

Analysis and Cryptanalysis of a Selective Encryption Method for JPEG Images

William Puech, José Marconi Rodrigues, Adrian G. Bors

▶ **To cite this version:**

William Puech, José Marconi Rodrigues, Adrian G. Bors. Analysis and Cryptanalysis of a Selective Encryption Method for JPEG Images. WIAMIS: Workshop on Image Analysis for Multimedia Interactive Services, Jun 2007, Santorini, Greece. lirmm-00192604

HAL Id: lirmm-00192604

https://hal-lirmm.ccsd.cnrs.fr/lirmm-00192604

Submitted on 28 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analysis and Cryptanalysis of a Selective Encryption Method for JPEG Images

W. Puech, J.M. Rodrigues
Laboratory LIRMM, UMR CNRS 5506, University of Montpellier II, FRANCE
jose-marconi.rodrigues@lirmm.fr, william.puech@lirmm.fr

A.G. Bors
Dept. of Computer Science, University of York, YORK YO10 5DD, U.K.
adrian.bors@cs.york.ac.uk

Abstract

This paper addresses the protection of images. We address the problem of simultaneous selective encryption (SE) and image compression. The SE is done by using the Advanced Encryption Standard (AES) algorithm with the Cipher Feedback (CFB) mode one a part of the Huffman coefficients corresponding to the AC frequencies. For the compression we consider the JPEG algorithm. Our approach is done without affecting the compression rate and by keeping the JPEG bitstream compliance. In the proposed method, the SE is performed in the Huffman coding stage of the JPEG algorithm without affecting the size of the compressed image. We provide an experimental analysis of the proposed method when applied on still images as well as the results when applying SE in JPEG compressed images.

1. Introduction

Digital rights management (DRM) systems enforce the rights of the multimedia property owners while ensuring the efficient rightful usage of such property. The technical challenges posed by such systems are formidable and previous approaches have not entirely succeeded in tackling them [4]. The variety of applications for secure multimedia requires either full encryption or selective encryption. However, there is a huge spectrum of applications that demands security on a lower level, as for example that ensured by selective encryption (SE). Such approaches reduce the computational requirements in networks with diverse client device capabilities [1]. The goal of SE is to encrypt a well defined range of parameters or coefficients. The security level of SE is always lower when compared with the full encryption. However, SE decreases the data size to be encrypted and consequently requires lower computational time. In this case we have a trade-off between the amount of encrypted data and the necessary computational resources. JPEG is a commonly used standard algorithm for image compression which is still largely employed in image processing for security communication and in industrial applications [6]. SE can be included inside a standard coding algorithm such as JPEG, JPEG 2000, MJPEG or MPEG, while maintaining the bitstream compliance. In fact, using a standard decoder it should be possible to visualize the SE data in low resolution. On the other hand, with a specific decoding algorithm and a secret key it should be possible to correctly decrypt the SE data and get the high resolution whenever it is desired. In this paper we present a new approach SE for JPEG compressed image sequences by using variable length coding (VLC). We propose to encrypt selected bits in the Huffman coding stage of the JPEG. In our approach we use the Advanced Encryption Standard (AES) [2] in the Cipher Feedback (CFB) mode which is a stream cipher algorithm.

2. Previous work

Confidentiality is very important for lower powered systems such as for example wireless devices. Always, when considering image processing applications on such devices we should use minimal resources. However, the classical ciphers are usually too slow to be used for image and video processing in commercial low powered systems. The selective encryption (SE) can fulfill the application requirements without the overhead of the full encryption. In the case of SE, only the minimum necessary data are ciphered. In [8] was proposed a technique called zigzag permutation applicable to DCT-based videos and images. On one hand this method provides a certain level of confidentiality, while on the other hand it increases the overall bit rate. Combining SE and image/video compression using the set partitioning in hierarchical trees was used in [1]. However, this approach requires a significant computational complexity. A method that does not require significant processing time and which operates directly on the bit planes of the image was proposed in [5]. An approach that turns entropy coders into encryption ciphers using statistical models was proposed

in [9]. In [3] it was suggested a technique that encrypts a selected number of AC coefficients. The DC coefficients are not ciphered since they carry important visual information and they are highly predictable. In spite of the constancy in the bit rate while preserving the bitstream compliance, this method is not scalable. Moreover, the compression and the encryption process are separated and consequently the computational complexity is increased. The robustness of partially encrypted images to attacks which exploit the information from non-encrypted bits together with the availability of side information, was studied in [7].

