RR 02 060

Dewatermarking Based on Self-Similarities

J.-L. Dugelay, C. Rey, G. Doërr and G. Csurka

January 16, 2002

Foreword

Digital watermarking allows owners or providers to hide an invisible and robust message inside a digital multimedia document, mainly for security purposes (in particular owner or content authentication). There exists a complex trade-off between three parameters: capacity, visibility and robustness. Robustness in watermarking means that the retriever is still able to recover the watermark even if the protected document has undergone some attacks, malicious or not. A significant effort has been put in designing watermarking algorithms during the last few years[1][2]. But today, the watermarking community needs some fair benchmarks in order to compare the performances of different watermarking technologies according to some realistic scenario of applications. This state of mind motivates the creation of the European Certimark project[3].

In order to compare the robustness of different algorithms, some attacks need to be designed and integrated into relevant benchmarks. Indeed, attacks permit to find the weaknesses of an algorithm and consequently trigger further research in order to overcome the problem. Currently, Stirmark is one of the most efficient attack. It is mainly based on random local geometric distortions (quite impossible to overcome) of the cover which succeed to trap the synchronization between the encoder and the decoder. However, this attack does not really remove the watermark. The mark is still here even if the decoder is not able to find it. But on the other hand nothing insures the attacker that a possible future improved version of the decoder will not resolve the problem.

By analogy with denoising, we introduce the keyword *dewatermarking*. The perfect *dewatermarking* attack would consist to blindly restore the original document from the original one. In practice, by *dewatermarking*, we mean an attack that respects the following conditions:

1. it makes the retriever unable to recover the watemark;

2. it keeps the possibility to compute a quantitative measure of distortion, such as PSNR or wPSNR, between the protected document and the attacked one;

[1] *Information Hiding Techniques for Steganography and Digital Watermarking*, S. Katzenbeisser and F. Petitcolas, Artech House Books, 1999, ISBN 1-58053-035-4.

[2] *Digital Watermarking*, I. Cox, M. Miller and J. Bloom, Morgan Kaufmann Publishers, 2001, ISBN 1-55860-714-5.

[3] http://www.certimark.org

3. it creates a fair additional distortion, that is to say, the distace between the protected and attacked documments is close (even possibly lower) to the distance existing between the original and protected documents;

4. it ensures that a future improved version of the decoder alone cannot overcome the problem (the protection of the pictures is definitively lost and technology providers have to rework both embedder and retriever).

Many attacks proposed in the litterature can be classified as *dewatermarking* attacks. For example, lossy compression, denoising attack, template attack and copy attack belong to this type of attack. Our goal is to provide an efficient *dewatermarking* attack in order to evaluate watermarking softwares and we hope that in a near future our attack will be integrated in popular widespread watermarking tools like Certimark or Stirmark.

We have investigated an original attack based on self-similarities. The basic idea consists in substituing some parts of the picture (or using an external codebook) by some other ones that are or look similar. The aim is to approximate the watermark signal while keeping clear the main signal (i.e. cover). Like in fractal image coding, similarities can be expressed modulo a pool of possible photometric and geometric transformations and can be realized in the spatial domain as well as in the frequency domain (i.e. DCT), or spatio-frequency domain (i.e. wavelets) in order to be as generic as possible. Moreover, several ways can include a random aspect in the process in order to make the manipulation unpredictable.

We have then evaluated three watermarking softwares publicly available on Internet (D*******, S***I** and S***S***). Our attack succeeds to remove the three different watermarks. However it introduces too much distortion with S***S*** and we consider it as a failure for the moment. During the evaluation we notice that each of the tested algorithms favor one channel from a specific colour space to insert its watermark. This triggers our ongoing research on steganalysis. The aim is to blindly find which colour channel is the most likely watermarked and how strong it has been watermarked. With this valuable information, we will be able to blindly tune the different parameters of our algorithm.

