Using Halftoning and Watermarking Techniques for Tampering Detection

Ronaldo Rigoni, Mylène C.Q. Farias, Pedro G. Freitas, and Aletéia P.F. Araújo

Abstract — In this paper we present a system with the goal of protecting and restoring tampered information in images and videos. With the proposed approach it is possible we implement a tampering detection algorithm. The proposed algorithm is based on watermarking and halftoning techniques. At the decoder side, the algorithm generates a binary version (mark) of the original image or video frame (picture) using a halftoning technique. Then, a watermarking technique is used to embed this mark into the host picture. At the decoder side, after the lost or tampered regions are identified, the original content is recovered by extracting the dithered mark corresponding to the affected areas. An inverse halftoning algorithm is used to convert the dithered version of the picture into a good quality colored multi-level approximation of the original content.

Index Terms — tampering, compression, halftoning, watermarking, data hiding, qim, forgery.

I. INTRODUCTION

The flexibility of digital images and videos is both a blessing and a curse. Digital technologies make it possible to create high quality pictures, animations, games, and special effects with an amazing realism. Digital pictures (images and videos) can be enhanced, compressed, transmitted, translated across different standards, and displayed in a variety of devices. Then, because of the significant advances in compression and transmission techniques, it is possible to deliver high quality visual content to the end user in many different ways. As a consequence, a variety of delivery services have been created in the last years, such as direct TV broadcast satellite, digital broadcast television, and IP-based video streaming.

A very important concern for image and video transmission and storage applications is tamper detection and copyright protection [6]. Powerful softwares are currently available, making it easy to alter (tamper) visual digital content without leaving any clear sign of these modifications. As a consequence, automatic methods for checking the authenticity and integrity of digital images and videos are, undoubtedly, very important. Several techniques have been proposed with the goal of detecting tampering of digital content [7]. These techniques can be divided in approaches that do not require the original (no-reference) and approaches that do require the reference. Since in most transmission applications the original is not available, techniques that do not require reference are the most adequate ones.

Most of the no-reference tampering detection techniques are specialized in detecting only one type of modification [8-9], what is not always useful in practical applications. One possible approach used by no-reference tampering detection techniques consists of using watermarking to embed invisible information into the host video [10-11]. To verify if the original content was tampered, the embedded information is extracted and its integrity is verified. In this approach, the “fragility” of the embedded mark is a key element that determines the amount of tampering that the algorithm is able to detect. Among the tampering detection algorithms that use this approach, we can cite the work of Wolfgang and Delp [10] that detects tampered areas by adding m-sequences of ‘−1’ and ‘1’ to 8×8 blocks, and the work of Yeung and Mintzer [11] that detects individual pixel modifications using a verification key. Among the methods available in the literature, few address the problem of detecting the location of tampered areas and restoring the original content with good quality [12]. Up to our knowledge, there are no watermark-based tampering detection algorithms for digital videos that is able to detect restored areas with a good quality.

In this paper, we present a system with the goal of protecting and restoring tampered information in images or videos. The system is based on watermarking and halftoning techniques. At the encoder side, the algorithm generates a binary version (mark) of the original image or video frame (picture) using a halftoning technique. Then, a watermarking technique is used to embed this mark into the host content. The watermarking technique used by the system is a simple modification of the Quantization Index Modulation (QIM) algorithm, which allows a slightly higher data hiding capacity [13]. For tampering detection, a ciphered key is also embedded into the host video to allow spatial and temporal localization of tampered regions. At the decoder side, after the tampered regions are identified, the original content is recovered by extracting the dithered mark corresponding to the affected areas. An inverse halftoning algorithm is used to convert the dithered version of the picture into a good quality colored multi-level approximation of the original picture.

The paper is divided as follows. In Sections II and III, the halftoning and watermarking embedding stages are described. In Sections IV and V, the watermarking extraction and inverse halftoning stages are detailed. In Sections VI, we describe the implementation and simulation results of the detection algorithm. Finally, in Section VIII the conclusions are discussed.

