Chapter

Application of Chaos-Based Fragile Watermarking to Authenticate Digital Video

Rinaldi Munir and Harlili Harlili

Abstract

Fragile watermarking algorithm is a technique used to authenticate of digital data multimedia such as video. A watermarking algorithm consists of two processes: embedding and extraction of watermark. In this paper, a secure video fragile watermarking algorithm in spatial domain based on chaos is proposed. The watermark is a binary image which has the same size with frame size of the video. Before embedding, in order to increase security, the watermark is encrypted using XOR operation with a random image. The random image is generated by using Cross-Coupled Chaotic random Bit Generator (CCCBG). The encrypted watermark is embedded to each frame. In the extraction process, the encrypted watermark is extracted from the watermarked video, decrypted it, and then compared to the original watermark. This algorithm has capability to localize the area being tampered in the video frames. We have performed some typical attacks to the watermarked video and then authenticated it. Based on the experiment results, the algorithm can detect and localize the modified region of video frames very well. Sensitivity to the slightest change on initial conditions of the chaos map provided security of the algorithm.

Keywords: fragile watermarking, authentication, digital video, chaos

1. Introduction

Nowadays digital video is widely used to present information. This is because a video is richer in information compared to a single image, and generally a video has more pictures and sound. However, digital data such as video have advantages and disadvantages. Digital video could be edited, manipulated, or altered easily by using a video editor or other tools. For example, someone could change contrast of the video, resize, remove or add some frames, or add a new object to the video. Unfortunately, once a digital video is manipulated, its integrity is questionable. In some cases, we need to know authenticity of the video. For example, a court need to decide if a video as evidence is genuine or has been manipulated. If the video has been manipulated, how to prove it?

The integrity problem of digital video could be solved by using a fragile watermarking technique. In the fragile watermarking technique, we could embed one or more watermarks into frames of video. Once the watermarked video is manipulated, altered, or modified, the watermark inside will be fragile or damage. The damaged watermark is indication that the video has been manipulated. Therefore, fragile watermarking can be used to prove authentication of a video.

A watermarking algorithm consists of two processes: embedding and extraction of watermark. Watermark is an information that refers to the video owner. Usually the watermark is a binary image such as logo, random bits, or other information. The watermark is inserted into a host video, frame by frame, become to a watermarked video without affecting its perceptual (and audio) quality. Through an inverse process, the embedded watermark can be extracted again from the video. When the extracted watermark is compared to the original watermark, we could conclude if the video has been altered. The damage extracted watermark is indication that the video has been altered.

Digital watermarking schemes can operate on spatial domain or frequency domain. Assume the watermark is represented as string of bits. Watermarking schemes that operate in spatial domain embed watermark bits into pixel values of the video frame directly [1, 2]. Otherwise, on watermarking schemes that operate in transform domain, a host video has to be transformed first into a transform domain by using a specific transformation (DCT, DWT, DFT, etc.) [3]. Next embedding bits of watermark is performed by modifying the transform coefficients [2, 4].

Generally watermarking in transform domain is more robust than spatial domain through non-malicious attacks such as cropping, compression, scaling, rotation, etc. Therefore, it is used to the problems of copyright protection, proving ownership, illegal copying, and transaction tracking of video. Otherwise, watermarking in spatial domain is suitable to solve the problem of tamper detection of video content. The watermark in the video must be fragile when the video is manipulated. Robustness is not important for fragile watermarking.

The watermark could be originated from internal or external. The internal watermark means that the watermark is derived from host video directly, and then it is embedded into frames of the video. The second is external watermark which means that the watermark is an input from the user, and usually the watermark is a meaningful binary image such as logo or other image.

Much research on fragile watermarking for digital video has been done by many scientists. This means that the fragile watermarking is interesting research topics. Some related research is from Elgamal et al. [1], Zhi-yu et al. [4], and Rupali et al. [5]. Elgamal et al. [1] proposed a fragile video watermarking algorithm on transform. The original video is transformed from RGB model to YCbCr model and then Cr-component is partitioned into non-overlapping blocks of pixel. The watermark is a binary image from the video owner. Bits of the binary image are embedded for each block separately.

Like Elgamal et al. [1], Zhi-yu et al. [4] also proposed a fragile watermarking algorithm on transform domain. The original video is transformed from RGB model to YST model. The T-component is divided 4 × 4 blocks and then each block is transformed to frequency domain using DCT. The watermark is generated from the quantized DCT coefficient and then it is embedded into the last non-zero DCT coefficient.

Rupali et al. [5] also proposed a fragile video watermarking algorithm on transform domain. The original video is transformed from spatial domain to frequency domain using DCT. Rupali et al. [5] used two watermarks to embed. Both of these watermarks are internal, that means from the original video itself. The first watermark is used to detect tampering and the second watermark is used to localize tampered area.

All of the watermarking algorithms above operated on transform domain. A digital video contains more frames, generally hundreds to thousands frames. Transformation of each frame from the spatial domain to the transform domain consumes a lot of computing time. We need a simple fragile watermarking for the digital video but still meet the security aspect. A simple fragile watermarking on spatial domain is by using LSB (least significant bit) modification method. This method is fast and it can detect tampering on video until pixel level. To fulfill the security aspect, we use the watermarking key(s) in embedding and extraction process, so that embedding and extraction of watermark are performed by an authorized party who has the secret key(s).

The watermark itself is confidential, only the video owner knows about it. Therefore, the watermark needs to be encrypted using the secret key(s). The secret key(s) also serve to prevent the watermark from being extracted and used in the reassembling of videos by an authorized party, thus avoiding counterfeiting of the videos.

The chaos system can be used to get a secure fragile watermarking scheme. In recent years, chaos theory has attracted the attention of scientists, especially in the information security field. Chaos has been used to increase security [6]. The reason is the chaotic systems that have sensitivity on initial conditions. It means if we perform a little bit change to the initial conditions of the chaos system, after some iterations, the system will result values that differ significantly. In the field of cryptography and watermarking, generally a chaos map is used to generate pseudo-random numbers [7, 8].

Munir et al. [9] used a chaos map, i.e. Arnold Cat's map, to encrypt the watermark before embedding it into video frames. The original watermark is encrypted by XOR-ing it with a random image. The random image is generated from the replicated watermark by using an Arnold Cat Map. The encrypted watermark is embedded into each frame of the video using LSB modification method.

Unfortunately, the random image generated using Arnold Cat's map still shows patterns of the replicated watermark, so it is not completely random. The embedding algorithm is also redundant, because the random image is XOR-ed with the replicated watermark. Therefore, the watermarking algorithm has redundancy.

In this paper, we modified the previous algorithm by using another chaos map so that the random image is generated from the chaos map directly. We used a random bit generator based on two Skew Tent Maps. The generator is abbreviated as CCCBG (Cross-Coupled Chaotic random Bit Generator) [10].

The paper is organized into six sections. The first section is this introduction. The second section will review some supported theories. The algorithm to embed and extract the watermark will be explained in the third section. The fourth and fifth section will present the experiment results and discussion. Finally, the last section will resume the conclusion and future work.

