Journal of Information Security, 2016, 7, 1-13
Published Online January 2016 in SciRes. http://www.scirp.org/journal/jis
http://dx.doi.org/10.4236/jis.2016.71001

Authentication Method Using a Discrete Wavelet Transform for a Digital Moving Image

Ren Fujii, Yasunari Yoshitomi, Taro Asada, Masayoshi Tabuse

Graduate School of Life and Environmental Sciences, Kyoto Prefectural University, Kyoto, Japan
Email: yoshitomi@kpu.ac.jp

Received 8 January 2016; accepted 24 January 2016; published 28 January 2016

Abstract

Recently, several digital watermarking techniques have been proposed for hiding data in the frequency domain of moving image files to protect their copyrights. However, in order to detect the water marking sufficiently after heavy compression, it is necessary to insert the watermarking with strong intensity into a moving image, and this results in visible deterioration of the moving image. We previously proposed an authentication method using a discrete wavelet transform for a digital static image file. In contrast to digital watermarking, no additional information is inserted into the original static image in the previously proposed method, and the image is authenticated by features extracted by the wavelet transform and characteristic coding. In the present study, we developed an authentication method for a moving image by using the previously proposed method for astatic image and a newly proposed method for selecting several frames in the moving image. No additional information is inserted into the original moving image by the newly proposed method or into the original static image by the previously proposed method. The experimental results show that the proposed method has a high tolerance of authentication to both compressions and vicious attacks.

Keywords

Authentication, Moving Image, Copyright Protection, Tolerance to Compression, Wavelet Transforms

1. Introduction

Recent progress in digital media technology and distribution systems, such as the Internet and cellular phones, has

How to cite this paper: Fujii, R., Yoshitomi, Y., Asada, T. and Tabuse, M. (2016) Authentication Method Using a Discrete Wavelet Transform for a Digital Moving Image. *Journal of Information Security*, **7**, 1-13.
http://dx.doi.org/10.4236/jis.2016.71001

enabled consumers to easily access, copy, and modify digital content, which includes electric documents, images, audio, and video. Therefore, techniques to protect the copyrights for digital data and to prevent unauthorized duplication or tampering are urgently needed.

Digital watermarking (DW) is a promising method for copyright protection of digital data. Several studies have developed a method in which 1) the DW can be sufficiently extracted from the watermarked digital data, even after compression, and 2) the quality of the digital data remains high after the DW is embedded [1]-[8]. However, a tradeoff generally exists between these two properties. Two important properties of the DW for digital data are imperceptibility of DW-introduced distortion, and robustness to signal processing methods, such as compressions and vicious attacks. However, the data rate and complexity of the DW have attracted attention when discussing the DW performance.

For overcoming the issue of performance, we previously developed authentication methods for digital audio [9] and a static image [10] without inserting a DW into them by using a discrete wavelet transform (DWT). In contrast to the DW, no additional information is inserted into the original digital data by the previous method [10], and the digital data are authenticated by features extracted by the DWT and characteristic coding [10].

In the present study, we developed an authentication method for a digital moving image to protect the copy rights by using the previously proposed method [10] for a static image and our newly proposed method for selecting several frames in the moving image. No additional information is inserted into the original moving image by the proposed method or into the original static image by the previously proposed method [10]. The digital moving image is authenticated by features extracted by the DWT and characteristic coding of the proposed method. This paper presents the method and also an analysis of its performance, including the tolerance of authentication to both compressions and vicious attacks.

2. Wavelet Transform

In the present study, we use the authentication method for a static image [10] to authenticate a moving image. First, we briefly describe the procedure of the DWT used in the authentication for a static image.

The original image $F(u,v)$, which is used as the level-0 wavelet decomposition coefficient sequence $s_{u,v}^{(0)}$, is decomposed into the multi-resolution representation (MRR) and the coarsest approximation by repeatedly applying the DWT. The wavelet decomposition coefficient matrix $s_{u,v}^{(j)}$ at level *j* is decomposed into four wavelet decomposition coefficient matrices at the level $j+1$ by using (1), (2), (3), and (4).

$$s_{u,v}^{(j+1)} = \sum_{l} \sum_{k} \overline{p_{k-2u}}\, \overline{p_{l-2v}}\, s_{k,l}^{(j)} \tag{1}$$

$$w_{u,v}^{(j+1,h)} = \sum_{l} \sum_{k} \overline{p_{k-2u}}\, \overline{q_{l-2v}}\, s_{k,l}^{(j)} \tag{2}$$

$$w_{u,v}^{(j+1,v)} = \sum_{l} \sum_{k} \overline{q_{k-2u}}\, \overline{p_{l-2v}}\, s_{k,l}^{(j)} \tag{3}$$

$$w_{u,v}^{(j+1,d)} = \sum_{l} \sum_{k} \overline{q_{k-2u}}\, \overline{q_{l-2v}}\, s_{k,l}^{(j)}, \tag{4}$$

