Multimed Tools Appl (2016) 75:10855–10885
DOI 10.1007/s11042-015-2800-5

A robust video watermarking technique for the tamper detection of surveillance systems

Farnaz Arab [1,2] · **Shahidan M. Abdullah** [1] ·
Siti Zaiton Mohd Hashim [2] · **Azizah Abdul Manaf** [1] ·
Mazdak Zamani [1]

Received: 8 October 2014 / Revised: 20 April 2015 / Accepted: 1 July 2015 /
Published online: 16 July 2015

Abstract A digital watermark embeds an imperceptible signal into data such as audio, video and images, for different purposes including authentication and tamper detection. A real-time video surveillance application requires a large quantity of sequences to be processed, which makes computational efficiency an additional constraint on video watermarking for surveillance systems. As a result, spatial domain schemes are a more efficient than frequency domain schemes. This paper focuses on video watermarking, particularly with respect to the Audio Video Interleaved (AVI) form of video file format. It proposes two new watermarking schemes which seem to offer a high degree of imperceptibility and efficient tamper detection. Both schemes were subjected to nine different types of common attack, which revealed one scheme, VW8F, to be superior, particularly in terms of imperceptibility. VW8F was then compared with a range of similar schemes by other authors. The results show that VW8F offers both improved imperceptibility (average PSNR of 47.87 dB) and proven efficiency at detecting a wider range of tampering compared to the other similar schemes.

Keywords Video watermarking · Tamper detection · Surveillance systems

✉ Farnaz Arab
arab.farnaz@gmail.com

Shahidan M. Abdullah
mshahidan@utm.my

Siti Zaiton Mohd Hashim
sitizaiton@utm.my

Azizah Abdul Manaf
azizaham.kl@utm.my

Mazdak Zamani
zamani.mazdak@gmail.com

[1] Advanced Informatics School, Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia

[2] Faculty of Computing, Universiti Teknologi Malaysia, Skudai, Malaysia

1 Introduction

A digital watermark is a kind of mark or indication embedded in a host medium such as a digital image, audio, text, software or video. Watermarking is a technique of concealing digital information in the carrier signal (host). The hidden data is not necessarily related to the content of the host [4, 14, 15]. Video watermarking is commonly applied to video files as a means of impeding unlawful manipulation or distribution of the product concerned [16, 20].

Digital Video Surveillance Systems (DVS) play a major role in forensic evidence in court [21, 26]. The video files of digital surveillance systems need to be authenticable. Thus a technique like watermarking is applied for tamper detection purposes. The watermark must not have any effect on visual information and or compromise the video evidence in any way. High imperceptible watermarks can meet these requirements [21]. Video tamper detection is a major challenge for today's researchers in the field of multimedia security [24].

Most tampered media have part of the medium altered using objects in the medium itself. Parts of a medium can be altered geometrically to change their appearance. commonly used by forgers include cropping, rotation and scaling [11]. An alteration using scaling changes the size of objects, often creating an illusion. Rotational modification, on the other hand, changes the angle of alignment of an object in the medium [7, 8].

Such modifications to an image can be detected using a combination of filters [12]. However, noise degradation can interfere with many tamper detection methods. Local noise is therefore often added to tampered digital media to try to conceal the tampering. It is common for forgers to add locally random noise to the altered parts of a medium [9]. The presence of varying levels of noise in a medium can thus be an indication of tampering.

Splicing is a common multimedia tampering technique. Nowadays there are many methods to detect spliced manipulations. Analysts usually try to identify abnormalities at the boundary of objects in the medium, which can point to the presence of splicing. Splicing is relatively easy to detect manually by expert visual analysis. Based on these premises, research on the Human Visual System (HVS) has been used to develop a procedure for splicing detection [11].

Although video watermarking has many properties, the main three ones are imperceptibility, robustness and payload or capacity. All three are closely related to each other: for example, when robustness increases, imperceptibility decreases, and vice-versa [1, 27]. The challenge is to develop watermarking applications and techniques which strike the correct balance between these conflicting requirements [1, 13].

2 Background to the problem

The rapid growth of communication networks, coupled with the ease with which digital products can be modified or copied, have made it more and more urgent to address the issue of digital tamper detection [19]. However, there has been much less progress in developing systems for video watermarking tamper detection than for digital images [1].

Because of the natural redundancy between video frames, schemes for still image tamper detection are not appropriate for digital video watermarking. Image tamper detection schemes cannot, for example, cater for attacks by means of frame dropping, frame inserting, frame shifting and so on. They also have only a limited ability to detect the areas where tampering has occurred [17].

There is therefore a clear need for a tamper detection scheme that can reliably verify video content and prevent forgery. A number of researchers have proposed digital watermarking as such a scheme [5, 18, 26]. A wide range of modifications to various aspects of digital videos could potentially be utilized to achieve this [14]. However, in order not to change the nature of the visual information in question, embedded data has to be imperceptible and robust. At the same time, the growing popularity of the surveillance market and its increasing role in environmental and personal safety have led to a sharp expansion in the number of real-time video surveillance systems. This imposes a further constraint on video watermarking for surveillance systems: computational capacity limits [10].

Existing techniques for concealing information in multimedia hosts are mostly based on the spatial rather than the frequency domain. Spatial domain watermarking involves slightly modifying the host pixel values but at a low level of complexity. In the spatial domain embedding method, information is embedded in the Least Significant Bit (LSB) of the host. To improve robustness, a small watermark can be embedded in this way several times; if a single copy of the watermark survives, the method passes the robustness test. While the spatial domain technique is easy to implement, sometimes adding noise can entirely demolish the watermark. Attackers may also be able to detect spatial domain watermarks by comparing the anticipated sample with the signal actually received [22].

In the frequency domain watermarking, first the host is converted to the frequency domain then the watermark is added and then the inverse frequency transform is applied. One common frequency domain technique is the Discrete Cosine Transform (DCT), which divides the image into low, middle and high frequency bands. In terms of imperceptibility, the middle band is the best option. Watermarks embedded in the high frequency band tend to affect the details of edges and other information. On the other hand, watermarks embedded in the low frequency band can also impact negatively on imperceptibility. All in all, the DCT is no more efficient than the spatial domain when it comes to transparency, and it also requires greater computational capacity [27].

Another common frequency domain technique is the Discrete Wavelet Transform (DWT), which breaks the image down into four sub-bands, representing a low resolution approximation (LL) and the horizontal (HL), vertical (LH) and diagonal (HH) detail of components. The edge and texture patterns are located in the high resolution sub-bands. The watermark cannot be embedded in LL because this part of the image is smoother; nor can it be embedded in HH because major details of the image would be lost. That is why the HL and LH are normally selected for watermarking [5, 20]. Overall, the DWT, like the DCT, is no more efficient than the spatial domain in terms of transparency; and it also requires if anything even more computational power than the DCT [5, 20].

In sum, the disadvantage of frequency domain methods is that they are computationally more resource-intensive than spatial methods. Spatial domain techniques are therefore, on balance, better suited for video watermarking than other watermarking domains.

3 Statement of the problem

A real-time video surveillance application requires a large number of sequences to be processed. Video watermarking for surveillance systems therefore faces the important constraint of computational efficiency and capacity. While watermarking schemes can also be applied in the frequency domain, with high imperceptibility and good robustness, the overriding disadvantage of these is that they are computationally much more expensive (resource-intensive) than spatial schemes. The latter are therefore better suited for watermarking surveillance devices.

At the same time, in order for spatial domain schemes to achieve similar levels of efficiency to frequency-based schemes, additional information needs to be embedded – requiring more payload. Such additional information may include redundant watermarks to ensure robustness as well as larger quantities of metadata of pixels to increase the efficiency of detecting attacks. All this additional information, however, may degrade quality (the imperceptibility of the watermarking).

4 Proposed schemes

This section discusses the design and implementation of two proposed watermarking schemes, both in the spatial domain. Each pixel is represented by 2 bytes in the AVI video file format. A block-wise technique is used to determine exactly which block is altered.

