Proceedings of the 6th WSEAS International Conference on Multimedia Systems & Signal Processing, Hangzhou, China, April 16-18, 2006 (pp76-81)

Digital Image Encryption Method Using Interleaving and Random Shuffling

JI-BUM LEE, HYUNG-HWA KO
Dept.of Electronics and Communications Engineering
Kwangwoon University
447-1 Wolkye-Dong, Nowon-Gu, Chambit-kwan 813
SEOUL, KOREA
HA-SUNG KOO
Dept.of Computer and information
Hanseo University
Daegok-ri, Haemi-myun, Seosan-si
KOREA

, http://ipcl.kw.ac.kr

Abstract:
Distribution of digital multimedia contents which are used for broadband communication makes it possible for users to have joint ownership of information. It means that illegal copy and forgery of digital contents is possible to individuals. Therefore, it is necessary to protect right of contents authorship using digital watermarking or digital encryption. When watermarking and encryption are combined, it is very powerful method to protect right of contents authorship, But in this paper, we focused on the encryption.
In this paper, we proposed digital image encryption method using interleaving and random shuffling to protect right of contents authorship. Experimental results show that proposed method can be used to encrypt digital image data.

Key-Words: Multimedia Encryption, Random Shuffling, Digital Right Management

1 Introduction

The rapid development of multimedia compression and processing techniques has brought a proliferation of novel multimedia application, such as video-on demand and real-time video multicast. As multimedia security plays an important role in multimedia applications, it has become a focus of research in recent years. Traditional crypto algorithms, such as private-key 3DES and the public-key RSA, work perfectly well with regular security application. However, traditional crypto algorithms fail to give a satisfactory performance when it comes to multimedia content encryption. Main reason is processing time overhead, standard compliance.

Tang proposed frequency domain selective encryption scheme, which is called block shuffle[1]. This algorithm shuffles the DCT coefficients in 8 by 8 blocks with a permutation table instead of the original zig-zag scanning order of MPEG. Although the algorithm introduces few processing overhead, it replaces the original zig-zag order with a random order. This scrambling scheme is very fast but it reveals serious security problems. This scrambling scheme based encryption method is very weak to plain text attack[2].

The MHT scheme proposed by C-P. Wu and C.C. Kuo chooses several different Huffman tables from a vast number of possible candidates, and uses them alternatively to encode multimedia data. The MHT scheme requires very little computational overhead, but it is vulnerable to chosen-plaintext attack. Also in the MHT scheme, the security is in direct proportion to the number of chosen tables[5].

Jiangtao Wen introduced a framework and new tools that could achieve any level of syntax compliance in any format/standard through a unique compliance-preserving encryption method of variable length coded fields in compressed bitstreams[2-4].

Encryption by shuffling is usually subject to

plaintext attack, where the attacker can reverse engineer the shuffling table if he has access to both the plaintext and ciphertext. In order to overcome this weak point, it is necessary to change the key used in generation the shuffling table. But, this may pose significant burden on the security key management system. A better approach is to generate the shuffling table based on encryption some local feature. Shuffling based on tables generated this way is self-synchronous[6,7,8].

In this paper , we propose an encryption scheme which is based on local feature of image, fast and also robust to both plaintext and ciphertext attacks.

2 Proposed method

The method proposed in this paper is multiple shuffling (interleaving + random shuffling) that shuffles elements at random through interleaving and again shuffles using a random permutation table in order to get a robustness against plaintext attacks, to which existing image or video encryption through random shuffling is very vulnerable. Although shuffling using a random permutation table is a fixed shuffling, the result of proposed multiple shuffling could be random due to interleaving at the first step. Therefore proposed shuffling method is robust to both plaintext and ciphertext attacks.

2.1 Proposed interleaving method

2.1.1 Definitions of terms

Shuffling element: Objects to be shuffled through interleaving or a random shuffling table are elements. Elements used in this study are 8 by 8 blocks resulting from DPCM processing after DCT operation and quantization. The use of DPCM-processed blocks is for avoiding the increase of the volume of data in compression.