Finally, we introduce the *antiwatermarking* concept by analogy with antivirus softwares. A basic framework has been defined for our *dewatermarking* attack based on self-similarities. However the parameters of the attack change from one watermarking algorithm to the other. The recent results in steganalysis may help to blindly set those parameters. As a result, as soon as a new watermarking software is launched, the attacker would only have to train the steganalysis module and to find the good parameters for the attack in order to keep its *dewatermarking* system up to date. From the point of view of the attacker, the watermark is indeed the virus to be removed!

The very first results of our investigations have been published during the French conference Coresa 2001 held in Dijon on November 12-13th. This work will be further presented during the conference Watermarking 2002 to be hold in Paris on March 5-8th and has been submitted on January 16th to ICIP 2002 to be hold in Rochester, USA

on September 22-25th. The submitted papers have been attached to this report for the interested reader.

ATTAQUE MALVEILLANTE D'IMAGES TATOUÉES BASÉE SUR L'AUTO-SIMILARITÉ[1]

Gabriella CSURKA, Jean-Luc DUGELAY, Caroline MALLAURAN, Jean-Pierre NGUYEN, Christian REY
Institut EURECOM, Département Communication Multimédia
2229 route des Crêtes B.P. 193, Sophia Antipolis, FRANCE
http://www.eurecom.fr/~image
jean-luc.dugelay@eurecom.fr

Résumé

Le tatouage d'images consiste à cacher de manière imperceptible et robuste une information dans une image, de manière à pouvoir extraire cette information, même si l'image a subi une attaque bien ou malveillante. Afin d'évaluer l'efficacité d'un algorithme de tatouage, il est important de tester sa robustesse par rapport à un ensemble de manipulations photométriques et géométriques classiques, compressions, mais également d'attaques malveillantes que l'image tatouée risque de subir. En conséquence, il est important de développer certaines attaques permettant de tester et donc d'améliorer les algorithmes de tatouage. Dans ce sens, l'objectif de ce papier est de proposer un algorithme d'attaque malveillante d'images tatouées en se basant sur la propriété d'auto-similarité des images.

Mots Clef

Tatouage d'images, évaluation, auto-similarité, attaque malveillante

1 Introduction

Le tatouage d'images consiste à cacher un filigrane digital imperceptible contenant un message dans une image de manière à pouvoir extraire ce filigrane (message) même si l'image a subi certaines manipulations bien ou malveillantes [3]. Depuis ces dernières années, beaucoup d'algorithmes de tatouage en images fixes ont été proposés. Certains algorithmes travaillent directement dans le domaine spatial, mais la plupart cachent le filigrane via un domaine transformé (la transformée discrète en cosinus, la transformée discrète de Fourier, les ondelettes ou les fractales).

Afin de pouvoir comparer ces systèmes de tatouage, il est nécessaire de tester leur résistance par rapport à des manipulations photométriques et géométriques classiques, compressions, mais également à des attaques malveillantes effectuées sur un même ensemble d'images de tests représentatives. Parmi de tels logiciels d'évaluation, on peut mentionner le logiciel Stirmark [4], qui propose non seulement une panoplie de manipulations géométriques et photométriques mais aussi l'attaque malveillante Stirmark, consistant en une succession de distorsions géométriques aléatoires appliquées localement à plusieurs endroits dans l'image. Immédiatement, cette attaque a mis en défaut la quasi totalité des tatoueurs. Depuis, certains tatoueurs ont réussi à améliorer leurs performances afin de résister à cette attaque.

Au sein de la communauté «watermarking», il existe depuis le départ, une sorte de compétition entre les «watermarkers» d'une part et les «crackers» d'autre part. Cependant, les recherches des «crackers» sont utiles aux recherches des «watermarkers». En effet, il est important de développer certaines attaques permettant d'évaluer et donc d'améliorer les algorithmes de tatouage. Parmi ces attaques malveillantes, nous pouvons distinguer celles qui perturbent l'image de telle sorte que, même si la marque reste présente dans l'image tatouée, le récupérateur de marque ne sait pas l'extraire sans avoir recours à l'image originale et celles qui «lessivent» la marque dans l'image.