3. The proposed selective encryption method

Our approach combines JPEG compression and the selective encryption during the Huffman coding stage of JPEG. Let $Y_i = X_i \oplus E_k(Y_{i-1})$ be the notation of the encryption of a n bit block X_i using the secret key k with the AES cipher in CFB mode. Let $D_k(Y_i)$ be the decryption of a ciphered text Y_i using the secret key k. In the CFB mode, the previously encrypted block is used to encrypt and to decrypt the current one. The proposed SE is applied in the entropy encoding stage during the creation of the Huffman vector. The three stages of the proposed algorithm are: the construction of the plaintext X_i, described in Section 3.1, the encryption of X_i to create Y_i which is provided in Section 3.2 and the substitution of the original Huffman vector with the encrypted information.

3.1. The construction of plaintext

For constructing the plaintext X_i, we take the non-zero AC coefficients of the current block i by accessing the Huffman vector from the end to the beginning in order to create {H, A} pairs. Indeed, the human visual system is more sensitive to the lower frequencies when compared to the higher range of frequencies. Therefore, by using the Huffman bits in their corresponding decreasing frequency ordering we can calibrate the visual appearance of the resulting image. This means that we can achieve a progressive or scalable encryption with respect to the visual effect. The resulting image will have a higher level of encryption as we increasingly use the lower range of frequencies. A constraint C is used in order to select the quantity of bits to encrypt from the plaintext X_i. The constraint C graduates the level of ciphering and the visual quality of the resulting image. For each block, the plaintext length $L(X_i)$ to be encrypted depends on both the homogeneity of the block and the given constraint C:

$$0 \leq L(X_i) \leq C, \tag{1}$$

where $C \in \{4, 8, 16, 32, 64, 128\}$ bits. When $C = 128$ the AES will fully use the available block of Huffman bits while for the other values several blocks are grouped in order to sum up to 128 bits which is the standard size of AES. The constraint C specifies the maximum quantity of bits that must be considered for encryption in each block as in VLC. The Huffman vector is encrypted while $L(X_i) \leq C$ and the bits corresponding to the DC coefficient are not reached. Then, we apply a padding function $p(j) = 0$, where $j \in \{L(X_i) + 1, \ldots, C\}$, to fill out with zeros the vector X_i up to C bits.

The length of amplitude A in bits is extracted using H. These values are computed and tested according to equation (1). In the proposed method, only the values of the amplitudes $(A_n, A_{n-1}...A_1)$ are considered to build the vector X_i. The Huffman vector is composed of a set of pairs {H, A} and of marker codes such as ZRL and EOB. If the smallest AC coefficients are zero, the Huffman bitstream for this block must contain the mark EOB. In turn, the ZRL control mark is found every time that sixteen successive AC coefficients are zero and they are followed by at least one non-zero AC coefficient. In our method, we do not make any change in the head H or in the mentioned control marks. To guarantee a full compatibility with any JPEG decoder, the bitstream should only be altered at places where it does not compromise the compliance with the original format.

3.2. Encryption of the plaintext with AES in the CFB mode

In the encryption step with AES in the CFB mode, the previous encrypted block Y_{i-1} is used as the input of the AES algorithm in order to create Z_i. Then, the current plaintext X_i is XORed with Z_i in order to generate the encrypted text Y_i. For the initialization, the IV is created from the secret key k according to the following strategy. The secret key k is used as the seed of PRNG (Pseudo-Random Number Generator). Firstly, the secret key k is divided in bytes. The PRNG produces a random number for each byte component of the key, that define the order of IV formation. We substitute then Y_0 by the IV, and Y_0 is used in AES to produce Z_1. With the CFB mode of the AES algorithm, the generation of the keystream Z_i depends of the previous encrypted block Y_{i-1}. Consequently, if two plaintexts are identical $X_i = X_j$ in the CFB mode, then always the two corresponding encrypted blocks are different, $Y_i \neq Y_j$.

3.3. Substitution of the original Huffman bitstream

The third step is the substitution of the original information in the Huffman vector by the encrypted text Y_i. As in the first step (construction of the plaintext X_i), the Huffman vector is accessed in the sequential order, while the encrypted vector Y_i is accessed in the inverse order. Given the length in bits of each amplitude $(A_n, A_{n-1}, \ldots, A_1)$, we start substituting the original amplitude in the Huffman vector by the corresponding parts of Y_i. The total quantity

of replaced bits is $L(X_i)$ and consequently we do not necessarily use all the bits of Y_i.

