

Authentication of Animated GIF Images by Using A Fragile Watermarking Scheme Based on EzStego Algorithm

Rinaldi Munir *

School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Indonesia

Abstract

Fragile watermarking could be used to authenticate a digital image due to modification or altering. A mark (or watermark) is embedded into the image. When the image was modified or altered, the mark is also altered or fragile. There is no a fragile watermarking algorithm especially for the animated GIF image. However, we could modify the existing steganography algorithm to build a fragile watermarking scheme. In this paper, we used EzStego, an existing steganography algorithm for GIG image, to embed a watermark into frames of the animated GIF image. Before embedding, the watermark is encrypted with the random bits. Authentication is performed by extracting the watermark from the watermarked image and then compare it with the original watermark. If the extracted watermark is fragile, so it indicates that the image is not authentic anymore. We also could find parts of the image that has been altered. We have test performance of the proposed scheme by doing some typical attacks to the watermarked images. Based on experiments results we conclude that the watermarking scheme can detect the authentication of the animated GIF image due to the attacks.

Keywords: authentication, fragile watermarking, animated GIF image, EzStego, watermark, fragile

1. Introduction

Information can be represented in many things, one of them in image form. Digital images play an important role in current digital era. However, the digital images can be edited, altered, or manipulated by an unauthorized party. When an image is altered (for example by using a software such as Photoshop), the image is not authentic anymore. To give proving of authentication of the image, the solution is by using fragile watermarking. Fragile watermarking is a technique by embedding a mark (we called it a watermark) into the image, so when the image is modified, the mark is also altered or destroyed. Even a very small change to the image and no impact on the visual quality of the image, it will result a tampered or fragile watermark. The sensitivity of fragile watermark to modification leads to their use in image authentication [1]. The third party can verify that an image has not been edited or altered since it was marked.

There are many kind of image formats. The popular image formats are BMP, JPEG, PNG, GIF, etc. Most of the work in watermarking is for BMP images. A popular scheme of fragile watermarking for BMP images was described in [2]. It collaborated a hash function (MD5) and blocks of image. Bits of

watermark were embedded into LSBs of pixels of the blocks. Unfortunately, the BMP images is rarely used in *World Wide Web* because of their large size. GIF and JPEG images were used widely in Internet. In this paper we focus watermarking on GIF images.

GIF (*Graphics Interchange Format*) image is a kind of the indexed image. GIF was introduced by CompuServe in 1987 and come into widespread usage on the World Wide Web because of its wide support and portability. A GIF image uses a palette of up to 256 colors from the 24-bit RGB color space with values in the range [0,1]. The pixel values represent index to a palette row.

GIF format supports animaton of images, we call it the animated GIF images. The animated GIF images consist of a number of frames. Each frame was displayed in succession like a video. The animated GIF images is usually used for displaying cartoon, funny images, or other interesting images.

Most of fragile watermarking research on the GIF images are only for still image (still image is a single image). Also, the research on the GIF images are still a few. Two of them were described clearly in [3] and [4]. However, none of fragile watermarking research have been done for animated GIF images. Therefore, in this paper we propose a fragile watermarking scheme for the animated GIF images. In this scheme, we need not to derive a new watermarking algorithm for animated GIF images, but we reuse an existing steganography algorithm for the GIF (still) images.

Tel.: (+62) 22-2502260

Email: rinaldi.munir@itb.ac.id

A simple steganography algorithm for the GIF images is EzStego. EzStego algorithm has been proposed by Machado [5]. EzStego has two main process, embedding process and extraction process. In embedding process, bits of the message are embedded to LSBs of indices pointing to the palette. To minimize color degradation due to changes in the indices, at first the palette is sorted so that the difference between two adjacent color is minimized. EzStego embeds message into the LSBs of indices pointing to the sorted palette. In extraction process, bits of the embedded message can be recovered easily by extracting LSBs of indices of the sorted palette. The simple algorithm makes embedding and extraction process can be performed quickly with minimal visual degradation.

EzStego is used for hiding information in GIF images only, it is never used for watermarking of the GIF images, especially for animated images. Therefore, as mentioned above, in this paper, we propose a fragile watermarking scheme for animated GIF images which is based on EzStego. The watermark, which is a binary image, is inserted into frames of GIF image based on EzStego embedding scheme. For increasing security, the watermark is encrypted with the random bits before embedding.