Notre objectif est donc de définir, valider et tester un nouvel algorithme d'attaque malveillante basée sur les auto-similarités incluses dans les images. L'attaque optimale souhaitée ferait en sorte qu'avec une distorsion minimale de l'image et tout en conservant une performance comparable à celle de Stirmark, mais sans ajouter de distorsions géométriques, le récupérateur de marque soit suffisamment perturbé pour ne plus pouvoir extraire la marque correctement. Contrairement à Stirmark, il est ici toujours possible de calculer une erreur 'pixel' à 'pixel' entre les images tatouées obtenues avant et après attaque, et de rapprocher cette erreur avec celle introduite par le marquage (i.e. différence entre image originale et tatouée).

2 Méthode proposée

La principale caractéristique de l'approche proposée est l'exploitation de la notion d'auto-similarité présente dans les images. Les auto-similarités dans une image peuvent être considérées comme un type particulier de redondances. En effet, au lieu de rechercher la corrélation

[1] Ce travail a été, en partie, réalisé dans le cadre du Projet Européen - IST-1999-10987, CERTIMARK - *Certification for watermarking technique* (http://www.certimark.org).

entre les pixels adjacents, on s'intéresse ici à des corrélations entre des parties plus ou moins espacées dans l'image. L'idée des auto-similarités dans les images a été exploitée avec succès pour la compression fractale [2].

Au niveau du codage fractal, deux approches ont été développées : une première approche dans le domaine spatial [2] et une seconde dans le domaine transformé [1]. De ce fait, l'attaque proposée présente plusieurs déclinaisons possibles liées au domaine dans lequel on désire attaquer. Etant donné que certains algorithmes de tatouage travaillent dans le domaine spatial et que d'autres tatouent dans le domaine transformé, il semble intéressant de travailler sur les deux plans.

2.1 L'attaque spatiale

Dans le domaine spatial, l'image initiale est balayée bloc par bloc avec un recouvrement éventuel. Ces blocs sont appelés *Range block* (bloc **R**) de dimension donnée. Chaque bloc $\mathbf{R_i}$ est ensuite mis en correspondance avec un autre bloc transformé $\mathbf{D_j}$ lui « ressemblant » (modulo des ajustements photométriques et géométriques) au sens d'une mesure d'erreur *RMS* (Root Mean Squared) définie par :

$$RMS(f,g) = \frac{1}{n}\sqrt{\sum_{x=1}^{n}\sum_{y=1}^{n}\left[f(x,y)-g(x,y)\right]^2}$$

Le bloc $\mathbf{D_j}$, appelé *Domain block,* est recherché à travers une librairie composée de **Q** blocs appartenant à l'image. Les **Q** blocs ne forment pas nécessairement une partition de l'image. Chaque bloc $\mathbf{Q_i}$ est ramené à l'échelle de manière à être de même taille que $\mathbf{R_i}$ (si leurs tailles ne sont pas les mêmes). Il subit ensuite une transformation géométrique $T_\mathbf{k}$ parmi un ensemble de transformations prédéfinies (identité, 4 réflexions et 3 rotations de k*90°). Pour chaque bloc $\mathbf{Q_i}$ transformé ($T_\mathbf{k}$ ($\mathbf{Q_i}$)), la contraction photométrique (scaling **s**) et le décalage (offset **o**) sont calculés en minimisant l'erreur entre ce bloc $g=T_\mathbf{k}$ ($\mathbf{Q_i}$) et le bloc $f=\mathbf{R_i}$ par la méthode des Moindres Carrés :

$$R = \sum_{x=1}^{n}\sum_{y=1}^{n}\left(\mathbf{s}.g(x,y)+\mathbf{o}-f(x,y)\right)^2$$

Finalement, le bloc $\mathbf{D_i}$ mis en correspondance avec $\mathbf{R_i}$ est le bloc $\mathbf{s}\times T_\mathbf{k}(\mathbf{Q_i})+\mathbf{o}$ pour lequel la distance RMS est minimale.