The simulation of each 2*2 block consists of four consecutive pixels in the video stream, as illustrated in Fig. 1. In the simulation, after reading the first two pixels there is no need to skip to the next row to read the last two pixels. In block simulations, video data is considered as coherent data, and the video frame does impose any limitation on the size of the block. One block may consist of pixels from two different frames: for example, the first three pixels could

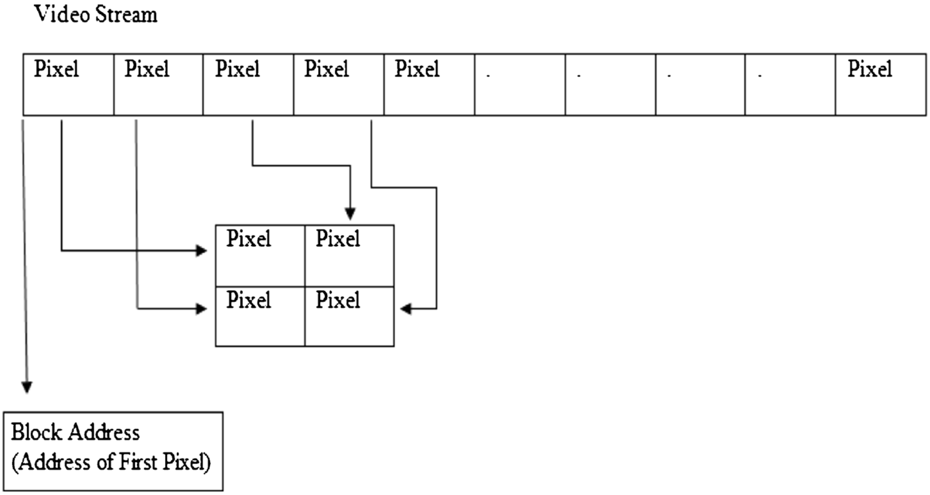


Fig. 1 Block Simulation

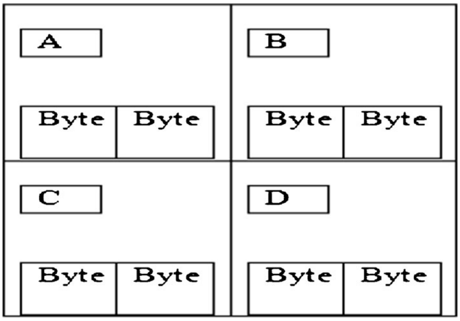


Fig. 2 Inside Each Block

come from the end of one frame, and the last pixel from the next frame. A scheme of this nature can watermark more pixels (indeed, almost all the pixels), unlike common schemes based on real 2*2 blocks, which inevitably exclude some pixels at the ends and beginnings of frames.

The first pixel's address in each block represents the address of the block. The schemes presented below divide the video stream into 2*2 non-overlapping simulated blocks. Thus, as shown in Fig. 2, there are 2 bytes in each cell of the block.

4.1 Scheme VW16E

The watermark for the first scheme (VW16E) has 16 bits. The watermark takes the place of the 4 Least Significant Bits (LSB) of each pixel. It is a combination of

A

0,7	0,6	0,5	0,4	0,3	0,2	0,1	0,0
▩	▩						

1,7	1,6	1,5	1,4	1,3	1,2	1,1	1,0
				░	░	░	░

B

2,7	2,6	2,5	2,4	2,3	2,2	2,1	2,0
		▩	▩				

3,7	3,6	3,5	3,4	3,3	3,2	3,1	3,0
				░	░	░	░

C

4,7	4,6	4,5	4,4	4,3	4,2	4,1	4,0
				▩	▩		

5,7	5,6	5,5	5,4	5,3	5,2	5,1	5,0
				░	░	░	░

D

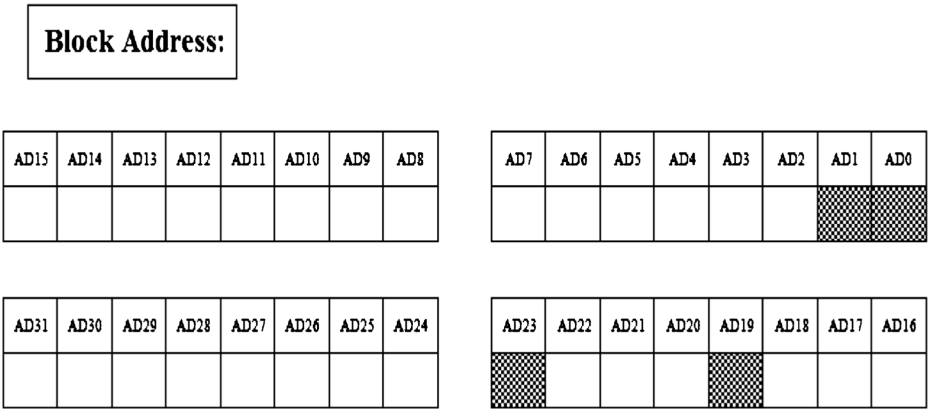
6,7	6,6	6,5	6,4	6,3	6,2	6,1	6,0
						▩	▩

7,7	7,6	7,5	7,4	7,3	7,2	7,1	7,0
				░	░	░	░

░ **4 LSB for Substitution**

▩ **Value of MSB is used in Watermark**

Fig. 3 Chosen Bits as Pixel Data for 16-Bit Watermark



Block Address:

AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8

AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
						▨	▨

AD31	AD30	AD29	AD28	AD27	AD26	AD25	AD24

AD23	AD22	AD21	AD20	AD19	AD18	AD17	AD16
▨				▨			

Fig. 4 Block Address Bytes for 16-Bit Watermark

integrity bits and confidential bits: 12 bit for integrity and 4 bits for confidential. The integrity bits consist of four bits of the block's address plus eight bits of pixel data. The 16-bit watermark is the signature for each simulated block, with a size of 2*2 pixels.

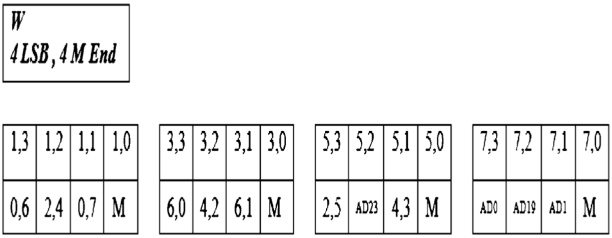
As Fig. 3 shows, all the pixels contribute to creating the 16-bit watermark. Two MSB bits in each pixel are used as integrity information, accounting for 8 of the total of 16 bits. As Fig. 4 shows, the block's address takes up a further 4 bits. This leaves the remaining 4 bits for the watermark, ie confidential bits.

The integrity and confidential bits come together to create the 16-bit watermark. As illustrated in Fig. 5, all the confidential bits are placed on the first layer of the LSB. This 16-bit watermarking scheme, in which the confidential bits are on the first layer of the LSB, is called "Video Watermarking 16", or in shortened form "VW16E".

4.2 Scheme VW8F

VW8F is another tamper detection scheme for video files, which uses 8 rather than 16 bits to embed a watermark. Thus, fewer bits and layers are modified. In fact, VW8F is an enhanced version of VW16F. The embedded 4 confidential bits in VW16F are reduced to 1 bit in VW8F. These confidential bits are therefore spread out over a

Fig. 5 VW16E Watermark



W
4 LSB, 4 M End

1,3	1,2	1,1	1,0
0,6	2,4	0,7	M

3,3	3,2	3,1	3,0
6,0	4,2	6,1	M

5,3	5,2	5,1	5,0
2,5	AD23	4,3	M

7,3	7,2	7,1	7,0
AD0	AD19	AD1	M

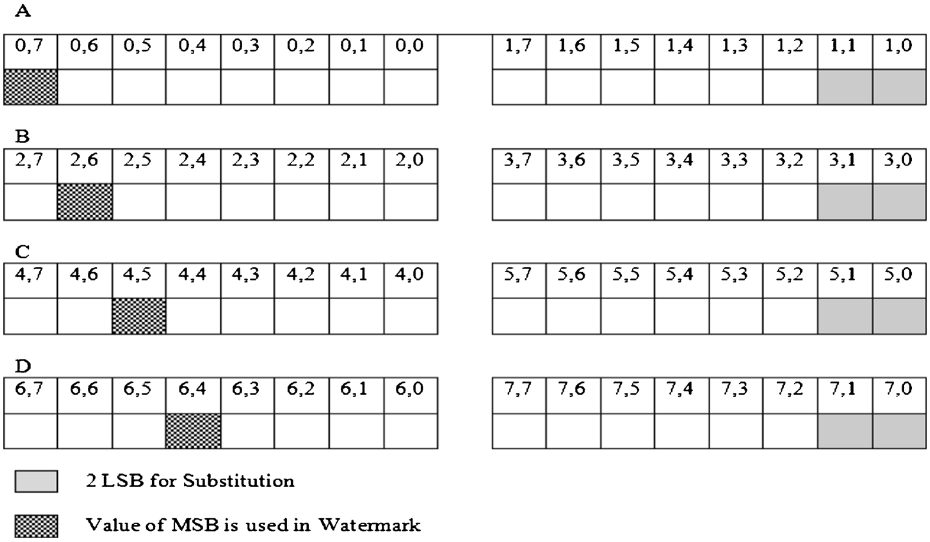


Fig. 6 Chosen Bits as Pixel Data for 8-Bit Watermark

wider area of the video steam. Meanwhile, the embedded 12 integrity bits in VW16F are reduced to 7 bits in VW8F. This 8-bit watermarking scheme, with the single confidential bit on the first layer of the LSB, is called “Video Watermarking 8 at First”, or in shorter form “VW8F”.

Fig. 6 illustrates all the pixels invovled in creating the 8-bit watermark, while Fig. 7 shows the bits from the address used in the watermark. Figure 8 shows the combination of confidential and integrity bits in VW8F.