2. Chaos map

One of the popular chaotic maps is a Logistic Map, described by

$$x_{i+1} = \mu x_i \left(1 - x_i\right) \tag{1}$$

where μ is a parameter of map and $0 < \mu \leq 4$. According to [7], the map is in chaotic state when $3.57 < \mu \leq 4$. In this state, the resulting values appear random. Because of its random behavior, a logistic map can be used as a pseudo-random generator. Hence, initial value of the chaos map, x_0, and constant μ serve as secret keys. When we iterate Eq. (1) from an initial value (x_0), we get a random sequences between 0 and 1. The random values generated from the chaos Logistic Map are sensitive to small changes of the initial values. By changing x_0, the random values generated different significantly from the previous chaotic values with initial value x_0.

Another chaos map is Tent Map. It iterates a point x_0 and gives a sequence x_i in [0, 1]:

$$x_{i+1} = f_\mu\left(x_i\right) = \begin{cases} \mu x_i & , x_i < \frac{1}{2} \\ \mu\left(1 - x_i\right) & , x_i \geq \frac{1}{2} \end{cases} \tag{2}$$

where μ is a positive real constant. Varian of Tent Map is Skew Tent Map [8] which is defined as:

$$x_{i+1} = f_\mu\left(x_i\right) = \begin{cases} \dfrac{x_i}{\mu} & , 0 \leq x_i < \mu \\ \dfrac{1 - x_i}{1 - \mu} & , \mu < x \leq 1 \end{cases} \tag{3}$$

where μ is system parameter and x_0 is initial condition of map. When a Skew Tent Map is iterated from x_0 value, it produces a sequence in the interval [0, 1] and distributed uniformly.

Narendra et al. in [10] proposed a random bit generator based on two Skew Tent Maps. The generator is abbreviated as CCCBG (Cross-Coupled Chaotic random Bit Generator). In the CCCBG, random bit stream is generated by comparing outputs of the couple maps. If $f_\mu(x_i)$ and $g_\mu(y_i)$ are two Skew Tent Maps and are given as:

$$x_{i+1} = f_\mu\left(x_i\right) \tag{4}$$

$$y_{i+1} = g_\mu\left(y_i\right) \tag{5}$$

where μ is system parameter and is same for both maps. The CCCBG generated a sequence of random bits by comparing the outputs of the maps in the following way:

$$h\left(x_{i+1}, y_{i+1}\right) = \begin{cases} 0 & , x_{i+1} < y_{i+1} \\ 1 & , x_{i+1} \geq y_{i+1} \end{cases} \tag{6}$$

Based on several tests performed by Narendra et al., the CCCBG successfully passes all the randomness tests [10], therefore the random binary sequences can be used for encryption.

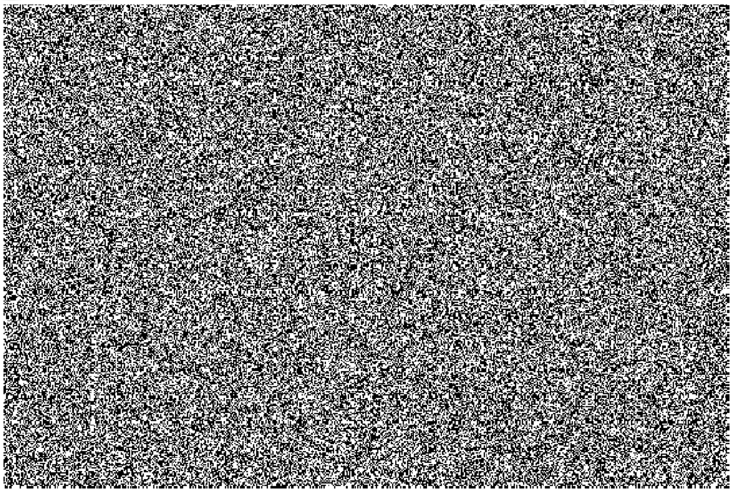

Figure 1.
A random image generated by using the CCCBG.

In digital watermarking, the random bits play important role to increase security. We will use CCCBG Map to generate random bits. The random bits will be-XOR-ed to the original watermark to yield the encrypted watermark. For example, by iterating (6) 240,000 times (i.e., 480 × 854) and using parameter $\mu = 0.48999$, initial conditions $x_0 = 0.500684$, and $y_0 = 0.538167586$, we get a sequence of random bits figured as a binary bit image (**Figure 1**).

3. Proposed algorithm

This section will explain the proposed fragile video watermarking on spatial domain based on the chaotic map. The watermark is a binary image. To detect manipulation in the video frames until pixel level, the watermark must have the same size as the video frame size. Therefore, if watermark size is less than video frame size, the watermark need to be replicated by duplicating it a number of times in order to produce a new watermark that has the same size with the host video frame size. **Figure 2** shows example of replication. The original watermark "ASEAN logo" has a size of 200 × 194 pixels, whereas the video frames have a size of 480 × 854 pixels. This original watermark must be duplicated a number of times so that produce a replicated watermark that has size 480 × 854.

Next, to increase security, before embedding, the replicated watermark is encrypted by XOR-ing it with a random image. A random image is generated by iterating CCCBG a number of mn times where m and n are frame sizes (**Figure 1**). The replicated watermark is encrypted with the random image by using XOR operation to produce an encrypted watermark (**Figure 3**). Next, we embed the encrypted watermark into the host video.

We design a simple, but secure, fragile video watermarking based on chaos. The fragile video watermarking algorithm consists of two processes: embedding algorithm and extraction algorithm, each will be described below.

3.1 Watermark embedding algorithm

There are two scenarios for embedding the watermark into a digital video. The first scenario is embedding each frame of the video with the same watermark. The second scenario is embedding each frame of the video with the different watermark. The second scenario is not practical because the video owner have to provide the watermark of as many frames. Actually, the watermarks can be generated from the video itself (i.e., internal watermarks), we generate the watermark for each video frame that

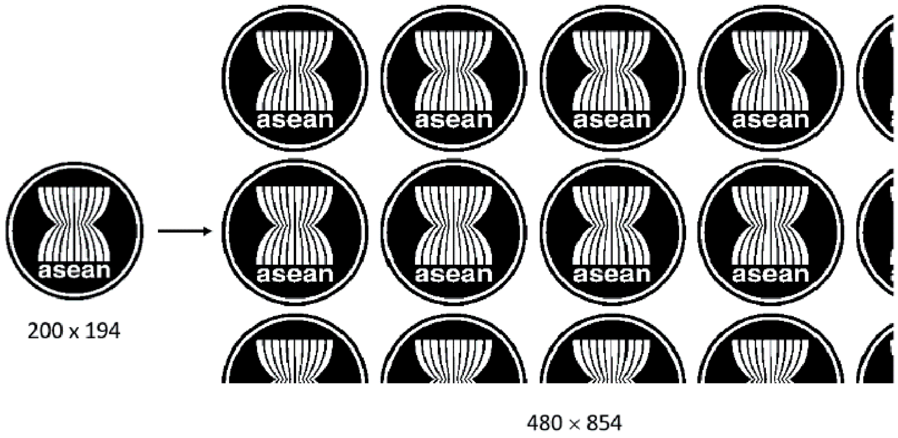

Figure 2.
Left: the original watermark; right: the replicated watermark.

