Available online at www.sciencedirect.com

SCIENCE DIRECT®

Signal Processing: *Image Communication* 20 (2005) 255–264

SIGNAL PROCESSING: IMAGE COMMUNICATION

www.elsevier.com/locate/image

Moving object detection in wavelet compressed video☆

B. Ugur Töreyin[a,*], A. Enis Çetin[a], Anil Aksay[a], M. Bilgay Akhan[b]

[a] *Department of Electrical and Electronics Engineering, Bilkent University, TR-06800 Bilkent, Ankara, Turkey*
[b] *Visioprime, 30 St. Johns Rd., St. Johns, Woking, Surrey, GU21 7SA, UK*

Received 3 June 2004; accepted 8 December 2004

Abstract

In many surveillance systems the video is stored in wavelet compressed form. In this paper, an algorithm for moving object and region detection in video which is compressed using a wavelet transform (WT) is developed. The algorithm estimates the WT of the background scene from the WTs of the past image frames of the video. The WT of the current image is compared with the WT of the background and the moving objects are determined from the difference. The algorithm does not perform inverse WT to obtain the actual pixels of the current image nor the estimated background. This leads to a computationally efficient method and a system compared to the existing motion estimation methods.
© 2005 Published by Elsevier B.V.

Keywords: Moving region detection; Wavelet compressed video

1. Introduction

In many surveillance systems, the video is compressed intra-frame only without performing motion compensated prediction due to legal reasons. Courts do not accept predicted image frames as legal evidence in many countries [5]. As a result, a typical surveillance video is composed of a series of individually compressed image frames. In addition, many practical systems have built-in VLSI hardware image compressors directly storing the compressed video data coming from several cameras into a hard-disc. The main reason for this is that standard buses used in PC's cannot handle the raw multi-channel video data. In this paper, it is assumed that the video data is available in wavelet compressed format. In many multi-channel real-time systems, it is not possible to use uncompressed video due to available bus and processor limitations. The proposed moving object detection algorithm compares the wavelet transform (WT) of the current image with the WTs of

☆B.U. Töreyin and A.E. Cetin's work is partially funded by EU NoE MUSCLE and Turkish Academy of Sciences, TUBA–GEBIB.

*Corresponding author. Tel.: +90 312 290 1477; fax: +90 312 266 4192.

E-mail addresses: ugur@ee.bilkent.edu.tr (B.U. Töreyin), cetin@ee.bilkent.edu.tr (A.E. Çetin), anil@ee.bilkent.edu.tr (A. Aksay), bilgay.akhan@visioprime.com (M.B. Akhan).

0923-5965/$ - see front matter © 2005 Published by Elsevier B.V.
doi:10.1016/j.image.2004.12.002

the past image frames to detect motion and moving regions in the current image without performing an inverse WT operation. Moving regions and objects can be detected by comparing the WTs of the current image with the WT of the background scene which can be estimated from the WTs of the past image frames. If there is a significant difference between the two WTs then this means that there is motion in the video. If there is no motion then the WTs of the current image and the background image ideally should be equal to each other or very close to each other due to quantization process during compression. Stationary wavelet coefficients belong to the WT of the background. This is because the background of the scene is temporally stationary [3–7,9,13]. If the viewing range of the camera is observed for some time, then the WT of the entire background can be estimated as moving regions and objects occupy only some parts of the scene in a typical image of a video and they disappear over time. On the other hand, pixels of foreground objects and their wavelet coefficients change in time. Non-stationary wavelet coefficients over time correspond to the foreground of the scene and they contain motion information. A simple approach to estimate the WT of the background is to average the observed WTs of the image frames. Since moving objects and regions occupy only a part of the image they can conceal a part of the background scene and their effect in the wavelet domain is cancelled over time by averaging. Any one of the space domain approaches [3,4,6–9,11,13] for background estimation can be implemented in wavelet domain providing real-time performance. For example, the background estimation method in [4] can be implemented by simply computing the WT of both sides of their background estimation equations.

2. Hybrid algorithm for moving object detection

Background subtraction is commonly used for segmenting out objects of interest in a scene for applications such as surveillance. There are numerous methods in the literature [3–7,9,13]. The background estimation algorithm described in [4] uses a simple IIR filter applied to each pixel independently to update the background and use adaptively updated thresholds to classify pixels into foreground and background. This is followed by some post processing to correct classification failures. Stationary pixels in the video are the pixels of the background scene because the background can be defined as temporally stationary part of the video. If the scene is observed for some time, then pixels forming the entire background scene can be estimated because moving regions and objects occupy only some parts of the scene in a typical image of a video. A simple approach to estimate the background is to average the observed image frames of the video. Since moving objects and regions occupy only a part of the image, they conceal a part of the background scene and their effect is cancelled over time by averaging. Our main concern is real-time performance of the system. In Video Surveillance and Monitoring (VSAM) Project at Carnegie Mellon University [4] a recursive background estimation method was developed from the actual image data. Let $I_n(x,y)$ represent the intensity (brightness) value at pixel position (x,y) in the n^{th} image frame I_n. Estimated background intensity value at the same pixel position, $B_{n+1}(x,y)$, is calculated as follows:

