Spatial-Domain Image Watermarking Robust against Compression, Filtering, Cropping, and Scaling *

Francesc Sebé[1], Josep Domingo-Ferrer[1], and Jordi Herrera[2]

[1] Universitat Rovira i Virgili, Department of Computer Science and Mathematics, Autovia de Salou s/n, E-43006 Tarragona, Catalonia, Spain, e-mail {fsebe,jdomingo}@etse.urv.es
[2] Universitat Oberta de Catalunya, Av. Tibidabo 39-43, E-08035 Barcelona, Catalonia, Spain, e-mail jherreraj@campus.uoc.es

Abstract. Two robust spatial-domain watermarking algorithms for image copyright protection are described in this paper. The first one is robust against compression, filtering and cropping. Like all published crop-proof algorithms, the one proposed here requires the original image for mark recovery. Robustness against compression and filtering is obtained by using the JPEG algorithm to decide on mark location and magnitude; robustness against cropping is achieved through a repetition code. The second watermarking algorithm uses visual components and is robust against compression, filtering, scaling and moderate rotations.

1 Introduction

Electronic copyright protection schemes based on the principle of copy prevention have proven ineffective or insufficient in the last years (see [9],[10]). The recent failure of the DVD copy prevention system [13] is just another argument supporting the idea that electronic copyright protection should rather rely on copy detection techniques. Watermarking is a well-known technique for copy detection, whereby the merchant selling the piece of information (*e.g.* image) embeds a *mark* in the copy sold.

In [12], three measures are proposed to assess the performance of information hiding schemes, which are a general class including watermarking schemes:

Robustness: Resistance to accidental removal of the embedded bits.
Capacity: The amount of information that may be embedded and later on recovered.
Imperceptibility: Extent to which the embedding process leaves undamaged the perceptual quality of the covertext (the copyrighted information being traded with).

* This work is partly supported by the Spanish CICYT under grant no. TEL98-0699-C02-02.

J. Pieprzyk, E. Okamoto, and J. Seberry (Eds.): ISW 2000, LNCS 1975, pp. 44–53, 2000.

Commercial watermarking schemes surviving a broad range of manipulations (*e.g.* Digimarc [3]) tend to be based on proprietary algorithms not available in the literature. Published watermarking algorithms can be divided into oblivious and non-oblivious; the first type does not require the original image for mark reconstruction (for example, see [5,6]), while the second type does. To our best knowledge, no published oblivious scheme can survive arbitrary cropping attacks. On the other hand, a large number of proposals operate on transformed domains (DCT, wavelet) rather than on the spatial domain. Spatial domain watermarking is attractive because it provides a better intuition on how to attain an optimal tradeoff between robustness, capacity and imperceptibility. Thus, coming up with public spatial domain algorithms which survive a broad range of manipulations is an important issue.

1.1 Our Results

We present in this paper two watermarking schemes which are robust against a very broad range of transformations. Both schemes operate in the spatial domain and are designed for image copyright protection. Their features are as follows:

- The first algorithm is based on the ideas of [8], but offers greater simplicity (*e.g.* visual components are not used), capacity (more pixels can be used to convey mark bits) and robustness than [8]. Transformations survived include compression, filtering and cropping. The new scheme also improves on the earlier version [15] both in imperceptibility (higher signal-to-noise ratios) and robustness (cropping is now survived).
- The second algorithm uses visual components, image tiling and mark redundancy to survive compression, filtering, scaling and moderate rotations.

Section 2 describes the crop-proof watermarking algorithm. Section 3 describes the scale-proof watermarking algorithm. Robustness of both algorithms is discussed in Section 4. Section 5 lists some conclusions and topics for further research.

2 Crop-Proof Watermarking

The scheme can be described in two stages: mark embedding and mark reconstruction. Like all practical schemes known to date, the following one is symmetric in the sense of [11]: mark embedding and recovery are entirely performed by the merchant M.