---

*This work was supported in part by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) - Brazil and in part by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - Brazil.*

The authors are currently with the University of Brasília (UnB), Campus Universitário Darcy Ribeiro, 70919-970, Brasília - DF, Brazil. (e-mails: mylene@ieee.org, rrigoni@gmail.com, sawp@sawp.com.br, mmcarnalho@unb.br, aleteia@cic.unb.br).
II. HALFTONING STAGE

Halftoning is a technique for converting multi-level images into binary images using patterns of white and black dots [14]. This technique creates the illusion of seeing multiple intensity levels in a binary image, what makes it suitable for applications where only a reduced number of levels is available, such as newspapers, fax machines, and document printing processes.

In this work, a halftoning algorithm is used to generate a dithered version of each picture (still image or video frame), which is later embedded into the host picture itself. At the decoder, if any loss or tampered area is detected, the dithered version of the picture is recovered and used to restore the content back to its original state. Therefore, the quality of the restored image or video depends strongly on the efficiency of the halftoning algorithm. One of the contributions of this work is the design of a halftoning and an inverse-halftoning algorithms that are able to generate “simple” dithered images that can be later inverted with a very good quality.

In order to guarantee a good quality, the halftoning algorithm must be able to represent the largest possible number of intensity levels with minimum number of bits. Also, since (at the decoder/receiver side) parts of the picture might be tampered, the halftoning algorithm needs to use only local information to generate the dithered picture and revert it. The simplest halftoning technique that satisfies these requirements is the ordered dithering algorithm. This class of algorithms generates dithered pictures with sets of pixel clusters that have a predictable pattern, what is very important for our target applications (transmission and storage of videos and images).

Given that the data hiding capacity of the host picture frame is relatively low, we do not use classical ordered dithering algorithm. We propose a combinatorial dispersed-dot patterns because they are capable of generating all combinations of bits necessary to represent a number of intensity levels [14-15]. This way, we can increase the number of mapped intervals without increasing the size of dot-pattern matrix. For example, using a 3x3 Bayesian matrix to generate a dispersed-dot dithering we can represent 10 distinct intensity levels. On the other hand, using a 3x3 combinatorial matrix allows for up to 512 intensity levels. Since our tests showed that a maximum of 3 bits per pixel (per color channel) can be embedded in the host picture without causing visible degradations, we can use combinatorial matrices to generate $2^3$ combinations, allowing mapping up to 8 different intervals.

Since pictures reconstructed from dispersed-dot dithered images can be slightly blurred, before we generate the halftoning picture we apply an unsharp-masking edge enhancement filter to the original picture frame ($I_o$), generating an image with enhanced details ($I_{eh}$). Then, we quantize $I_{eh}$ using 8 distinct intervals, using the following equation:

$$I_Q(x, y, c) = \left\lfloor \frac{8}{255} I_{eh}(x, y, c) \right\rfloor$$

in which $x$ and $y$ are the horizontal and vertical spatial dimensions, respectively, $c$ refers to the color channel ($1 \leq c \leq 3$), and $I_Q$ is the resulting dithered image. As mentioned before, up to 3 bits can be embedded in each color channel. This means that the dithered mark ($I_Q$) must be represented with a total of 9 bits per pixel. So, we substitute each value of $I_Q(x, y, c)$ by the corresponding 3-bits combinatorial dispersed-dot patterns shown in Figure 1(a).

In case the target application requires that the spatial and temporal position of the lost or tampered areas be identified before restoration, 1 bit out of the available 9 bits should be used to store a ciphered key code. In these cases, we are left with 8 bits to represent the dithered version of each color channel of the image or video frame. Since the Human Visual System (HVS) is less sensitive to degradations in the blue channel, we opt to use 2 bits to represent the dithered version of the blue channel and 3 bits to represent the dithered version of the red and green channels. In other words, for the red and green channels, we substitute each 8-bit pixel of $I_Q$ by one of the 3-bits dot-patterns shown in Figure 1(a), while for the blue channel we substitute each 8-bit pixel of $I_Q$ by one of the 2-bits dot-patterns shown in Figure 1(b).