This paper is organized into five sections. The first section is introduction. The second section will review some study of literatures such as GIF images and the EzStego algorithm. In the third section, we explain a fragile watermarking scheme for the animated GIF images based on EzStego algorithm. The fourth section describes the experiments and discuss the results. Finally, in last section we give conclusion and suggest future works..

2. Literature Studies

In this section, we will describe some reviews of GIF images and EzStego algorithm. “

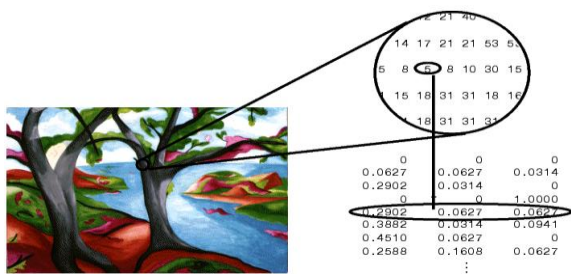


Fig.1. GIF image structure (Source: Matlab)

2.1 Animated GIF Images

A GIF image consists of a color palette and a matrix which entries (pixel values) refer to the palette row. Color of the pixel is combination of each channel red (*R*), green (*G*), and blue (*B*) in the palette row. Fig.

1 shows model of a GIF image, the pixel value 5 represents the fifth row of the palette (*R* = 0.2902, *G* = 0.0627, *B* = 0.0627). The color depth of the GIF images is up to 256 colors, therefore the GIF images are suitable for human-made graphics such as cartoon, animation.

The animated GIF images consist of a number of frames, each frame may has independent palette itself. Fig. 2 shows the six frames of 48 frames of an animated GIF image (*walk.GIF*). Every animated GIF image has a property that is called “delay time” in hundredth of seconds. It specifies delay every frame. For example, an animated GIF image contains 40 frames, with a 0.03 second delay specified between each frame. It means the animated GIF has runtime of 1.2 seconds per loop.

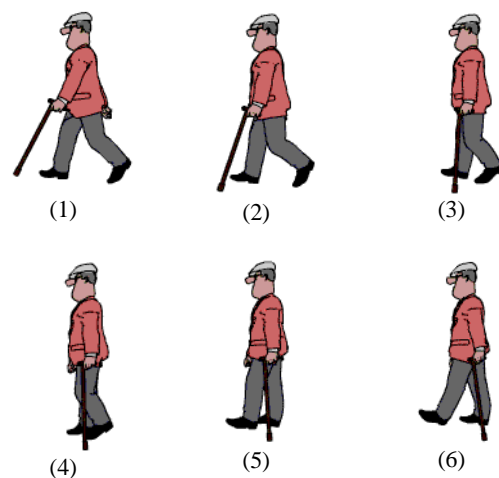


Fig. 2. The first six frames of an animated GIF image (*walk.gif*)

2.2 EzStego

The EzStego algorithm is very simple. The original EzStego is a sequential embedding type of stego system. It means that bits of the message are embedded sequentially in the LSBs of the pixels values. No key required for embedding and extraction the message

We resume the steps of embedding and extraction message in EzStego algorithm as follow:

Embedding Process

1. Sort the palette of the original image by distance between color of the pixels. The distance between the color (*R*₁, *G*₁, *B*₁) dan (*R*₂, *G*₂, *B*₂) is calculated by Euclidean distance:

$$d = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2} \quad (1)$$

2. Assign the new index of the sorted palette by numbering 0, 1, 2, ... etc.

3. Replace the LSB of index of the sorted palette by bits of message. Finally we get a stego-image.

Extraction Process

1. Sort the palette of the stego-image by distance between color of the pixels by using Eq. (1).
2. Assign the new index of the sorted palette by numbering 0, 1, 2, ... etc.
3. Extract the LSB of the index of the sorted of palette. Finally we get the original message.

Any message of any type can be embedded into the image by EzStego. We can embed text, image, etc. Therefore, EzStego can be used to embed a watermark such as logo or other binary image. This is opportunity to apply EzStego algorithm for a proposed fragile watermarking scheme described below.

3. A Proposed Scheme

Fragile watermarking techniques has two main process: embedding and extraction. Fig. 3 shows block diagram of embedding and extraction.