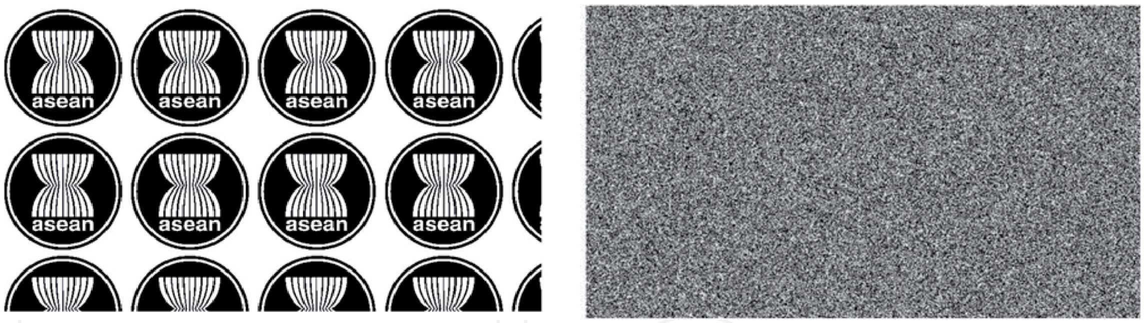

Figure 3.
Left: the replicated watermark. Right: the encrypted watermark.

depend on the frame content itself. However, the resulting watermark is meaningless and cannot be perceived visually. We want the watermark to be meaningful and can be perceived visually. Therefore, we choose the watermark is a meaningful binary image and, for practical reason, the same watermark is embedded to each video frame.

Now, we can describe the watermark embedding algorithm into the digital video in more detail as follows:

Input: a host video file (v), a watermark file (w), and CCCBG's parameter and initial conditions (μ, x_0, y_0).
Output: a watermarked video (v').
Step 1: Read the frames of video v, the watermark w, and CCCBG's parameter and initial conditions (μ, x_0, and y_0). If the video has an audio, then separate the audio.
Step 2: If size(w) < size(video frame of v), copy the single watermark to produce a replicated watermark w' which has the same size with the host video frames.
Step 3: Iterate CCCBG mn times to produce a random image r.
Step 4: Encrypt w' by XOR-ing it with r as follows:

$$w" = w' \oplus r \tag{7}$$

Step 5: Embed the encrypted watermark, w", into each frame of the video by manipulating the least significant bit (LSB) of pixels. If the frame has R, G, and B component, then it performs embedding to each component.

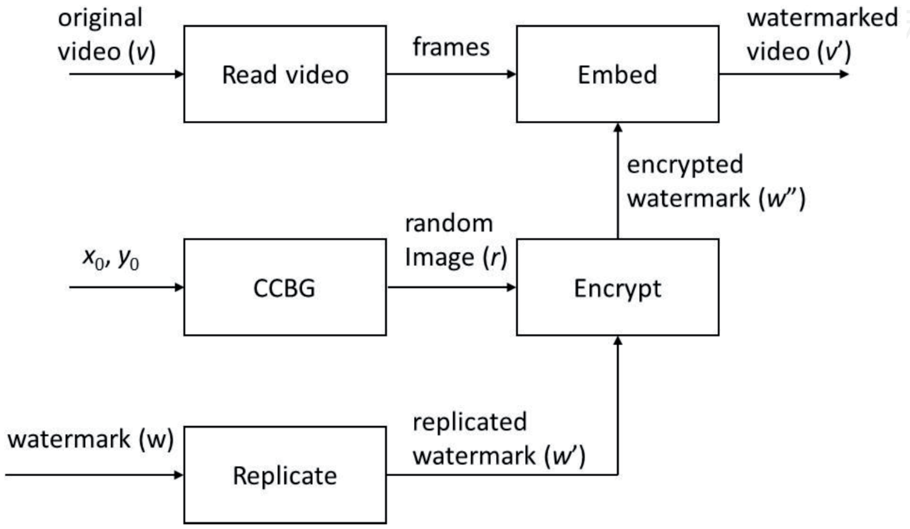

Figure 4.
Watermark embedding algorithm.

Step 6: If the original video has audio, merge it to the watermarked frames to produce a watermarked video.

Figure 4 shows stages in the watermark embedding algorithm. The watermark is embedded into each frames of video. If the video has audio, the audio has not been changed. After embedding of the watermark, audio is merged back into the video.

3.2 Watermark extraction algorithm

Watermark extraction from the video is performed to prove video authentication. Only the video owner can do it, because the owner has the original watermark. The original watermark is required to compare it with the extracted watermark. If the extracted watermark is the same as the original watermark (can be observed visually), then we decide that the video is authentic, otherwise the video has been altered, tampered, or manipulated. The original watermark is also required to localize the tampered region in a frame. Watermark extraction can be performed on all video frames or only on certain frames.

Now, we can describe the watermark extraction algorithm from the digital video in more detail as follows:

Input: a watermarked video file (v'), an original watermark file (w), and CCCBG’s parameter and initial conditions (μ, x_0, and y_0).

Output: an extracted watermark, location of tampered frame (if any).

Step 1: Read the frames of video v', watermark w, and CCCBG’s parameter and initial conditions (μ, x_0 and y_0).

Step 2: If size(w) < size(video frame of v), copy the single watermark to produce a replicated watermark w' which has the same size with the host video frames.

Step 3: Iterating CCCBG mn times to produce a random image r.

Step 4: For each frame, extract all of the least significant bit (LSB) of pixels. This step yields an extracted watermark w''.

Step 4: Decrypt the watermark w'' by XOR-ing it with r as:

$$w''' = w'' \oplus r. \tag{8}$$

Step 5: Compare w''' with w'. If $w''' = w'$, we conclude that the integrity of video is authenticated. If not, go to step 6 and 7.

Step 6: To localize tampered region, subtract w' to w'''. If a pixel is not changed, the subtraction yields 0, else the subtraction yields 1.

Step 7: Identify pixels in the watermarked framed in position where the subtraction above yields 1. Those are pixels that have been manipulated.

Figure 5(a) shows stages in the watermark extraction algorithm. The watermark is extracted from each frame of video and then compares the extracted watermark to the original watermark. The decision is binary (1 or 0), 0 means the watermarked video has not changed (authentic), 1 means the watermarked video has been manipulated, altered, or tampered. **Figure 5(b)** shows how to localize tampered region in a frame.