5 Testing imperceptibility

To test the imperceptibility of the scheme, 14 samples from surveillance system videos in AVI file format were used. The parameters of each experimental video

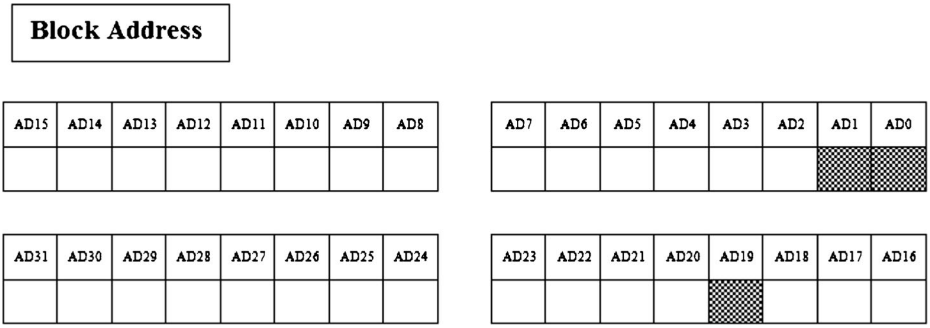

Block Address
AD15 AD14 AD13 AD12 AD11 AD10 AD9 AD8
AD7 AD6 AD5 AD4 AD3 AD2 AD1 AD0
AD31 AD30 AD29 AD28 AD27 AD26 AD25 AD24
AD23 AD22 AD21 AD20 AD19 AD18 AD17 AD16

Fig. 7 Block Address Bytes for 8 Bits Watermark

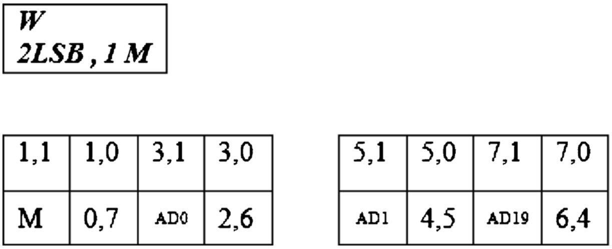


Fig. 8 VW8F Watermark

sequence are shown in Table 1. They were chosen based on different lengths and different numbers of total frames. For the confidential bits, four different sizes of text file were chosen. The parameters of the message samples chosen are shown in Table 2.

After embedding all four messages in all of the video samples using each of the two different watermarking schemes, the Peak Signal-to-Noise Ratio (PSNR) was calculated. The two watermarking schemes, VW16E and VW8F, were then compared based on their respective PSNR results for the same video samples. The results of these comparisons are shown in the sections below.

6 Test results of proposed schemes

6.1 Comparison of VW16E and VW8F for video sample 7

Table 3 and Fig. 9 show the PSNR results of embedding four messages into video sample No 7 with the VW16E and VW8F schemes respectively.

Table 1 Specification of video samples

Video Sample No	Size (Bytes)	Length (Second)	Frame Width (Pixel)	Frame Height (Pixel)	Frame Rate (Frame / Second)	Total Frames	Total Pixels	Integrity Bits (VW16E)	Integrity Bits VW8F
1	16,080,844	0.00.03	320	180	29	93	5,356,800	16,070,400	9,374,400
2	29,272,060	0.00.04	320	240	29	127	9,753,600	29,260,800	17,068,800
3	39,239,260	0.00.07	320	180	29	227	13,075,200	39,225,600	22,881,600
4	57,212,956	0.00.11	320	180	29	331	19,065,600	57,196,800	33,364,800
5	68,964,988	0.00.13	320	180	29	399	22,982,400	68,947,200	40,219,200
6	78,813,220	0.00.11	320	240	29	342	26,265,600	78,796,800	45,964,800
7	125,128,444	0.00.18	320	240	29	543	41,702,400	125,107,200	72,979,200
8	129,042,292	0.00.14	320	320	29	420	43,008,000	129,024,000	75,264,000
9	243,690,052	0.00.47	320	180	29	1410	81,216,000	243,648,000	142,128,000
10	245,936,764	0.00.47	320	180	29	1423	81,964,800	245,894,400	143,438,400
11	259,417,036	0.00.50	320	180	29	1501	86,457,600	259,372,800	151,300,800
12	260,626,804	0.00.50	320	180	29	1508	86,860,800	260,582,400	152,006,400
13	296,064,676	0.00.53	320	192	29	1606	98,672,640	296,017,920	172,677,120
14	391,159,132	0.00.51	320	262	29	1555	130,371,200	391,113,600	228,149,600

Table 2 Specification of message samples

Name	File Type	Size (Bytes)	Confidential Bits
M1	TXT	1669	13,352
M2	TXT	30,044	240,352
M3	TXT	353,828	2,830,624
M4	TXT	1,972,764	15,782,112

6.2 Comparison of VW16E and VW8F for video sample 14

Table 4 and Fig. 10 show the PSNR results of embedding four messages into video sample No 14 with the VW16E and VW8F schemes respectively.

7 Efficiency evaluation

In this section, we subject the embedded video streams to a variety of attacks: Frame Insert, Frame Exchange, Frame Deletion, Crop, Rotate, Reverse Rotate, Frame Shifting, Salt & Pepper and Superimpose. We then analyze whether and how these types of tampering can be detected.

7.1 Attacks on video sample no 1, watermarked by VW16E

The second message sample (M2), with the VW16E watermark, was embedded in video sample No.1. The nine types of attack mentioned above were then applied to the embedded video stream, which was subsequently analyzed to see whether the tampering could be detected or not. Table 5 shows the tamper detection results. These are highly robust, with even a single tampered key being detected.

7.1.1 Crop attack

Forty (40) Pixels from the top, bottom, left and right of all the frames in video sample No 1 were cropped. The results show that this modification was detected, with the address 6081B identified as the first block tampered with. Figure 11a is the original frame; Fig. 11b shows the

Table 3 Comparison of VW16E and VW8F for video sample 7

Host	Host Size (Byte)	Confidential Message	Confidential Bit	Integrity Bits for VW16E	Integrity Bits for VW8F	PSNR (VW16E)	PSNR (VW8F)
Video Sample 7	125,128,444	M1	13,352	125,107,200	72,979,200	34.917	47.896
Video Sample 7	125,128,444	M2	240,352	125,107,200	72,979,200	34.917	47.881
Video Sample 7	125,128,444	M3	2,830,624	125,107,200	72,979,200	34.912	47.716
Video Sample 7	125,128,444	M4	15,782,112	125,107,200	72,979,200	34.886	M4 too big

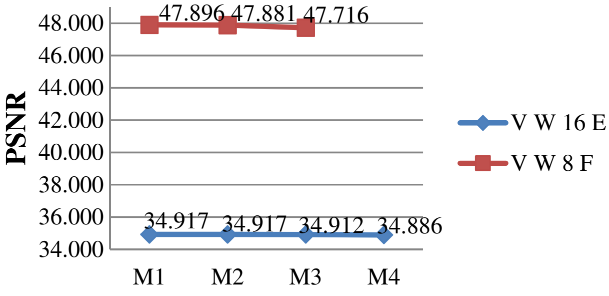


Fig. 9 Comparison of VW16E and VW8F for Video Sample 7

embedded frame; Fig. 11c is the tampered frame; and Fig. 11d shows the results of tamper detection for the same frame.

7.1.2 Frame deletion attack

Frame number 70 was deleted in sample No 1. The results show that this modification was detected, with address B6113B identified as the first block tampered with. These results are shown in Fig. 12. The Normalized Correlation (NC) for the extracted watermark was 99.98 %.

7.1.3 Frame exchange attack

Frame numbers 20 and 93 were swopped in sample No 1. The results show that this modification was detected, with address 3239AB as the first tampered block. Figure 13 shows the result of the tamper detection for the 20th frame. The NC of the extracted watermark was 99.98 %.

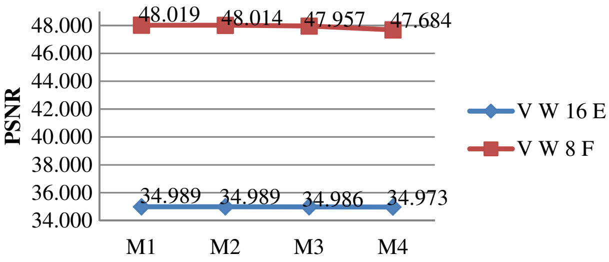
7.1.4 Frame insert attack

Frame number 93 was duplicated in the host. Again, the results show that this modification was detected, with address 4C97FB being the first block tampered with. Figure 14 shows the result of the tamper detection for the 30th frame, where the duplicate frame 93 was inserted. The NC of the extracted watermark was 99.98 %.

Table 4 Comparison of VW16E and VW8F for video sample 14

Host	Host Size (Byte)	Confidential Message	Confidential Bit	Integrity Bits for VW16E	Integrity Bits for VW8F	PSNR (VW16E)	PSNR (VW8F)
Video Sample 14	391,159,132	M1	13,352	391,113,600	228,149,600	34.989	48.019
Video Sample 14	391,159,132	M2	240,352	391,113,600	228,149,600	34.989	48.014
Video Sample 14	391,159,132	M3	2,830,624	391,113,600	228,149,600	34.986	47.957
Video Sample 14	391,159,132	M4	15,782,112	391,113,600	228,149,600	34.973	47.684

Fig. 10 Comparison of VW16E and VW8F for Video Sample 14



7.1.5 Rotate attack

All the frames of video sample No 1 were rotated by 10°. Once again, this modification was detected, with address 2013 as the first block tampered with. Figure 15a is the original frame; Fig. 15b shows the embedded frame; Fig. 15c contains the tampered frame; and finally Fig. 15d depicts the results of the tamper detection for the same frame.