Puisque le bloc $\mathbf{R_i}$ et le bloc $\mathbf{D_i}$ sont similaires, nous pouvons remplacer $\mathbf{R_i}$ par $\mathbf{D_i}$. Ainsi, le contenu de l'image va peu ou ne pas changer, mais les informations concernant le tatouage seront dispersées dans l'image et donc le décodeur sera incapable de retrouver les informations aux endroits prévus. L'inconvénient de cette approche est que tous les blocs n'ont pas de correspondants qui soient suffisamment similaires pour maintenir une qualité d'image acceptable (voir résultats expérimentaux).

2.2 L'attaque fréquentielle

L'approche via le domaine fréquentiel est inspirée du codage fractal dans le domaine transformé [1]. L'idée de base est de chercher pour la DCT (Transformée Discrète en Cosinus) du bloc $\mathbf{R_i}$ un bloc $\mathbf{D_i}$ transformé DCT. Mais, puisque les coefficients n'ont pas la même importance, le calcul global d'un «scaling **s**» et d'un «offset **o**» par bloc a peu de sens. Nous avons donc essayé d'utiliser plusieurs **s** et **o** en regroupant les coefficients selon les différents niveaux de fréquences. Cependant, nous avons rencontré une autre difficulté qui était de définir une mesure de ressemblance adéquate dans le domaine fréquentiel car une simple *RMS* ne tient pas compte des disparités entre les coefficients DCT. Une solution envisageable est d'introduire une forme de pondération ou bien d'utiliser des mesures plus complexes telle que la mesure Watson [6] qui est une mesure d'erreur agissant directement dans le domaine DCT.

Cependant, nous n'avons pas poursuivi nos investigations dans cette direction car nous avons choisi de développer une approche hybride «*spatio-fréquentielle*».

2.3 L'attaque spatio-fréquentielle

L'idée de base est de rechercher d'abord des blocs similaires dans le domaine spatial comme décrit pour «*l'attaque spatiale*», mais ensuite de transformer par la transformée discrète en cosinus les blocs $\mathbf{R_i}$ et $\mathbf{D_i}$ mis en correspondance dans le domaine direct. Afin de garder une meilleure qualité d'image, le bloc $\mathbf{R_i}$ conservera les *N* premiers coefficients DCT selon un parcours en zigzag (voir Figure 1). Les autres coefficients du bloc DCT($\mathbf{R_i}$) seront substitués par ceux du bloc DCT($\mathbf{D_i}$). Suite au calcul de la transformée discrète inverse en cosinus du bloc obtenu après les modifications des coefficients, ce dernier sera intégré dans l'image de départ pour remplacer le bloc $\mathbf{R_i}$.

1	**2**	**6**	**7**	**15**	**16**	**28**	**29**
3	**5**	**8**	**14**	**17**	**27**	**30**	**43**
4	**9**	**13**	**18**	**26**	**31**	**42**	**44**
10	**12**	**19**	**25**	**32**	**41**	**45**	**54**
11	**20**	**24**	**33**	**40**	**46**	**53**	**55**
21	**23**	**34**	**39**	**47**	**52**	**56**	**61**
22	**35**	**38**	**48**	**51**	**57**	**60**	**62**
36	**37**	**49**	**50**	**58**	**59**	**63**	**64**

Figure 1. *Parcours diagonal en zigzag dans un bloc de taille 8x8.*

Le compromis entre la qualité de l'image et l'efficacité de l'attaque est définie par le choix de *N*. Plus grand est *N* plus la qualité de l'image est préservée et inversement en diminuant *N* l'attaque devient plus efficace mais la qualité d'image diminue.

De plus, les tests menés ont montré qu'un *N* global n'était pas satisfaisant. Pour cette raison, le choix de *N* s'effectue localement en fonction de l'erreur entre $\mathbf{R_i}$ et $\mathbf{D_i}$ d'une part, et du contenu du bloc $\mathbf{R_i}$ d'autre part (zone uniforme, texturée, ou incluant des contours).

