhao, D.: Nonparametric background generation. In: IEEE International Conference on Pattern Recognition, vol. 4, pp. 916–919 (2006)
5. Zhang, X., Tian, Y., Huang, T., Gao, W.: Low-Complexity and High-Efficiency Background Modeling for Surveillance Video Coding. In: IEEE International Conference on Visual Communications and Image Processing, pp. 1–6 (2012)
6. Zhang, S., Wei, K., Jia, H., Xie, X., Gao, W.: An efficient foreground-based surveillance video coding scheme in low bit-rate compression. In: IEEE International Conference on Visual Communications and Image Processing, pp. 1–6 (2012)
7. Yang, W., Yin, H., Gao, W., Qi, H., Xie, X.: Multi-stage motion vector prediction schedule strategy for AVS HD encoder. In: IEEE International Conference on Consumer Electronics, pp. 339–340 (2010)
8. Zhang, X., Tian, Y., Liang, L., Huang, T., Gao, W.: Macro-Block-Level Selective Background Difference Coding for Surveillance Video. In: IEEE International Conference on Multimedia and Expo, pp. 1067–1072 (2012)
9. `ftp://124.207.250.92/public/seqs/video/` (accessed by AVS member)