7.1.6 Reverse rotate attack

All the frames of sample No 1 were rotated by 10° and then reversed. The results show that, in this case, the modifications were not detected. Figure 16a is the original frame; Fig. 16b shows the embedded frame; Fig. 16c shows the tampered frame; and finally Fig. 16d illustrates the results of tamper detection for the same frame.

7.1.7 Salt and pepper attack

Salt and Pepper attack was applied to the 10th, 11th, and 12th frames of video sample No 1. The results show that these modifications were detected, with address 17F6BB

Table 5 Attack sample results on video no 1

Attack	Address of first tampered pixel (Hexadecimal)	Tamper Detect	Watermark Extract	Normalized Correlation NC (%)	Key Availability	Effects
Crop	6081B	Yes	No	NA	NA	40 Pix from left and right
Frame Deletion	B6113B	Yes	Yes	99.99 %	Yes	frame 70th has deleted
Frame Exchange	3239AB	Yes	Yes	99.99 %	Yes	20 and 93 exchange
Frame Insert	4C97FB	Yes	Yes	99.99 %	Yes	93 duplicated and insert before 30–50
Reverse Rotate	2013	Yes	No	NA	NA	10°
Rotate	2013	Yes	No	NA	NA	10°
Salt & Pepper	17F6BB	Yes	Yes	99.99 %	Yes	10-11-12 applied
Frame Shifting	55CFB	Yes	Yes	80.85 %	Yes	1 shift left
Superimpose	1828B	Yes	Yes	95.76 %	Yes	1-2-3-4-5

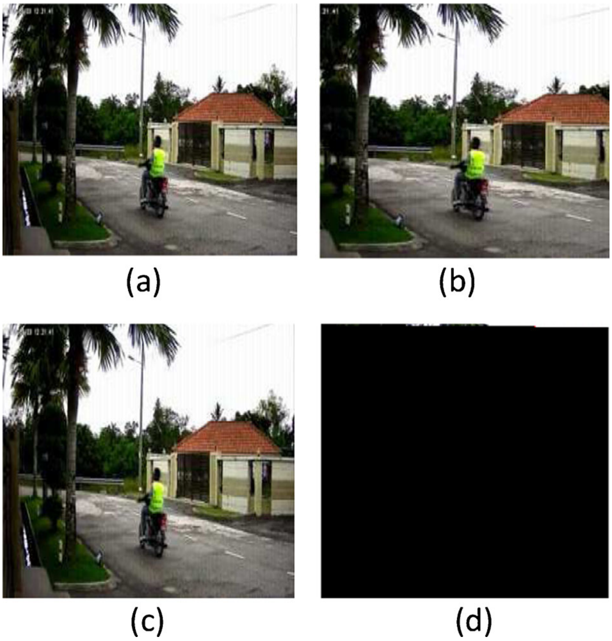


Fig. 11 **a** Original Frame for Crop Attack; **b** Tampered Frame for Crop Attack; **c** Watermarked Frame for Crop Attack; **d** Result of Tamper Detection for Crop Attack

as the first block tampered with. Figure 17a contains the original frame; Fig. 17b the embedded frame; Fig. 17c the tampered frame; and finally Fig. 17d the results of the tamper detection for the same frame. The NC of the extracted watermark was 99.98 %.

7.1.8 Frame shifting attack

One frame was shifted to the left in sample No 1. This modification was detected, with address 55CFB identified as the first block tampered with. The results of this

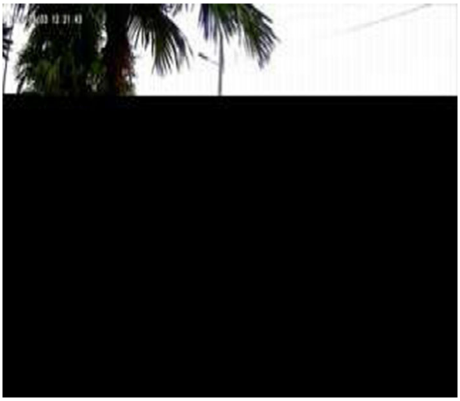


Fig. 12 Result of Tamper Detection for Frame Deletion Attack

Fig. 13 Result of Tamper Detection for Frame Exchange Attack

tamper detection are shown in Fig. 18. The NC of the extracted watermark was 80.85 %.

7.1.9 Superimpose attack

A superimpose attack was applied to frame numbers 1, 2, 3, 4 and 5 in sample No 1, with a total size of 16,080,844 Bytes and a message size of 30,044 Bytes. This modification was, again, detected and address 1828B was identified as the first tampered block. Figure 19a shows the original frame; Fig. 19b the embedded frame; Fig. 19c the tampered frame; and finally Fig. 19d the results of the tamper detection for the same frame. The NC of the extracted watermark was 95.76 %.

7.2 Attacks on video sample no 13, watermarked by VW16E

The fourth message (M4) with a size of 1,972,764 Bytes, with the VW8F scheme, was embedded in video sample No.13 with a size of 296,064,676 Bytes. The same nine attacks as previously (Frame Insert, Frame Exchange, Frame Deletion, Crop, Rotate, Reverse

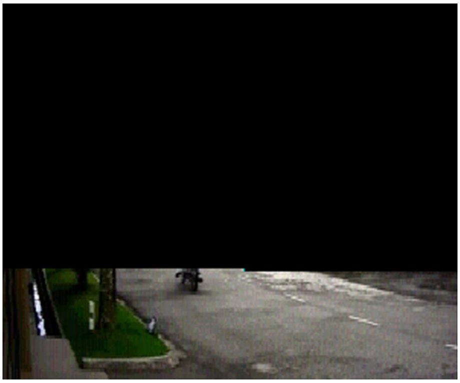


Fig. 14 Result of Tamper Detection for Frame Insert Attack

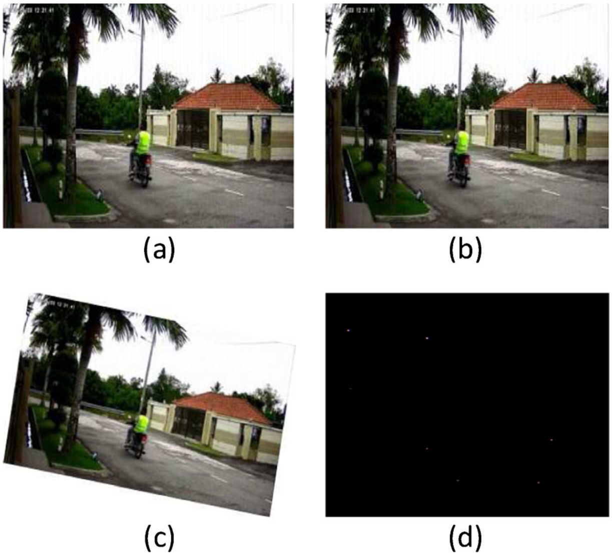


Fig. 15 **a** Original Frame for Rotate Attack; **b** Watermarked Frame for Rotate Attack; **c** Tampered Frame for Rotate Attack; **d** Result of Tamper Detection for Rotate Attack

Rotate, Shift, Salt and Pepper and Superimpose) were applied to the embedded video stream, which was subsequently analyzed to see whether the tampering could be detected or not. Table 6 shows the tamper detection results of these attacks. In those cases where it was possible to extract the watermark, the NC for the extracted watermark was calculated. Table 6 confirms that the tamper detection results were robust, with even a single key being detected.

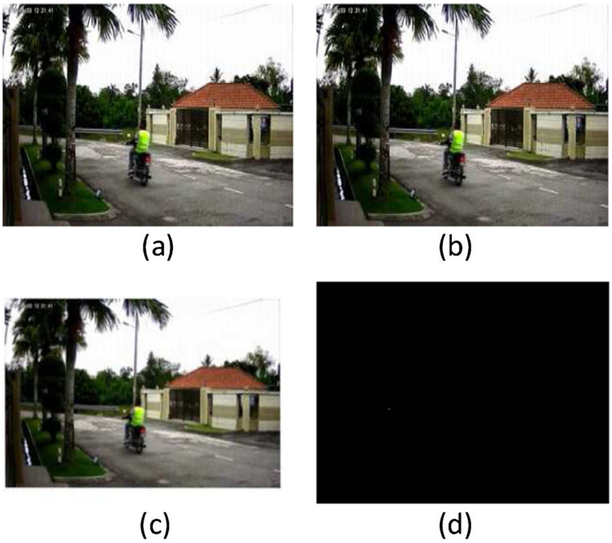


Fig. 16 **a** Original Frame for Reverse Rotate Attack; **b** Watermarked Frame for Reverse Rotate Attack; **c** Tampered Frame for Reverse Rotate Attack; **d** Result of Tamper Detection for Reverse Rotate Attack

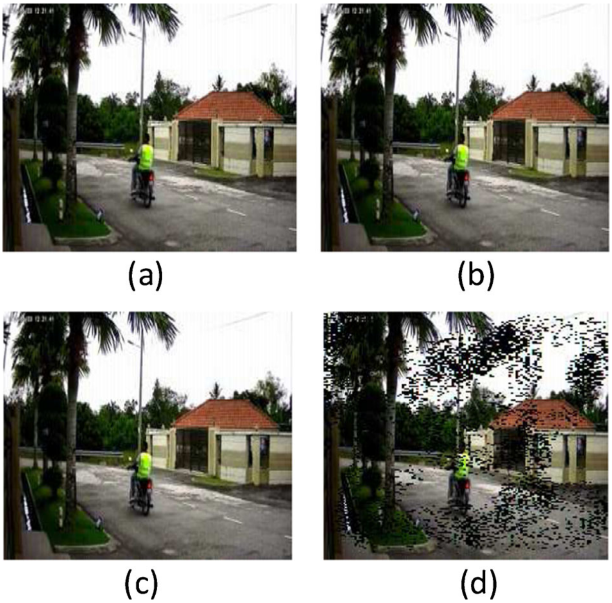


Fig. 17 **a** Original Frame for Salt and Pepper Attack; **b** Watermarked Frame for Salt and Pepper Attack; **c** Tampered Frame for Salt and Pepper Attack; **d** Result of Tamper Detection for Salt and Pepper Attack

7.2.1 Crop attack

All the frames of the thirteenth video host were cropped by 10 pixels from the left and 10 pixels from the bottom. The results show that these modifications were detected, with address 2013 as the first block tampered with. Figure 20a shows the original frame; Fig. 20b the embedded frame; Fig. 20c the tampered frame; and Fig. 20d the results of the tamper detection for the same frame.