$$B_{n+1}(x,y) = \begin{cases} aB_n(x,y) + (1-a)I_n(x,y) \\ \quad \text{if } (x,y) \text{ is non-moving}, \\ B_n(x,y) \\ \quad \text{if } (x,y) \text{ is moving}, \end{cases} \tag{1}$$

where $B_n(x,y)$ is the previous estimate of the background intensity value at the same pixel position. The update parameter a is a positive real number close to one. Initially, $B_0(x,y)$ is set to the first image frame $I_0(x,y)$. A pixel positioned at (x,y) is assumed to be moving if the brightness values corresponding to it in image frame I_n and image frame I_{n-1}, satisfy the following inequality:

$$|I_n(x,y) - I_{n-1}(x,y)| > T_n(x,y) \tag{2}$$

where $I_{n-1}(x,y)$ is the brightness value at pixel position (x,y) in the $(n-1)^{\text{st}}$ image frame I_{n-1}. $T_n(x,y)$ is a threshold describing a statistically significant brightness change at pixel position (x,y). This threshold is recursively updated for each pixel as follows:

$$T_{n+1}(x,y) = \begin{cases} aT_n(x,y) + (1-a)(c|I_n(x,y) - B_n(x,y)|) \\ \quad \text{if } (x,y) \text{ is non-moving}, \\ T_n(x,y) \\ \quad \text{if } (x,y) \text{ is moving}, \end{cases} \tag{3}$$

where c is a real number greater than one and the update parameter a is a positive number close to one. Initial threshold values are set to an experimentally determined value. As it can be seen from (3), the higher the parameter c, higher the threshold or lower the sensitivity of detection scheme. It is assumed that regions significantly different from the background are moving regions. Estimated background image is subtracted from the current image to detect moving regions. In other words all of the pixels satisfying

$$|I_n(x,y) - B_n(x,y)| > T_n(x,y) \tag{4}$$

are determined. These pixels at (x,y) locations are classified as the pixels of moving objects.

3. Moving object detection in wavelet domain

Above arguments and the methods proposed in [8,11] are valid in compressed data domain as well, [9]. In [9], DCT domain data is used for motion detection in video. In our case, a WT-based coder is used for data compression. The WT of the background scene can be estimated from the wavelet coefficients of past image frames, which do not change in time, whereas foreground objects and their wavelet coefficients change in time. Such wavelet coefficients belong to the background because the background of the scene is temporally stationary. Non-stationary wavelet coefficients over time correspond to the foreground of the scene and they contain motion information. If the viewing range of the camera is observed for some time, then the WT of the entire background can be estimated because moving regions and objects occupy only some parts of the scene in a typical image of a video and they disappear over time.

Our method is different than the methods proposed in [7,13]. Because both of these methods are based on frame differencing whereas our method estimates the background in the wavelet domain and subtracts it from the current frame. Estimating the background and using this kind of a hybrid method results in a superior detection performance when compared with the frame differencing-based methods.

Let B be an arbitrary image. This image is processed by a single stage separable Daubechies 9/7 filterbank and four quarter size subband images are obtained. Let us denote these images as $LL(1), HL(1), LH(1), HH(1)$ [1]. In a Mallat wavelet tree, $LL(1)$ is processed by the filterbank once again and $LL(2), HL(2), LH(2), HH(2)$ are obtained. Second scale subband images are the quarter size versions of $LL(1)$. This process is repeated several times in a typical wavelet image coder. A three scale wavelet decomposition of an image is shown in Fig. 1.

Let D_n represent any one of the subband images of the background image B_n at time instant n. The subband image of the background D_{n+1} at time instant $n+1$ is estimated from D_n as follows:

$$D_{n+1}(i,j) = \begin{cases} aD_n(i,j) + (1-a)J_n(i,j) \\ \quad \text{if } (i,j) \text{ is non-moving}, \\ D_n(i,j) \\ \quad \text{if } (i,j) \text{ is moving}, \end{cases} \tag{5}$$

where J_n is the corresponding subband image of the current observed image frame I_n. The update parameter a is a positive real number close to one. Initial subband image of the background, D_0, is assigned to be the corresponding subband image of the first image of the video I_0. In Eqs. (1)–(4), (x,y)'s correspond to the pixel locations in the original image, whereas in (5) and in all the equations in this section, (i,j)'s correspond to locations of subband images' wavelet coefficients.

