FRAGILE WATERMARKING WITH IMPROVED TAMPERING LOCALISATION AND SELF-RECOVERY CAPABILITIES

Sergio Bravo-Solorio and Asoke K. Nandi

Department of Electrical Eng. & Electronics. The University of Liverpool
Brownlow Hill, Liverpool, L69 3GZ, UK; {sbravo,aknandi}@liverpool.ac.uk

ABSTRACT

This paper presents a new image fragile watermarking method aimed at affording higher tampering localisation accuracy and self-recovery capabilities. First, the distorted regions are roughly localised by means of a secure block-wise mechanism resilient to cropping. An additional pixel-wise mechanism is proposed to refine the tampering localisation map and estimate the value of the distorted pixels prior to the manipulation. We present experimental comparisons between the proposed method and a scheme reported in recent literature. Results show that our method compares favourably to the existing scheme in terms of tampering localisation and self-recovery performance. Furthermore, we show that the proposed scheme is capable of partially recovering pixels in missing regions, when the cover image has been cropped.

1. INTRODUCTION

The availability of sophisticated image processing tools in existing image editing software facilitates the creation of forgeries that cannot be easily identified, even by trained observers. Fragile watermarking has been proposed to address this concern and provide reliable mechanisms towards the authentication and integrity verification of digital images.

A highly desirable feature in fragile watermarking, usually referred to as *tampering localisation*, consists in identifying the manipulated regions, whilst the remainder of the image is properly verified [1]. Thus, valuable information could still be derived from genuine regions, even when other parts of the image have been reckoned to be fake. In Wong's scheme [8], for instance, an 8-bit grey-scale image is firstly split into non-overlapping blocks of pixels. Then, a message authentication code (MAC) is independently computed over the 7 most significant bit-planes (MSBPs) in each pixel-block, and then embedded over its least significant bit-plane (LSBP). This method is capable of readily localising altered pixel-blocks, but it has been proved to be vulnerable to vector quantisation (VQ) attacks, whereby block-wise independence is exploited to generate counterfeits that would go unnoticed [4].

Different mechanisms have been proposed to establish block-wise dependence in order to thwart VQ attacks effectively [7, 2]. However, these methods are intended to localise tampered blocks, but cannot differentiate between genuine and altered pixels within each block. To address this concern, some statistical elements have been recently studied in the generation of the authentication data in order to discriminate altered from genuine pixels [9, 3].

Some fragile watermarking methods have proposed to reconstruct part of the original content in manipulated regions [6, 5]; this capability is commonly referred to as *self-recovery*. In these schemes, a coarse approximation of the relevant perceptual information of the host image, encoded as a watermark, can be retrieved, at some point, to restore altered regions. A novel approach though, has been recently proposed by Zhang and Wang [10], whereby a pixel-wise and a block-wise mechanism are hierarchically structured to calculate not an estimate, but the original watermarked pixels by means of exhaustive attempts. First, each pixel is associated with a unique binary matrix, which is then multiplied by the five most significant bits (MSBs) of the pixel. The resulting bit-streams are assembled together, and divided in subsets. The bits in each subset are summed, using modulo-2 arithmetic, and concatenated in a single bit-stream, which is then embedded in the 160 bit positions selected in the 3 LSBPs of non-overlapping blocks of 8×8 pixels. Finally, a MAC, independently computed for every pixel-block, is embedded in the 32 remaining bit-positions in the LSBP of the block. At the receiver side, the MAC is employed to localise possible corrupted blocks, while the rest of the payload is used to enhance the tampering localisation and calculate the original MSBs of the distorted pixels, by means of exhaustive attempts. Unfortunately, not a single pixel can be recovered when the cover image has been cropped. In fact, this is a common limitation in fragile watermarking schemes with self-recovery capabilities.