Before embedding this information into the host picture, we concatenate the resulting 3 bits corresponding to each pixel and generate an integer number. This way, the resulting halftoning image is actually an “integer dithered image” ($I_{ds}$), with values ranging from 0 to 7. This integer mark can be easily embedded in the original image, without requiring extra space.

![Figure 1: Combinatorial dispersed-dot patterns: (a) 3-bits dispersed-dot patterns used for mapping 8 intensity levels and (b) 2-bits dispersed-dot patterns used for mapping 4 intensity levels.](image)

III. WATERMARKING EMBEDDING STAGE

After generating the mark using the techniques described in the previous section, the next stage consists of embedding it into the host image or video. But, in order to make it possible to recover the original content, the dithered mark corresponding to a specific region cannot be embedded in the same spatial and temporal position of the host image or video. Therefore, we spatially distribute the mark over the host image or video frame using a split-flip operation, which consists of splitting the halftone image into sub-blocks and flipping them to a different spatial region. More specifically, we divided the picture in 32x32 sub-blocks, rotated each sub-block by 180 degrees, and shuffled the regions. Figure 2 shows one example of the process for the three color channels.
of the image “Lena”.

For videos signals, we also perform an embedding temporal distribution by inserting the mark corresponding to the current picture frame in a previous picture frame, located 1 second before. By distributing the mark spatially and temporally, a region does not store the mark necessary to restore it, what increases the probability that the algorithm is able to restore the content to its original version. The mark is also encrypted with AES-256 block Cipher to protect it from being extracted by an unauthorized user [16].

Among the available watermarking methods, the Quantization Index Modulation (QIM) algorithm is one of the methods with the best performance [13]. The QIM algorithm inserts a mark into a host signal by quantizing it with a uniform scalar quantizer. The standard quantization operation with step size \( \delta \) is given by the following equation:

\[
Q(x, \delta) = \text{round} \left( \frac{x}{\delta} \right),
\]

where \( \text{round}(\cdot) \) denotes the mathematical operation of rounding a value to the nearest integer. The watermarked pixel is obtained using the following equation:

\[
s(x) = Q(x, \delta) + d(m),
\]

in which \( d(m) \) is the perturbation value, which depends on the mark signal \( m \) to be embedded.

In this work, we propose a modification of the QIM algorithm that inserts an integer mark, instead of a function of the 1-bit mark. Also, the modulation function is set to \( d(m) = m \). So, the integer dithered image (I_{dth}) is inserted in each pixel of the corresponding color channel of the original picture using the following equation:

\[
I_{m}(x, y, c) = Q \left( I_{o}(x, y, c), \delta \right) + I_{dth}(x, y, c),
\]

where \( I_{o} \) is the original color channel of the picture frame, \( I_{m} \) is the resulting watermarked color channel, \( \delta \) is the quantization step, and \( c \) is the corresponding color channel.

To detect if a region of the image or video is tampered, we compare the cipher detection key with the secret key contained in the mark extracted from every pixel of the blue channel of the dithered image. If differences are found, the pixels where the differences were found are classified as “tampered”. In applications like error concealment, where the position of lost areas are identified by the decoder, the key is not needed and the extra bit can be used to convey a better quality for the reconstructed areas.

**V. INVERSE HALFTONING STAGE**

If any region is classified as lost or tampered, we search the correspondent mark in the appropriate location. Then, to generate a multi-level colored picture frame, we propose the following inverse halftoning algorithm [15]. Given that \( I_{dth} \) is the dithered picture frame, \( D(p) \) is the distribution of the area surrounding the pixel \( p \) in \( I_{dth} \). To reconstruct an 8-bit pixel from the dithered picture, we first calculate the local distribution \( D(p) \) for all pixels in \( I_{dth} \). From this distribution, we find the corresponding mapped interval that contains the most probable pixel value in the corresponding color channel, according with the indices of the dot-patterns.