4. Experimental results

4.1. Analysis of joint selective encryption and JPEG compression

We have applied simultaneously our selective encryption and JPEG compression as described in Section 3, on several images. In this section, we show the results of SE in the whole JPEG compressed image. The compressed JPEG Lena image 512×512 pixels with a quality factor (QF) of 100 % is shown in Fig. 1.a and the compressed JPEG with a QF of 10 % is shown in Fig. 1.d.

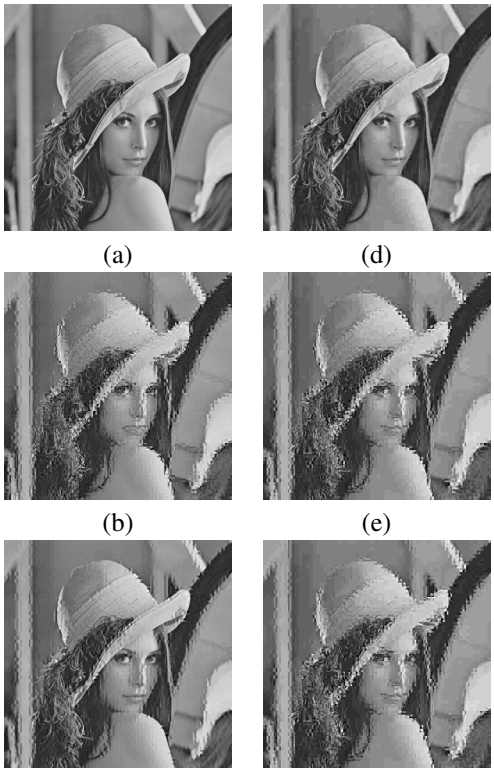


Figure 1. a) JPEG compressed image with QF=100%, b) Image (a) with $C = 128$ bits/block, c) Image (a) with $C = 8$ bits/block, d) JPEG compressed image with QF=10%, e) Image (d) with $C = 128$ bits/block, f) Image (d) with $S = 8$ bits/block.

In a first set of experiments, we have analyzed the available space for encryption in JPEG compressed images. For each QF we provide the distortion calculated as the PSNR (Peak to Signal Noise Ratio) as well as the average number of available bits for SE per block of quantized DCT coefficients. We can observe that when QF is lower and implicitly the image compression is higher, we are able to embed fewer bits in the compressed image. This is due to the fact that JPEG compression creates flat regions in the image blocks increasing the number of AC coefficients equal to zero. Consequently, the Huffman coding creates special blocks for such regions which our method does not encrypt. Not all the available bits are actually used for SE because of the limit imposed by the constraint C. For optimizing the time complexity, C should be smaller than the average bits/block available. In Fig. 2 we provide the graphical representation displaying the variance in the number of available bits for SE per block. We can observe that the variance decreases together with the QF as the number of flat regions in the compressed image increases. For optimal time requirements we can observe that for a compressed JPEG image we should use a smaller constraint C. In Fig. 3 we show the evaluation of the PSNR between the crypto-compressed Lena image and the original, for several QF and for various constraints C. In the same figure, for comparison purposes we provide the PSNR between the compressed image with different QF and the original image. From this figure we can observe that for a higher C we encrypt a larger number of bits and consequently the image is more distorted with respect to the original. However, when $C \in \{32, 64, 128\}$, the difference in the PSNR distortion is similar and varies slowly when decreasing the QF. In Fig. 1.b we show the original Lena image encrypted using a constraint $C = 128$ bits per block of quantized DCT coefficients, while in Fig. 1.c is the same image encrypted using a constraint $C = 8$ bits/block.

In Fig. 1.e we show Lena image with QF of 10 %, encrypted using a constraint $C = 128$ bits/block, while in Fig. 1.f the same image is encrypted using a constraint $C = 8$ bits/block. We can see that the degradation introduced by the encryption in the image with QF=100 %, from Fig. 1.b, is higher than the degradation in the image from Fig. 1.c because in the latter image we encrypt more bits per block. When combining a high JPEG compression level (QF=10 %) with selective encryption, as shown in the images from Figs. 1.e and 1.f, we can observe a high visual degradation with respect to the images from Figs. 1.b and 1.c, respectively. The higher distortion is caused by the increase in the number of block artifacts. The distortion is more evident when observing some image features as for example the eyes.