4. Experiment and results

We have implemented the proposed algorithm to be a computer program. Next we test the algorithm on a sample video to determine authentication of the video. The

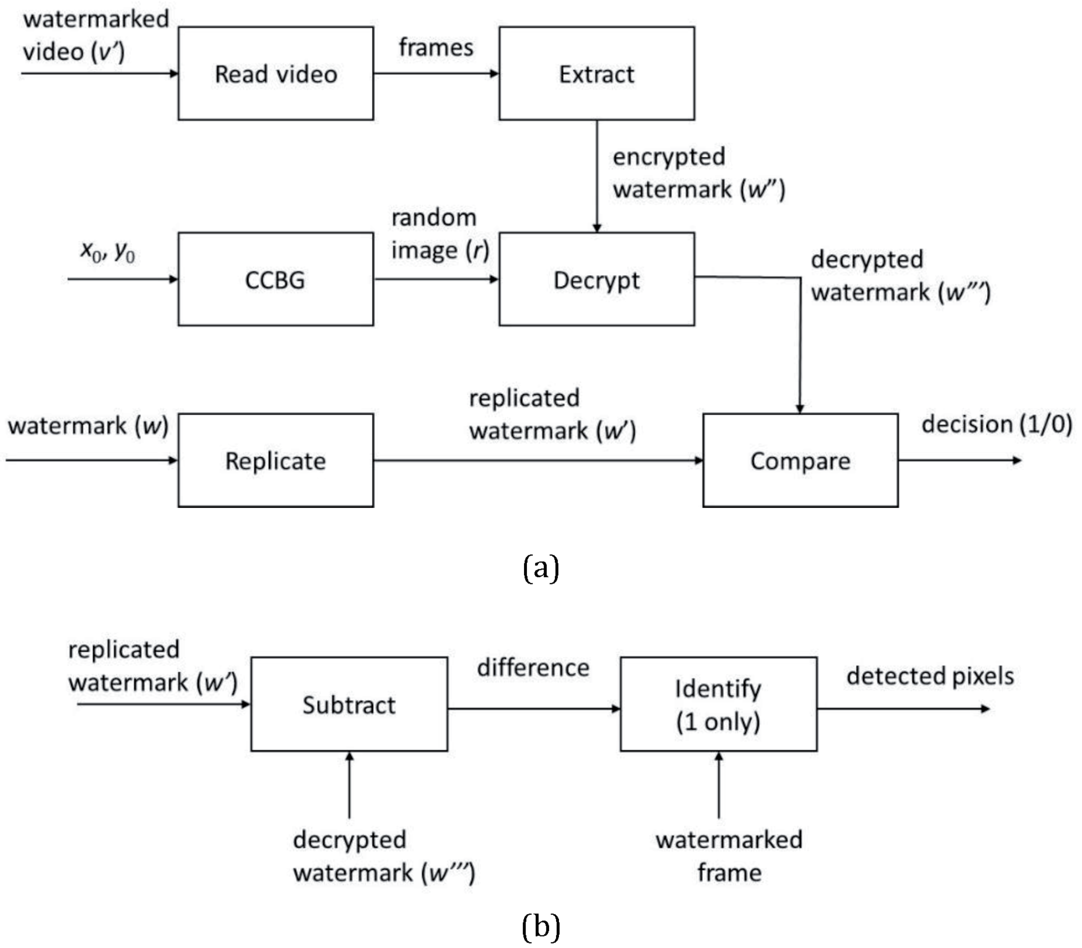

Figure 5.
(a) Watermark extraction algorithm and (b) localize tampered region.

sample video was a video clip which has 394 frames and long 16 seconds, each frame has a size 480 × 854 (**Figure 6a**). This video contains audio inside. **Figure 6b** and **c** show two frames of the video (frame 1 and frame 175).

The watermark to be embedded into the host video is "ASEAN logo" as shown in **Figure 3** (after replicated). In these experiments below, we used parameters of CCCBG as follows: μ = 0.48999, initial conditions x_0 = 0.5006841, and y_0 = 0.538167586. The two initial conditions serve as the secret keys. These keys were used in both watermark embedding algorithm and extraction algorithm.

This algorithm can only be used to test the authentication of digital videos that have been spatially manipulated. Spatial manipulation such as changing the contrast of the image, adding noise, changing the size of the frame, copy and paste an object into the frame, and others. The algorithm cannot temporarily detect video manipulation, for example, by removing one or more frames.

In the section below, we divide experiments into two cases: (i) no attack case and (ii) tamper detection test, each will be described below.

4.1 No-attack case

If the watermarked video is not manipulated, we categorized it as no attack case. To prove the authentication of the video, we extracted all watermarks from each video frame. All watermarks should be the same as the original watermark. **Figure 7** shows the extracted watermarks but only from frame number 1 and frame number 283.

Visually, there are no damages in the extracted watermarks. When we compare the extracted watermark to the original watermark by subtracting them, we get

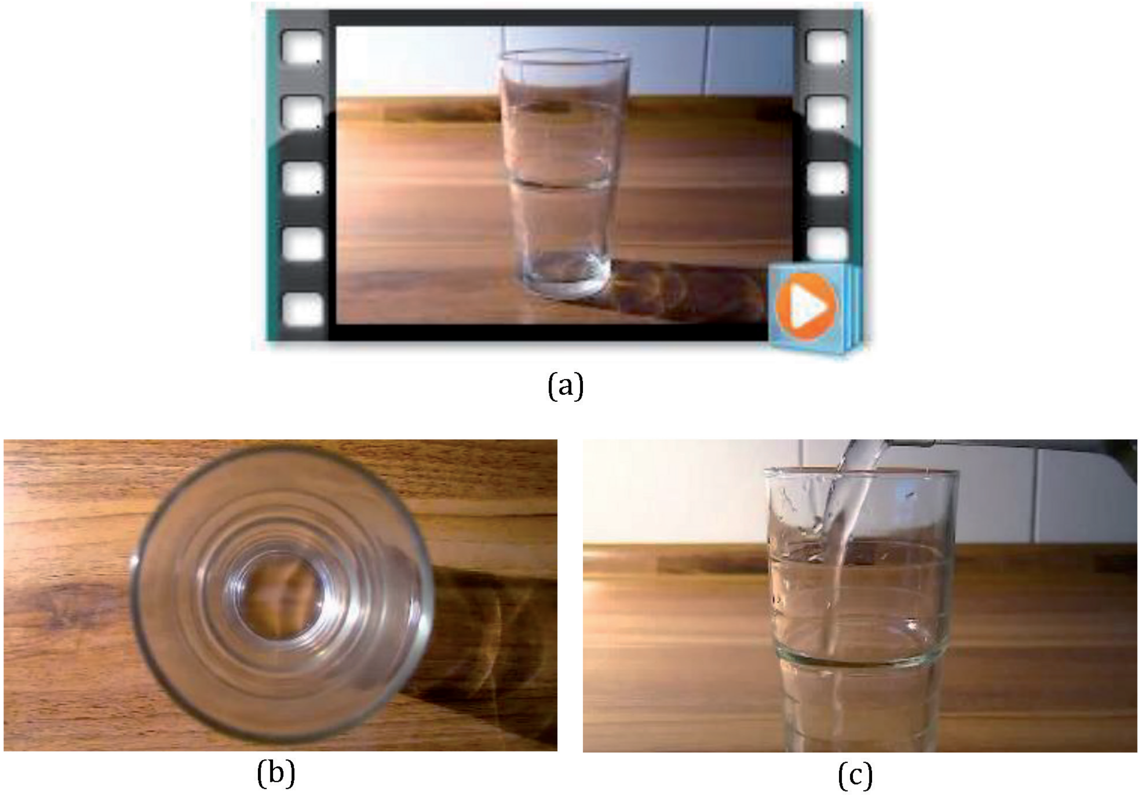

Figure 6.
(a) A host video to be watermarked, (b) frame 1, and (c) frame 175.