Enlightened by Zhang and Wang's idea of combining block-wise and pixel-wise mechanisms for tampering localisation and self-recovery, we propose a new scheme with improved capabilities. In particular, we show that the tampering localisation/recovery performance of the proposed method compares favourably with Zhang and Wang's scheme. Furthermore, when the cover image has been cropped, we show that our method manages to recover the original MSBs of pixels in the missing portion. In Section 2, the proposed scheme is detailed, and some results are presented in Section 3. Finally, the paper is concluded in Section 4.

2. PROPOSED METHOD

A block-diagram that illustrates the general framework of the proposed method is shown in Fig. 1. Both phases will be detailed throughout this section.

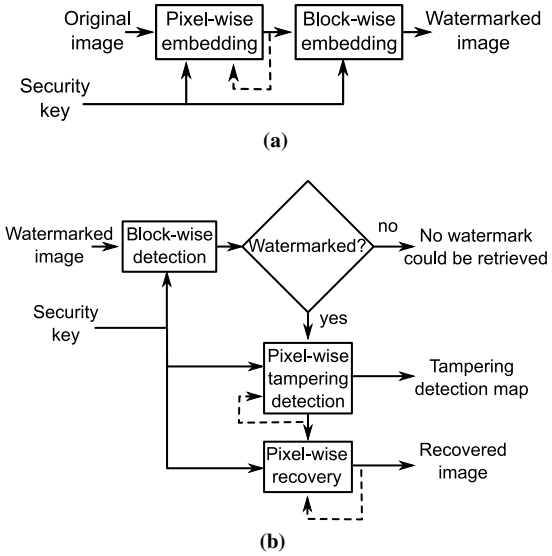


Figure 1: General framework of the proposed method. (a) embedding phase. (b) image verification and recovery phase.

Sergio Bravo-Solorio is supported by the National Council of Science and Technology (CONACyT) of Mexico.

 ISSN 2076-1465

Table 1: Pixel-wise embedding algorithm.

Require: $\hat{X}_q$, q, c, u, k.
Ensure: a watermarked subset $\hat{X}_q$.
1: Compute, $z_q = \mathcal{H}(k,c,q,\hat{X}_q)$, where $\mathcal{H}(\cdot)$ is a hash function.
2: Let $\delta_{q,1},\ldots,\delta_{q,m_{\text{pw}}}$ be a sequence of pseudo-random integers; the seed $(k+q)$ is employed to initialise the pseudo-random number generator.
3: **for** $p = 1$ to m_{pw} **do**
4: Compute, $l_{q,p} = z_q + \delta_{q,p}\hat{X}_{q,p}$.
5: Embed the watermark as,

$$\hat{X}_{q,p} = \hat{X}_{q,p} + parity(l_{q,p})\,2^{(c-1)} \quad , \tag{1}$$

where $parity(\cdot)$ is 1 if the input bit-stream has an odd number of ones, and 0 otherwise.
6: **end for**

Table 2: Block-wise embedding algorithm.

Require: $\bar{X}_r$, k, $\mathcal{I}_X$, n_X, m_X, r.
Ensure: a watermarked pixel-block X_r^{w}.
1: Encode an authentication structure, $s_r = \mathcal{I}_X \,||\, n_X \,||\, m_X \,||\, r$, where $\mathcal{I}_X$ is an image index exclusively associated to X, and $||$ denotes concatenation of bits.
2: Compute,

$$d_r = \mathcal{H}(k,\bar{X}_r) \oplus s_r \quad . \tag{2}$$

3: Spread the bit-stream d_r over the LSBP of $\bar{X}_r$ to create a watermarked block X_r^{w}.

2.1 Embedding phase

This phase is comprised of the two main stages in Fig. 1(a). Consider an 8-bit grey-scale image X, of size $n_1 \times n_2$, and let n_X denote the total number of pixels (i.e. $n_X = n_1 n_2$). Let u be the number of LSBPs that will be modified by the embedding process (in practice, we set u to 2 or 3). The remaining v MSBPs contain the relevant perceptual information, which will remain intact during the embedding process. For convenience, we will refer to the u LSBPs as to *authentication bit-planes*, and the v MSBPs as to *protected bit-planes*.