A wavelet coefficient at the position (i,j) in a subband image is assumed to be moving if

$$|J_n(i,j) - J_{n-1}(i,j)| > T_n(i,j) \tag{6}$$

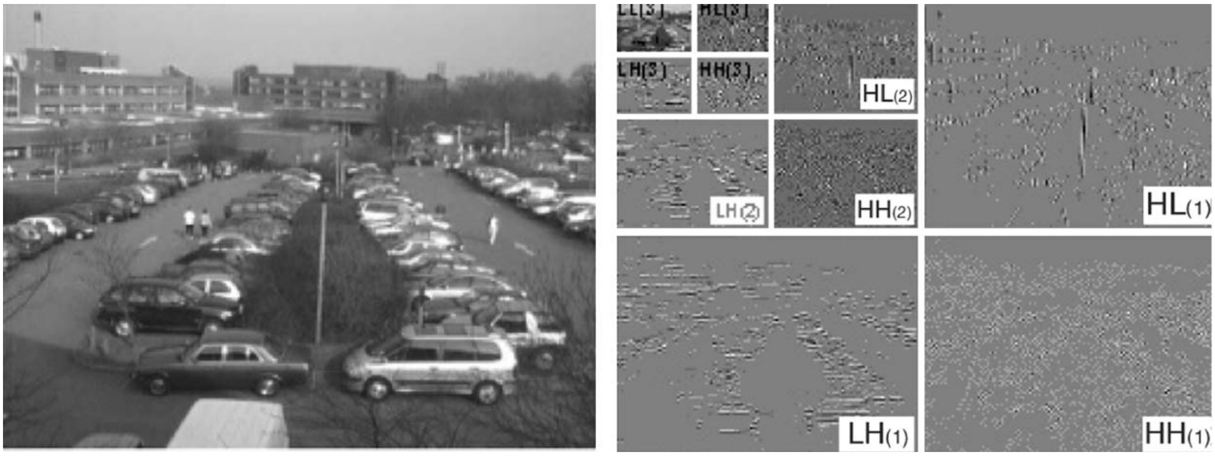

Fig. 1. Original image and its corresponding three levels of the wavelet tree consisting of subband images (luminance data is shown).

where $T_n(i,j)$ is a threshold recursively updated for each wavelet coefficient as follows:

$$T_{n+1}(i,j) = \begin{cases} aT_n(i,j) + (1-a)(b|J_n(i,j) - D_n(i,j)|) \\ \quad \text{if } (i,j) \text{ is non-moving,} \\ T_n(i,j) \\ \quad \text{if } (i,j) \text{ is moving,} \end{cases} \tag{7}$$

where b is a real number greater than one and the update parameter a is a positive real number close to one. Initial threshold values can be experimentally determined. As it can be seen from the above equation, the higher the parameter b, higher the threshold or lower the sensitivity of detection scheme. Estimated subband image of the background is subtracted from the corresponding subband image of the current image to detect the moving wavelet coefficients and consequently moving objects as it is assumed that the regions different from the background are the moving regions. In other words, all of the wavelet coefficients satisfying the inequality

$$|J_n(i,j) - D_n(i,j)| > T_n(i,j) \tag{8}$$

are determined.

It should be pointed out that there is no fixed threshold in this method. A specific threshold is assigned to each wavelet coefficient and it is adaptively updated according to (7). Therefore a recursively updated threshold is assigned to each wavelet coefficient in all subband images.

Once all the wavelet coefficients satisfying the above inequalities are determined, locations of corresponding regions on the original image are determined. If a single stage Haar WT is used in data compression then a wavelet coefficient satisfying (8) corresponds to a 2 × 2 block in the original image frame I_n. For example, if $(i,j)^{\text{th}}$ coefficient of the subband image $HH_n(1)$ (or other subband images $HL_n(1), LH_n(1), LL_n(1)$) of I_n satisfies (8), then this means that there exists motion in a 2 pixel × 2 pixel region in the original image, $I_n(k,m)$, $k = 2i, 2i-1$, $m = 2j, 2j-1$, because of the subsampling operation in the discrete WT computation. Similarly, if the $(i,j)^{\text{th}}$ coefficient of the subband image $HH_n(2)$ (or other second scale subband images $HL_n(2), LH_n(2), LL_n(2)$) satisfies (8) then this means that there exists motion in a 4 pixel × 4 pixel region in the original image, $I_n(k,m)$, $k = 4i, 4i-1$, $4i-2$, $4i-3$ and $m = 4j$, $4j-1$, $4j-2$, $4j-3$. In general, a change in the l^{th} level wavelet coefficient corresponds to a $2^l \times 2^l$ region in the original image.