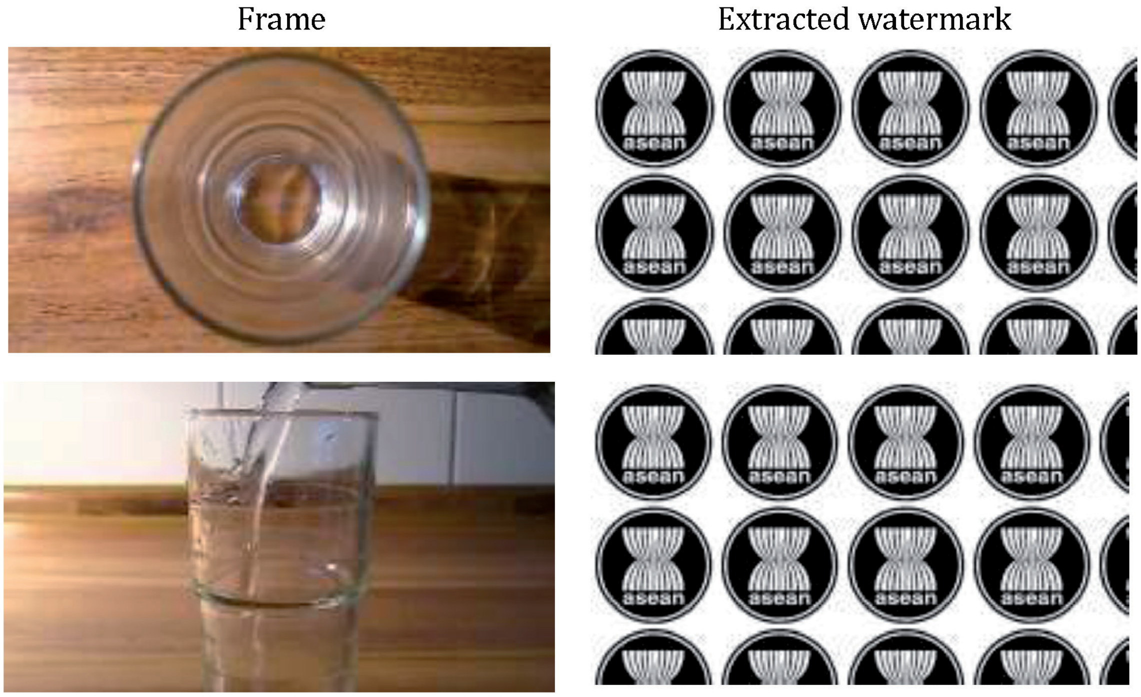

Figure 7.
The watermarked frames and the extracted watermarks.

the difference is a black image (all of pixels are 0). Therefore, we conclude no tampering performed to the watermarked video. In this algorithm, parameter and initial condition of CCCBG behave as secret keys. Embedding and extraction of the watermark could be done by the authorized party only. If the receiver did not have the same keys, then the extracted watermark is not the same as the original watermark.

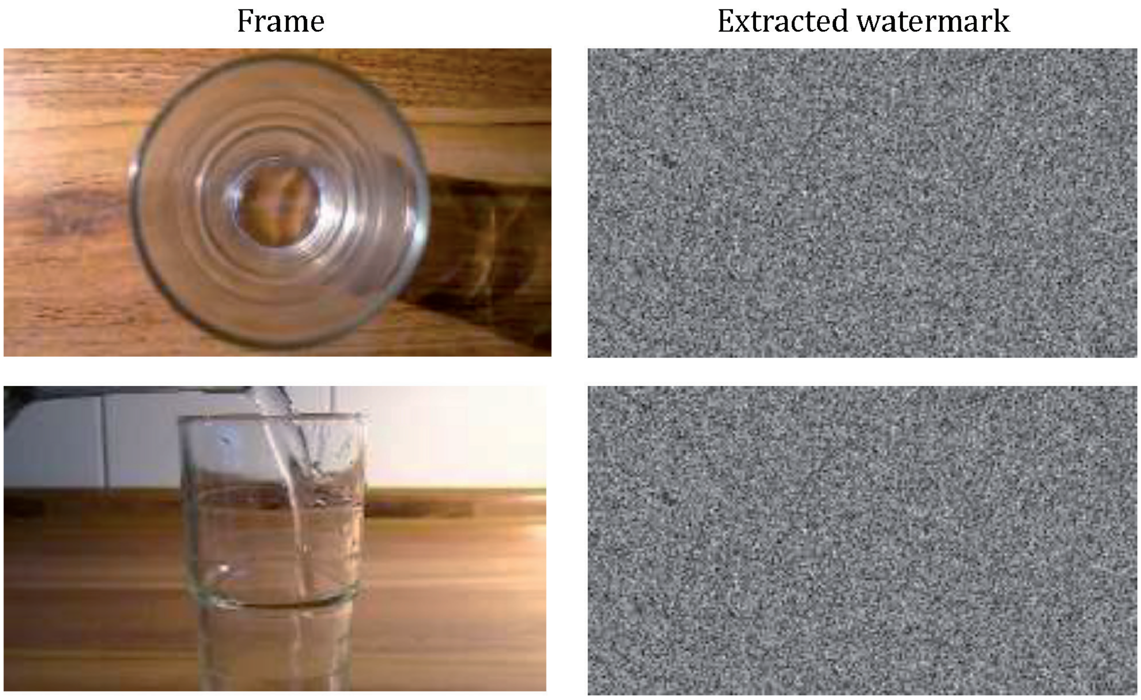

Figure 8.
The watermarked frames and the extracted (wrong) watermarks.

4.2 Sensitivity to initial condition

As mentioned before, the chaos system has a sensitivity to the slightest change on initial conditions. This characteristic provides the security aspect of watermarking. For example, the receiver used μ = 0.4900 (before was 0.48999), initial conditions x_0 = 0.5006840 (before was 0.5006841), and y_0 = 0.538167585 (before was 0.5381675865) to extract the watermark. **Figure 8** shows the extracted watermarks from two frames. The extracted watermarks are wrong! Compared to the original watermark, this extracted watermarks look like the random images. This happens because CCBG produces random bits that are very different from previous bits.

4.3 Tamper detection test

In most cases, a digital video is often edited or manipulated using the video editor. If a video has been manipulated, the video is no longer original. Main goal of fragile watermarking is to determine if the video has been manipulated or not. If the video has been manipulated, the algorithm should able to locate where the alteration made on the video frames. In these experiments, we performed some typical attacks to the watermarked video. The attacks are (1) adding a text to the watermarked video, (2) copy-paste attack, (3) adding some noises, (4) modifying video contrast, and (5) cropping the frames. The following are the attacks.