2.1.1 Pixel-wise embedding method

The authentication bit-planes are firstly removed from the input image by $\hat{X} = 2^u \lfloor 2^{-u} X \rfloor$, where $\lfloor\cdot\rfloor$ is the floor function. The procedure below will be iteratively repeated $(u-1)$ times; let $c = 2$ be the bit-plane to be watermarked in the first iteration.

A pseudo-random seed $(k+c)$, where k is a private key, is used to shuffle the pixels in $\hat{X}$. Divide $\hat{X}$ into disjoint subsets of m_{pw} pixels; let $\hat{X}_q$ denote the q-th subset in the image, for $q = 1,\ldots,(n_X/m_{\text{pw}})$, and $\hat{X}_{q,p}$ denote the p-th pixel in $\hat{X}_q$, for $p = 1,\ldots,m_{\text{pw}}$. Having watermarked every single subset, using the algorithm in Table 1, return the pixels in $\hat{X}$ to their original location. Increase, by one, the value of c and repeat the procedure if $c < u$.

Observe that the bit-stream $l_{q,p}$, defined in Step 4 (Table 1), is comprised of two terms, namely z_q and $\delta_{q,p}\hat{X}_{q,p}$. Besides, z_q will remain the same as long as the protected bit-planes are kept intact. These observations are crucial to understand the core functionality of the system at the receiver side.

2.1.2 Block-wise embedding method

To afford resilience to cropping and vector quantisation (VQ) attacks [4], we propose a tailored version of the block-wise method in [2] with improved tampering localisation capabilities. This is achieved by means of a new detection scheme, presented in Section 2.2.1, which requires no duplicated data in the structure encoded for every pixel-block.

Let $\bar{X}$ denote the watermarked image resulting from the pixel-wise embedding process; note that, at this point, all the bits in the LSBP of $\bar{X}$ are zero. Divide $\bar{X}$ into non-overlapping blocks of $m_1 \times m_2$ pixels. Let $\bar{X}_r$ be the r-th block in $\bar{X}$, for $r = 1,\ldots,(n_X/m_{\text{bw}})$, where $m_{\text{bw}} = m_1 m_2$ is the total number of pixels in each block. Having submitted every pixel-block to the algorithm in Table 2, assemble all the watermarked blocks together to form the watermarked image X^{w}.

Observe that the authentication structures generated in Step 1 (Table 2), whose length in bits is assumed to be m_{bw}, are comprised of a common prefix (i.e. $\mathcal{I}_X \,||\, n_X \,||\, m_X$), concatenated with a unique block-index. This is exploited by the proposed authenticator, elucidated in the forthcoming section.

2.2 Image verification and recovery phase

As illustrated in Fig. 1(b), this phase is comprised of the three main stages detailed below.

2.2.1 Block-wise detection method

Let Y be the input image of size $n_1' \times n_2'$. First, divide Y into non-overlapping blocks of $m_1 \times m_2$ pixels. Retrieve the bit-stream d_r' from the LSBP in each pixel-block Y_r, and compute,

$$s_r' = \mathcal{H}(k',\bar{Y}_r) \oplus d_r' \quad , \tag{3}$$

where k' is the secret key, and $\bar{Y}_r = 2\lfloor 2^{-1} Y_r \rfloor$. If Y is a watermarked, possibly tampered, image, and $k' = k$, it is expected that most of the extracted authentication structures will contain the same prefix (recall Section 2.1.2). If the number of structures containing an identical prefix is equal or greater than a predefined threshold τ_1, the image is regarded as authentic. The shape of the original cover image, $n_1 \times n_2$, can then be retrieved from the prefix with higher occurrence. If the cover image has not been cropped (i.e. $n_1' = n_1$ and $n_2' = n_2$), the tampered blocks can be readily localised in an $n_1 \times n_2$ binary bitmap M; pixels in genuine blocks are set to zero, otherwise they are set to one. Then, proceed with the pixel-wise detection method in Section 2.2.2.