where *u* and *v* denote the horizontal and vertical directions, respectively, and p_k and q_k denote the scaling and wavelet sequences, respectively. In addition, $w_{u,v}^{(j+1,h)}$ denotes the development coefficient obtained by operating the scaling function in the direction of the horizontal axis and the wavelet in the vertical axis; $w_{u,v}^{(j+1,v)}$ denotes the development coefficient obtained by operating the wavelet in the direction of the horizontal axis and the scaling function in the vertical axis, and; $w_{u,v}^{(j+1,d)}$ denotes the development coefficient obtained by operating the wavelet in the direction of both the horizontal and vertical axes. The development coefficients at level *J* are obtained by repeatedly using (1), (2), (3), and (4) from $j=0$ to $j=J-1$. In this method, the image must be a square composed of $2^L \times 2^L \left(J \le L\right)$ pixels. For example, the image is decomposed by the DWT to level 3, as shown in **Figure 1**, where the original image $s_{u,v}^{(0)}$ is decomposed into $s_{u,v}^{(1)}$ expressed by 1LL, $w_{u,v}^{(1,v)}$ is expressed by 1HL, $w_{u,v}^{(1,h)}$ is expressed by 1LH, and $w_{u,v}^{(1,d)}$ is expressed by 1HH. Here, H and L denote the high- and low-frequency components, respectively.

3. Authentication Algorithm for Static Image

It is known that the histogram of wavelet coefficients of each domain at MRR parts has a distribution in which

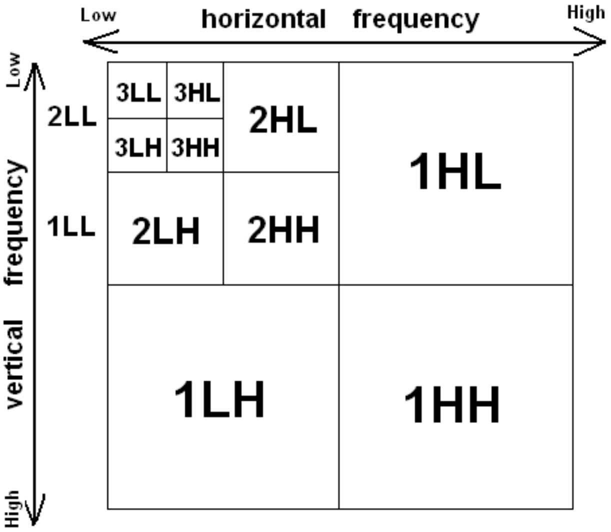

Figure 1. Mallat division [8].

the center is almost 0 when the DWT is performed on a natural image (**Figure 2**) [4]. Exploiting this phenomenon, we developed an authentication method for a static image [10].

3.1. Coding

For coding the image, we obtain the histogram of wavelet coefficients V at the selected level of an MRR sequence (**Figure 3**). As with the DW techniques for images [10] and digital audio [9], we set the following coding parameters.

The values of *Th*(minus) and *Th*(plus) in **Figure 3** are chosen such that the non-positive wavelet coefficients (S_m in total frequency) are equally divided into two groups by *Th*(minus), and the positive wavelet coefficients (S_p in total frequency) are equally divided into two groups by *Th*(plus). Next, the values of $T1$, $T2$, $T3$, and $T4$, which are the parameters for controlling the authentication precision, are chosen to satisfy the following conditions:

1) $T1 < Th(\text{minus}) < T2 < 0 < T3 < Th(\text{plus}) < T4$.

2) The value of S_{T1}, which is the number of wavelet coefficients in $\left(T1, Th(\text{minus})\right)$, is equal to S_{T2}, which is the number of wavelet coefficients in $\left[Th(\text{minus}), T2\right)$, *i.e.*, $S_{T1} = S_{T2}$.

3) The value of S_{T3}, the number of wavelet coefficients in $\left(T3, Th(\text{plus})\right]$, is equal to S_{T4}, the number of wavelet coefficients in $\left(Th(\text{plus}), T4\right)$, *i.e.*, $S_{T3} = S_{T4}$.

4) $S_{T1}/S_m = S_{T3}/S_p$.

In the present study, the values of both S_{T1}/S_m and S_{T3}/S_p are set to 0.25, which was determined experimentally.

To prepare the coding for authentication, the procedure separates the wavelet coefficients V of an MRR sequence into five sets (hereinafter referred to as *A*, *B*, *C*, *D*, and *E*), as shown in **Figure 4**, by using the following criteria:

- $A = \left\{V \mid V \in V^{SC}, V \le T1\right\}$,
- $B = \left\{V \mid V \in V^{SC}, T1 < V < T2\right\}$,
- $C = \left\{V \mid V \in V^{SC}, T2 \le V \le T3\right\}$,
- $D = \left\{V \mid V \in V^{SC}, T3 < V < T4\right\}$,
- $E = \left\{V \mid V \in V^{SC}, T4 \le V\right\}$,

where V^{sc} is the set of wavelet coefficients in the original image file.

The wavelet coefficients of an MRR sequence are coded according to the following rules, in which V_i denotes