Visioprime [12] designed a video processing system which feeds the compressed video data in Aware Inc.'s Motion Wavelet format to our system [2]. It uses Daubechies' 9/7 biorthogonal wavelet. In this biorthogonal transform, the number of pixels forming a wavelet coefficient is larger than four but most of the contribution comes from the immediate neighborhood of the pixel $I_n(k,m) = (2i,2j)$ in the first level wavelet decomposition, and $(k,m) = (2^l i, 2^l j)$ in the l^{th} level wavelet decomposition, respectively. Therefore, in this paper, we classify the immediate neighborhood of $(2i,2j)$ in a single stage wavelet decomposition or in general $(2^l i, 2^l j)$ in the l^{th} level wavelet decomposition as a moving region in the current image frame, respectively. We take the union of the moving regions originating from different subband images in the current image frame. Moving region pixels are then processed by a region growing algorithm to include the pixels located at immediate neighborhood of them. This region growing algorithm checks whether the following condition is met for these pixels:

$$|J_n(i+m,\ j+m) - D_n(i+m,\ j+m)| \\ > KT_n(i+m,\ j+m) \tag{9}$$

where $m = \pm 1$, and $0.8 < K < 1, K \in \mathbf{R}^+$. If this condition is satisfied, then that particular pixel is also classified as moving. After this classification of pixels, moving objects are formed and encapsulated by their minimum bounding boxes.

4. Experimental results

The above algorithm is implemented using C++ 6.0, running on a 1500 MHz Pentium 4 processor. The PC-based system can handle 16 video channels captured at 5 frames per second in real-time. Each image fed by the channels has the frame size of PAL composite video format, which is 720 pixel × 576 pixel.

The video data is available in compressed form. For detection and time performance analysis of our moving object detection method, we use only the 3rd level low–low, low–high, high–low, and high–high coefficients. Higher resolution wavelet subimages are not decoded.

The performance of our algorithm is tested using 65 different video sequences. These sequences have different scenarios, covering both indoor and outdoor videos under various lighting conditions containing different video objects with various sizes. Some example snapshots are shown in Fig. 2. In a typical surveillance system, 16 video channels are displayed in a monitor simultaneously as shown in Fig. 3. The size of each video window is 256 × 192 in a 1024 × 768 monitor. Therefore, there is no need to reconstruct full-resolution images during regular screening. If the security guard wants to take a look at one of the video channels more carefully then one needs to decode the entire bit-stream of that particular channel and synthesize the full-resolution image using the reconstruction filter-bank of the WT. Otherwise there is no need to fully decompress 16 channels.

The moving regions are also detected by using two different background subtraction methods over 180 × 144 size images. They are the hybrid method of VSAM [4] and the method based on modeling the background using Gaussian mixture models (GMM) [10]. The low-resolution 180 × 144 images can be obtained from the 2$^{\text{nd}}$ low–low of the wavelet pyramid and the composite image shown in Fig. 3 can be populated by these images. The size of 2$^{\text{nd}}$ low–low images are close to the allocated window size of 256 × 192 in Fig. 3.

Some moving object detection results are shown in Figs. 4 and 5. In all of the sequences, the regions obtained by the methods using 180 × 144 low–low data are tighter than the ones detected by our method using subband images as expected. This is natural because there is a compromise between location and scale as we go up in the wavelet pyramid, i.e., smaller size images are used in our method. However, this is not important in a video surveillance system because smaller size images are displayed in a regular monitor during 16-channel simultaneous screening. The important issue is to detect moving objects and produce alarms for the security guard watching the surveillance system because guards may get dizzy quickly and may ignore events taking place in front of his or her eyes without an automatic motion detection-based alarm system.

Moving objects of various sizes are successfully detected by all three methods as summarized in Tables 1–3. The numbers listed in Tables 1–5 are the frame numbers of frames in which detection