If the number of structures containing an identical prefix is less than τ_1, one can generate a set of $m_1 m_2$ different shifted versions of the input image. In a shifted version, all the pixels of the image are displaced t_1 rows and t_2 columns, where $-m_1 < t_1 \leq 0$ and $-m_2 < t_2 \leq 0$. Each version is analysed by the detection procedure, described above, in a force-brute fashion. This will allow us to detect cover images whose left and/or upper-most edges have been removed by cropping. If none of the shifted versions were deemed authentic, the image is regarded as fake and the detection process is terminated altogether.

In case of cropping (i.e. $n_1' \neq n_1$ or $n_2' \neq n_2$), the block-index retrieved from every genuine block is used to calculate the displacement with respect to its location in the original non-cropped cover image. Assuming the displacement with higher occurrence, we can localise the altered blocks in an $n_1 \times n_2$ bitmap M (including blocks with atypical displacements). Additionally, Y must be reshaped to fit its original size $n_1 \times n_2$ (missing pixel-blocks can be filled with zeros). These steps are necessary to keep the synchronisation between the pixel-wise method and the watermark.

2.2.2 Pixel-wise detection method

This method is aimed at identifying authentic pixels that were mistakenly regarded as fake by the block-wise detector scheme, as they belonged to blocks containing one or more tampered pixels. As in the embedding method, the following procedure will be iteratively applied for $c = 2,\ldots,u$, where c denotes the bit-plane to be analysed in each iteration.

In every iteration, Y and M are shuffled by employing the key $(k'+c)$, and then split into disjoint subsets of m_{pw} pixels. Let $\hat{W}_q$

denote the c-th bit-plane in Y_q. For every subset Y_q, compute, $z'_q = \mathcal{H}(k', c, q, \hat{Y}_q)$, where $\hat{Y}_q = 2^u \lfloor 2^{-u} Y_q \rfloor$. Recalling the embedding method in Section 2.1.1, note that it is expected that $z'_q = z_q$, only if $\hat{Y}_q = \hat{X}_q$ and $k' = k$. Next, generate a pseudo-random sequence of integers $\delta'_{q,1}, \ldots, \delta'_{q,m_{\text{pw}}}$, by using the seed $(k'+q)$, and calculate,

$$l'_{q,p} = z'_q + \delta'_{q,p} \hat{Y}_{q,p} \ , \tag{4}$$

for every pixel in the subset.

Three different scenarios are being considered to determine whether an analysed subset will be deemed authentic.

a) The computed bits are identical to the ones retrieved from the authentication bit-planes, i.e., $parity(l'_{q,p}) = \hat{W}_{q,p}$, for every $p = 1, \ldots, m_{\text{pw}}$. Such a situation arises when all the pixels in Y_q are genuine, but some were mistakenly marked as altered by the block-wise detector. Hence, the protected bit-planes in Y_q are regarded as authentic, and M_q is set to zero.

b) Two conditions must be satisfied in this scenario. First, some of the bits computed from pixels regarded as fake in M_q do not match the retrieved authentication bits; i.e. for some $p \in \{1, \ldots, m_{\text{pw}}\}$, it follows that $M_{q,p} = 1$ and $parity(l'_{q,p}) \neq \hat{W}_{q,p}$. Second, all the bits calculated from pixels marked as genuine in M_q match the retrieved authentication bits; i.e. for all $p \in \{1, \ldots, m_{\text{pw}}\}$, such that $M_{q,p} = 0$, it follows that $parity(l'_{q,p}) = \hat{W}_{q,p}$. This case occurs when only the authentication bit-planes in Y_q have been altered, but the protected bit-planes remain unchanged. Therefore, all the pixels in the current subset are deemed authentic and M_q is set to zero. Since the probability of validating tampered pixels increases in this scenario, the method should skip the analysis of subsets that contain more than τ_2 pixels marked as fake in M_q, where τ_2 is a predefined threshold.

c) Some of the bits computed from pixels marked as genuine in M_q do not match the retrieved authentication bits, i.e. for some $p \in \{1, \ldots, m_{\text{pw}}\}$, it follows that $M_{q,p} = 0$ and $parity(l'_{q,p}) \neq \hat{W}_{q,p}$. This happens when the protected bit-planes in the current subset are corrupted. Hence, M_q is kept intact.