Once this interval is found, we randomly select a value within it, generating a slightly noisy picture \( I_{inv} \). Then, we filter \( I_{inv} \) with 2 filters: a Gaussian lowpass and a Laplacian-of-Gaussian. The idea here is to spatially decorrelate the pixels in order to be able to process each pixel independently. This allows for an independent reconstruction, even when the neighboring pixels are missing. The resulting pictures, \( I_{gauss} \) and \( I_{log} \), are used to compose another picture, \( I_{bld} \), given by the following equation:

\[
I_{bld}(x, y, c) = \gamma I_{gauss}(x, y, c) + (1 - \gamma) I_{log}(x, y, c),
\]

where \( \gamma \) is the blending-ratio matrix that determines the proportion of each input filtered picture in the output. In our simulations, we observed that using \( \gamma = I_{inv} \) preserves the edges of the pictures.

Unfortunately, when we combine all 3 channels, the resulting picture frame contains visible color distortions. If, on the other hand, we use \( \gamma \) as a constant matrix, \( I_{bld} \) is a blurred version of \( I_{inv} \). Therefore, with the goal of minimizing color distortion and keeping the details of the original image, it is necessary to make another composition of \( I_{inv} \) and \( I_{bld} \),

\[
I_{dth}(x, y, c) = I_{m}(x, y, c) \mod \delta,
\]

where \( I_{m} \) is the watermarked color channel and \( I_{dth} \) is the recovered integer dithered mark. Then, the extracted integer dithered mark is converted to a 3-bit binary number that results in the recovered dithered picture.

Fig. 2: Original dithered versions of each color channel of the image Lena (left) and the corresponding split-flip versions (right).
given by the following equation:

\[
\hat{I}_o(x, y, c) = \left[ \sqrt{I_{col}(x, y, c)I_{bld}(x, y, c)} \right],
\]

where \( \hat{I}_o \) is the recovered 8-bit version of original video frame, \( I \). The final result of inverse halftoning process, \( \hat{I}_o \), is a picture the best visual quality possible, as compared to the techniques available in the literature. Obviously, the procedure can be applied to videos and images alike.

VI. TAMPERING DETECTION ALGORITHM

As pointed in the Section I, tampering detection of video and image content is also a crucial problem in multimedia applications. We used the techniques described in Sections I-V to design a tampering detection algorithm that is able not only to detect tampered regions, but also to recover the original content. For that, we used one of the 9 bits embedded in the host image or video frame to store a secret key code (see Section II). As described earlier, the dithered versions of the red and green channel use 3 bits each, while the dithered version of the blue channel uses only 2 bits.

First, we tested the proposed algorithm using still images with different characteristics: high-detailed, low-detailed, color, grayscale, documents, landscape, person pictures, etc. Different kinds of attacks were applied to these images: blurring of selected/random areas, noise addition, cut-and-paste, region deletion, and resizing. For all test cases, we were able to detect tampered regions in 100% of the cases. Also, there were no false-positives or false-negatives. In terms of reconstruction, the algorithm was able to recover tampered regions with good quality, as long as the tampered regions did not exceed 50% of the image.

Figures 9, 10, and 11 depict three examples of the use of the proposed algorithm to detect tampered regions in still images. The first row of the examples (from left to right) shows the original (untampered) images, the tampered images, and the regions detected as tampered. The second row (from left to right) shows the recovered tampered regions, an image recovered without the key, and the image reconstructed with original content restored. As can be observed, the algorithm is able to detect tampered regions, independent of their size. Also, since the embedded half-tone watermark is encrypted, an unauthorized user is unable to know if an image was tampered and, consequently, unable to reconstruct the image back to its original state (see Figures 9(e), 10(e), and 11(e)). The algorithm is able to restore content of tampered images with good quality, as long as at least 50% of the regions of the embedded half tone are intact (see Figures 9(f), 10(f), and 11(f)).

