PROTECTION OF JPEG COMPRESSED E-COMICS BY SELECTIVE ENCRYPTION

Mickael Pinto, William Puech and Gérard Subsol

LIRMM Laboratory, UMR 5506 CNRS, University of Montpellier II
161, rue Ada, 34392 MONTPELLIER CEDEX 05, FRANCE

ABSTRACT

Nowadays, there is an increasing number of digital comics, so-called e-comics, available on Internet. Protecting these e-comics without Digital Rights Management (DRM) is a new issue. In this paper, we propose a method to ensure the protection of e-comics by encrypting only bubbles so that the story will not be understandable. To develop this method, we used standards like JPEG compression and AES encryption. The first step of the method involves detecting bubbles, and we decided to use the MSER algorithm, which is the most commonly used for text detection. The second main step involves making these areas unreadable by encrypting AC coefficients of the JPEG block, corresponding to high frequencies during entropy coding. Several experimental results show the efficiency of the proposed method.

Index Terms— E-comics, protection, selective encryption, JPEG, AES, OFB mode, CFB mode, MSER

1. INTRODUCTION

The advent of high-resolution, multi-color and high-contrast screen technologies has made it possible to comfortably read digital comic strips, so-called e-comics, even on small-sized displays. Several companies propose technological infrastructures and authoring tools to transform comics into digital format[1]. Traditional comics companies, such as Marvel Comics, even propose a digital version of their comics on their official website[2].

A major problem is to protect the digital content and several Digital Rights Management (DRM) technologies have been implemented for this purpose. For most of them, the e-comic file, which is encrypted in a specific format, is read with a dedicated application. This is quite a heavy constraint and if you crack the application, the file can be converted into a standard format and distributed without limitation. A way to solve these drawbacks would be to integrate the DRM system directly into the image itself, which would then be stored in a standard format such as JPEG. There would be no specific software to develop and no piracy attacks aiming to modify the DRM could distort the image content.

In this paper, we propose an algorithm for reversible selective encryption during the JPEG compression process. This algorithm allows blurring of the e-comic image so as to make it unreadable for unauthorized users, while still allowing providing an overview of the content to these users. The idea is to encrypt some critical parts of the comics, such as the bubbles, in order to be able to assess the quality of the drawing or the coloring, which may be very important for marketing purposes, while not being able to read the texts.

In Section 2, previous work in text detection and selective encryption are presented. Then, in Section 3, we detail the steps of our method to protect e-comics. In Section 4, experimental results and the limitation of our proposed method are presented. Finally, in Section 5, concluding remarks and future prospects on the proposed process are discussed.

2. PREVIOUS WORK

Considerable research is focused on text detection in e-comics, but often the outcomes are different. For example, Su *et al.* proposed a text detection method based on SVG compression by using sliding concentric windows followed by a support vector machine and finally optical character recognition [1]. Their aim was to recognize characters and words so as to be able to automatically translate e-comics. Arai and Tolle had the same aim but they proposed a method to detect bubbles in images by blob detection [2]. However, this solution is not robust because it considers that bubbles are always proportional to the image size. On the other hand, Yamada *et al.* sought to make e-comics readable on low resolution mobile phones by extracting texts and showing them bigger on screens [3].

More generally, many methods have been proposed to detect text in any images. Neumann and Matas proposed to use the Maximally Stable Extremal Regions (MSER) algorithm [4]. Initially, the MSER algorithm, introduced by Matas *et al.* [5], was developed to detect blobs in images and find correlations between elements from two images with different viewpoints. A method to implement this algorithm consists of using a component tree. After consecutive binarization in the image with different thresholds, each region, defined by connexity, of each binarized image in the tree is arranged. Each level of the tree corresponds to a unique threshold value, and each level is composed of several image regions (nodes). An

[1] http://www.aquafadas.com/en/digital-publishing/comiccomposer/

[2] http://marvel.com/

978-1-4799-2341-0/13/$31.00 ©2013 IEEE

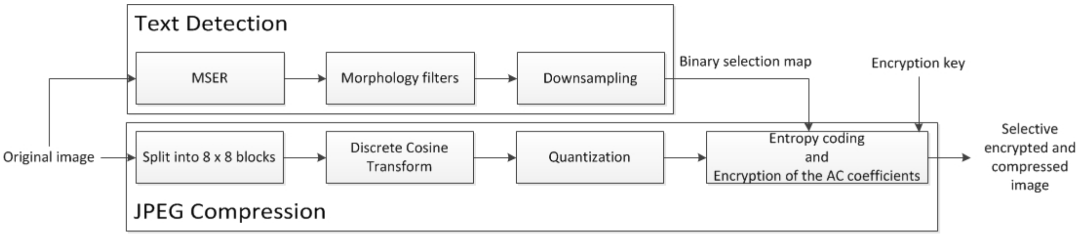

Fig. 1: Overview of the proposed method.