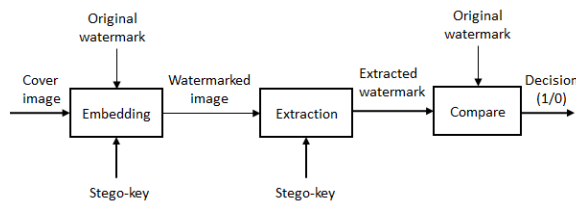


Fig. 3. Block diagram of fragile watermarking

If we want to embed a watermark into an animated GIF image, then there are two options. The first option is embedding all of frames with the same watermark. The second option is embedding each frame with the different watermarks randomly. We choose the first option because of its simplicity. The watermark is a binary image such as logo or something like that. Because of the original EzStego has no key for embedding and extraction, therefore for increasing security the watermark is encrypted before embedding. The encryption key(s) behave as the secret key.

3.1 Watermark Embedding

Fig. 4 shows the block diagram of watermark embedding. Suppose the image of size $M \times N$ and the watermark of size $m \times n$. Usually the watermark size is smaller than the host image (GIF image) size. In order to the modification in the GIF image can be detected until pixel level, we must duplicate the watermark ($m \times n$) by copying and pasting repeatedly until it has the same size with host image ($M \times N$).

For increasing security, the resulted watermark is encrypted with the random bits.

Finally, the encrypted watermark is embedded into each frame of the animated GIF image based on EzStego embedding scheme. The result is a watermarked animated GIF image.

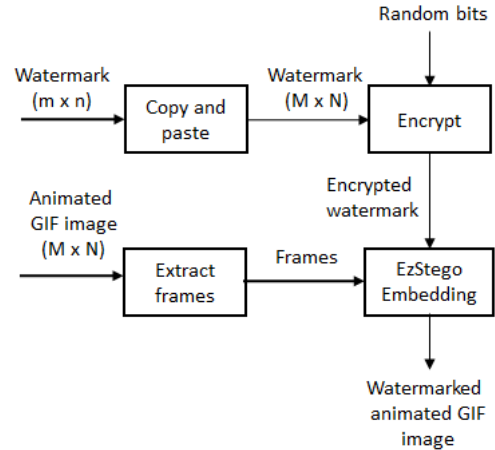


Fig 4. Block diagram of watermark embedding

3.2 Watermark Extraction and Comparing

The watermark is extracted for proving authentication of the watermarked image. Fig. 5 shows the block diagram of watermark extraction. The watermark is extracted from the watermarked animated GIF image by EzStego extraction scheme. The result is an encrypted watermark. We decrypt it with a random bits. Finally, we compare between decrypted watermark and original watermark to decide authentication of the image. If they are same then we conclude that the image is authentic, otherwise the image is not authentic anymore.

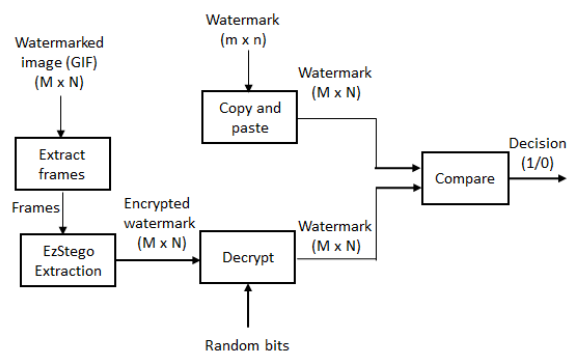


Fig 5. Block diagram of watermark extraction.

4. Implementation, Experiment and Results

The proposed scheme has been implemented by using MATLAB. We have modified and programmed the

EzStego algorithm so that it can be used to embed a watermark into frames of an animated GIF image. Next, we have performed some experiments to get performances of the scheme to some typical attacks. In this experiment we used an animated GIF image as host image which has six frames of size 300×200 pixel ('jogging.gif') with delay time 0.1, see Fig.6. The watermark is a 'ganeca' logo which duplicated a number of times in order to result the same size with the host image, see Fig.7.

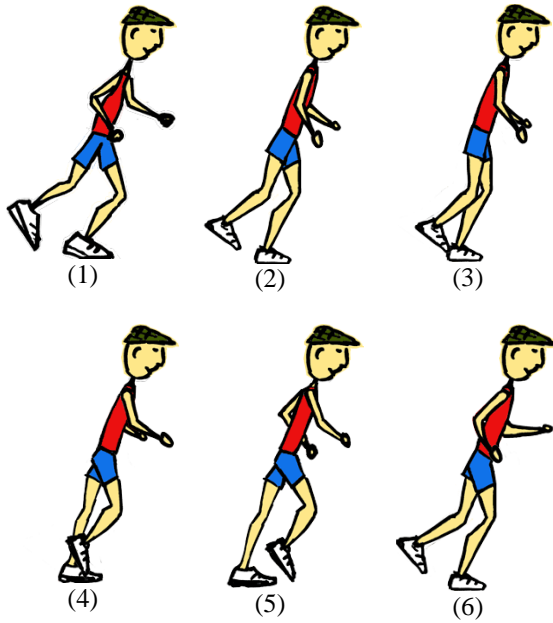


Fig. 6. Six frames of an animated GIF image (jogging.gif)

