Joint near-lossless compression and watermarking of still images for authentication and tamper localization

Roberto Caldelli [a,*] Francesco Filippini [a] Mauro Barni [b]

[a] *Department of Electronics and Telecommunications, University of Florence, Florence, Italy*

[b] *Department of Information Engineering, University of Siena, Siena, Italy*

Abstract

A system is presented to jointly achieve image watermarking and compression. The watermark is a fragile one being intended for authentication purposes. The watermarked and compressed images are fully compliant with the JPEG-LS standard, the only price to pay being a slight reduction of compression efficiency. Watermark detection is possible both in the compressed and in the pixel domain, thus increasing the flexibility and usability of the system. The system is expressly designed to be used in remote sensing and telemedicine applications, hence we designed it in such a way that the maximum compression and watermarking error can be strictly controlled (near lossless compression and watermarking). Experimental results show the ability of the system to detect tampering and to limit the peak error between the original and the processed images.

Key words: Image authentication, near-lossless JPEG, digital watermarking, tamper localization, remote sensing, medical imagery.

1 Introduction

The demand for image authentication and for effective means to control image integrity has been steadily increasing in the last years. Such a demand is due to the ease with which digital images can be tampered with thus compromising

* Department of Electronics and Telecommunications, University of Florence, Via di Santa Marta, 3, 50139, Florence, Italy.
Ph. +390554796380, Fax +39055494569, e-mail caldelli@lci.det.unifi.it

their credibility as faithful pictures of the scene they represent. Several techniques have been developed in order to prevent or at least detect unwanted alteration of digital images. Among them, digital watermarking has gained more and more popularity due to its versatility and its potential to localize tampering and the possibility (at least theoretical) to distinguish between different kinds of manipulations (usually split into allowed and not allowed manipulations). Two possible approaches can be distinguished, one based on (semi) fragile watermarking and the other relying on robust watermarking. Authentication through fragile watermarking [1,2] is accomplished by inserting within the image a watermark that is readily altered or destroyed as soon as the host image undergoes any manipulations. The alteration or deletion of the watermark allows to discover that the image has been modified, whereas the correct recovery of the hidden information permits to prove the integrity of the image and, possibly, to establish its origin. Some techniques permit also to localize the altered zones on a block basis [6,7]. Systems based on robust watermarking [8,11] assume that the watermark is not affected by image manipulations. Specifically, a summary of the to-be-authenticated image is computed and embedded within the image itself (possibly together with additional information about the origin of the image). Subsequently, the hidden information is recovered and compared with the actual content of the image: a mismatch reveals that the image was tampered with.

In this paper we focus on authentication and tamper localization through fragile watermarking. Specifically, our system is built by relying on a scheme by J. Fridrich [2] that embeds the watermark in the Least Significant Bits (LSB) of the host image. The choice of Fridrich's algorithm is justified by its security features and its good localizing capabilities (more details on this scheme are given in section 2).

Together with the demand for integrity verification the demand for image compression is everyday more pressing. The great majority of the images exchanged in digital format are stored in a compressed format, with lossy compression being definitely much more popular than lossless compression. Hence, a first crucial choice must be made to decide whether to embed the watermark in the raw domain (i.e. before compression takes place) or in the compressed domain (e.g. by jointly coding and watermarking the image). In the context of image authentication through fragile watermarking, joint coding and watermarking is highly desirable, since otherwise the fragile nature of the watermark will identify image compression as an unwanted manipulation hence failing to distinguish between (allowed) compression and (not allowed) tampering. On the other side, tying the watermarking system to a particular coding format limits the flexibility of the authentication scheme, since the watermark is likely not to survive lossless format changes, e.g. conversions from the coded and the raw format. It is one of the goals of the system presented in this paper to embed the watermark in the compressed domain, while still

allowing the recovery of the watermark in the raw pixel domain.

Though lossy compression is by far the most popular coding strategy used today, in some application scenarios the loss of information accompanying the compression process can not be tolerated or, at least, must be strictly controlled. This is the case of remote sensing and medical applications. In both cases the risk of discarding useful information calls for the adoption of lossless compression, however the huge amount of data acquired by sensors during earth observation missions and the large volume of images produced by modern telemedicine applications [15,16] make the use of efficient lossy coding algorithms unavoidable. In order to control the amount of information lost during the compression process, a class of algorithms capable of strictly controlling the compression loss have been devised and grouped under the term Near-Lossless compression, whose main requirement is that of ensuring that the maximum error between the original and the compressed image does not exceed a fixed threshold. In the same line, the concept of near-lossless watermarking has been introduced recently to satisfy the strict requirements set by the remote sensing scenario [3,5]. In this paper we propose a system that permits to jointly compress and watermark the to-be-protected image in a near-lossless fashion, thus resulting particularly suited for remote sensing and medical applications.

Specifically, Fridrich's authentication algorithm [2] is modified so to make it compliant with the JPEG-LS coding standard. JPEG-LS [9,10] is a lossless/near-lossless image coding scheme based on differential pulse code modulation (DPCM) [12]. In the near-lossless mode each pixel of the reconstructed image differs from the corresponding original pixel by up to a preset (usually small) amount, called NEAR in the following. By slightly modifying the quantization process, our system is able to embed an LSB message similar to that used by Fridrich directly in the compressed domain, thus keeping complete compliance with JPEG-LS. At the same time, the maximum amount of distortion introduced by the watermark can be strictly controlled thus satisfying the near-lossless requirement. As already said, the watermark can be recovered both in the compressed and in the raw domain, thus increasing the flexibility of the system and its practical usability. Finally, the security features of Fridrich's algorithm are retained together with its localizing properties (the localization accuracy being reduced only slightly).