input parameter for this algorithm is step S, which defines all used thresholds T_i by the equation $T_i = S \times i$, while T_i is less than 256. So except for root nodes, which correspond to a region covering all images (since $T_0 = 0$, thus the binarized image contains only white pixels), each region (node) is included in his father. The second input parameter, δ, specifies if a region is a MSER region or not. A region is considered as a MSER region if it has approximately the same region size across 2δ neighboring threshold images. Several new techniques are based on MSER. For example, Chen *et al.* used the MSER algorithm and proposed to improve it by initially using the Canny edge detector and then filtering candidates using geometric and stroke width information to exclude non-text objects [6].

Multimedia data require full encryption or selective encryption depending on the application requirements [7]. Selective encryption of images and videos has two main advantages. First, it reduces the computational requirements, since only a part of the plain-text is encrypted [8]. Second, encrypted bit stream maintains the essential properties of the original bit stream. In the decoding stage, both the encrypted and the non-encrypted information should be appropriately identified and displayed. Selective encryption can be applied to different kinds of multimedia, for example Alattar *et al.* worked on MPEG videos [9] while Droogenbroeck and Benedett worked on images [10]. These selective encryption techniques may be used in spatial or frequency domains. For instance, Martin *et al.* [11] worked on the wavelet transform domain. Rodrigues *et al.* focused their research on selective encryption during JPEG compression to encrypt human faces in images or videos [12].

3. PROPOSED METHOD

In this section, we present our proposed method by firstly explaining text detection and secondly selective encryption during the compression step, and finally the decoding step. Fig. 1 presents an overview of the method.

We present first how the Region Of Interest (ROI), *i.e.* bubbles with text regions for our application, are detected. In e-comics, the text could be totally different. Indeed, it can appear in different colors, can have different sizes, and can also use different alphabets. To solve the text detection problem in such images, we decided to use the MSER algorithm as illustrated in Fig. 1. This technique has excellent results with dark text and bright homogeneous backgrounds, as is almost always the case in e-comics. Moreover, Nister and Stewenius proposed a method to implement the algorithm in linear time [13]. This detection selects some areas that are not text, but rather thin lines most of the time, representing facial features, for example. So we use morphological filters to remove such areas. We first apply a closing filter to join characters, secondly we apply an opening filter to remove areas without neighbors (because a character is never alone). Then this image is downsampled by 64 to match one pixel with an 8×8 JPEG block in the initial image. In this binary selection map, if a pixel is white this means that the corresponding 8×8 JPEG block contains text, so it has to be encrypted. Fig. 1 shows that this binary selection map is used during entropy coding of the JPEG compression for the encryption of AC coefficients.

We present now how ROIs are encrypted during compression. From the original image, as described in the JPEG standard, we split the image into 8×8 blocks and then apply the discrete cosine transform, quantization, and finally entropy coding step. As noted in Fig. 1, this step takes as input the binary selection map. Fig. 2 shows the selective encryption method for a block during this JPEG entropy coding. To make the text unreadable and encrypt the least bits possible, we decide to encrypt only the AC values of the luminance block, corresponding to high frequencies. Encryption is done using the AES algorithm. We use the stream cipher OFB mode of AES because it allows us to encrypt less than 128 bits, so each block is encrypted separately. After retrieving all AC values of a block, if the length is less than 128 bits, we pad with zeros and then encrypt the values. If the length is more than 128 bits, we encrypt the first 128 bits normally, and pad the remaining values with zeros and then encrypt them too.

Finally, we present how the decoding step is conducted. Firstly, we have to know which blocks are encrypted, and multiple solutions exist. For example, the easiest solution is to reuse the same matrix as that used during encryption which should be saved in a file. Of course this is not a good solution because the JPEG e-comics file has to be broadcast with this other attached file. Another solution we decided to use is to embed the matrix on the metadata of the JPEG file, but the metadata can be removed if the JPEG file is re-saved by