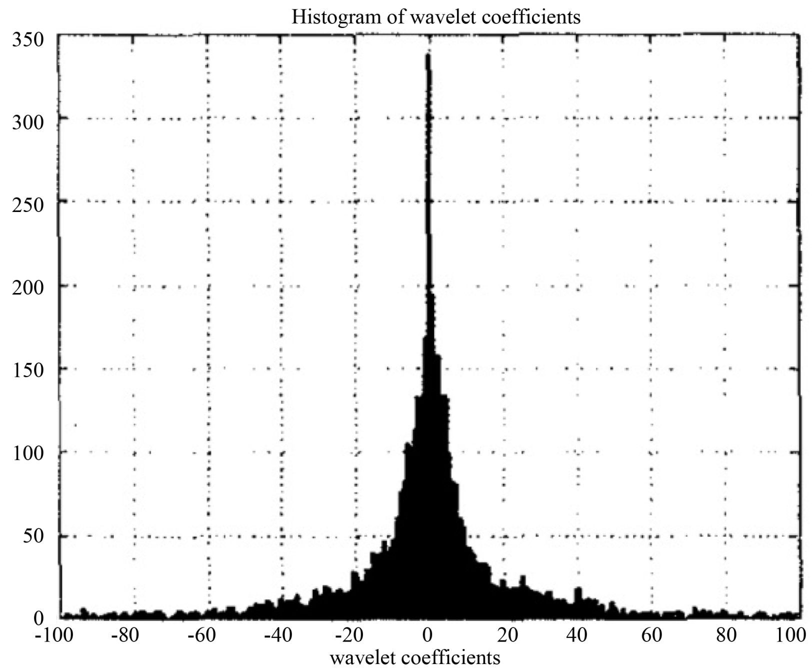

Figure 2. Histogram of the wavelet coefficients [4].

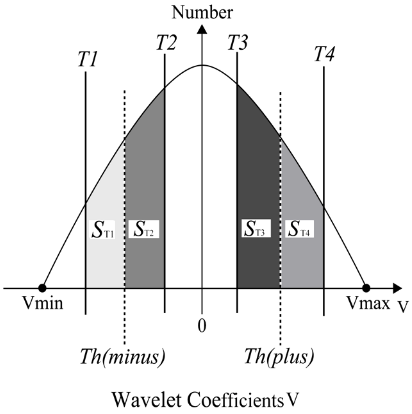

Figure 3. Schematic diagram of the histogram of MRR wavelet coefficients [9].

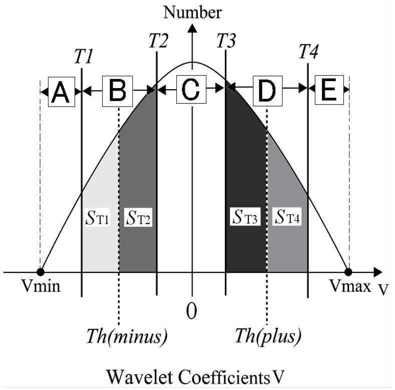

Figure 4. Five sets (*A*, *B*, *C*, *D*, and *E*) of the histogram of wavelet coefficients *V* of an MRR sequence for the assignment of a bit [9].

one of the wavelet coefficients:

When $V_i \in C$, flag f_i is set to 1, and bit b_i is set to 0.

When $V_i \in (A \cup E)$, flag f_i is set to 1, and bit b_i is set to 1.

When $V_i \in (B \cup D)$, flag f_i is set to 0, and bit b_i is set to 0.5.

For authentication of the digital image, we use a code $\boldsymbol{C}$ (hereinafter referred to as an original code), which is the sequence of b_i defined above. For coding and authentication, we assign a sequence number and a flag for each wavelet coefficient. The flag $f_i = 1$ for V_i means that V_i is assigned a bit ($b_i = 0$ or 1) for the coding. The flag $f_i = 0$ for V_i means that V_i is not assigned a bit of 0 or 1: b_i is externally set to 0.5 as an arbitrary constant, and the value of b_i does not influence the performance of the proposed method described in Section 3.2. The coding with the exclusion of all V_i belonging to sets *B* and *D*, where the magnitudes of V_i are intermediate, is a novel feature of our previously reported studies [9] [10].

3.2. Authentication

We authenticate not only an original digital image file but also a signal-processed version. Compression, which is one example of signal processing, is often applied to a digital image for the purposes of distribution via the Internet or for saving on a computer. Through the same coding procedure described in Section 3.1, we applied the DWT to a digital image and obtained a histogram of wavelet coefficients V' at the same level of the DWT as that coded for the original image file, as described in Section 3.1. Then, we set the authentication parameters as follows.

The values of $Th'(\text{minus})$ and $Th'(\text{plus})$ (**Figure 5**) are chosen such that the non-positive wavelet coefficients (S'_m in total frequency) are equally divided into two groups by $Th'(\text{minus})$, and the positive wavelet coefficients (S'_p in total frequency) are equally divided into two groups by $Th'(\text{plus})$. Next, the values of $T1'$, $T2'$, $T3'$, and $T4'$, the parameters for controlling the authentication precision, are chosen to satisfy the following conditions:

1) $T1' < Th'(\text{minus}) < T2' < 0 < T3' < Th'(\text{plus}) < T4'$.

2) The value of S'_{T1}, the number of wavelet coefficients in $(T1', Th'(\text{minus}))$, is equal to S'_{T2}, the number of wavelet coefficients in $[Th'(\text{minus}), T2')$, *i.e.*, $S'_{T1} = S'_{T2}$.

3) The value of S'_{T3}, the number of wavelet coefficients in $(T3', Th'(\text{plus})]$, is equal to S'_{T4}, the number of wavelet coefficients in $(Th'(\text{plus}), T4')$, *i.e.*, $S'_{T3} = S'_{T4}$.

4) $S'_{T1}/S'_m = S'_{T3}/S'_p$.

In the present study, the values of both S'_{T1}/S'_m and S'_{T3}/S'_p are set to 0.25, which is the same setting used for coding of the original image file, as described in Section 3.1.