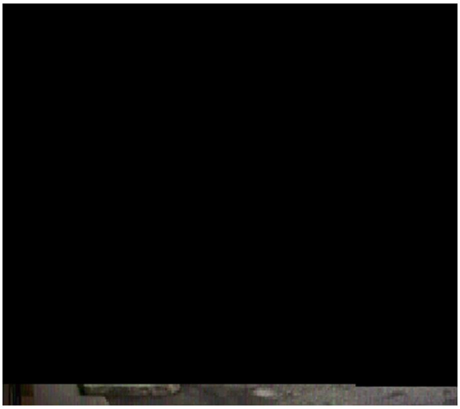


Fig. 18 Result of Tamper Detection for Frame Shift Attack

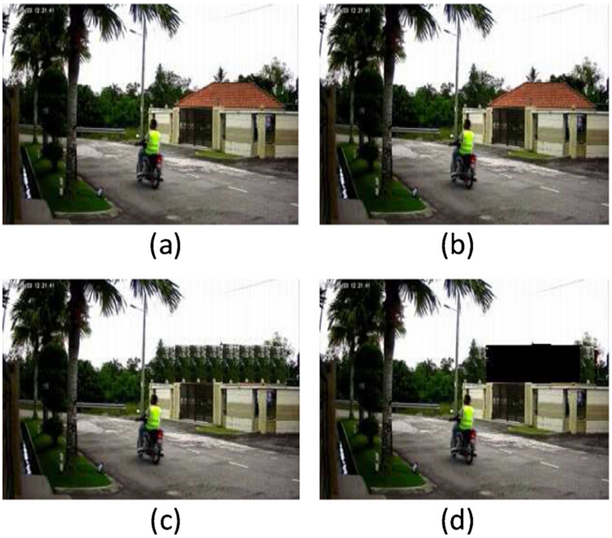


Fig. 19 **a** Original Frame for Superimpose Attack; **b** Watermarked Frame for Superimpose Attack; **c** Tampered Frame for Superimpose Attack; **d** Result of Tamper Detection for Superimpose Attack

7.2.2 Frame deletion attack

One hundred (100) frames were deleted (from 250 to 299 and 1000 to 1049) from the video host. The results show that these modifications were detected, with address 2C00003 as the first block to be tampered with. The extracted watermark was visible, with an NC of 90.50 %. Figure 21 shows the 252nd frame of the tamper detection result.

Table 6 Attack results on video sample no 13

Attack	Address of first tampered pixel (Hexadecimal)	Tamper Detect	Watermark Extract	NC (%)	Key Availability	Effects
Crop	2013	Yes	No	NA	NA	10 pix left and bottom
Frame Deletion	2C00003	Yes	Yes	90.50 %	Yes	250–299, 1000–1049 (100 frame)
Frame Exchange	E6C2A3	Yes	Yes	99.90 %	Yes	83 –>84, 212 –>312, 644–>120, 900–> 1001, 1300–>1600
Frame Insert	3590AB	Yes	Yes	88.85 %	Yes	404 duplicated and inserted before 20-78- 563- 564- 1560
Reverse Rotate	2013	Yes	No	NA	NA	−2°
Rotate	2013	Yes	No	NA	NA	−2°
Salt & Pepper	49AD02B	Yes	Yes	99.95 %	Yes	420–429 (10 frame salt)
Frame Shifting	2013	Yes	Yes	88.34 %	Yes	5 frame shift right
Superimpose	25D858B	Yes	Yes	99.92 %	Yes	216–225, 1500–1509 (20 frame)

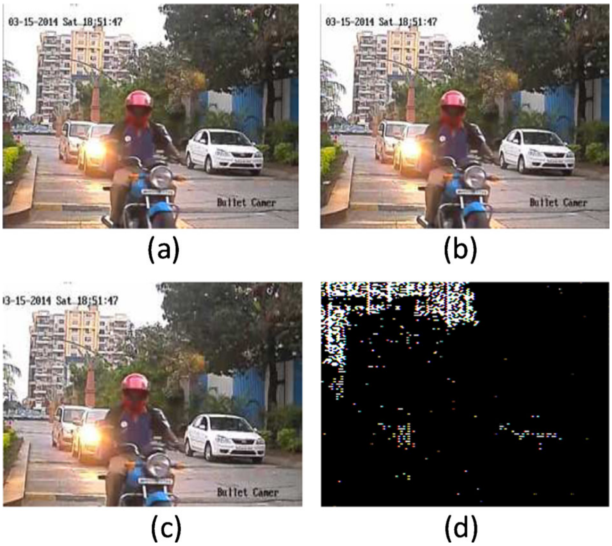


Fig. 20 **a** Original Frame for Crop Attack;**b** Watermarked Frame for Crop Attack; **c** Tampered Frame for Crop Attack; **d** Result of Tamper Detection for Crop Attack

7.2.3 Frame exchange attack

Five frames were exchanged in the video host: frame number 83 with 84, 212 with 312, 120 with 644, 900 with 1001, and 1300 with 1600. The results show that these modifications were detected, with address E6C2A3 as the first block tampered with. The extracted watermark was visible, with an NC of 99.90 %. Figure 22 shows the 83rd frame of the tamper detection results.

7.2.4 Frame insert attack

Frame number 404 was duplicated before frames 20, 78, 563, 564 and 1560 in the video host. The results show that these modifications were detected, with address 3590AB the first block

Fig. 21 Result of Tamper Detection for Frame Deletion Attack

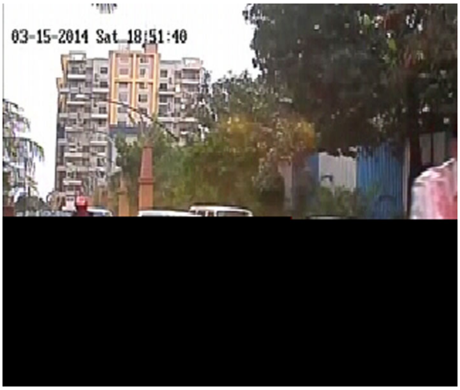


Fig. 22 Result of Tamper Detection for Frame Exchange Attack

to be tampered with. The extracted watermark was visible, with an NC of 88.85 %. The results of tamper detection for the 20th frame are shown in Fig. 23.

7.2.5 Rotate attack

All the frames of the video host were rotated by −2°. The results show that this modification was detected, with address 2013 as the first block which was tampered with. Figure 24a is the original frame; Fig. 24b shows the embedded frame; Fig. 24c contains the tampered frame; and Fig. 24d displays the results of the tamper detection for the same frame.

7.2.6 Reverse rotate attack

All the frames of the video host were rotated by −2° and then reversed. The results show that this modification was detected, with address 2013 as the first block tampered with. Figure 25a is the original frame; Fig. 25b the embedded frame; Fig. 25c the tampered frame; and finally Fig. 25d displays the results of the tamper detection for the same frame.