For mark embedding, we assume that the image allows sub-perceptual perturbation. Assume that q is a JPEG quality level chosen in advance by the merchant M; q will be used as a robustness and capacity parameter. Also, let p be a Peak Signal-to-Noise Ratio (PSNR,[7]) chosen in advance by the merchant; p will be used as an imperceptibility parameter, *i.e.* M requires that the embedding process does not bring the PSNR below p dB. Let $\{s_i\}_{i\geq 1}$ be a random bit sequence generated by a cryptographically sound pseudo-random

generator with secret key k only known to M. An image X will be represented as $X = \{x_i : 1 \leq i \leq n\}$, where n is the number of pixels and x_i is the color level of the i-th pixel. For monochrome images, $n = w \times h$, where w and h are, respectively, the width and height of the image. For RGB color images, $n = 3(w \times h)$ (there are three matrices rather than one).

Algorithm 1 (Mark embedding(p,q))

1. *Compress X using the JPEG algorithm with quality q as input parameter. Call the bitmap of the resulting compressed image X'. Let $\delta_i := x_i - x'_i$ be the difference between corresponding pixels in X and X'. Only positions i for which $\delta_i \neq 0$ will be useable to embed bits of the mark.*
2. *Call ε the mark to be embedded. Encode ε using an error-correcting code (ECC) to obtain the encoded mark E; call $|E|$ the bit-length of E. Replicate the mark E to obtain a sequence E' with as many bits as pixels in X with $\delta_i \neq 0$.*
3. *Let $j := 0$. For $i = 1$ to n do:*
   a) *If $\delta_i = 0$ then $x''_i := x_i$.*
   b) *If $\delta_i \neq 0$ then*
      i. *Let $j := j + 1$. Compute $s'_j := e'_j \oplus s_j$, where e'_j is the j-th bit of E'. The actual bit that will be embedded is s'_j.*
      ii. *If $s'_j = 0$ then compute $x''_i := x_i - \delta_i$.*
      iii. *If $s'_j = 1$ then compute $x''_i := x_i + \delta_i$.*
4. *While $PSNR(X''|X) < p$ do*
   a) *Randomly pick an index i such that $1 \leq i \leq n$.*
   b) *If $x''_i - x_i > 3$ then $x''_i := x''_i - 1$.*
   c) *If $x''_i - x_i < -3$ then $x''_i := x''_i + 1$.*

$X'' = \{x''_i : 1 \leq i \leq n\}$ is the marked image, which yields at least $PSNR(X''|X) = p$ dB (PSNR of X'' with respect to X). The influence of q on capacity and robustness is discussed below. The use of the value 3 when adjusting the PSNR is empirically justified: this is the minimal magnitude that reasonably survives the attacks considered in the next section. In addition to PSNR, quality metrics such as the ones in [4,7] can be used to measure imperceptibility of the mark; if such complementary measures are not satisfactory, then re-run Algorithm 1 with a higher q.

For mark reconstruction, knowledge of X and the secret key k is assumed (k is used to regenerate the random sequence $\{s_i\}_{i \geq 1}$); knowledge of the original mark ε is not assumed, which allows to use the proposed algorithm for fingerprinting (some collusion-security can be achieved using dual binary Hamming codes as ECC, see [15]). Using X for mark reconstruction is a common feature in all published crop-proof watermarking algorithms; on the other hand, it is not a serious shortcoming in symmetric marking algorithms, where mark reconstruction is performed by and for the merchant M; moreover, for still images, one can consider that the secret key is (k, X).

Algorithm 2 (Mark reconstruction(q))

