Selective Encryption of Human Skin in JPEG Images

José Marconi Rodrigues, William Puech, Adrian G. Bors

SELECTIVE ENCRYPTION OF HUMAN SKIN IN JPEG IMAGES

J.M. Rodrigues[a], W. Puech[a] and A.G. Bors[b]

[a]Laboratory LIRMM, UMR CNRS 5506, University of Montpellier II
161, rue Ada, 34392 MONTPELLIER CEDEX 05, FRANCE
[b]Dept. of Computer Science, University of York, YORK YO10 5DD, U.K.
jrodrigu@lirmm.fr, william.puech@lirmm.fr, adrian.bors@cs.york.ac.uk

ABSTRACT

In this study we propose a new approach for selective encryption in the Huffman coding of the Discrete Cosine Transform (DCT) coefficients using the Advanced Encryption Standard (AES). The objective is to partially encrypt the human face in an image or video sequence. This approach is based on the AES stream ciphering using Variable Length Coding (VLC) of the Huffman's vector. The proposed scheme allows the decryption of a specific region of the image and results in a significant reduction in encrypting and decrypting processing time. It also provides a constant bit rate while maintaining the JPEG and MPEG bitstream compliance.

Index Terms— Image encryption, JPEG compression, face protection, AES-OFB.

1. INTRODUCTION

The variety of applications for secure multimedia requires either full encryption or selective encryption. For example military and law enforcement applications require full encryption. However, there is a huge spectrum of applications that demands security on a lower level, *i.e.* partial or selective encryption (SE). SE is also an approach that reduces the computational requirements in networks with various client device capabilities [1]. One example of SE application in multimedia for security is for processing images acquired by a surveillance camera. Such images must be quickly transmitted and the full encryption is not necessary. The demand for security in SE applications is always lower when compared to the full encryption. In this case we have a trade-off between the amount of encrypted data and the necessary computational resources.

In this paper we propose a new approach for SE of the Huffman coding for JPEG images. In our approach we use the Advanced Encryption Standard (AES) [2] cipher in the Output Feedback Block (OFB) mode that allows versatility at the decoding stage. In Section 2, we review the previous research findings and discuss a possible application scenario. In Section 3, we introduce the proposed method. Finally in Section 4 we show the experimental results when we apply our algorithm in image sequence and provide the conclusions of this study in Section 5.

2. PREVIOUS WORK

Several selective encryption methods have been proposed for DCT compressed images. Droogenbroeck and Benedett [3] selected AC coefficients from compressed images for encryption. In their method the DC coefficients are not ciphered because they carry important visible information and they are highly predictable. The compression and encryption stages are separated in this approach and this requires an additional operating cost. Fisch et al. [4] have proposed a partial image encryption where the data are organized in a scalable bitstream form. It is well known that color skin is characterized by a specific chrominance range [5]. Bit streams are constructed from the DC and some AC coefficients from every image block and then arranged in layers according to their visual importance. Recently, Said [6] measured the strength of partial encryption describing attacks that exploit the information from non-encrypted bits.

2.1. Selective Encryption of JPEG Images

In the Huffman coding block, the quantized DCT coefficients are coded by the pair {HEAD, AMPLITUDE}. The HEAD contains the controllers provided by the Huffman's tables. The AMPLITUDE is a signed-integer that is the amplitude of the nonzero AC, or in the case of DC is the difference between two neighbouring DC coefficients. For the AC coefficients, the HEAD is composed from (RUNLENGTH, SIZE), while for the DC coefficients it is only made up by SIZE. In Huffman coding because DC coefficients are highly predictable they are not used for encryption. The method proposed in this paper is based on encrypting certain AC coefficients.