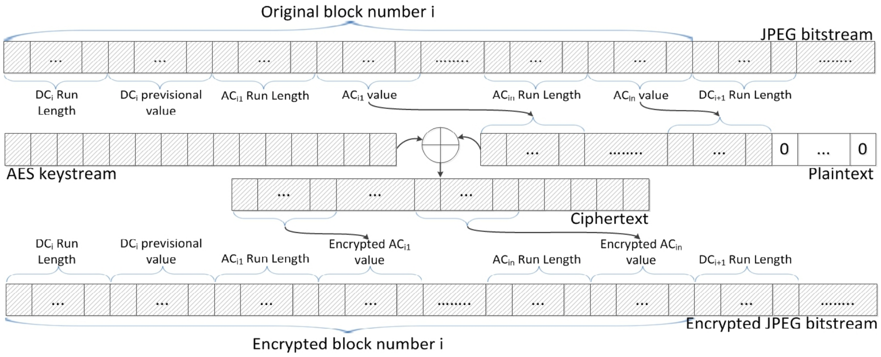

Fig. 2: Selective encryption method for a block using OFB mode of AES

another program. So in the decryption step, we just have to retrieve the binary selection map in the JPEG metadata. A good alternative to our solution would be to detect bubbles with non-encrypted information. Our method consists of encrypting the AC values of the luminance, so instead of using the MSER algorithm on the luminance of the image we could try to implement a method to detect bubbles from Cb and Cr components, for example. The detection method does not take into account the encrypted AC values, so the bubble detection on original image will have the same result as the bubble detection on the selective encrypted image. Once we know which blocks are encrypted, we just have to use the same method as encryption. After retrieving all AC values of an encrypted block, we decrypt all bits using the OFB mode of AES. Of course, the decryption key has to be the same as that used to encrypt.

4. EXPERIMENTAL RESULTS

In this section, we present our results. Firstly, we present the detailed results, and secondly an assessment of the parameters. Then results on the Japanese alphabet and finally the limitations of our technique are presented.

From the original image, in Fig. 3.a we apply the MSER algorithm to detect the text areas, as illustrated in Fig. 3.b. Note that some non-text areas are also selected. We apply morphological filters to remove these isolated areas. We apply 6 iterations of dilation to join all letters in a paragraph, followed by 10 iterations of erosion to remove small areas without MSER area neighbors, at the end 7 iterations of dilation to exceed the initial text size to be sure we encrypt all text blocks. Fig. 3.c illustrates the results of these filters, and we can note that most of non-text areas are removed. Fig. 3.d illustrates the original image (Fig. 3.a) overlaid by the downsampled image of Fig. 3.c, while blue squares represent blocks to encrypt. The results of our selective encryption during JPEG compression are presented in Fig. 3.e. The text of the bubble is no more understandable whereas the rest of the image was not changed. After decryption, the image is exactly the same as the original image (Fig. 3.a).

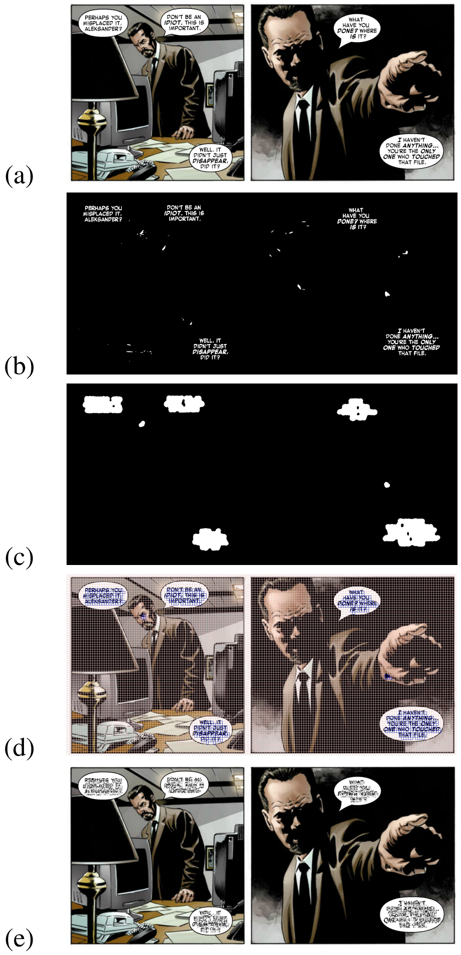
(a)
PERHAPS YOU MISPLACED IT, ALEKSANDER?
DON'T BE AN IDIOT. THIS IS IMPORTANT.
WELL, IT DIDN'T JUST DISAPPEAR, DID IT?
WHAT HAVE YOU DONE? WHERE IS IT?
I HAVEN'T DONE ANYTHING... YOU'RE THE ONLY ONE WHO TOUCHED THAT FILE.
(b)
(c)
(d)
(e)

Fig. 3: a) Original image (1347 × 626), b) Text detection with the MSER algorithm, c) Morphological filter result, d) Original image with text detection overlaid, e) Selective encryption result.