The rest of the paper is organized as follows. In section 2 the proposed watermark embedding algorithm is described. In section 3, watermark detection is considered. In section 4 security issues are discussed. Section 5 is devoted to the presentation of experimental results. Finally, some conclusions are drawn in section 6.

2 Encoding phase

As we already said, the main goal of the new watermarking scheme is to grant robustness against near lossless JPEG image compression, while maintaining the usual features of an authentication technique. This aim is achieved by designing a system which is based on JPEG-LS coding standard in order to generate compressed data and, by simultaneously realizing, the authentication of these compressed data with the integration of a secure fragile watermarking technique, that in our approach has been individuated in a technique developed by Fridrich [2].

In the first encoding step, image decorrelation is obtained by determining the prediction error using the same approach adopted in JPEG-LS algorithm. In this way, the input image is scanned left to right and top to bottom by successive lines to produce a sample sequence. Let us indicate with Ix the brightness of the pixel x and with Px the predicted value.

After this procedure, the prediction error $Errval = Ix - Px$ is computed and, in the near-lossless coding ($NEAR > 0$), this error is quantized ($qErrval$) according to the following rule (1):

$$qErrval = \left\lfloor \frac{Errval + NEAR}{2NEAR + 1} \right\rfloor \tag{1}$$

where $NEAR$ is the maximum guaranteed preset error between the original and the compressed images.

2.1 Watermark generation

Let us first summarize how the watermark is generated in the secure fragile watermarking technique developed by Fridrich [2] (a block diagram of this approach is given in Figure 1). During watermark embedding, the algorithm proceeds by dividing the image into 8×16 pixel blocks and by separately modifying the LSBs of each block. To do so, the seven Most Significant Bits (MSBs) of the pixels in the block are hashed by using a proper hash function. Then, a binary logo carrying information about the block position, image index and possibly other information relevant to the image is constructed, and is XORed with the hash. After that, the XORed result is encrypted using a secret-key dependent encryption function, and inserted into the LSBs of the same block.

In the watermark detection phase, the to-be-authenticated image is divided

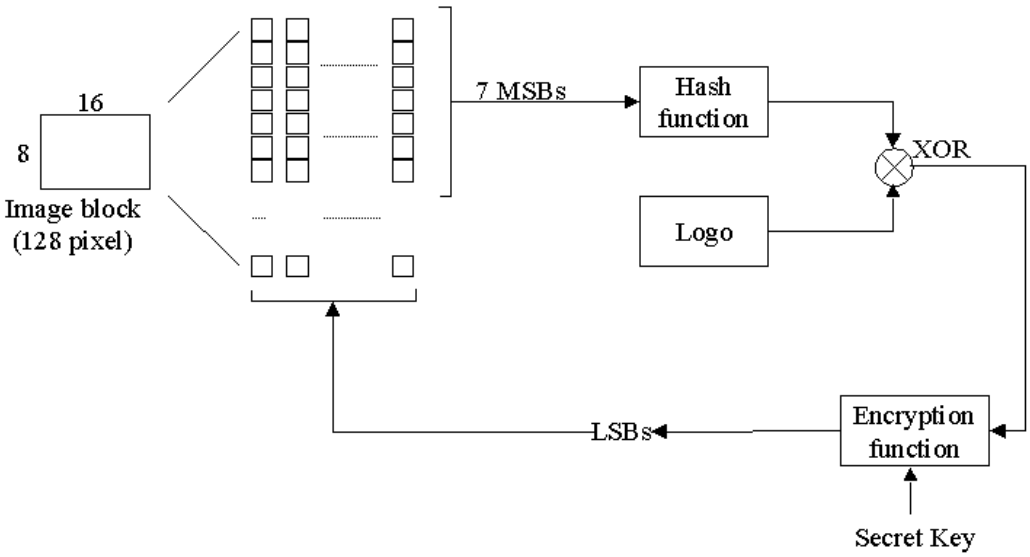


Fig. 1. *Watermark embedding diagram of Fridrich's algorithm [2].*

again into 8×16 blocks and for each block the following procedure is applied. The seven MSBs of each pixel are extracted and hashed, while the LSBs are decrypted by using the secret key. In the end, the hashed MSBs and the result of LSBs decryption are XORed to obtain back the logo. Through an automatic examination of the logo block-wise image authentication can be achieved. In this way, the watermarking scheme is robust to authentication attacks, such as Stego-Image Attack, Multiple Stego-Image Attack and Holliman-Memon Attack [6,7] (see Section 4); furthermore, localization of image tamper is also granted.

By taking into account the JPEG-LS and Fridrich's algorithms, we developed a watermarking system that allows a near-lossless compression of the image and, at the same time, permits to insert a watermark into the to-be-authenticated image. To do so, the encoding procedure of the JPEG-LS algorithm has been modified in order to integrate the watermarking system while maintaining compliance with JPEG-LS standard.