In preparation of the coding for authentication, the procedure separates the wavelet coefficients V' of an MRR sequence into three sets (hereinafter referred to as *F*, *G*, and *H*), shown in **Figure 5**, by using the following criteria:

- $F = \{V' \mid V' \in V'^{AC}, V' < Th'(\text{minus})\}$,
- $G = \{V' \mid V' \in V'^{AC}, Th'(\text{minus}) \le V' \le Th'(\text{plus})\}$,
- $H = \{V' \mid V' \in V'^{AC}, Th'(\text{plus}) < V'\}$,

where V'^{AC} is the set of wavelet coefficients of the target image file for making the code for authentication.

The wavelet coefficients of an MRR sequence are coded according to the following rules, in which V'_i denotes one of the wavelet coefficients:

When $f_i = 1$ and $V'_i \in G$, bit b'_i is set to 0.

When $f_i = 1$ and $V'_i \in (F \cup H)$, bit b'_i is set to be 1.

When $f_i = 0$, bit b'_i is set to 0.5.

When $f_i = 0$, b'_i is externally set to 0.5 as an arbitrary constant, and the value of b_i does not influence the performance of the proposed method described below.

For authentication of the digital image, we use the code $\boldsymbol{C}'$ (hereinafter referred to as an authentication code), which is the sequence of b'_i defined above. The authentication ratio *AR* (%) is defined by the following:

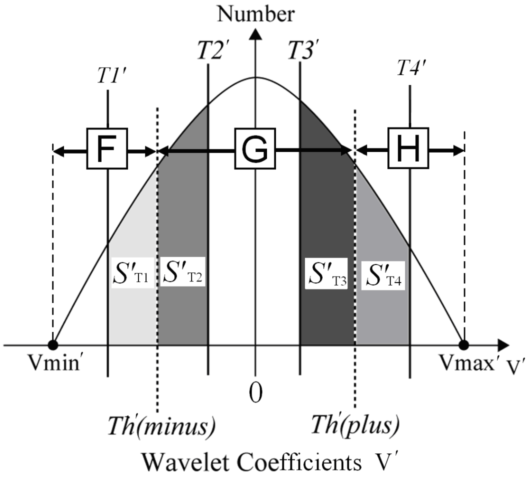

Figure 5. Three sets (*F*, *G*, and *H*), indicated on the histogram, of MRR wavelet coefficients used for authentication [9].

$$AR = \frac{100\sum_{i=1}^{N} f_i\left(1-\left|b_i - b_i'\right|\right)}{\sum_{i=1}^{N} f_i}, \tag{5}$$

where *N* is the number of wavelet coefficients assigned flags in the coding for the original image file, described in Section 3.1. According to (5), the values of neither b_i nor b_i' influence the value of AR in the case that $f_i = 0$, which means the corresponding V_i is not assigned a bit of 0 or 1 in the coding for the original image file.

To use the proposed method, we need to store flags f_i and the original code $\boldsymbol{C}$ of each image file having a copyright we want to protect. In calculating (5) for the authentication of an original image file, we do not use an original image file; instead, we use the flags f_i and the original code $\boldsymbol{C}$ of the original image file.

4. Authentication Algorithm for Moving Image

4.1. Coding

We obtain 10 pieces of coding per moving image, which is assumed to have 13 or more frames, in a database as follows.

Step 1: Calculate frame No. k_B, which is the smallest integer not less than $0.1 \times N_{total}$, and calculate frame No. k_E, which is the largest integer not greater than $0.9 \times N_{total}$. Here, N_{total} is the total number of frames of the moving image. Then, go to Step 2.

Step 2: Output the images of frame Nos. k_B and $k_B + 1$ as BMP files. Calculate the total sum $S[1]$ of the squares with a gray level difference at each pixel between frame Nos. k_B and $k_B + 1$. Store the gray levels of frame No. $k_B + 1$ in an array, A_1. Give the initial condition as $C[1] := k_B + 1$, $count := 1$, $k := k_B + 2$. Then, go to Step 3.

Step 3: If $k = k_E + 2$, go to Step 8. Otherwise, go to Step 4.

Step 4: Overwrite the image of frame No. *k* on that of $k_B - 2$ as the BMP file. Store the gray levels of frame No. *k* in array *B*. Calculate the total sum *T* of the squares with a gray level difference at each pixel between frame Nos. $k - 1$ and *k*. Update the value of *k* as $k := k + 1$. Then, go to Step 5.

Step 5: If $count < 10$, update the value of $count$ as $count := count + 1$ and go to Step 6. Otherwise, go to Step 7.

Step 6: Overwrite the array as $A_{count} := B$. Set the values as $S[count] := T$, $C[count] := k - 1$. Then, go to Step 4.

Step 7: Calculate $l = \arg\max_{i=1,2,\cdots,10} S[i]$. If $S[l] > T$, overwrite the array as $A_l := B$ and set the values as $S[l] := T$, $C[l] := k-1$. If $k = k_E + 1$, go to Step 8. Otherwise, go to Step 4.

Step 8: Output the arrays $A_i\,(i = 1, 2, \cdots, 10)$ as BMP files. Obtain the codes and flags used for selective coding by using the method described in Section 3.1. Output the codes, the flags, $C[i]\,(i = 1, 2, \cdots, 10)$, and $S[i]\,(i = 1, 2, \cdots, 10)$. Then finish.