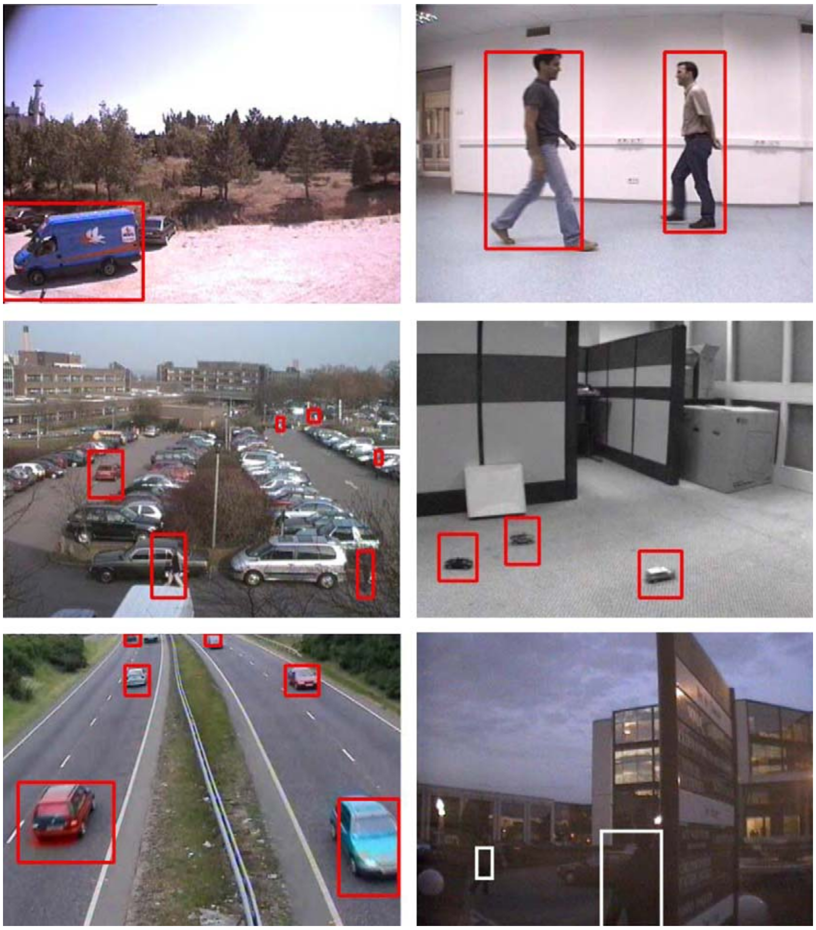

Fig. 2. Detection results of subband domain method with various object sizes and lighting conditions both indoor and outdoor.

took place. For example, MAN2 object in VIDEO-1 sequence in Table 1 is detected at the 19th frame in all three methods, namely our method utilizing the subband data only, the methods of VSAM [4] and GMM [10].

Motion detection results in videos containing objects with sizes ranging from 20 × 20 to 100 × 100 objects are presented in Table 1. Such large moving objects are detected at the same time by all three methods.

In Table 2, motion detection results of the algorithms with videos containing objects having sizes comparable to 8 pixel × 8 pixel are presented. In these videos, there is not much difference in terms of time delay between the methods. Smaller size objects are not important in a surveillance system with well-placed cameras because they may be moving tree leaves, small birds, animals, etc.

Table 3 presents a comparison of the methods with a video called "dark parking lot". In this video, the system is tested in a parking lot at night. The three methods raise alarms at around the same time instants.

Time performance analysis of the methods are also carried out. The methods of VSAM and GMM are implemented using videos with frame size of 180 × 144. This image data is extracted from the low–low image of the 2nd level WT. Performance results show that subband domain