2.2 Watermark embedding phase

The second step, integrated within the JPEG-LS algorithm, is devoted to watermark embedding on the basis of Fridrich's algorithm. The quantized prediction errors are modified in order to insert the watermark into the image and finally, the corrected quantized prediction errors are Golomb-Rice coded and the compressed image obtained. More specifically, we proceed as follows. Let us consider an image of $D_R \times D_C$ pixels, composed by blocks each of 8x16 pixels (i.e. $\frac{D_R}{8}$ stripes of blocks). For the first stripe S_1, the reconstructed

samples Rx are computed and stored to form a reconstructed sample stripe RS_1, which is made by $\frac{D_C}{16}$ blocks each of 8x16 pixels. Then, each reconstructed sample block is processed by the watermarking system whose output is a 8x16 binary matrix (the authenticating message). When all the reconstructed sample blocks of the reconstructed stripe RS_1 have been processed, an $8 \times D_C$ binary stripe BS_1 is created. At the end of this process, for each sample in position *(i,j)* in the second stripe S_2 of the image, the quantized prediction error is calculated. Then, in order to insert the watermark into the image pixel in position *(i,j)*, the quantized prediction error has to be modified by altering its LSB according to the correspondent bit of the authenticating message of the previous stripe.

It is crucial to take into account this change also in the value of the reconstructed pixel Rx (see Equation 2 where $qErrval^{-1}$ states for the dequantized prediction error) to perform a computation that is equal to that the JPEG-LS decoder will make during the decoding phase.

$$Rx = Px + qErrval^{-1} \tag{2}$$

In fact, during the decoding phase, the predictor Px must be calculated on the same values of the reconstructed pixels otherwise it would result different than that computed in the coding phase being the reconstructed pixels modified because the quantized prediction error has been changed to allow watermark insertion. According to this consideration, the parity of Rx has to be checked before performing any modification. For sake of clarity, let us make an example and let us suppose that Rx assumes an odd value and that a bit 0 has to be inserted (if a bit 1 has to be embedded no action is needed). To do this, the original $qErrval$ is augmented or decreased by one quantization level to change its parity to obtain a $qErrvalmod = qErrval \pm 1$. By applying dequantization we obtain:

$$qErrvalmod^{-1} = qErrvalmod * (2 * NEAR + 1) \tag{3}$$

and then

$$Rxmod = Px + qErrvalmod^{-1} \tag{4}$$

Being the quantization step $(2 \times NEAR + 1)$ an odd value, the modified parity is transferred to $qErrvalmod^{-1}$ and consequently to $Rxmod$ as required.

What is important is that it has been obtained a reconstructed sample Rx *that contains the authenticated LSB related to the element in position* (i,j) *of the binary stripe BS_1 previously computed.*

Figure 2 summarizes the steps of the procedure for the authentication of the samples which belong to the first stripe. First of all, it is possible to notice the *Authentication* block which performs the hash of the 7 MSBs and the LSBs encryption for each image block in order to generate the binary stripe BS_1. Consequently, the prediction error of the successive stripe is computed, quantized and modified. Through the block that compares the difference between Rx and Ix, it is possible to choose the best modified prediction error, which limits the *MaxError* between the original and the authenticated image.

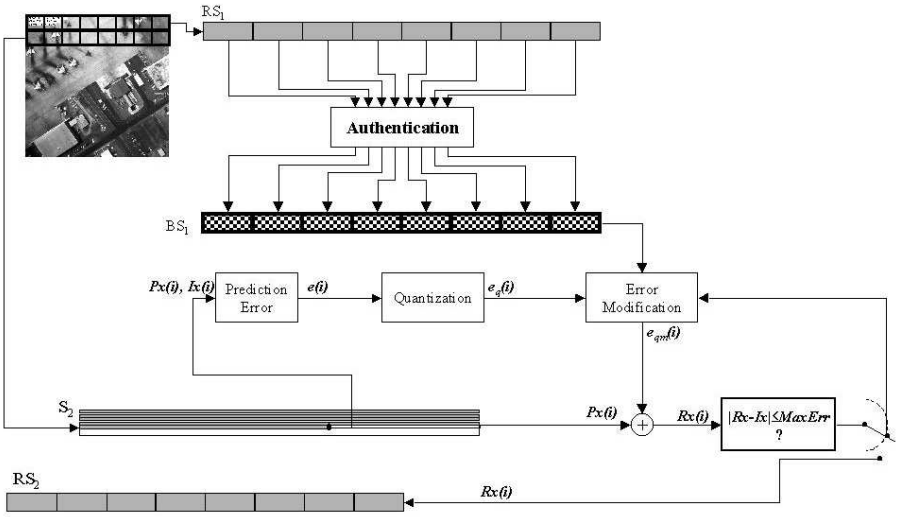


Fig. 2. *Stripe watermark embedding scheme*

The main idea of this approach is to opportunely modify the quantized prediction error in such a way that the LSB of the successive reconstructed value Rx is related to the respective element of the previous binary stripe BS_1. Roughly speaking, the authentication information of a stripe is embedded into the reconstructed samples of the stripe below. Finally, the reconstructed value Rx is stored to form the second reconstructed samples stripe RS_2, whereas Golomb-Rice coding of the modified quantized prediction error is performed. When all reconstructed samples Rx of the second stripe S_2 have been computed and stored, the RS_2 has been constructed. The result of this process is RS_2 that has been modified according to BS_1. Generally, by following this procedure each RS_{i+1} is modified on the basis of BS_i and the watermarked-compressed image is generated. It is important to note that, in this system, the authentication binary matrix BS_i is embedded into the subsequent reconstructed sample stripe RS_{i+1}. This approach has been adopted because JPEG-LS is based on a sequential procedure, whereas Fridrich's algorithm works block-wise. In the JPEG-LS algorithm, for each sample of the input image the corresponding