Shuffling size: Shuffling size means the total number of elements to be shuffled at random through interleaving or using a random table. This variable determines the security level. For example, if shuffling elements are all different from one another and its size is n the probability for the elements to be relocated to their original position will be n!. In the proposed method, there are two shuffling spaces, one for interleaving and the other for random table, so there can be two size variables.

Feature value: A feature value indicates the local characteristic of shuffling elements. To enhance the security level, XOR operation between a feature value and random sequence is used. The feature value used in the experiment was the value of the lowest four bits of DPCM coefficient.

Interval index: Interval index is used to randomize shuffling order through XOR operation with a feature value. It is a random value generated by a security key and a variable determining the security level of the proposed interleaving method.

Relocation interval: Interleaving in the communication system has a certain regular formula such as the prior transmission of the first bit. According to the formula, the receiver can restore the original order. If shuffling elements are relocated according to such a formula, they are always relocated by the same formula regardless of the feature of the elements and attackers can find the interleaving formula through repetition. Thus, the present study made interleaving elements relocated at random by the result of XOR operation between feature value and interval index.

Status code : This value indicates whether the current shuffling elements have been relocated. Its initial value in the stage of encryption is 0, and the status code for the position of a relocated shuffling element is switched to 1. The reverse procedure is followed in the stage of decryption.

Number of iteration : Iteration is made until the status code is switched to 1 at all positions or a preset number of times. If the predefined relocation interval is too large, neighboring elements can happen in sequence. Thus, it is desirable to set the number of iteration adequately according to the size of relocation interval.

Pass : A pass means the route of a iteration

2.1.2 Proposed interleaving

Interleaving applied in the experiment used DPCM-processed 8 by 8 blocks as shuffling elements, and the size of shuffling space of interleaving was 512. This corresponds to the total number of blocks of CbCr elements in the 256*256 color image of Lena used in the experiment. Interleaving was performed in the following order.

1. Arrange shuffling elements in their original order.

2. Do not relocate the first element but switch its status code to 1.
3. Calculate the feature value from the first element.
4. Perform XOR operation between the obtained feature value and interval index, and use the result as relocation interval.
5. Relocate the element apart as long as the calculated relocation interval next to the first element. Switch the status code of the corresponding position to 1.
6. Repeat Step 3 ~ 5. If the current element is the last element, go back to the beginning after a one pass.
7. Perform Step 3 ~ 6 as many as the number of iteration, and stop when all status codes have been switched to 1 or the present number of iteration has been made.

As a simple example, character string IMG ENCRYPTION the array size of which is 13, is reshuffled through the following procedure. Figure 1 shows the initial state of memory before interleaving.

Normal order of elements - 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 12, 13
Relocation interval - 3, 1, 2, 1, 2, 1, 2, 3, 1, 1, 3, 2, 1
Status code - 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
The maximum number of repetition : 3

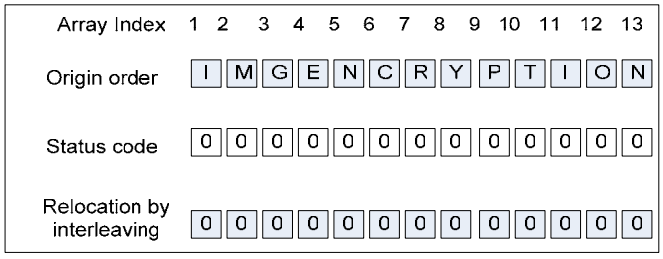

Fig.1 Initial State

1. The 1st element(I) is not relocated but maintains its current position. Because relocation interval obtained from the 1st element is 3, the 4th element(E) which is 3 apart from the 1st element, is relocated.
2. Because the relocation interval of the 4th element is "1" the 5th element is relocated. The process is repeated with increasing the array index. The relocation interval of the 10th element is 1 the next one is the 11th element. The relocation interval of the 11th element is 3, the next one is the 14th element but there is no 14th element. Thus, the 11th element is the last relocated element and the first pass is finished.

After the completion of the first pass, values in the temporary memory are as in Figure 2.