5. Detection test against text addition

We attack the watermarked video by writing a text “GLASS and WATER” at the left top of the frames (**Figure 9a**). To prove the authentication of the video, we extracted the watermarks from the video frames to get the extracted watermarks. The extracted watermarks contain the text (**Figure 9b**). Therefore, we conclude that the watermarked video has been manipulated. **Figure 9c** shows detection of

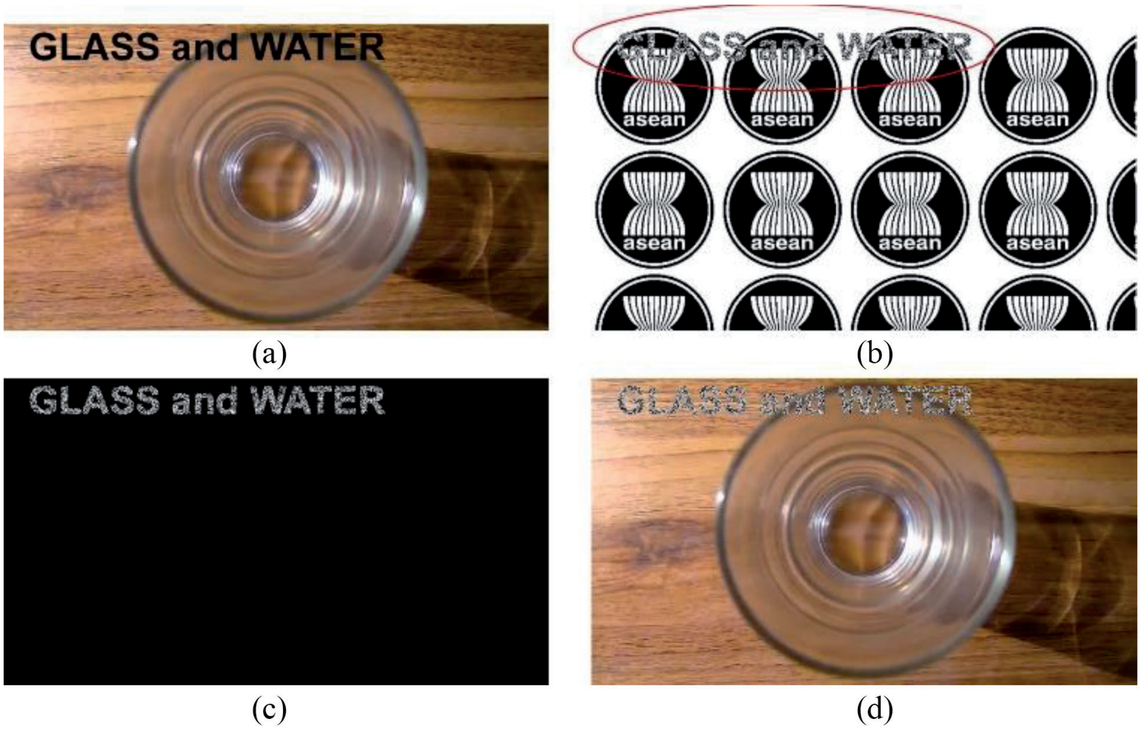

Figure 9.
(a) Watermarked frame after adding a text; (b) extracted watermark; (c) and (d) detected tampering region.

pixels that have been manipulated by adding a text "GLASS and WATER." **Figure 9d** shows the tampered pixels in the correspondence frame.

6. Detection test against copy-paste attack

In the second attack, we copied an object "coca-cola bottle" and then pasted it into the watermarked video (**Figure 10a**). When we extracted the watermarks from the

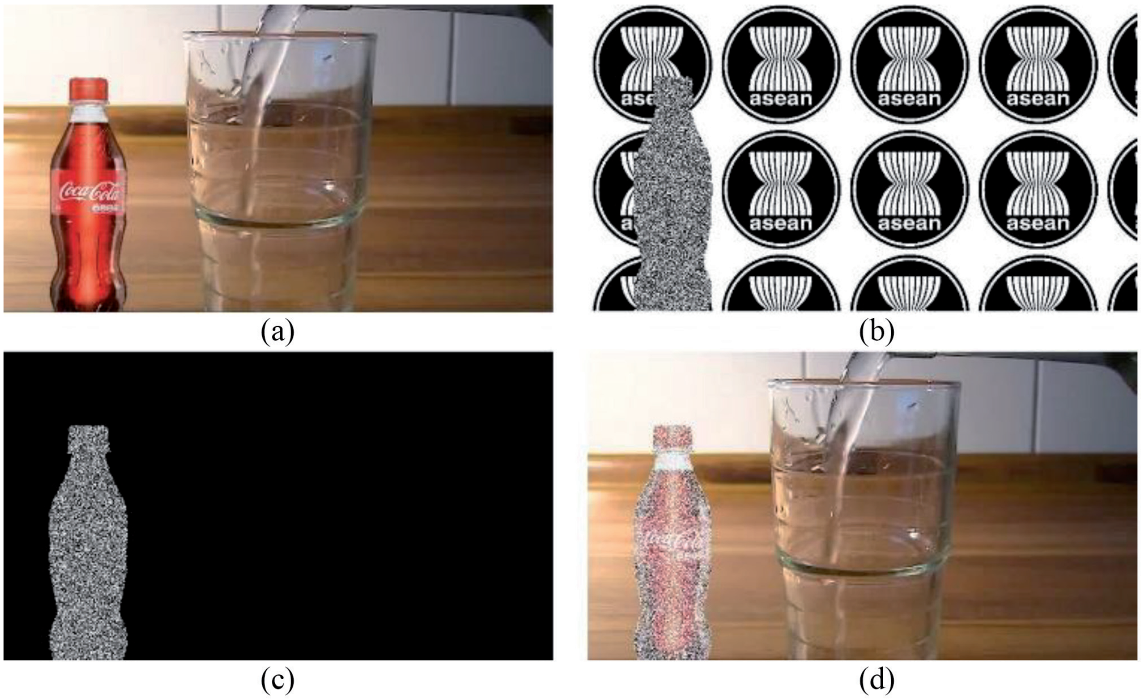

Figure 10.
(a) Watermarked frame after copy-paste attack; (b) extracted watermark; (c) and (d) detected tampering region.

video, we got an extracted watermark as shown in **Figure 10b**. The extracted watermark contains a silhouette of strange object inside. Localize the tampered region and we can detect copy-paste object in the video frames as shown in **Figure 10c** and **d**.

7. Detection test against adding some noises

There are some kinds of noise such as Gaussian noise, salt and pepper noise, Poisson noise, etc. In the third attack, we added "salt and pepper" noise with density 0.1 into the watermarked video (**Figure 11a**). When we extracted the watermarks from the video, we got the watermarks also contained noise. The noisy watermarks indicated that the video has been altered (**Figure 11b**). The tampering region is entire of frame (**Figure 11c** and **d**).