1. *Upon detecting a redistributed item* $\hat{X}$*, restore it to the bitmap format.*
2. *Compress the corresponding original* X *with quality* q *to obtain* X'*.*
3. *Let* $ones[\cdot]$ *and* $zeroes[\cdot]$ *be two vectors with* $|E|$ *integer positions initially all set to 0.*
4. *Let* $j := 0$*. For* $i = 1$ *to* n *do:*
   a) *Compute* $\delta_i := x_i - x'_i$*.*
   b) *If* $\delta_i \neq 0$ *then*
      i. *Let* $j := j + 1$*. If* $j > |E|$ *then* $j := 1$*.*
      ii. *Compute* $\hat{\delta}_i := \hat{x}_i - x_i$*.*
      iii. *If* $\hat{\delta}_i = 0$ *then* $\hat{s}_j := \#$*, where* $\#$ *denotes erasure.*
      iv. *If* $\delta_i \times \hat{\delta}_i > 0$ *then* $\hat{s}_j := 1$*.*
      v. *If* $\delta_i \times \hat{\delta}_i < 0$ *then* $\hat{s}_j := 0$*.*
      vi. *If* $\hat{s}_j \neq \#$ *then* $\hat{e'}_j := \hat{s}_j \oplus s_j$*; otherwise* $\hat{e'}_j := \#$*.*
      vii. *If* $\hat{e'}_j = 1$ *then* $ones[j] := ones[j] + 1$*.*
      viii. *If* $\hat{e'}_j = 0$ *then* $zeroes[j] := zeroes[j] + 1$*.*
5. *For* $j = 1$ *to* $|E|$ *do:*
   a) *If* $ones[j] > zeroes[j]$ *then* $\hat{e}_j := 1$*, where* $\hat{e}_j$ *is the* j*-th bit of the recovered mark* $\hat{E}$*.*
   b) *If* $ones[j] < zeroes[j]$ *then* $\hat{e}_j := 0$*.*
   c) *If* $ones[j] = zeroes[j]$ *then* $\hat{e}_j := \#$*.*
6. *Decode* $\hat{E}$ *with the same ECC used for embedding to obtain* $\hat{\varepsilon}$*.*

Note that the redistributed $\hat{X}$ may have width $\hat{w}$ and height $\hat{h}$ which differ from w and h due to manipulation by the re-distributor; this would cause the number of pixels $\hat{n}$ in $\hat{X}$ to be different from n. In Section 4, we discuss how to deal with attacks altering n.

3 Scale-Proof Watermarking

Like the previous one, this scheme operates in the spatial domain and is a symmetric one. Again, we assume the original image is represented as $X = \{x_i : 1 \leq i \leq w \times h\}$, where x_i is the color level of the i-th pixel and w and h are, respectively, the width and height of the image. For RGB color images, mark embedding and reconstruction is independently done for each color plane.

We next propose an algorithm for computing the visual components of the pixels in the image, that is their perceptual value. The idea underlying Algorithm 3 is that dark pixels and those pixels in non-homogeneous regions are the ones that can best accomodate embedded information while minimizing the perceptual impact.

Algorithm 3 (Visual components)

1. *For* $i = 1$ *to* n *do:*
   a) *Compute* $m_i := \max_j |x_i - x_j|/4$, *for all pixels* j *which are neighbors of pixel* i *on the image (there are up to eight neighbors);* m_i *can be regarded as a kind of discrete derivative at pixel* i. *To bound the value of* m_i *between 1 and 4, perform the following corrections:*
      i. *If* $m_i > 4$ *then* $m_i := 4$.
      ii. *If* $m_i = 0$ *then* $m_i := 1$.
   b) *Compute the darkness of the* i*-th pixel as* $d_i := (70 - x_i) * 4/70$ *if* $x_i < 70$ *and* $d_i := 0$ *otherwise. We consider a pixel to be dark if its color level is below 70. The value of* d_i *lies between 0 and 4.*
   c) *Compute the preliminary visual component of the* i*-th pixel as* $v_i := \max(m_i, d_i)$.
2. *For* $i = 1$ *to* n *compute the final visual component of the* i*-th pixel as* $V_i := \max_j v_j$, *for all pixels* j *which are neighbors of* i *on the image plus pixel* i *itself.*

Mark embedding is based on visual components and encrypts the mark bits using a random bit sequence $\{s_i\}_{i \geq 1}$ generated by a cryptographically sound pseudo-random generator with secret key k only known to M.