Fig.2 State after 1st pass

3. As the first pass has been finished, among unrelocated elements in the array index, the one of the highest order, which is the second element in this case, becomes the first element in the second pass. Following the same procedure as the first pass, the 2nd, 3rd and 8th elements are relocated.
4. The third pass begins from the 6th element, which has not been relocated yet. Because the relocation interval of the 6th element is 1, the 12th element, which is 1 apart from the 6th one among unrelocated elements, is relocated. The relocation interval of the 12th element is 2, so the third pass is finished without the relocation of the 13th and last element.
5. Because the maximum number of repetition is 3, unrelocated elements are relocated in order. In this case, the unrelocated 13th element is put in the last position.
6. After relocation has been completed, the state of memory is as in Figure 3.

Fig.3 Completion of relocation

In the stage of decryption, deinterleaving should be performed. For this, received elements are stored in a temporary memory as large as the predefined size of interleaving space. Because the element received first is the 1st element, its feature value is extracted, and relocation interval is obtained through XOR operation between the extracted feature value and the interval index. If the obtained relocation interval is 3, the original position of the element in the second position in the receiving memory is the 4th position, which is 3 apart from the 1st. This process is repeated as many times as the given number of repetitions or until all status codes are switched to 0. Interval index used in deinterleaving is generated by the same security as that used in interleaving.

2.2 Combination of interleaving and random shuffling

In existing random shuffling, the security level is determined by the number of shuffling elements. However, if an attack is made with both encrypted data (ciphertext) and pre-encrypted data (plaintext), the security level falls down significantly. Particularly when the attacker attacks using ciphertext obtained by putting arbitrarily manipulated plaintext into the cipher, the security level of random shuffling goes down in geometric progression. For example, if 128 shuffling elements have been relocated at random and the probability of restoring their original order is 128!, a chosen-text attack mentioned above can restore the original order easily through a maximum of 128 repetitions. That is, assuming that only one element is different from the other 127 elements and the 127 elements have the same value, the order can be restored through 128 repetitions, and if 128 different values can be assigned to the elements, the order can be found through a repetition.

In the proposed interleaving method, the minimum security level was determined by the feature value of interleaving elements and interval index. An advantage of the proposed interleaving method is that the form of shuffling is irregular by the feature value of elements and interval index. Accordingly, if the result of shuffling through the proposed interleaving is relocated again using a random shuffling table, the final form of random shuffling become irregular rather than fixed, so pre-encrypted plaintext known to the attacker is different from the original plaintext as a result of interleaving, losing the meaning as a plaintext. In consequence, thanks to the non-linearity of the proposed interleaving method, the combination of interleaving and random shuffling can be a robust encryption method against plaintext attacks.

2.3 Image encryption in JPEG compression using the proposed algorithm

In order to examine the encryption effectiveness of the multiple shuffling method combining the proposed interleaving and shuffling, we conducted an experiment using a still image. In case of DCT-based still image compression, the shuffling elements can be 8*8 blocks, macro-blocks, DCT coefficient of blocks, Huffman codeword, run length codeword, etc. This study used DPCM-processed 8*8 blocks as shuffling elements. Encryption using the proposed method follows the following procedure.

1. Decide the number of shuffling elements and the size of shuffling space.
2. Generate random numbers fit to the size of random shuffling space using an arbitrary seed value.
3. Carry out DCT and quantization for each block through JPEG compression.
4. Obtain DPCM values from quantized blocks.
5. Obtain the feature value and relocation interval of each block through the proposed interleaving procedure. The feature is the lowest four bits of DPCM coefficient.
6. Relocate blocks through the interleaving procedure. In this experiments, the number of iteration is limited to 3.
7. Shuffle relocated blocks again using a shuffling table generated by random numbers.
8. Carry out JPEG coding of the shuffled blocks through run length codeword, Huffman codeword, etc.

3. Simulation and Results

In order to verify the method combining the proposed interleaving and random shuffling, we conducted an experiment with a still image. The experiment was carried out on a Pentium PC 4(2.1Ghz, 512Mbyte) in Windows environment using a color BMP image of Lena (256*256).