According to our experience, sometimes several consecutive frames including the first or the last frame of a moving image have black for almost all pixels or white for almost all pixels. If such a frame is selected for coding despite that it is neither a representative nor a unique frame for the moving image, the code of the frame might be very detrimental to the authentication of the moving image. Therefore, in Step 1, as the objects for coding, we exclude the first and last 10% of the frames.

4.2. Authentication

The selection of frames is performed for test moving images with the same method described in Section 4.1. Assuming that the codes and the flags for selective coding for selected frames of moving images in a database were obtained by using the methods described in Section 4.1, the authentication ratios of each selected frame of a test moving image to each selected frame of the moving image in the database are calculated with the method described in Section 3.2. The authentication ratio of a test moving image to a moving image in a database is defined as the highest among all values of authentication ratios of selected frames of the test moving image to selected frames of the moving image in the database. Then, the moving image having the highest authentication ratio among all moving images in the database is found.

5. Experiment

In this section, we describe computer experiments and the results for evaluating the performance of the proposed method.

5.1. Method

The experiment was performed in the following computational environment: the personal computer was a DELL OPTIPLEX3020 (CPU: Intel(R) Core(TM) i5-4570 3.20 GHz, memory: 4.0 GB); the OS was Microsoft Windows 7 Professional; the development language was Microsoft Visual C++6.0.

The process of the experiment was as follows. Obtain 100 sets of codes consisting of ten codes for the Y components, which are obtained by the conversion from RGB components into YCrCb components per moving image from the original 100 MPEG-1 files of MUSCLE-VCD-2007 [11]. Use these sets for the authentication among themselves. Convert three target files, which are selected according to the recording time of the shortest (No. 31: 17 s), the closest to the average recording time (No. 49: 34 min 56 s), and the longest (No. 47: 1 hour 55 min 46 s) among the moving images in the database into MPEG-2 and MPEG-4, followed by the authentication to the converted files. Investigate the tolerance of the authentication ratios to the frame rate of three target files after converting their file format from MPEG-1 with the frame rate of 25 fps to MPEG-4 with the frame rate of 5 to 60 fps. Investigate the tolerance of the authentication ratios to the compression of three selected files by using libjpeg [12] to change the values of the quality of the images obtained from the selected frames and by using FFmpeg [13] to generate moving images with the images modified by libjpeg, Investigate the authentication ratios to the frames whose frame distance from a selected frame of a target file is within 5.

We used FFmpeg to output a BMP file having 24 bits as the gray level and 256 × 256 pixels from the moving image. For the DWT, we use Daubechies wavelets. The 4LH components obtained from the DWT up to Level 4 were chosen for coding and authentication, based on the analysis in the preliminary experiments. **Figure 6** shows ten image examples consisting of one frame per moving image.

Several consecutive frames having black for almost all pixels or white for almost all pixels are sometimes inserted in a moving image because they can be useful for scene transitions. If such a frame is selected for coding despite that it is neither a representative nor a unique frame for the moving image, the code of the frame might be very harmful for authentication of the moving image. Therefore, in addition to the exclusion of frames as the objects of coding described in Step 1 in Section 4.1, we used only the frames having an average gray level of α

Figure 6. Ten image examples consisting of one frame per moving image of the specified number.

to 250 for the Y component, which has a gray level range of 0 to 255. The value of α was decided to be 11.1 by trial and error. However, when we did not obtain ten frames with α set to 11.1, we changed the value of α to 5 and were able to obtain ten frames for coding.

5.2. Results and Discussion

5.2.1. Authentication for MPEG-1 Files

The purpose of authentication is to protect the copyrights on moving image data. In the ten moving images listed in **Table 1**, when the moving image file targeted for authentication was different from that used for making the code of the original moving image file, the average authentication ratio was 60.2% (more precisely, in the range 40.9% to 81.1%), which was much smaller than the authentications ratios when authenticating the same moving image file as the original moving image file (100% in all cases in this experiment; see **Table 1**). An authentication ratio of 50% corresponds to the value in the case that randomly generated bits are used for b_i and/or b_i' in (5). Accordingly, the proposed method distinguished the original moving image file from each of the other nine shown in **Table 1**. When all moving images in the database [11] were authenticated by using all moving images in the database [11], the average authentication ratios between each pair of different moving images was 66.0% (more precisely, in the range 34.1% to 94.1%), which was fairly smaller than the authentications ratios when authenticating the same moving image file as the original moving image file (100% in all cases in this experiment) [14]. In other words, we could identify all moving image files in the database [11] by using all moving image files in the database [11]. When authenticating the No. 59 moving image file with the codes of the No. 73 moving image file as the original moving image file, the authentication ratio was 94.1%. The high authentication ratio was caused by the pair of similar frames shown in **Figure 7** [14].

5.2.2. Robustness to Conversion into MPEG-2 or MPEG-4

When the three moving images of Nos. 31, 47, and 49, converted from MPEG-1 to MPEG-2 or MPEG-4, were authenticated by using 100 moving images (MPEG-1) in the database [11], the authentication ratios was between each pair of different moving images (in the range 43.9% to 79.5%), and were much smaller than the authentications ratios when authenticating the same moving image file as the original moving image file (100% in all cases in this experiment). Therefore, we could conclude that the proposed method has excellent robustness to conversion to MPEG-2 or MPEG-4. **Table 2** shows several authentication ratios from this experiment.