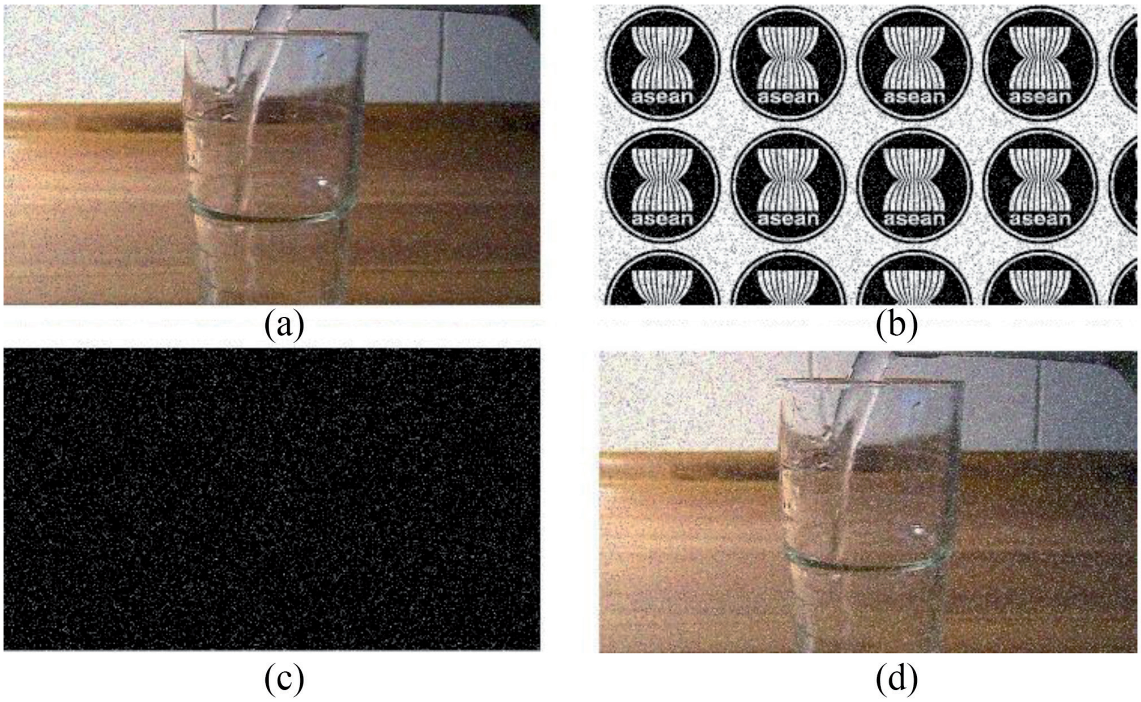

Figure 11.
(a) The watermarked frames after adding noise "salt and pepper"; (b) the extracted watermark; (c) and (d) detected tampering region.

8. Detection test against contrast change

A digital video can be changed so that the contrast becomes brighter or darker. In this attack, we manipulated the watermarked video by changing the contrast so that make it brighter. After that, we extracted the watermarks from the video (**Figure 12**, top right). We can see that the extracted watermarks are damaged and cannot be recognized anymore. Localization of tampered region shows that whole of image has been manipulated (**Figure 12**, bottom right).

9. Detection test against cropping

One of the geometrical attacks is cropping. In this attack, we manipulated the watermarked video by cropping the video frames. The cropping can be performed horizontally or vertically. In this experiment, we cropped a left side of the video. To extract the

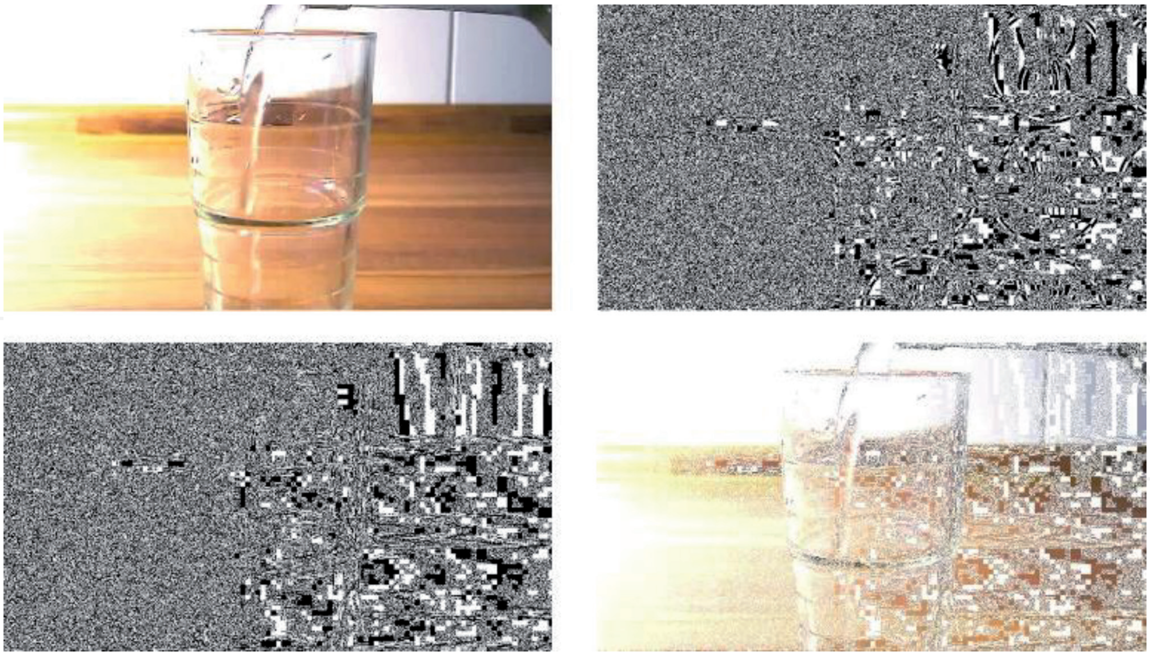

Figure 12.
Top left: the watermarked frame after changing the brightness; top right: the extracted watermark. Bottom right: the tampered region.