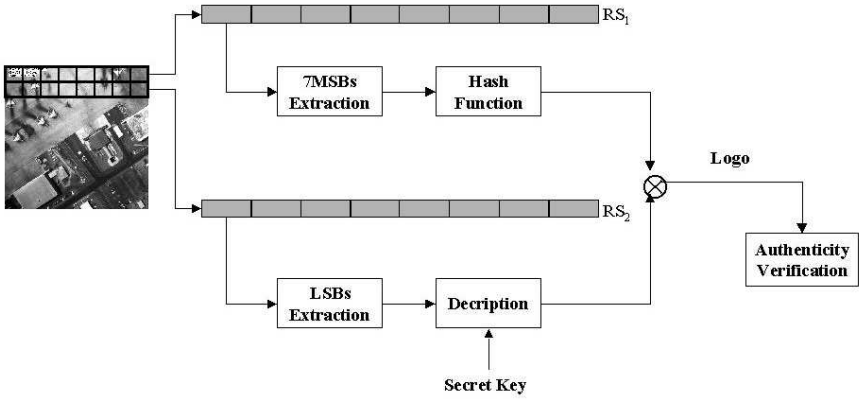


Fig. 3. *Watermark detection scheme: the scheme represents the block-wise procedure that is followed to check image authenticity.*

reconstructed sample is found. If we desire to watermark this reconstructed value using Fridrich's algorithm, the binary matrix must be calculated previously; at the same time, this binary matrix can be only computed if all the reconstructed samples belonging to the block are known. This requirement contrasts with the sequential flow of JPEG-LS.

3 Watermark detection

Watermarked compressed data, generated during watermarking embedding phase, can be decoded using a standard JPEG-LS decoder.

In order to describe the authentication process let us consider, as we did for the coding phase, a $D_R \times D_C$ image. Watermark detection starts by dividing the image into 8-row stripes each consisting of 8×16 pixel blocks, as in the embedding phase. Then, for each image stripe the following procedure is applied. First of all, in order to verify integrity of the first image stripe S_1, the second stripe S_2 must be known to extract the LSBs and to complete the watermark detection (see Figure 3). For each image block belonging to the first stripe S_1, the verification procedure continues as in Fridrich's algorithm. The 7 MSBs are extracted and then hashed. At the same time, the LSBs of the corresponding image block in the second stripe S_2 are extracted and decrypted by using the same secret key used by the embedder. Finally, the

hashed data and the decrypted LSBs are XORed and the authenticating logo is found. The information carried by the Logo permits to verify the authenticity of each image block. A similar approach is followed for the subsequent stripes. In general, by analyzing two consecutive stripes it is possible to check each image stripe and in the end to check if the image is authentic as a whole or which part (blocks) of it has been manipulated.

4 Security issues

Security issues play a central role in watermarking-based authentication. In fact, content authenticity can be compromised by an ad-hoc action puts in place by an attacker who wants to create a fake document by resorting to all the information and capabilities available to him. It is important that an authentication algorithm is robust not only when a hacker has a unique image at his disposal (*Stego-Image Attack*) but also when he can access other supplementary knowledge; hereafter some of the main security attacks against watermarking-based authentication are listed.

- *Multiple Stego-Image Attack* The counterfeiter has many authenticated documents and his action aims at making changes in such a way that the detector cannot reveal them or at gaining knowledge about the secret keys used by the scheme. A particular application of this attack is well known as the *Holliman and Memon Attack* [6].
- *Verification Device Attack* The aim of the counterfeiter is the same as before, but, in this case, he has access to the verification device and can use it to check the integrity of any image he likes. On the basis of the answer he gets he can rearrange the applied modifications to achieve a successful result. The kind of output the hacker obtains, either a simple Yes/No or a binary map containing authentic and tampered blocks, plays a key role in determining the potentialities of the attack.
- *Cover-Image Attack* The counterfeiter has multiple pairs of original and authenticated images; this can happen when one has access to the image before authentication or when an estimate of the original can be performed. Again the hacker aims at making changes in such a way that the detector cannot reveal them or at gaining knowledge about the secret keys of the scheme.
- *Chosen Cover-Image Attack* The counterfeiter has the authentication device at his disposal and can submit his images to the authentication process; this could lead him to violate the secrets of the system.

Since the technique presented in this paper is based on the work by Fridrich [2], it inherits all the main security features of that algorithm. In particular, due to the specific structure of the logo, robustness to all the previous secu-

rity attacks, included the *Holliman and Memon* one, is granted (see [2] for a discussion about the security of Fridrich's scheme).