5.2.3. Robustness to Frame Rate

Table 3 shows the robustness of the proposed method according to the frame rate. In the cases of Nos. 31 and 47 moving image files, the authentication ratios were kept at 100% for 5 to 60 fps of the file authenticated when authenticating the moving image file with 5 to 60 fps by using the original moving image file with 25 fps. Moreover, in the case of the No. 49 moving image file, the authentication ratios were kept to 100% only for 20 and 25 fps of the file authenticated when authenticating in the same manner (**Table 3**). The frame rate of the moving image file could be changed to avoid the copyright protection by the proposed method; it was effectively

Table 1. Authentication ratios in 100 combinations of original moving images expressed by file No. for columns and by authentication targets expressed by file No. for rows.

File No.	3	24	25	31	61	69	70	81	94	100
3	100	61.4	56.8	62.9	55.2	55.8	60.6	47.7	63.6	72.0
24	64.4	100	54.5	60.6	50.0	48.5	62.9	51.5	68.2	65.9
25	61.4	56.1	100	57.6	50.0	50.0	70.5	44.7	61.4	63.6
31	62.1	61.4	56.1	100	55.9	50.8	62.9	40.9	70.5	71.2
61	59.1	56.1	50.8	58.3	100	77.5	76.5	56.8	62.9	53.0
69	57.6	51.5	59.1	51.5	78.5	100	75.8	81.1	56.8	49.2
70	66.7	58.3	68.2	65.9	69.8	72.5	100	50.0	70.5	69.7
81	51.5	56.8	51.5	44.7	57.5	81.1	50.0	100	50.0	41.7
94	65.9	68.9	65.2	73.5	60.2	50.7	67.4	46.2	100	72.0
100	72.0	67.4	61.4	68.9	55.2	47.1	72.0	40.9	67.4	100
										(%)

Table 2. Authentication ratios of target files with MPEG-2 or MPEG-4 formats.

Original file No. (format)	Target file No. & format					
	31		47		49	
	MPEG-2	MPEG-4	MPEG-2	MPEG-4	MPEG-2	MPEG-4
31 (MPEG-1)	100	100	58.3	59.8	65.9	56.8
47 (MPEG-1)	63.6	65.2	100	100	63.6	59.8
49 (MPEG-1)	62.9	63.6	59.1	59.8	100	100
						(%)

Table 3. Tolerance of authentication ratio to frame rate of target file having a MPEG-4 format.

Original file No. (fps)	Fps of target file No. 31							
	60	50	30	25	20	15	10	5
31 (25)	100	100	100	100	100	100	100	100
47 (25)	65.9	65.9	65.9	63.6	65.9	65.9	65.2	63.6
49 (25)	55.3	55.3	56.1	53.0	54.5	54.5	53.8	53.8

Original file No. (fps)	Fps of target file No. 47							
	60	50	30	25	20	15	10	5
31 (25)	59.1	59.1	59.1	59.1	59.1	59.1	59.1	59.1
47 (25)	100	100	100	100	100	100	100	100
49 (25)	59.1	60.6	60.6	60.6	59.8	59.8	59.8	60.6

Original file No. (fps)	Fps of target file No. 49							
	60	50	30	25	20	15	10	5
31 (25)	56.8	54.5	55.3	53.8	53.0	53.0	62.9	54.5
47 (25)	52.3	47.0	47.7	56.1	60.6	47.7	59.8	52.3
49 (25)	75.8	79.5	72.7	100	100	98.5	69.7	98.5
								(%)

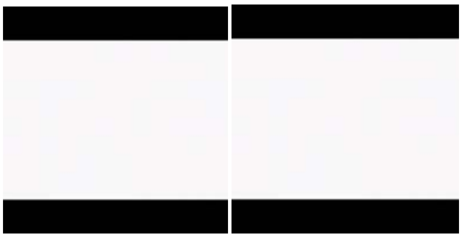

Figure 7. Images of a pair of similar frames; left figure: selected from No. 73 moving image, right figure: selected from No. 59 moving image [14].

more beneficial to use the higher authentication ratio between the authentication ratio obtained by using the raw frame rate of the authenticated file and that obtained by converting the frame rate of the authenticated file into the same value as that in the database (**Tables 3-5**).

5.2.4. Robustness to Compression Rate

Figure 8 illustrates the flow for generating new MPEG-1 files from an original MPEG-1 file through JPEG files with several compression rates. When the moving image file targeted for authentication was different from that used for making the code of the original moving image file, the average authentication ratio was 68.9% (more precisely, in the range 62.9% to 75.0%), which was fairly smaller than the authentications ratios when authenticating the same moving image file as the original moving image file (100% in almost all cases with the JPEG quality of 0 to 100 in this experiment; see **Table 6**). The image obtained with very low JPEG quality had a very poor appearance (see the left figure in **Figure 9**). Therefore, we could conclude that the proposed method has excellent robustness to the compression rate when keeping a good image appearance.