Text detection is a critical step and is very sensitive to parameters. In this part, we describe now how the step parameter S is selected. From the original image Fig. 4.a, we extract text manually, as illustrated in Fig. 4.b, and then we downsample this image (Fig. 4.c) to determine which JPEG block must be encrypted. We then study the influence of the step parameters S by comparing the results obtained with our algorithm (Fig. 4.d) and the ground-truth. Fig. 4.e illustrates the comparison: red pixels correspond to shared white pixels (true positive results), green pixels represent white pixels present only in the reference image (false negative results), blue pixels represent white pixels present only in our text detection (false positive results), and finally black pixels correspond to shared black pixels (true negative results). All of this information allows us to analyze the method with a Receiver Operating Characteristic (ROC) curve.

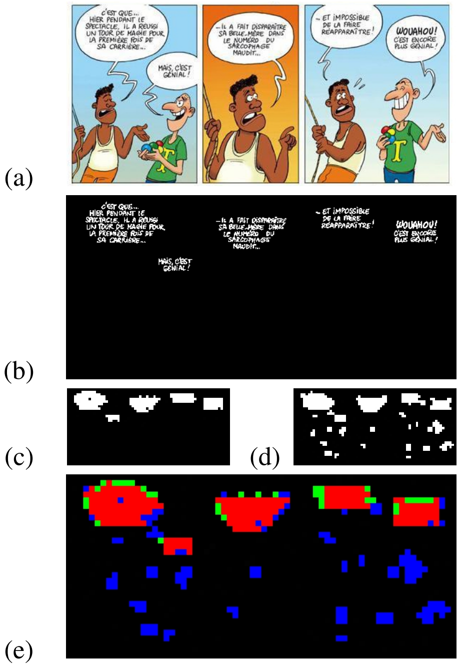
(a)
C'EST QUE... HIER PENDANT LE SPECTACLE, IL A RÉUSSI UN TOUR DE MAGIE POUR LA PREMIÈRE FOIS DE SA CARRIÈRE...
MAIS, C'EST GÉNIAL!
...IL A FAIT DISPARAÎTRE SA BELLE-MÈRE DANS LE NUMÉRO DU SARCOPHAGE MAUDIT...
...ET IMPOSSIBLE DE LA FAIRE RÉAPPARAÎTRE!
WOUAHOU! C'EST ENCORE PLUS GÉNIAL!
(b)
(c)
(d)
(e)

Fig. 4: a) Original image, b) Manual text extraction, c) Downsampled image b, d) Our text detection algorithm results with $S = 20$, e) Comparison between the manual extraction and our results.

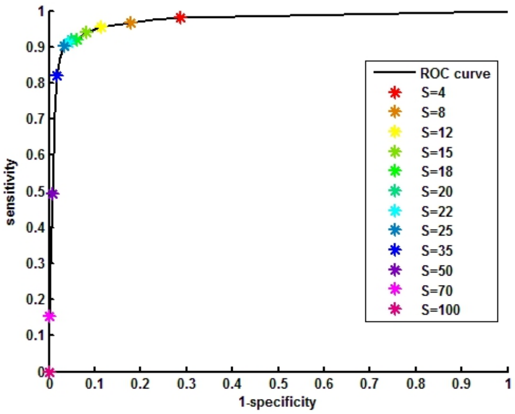

Fig. 5: Sensitivity analysis depending on parameter S.

Fig. 5 illustrates the ROC curve from an average of seven e-comics. The variable parameter is the input step parameter S of the MSER algorithm. From to this curve, we can choose the best parameter for our application. It is essential to encrypt the maximum number of blocks with text, which means that we want to minimize false negatives, so we have to maximize the sensitivity. We decided to set the parameters at 20, with this value the sensitivity is close to 0.92.

We present now some experiments carried out with the Japanese alphabet. Indeed, electronic mangas are very popular and the problem is that the Japanese text is vertical and the characters are more spaced. Fig. 6 shows some results obtained with manga images. Note that even with the Japanese alphabet our selective encryption method works.