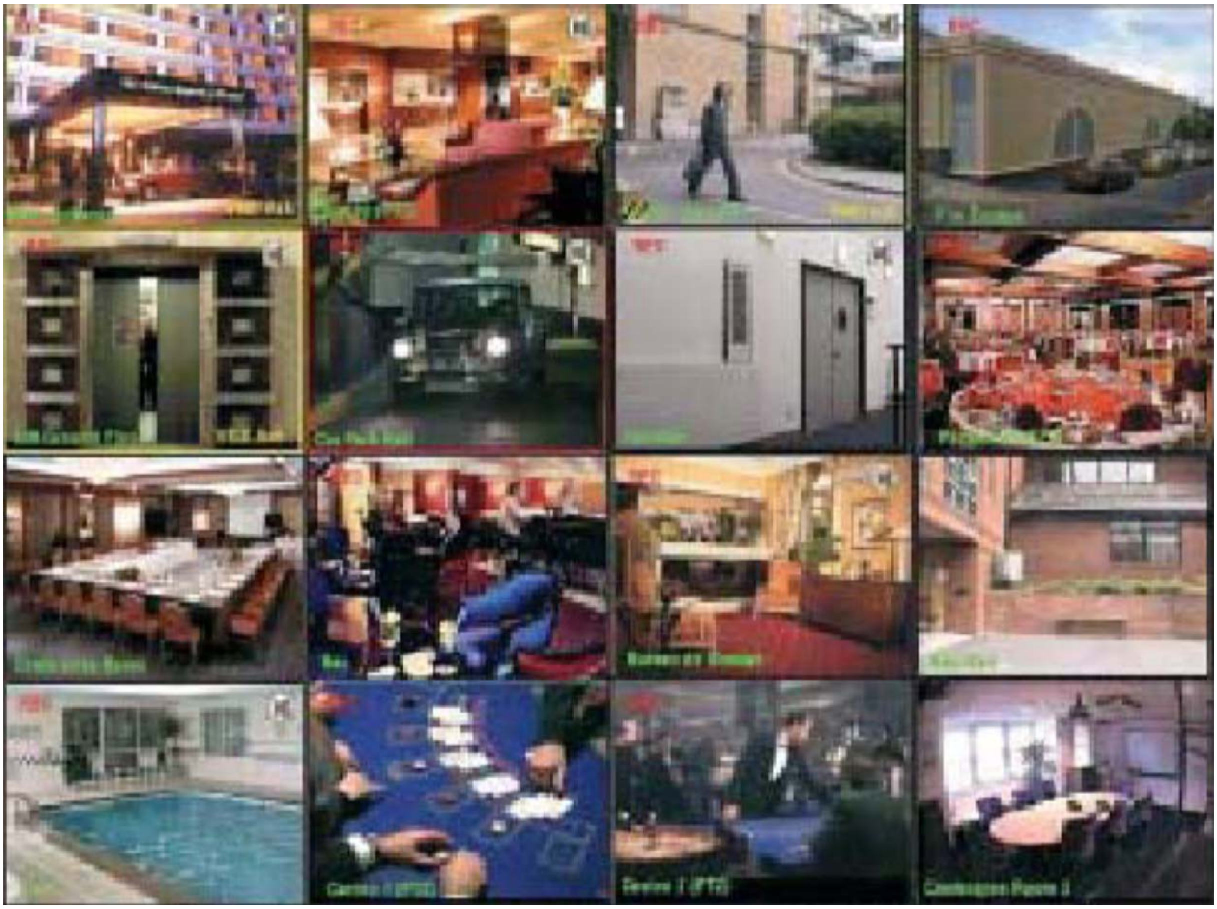

Fig. 3. A typical surveillance system monitoring 16 channels simultaneously.

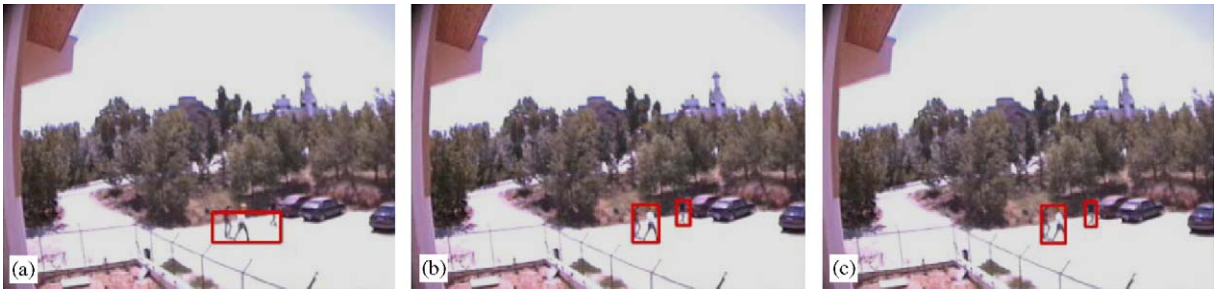

Fig. 4. Detection results for (a) subband domain, (b) VSAM and (c) GMM based methods in a parking lot sequence recorded in the day-time.

method is by far the fastest one. Our method processes an image in 1.1 ms, whereas ordinary VSAM method processes an image in 3.1 ms and the time for the GMM-based background estimation approach to process an image takes about 28 ms, on average. It is impossible to process

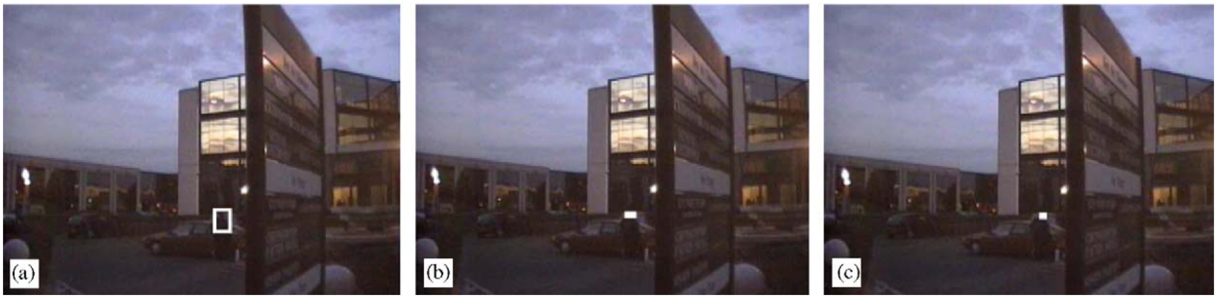

Fig. 5. Detection results for (a) subband domain, (b) VSAM and (c) GMM based methods in a parking lot sequence recorded at night.