5.2.5. Robustness to Frame Exclusion

Table 7 shows authentication ratios for frames near each selected frame. When authenticating the frame near the selected one in the same moving image file by using the code of the selected frame of the original moving image file, the authentication ratio was100% in most cases (**Table 7**). However, in some cases, the attack of frame extraction effectively avoided the copyright protection by using the proposed method (see the three cases of frames from the bottom of the No. 49 moving image file in **Table 7**). The three frames in the No. 49 moving image file were within continual frames under the conditions of both rapid rotation and fairly dark appearance. Therefore, the robustness of the proposed method to frame extraction for authentication might be effectively improved by

Table 4. Authentication ratios after the frame rate of target file No. 49 is changed twice to 25 (original file; MPEG-1) → [5 to 60] (MPEG-4) → 25 (MPEG-4) fps.

Original file No. (fps)	Fps of target file No. 49 before changed to 25							
	60	50	30	25	20	15	10	5
31 (25)	51.5	59.1	55.3	53	51.5	59.1	56.1	56.1
47 (25)	46.2	53.8	50.0	56.8	53.8	52.3	59.1	63.6
49 (25)	73.5	100	69.7	100	81.1	79.5	76.5	78.8
								(%)

Table 5. Highest authentication ratios of those in **Table 3** and **Table 4**.

Original file No. (fps)	Fps of target file No. 49 before changed to 25							
	60	50	30	25	20	15	10	5
31 (25)	56.8	59.1	55.3	53.8	53.0	59.1	62.9	56.1
47 (25)	52.3	53.8	50.0	56.8	60.6	52.3	59.8	63.6
49 (25)	73.8	100	72.7	100	100	98.5	76.5	98.5
								(%)

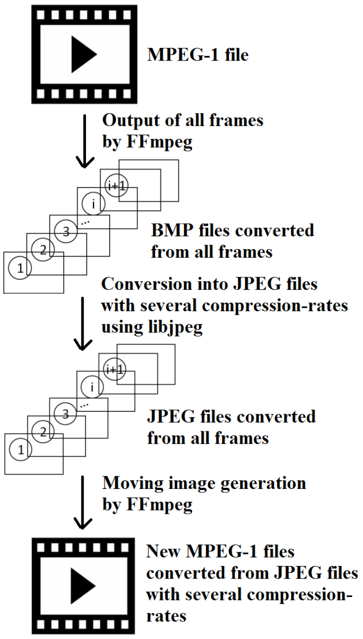
MPEG-1 file
Output of all frames by FFmpeg
1
2
3
...
i
i+1
BMP files converted from all frames
Conversion into JPEG files with several compression-rates using libjpeg
1
2
3
...
i
i+1
JPEG files converted from all frames
Moving image generation by FFmpeg
New MPEG-1 files converted from JPEG files with several compression-rates

Figure 8. Flow for generating new MPEG-1 files from an original MPEG-1 file through JPEG files, corresponding to all frames of an original MPEG-1 file. The JPEG files have several compression rates.

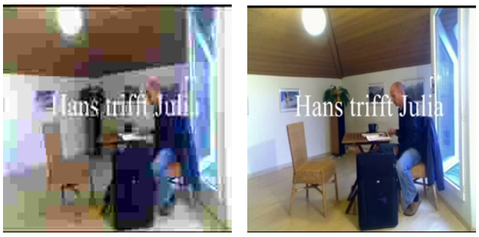

Figure 9. Example of compressed image of moving image No. 31; left figure: quality 5, right figure: quality 100.

Table 6. Tolerance of authentication ratios to compression of target file.

Original file No.	Quality value of JPG compression for each frame of target file No. 31						
	0	5	10	15	50	75	100
31	96.2	98.5	100	100	100	100	100
94	70.5	73.5	72.7	74.2	73.5	74.2	75.0
100	62.9	62.9	64.4	66.7	67.4	66.7	67.4

Original file No.	Quality value of JPG compression for each frame of target file No. 94						
	0	5	10	15	50	75	100
31	68.7	72.7	70.5	68.2	71.2	71.2	71.2
94	97.0	100	100	100	100	100	100
100	68.9	67.4	67.2	66.7	68.2	68.9	67.4

Original file No.	Quality value of JPG compression for each frame of target file No. 100						
	0	5	10	15	50	75	100
31	64.2	68.2	65.2	68.2	65.9	67.4	66.7
94	68.1	68.9	68.9	70.5	68.9	69.7	71.2
100	94.0	97.0	98.5	100	100	100	100
							(%)

Table 7. Authentication ratios for frames near each selected one.

Selected frame No.	Frame distance from selected frame of target file No. 31										
	−5	−4	−3	−2	−1	0	1	2	3	4	5
192	97	97	97	100	100	100	100	99.2	99.2	99.2	98.5
195	98.5	98.5	98.5	98.5	100	100	100	97.7	97.7	97.7	97.7
198	97	99.2	99.2	99.2	100	100	100	100	100	100	100
312	100	100	100	100	100	100	100	100	100	100	100
201	100	100	100	100	100	100	100	100	99.2	99.2	99.2
162	100	100	100	100	100	100	100	100	100	100	99.2
342	99.2	99.2	99.2	100	100	100	100	99.2	99.2	99.2	99.2
299	100	100	100	100	100	100	100	100	100	100	100
222	100	100	100	100	100	100	100	100	100	100	100
282	100	100	100	100	100	100	100	100	100	100	100