At the end of each iteration, the pixels in Y and M are returned back to their original location.

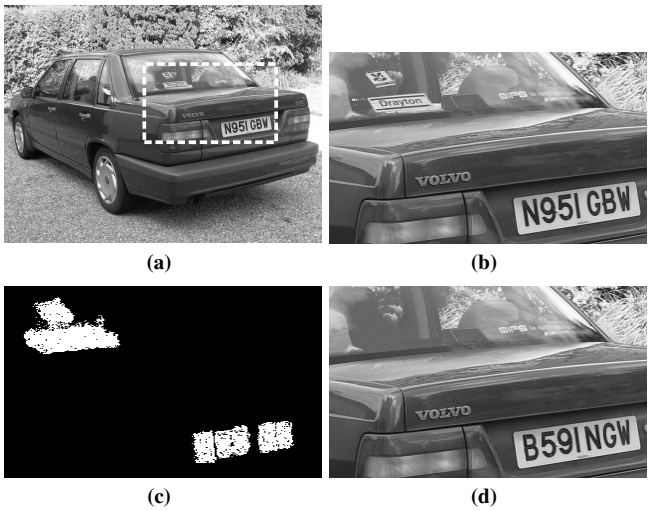
To determine the likelihood that a corrupted subset will be mistakenly validated in one of the iterations, let d_{q_1} denote the total number of altered pixels in Y_{q_1}, provided that $1 \leq d_{q_1} \leq \tau_2$. Besides, let E_{q_1} be the event that the authentication bits computed from the $(m_{\text{pw}} - d_{q_1})$ pixels marked as genuine, in M_{q_1}, match the retrieved bits. The probability that E_{q_1} will occur is $P_{E_{q_1}} = 2^{-(m_{\text{pw}} - d_{q_1})}$.

2.2.3 Pixel-wise recovery method

The aim of this method is to estimate the original v MSBs of tampered pixels. The rationale behind this method will be explained through the following example. Recalling the pixel-wise detector described above, let us assume that $Y_{q,a}$ is a pixel in Y_q whose v MSBs have been altered, for $a \in \{1, \ldots, m_{\text{pw}}\}$. Let d_{q_2} denote the number of pixels marked as fake in M_q, provided that $1 \leq d_{q_2} \leq \tau_2$ (we skip the analysis of the subsets where d_{q_2} is out of this range). In this scenario, it is expected that $parity(l'_{q,i}) \neq \hat{W}_{q,i}$, for some $i \in \{1, \ldots, m_{\text{pw}}\}$, such that $M_{q,i} = 0$ and $i \neq a$. Nonetheless, we can exhaustively try $Y_{q,a} = 0(2^u), 1(2^u), \ldots, 2^v(2^u)$ to find a *valid pixel value* that makes $parity(l'_{q,j}) = \hat{W}_{q,j}$, for every $j \in \{1, \ldots, m_{\text{pw}}\}$, such that $M_{q,j} = 0$. This procedure is repeated for each one of the d_{q_2} pixels marked as tampered in M_q. If only one valid pixel value is found, the subset is deemed genuine, the v MSBs of the pixel is replaced by the valid pixel value and M_q is set to zero. Otherwise, both the subset Y_q and M_q are kept intact.

Now, let us determine the likelihood that an erroneously estimated pixel value will cause the validation of a tampered subset in one of the iterations. Let E_{q_2} be the event that, for a single trial, the authentication bits computed from the $(m_{\text{pw}} - d_{q_2})$ genuine pixels match the retrieved bits. The probability of success of a single trial is $P_{E_{q_2}} = 2^{-(m_{\text{pw}} - d_{q_2})}$. Denoting, as $\mathcal{X}_{q_2}$, the total number of occurrences of E_{q_2}, we can express the probability that only one trial, out of the total $d_{q_2} 2^v$, will succeed as,

$$P_{\mathcal{X}_{q_2}=1} = \binom{d_{q_2} 2^v}{1} P_{E_{q_2}} (1 - P_{E_{q_2}})^{d_{q_2} 2^v - 1} \ . \tag{5}$$



(a) (b) (c) (d)

Figure 2: Conventional tampering. (a) Original image. (b) Fragment of the original image bounded by the dashed rectangle in (a). (c) Tampered pixels (fragment). (d) Tampered cover image (fragment).