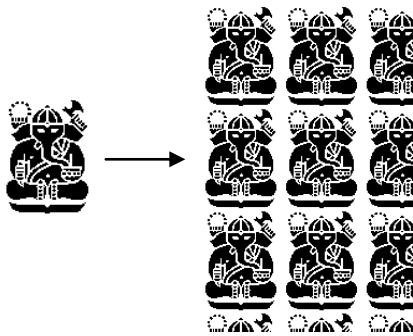


Fig. 7. Left: an original watermark. Right: a duplicated watermark a number of times

Before the watermark was embedded into the host image, the watermark was encrypted with the random bits. Table 1 shows the frames of the watermarked GIF image. PSNRs of each frame are displayed in right side. We can see that visually the watermarked frames is very similar with their cover frames. The

watermarked GIF image can be displayed perfectly as fine as the cover image.


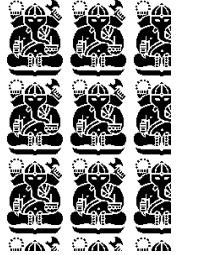

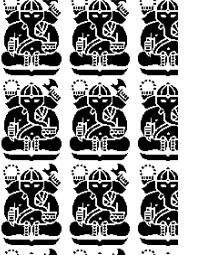
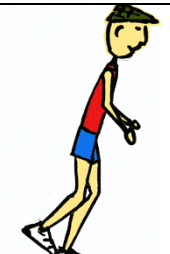
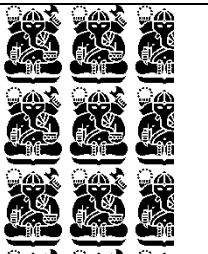
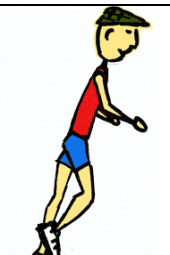
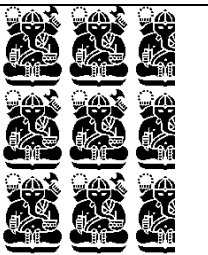

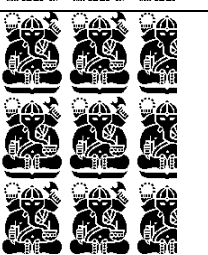
Table 1. The frames of the animated GIF image

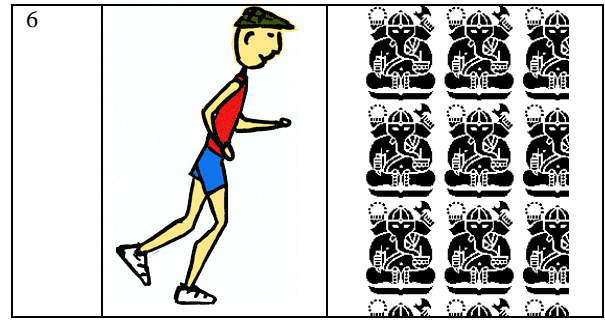
Frame	Cover frame	Watermarked frame	PSNR
1			78.6228 dB
2			79.3291 dB
3			79.4453 dB
4			79.5033 dB
5			79.2489 dB
6			79.3955 dB

To test wheater the animated GIF image is not modified, we only need to extract the embedded watermark from the image and then compare it

(visually or bit-per-bit comparison) with the original watermark. If both watermark are same exactly, we conclude the animated GIF image is still authentic, not modified..Fig. 9 shows the extracted watermark from the watermarked animated GIF image. The extracted watermarks are same exactly with the original watermark, therefore we conclude that the image have not been altered.

Table 2. The extracted watermark of each frames

Frame	Watermarked frame	Extracted watermark
1		
2		
3		
4		
5		



Next we tried to attack the watermarked GIF image by adding a ball in front of the runner. Then, we extracted the watermark from it, and we got the watermark was broken (see Fig. 8). We found shape such as a circle in the watermark. The circle shape didn't exist in the original watermark. It was a ball in the modified watermarked GIF image. Therefore, we conclude that the animated GIF image has been altered. Also, by additional process, we could identify part of object in the image which has been altered.