5 Experimental Results

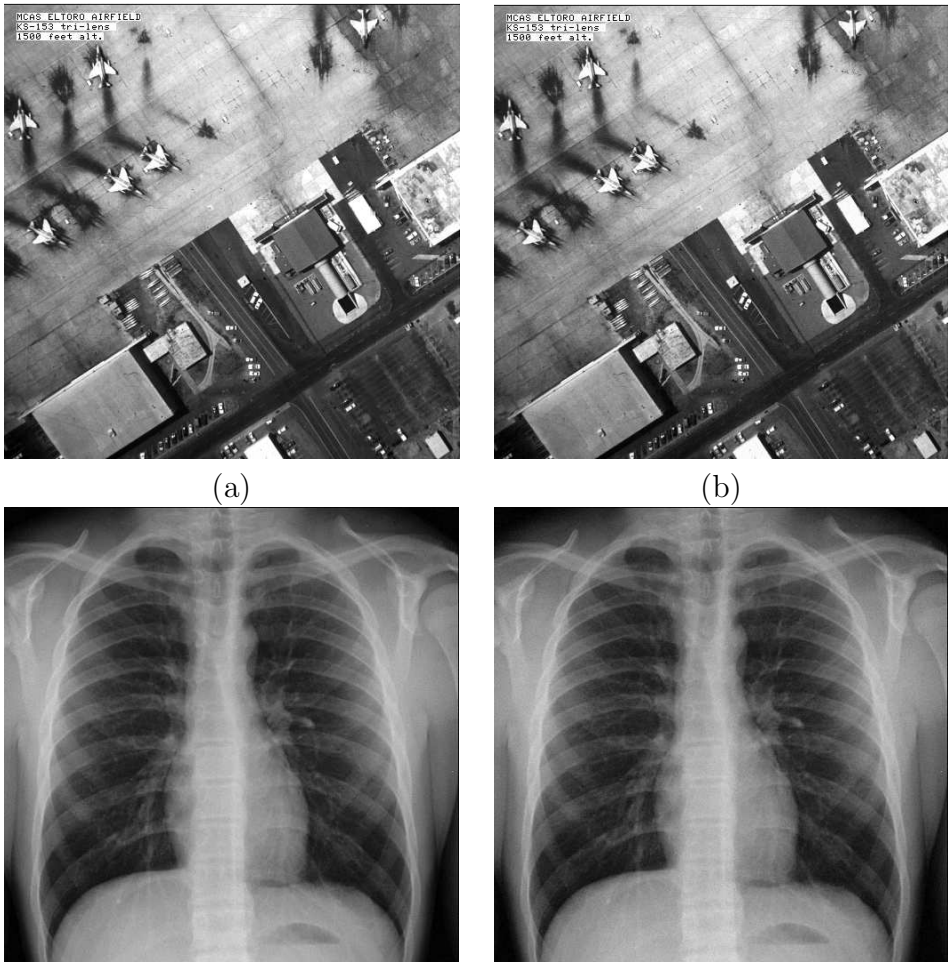
In this section some experimental results are given so to evaluate the performance of the authentication algorithm.

5.1 Watermark distortion

In this sub-Section, image distortion due to the watermark insertion is considered. Images belonging to the remote sensing and the biomedical scenarios are considered.

First of all, in Figure 4 an example of original and authenticated images ($NEAR = 2$) is given, both for the case of remote sensing (*El Toro Airfield* 512×512) and for the case of medical imaging (*RX-Chest* 512×512). In both circumstances authentication does not determine perceptual artifacts. To carry out a more objective analysis, the Peak-Signal-to-Noise-Ratio (PSNR) between the original image and the compressed one with different values of the $NEAR$ factor have been computed both in the case of near-lossless JPEG coding and in the case of joint authentication and coding. These results are presented in the graphs of Figure 5(a) for *El Toro Airfield* image and in Figure 5(b) for *RX-Chest*. It can be noticed that, as expected, there is a decrement (approximately $6 - 7dB$ for each level of $NEAR$ factor) in the value of PSNR when the authentication information is embedded within the image. This worsening is about the same for both the types of image and is almost constant when the $NEAR$ factor increases.

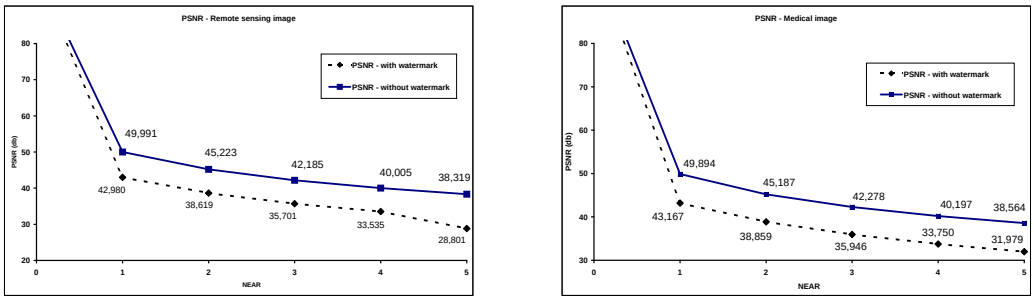
Being our primary aim that of designing a near-lossless scheme, where the maximum error can be strictly controlled, it is important to examine how the peak error varies as a consequence of watermark insertion. In fact each reconstructed pixel can differ from the corresponding original pixel by an amount bigger than $NEAR$, the maximum preset error. This effect is due to the fact that during the watermark embedding phase, the quantized errors are modified in order to accomplish image authentication. In particular, each quantized error is changed to obtain a reconstructed sample whose LSB is equal to that of the corresponding binary stripe. As a result of this process, the quantized prediction error is varied by one quantization step whose value is $(2 \times NEAR + 1)$. Because two possible quantization levels exist, the one which determines the minimum distance between the reconstructed sample



(a) (b)

(c) (d)