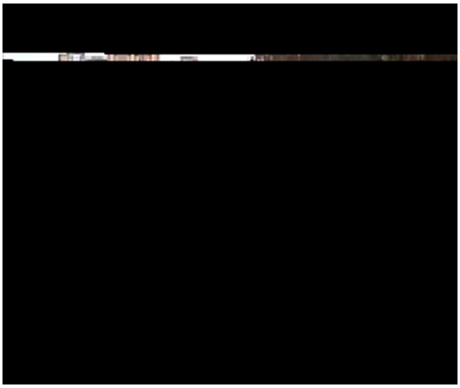
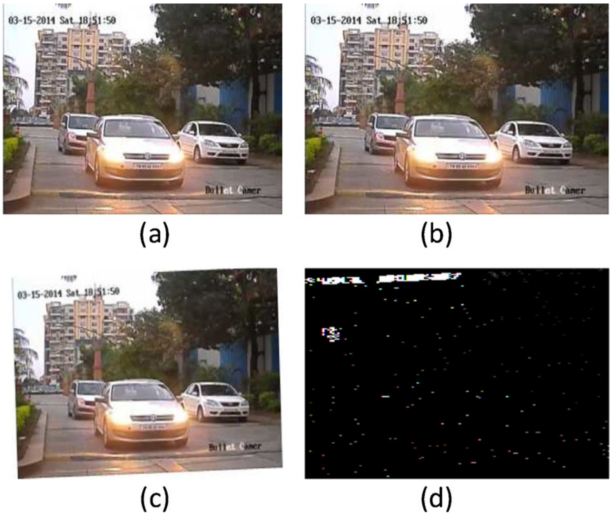


Fig. 23 Result of Tamper Detection for Frame Insert Attack

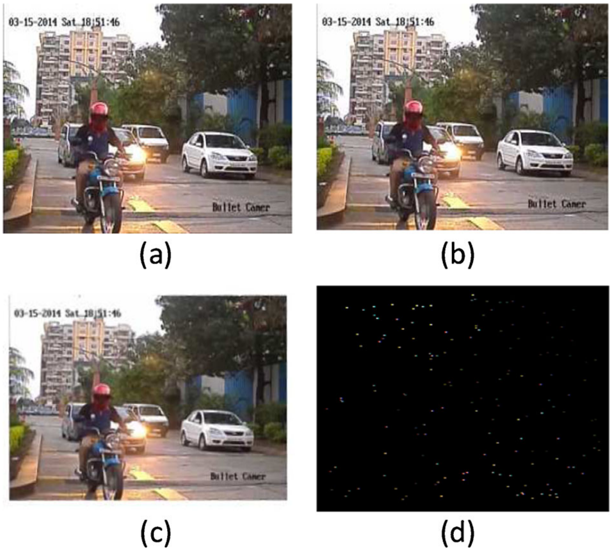
Fig. 24 **a** Original Frame for Rotate Attack; **b** Watermarked Frame for Rotate Attack; **c** Tampered Frame for Rotate Attack; **d** Result of Tamper Detection for Rotate Attack



7.2.7 Salt and pepper attack

Ten frames (frames from 420 to 429) in the video host were subjected to a salt and pepper attack. The results show that these modifications were detected, with address 49AD02B as the first block to be tampered with. The extracted watermark was visible, with an NC of 99.95 %. Figure 26a is the original frame; Fig. 26b shows the embedded frame; Fig. 26c the tampered frame; and Fig. 26d the results of the tamper detection for the same frame.

Fig. 25 **a** Original Frame for Reverse Rotate Attack, **b** Watermarked Frame for Reverse Rotate Attack, **c** Tampered Frame for Reverse Rotate Attack, **d** Result of Tamper Detection for Reverse Rotate Attack



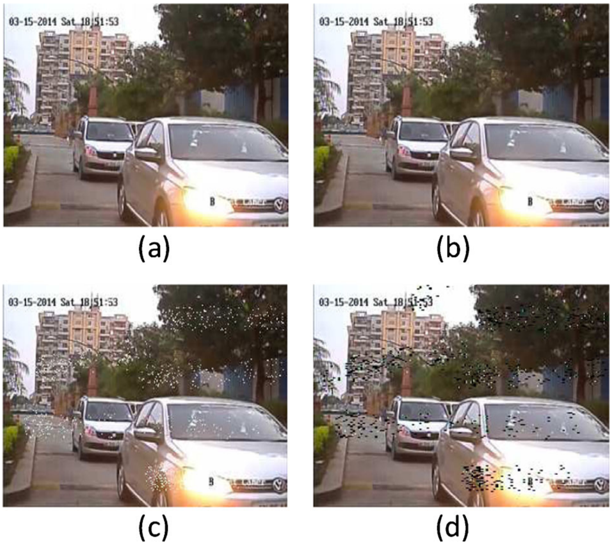


Fig. 26 **a** Original Frame for Salt and Pepper Attack; **b** Watermarked Frame for Salt and Pepper Attack; **c** Tampered Frame for Salt and Pepper Attack; **d** Result of Tamper Detection for Salt and Pepper Attack

7.2.8 Shift frame attack

Five frames were shifted to the right in the video host. This modification was detected, with address 2013 as the first block which was tampered with. The results of the tamper detection for the 90th frame are shown in Fig. 27. The NC of the extracted watermark was 88.34 %.

7.2.9 Superimpose attack

Twenty (20) frames in the video host (from 216 to 225 and 1500 to 1509) were altered by a superimpose. These modifications were detected, with address 25D858B as the first

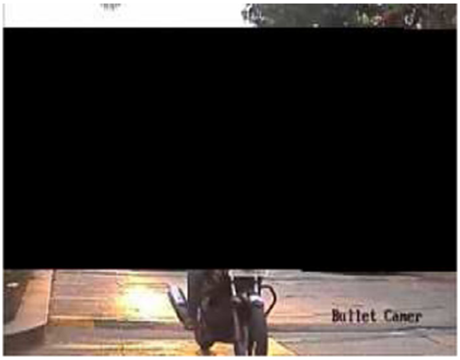


Fig. 27 Result of Tamper Detection for Shift Attack

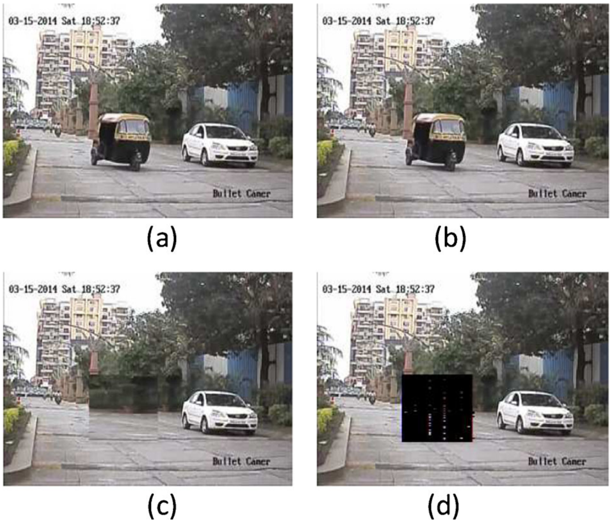


Fig. 28 **a** Original Frame for Superimpose Attack; **b** Watermarked Frame for Superimpose Attack; **c** Tampered Frame for Superimpose Attack; **d** Result of Tamper Detection for Superimpose Attack

block tampered with. The extracted watermark was visible, with an NC of 99.92 %. Figure 28a shows the original frame; Fig. 28b the embedded frame; Fig. 28c the tampered frame; and finally Fig. 28d displays the results of the tamper detection for the same frame.

8 Discussion and comparison of proposed schemes

To be effective, a watermarking scheme must not cause perceptible distortion but at the same time must make it possible to identify where tampering to a host has taken place. With this in mind, two novel spatial schemes were developed and tested with a view to improving the imperceptibility and efficiency of tamper detection in surveillance systems. The two schemes proved to be equally efficient in detecting tampering and in their overall robustness, but not in terms of their imperceptibility.

The first scheme to be tested, VW16E, was able to embed confidential and integrity information into hosts effectively, and also improved the efficiency of detecting tampering. However, it did not produce any significant improvement in the PSNR, which remained at an average of 34.84 dB.

The second scheme, VW8F, involved embedding only half the amount of information into hosts compared to the first scheme. It nevertheless maintained roughly the same level (efficiency) of tamper detection, and at the same time produced an improved average PSNR of 47.82. In sum, VW8F achieved considerable improvements in both imperceptibility and the efficiency of tamper detection over similar video watermarking schemes.

Table 7 Comparison the proposed scheme (VW8F) with other schemes

Name	Average PSNR (dB)	Tested on different samples	Crop	Delete Frame	Ex-change Frame	Insert Frame	Rotate	Reverse Rotate	Salt & Pepper	Shift Frame	Super-impose	Other Attacks	Total Attacks
[23]	40.86	3	_	_	_	_	_	_	_	_	√	_	1
[25]	42.95	1	_	_	_	_	_	_	_	_	√	_	1
[2]	42.24	2	_	_	_	_	_	_	_	_	√	1	2
[3]	45.85	1	_	_	_	_	_	_	_	_	_	2	2
[6]	45.64	5	√	_	_	_	√	_	_	_	_	6	8
[5]	44.21	1	√	_	_	_	√	_	√	_	_	2	5
Proposed Scheme	47.82	14	√	√	√	√	√	√	√	√	√	_	9

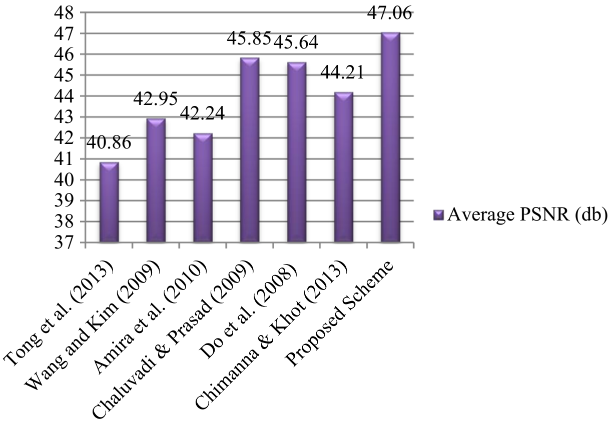


Fig. 29 PSNR Comparison of Other Schemes with Proposed Scheme

9 Comparison with other schemes

Table 7 shows a comparison between one of the proposed schemes (VW8F) and other tamper detection schemes. In such schemes the size of the medium and the payload are not important issues because the integrity bits are distributed throughout the entire medium. Rather, the level of imperceptibility and the efficiency of tamper detection are the most important issues. In most of the studies on similar schemes, the authors do not mention the size of their samples or the payload of their watermarking. Figure 29 shows that our proposed scheme offers a better PSNR than other similar schemes; while Fig. 30 shows that it is more efficient at tamper detection—considerably more so compared to most of the other schemes.