JPEG uses a method based on combining run-length and amplitude information for the AC coding. It aggregates zero coefficients into runs of zeros. RUNLENGTH is a consecutive array of numbers representing zero valued AC coefficients which precedes the nonzero values in the zigzag sequence. SIZE is the amount of necessary bits to represent

the AMPLITUDE. Two extra codes that correspond to (RUNLENGTH, SIZE) = (0, 0) and (15, 0) are used for symbolize EOB (End Of Block) and ZRL (Zero Run Length), respectively. The EOB is transmitted after the last nonzero coefficient in a quantized block. The ZRL symbol is transmitted whenever RUNLENGTH is greater than 15 and represents a run of 16 zeros.

2.2. The Advanced Encryption Standard Algorithm

Advanced Encryption Standard (AES) is a very powerful standard cipher that operates by performing a set of steps, for a number of iterations called rounds. The enciphering of a plain text X_i, where i is the current block, in AES is described in Fig. 1.a. The AES algorithm can support several modes such as Electronic CodeBook (ECB), Cipher Block Chaining (CBC), Output FeedBack (OFB), Cipher FeedBack (CFB) and Counter (CTR). In OFB mode, which is the mode selected for encryption in this study, the ciphertext block Y is produced by performing a XOR with Z_i, where $Z_i = E_k(Z_{i-1}), i \geq 1$ and $Y_i = X_i \oplus Z_i$, as illustrated in Fig. 1.b.

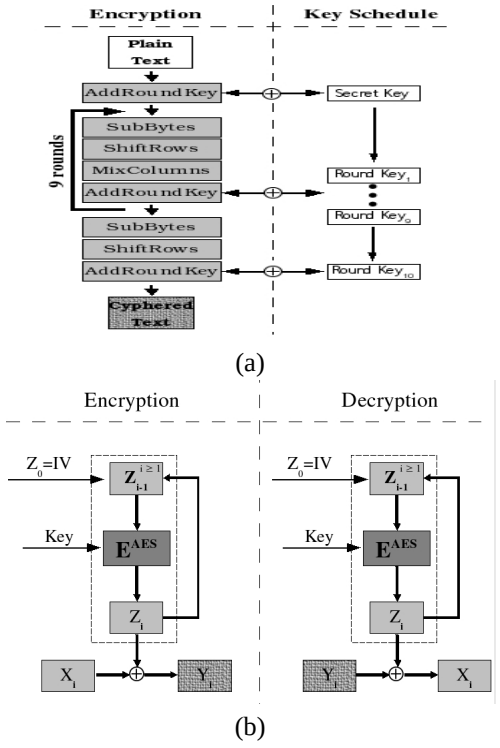

Fig. 1. a) AES general scheme, b) Encryption and decryption in OFB mode.

Although AES is a block cipher, in the OFB, CFB and CTR modes it operates as stream cipher. Each mode has advantages as well as drawbacks. In ECB and OFB modes for example any modification in the plaintext block causes the corresponding ciphered block to be altered, but other ciphered blocks are not affected. On the other hand, if a plaintext block is changed in CBC and CFB modes, then all subsequent ciphered blocks will be affected. These properties mean that OFB mode treats separately each block. From Fig. 1.b, we can observe that the encryption function $E_k(X)$ is used for both encryption and decryption in the OFB mode.

3. ENCRYPTION OF DCT CODED IMAGES

Let $E_k(X)$ be the encryption of a n bit data block X using the secret key k with AES cipher in the OFB mode. For practicality, we assume that $n = 12$ and that X is a plaintext. Let $D_k(Y)$ be the decryption of a ciphered text Y using the secret key k.

3.1. The Encryption Procedure

The proposed method for selective encryption is applied in the entropy encoding stage during the creation of the Huffman's vector [7]. The stages of the proposed algorithm are: the construction of the plaintext X_i, the ciphering of X_i to create Y_i, and the substitution of the original Huffman's vector with the ciphered information. These operations are performed separately in each quantized DCT block. The homogeneous blocks are not ciphered due to the scarcity of the AMPLITUDE information in the Huffman code.