We also tested the performance of the algorithm for tampering detection and recovery of digital videos. As in the previous examples, we used publicly-available videos in YUV 4:4:4 color CIF (352 x 288, progressive) format with around 300 frames each, downloaded from the Video Trace Library.
We tested the following attacks: blurring of selected areas, cut-and-paste, region-deletion, and object-addition. Again, for all test cases, we were able to detect tampered regions in 100% of the cases. There were no false-positives or false-negatives. In terms of reconstruction, the algorithm was able to recover tampered regions with good quality, as long as the tampered regions did not exceed 50% of the frame.

Figures 12, 13, and 14 depict examples of the use of the proposed algorithm to detect tampered regions for 3 different videos and 3 different types of attacks: blurring of selected areas, cut-and-paste, and object-addition. In these figures, the first row (from left to right) shows the original (un-tampered) and the tampered frame, while the second row (from left to right) shows the recovered tampered regions and the frame with the original content restored. Notice that the algorithm is able to detect tampered regions, independent of their size, position or the level of difference in comparison to the original. Also, the proposed algorithm is able to restore content of tampered images with very good quality.

I. CONCLUSIONS

In this paper we present a system with the goal of protecting and restoring tampered information in images or videos using a tampering detection algorithm. The system is based on watermarking and halftoning techniques. In order to increase the data hiding capacity, the work proposes a simple modification of the QIM watermarking algorithm. To obtain higher quality restored areas, a halftoning and inverse halftoning algorithms are proposed. A secret key code is embedded to the host content to identify the spatial and temporal positions of tampered regions, taking advantage of the lower sensitivity of the HVS to degradations in the blue color channel. The proposed algorithm presents good performance, being able to identify tampered or lost areas and restore them with very good quality.

Future works include an increase of the data hiding capacity with the goal of embedding more information. In particular, the quality of the restored content can be increased by adding more bits to represent the dithered version of the picture. Additional bits can also be used to improve the protection of the data against tampering. For example, using some bits to embed additional temporal information can help counter other attacks, such as frame shuffle.

REFERENCES

Mylène C.Q. Farias (M’01) received her B.Sc. degree in electrical engineering from Universidade Federal de Pernambuco (UFPE), Brazil, in 1995 and her M.Sc. degree in electrical engineering from the Universidade Estadual de Campinas (UNICAMP), Brazil, in 1998. She received her Ph.D. in electrical and computer engineering from the University of California Santa Barbara, USA, in 2004 for work in no-reference video quality metrics. Dr. Farias has worked as a research engineer at CPqD (Brazil) in video quality assessment and validation of video quality metrics. She has also worked for Philips Research Laboratories (The Netherlands) in video quality assessment of sharpness algorithms and for Intel Corporation (Phoenix, USA) developing no-reference video quality metrics. Currently, she is an Assistant Professor in the Department of Computer Science at the University of Brasilia (UnB), where she is a member of the Graduate Program in Informatics and of the Graduate Program on Electronic Systems and Automation Engineering (PGEA). Dr. Farias is a researcher of the Laboratory of Images, Signals and Audio (LISA) and her current interests include video quality metrics, video processing, multimedia signal processing, watermarking, and information theory. Dr. Farias is a member of IEEE and of the IEEE Signal Processing Society.

Ronaldo Rigoni received his B.Sc. degree in Systems Information from Faculdade Michelangelo, Brazil, in 2010. He works as a Software Architect for the Brazilian Army. He is currently pursuing a M.Sc. in Computer Science in the University of Brasilia (UnB), Brasilia. His interests are parallel computing models, machine learning, distributed and parallel video coding, optimization, signal processing, and programming languages.

Aletéia Patrícia Favacho de Araújo received her B.Sc. degree in Computer Science from the Universidade Federal do Pará (UFPA), Brazil, in 1997 and her M.Sc. degree in Computer Science and Mathematics from the Universidade de São Paulo (USP), Brazil, in 1999. She received her Ph.D. in Informatics from the Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Brazil, in 2008 for work in grid computing. Currently, Dr. Araújo is an Assistant Professor in the Department of Computer Science at the University of Brasilia (UnB). Her current interests include cloud computing, grid computing, parallel and distributed algorithms, and scheduling on distributed platforms.