Algorithm 4 (Mark embedding(p,r))

1. *Divide the image into the maximum possible number of square tiles of* p *pixels side, so that there is a* r *pixels wide band between neighboring tiles (the band separates tiles). Let* q *be the number of resulting tiles. Each tile will be used to embed one bit, so* q *is the capacity of this watermarking scheme.*
2. *Call* ε *the mark to be embedded. Encode* ε *using an error-correcting code (ECC) to obtain the encoded mark* E. *If* $|E|$ *is the bit-length of* E, *we must have* $|E| \leq m$. *Replicate the mark* E *to obtain a sequence* E' *with* q *bits.*
3. *For* $i = 1$ *to* q *compute* $s'_i = e'_i \oplus s_i$, *where* e'_i *is the* i*-th bit of* E'.
4. *To embed the* i*-th encrypted mark bit* s'_i *into the* i*-th tile do:*
   a) *If* $s'_i = 0$ *then* $x'_j := x_j - V_j$ *for all pixels* x_j *in the* i*-th tile.*
   b) *If* $s'_i = 1$ *then* $x'_j := x_j + V_j$ *for all pixels* x_j *in the* i*-th tile.*

$X' = \{x'_i : 1 \leq i \leq w \times h\}$ is the marked image. The band between tiles is never modified and it helps to avoid perceptual artifacts that could appear as a result of using two adjacent tiles to embed a 0 and a 1. The use of a range 1 to 4 for visual components is empirically justified: an addition or subtraction of up to 4 to the color level can hardly be perceived but at the same time survives most subperceptual manipulations. Regarding parameters p and r, we recommend to use $p = 5$ and $r = 3$ as a tradeoff between capacity —which would favor tiles as small as possible and intertile bands as narrow as possible—, robustness — the larger a tile, the more redundancy in bit embedding and the more likely is

correct bit reconstruction— and imperceptibility —the wider a band, the less chances for artifacts.

The assumptions for mark reconstruction are identical to those made for the scheme of Section 2, namely knowledge of X and k (to regenerate the random sequence $\{s_i\}_{i\geq 1}$. No knowledge of the original mark ε is assumed, so that the proposed scheme is also useable for fingerprinting. Let $\hat{X}$ be the redistributed image, and let $\hat{w}$ and $\hat{h}$ be its width and height.

Algorithm 5 (Mark reconstruction(p,r))

1. *Let ones*$[\cdot]$ *and zeroes*$[\cdot]$ *be two vectors with* $|E|$ *integer positions initially all set to 0.*
2. *From the length p of the tile side, the width r of the intertile band and* $\hat{X}$*, compute the estimated number of tiles* $\hat{q}$*.*
3. *For* $t = 1$ *to* $\hat{q}$ *do:*
   a) *Let* $u := 1 + ((t-1) \bmod |E|)$
   b) *For each pixel in the t-th tile of the original image X do:*
      i. *Let i and j be the row and column of the considered original pixel, which will be denoted by* x_{ij}*.*
      ii. *Locate the pixel* $\hat{x}_{ab}$ *in the marked image* $\hat{X}$ *corresponding to* x_{ij}*. To do this, let* $a := i \times \hat{h}/h$ *and* $b := j \times \hat{w}/w$*.*
      iii. *Compute* $\hat{\delta}_{ij} := \hat{x}_{ab} - x_{ij}$*.*
      iv. *If* $\hat{\delta}_{ij} > 0$ *then* $ones[u] := ones[u] + 1$*.*
      v. *If* $\hat{\delta}_{ij} < 0$ *then* $zeroes[u] := zeroes[u] + 1$*.*
4. *For* $u = 1$ *to* $|E|$ *do:*
   a) *If* $ones_u > zeroes_u$ *then* $\hat{s}_u := 1$*, where* $\hat{s}_u$ *is the recovered version of the u-th embedded bit.*
   b) *If* $ones_u < zeroes_u$ *then* $\hat{s}_u := 0$*.*
   c) *If* $ones_u = zeroes_u$ *then* $\hat{s}_u := \#$*, where # denotes erasure.*
   d) *If* $\hat{s}_u \neq \#$ *then* $\hat{e}_u := \hat{s}_u \oplus s_u$ *otherwise* $\hat{e}_u := \#$*, where* $\hat{e}_u$ *is the u-th bit of the recovered mark* $\hat{E}$*.*
5. *Decode* $\hat{E}$ *with the same ECC used for embedding to obtain* $\hat{\varepsilon}$*.*

4 Robustness Assessment

The two schemes described above were implemented using a dual binary Hamming code $DH(31, 5)$ as ECC. The base test of the StirMark 3.1 benchmark [9] was used to evaluate their robustness. The following images from [14] were tried: Lena, Bear, Baboon and Peppers. A 70-bit long mark ε was used for both schemes, which resulted in an encoded E with $|E| = 434$.