3.1.1. The construction of plaintext X_i

For constructing the plaintext X_i, we consider the non-zero AC coefficients of the current block i by accessing the Huffman's vector to create the {HEAD, AMPLITUDE} pairs. The length of AMPLITUDE is extracted from each HEAD number. These values are tested according to the following :

$$f(\rho) \;\leq\; L_{X_i} \;\leq\; C, \qquad (1)$$

where ρ is the homogeneity of the block, $f(\rho) = 0$ for $\rho \rightarrow \infty$ and $C \in \{128,\ 64,\ 32,\ 16,\ 8,\ 4\}$ bits. As shown in Fig. 2, only the AMPLITUDE's $(A_n, A_{n-1}...A_1)$ are considered to build the vector X_i. The final plaintext length L_{X_i} depends on both the homogeneity of the block ρ and the given constraint C. The constraint C specifies the maximum number of bits that must be considered in each block. In this study we consider $C = 128$. Then, we apply the padding function $p(j) = 0$, where $n \geq j > L_{X_i}$, for filling in the vector X_i with zeros.

3.1.2. Ciphering of X_i in the OFB mode of the AES

In the ciphering step, the plaintext X_i is input into the AES cipher in order to create Y_i. Vector IV is created from the secret key k according to the following algorithm. The secret key k is used as the seed of PRNG (Pseudo-Random Number Generator). k is divided into 16 sets of 8 bits each. The

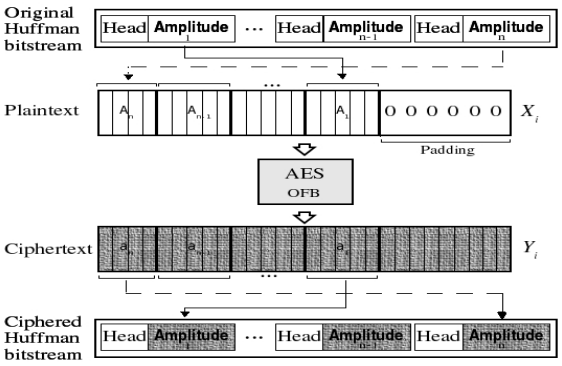

Fig. 2. Global overview of the proposed method.

PRNG produces 16 random numbers that define the order of formation for the IV vector. For example, if the first random number generated is 7, the first byte of the secret key will be copied in the seventh element of the vector IV. After generating the vector $IV = Z_0$, firstly it is ciphered to produce Z_1, then ciphered to produce Z_2, afterwards Z_3 and so on, as shown in Fig. 1b. Z_i is XOR-ed with the plaintext X_i producing Y_i. The use of OFB mode for ciphering purpose allows for an independent generation of the keystream Z_i.

3.1.3. Substitution of the original Huffman's bitstream

The final step is the substitution of the ciphered information in the Huffman's vector. As in the first step (construction of the plaintext X_i), the Huffman's vector is accessed in sequential order, while the ciphered vector Y_i is accessed in the reversed order. Given the length in bits of each AMPLITUDE $(A_n, A_{n-1}...A_1)$, we start replacing these parts of Y_i with the AMPLITUDE in the Huffman's vector. The total quantity of replaced bits is L_{X_i}. The block length does not change following the encryption. This is a very important property of this approach because the JPEG compression rate is not changed after the encryption.

3.2. The Decryption Procedure

The decryption process proceeds as follows. The secret key is used to build the vector $IV = Z_0$. Z_1 is created by encoding IV, while Z_n is created by encoding Z_{n-1}. Therefore, the same procedures for encryption as described in the previous section are employed at the decryption. The difference consists of the fact that the entry of the ciphering process is the ciphered Huffman's vector. This ciphered vector is also accessed in reversed order of its bits in order to construct the plaintext Y_i. Then, Y_i will be used together with Z_i in AES decoding procedure as shown in Fig. 1b. The resulted plaintext vector is split in order to substitute the AMPLITUDE in the ciphered Huffman code and to generate its corresponding original Huffman vector.