Figure 4, 5 shows the results of interleaving when interleaving was repeated once, twice and three times, respectively. The image is hardly recognizable only with three repetitions. Only the Lena's eye part is barely recognizable. Figure 6 and 7 are the results of interleaving iterated 1 and 3 times, respectively, and random shuffling. Computing time in encryption was slightly different according to the number of iteration of interleaving, and increased by around 10 -14 % on the average compared to that taken when only JPEG was performed.

Number of interleaving iteration	JPEG only	Shuffling +JPEG	proposed
1	9.800ms	11.117ms	11.105ms
3	9.800ms	11.117ms	11.133ms
5	9.800ms	11.117ms	11.154ms

*No bit overhead at JPEG Compression @ Q=70

This is not because interleaving takes a long processing time but because block shuffling involves memory copy operation. In addition, it is because of difference in the size of shuffling space that time for both interleaving and random shuffling is slightly shorter than that for only random shuffling. That is, in the proposed method, the size of shuffling space was set at 512 identical to the size of 256 by 256 Lena image's color elements. This is because interleaving was made first before random shuffling and, as a result, the two spaces are mixed evenly even if the size of shuffling space is a half of the entire size. On the contrary, if only random shuffling is used, the size of shuffling space would be a 1024 in order to mix luminance elements throughout the entire area.

4. Conclusions

The present study proposed a method of enhancing security against known-text attacks and chosen-text attacks, to which image scrambling encryption using existing random shuffling is very vulnerable. Because elements are relocated first by the feature of the image and then shuffling is made by a shuffling table, we can obtain security as much as the size of shuffling space. In addition, when images are encrypted using the proposed shuffling method, image quality and bit rate after encryption are the same as those before encryption. Moreover, because encrypted bit stream satisfies the need of interoperability, the method can be applied advantageously to video encryption. What is more, compared to random shuffling only, the method needs only a half size of random shuffling space to support the same security level and this also reduces calculation time. A disadvantage is the increased use of memory in order to expand the size of shuffling space for higher encryption security. Thus, future research will be made on how to reduce the use of memory using.

Reference

[1]L.Tang, Methods for encrypting and decrypting MPEG video data efficiently, *Proc. the Fourth ACM Internal Multimedia Conference*, pp.219-229, 1996

[2]J.Wen, M.Severa, W.Zeng, M.Luttrell and W.Jin, A format compliant configurable encryption framework for access control of video, *IEEE Trans. Circuits & Systems for Video Technology*, Special Issue on Wireless Video, 2002.

[3]Wenjun Zeng and Shwwmin Lei, Efficient frequency domain selective scrambling of digital video, *IEEE Trans. Multimmedia*, 2002

[4]J.Wen, M.Severa, W.Zeng, M.Luttrell and W.Jin, A format compliant configurable encryption framework for access control of multimedia, *Proc. IEEE Workshop and Multimedia Signal Processing*, pp.435-440, Cannes, Frances, Oct.2001.

[5]C.-P. Wu and C.-C. Kuo, Efficient Multimedia Encryption via Entropy Codec Design, *Proceedings of SPIE International Symposium Electronics imaging 2001*, Vol.4314, Jan.2001.

[6] W.Zeng, J.Wen, and M.Severa, Fast self-synchronous content scrambling by spatially shuffling codewords of compressed bitstream, *Proc. IEEE ICIP*, 2002

[7]G. Liu, T. Ikenaga, S. Goto and T. Baba, A selective video encryption scheme for MPEG compression standard, *IEICE Trans. Fundamentals, Vol.E89-A*, No.1 Jan. 2006.

[8]L.Qiao and K. Nahrstedt, Comparison of MPEG encryption algorithms, *Inter. Journal on Computer & Graphics*, Special Issue on Data Security in Image Comm. and Network, 22(3), 1998

Fig. 4 Interleved image(1 iteration)

Fig. 5 Interleved image(3 iteration)

Fig. 7 shuffled image(3 iteration + random shuffling)

Fig. 6 shuffled image(1 iteration+ random shuffling)

Fig. 8 Decrypted JPEG format image(Q is 70)