4.1 Robustness of the Crop-Proof Scheme

Using the scheme of Section 2, the percent of the n pixels that can be used to convey a mark bit (those with $\delta_i \neq 0$) ranges between 78.9% for $q = 90\%$ and 90.7% for $q = 10\%$ (there are slight variations between images). Values $p = 38dB$, $q = 60\%$ were tried on the above images, and the following StirMark manipulations were survived by the embedded mark:

1. Color quantization.
2. All low pass filtering manipulations. More specifically:
   a) Gaussian filter (blur).
   b) Median filter (2×2, 3×3 and 4×4).
   c) Frequency mode Laplacian removal [1].
   d) Simple sharpening.
3. JPEG compression for qualities 90% down to 30% (to resist 20% and 10% a lower value for q would be necessary).
4. Rotations with and without scaling of -0.25 up to 0.25 degrees.
5. Shearing up to 1% in the X and Y directions.
6. All StirMark cropping attacks. These are resisted thanks to the mark being repeatedly embedded in the image. To resynchronize, keep shifting the cropped $\hat{X}$ over X until the the "right" relative position of $\hat{X}$ on X is found. The right position is estimated to be the one that, after running Algorithm 2 on $\hat{X}$ and the corresponding cropping of X, yields the minimal number of corrected errors at Step 6.
7. Removal of rows and columns from the marked image X'' was automatically detected, dealt with, and survived exactly like cropping attacks.

Additional rotation, scaling and shearing StirMark attacks can be detected and undone by M prior to mark reconstruction by using computer vision techniques to compare with the original image. The really dangerous attacks for the scheme presented here are random geometric distortions and attacks combining several of the aforementioned elementary manipulations. Figure 1 tries to show that marking is imperceptible. Extreme compression and cropping attacks for which the mark still survives are presented in Figure 2.

An additional interesting feature of the presented algorithm is that multiple marking is supported. For example, merchant M_1 can mark an image and sell the marked image to merchant M_2, who re-marks the image with its own mark, and so on. As an example, if ten successive markings are performed on Lena each marking with $p = 38dB$, the PSNRs with respect to the original image decrease as follows: 38, 35.1, 33.4, 32.28, 31.37, 30.62, 30.04, 29.5, 29.06 and 28.7. It is worth noting that multiple marking does not reduce the robustness of each individual marking. The result of five successive markings is presented in Figure 3.

4.2 Robustness of the Scale-Proof Scheme

Using the scheme of Section 3 with $p = 5$ and $r = 3$, the following StirMark manipulations were survived by the embedded mark:

Fig. 1. Left, original Lena. Right, Lena after embedding a 70-bit mark using the scheme of Section 2 with $q = 60\%$ and PSNR=38dB

Fig. 2. Scheme of Section 2. Left, marked Lena after JPEG 10% compression. Right, marked and cropped Lena. The mark survives both attacks.

Fig. 3. Scheme of Section 2. Lena after five successive markings.

1. Color quantization.
2. Most low pass filtering manipulations. More specifically:
   a) Gaussian filter (blur).
   b) Median filter (2×2 and 3×3).
   c) Frequency mode Laplacian removal [1].
   d) Simple sharpening.
3. JPEG compression for qualities 90% down to 20% (for some images down to 10%).
4. Rotations with and without scaling of -0.25 up to 0.25 degrees.
5. Shearing up to 1% in the X and Y directions.
6. Cropping up to 1%.
7. Row and column removal.
8. All StirMark scaling attacks (scale factors from 0.5 to 2).

Only small croppings are resisted because each tile embeds one bit of the mark; so, even if an ECC is used, we need at least $|E|$ tiles to be able to recover the mark. On the other hand, scaling is resisted because a mark bit is embedded in each pixel of a tile; even if the tile becomes smaller or larger, the correct bit can still be reconstructed. Extreme compression and scaling attacks for which the mark still survives are presented in Figure 4.