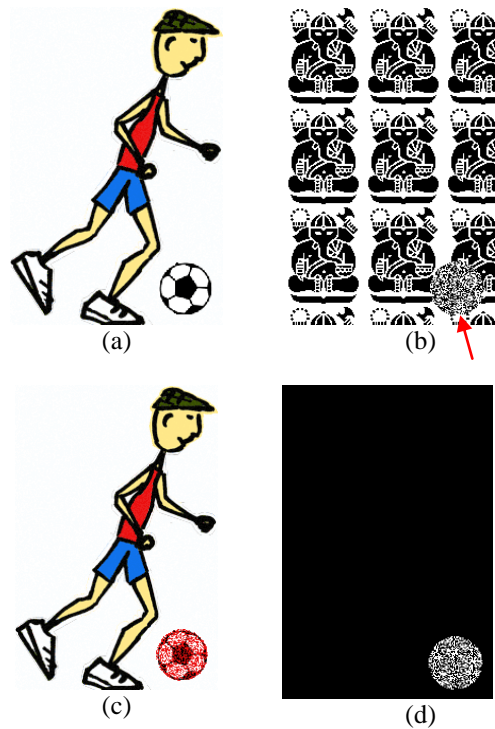


Fig 8. The first attack. (a) the watermarked GIF image has been altered by adding ball; (b) the extracted watermark contains object that has been altered. (c) identification of altered object in the watermarked image. (d) part of object which has been altered.

In the second attack, we changed color of the pants from blue to white. When the watermark was

extracted from the modified image, we found that the watermark was broken. Immediately we found part of the image which has been modified (Fig. 9).

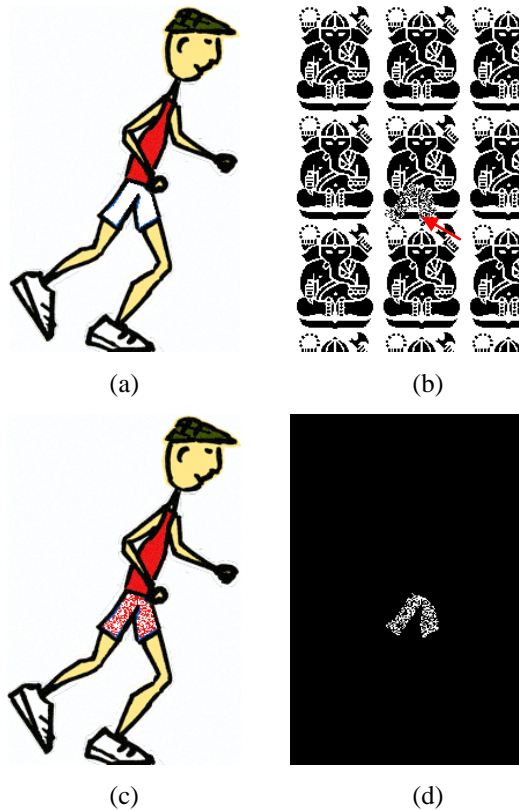


Fig 9. The second attack. (a) the watermarked GIF image has been modified by changing color of the pants from blue to white; (b) the extracted watermark contains object that has been modified. (c) identification of modified object in the watermarked image. (d) part of object which has been modified;

5. Conclusions

In this paper we have proposed a fragile watermarking scheme based on EzStego algorithm for the animated GIF images. The scheme can be used to verify authenticity of the image by comparing the original watermark and the extracted watermark. Based on some attacks to the watermarked GIF image, we can identify parts of the image which has been altered or modified. In the

future works, we can develop the proposed scheme for another animated image format.

References

- [1] Eugene T. Lin and Edward J. Delp, A Review of Fragile Image Watermarks, Proceedings of the Multimedia and Security (ACM Multimedia'99), pp. 25-29, 1999.
- [2] P.W. Wong, A Watermark for Image Integrity and Ownership Verification, Proceeding of The IS & T PIC Conference, 1999.
- [3] M.A. Hassan and S.A.M. Gillani, A Fragile Watermarking Scheme for Color Image Authentication, World Academy of Science, Engineering and Technology 19, 2006, pp. 39-42.
- [4] Chang, C., Lin, P. A Color Image Authentication Method Using Partitioned Palette and Morphological Operations, Journal IEICE – Transaction on Information and Systems, Vol. E91-D Issu1 1, January 2008, pp. 54-61.
- [5] R. Machado, EZStego, <http://www.stego.com>