Table 1
Comparison of motion detection methods with videos having large moving objects. All videos are captured at 10 fps

Large object videos	Objects	Subband domain method	VSAM	GMM
VIDEO-1	MAN1	15	15	16
	MAN2	19	19	19
VIDEO-2	MAN1	13	13	13
	MAN2	74	74	74
	MAN3	164	164	164
VIDEO-3	MAN1	15	15	15
	MAN2	20	20	21

Table 2
Comparison of motion detection methods with videos having small moving objects. Toy car videos and Crowded parking lot video are captured at 15 and 5 fps, respectively

Small object videos	Objects	Subband domain method	VSAM	GMM
TOY CARS-1	CAR1	40	40	40
	CAR2	65	65	65
	CAR3	75	75	76
TOY CARS-2	CAR1	70	70	71
	CAR2	78	78	78
CROWDED PARKING LOT	MAN1	12	4	6
	COUPLE1	14	11	11
	WOMAN1	16	5	6
	WOMAN2	18	17	17
	COUPLE2	29	29	29
	CAR1	34	34	35
	CAR2	94	94	95

16 video channels consisting of 180 × 144 size images simultaneously using the VSAM and GMM-based motion detection methods in a typical surveillance system implemented in a PC.

False alarms occur in all three methods due to leaves and tree branches moving in the wind, etc. In indoor surveillance applications, neither of the three methods produce false alarms. On the other hand, in outdoor surveillance applications, the GMM-based method have the least false alarm performance among the three methods studied in this paper as shown in Table 4. This is because it uses color information and it models possible background scenarios in a probabilistic framework. As a result, it is more robust against periodic motion such as motion of leaves. However, it still exhibits false alarms. The GMM-based method can be also implemented in the wavelet domain. However, even the wavelet domain implementation is computationally too costly compared to other methods.

Motion sensitivity of our subband domain method can be adjusted to detect any kind of motion in the scene, by going up or down in the wavelet pyramid and playing with the parameter b in Eq. (7). However, by going up to higher resolution levels in the pyramid, the processing time per frame of the subband domain method approaches to that of the ordinary background subtraction method of VSAM. Similarly, false alarms may be reduced by increasing b in (7) at the expense of delays in actual alarms.

Table 3
Comparison of motion detection methods in a parking lot at night. This video is captured at 3 fps

	Objects	Subband domain method	VSAM	GMM
Dark parking lot	MAN1	67	67	67
	MAN2	233	233	235
	MAN3	603	602	604

Table 4
Frame numbers of some outdoor videos at which false alarms occur when leaves of the surrounding trees move with the wind. Indoor videos yield no false alarms

Videos	Subband domain method	VSAM	GMM
OUTDOOR-1	72, 81, 86, 91	51, 61, 72, 81, 91	69, 72, 83
OUTDOOR-2	420, 440, 462, 481, 497	419, 432, 449, 463, 480, 498	422, 481, 487, 500
OUTDOOR-3	No false alarms	No false alarms	No false alarms
INDOOR-1	No false alarms	No false alarms	No false alarms
INDOOR-2	No false alarms	No false alarms	No false alarms
INDOOR-3	No false alarms	No false alarms	No false alarms

Moving object detection was also carried out using the DC values of 8 × 8 DCT blocks of the original frame of size 720 × 576. A comparison of detection performance of this method with our method is presented in Table 5. It is observed that, both object detection and time performance of the wavelet domain method is similar to the DCT domain method. Some IP cameras and video capture systems produce video using motion-JPEG or MPEG-4. In these cameras, the method described in [9] should be used. Some other IP cameras produce video in JPEG-2000 or proprietary wavelet-based compression formats, as in ADV601 chip. In these cameras and systems, the video data is available in wavelet domain and as a result our algorithm is a computationally efficient choice for moving object detection.