Fig. 4. *El Toro Airfield: (a) Original image and (b) authenticated image (NEAR = 2). RX-Chest: (c) Original image and (d) authenticated image (NEAR = 2).*



(a) (b)

Fig. 5. *(a) El Toro Airfield. Graph of PSNR versus preset error NEAR: continuous line JPEG-LS and dotted line JPEG-LS+WAT. (b) RX-Chest. Graph of PSNR versus preset error NEAR: continuous line JPEG-LS and dotted line JPEG-LS+WAT.*

and the original pixel Ix is chosen. This means that the two modifications (the one due to coding and the one due to watermarking) do not add each other, in such a way that the error is at most $2 \times NEAR + 1$. Anyway this choice is not possible in the case of pixel values that are near to 0 and 255 due to overflow and underflow problems. In this case, the choice to augment or decrease the quantized prediction error is obliged and the error could be equal to $NEAR + (2 \times NEAR + 1)$: the amount equal to $NEAR$ is due to JPEG-LS and the amount $(2 \times NEAR + 1)$ is due to watermark embedding. Strictly speaking, then the total maximum error ensured by the system that is equal to $(3 \times NEAR + 1)$.

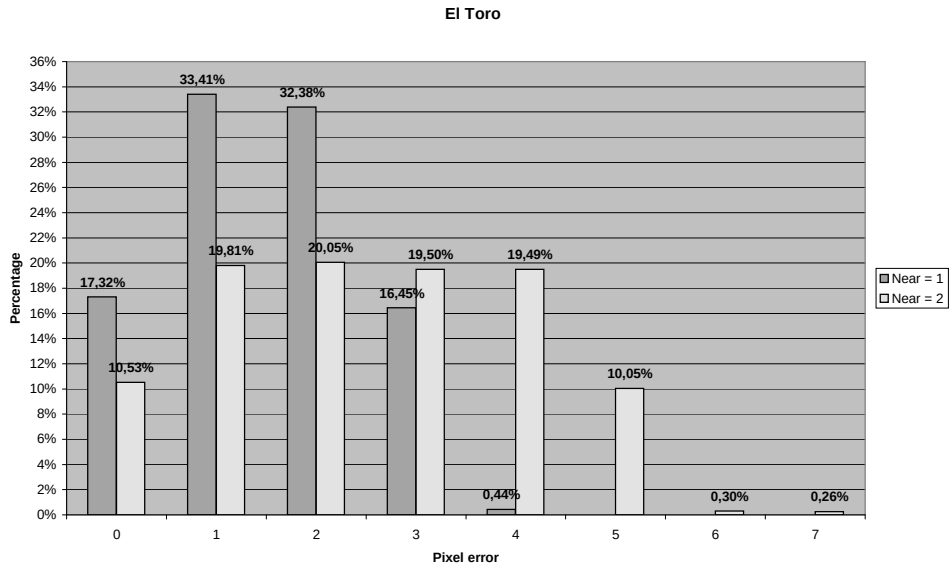


Fig. 6. *El Toro Airfield. Histogram of the percentage of image pixels having a certain distortion error ($NEAR = 1$ dark and $NEAR = 2$ bright). The maximum error between the original image and authenticated one is $(3 \times NEAR + 1)$, that is 3 and 7 respectively.*

In Figure 6 and Figure 7, the percentages of image pixels having a certain distortion error with respect to the original image for two sample images when $NEAR$ has been set to 1 and 2 are reported. It can be noticed that in all the cases about 50% of the image pixels have a distortion within the preset error value $NEAR$ and almost 80% of the image pixels is at most one gray level beyond $NEAR$.

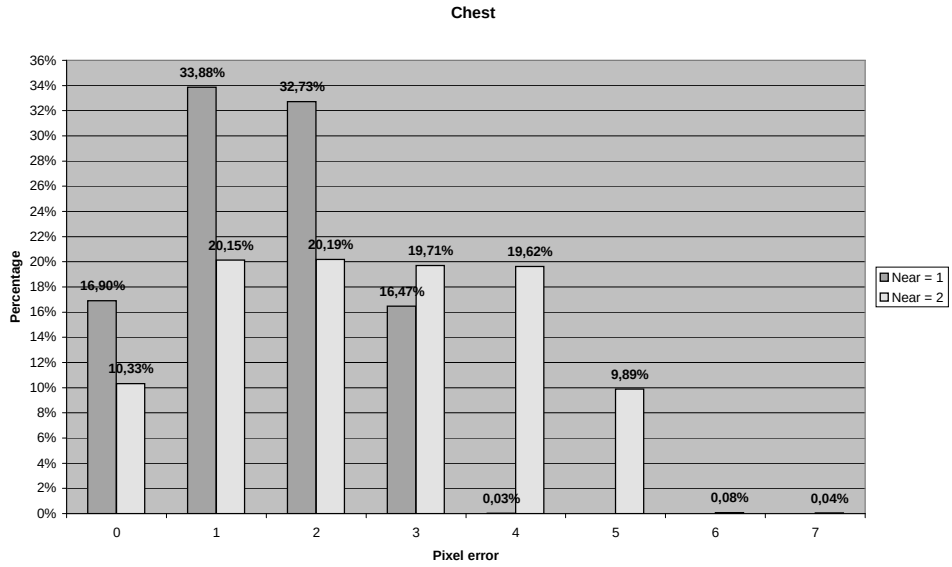


Fig. 7. *RX-Chest. Histogram of the percentage of image pixels having a certain distortion error ($NEAR = 1$ dark and $NEAR = 2$ bright). The maximum error between the original image and authenticated one is* $(3 \times NEAR + 1)$*, that is* 3 *and* 7 *respectively.*

5.2 Performance against attacks to authenticity

To examine the ability of the algorithm to ascertain image authenticity and to detect local modifications, near-lossless compressed and authenticated images have been tampered with and then authenticated.