Finalement, afin d'éviter les effets blocs, les range blocs sont choisis avec un recouvrement et la substitution est effectuée avec un masque donné (dans notre cas, un cercle inscrit dans le bloc) ; c'est-à-dire que seule une partie du bloc définie par le masque est remplacée.

3 Résultats expérimentaux

Pour effectuer nos tests, nous avons utilisé plusieurs images de tailles différentes, plus ou moins texturées, souvent utilisées pour tester des tatoueurs [7]. Ces images sont présentées dans la Figure 2.

Figure 2. *Les images originales utilisées et leurs tailles* : *Baboon (512×512), Bear (394×600), Skyline_arch (400×594), Lena (512×512), Newyork (842×571)*

Nous avons évalué notre attaque en marquant les images avec comme tatoueur de référence D*******, qui reste, à l'heure actuelle, un des tatoueurs le plus utilisé.
Dans un premier temps, nous avons testé l'attaque «*spatiale*», c'est-à-dire l'attaque pour laquelle nous remplaçons chaque bloc $\mathbf{R_i}$ par le bloc $\mathbf{D_i}$. Les Figures 3 et 4 montrent les images Lena et Baboon tatouées et leurs correspondantes marquées et attaquées. Nous pouvons constater les dégradations sur les images attaquées. Par contre si nous appliquons l'attaque «*spatio-fréquentielle*», la qualité des images est préservée comme nous pouvons constater sur la Figure 5. La Figure 6. montre d'autres images marquées et attaquées avec l'attaque «*spatio-fréquentielle*». Dans les trois cas, comme pour Lena et Baboon, le marqueur D******* n'a retrouvé aucun filigrane après notre attaque.

Figure 3. *L'image Lena tatouée et l'image marquée, puis attaquée. Le PSNR entre les deux images est de 25.5dB.*

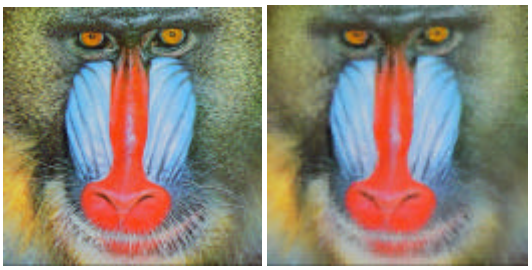

Figure 4. *L'image Lena tatouée et l'image marquée, puis attaquée. Le PSNR entre les deux images est de 19.25dB.*

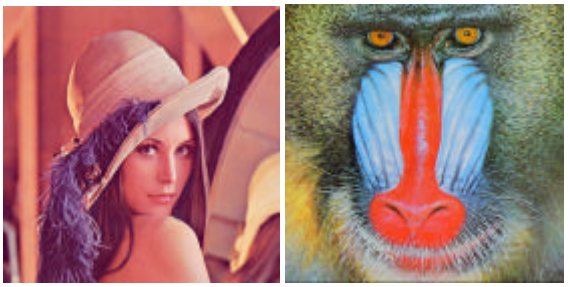

Figure 5. *Les images Lena et Baboon marquée, puis attaquées. Les PSNR entre les images tatouées et celles attaquées sont respectivement de 34.54dB et de 24.51dB.*

3.1 Analyse des résultats

Il est important de noter que par souci de ne pas perdre l'information par une simple compression JPEG, les tatoueurs récents insèrent le plus souvent les informations concernant le tatouage dans les fréquences moyennes. Il est donc important pour qu'une attaque soit efficace que les *N* coefficients qui ne seront pas remplacés soient entièrement dans les basses fréquences.