9.1 Compared with Tong et al. [23]

Tong et al's [23] scheme is based on a chaotic map. The entire image is divided into 2*2 non-overlapping blocks and 3LSB (3*4=12 bits watermark) are reserved for the

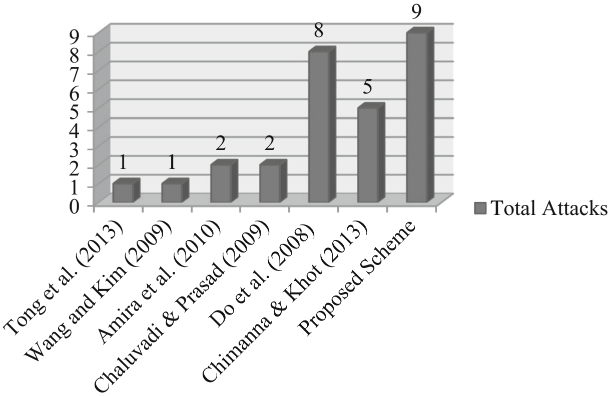


Fig. 30 Tamper Detection Efficiency of Other Schemes and Proposed Scheme

Table 8 Tong et al.'s scheme performance

Image	[23] Scheme	Scheme 1	Scheme 2	Scheme 3
Lena	40.73	36.67	38.55	39.33
Baboon	40.71	36.69	38.57	39.36
Boat	40.58	36.72	38.59	36.35
Road	40.67	36.70	38.61	39.31
Couple	40.79	36.66	35.59	39.38
Airplane	40.86	36.68	38.55	39.37

watermark. Three attacks—add objects, wrap objects and delete objects—all of them superimpose attacks, were tested to evaluate the performance of the scheme. The results show a PSNR of 40.86 dB.

Table 8 above summarises Tong et al.'s Scheme Performance and PSNRs compared to three other similar schemes. Six identical images were tested in each case. Tong et al.'s Scheme produced a highest PSNR of 40.86 dB and an average of 40.72 dB; whereas the average PSNR of the three other similar schemes was 36.68, 38.07 and 39.35 dB respectively. In other words, Tong et al.'s Scheme improved the PSNR over these comparators. However, our proposed scheme (tested under nine different types of attack) achieved a PSNR of 47.82 dB—16.93 % better than Tong et al.

9.1.1 Compared with Wang and Kim [25]

Wang and Kim [25] propose a novel scheme, using the colour channel (RGB)—and hence only usable with colour images. The image is divided into 2*2 non-overlapping blocks, and 3LSB are allocated for watermarking, including the authentication and tamper detection codes. A Lena 24-bit colour scale image was used to evaluate performance, and a superimpose attack with different percentages across the various orientations of the image was applied for tamper detection. The watermarked Lena image produced a PSNR value of 42.95 dB.

Table 9 Wang and Kim's scheme performance

Attack Position		PSNR (dB)
5 % tampered	Center	44.362
	Left	49.639
10 % tampered	Center	41.348
	Left	45.781
20 % tampered	Center	36.625
	Left	39.263
30 % tampered	Center	33.922
	Left	34.784

Table 10 Amira's scheme performance

Image	PSNR (dB)
Sofa	42.24
Bears	41.11

Table 9 above illustrates the performance of Wang and Kim's Scheme, based on attacks at different percentages in different locations across the image. The average PSNR is 40.75 dB. Again, these results show that our proposed scheme, with its average PSNR of 47.82 dB, is substantially (11.36 %) better in terms of imperceptibility, as well as being more efficient at detecting tampering.

9.1.2 Compared with Amira et al. [2]

Amira et al. [2] propose a novel fragile watermark method for tamper detection in gray level images, using a chaotic map. They used two 8-bit gray-level images to test their scheme. After embedding the watermark in two images, the PSNRs were 42.24 dB and 41.11 respectively, as shown in Table 10 below, an average of 41.67 dB. To evaluate tamper detection performance, two types of attack were applied: a collage attack (similar to a superimpose attack) and a vector quantization attack.

Once again, our proposed scheme emerges as clearly better in terms of both imperceptibility (a higher PSNR of 47.82 dB) and efficiency in detecting a wider range of tampering.

9.1.3 Compared with Chaluvadi and Prasad [3]

Chaluvadi and Prasad [3] propose an image tamper detection scheme using dual watermarks. They use 3LSB for watermarking, with the image divided into 2*2 non overlapping pixels. They also apply a Smoothing Function. They subjected their scheme to two kinds of bit injection attacks. Their imperceptibility results, shown in Table 11 below, were PSNRs of 45.85 dB and 42.66 dB for the two attacks, an average of 44.25 dB.

Our proposed scheme's imperceptibility result of 47.82 dB is slightly (4.29 %) better than Chaluvadi and Prasad's [3]. In addition, our proposed scheme has the proven ability to detect nine types of attack—four of them (frame insertion, frame deletion, frame exchanging and frame shifting) exclusively for video and the other five (salt & pepper, crop, superimpose, rotate and reverse rotate) common to both still images and video—compared to only two tested by Chaluvadi and Prasad [3].

Table 11 Chaluvadi and Prasad's scheme performance

	Attack1	Attack2
PSNR (dB)	45.85	42.66

Table 12 Do et al.'s scheme performance

Video Sequence	Average PSNR (dB)	Minimum PSNR (dB)
Seq.1	44.67	42.29
Seq.2	46.39	44.58
Seq.3	46.06	41.77
Seq.4	43.84	41.34
Seq.5	47.27	42.85

9.1.4 Compared with Do et al. [6]

Do et al.'s [6] blind watermarking scheme is based on temporal modulation of the frames and histogram, dividing the entire frames into two areas using a histogram-based watermark pattern (HWP). Five video sequences were tested by subjecting them to eight attacks: four geometric, three video processing and a camcorder recording. The overall average PSNR was 45.64 dB—slightly (4.77 %) lower than our proposed scheme's PSNR of 47.82 dB. Additionally, our proposed scheme is able to detect against nine different attacks compared to eight in the case of Do et al.'s [6] scheme (Table 12).

Once again, our scheme emerges as better in terms of both imperceptibility and its ability to detect a wider range of attacks.

9.1.5 Compared with Chimanna and Khot [5]

Chimanna and Khot's [5] proposed scheme is based on Discrete Wavelet Transform (DWT). The watermark is an image. Video frames are broken down into images, and two levels of DWT and Principal Component Analysis (PCA) are applied to them. As Table 13 below, shows, their scheme was tested and proved efficient against five types of attack. The average PSNR was 44.21 dB.

Our proposed scheme's average PSNR of 47.82 dB is 8.16 % better than Chimanna and Khot's [5] in terms of imperceptibility. Moreover, our scheme isefficient against a wider range of attacks—nine in all, three of them common with the attacks tested by Chimanna and Khot [5].

Table 13 Chimanna and Khot's scheme performance

Attacks	Extracted Watermark	
	PSNR (dB)	NC
Salt & Pepper Noise	34.59	0.851
Gaussian Noise	43.32	0.890
Median Filtering	44.21	0.914
Rotation	17.75	0.858
Cropping	18.42	0.880

10 Conclusion

This study looked at a range of recent tamper detection schemes for surveillance systems which use video watermarking. Video watermarking schemes can be divided into two broad categories: the spatial domain and the frequency domain. Our brief survey of the advantages and disadvantages of each of these categories concluded that spatial domain schemes are, overall, superior for surveillance systems. The two new schemes proposed and tested in this study, VW16E and VW8F, therefore both belong to the spatial domain.

VW16E is a tamper detection scheme for video files in which 16 bits are used to embed a watermark. Unlike images, video streams have no fixed specification, which means there may be different sizes and locations for video components even for same stream and same format. That is why far fewer schemes have been developed for video applications than image applications. Finding a way to modify video files was the first challenge in developing VW16E. In addition, in order for VW16E to be efficient, the simulated segmentation of each 2*2 block has to consist of four consecutive pixels in the video stream. This structure is an enhancement for two reasons. Firstly, it makes the algorithm faster because, after reading the first two pixels, there is no need to skip to the next row to read the last two pixels. Secondly, the scheme is capable of watermarking more pixels (indeed, almost all the pixels), unlike other common schemes which create real 2*2 blocks and hence exclude pixels at the edges of blocks. More watermarked pixels mean more detectable pixels. Last but not least, in VW16E the address of each block is contained in the embedded 16-bit watermark. In most other schemes, the bits embedded into the blocks as watermarks are generated only from the data of the pixels. This means that our first proposed scheme is able to detect accurately even very fine cut-and-paste blocks.