In Figure 8, three examples of counterfeited images are illustrated: images in the left column have been modified by inserting some artifacts, in particular, in Figure 8(a) an airplane originally belonging to the image has been duplicated on the airfield, while in Figure 8(c) another airplane, a *B-52* taken from a different picture, has been added and finally in Figure 8(e) a "false fracture" has been artificially induced on the right collarbone of the chest. In the corresponding right columns these alterations have been rightly detected by the proposed technique, the image blocks that the detectors estimates to be altered are in black: the results demonstrate that the image authenticity is correctly verified, but the tamper localization resolution is decreased with respect to Fridrich's original work. In fact, because the embedding procedure inserts into an image block b_i the binary map found utilizing the pixels of its upper block, it is impossible to distinguish if block modification has been applied to block b_i or to its upper neighbor.

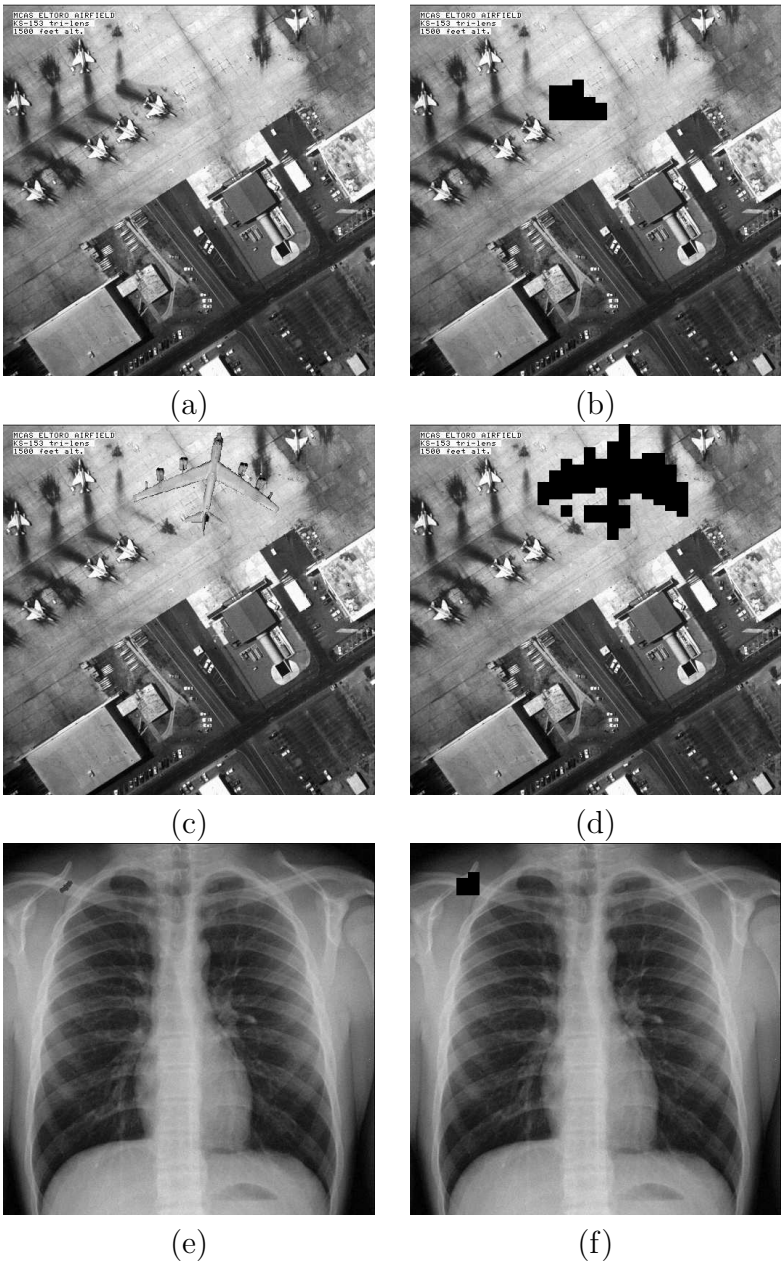


Fig. 8. *El Toro Airfield: Authenticated image after manipulation and detection of tampered zones (dark blocks) in the authenticated image respectively: object replication (a) and (b), object insertion (c) and (d). RX-Chest: (e) Authenticated image after manipulation. (f) Detection of tampered zones (dark blocks) in the authenticated image.*

5.3 Compression performance

Some tests, whose results are summarized in Table 1 for remote sensing and in Table 2 for medical images, have been carried out to establish the variation of compression rate between the JPEG-LS standard and the new integrated

system. Through these tests, it is possible to conclude that the authentication procedure leads to a decrement of the compression efficiency compared to that achieved by the plain JPEG-LS algorithm. This result is mainly due to the fact that in the watermarking embedding procedure the difference between smooth and no-smooth regions can not be exploited as usually done by switching between *run mode* and *regular mode* in JPEG-LS coding. As a proof of this thesis, it has been noted that the compression rate decrement for highly textured images is less than that experienced in flat images, where the *run mode* allows to improve the compression performance.