3.3. Detection of Human Skin in JPEG Images

There are many potential applications for the selective encryption methodology. In the following, the method described in Section 3.1 is applied to an image sequence in order to encrypt human faces selected based on skin detection.

The first step of the JPEG algorithm is the color space transformation. The image is mapped from RGB to YCbCr space. DCT is then calculated in each 8×8 block, generating the DC and AC coefficients. In order to detect human skin we use the fact that human skin is defined in a specific chrominance range [5]. We select Cb and Cr components such that:

$$\sqrt{(Cr - Cr_s)^2 + (Cb - Cb_s)^2} < S, \qquad (2)$$

where Cb_s and Cr_s are the reference skin color in $YCbCr$ space, and S is the threshold. A binary image is generated addressing the blocks that must be encrypted. The SE method is then applied in the Y Huffman vector corresponding to these blocks.

4. EXPERIMENTAL RESULTS

For our experiments, we have used JPEG compressed images in the baseline sequential mode with a Quality Factor (QF) of 100%. For the encryption we have used the AES in stream cipher mode (OFB) with a key of 128 bits long.

From the original RGB image of size 1024×696 pixels shown in Fig. 3.a, we have extracted the $YCbCr$ components. From the DCs of the components Cb and Cr displayed in Figs. 3.b and 3.c, we have obtained the binary image Fig. 3.d using equation (2) with $S = 15$, $Cr_s = 140$ and $Cb_s = 100$. In this binary image, the human skin is represented by the white pixels and we observe a good skin segmentation in the given image. Each white pixel from the skin segmentation map corresponds to a 8×8 pixels block in the original image. Only these blocks are selectively encrypted. In order to clearly show our results we have cropped a sub-image of 416 pixels displayed in Fig. 3.e. Each sub-image must be split in blocks of 8×8 pixels as it is required by the JPEG compression standard. We have used the selective encryption method described in this paper on the original image from Fig. 3.a after using skin detection. Only 3597 blocks (from a total of 11136 blocks) have been selective encrypted. In the compressed image (536 kB), only 7.32 $\%$ of the bits are encrypted. The detail from the resulting SE image is shown in Fig. 3.f.

It should be noted that possible attacks rely on the ability to guess the values of the encrypted data. From an image security point of view, it is preferable to encrypt the bits that look the most random. However, in practice the trade-off is more difficult to define because the most relevant information, like DC coefficients in a JPEG encoded image, usually are highly predictable [3, 8].