4.2. Cryptanalysis of the SE method

It should be noted that security is linked to the ability to guess the values of the encrypted data. For example, from a security point of view, it is preferable to encrypt the bits that look the most random. However, in practice this trade-off is challenging because the most relevant information, such as the DC coefficients in a JPEG encoded image are usually highly predictable [3]. In another experiment we have

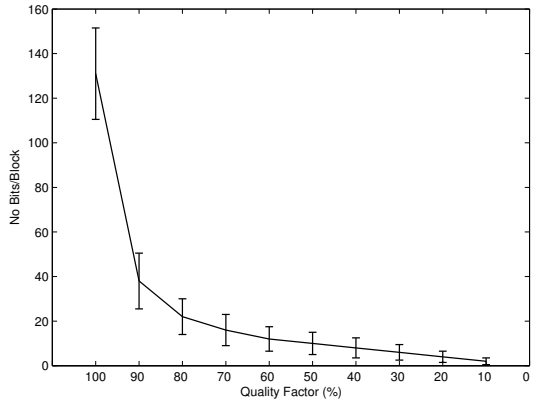


Figure 2. The average number of bits available for SE per block, where the variance is indicated as a confidence interval.

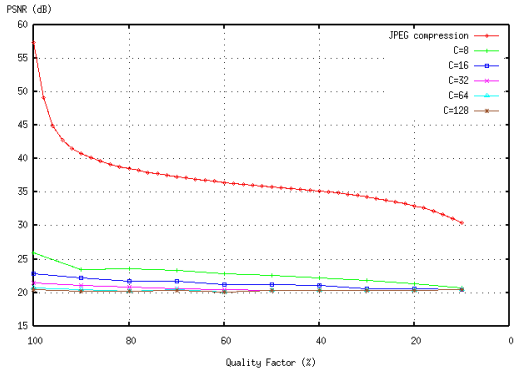


Figure 3. PSNR of crypto-compressed Lena image for various quality factors and constraints.



Figure 4. Attack in the selectively encrypted image (Fig.1.b) by removing the encrypted data.

replaced the encrypted AC coefficients with constant values. For example, if we set the encrypted AC coefficients of all blocks from Fig. 1.b which shows Lena with QF = 100 %, $C = 128$ having $PSNR = 20.46\ dB$ to zero, we get the image illustrated in Fig. 4. Its PSNR with respect to the original image is $23.66\ dB$. We can observe that in SE, since we do not encode the Huffman coefficients corresponding to the DC component, the rough visual information can be simply recovered by replacing the ciphered AC coefficients with constant values.

5. Conclusion

In this paper, a selective encryption system for JPEG compressed images has been analyzed. The encryption is performed in the Huffman coding stage of the JPEG algorithm using the AES encryption algorithm in the CFB mode. In this way the proposed encryption method does not affect the compression rate and the JPEG bitstream compliance. The selective encryption (SE) is performed only on the Huffman vector bits that correspond to the AC coefficients as provided by the DCT block of JPEG. The proposed methodology is applied for ensuring the protection of images. Only authorized users that possess the key can decrypt the entire encrypted image sequences. The proposed method has the advantage of portability on mobile devices, which currently embed the JPEG image compression algorithm, which corresponds to their low computational requirements.

References

[1] H. Cheng and X. Li. Partial Encryption of Compressed Images and Videos. *IEEE Trans. on Signal Processing*, 48(8):2439–2445, Aug. 2000.

[2] J. Daemen and V. Rijmen. AES Proposal: The Rijndael Block Cipher. Technical report, Proton World Int.l, Katholieke Universiteit Leuven, ESAT-COSIC, Belgium, 2002.

[3] M. V. Droogenbroeck and R. Benedett. Techniques for a Selective Encryption of Uncompressed and Compressed Images. In *Proc. of Advanced Concepts for Intelligent Vision Systems (ACIVS) 2002, Ghent, Belgium*, pages 90–97, Sept. 2002.

[4] E. Lin, A. Eskicioglu, R. Lagendijk, and E. Delp. Advances in Digital Video Content Protection. *Proc. of the IEEE*, 93(1):171–183, 2005.

[5] R. Lukac and K. Plataniotis. Bit-Level Based Secret Sharing for Image Encryption. *Pattern Recognition*, 38(5):767–772, May 2005.

[6] W. Pennebaker and J. Mitchell. *JPEG: Still Image Data Compression Standard*. Van Nostrand Reinhold, San Jose, USA, 1993.

[7] A. Said. Measuring the Strength of Partial Encryption Scheme. In *Proc. IEEE Int. Conf. on Image Processing, Genova, Italy*, volume 2, pages 1126–1129, 2005.

[8] L. Tang. Methods for Encrypting and Decrypting MPEG Video Data Efficiently. In *Proc. ACM Multimedia*, volume 3, pages 219–229, 1996.

[9] C. Wu and C. Kuo. Design of Integrated Multimedia Compression and Encryption Systems. *IEEE Trans. on Multimedia*, 7(5):828–839, Oct. 2005.