Figure 6. *Les images Skyline_arch, Newyork et Bear marquées et attaquées. Les PSNR entre les images tatouées et celles attaquées sont respectivement de 34.28dB, de 24.9dB et de 33.58dB.*

Mais comme nous l'avons dit précédemment, si *N* est trop petit, on diminue forcement la qualité d'image. Notre but était d'arriver à avoir une attaque efficace avec une distorsion équivalente à celle provoquée par le tatouage (≈38-40dB). Mais atteindre ce but n'est pas évident car les tatoueurs sont de plus en plus performants (grâce aussi à des attaques qui ont montré les faiblesses des anciens tatoueurs). En effet, les filigranes étant dépendants de l'image, il est difficile de les «effacer» ou même les «perturber» sans affecter les informations concernant l'image.

Finalement, il faut noter que les valeurs numériques (i.e. PSNR) mentionnées pour donner une indication sur la qualité des images ne sont pas très significatives. En effet, il est bien connu que le PSNR comme mesure de qualité n'est pas bien adapté (les images dans la Figure 7 en sont des bons exemples) et des mesures plus proches du système visuel humain (SVH) sont nécessaires pour mieux évaluer la distorsion introduite par l'attaque. Même si le PSNR est encore largement utilisé, des nouvelles mesures basées sur le SVH ont été proposées parmi lesquelles nous pouvons mentionner celle de Watson [6] ou Saadane et. al. [5].

4 Conclusion

Dans ce papier, nous avons présenté une attaque malveillante basée sur les auto-similarités dans les images. Une première déclinaison de cette attaque opère dans le domaine spatial, et une seconde dans le domaine fréquentiel (DCT). Cependant, afin d'avoir une attaque simple, efficace tout en préservant au mieux la qualité des images, nous avons proposé une attaque « *spatio-fréquentielle* » où la recherche des bloc similaires s'effectue dans le domaine spatial, mais la desynchronisation dans le domaine fréquentiel. L'attaque a été testée avec succès sur le tatoueur D*******.

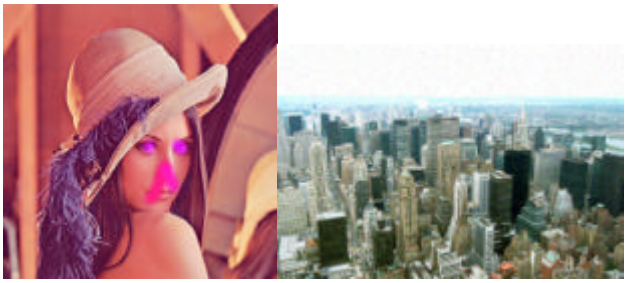

Figure 7. *L'image Lena marquée sur laquelle on a ajouté des tâches visibles et gênantes et l'image Newyork marquée sur laquelle nous avons ajouté des bruits gaussiens visibles. Les PSNR entre les images marquées et ces images manipulés sont plus grand (35.32dB pour Lena et 25.3dB pour Newyork) que dans le cas de notre attaque (34.54dB et 24.9dB) malgré le fait qu'il soit clair que visuellement nos images attaquées sont de qualités supérieures.*

Références

[1] Barthel (K-U), Schüttemeyer (J.), Noll (P.), « A new image coding technique unifying fractal and transform coding », IEE on Image Processing, Austin Texas, 13-16 November 1994.

[2] Fisher (Y.), « Fractal Image Compression – Theory and Application », Springer-Verlag, New-York, 1994.

[3] Katzenbeisser (S.), Petitcolas (F. A.P.), « Information Hiding – Techniques for Steganography and Digital Watermarking », Artech House, Boston-London, 2000.

[4] Kuhn (M. G.), Petitcolas (F. A.P.), Stirmark, 1997 : http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark/

[5] A. Saadane, N. Bekkat, D. Barda, « On the masking effects in a perceptually based image quality metric », Advances in the theory of computation and computational matematics book series, Vol. Imaging and Vision Systems, 2001.

[6] A. B. Watson. DCT quantization matrices visually optimized for individual images. Proceedings of SPIE : Human vision, Visual Processing and Digital Display IV, Vol. 1913, pp 202-216, 1993.