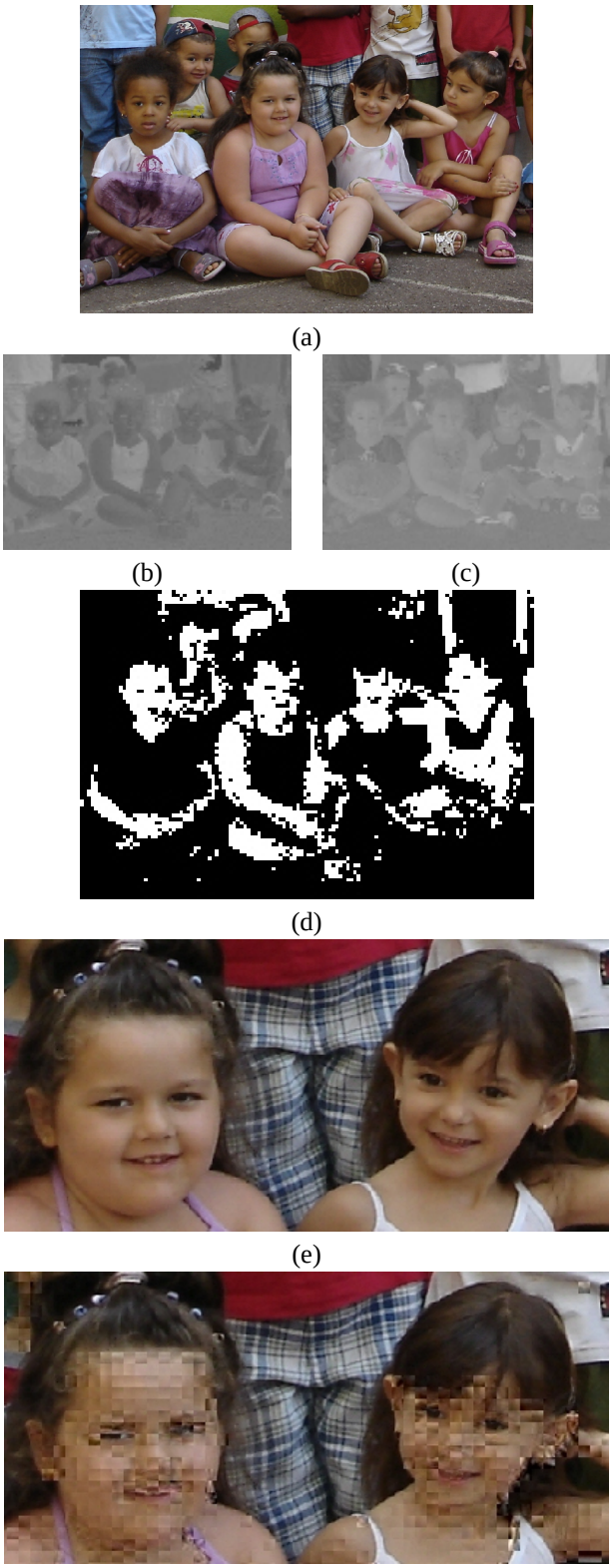

Fig. 3. a) Original image 1024×696 pixels; b) Cb component; c) Cr component; d) Binary image; e) Original sub-image; f) Selective encryption ciphered sub-image.

5. CONCLUSION

In this paper, we have proposed a new scheme for selective encryption of JPEG images by using the AES cipher in OFB mode. Our approach brings several advantages such as portability, constant bit rate, JPEG format compliance, scalable selective encryption and a progressive decryption of the region of interest. By using the proposed algorithm we are able to encrypt an image without affecting at all the JPEG compression rate. In the decoding stage we are able to decrypt selected areas, which makes the proposed method useful for a large range of applications.

6. REFERENCES

[1] H. Cheng and X. Li, "Partial Encryption of Compressed Images and Videos," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 239–245, 2000.

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

[3] M. Van Droogenbroeck and R. Benedett, "Techniques for a Selective Encryption of Uncompressed and Compressed Images," in *Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS) 2002, Ghent, Belgium*, Sept. 2002.

[4] M. M. Fisch, H. Stgner, and A. Uhl, "Layered Encryption Techniques for DCT-Coded Visual Data," in *European Signal Processing Conference (EUSIPCO) 2004, Vienna, Austria*, Sep., 2004.

[5] Y. Sheng, A.H. Sadka, and A.M. Kondoz, "Automatic Face Segmentation for 3D Model-Based Video Coding," in *IEE International Conference on Visual information Engineering, UK*, 2003, pp. 274–277.

[6] A. Said, "Measuring the Strength of Partial Encryption Scheme," in *ICIP 2005, IEEE International Conference in Image Processing, Genova, Italy*, 2005, vol. 2, pp. 1126–1129.

[7] J.-M. Rodrigues, W. Puech, and A.G. Bors, "A Selective Encryption for Heterogenous Color JPEG Images Based on VLC and AES Stream Cipher," in *CGIV'06, Leeds, UK*, 2006.

[8] T. Lookabaugh, "Selective encryption, information theory, and compression," in *38th ASILOMAR Conference on Signals, Systems and Computers*, 2004, vol. 1, pp. 373–376.