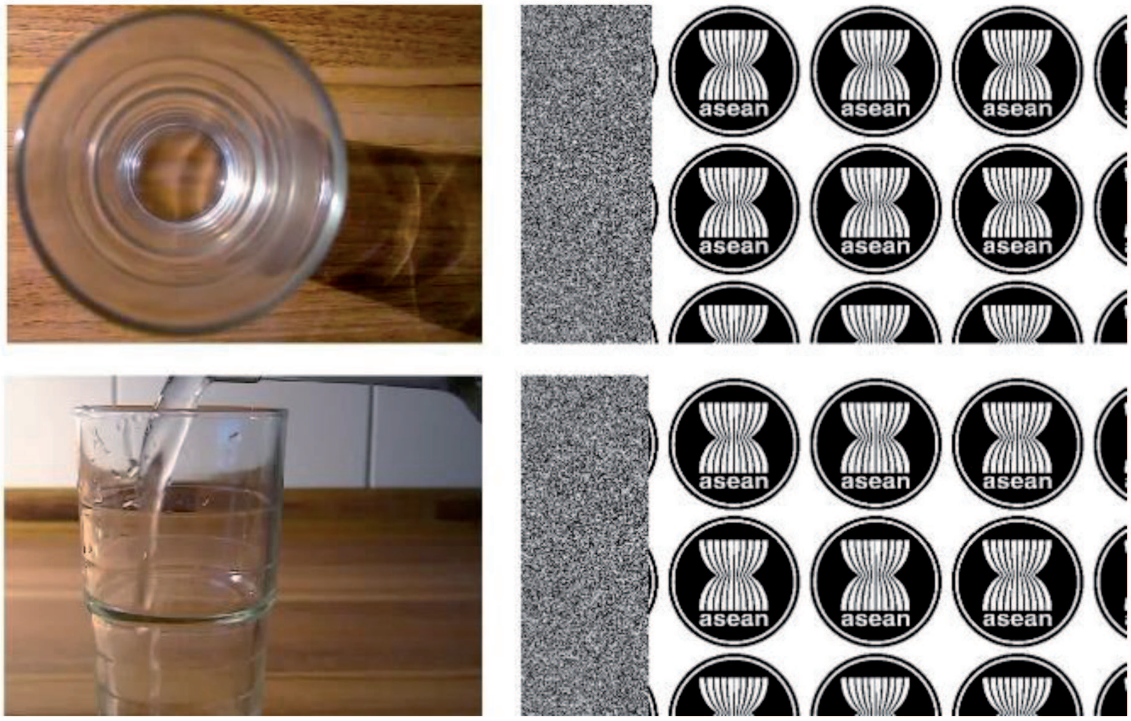

Figure 13.
The watermarked frames after cropping and the extracted watermarks.

watermarks, we returned the frame size into original size first by adding white or black pixels (in this experiment we added white pixels). We found the extracted watermarks contained the black region that indicated the cropped region in the frames (**Figure 13**).

10. Discussion

The proposed chaos-based fragile watermarking algorithm is simple but secure; it can be used to authenticate the digital video. Some experiments have been done to test performance of the algorithm. If there is no manipulation done to the watermarked video (no attack case), then the extracted watermark is same exactly to the original watermark. Therefore, we conclude that the video is still original, has not been changed or manipulated.

Common manipulations of video have been done to test authentication and localize altering in the watermarked video. These manipulations are adding a text label into video frames, inserting a new object into the video, changing contrast, and cropping some pixels. In the case of adding text and inserting an object into the video frames, we got the extracted watermarks that contain silhouette of the object or text. The silhouettes can be seen visually. When we compared to the original watermark, the extracted watermark is not the same. Therefore we conclude that the video has been manipulated. By subtracting the original watermark from the extracted watermark and adjusting the results on the watermarked video, we can find the video frame portion that has been changed.

Common manipulation of video is changing the contrast or brightness of the video. By changing the contrast or brightness of the video, it means changing all pixel values in the video frame. When the watermarks are extracted from the video, we found that extracted watermarks also change entirely. The extracted watermarks are totally damaged; therefore we can conclude that the video has been manipulated.

When a block area in the watermarked video frame is cropped, the extracted watermark is also cropped in the correspondence block. The extracted watermark has a black region in the cropped area of the correspondence frame.

This proposed algorithm has some weakness. It cannot detect manipulation of the watermarked video if one or more fames are removed. However, if some video frames are inserted to the watermarked video, the algorithm can still detect this manipulation, because the new frames do not contain the embedded watermarks.

Other weakness is LSB modification method itself. Bits of the watermark are only embedded to one least significant bit of pixel values. If manipulation of the watermarked video is performed on other than the least significant bit, the algorithm cannot detect it. However, this manipulation is considered uncommon so it can be ignored.

11. Conclusion and future works

We have proposed a fragile video watermarking based on the chaotic map. In order to increase security, the watermark is encrypted using XOR operation with a random image. The random image is generated by using Cross-Coupled Chaotic random Bit Generator (CCCBG). The encrypted watermark is embedded to every RGB component of each frame. In the extraction process, the encrypted watermark is extracted from the watermarked video and compared to the original watermark.

Some experiments have been done to test capability of the algorithm to detect tampering to the watermarked video. We have tried some common attacks to the watermarked video. The experiment results showed that the algorithm could detect tampering on the watermarked video. This algorithm has also capability to localize the area being tampered in the video frames.

The algorithm can only be used to the uncompressed videos. It can be developed for the compressed video format such as MPEG-4. Embedding of watermarks is performed in encoding and decoding must be operated in transform domain. Some transform methods such as Fourier Transform, DCT, or wavelet transform can be used.

This algorithm can also be developed so that it can detect the manipulation that removes one or more frames from the watermarked video. This can be done by using the internal watermarks that depend on the entire video content. Therefore, if some video frames are removed, the internal watermarks also change.

Author details

Rinaldi Munir* and Harlili Harlili
School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung, Indonesia

*Address all correspondence to: rinaldi@informatika.org

References

[1] Elgamal AF, Mosa NA, ElSaid WK. A fragile video watermarking algorithm for content authentication based on block mean and modulation factor. International Journal of Computer Applications. 2013;**80**(4):0975-8887

[2] Jayamalar T, Radha V. Survey on digital watermarking techniques and attacks watermark. International Journal of Engineering, Science and Technology. 2010;**2**(12):6963-6937

[3] Maryam A, Mansoor R, Hamidreza A. A novel robust scaling image watermarking scheme based on Gaussian mixture model. Expert Systems with Applications. 2015;**42**(4):1960-1971. Available from: https://www.sciencedirect.com/science/article/abs/pii/S0957417414006381

[4] Zhi-Yu H, Xiang-Hong T. Integrity authentication scheme of color video based on the fragile watermarking. In: Proceedings of 2011 International Conference on Electronics, Communications and Control (ICECC). 2011. Available from: https://ieeexplore.ieee.org/document/6067709

[5] Rupali DP, Shilpa M. Fragile video watermarking for tampering detection and localization. Proceedings of 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI). 2015

[6] Dawei Z, Guanrong C, Wenbo L. A chaos-based robust wavelet-domain watermarking algorithm. Chaos, Solitons & Fractals. 2004;**22**:47-54

[7] Bose R, Banerjee A. Implementing symmetric cryptography using chaos function. In: Proceeding 7th International Conference on Advanced Computing and Communication (ADCOM). Indian Institute of Technology; 20 Decembe 1999. pp. 318-321

[8] Stojanovski T, Pihl J, Kocarev L. Chaos-based random number generators - part II: Practical realization. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications. 2001;**48**(3):382-385

[9] Munir R, Harlili H. A secure fragile video watermarking algorithm for content authentication based on Arnold Cat's map. Proceedings of the 4th International Conference on Information Technology (InCIT2019). Bangkok, Thailand; 24-25 October 2019

[10] Narendra KP, Vinod P, Krishan K. A random bit generator using chaotic maps. International Journal of Network Security. 2010;**10**(1):32-38