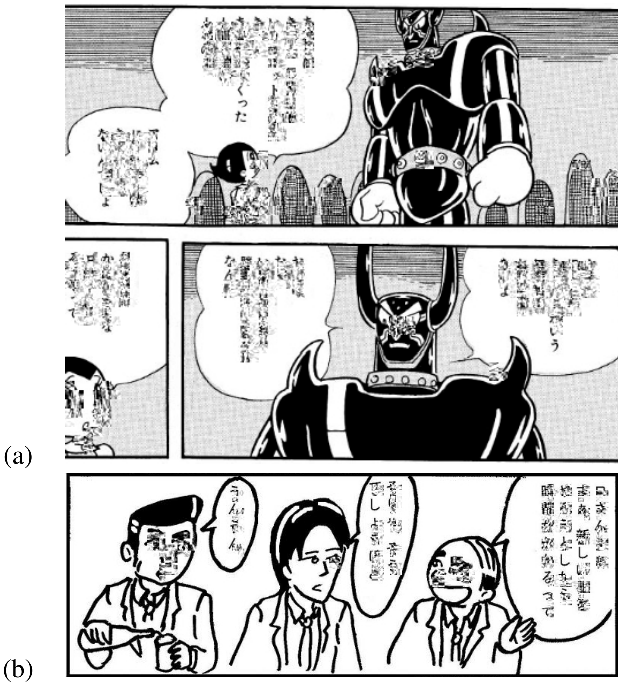

Fig. 6: Application of our method to e-mangas: a) Image from Astro Boy, b) Image drawn by Ming Zhujiang.

5. CONCLUSION AND PERSPECTIVES

In this paper, we have presented how a selective encryption method can be an efficient alternative to classic DRM systems for e-comics. In particular, as it is compatible with JPEG, it allows users to obtain a free overview of the e-comics and it may prompt them to buy the comics or make some publicity. Thanks to the totally reversible aspect of our method, customers will fully avail the e-comics. Our selective encryption method can also be applied for e-mangas, as we obtained good results with Latin and Japanese alphabets.

Our problem concerns the large text, but a solution could be to switch some neighboring JPEG blocks, moreover the PCBC (Propagating Cipher-Block Chaining) mode of AES allows users to decrypt even if two blocks are permuted. We now have to develop and analyze a new technique to recover the original position of each block.

6. REFERENCES

[1] C.Y. Su, R.I. Chang, and J.C. Liu, "Recognizing text elements for svg comic compression and its novel applications," *Document Analysis and Recognition (ICDAR), International Conference on*, pp. 1329–1333, 2011.

[2] K. Arai and H. Tolle, "Method for real time text extraction of digital manga comic," *Int Journal of Image Processing (IJIP)*, vol. 4, no. 6, pp. 669–676, 2011.

[3] M. Yamada, R. Budiarto, M. Endo, and S. Miyazaki, "Comic image decomposition for reading comics on cellular phones," *IEICE Transactions on Information and Systems*, vol. E87-D, no. 6, pp. 1370–1376, June 2004.

[4] L. Neumann and J. Matas, "Text localization in real-world images using efficiently pruned exhaustive search," *Document Analysis and Recognition (ICDAR), International Conference on*, pp. 687–691, September 2011.

[5] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and Vision Computing*, vol. 22, no. 10, pp. 761–767, Sept. 2004.

[6] H. Chen, S.S. Tsai, G. Schroth, D.M. Chen, R. Grzeszczuk, and B. Girod, "Robust text detection in natural images with edge-enhanced maximally stable extremal regions," in *Image Processing (ICIP), IEEE International Conference on*, Sept 2011, pp. 2609 – 2612.

[7] A. Pommer A. Uhl, *Image and Video Encryption - From digital Rights Management to Secured Personal Communication*, Springer, 2005.

[8] H. Cheng and X. Li, "Partial encryption of compressed images and videos.," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2439–2451, 2000.

[9] A.M. Alattar, G. Al-Regib, and S. Al-Semari, "Improved selective encryption techniques for secure transmission of mpeg video bit-streams.," in *Image Processing (ICIP), IEEE International Conference on*, 1999, pp. 256–260.

[10] M.V. 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*, 2002, pp. 9–11.

[11] K. Martin, R. Lukac, and K. N. Plataniotis, "Efficient encryption of wavelet-based coded color images," *Pattern Recognition*, vol. 38, no. 7, pp. 1111 – 1115, 2005.

[12] J.M. Rodrigues, W. Puech, and A.G. Bors, "Selective encryption of human skin in jpeg images," *Image Processing (ICIP), IEEE International Conference on*, pp. 1981–1984, 2006.

[13] D. Nister and H. Stewenius, "Linear time maximally stable extremal regions," in *Proceedings of the 10th European Conference on Computer Vision: Part II*, Berlin, Heidelberg, 2008, ECCV '08, pp. 183–196, Springer-Verlag.