Table 5
Comparison of wavelet and DCT based compressed domain moving object detection methods

Videos	Objects	Wavelet-based method	DCT-based method
VIDEO-1	MAN1	15	15
	MAN2	19	19
	MAN3	121	121
VIDEO-2	CAR1	74	74
	CAR2	76	76
VIDEO-3	CAR1	115	115
	MAN1	191	190
	CAR2	165	165

Our method is designed to be used in video-based surveillance systems. In these systems, detection of moving people in the viewing range of cameras is a good measure of performance of the method. For example, the man marked as MAN1 in Fig. 6 in the CROWDED PARKING LOT sequence which is detected at 12th frame using 3rd level wavelet coefficients, is not detected at all when only 4th level wavelet coefficients are used in the subband domain method. MAN1 is pointed with an arrow in Fig. 6. Walking person MAN1 has a size of 28 × 16 pixels in a typical surveillance camera capturing 720 × 576 size images. Due to resolution loss at high-level wavelet coefficients, it is not possible to detect objects of size less than 16 × 16 pixels with the subband domain method using only 4th level wavelet

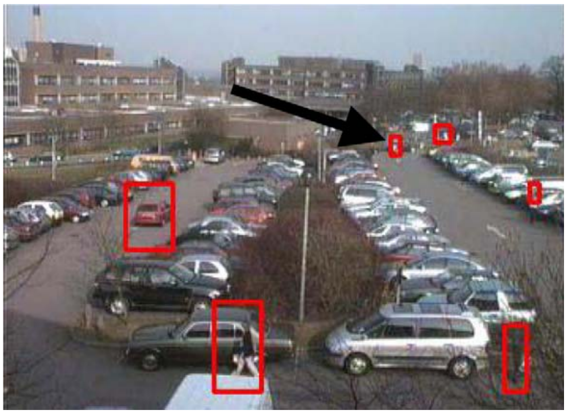

Fig. 6. Detection result of the subband domain method using all of the 3rd level wavelet coefficients. Walking man marked as MAN1 is pointed.

coefficients. Because, 4th level wavelet coefficients are obtained after four consecutive down-sampling steps, and a 16 × 16 object reduces to a 1 × 1 object. In our software implementation, we ignore isolated coefficients to eliminate noise. Therefore, the method works up to the 4th level subband decomposition.

5. Conclusion

We developed a method for detecting motion in wavelet compressed video using only subband domain data without performing inverse wavelet transform. Our results assure us that the motion detection performance of the wavelet domain method is almost the same as methods using actual pixel data for motion detection. This is an expected result because subband domain data contains all the necessary information to reconstruct the actual image.

The main advantage of the proposed method compared to regular methods is that it is not only computationally efficient but also it solves the bandwidth problem associated with video processing systems. It is impossible to feed the pixel data of 16 video channels into the PCI bus of an ordinary PC in real-time. However, compressed video data of 16 channels can be handled by an ordinary PC and its buses, hence real-time motion detection can be implemented by the proposed algorithm.

References

[1] M. Antonini, M. Barlaud, P. Mathieu, I. Daubechies, Image coding using wavelet transforms, IEEE Trans. Image Process. 1 (2) (April 1992) 205–220.

[2] Aware Inc., 40 Middlesex Turnpike, Bedford, Massachusetts, 01730, URL:www.aware.com, MotionWavelets™ real-time software video codec, 1999.

[3] M. Bagci, Y. Yardimci, A.E. Cetin, Moving object detection using adaptive subband decomposition and fractional lower order statistics in video sequences, Signal Process. (December 2002), 1941–1947.

[4] R.T. Collins, A.J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, L. Wixson, A system for video surveillance and monitoring: VSAM final report, Technical Report CMU-RI-TR-00-12, Carnegie Mellon University, May 1998.

[5] G.L. Foresti, P. Mahonen, C.S. Regazzoni, Multimedia Video-Based Surveillance Systems: Requirements, Issues and Solutions, Kluwer, Dordrecht, 2000.

[6] I. Haritaoglu, D. Harwood, L. Davis, W4: who, when, where, what: a real time system for detecting and tracking people, in: Third Face and Gesture Recognition Conference, April 1998, pp. 222–227.

[7] J.-C. Huang, W.-S. Hsieh, Wavelet-based moving object segmentation, Electron. Lett. 39 (19) (September 2003).

[8] S. Naoi, H. Egawa, M. Shiohara, Image processing apparatus, U.S. Patent 6,141,435, October 31st, 2000.

[9] I.B. Ozer, W. Wolf, A hierarchical human detection system in (un)compressed domains, IEEE Trans. Multimedia (June 2002) 283–300.

[10] C. Stauffer, W.E.L. Grimson, Adaptive background mixture models for real-time tracking, in: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 1999.

[11] Y. Taniguchi, Moving object detection apparatus and method, U.S. Patent 5,991,428, November 23rd, 1999.

[12] Visioprime Ltd., 30 St. Johns Road, St. Johns, Woking, Surrey, GU21 7SA, URL:www.visioprime.com.

[13] R. Zaibi, A.E. Cetin, Y. Yardimci, Small moving object detection in video sequences, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'00, vol. 4, 2000, pp. 2071–2074.