3. RESULTS

We present some experiments to test the tampering localisation and self-recovery performance of the proposed scheme. We tested our system for watermarks embedded in 2 and 3 LSBPs, i.e. $u = 2$ and $u = 3$. In both cases, the block-size used in the block-wise method was $8(m_1) \times 8(m_2)$, while the subsets in the pixel-wise method were formed by $m_{\text{pw}} = 16$ pixels. For comparison purposes, the same experiments were carried out in images watermarked with Zhang and Wang's method [10], whereby the watermark is always embedded in the 3 LSBPs. Our interest in comparing our scheme with this method arises from the fact that, Zhang and Wang's method is designed to restore the original value of the watermarked pixels, instead of retrieving coarse approximations as in [6, 5].

The following measures are used to compare the localisation accuracy. Let E be the bitmap encoded by our system, and G be the ground truth, which contains all the pixels whose v MSBs have been corrupted[1]. The accuracy (ACC) and the false positive rate (FPR) will be calculated as,

$$\text{ACC} = \frac{E \bigcap G}{G} \ , \text{and,} \quad \text{FPR} = \frac{E \bigcup G}{G} - 1 \ ,$$

where the ideal detection would be: ACC = 1 and FPR = 0.

3.1 Conventional tampering

The test image, of size 600×800, used in our experiments is shown in Fig. 2(a). We employed the peak signal-to-noise ratio (PSNR) to measure the embedding distortion. The distortion induced to the images watermarked with the proposed method was assessed to be 44.1 dB, for $u = 2$, and 37.9 dB, for $u = 3$. On the other hand, the distortion induced by Zhang and Wang's method was assessed to be

[1] The ground truth used when $u = 2$ is different to the one used when $u = 3$, as the number of protected bit-planes (i.e. v) is different.

(a) **(b)** ACC=1.0, FPR=0.11

(c) ACC=1.0, FPR=0.01 **(d)** ACC=1.0, FPR=0.13

Figure 3: Localisation results from the conventional tampering test. (a) Proposed detector using 2 LSBPs. (b) Region enclosed by the dashed rectangle in (a). (c) Proposed method using 3 LSBPs. (d) Zhang and Wang's method (uses 3 LSBPs).

(a) **(b)** Restored pixels: 89.6%

(c) Restored pixels: 99.4% **(d)** Restored pixels: 90.1%

Figure 4: Self-recovery results from the conventional tampering test. (a) Proposed detector using 2 LSBPs. (b) Region enclosed by the dashed rectangle in (a). (c) Proposed method using 3 LSBPs. (d) Zhang and Wang's method (uses 3 LSBPs).

38.1 dB. The watermarked images were used to generate identical counterfeits: the original vehicle registration plate, "N95I GBW", was changed to "B59I NGW" and the stickers on the rear window were concealed. To facilitate visual inspection, the region bounded by the dashed rectangle in Fig. 2(a) is shown in Fig. 2(b). The corresponding fragment of the counterfeit, which contains all the tampered pixels, is shown in 2(d); the altered pixels are illustrated in Fig. 2(c). The tampering localisation results are presented in Fig. 3, where pixels deemed tampered can be identified as black spots. Note that the results achieved with the proposed method, using only 2 LSBPs (i.e. $u = 2$), are comparable to those obtained with Zhang and Wang's method, which embeds the data in the 3 LSBPs. On the other hand, the false positive rate decreased significantly when using $u = 3$ in the proposed method. Self-recovery results are summarised in Fig. 4 (pixels whose value could not be estimated are depicted as black spots). Observe that the percentage of pixels restored by the proposed method, when using 2 LSBPs, is comparable

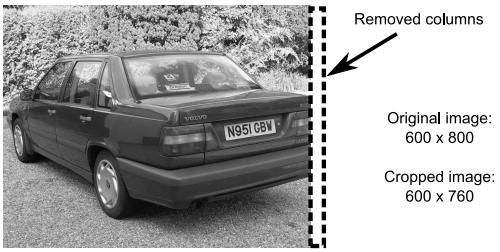


Figure 5: Tampering+cropping attack.