El Toro Airfield **pgm** 512×512 **size:** 262159 bytes		
	JPEG-LS + WAT	*JPEG-LS*
Near	Data Size (percentage)	Data Size (percentage)
0	62.92%	62.92%
1	44.87%	43.67%
2	37.63%	35.29%
3	33.38%	29.96%
4	30.61%	26.50%
5	28.84%	23.99%

Table 1
El Toro Airfield: output data size (percentage) with respect to the original size, obtained by JPEG-LS+WAT and JPEG-LS.

RX-Chest **pgm** 512×512 **size:** 262159 bytes		
	JPEG-LS + WAT	*JPEG-LS*
Near	Data Size (bytes)	Data Size (bytes)
0	42.19%	42.19%
1	29.73%	24.91%
2	25.56%	19.81%
3	24.10%	17.41%
4	23.50%	15.68%
5	23.22%	14.19%

Table 2
RX-Chest: output data size (percentage) with respect to the original size, obtained by JPEG-LS+WAT and JPEG-LS.

6 Conclusions

The system we presented in this paper permits to jointly compress and watermark a still image to allow its subsequent and tamper localization. The system was designed so to take into account the peculiarities of application scenarios requiring that the degradation of the original image content is strictly controlled (near-lossless compression and watermarking). Particular care was paid to ensure the security of the system. While the proposed system was expressly designed and tested to work on remote sensing and telemedicine imagery, its use is not limited to these scenarios. On the contrary, thanks to the compliance with the JPEG-LS coding standard and to the possibility of retrieving the watermark even in the raw pixel domain, we believe that our system can find application in a wide variety of real scenarios.

References

[1] M. M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification", in *Proc. ICIP97, IEEE Int. Conf. on Image Processing*, Santa Barbara, CA, Oct. 1997, vol. 2, pp. 680-683.

[2] J. Fridrich, "Security of fragile authentication watermarks with localization" in *Security and Watermarking of Multimedia Contents* Proc. SPIE vol. 4675, San Jose, CA, January 2002, pp. 691-700.

[3] M. Barni, F. Bartolini, V. Cappellini, E. Magli and G. Olmo, "Near-lossless digital watermarking for copyright protection of remote sensing images" in *IEEE International Geoscience and Remote Sensing Symposium*, 2002. IGARSS '02. 24-28 June 2002. Volume 3, pp.1447 - 1449.

[4] A. Kaarna, P. Zemcik, H. Kalviainen, J. Parkkinen, "Compression of multispectral remote sensing images using clustering and spectral reduction" in *IEEE Transactions on Geoscience and Remote Sensing*, Volume 38, Issue 2, March 2000, pp. 1073-1082.

[5] M. Barni and F. Bartolini and V. Cappellini and E. Magli and G. Olmo, "Watermarking-based protection of remote sensing images: requirements and possible solutions" in *Proc. of 2001 SPIE Annual Meeting.*

[6] M. Holliman and N. Memon, "Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes", in *IEEE Trans. on Image Processing*, vol. 9, Issue 3, March 2000, pp. 432-441.

[7] J. Fridrich, M. Goljan and N. Memon, "Further attacks on Yeung-Mintzer fragile watermarking scheme", in *Security and Watermarking of Multimedia Contents* Proc. SPIE, San Jose, CA, January 24-26, 2000, pp. 428-437.

[8] J. Fridrich, "Image watermarking for tamper detection", in *Proc. ICIP98, IEEE Int. Conf. on Image Processing*, vol. II, Chicago, IL, Oct. 1998, pp. 404-408.

[9] "ISO/IEC FCD 14495-1, Information technology - Lossless and near-lossless compression of continuous-tone still images - Part 1: Baseline [JTC 1/SC 29/WG 1 N 575]", in *ISO/IEC JTC 1/SC 29*, July 1997.

[10] M. Weinberger, G. Seroussi, G. Sapiro, "The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS", in *IEEE Trans. on Image Processing*, vol. 9, n. 8, August 2000, pp. 1309-1324.

[11] A. Piva, R. Caldelli, F. Bartolini and M. Barni, "Semi-fragile watermarking for still images authentication and content recovery", in CD-ROM Proc. of WIAMIS 2004, 5-th International Workshop on Image Analysis for Multimedia Interactive Services, 21-23 April 2004, Lisboa, Portugal.

[12] A. M. Tekalp, "Digital Video Processing", *Prentice Hall*, August 1995.

[13] B. Aiazzi, L. Alparone and S. Baronti, "Context Modeling for Near-Lossless Image Coding", in *IEEE Signal Processing Letters* vol. 9, n. 3, March 2002, pp. 473-483.

[14] A.G. Tescher, J.T. Reagan and J.A. Saghri, "Near lossless transform coding of multispectral images", in *International Geoscience and Remote Sensing Symposium, 1996. IGARSS '96*, vol. 2, 27-31 May 1996, pp. 1020-1022.

[15] A. Krivoulets, "Progressive near-lossless coding of medical images"; *Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis, 2003* ISPA 2003. 18-20 Sept. 2003, Volume 1, pp. 202-207.

[16] K. Chen and T.V. Ramabadran, "Near-lossless compression of medical images through entropy-coded DPCM"; *IEEE Transactions on Medical Imaging*. Sept. 1994, Volume 13, Issue 3, pp. 538-548.