Fig. 4. Scheme of Section 3. Left, marked Lena after JPEG 15% compression. Right, marked Lena after 50% scaling. The mark survives both attacks.

5 Conclusion and Future Research

The two watermarking schemes presented in this paper operate in the spatial domain and are thus simple and computationally efficient. Both proposals behave well in front of compression and low-pass filtering. The most prominent feature of the first scheme is that its marks survive cropping attacks; like all publicly known crop-proof schemes, the one presented here requires the original image for mark reconstruction. The second scheme presented in this paper has the advantage over the first one of resisting scaling attacks; however, its tiling approach does not allow to survive substantial croppings.

It is probably impossible to come up with a *public-domain* watermarking algorithm (*i.e.* one that satisfies Kerckhoff's assumption of being entirely known) that can resist all kinds of manipulations. Therefore, future research will be devoted to combining the two schemes presented in this paper between them and with other schemes in the literature (one possibility is to use a different scheme for each color plane). We believe that surviving a broad range of manipulations will only be feasible through combined use of watermarking algorithms based on a variety of principles. In fact, one could imagine an expert system which proposes a particular combination of watermarking methods depending on the robustness requirements input by the user.

References

1. R. Barnett and D. E. Pearson, "Frequency mode L.R. attack operator for digitally watermarked images", *Electronics Letters*, vol. 34, Sep. 1998, pp. 1837-1839.
2. A. E. Bell, "The dynamic digital disk", *IEEE Spectrum*, vol. 36, Oct. 1999, pp. 28-35.
3. Digimarc,`http://www.digimarc.com`
4. J. Fridrich and M. Goljan, "Comparing robustness of watermarking techniques", in *Security and Watermarking of Multimedia Contents*, vol. 3567. San José CA: Soc. for Imaging Sci. and Tech. and Intl. Soc. for Optical Eng., 1999, pp. 214-225.
5. J. Fridrich, "Visual hash for oblivious watermarking", in *Proc. of SPIE Photonic West Electronic Imaging 2000, Security and Watermarking of Multimedia Contents*. San José CA: January 24-26, 2000.
6. F. Hartung and B. Girod, "Digital watermarking of raw and compressed video", in *Digital Compression Technologies and Systems for Video Communications*, SPIE Proceedings 2952, San José CA, 1996, pp. 205-213.
7. M. Kutter and F. A. P. Petitcolas, "A fair benchmark for image watermarking systems", in *Security and Watermarking of Multimedia Contents*, vol. 3657. San José CA: Soc. for Imaging Sci. and Tech. and Intl. Soc. for Optical Eng., 1999, pp. 226-239.
8. H. J. Lee, J. H. Park and Y. Zheng, "Digital watermarking robust against JPEG compression", in *Information Security*, LNCS 1729. Berlin: Springer-Verlag, 1999, pp. 167-177.
9. F. A. P. Petitcolas, R. J. Anderson and M. G. Kuhn, "Attacks on copyright marking systems", in *2nd International Workshop on Information Hiding*, LNCS 1525. Berlin: Springer-Verlag, 1998, pp. 219-239.
10. F. A. P. Petitcolas, R. J. Anderson and M. G. Kuhn, "Information hiding–A survey", *Proceedings of the IEEE*, vol. 87, July 1999, pp. 1062-1078.
11. B. Pfitzmann and M. Schunter, "Asymmetric fingerprinting", in *Advances in Cryptology-EUROCRYPT'96*, LNCS 1070. Berlin: Springer-Verlag, 1996, pp. 84-95.
12. J. R. Smith and B. O. Comiskey, "Modulation and information hiding in images", in *1st Workshop on Information Hiding*, LNCS 1174. Berlin: Springer-Verlag, 1996, pp. 207-226.
13. `http://www.lemuria.org/DeCSS`
14. `http://www.cl.cam.ac.uk/~fapp2/watermarking/benchmark/image_database.html`
15. Some of these authors, "Simple collusion-secure fingerprinting schemes for images", in *Proc. of An IEEE Conference*, 2000.