(a) **(b)** ACC=1.0, FPR=0.07

(c) ACC=1.0, FPR=0.05 **(d)** ACC=0.25, FPR=0.13

Figure 6: Localisation results from the tampering+cropping test. (a) Proposed detector using 2 LSBPs. (b) Region enclosed by the dashed rectangle in (a). (c) Proposed method using 3 LSBPs. (d) Zhang and Wang's method (uses 3 LSBPs).

to the percentage of pixels restored by Zhang and Wang's method, which uses 3 LSBPs. Even better is the fact that, embedding the watermark in the 3 LSBPs, as in Zhang and Wang's method, our method managed to restore over 99% of the altered pixels.

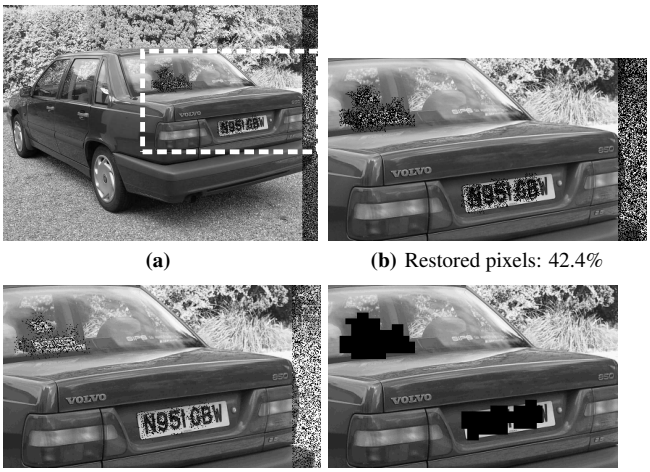
3.2 Tampering detection in presence of cropping

As illustrated in Fig. 5, the 40 right-most columns were removed from the counterfeits generated in the preceding experiment, resulting in a 600×760 forgery. Figure 6 summarises the tampering localisation results. Since our method managed to detect the cropping, the image could be reshaped to fit its original size (recall Section 2.2.1). Hence, the manipulation could be effectively localised. In contrast, Zhang and Wang's method failed to detect the cropping, affecting thus the tampering localisation performance. Figure 7 summarises the self-recovery results. Observe that the proposed scheme managed to restore over 40% of the tampered pixels when using 2 LSBPs, and over 70% when using 3 LSBPs. In fact, the original registration plate can be readily discernible in the later case. On the other hand, as a result of cropping, Zhang and Wang's recovery method lost synchronisation with the watermark. Therefore, it failed to restore any pixel.

3.3 Self-recovery: performance comparison

To compare the self-recovery performance of the two schemes, we used ten 512×512 images in the "miscellaneous" collection of the USC-SIPI image database[2]; the test images are named: 7.1.01 to 7.1.10. Each cover image was used to generate a set of 10 different forgeries. Every single pixel within a centred disk was deliberately distorted. The diameter of the disk was adjusted to include a

[2]Available at: http://sipi.usc.edu/services/database/ .



(a) **(b)** Restored pixels: 42.4%

(c) Restored pixels: 71.5% **(d)** Restored pixels: 0%

Figure 7: Self-recovery results from the tampering+cropping test. (a) Proposed detector using 2 LSBPs. (b) Region enclosed by the dashed rectangle in (a). (c) Proposed method using 3 LSBPs. (d) Zhang and Wang's method (uses 3 LSBPs).