Selected frame No.	Frame distance from selected frame of target file No. 47										
	−5	−4	−3	−2	−1	0	1	2	3	4	5
139911	100	100	100	100	100	100	100	100	100	100	100
139926	100	100	100	100	100	100	100	100	100	100	100
139941	100	100	100	100	100	100	100	100	100	100	100
139956	100	100	100	100	100	100	100	100	100	100	100
139971	100	100	100	100	100	100	100	100	100	100	100
140736	100	100	100	100	100	100	100	100	100	100	100
140751	100	100	100	100	100	100	100	100	100	100	100
140766	100	100	100	100	100	100	100	100	100	100	100
140781	100	100	100	100	100	100	100	100	100	100	100
140781	100	100	100	100	100	100	100	100	100	100	100

Selected frame No.	Frame distance from selected frame of target file No. 49										
	−5	−4	−3	−2	−1	0	1	2	3	4	5
17178	78.8	78.8	81.1	100	100	100	78.8	77.3	78.0	77.3	77.3
17188	100	100	100	100	100	100	76.5	76.5	77.3	74.2	73.5
17158	87.1	87.9	84.8	100	100	100	83.3	84.8	85.6	90.9	91.7
17174	81.8	80.3	81.1	100	100	100	100	76.5	76.5	76.5	68.2
17071	100	100	100	100	100	100	92.4	91.7	90.9	90.2	87.9
17183	75.0	74.2	75.8	100	100	100	100	100	100	100	81.1
17168	72.0	72.0	72.7	100	100	100	100	100	100	75.8	73.5
38273	60.6	56.1	56.8	56.1	63.6	100	60.6	52.3	58.3	56.1	52.3
38275	68.9	62.1	56.8	57.6	61.4	100	52.3	49.2	51.5	52.3	53.0
25880	59.1	54.5	57.6	63.6	64.4	100	63.6	57.6	58.3	53.0	55.3
											(%)

selecting frames that do not have a dark appearance, which could be judged by using the information of its gray levels.

6. Conclusions

We have proposed an authentication method for a moving image by using our previously proposed method for a static image and a newly developed method for selecting several frames in the moving image. No additional information is inserted into the original moving image by the proposed method or in the previously proposed method for the original static image. The experimental results show that the proposed method has high tolerance of authentication to both compressions and vicious attacks.

To use the proposed method, we need to store in a database 1) flags used for selective coding, and 2) an original code for several selected frames of each moving image file whose copyright we want to protect. In calculating the authentication ratio for authentication of an original moving image file, we do not need an original moving image file, but we do need 1) the flags, and 2) the original code for several selected frames of the original moving image file.

References

[1] Cox, I.J., Kilian, J., Leighton, T. and Shamoon, T. (1997) Secure Spread Spectrum Watermarking for Multimedia. *IEEE Transactions on Image Process*, **6**, 1673-1687. http://dx.doi.org/10.1109/83.650120

[2] Inoue, H., Miyazaki, A., Miyamoto, A. and Katsura, T. (1988) A Digital Watermark Based on the Wavelet Transform and Its Robustness on Image Compression. 1998 *International Conference on Image Processing*, **2**, 391-395.

[3] Kundur, D. and Hatzinakos, D. (1997) A Robust Digital Image Watermarking Method Using Wavelet-Based Fusion. *International Conference on Image Processing*, **1**, 544-547. http://dx.doi.org/10.1109/ICIP.1997.647970

[4] Shino, M., Choi, Y. and Aizawa, K. (2000) Wavelet Domain Digital Watermarking Based on Threshold-Variable Decision. *Technical Report of IEICE*, *DSP*2000-86, **100**, 29-34. (In Japanese)

[5] Swanson, M.D., Zhu, B. and Tewfik, A.H. (1996) Transparent Robust Image Watermarking. *International Conference on Image Processing*, **3**, 211-214. http://dx.doi.org/10.1109/ICIP.1996.560421

[6] Tsai, M.J., Yu, K.Y. and Chen, Y.Z. (2000) Joint Wavelet and Spatial Transformation for Digital Watermarking. *IEEE Transactions on Consumer Electronics*, **46**, 241-245.

[7] Xia, X.G., Boncelent, C.G. and Arce, G.R. (1997) A Multiresolusion Watermark for Digital Image. *International Conference on Image Processing*, **1**, 548-551. http://dx.doi.org/10.1109/ICIP.1997.647971

[8] Inoue, D. and Yoshitomi, Y. (2009) Watermarking Using Wavelet Transform and Genetic Algorithm for Realizing High Tolerance to Image Compression. *Journal of the IIEEJ*, **38**, 136-144.

[9] Yoshitomi, Y., Asada, T., Kinugawa, Y. and Tabuse, M. (2011) An Authentication Method for Digital Audio Using a Discrete Wavelet Transform. *Journal of Information Security*, **2**, 59-68. http://dx.doi.org/10.4236/jis.2011.22006

[10] Asada, T., Yoshitomi, Y. and Tabuse, M. (2010) A Verification Method for a Digital Image File Using a Discrete Wavelet Transform. *Journal of IIEEJ*, **39**, 1088-1094. (In Japanese)

[11] MUSCLE-VCD-2007. http://www-rocq.inria.fr/imedia/civrbench/index.html

[12] Libjpeg (2015) http://www.ijg.org/

[13] FFmpeg (2015) http://ffmpeg.org/

[14] Fujii, R., Noguchi, T., Tamura, M., Yoshitomi, Y., Asada, T. and Tabuse, M. (2014) A Verification Method for a Moving Image File Using a Discrete Wavelet Transform. *Proceedings of Forum on Information Technology*, **3**, 195-196. (In Japanese)