Our second new scheme, VW8F, is another similar tamper detection scheme for video files. However, in this case only 8 bits are used to embed a watermark. Moreover, the four embedded confidential bits used in VW16E are reduced to one single bit in VW8F. This means that the confidential bits can be spread over a wider area of the video stream, which can improve robustness. In addition, the 12 embedded integrity bits used in VW16E are reduced to seven bits in VW8F. As a consequence, fewer bits and layers are modified, all of which gives VW8F greater imperceptibility than VW16E.

The efficiency and imperceptibility of the two proposed schemes were tested and evaluated, using a range of video samples watermarked by these, which were then subjected to nine common types of attack. For samples from which a watermark was extracted, the NC of the extracted watermarks was calculated. Moreover, in order to test the robustness of the proposed schemes, the resistance of the embedded key was also checked.

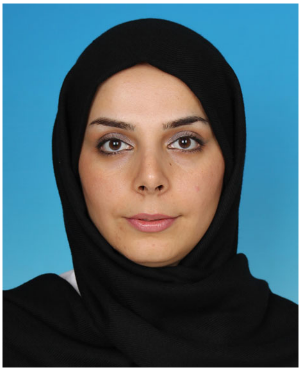
In order to evaluate further the efficiency of the proposed schemes, and specifically of our preferred scheme VW8F, we compared its test results with those of other similar schemes in two areas: (1) efficiency at detecting a range of attacks (tampering); and (2) imperceptibility, as measured by the PSNR. On imperceptibility, the VW8F scheme's average PSNR of 47.82 dB was higher than all the other similar schemes, and substantially higher than most of them. VW8F also demonstrated its ability to detect all eight of the different types of attack to which it was subjected—more than any of the other similar schemes.

In summary, our results suggest that VW8F offers both wider detection capabilities and better imperceptibility than other recent similar digital video watermarking schemes.

References

1. Agarwal H, Ahuja R, Bedi S (2012) Highly Robust and Imperceptible Luminance Based Hybrid Digital Video Watermarking Scheme for Ownership Protection. International Journal of Image, Graphics & Signal Processing 4(11)
2. Amira H, Rhouma R, Belghith S (2010) An Eigen value based Watermarking scheme for tamper detection in gray level images. Systems Signals and Devices (SSD), 2010 7th International Multi-Conference on 1–5
3. Chaluvadi SB, Prasad MV (2009) Efficient image tamper detection and recovery technique using dual watermark. Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on 993–998
4. Chang X, Wang W, Zhao J, Zhang L (2011) A survey of digital video watermarking. Natural Computation (ICNC), 2011 Seventh International Conference on 61–65
5. Chimanna MA, Khot S (2013) Robustness of video watermarking against various attacks using Wavelet Transform techniques and Principle Component Analysis. Information Communication and Embedded Systems (ICICES), 2013 International Conference on 613–618
6. Do H, Choi D, Choi H, Kim T (2008) Digital video watermarking based on histogram and temporal modulation and robust to camcorder recording. Signal Processing and Information Technology, 2008. ISSP IT 2008. IEEE International Symposium on 330–335
7. Giovanni B, Francesco F, Concetta P, Alfio P (2009) Dependable integrated surveillance systems for the physical security of metro railways. Third ACM/IEEE International Conference on Distributed Smart Cameras
8. Giovanni G, Pierpaolo M, Alessandro C, Stefano D, Ugo P, Francesco F (2013) White paper on industrial applications of computer vision and pattern recognition. Lect Notes Comput Sci 8157:721–730
9. Haouzia A, Noumeir R (2008) Methods for image authentication: a survey. Multimed Tools Appl 39(1):1–46
10. Hasnaoui M, Mitrea M (2012) Semi-fragile watermarking for video surveillance applications. Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European 1782–1786
11. He HJ, Zhang JS, Tai HM (2011) A neighborhood-characteristic-based detection model for statistical fragile watermarking with localization. Multimed Tools Appl 52:307–324
12. Huo YR, He HJ, Chen F (2014) A semi-fragile image watermarking algorithm with two-stage detection. Multimed Tools Appl 72(1):123–149
13. Ishtiaq M, Jaffar MA, Khan MA, Jan Z, Mirza AM (2009) Robust and imperceptible watermarking of video streams for low power devices. In Signal Processing, Image Processing and Pattern Recognition (pp. 177–184)
14. Junxiao X, Qingbin L, Zhiyong L (2011) A novel digital video watermarking algorithm. Procedia Eng 24: 90–94
15. Liu M (2012) Study of Digital Video Watermarking. Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on 77–80
16. Liu Z, Li Q, Guan S, Peng X (2009) A robust watermarking algorithm based on differential energy and QIM for uncompressed video. Intelligent Information Hiding and Multimedia Signal Processing, 2009. IIH-MSP'09. Fifth International Conference on 382–385
17. Luo T, Jiang GY, Wang XD, Yu M, Shao F, Peng ZJ (2014) Stereo image watermarking scheme for authentication with self-recovery capability using inter-view reference sharing. Multimed Tools Appl 73(3): 1077–1102
18. Nithyanandam S, Gayathri K, Raja K, Priyadarsini P (2011) Recent Trends in Secure Personal Authentication for Iris Recognition Using Novel Cryptographic Algorithmic Techniques. Process Automation, Control and Computing (PACC), 2011 International Conference on 1–6
19. Redi JA, Taktak W, Dugelay JL (2011) Digital image forensics: a booklet for beginners. Multimed Tools Appl 51(1):133–162
20. Sinha S, Bardhan P, Pramanick S, Jagatramka A, Kole DK, Chakraborty A (2011) Digital video watermarking using discrete wavelet transform and principal component analysis. Int J Wisdom Based Comput 1(2):7–12
21. Su P.-C., Wu C.-Y., Chen Y.-C. (2008) A digital video watermarking scheme for annotating traffic surveillance videos. Multimedia Signal Processing, 2008 I.E. 10th Workshop on 742–747
22. Tokar T, Kanocz T, Levicky D (2009) Digital watermarking of uncompressed video in spatial domain. Radioelektronika, 2009. RADIOELEKTRONIKA'09. 19th International Conference 319–322
23. Tong X, Liu Y, Zhang M, Chen Y (2013) A novel chaos-based fragile watermarking for image tampering detection and self-recovery. Signal Process Image Commun 28(3):301–308
24. Van Schyndel R (2010) A Hardware-based Surveillance Video Camera Watermark. Digital Image Computing: Techniques and Applications (DICTA), 2010 International Conference on 343–348

25. Wang N, Kim C.-H. (2009) Color image of tamper detection and recovery using block-based watermarking. Embedded and Multimedia Computing, 2009. EM-Com 2009. 4th International Conference on 1–6
26. Xu D, Zhang J, Pang B (2010) A Digital Watermarking Scheme Used for Authentication of Surveillance Video. Computational Intelligence and Security (CIS), 2010 International Conference on 654–658
27. Yu PF, Yu PC, Xu D (2014) Palmprint authentication based on DCT-based watermarking. Appl Mech Mater 457:893–898



Farnaz Arab received her Ph.D. from Universiti Teknologi Malaysia. Her research area and interest includes Data Hiding and Multimedia Security.

Shahidan M. Abdullah is a Senior Lecturor with Advanced Informatics School, Universiti Teknologi Malaysia. He receieved his Bachelor of Science (Hons) in 1984 from Universiti Kebangsaan Malaysia in the field Computer Science and his Master of Science in 1997 from Universiti Putra Malaysia in the field of Database and his Doctor of Philosophy in 2010 from Universiti Teknologi Malaysia in the field of Data Hiding and Multimedia Security.

Siti Zaiton Mohd Hashim is an Associate Professor with Department of Software Engineering, Faculty of Computing, Universiti Teknologi Malaysia. She also is Deputy Dean of Office of Postgraduate Department of Software Engineering, Faculty of Computing, Universiti Teknologi Malaysia.

Azizah Bt Abdul Manaf (Ph.D.) is a Professor of Image Processing and Pattern Recognition at Universiti Teknologi Malaysia (UTM). She graduated with B.Eng. (Electrical-Communication & Control) in 1980, M.Sc. Computer Science (1985) and Ph.D. (Image Processing) in 1995. Her current research areas are in Image Processing and Pattern Recognition, Information Security, Watermarking, Steganography and Digital Computer Forensics. She has supervised and graduated a large number of postgraduate students at the Masters and PhD level in these areas and has also authored and co-authored a number of computer related books, written numerous articles in journals and has presented an extensive amount of research papers at national and international conferences on her area of expertise. She has also been invited as keynote speakers and academic talks on her research area at international and national conferences and seminars. Besides being member of ACM, IEEE Computer Society and SDIWC Advisory Board Member, Prof. Dr. Azizah is also currently the President of Malaysia IRRS (International Rough Set Society)-Malaysian Chapter.

Mazdak Zamani received his Ph.D. degree in 2010 on the topic of "Genetic based substitution techniques for audio steganography" from Universiti Teknologi Malaysia (UTM), Malaysia. He was then appointed as Visiting Lecture at UTM until 2012. He has been with UTM as Senior Lecturer since 2012. His main research interests include Multimedia Security, Wireless Security and Secure Architecture and Models.