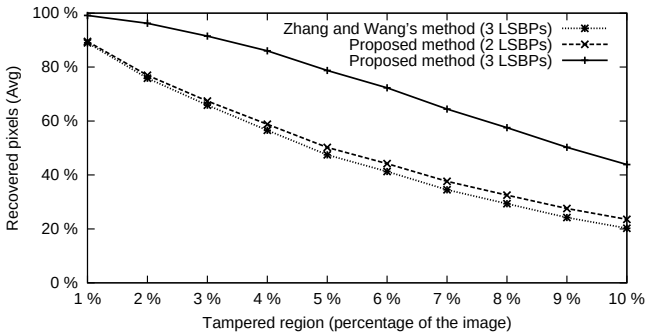


Figure 8: Image restoration comparison results.

percentage of the total pixels in the image; the tested percentages varied from 1% to 10%. Figure 8 shows the average percentage of recovered pixels in each test. The performance of the proposed method using 2 LSBPs is comparable to the one shown by Zhang and Wang's scheme in all the tests. However, when our method used 3 LSBPs, the same as in Zhang and Wang's scheme, the percentage of recovered pixels increased over 20% in most of the tests. In fact, using this set-up, the proposed scheme managed to restore over half of the altered pixels when the tampered region extended up to 8% of the image.

4. CONCLUSIONS AND FURTHER WORK

We have proposed a new fragile watermarking method that combines both block-wise and pixel-wise mechanisms to afford higher tampering localisation accuracy and self-recovery capabilities. The presented method is aimed at recovering content from images that have been either tampered with or cropped, as long as the distorted/missing regions are not a large fraction of the total image. Thus, our method is not intended to recover images affected by global distortions, such as JPEG compression or filtering. We presented experiments to demonstrate the effectiveness of the proposed method to recover content from images that have undergone conventional distortions and cropping. Comparison results show that our scheme outperforms an existing method, reported in recent literature, in terms of tampering localisation and self-recovery performance. Future research involves combining standard image restoration techniques with the proposed scheme to constrain the exhaustive search to values predicted from genuine pixels in a neighbourhood, and thus improve the self-recovery performance.

REFERENCES

[1] I. J. Cox, M. L. Miller, J. A. Bloom, J. Fridrich, and T. Kalker. *Digital Watermarking and Steganography*. Morgan Kauffman, 2nd edition, 2008.

[2] J. Fridrich. Security of fragile authentication watermarks with localization. In *Proc. of SPIE - The International Society for Optical Engineering*, volume 4675, pages 691 – 700, CA, USA, 2002.

[3] H. J. He, J. S. Zhang, and F. Chen. Adjacent-block based statistical detection method for self-embedding watermarking techniques. *Signal Processing*, 89(8):1557–1566, Aug. 2009.

[4] M. Holliman and N. Memon. Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes. *IEEE Transactions on Image Processing*, 9(3):432 – 441, 2000.

[5] T. Y. Lee and S. D. Lin. Dual watermark for image tamper detection and recovery. *Pattern Recognition*, 41(11):3497–3506, Nov. 2008.

[6] P.-L. Lin, C.-K. Hsieh, and P.-W. Huang. A hierarchical digital watermarking method for image tamper detection and recovery. *Pattern Recognition*, 38(12):2519 – 2548, 2005.

[7] P. Wong and N. Memon. Secret and public key image watermarking schemes for image authentication and ownership verification. *IEEE Transactions on Image Processing*, 10(10):1593 – 1601, 2001.

[8] P. W. Wong. A public key watermark for image verification and authentication. In *Proc. of ICIP – International Conference on Image Processing*, volume 1, pages 455 – 9, Chicago, IL, USA, 1998.

[9] X. Zhang and S. Wang. Statistical fragile watermarking capable of locating individual tampered pixels. *IEEE Signal Processing Letters*, 14(10):727–730, 2007.

[10] X. Zhang and S. Wang. Fragile watermarking scheme using a hierarchical mechanism. *Signal Processing*, 89(4):675–679, Apr. 2009.