[7] Base d'image : http://www.cl.cam.ac.uk/~fapp2/watermarking/benchmark/image_database.html

TOWARD GENERIC IMAGE DEWATERMARKING?

C. Rey, G. Doërr, J.-L. Dugelay and G. Csurka

Institut Eurécom
Multimedia Communications
Sophia Antipolis, France.

ABSTRACT

A significant effort has been put in designing watermarking algorithms during the last decade. But today, the watermarking community needs some fair attacks and benchmarks in order to compare the performances of different watermaking technologies. Moreover attacks permit to find the weaknesses of an algorithm and consequently trigger further research in order to overcome the problem. This state of mind motivates the creation of the European Certimark project.

After a short definition of the keyword dewatermarking, we present an original attack based on self similarities. This attack is then put to the test with three different publicly available watermarking tools. Finally we shortly discuss the feasability of a generic attack i.e. a dewatermarking attack which should succeed in removing whatever watermark inserted by whatever watermarking tools.

1. INTRODUCTION

Image watermarking is now a major domain. Basically, digital watermarking allows owners or providers to hide an invisible and robust message inside multimedia content, often for security purposes, in particular owner or content authentication. There exists a complex trade-off between three parameters in digital watermarking: capacity, visibility and robustness. Robustness means that the retriever is still able to recover the hidden message even if the watermarked content has been altered after embedding. Today, most of the proposed watermarking schemes are robust against normal processing e.g. low pass filtering, JPEG compression. However most of them are still very weak against malicious attacks.

From the beginning, a competition between *attackers* and *watermarkers* has existed. Nevertheless, research from the attackers benefits to the whole watermarking community. As soon as a new attack is found, watermarkers try to improve their algorithms in order to survive to this new attack, often via a preventive procedure. Moreover it is necessary to develop attacks in order to set up benchmarks which will allow a fair comparison between the different proposed watermarking schemes. Stirmark[8] is currently recognized as one of the most efficient malicious attack. It is mainly based on random local geometric distortions (hard to prevent or to compensate) of the cover that traps the synchronization between the encoder and the decoder. But the watermark is still here and there is no guarantee for the attacker that a possible future improved version of the decoder will not resolve the problem.

In the present paper, we present an original attack which is assumed to definitely remove the watermark. In Section 2, we specify the basic requirements that an attack should met in order to be considered as a dewatermarking attack. In Section 3, we present our approach for still images based on self similarities. In Section 4, we show the performances of our attack against three publicly available watermarking tools. Finally we discuss in Section 5 the feasability of a generic dewatermarking attack.

2. IMAGE DEWATERMARKING

The keyword *dewatermarking* is partially self-explanary by analogy with denoising, even if it is not yet commonly used in the litterature. It means that the attack shoud not leave any underlying evidence of the presence of the watermark. It is fondamuntaly different from a desynchronization attack like Stirmark. When an attacker hacks a large database, he does not want to get caught the following month because a new version of the detector is not trapped any more by his attack. He wants to be sure that any copyright information has been removed once for all.

Obviously, the ideal dewatermarking attack would consist to blindly restore the original document from the watermarked one. But such a perfect attack is quite impossible to implement in practice. As a result, by dewatermarking, we mean an attack that fullfills the following specifications:

1. The detector is no longer able to recover the watermark.

2. The computation of a quantitative measure of distor-

This work has been supported by the Certimark[2] project.

tion, e.g. PSNR or wPSNR[10], between the watermarked document and the document resulting from the malicious manipulation remains pertinent i.e. the attack introduces no geometric distortion in order to remain compliant with the recent modelisation of the attack channel[7].

3. The attack should introduce a fair additional distortion. The distance between the watermarked and the attacked documents should be close (or even inferior) to the distance existing between the original and the watermarked documents. That is to say, the distance between the watermarked and the attacked documents is less than twice the distance between the original and the watermarked documents.

4. The attack should insure that a future improved version of the decoder alone cannot overcome the problem. The protection of the documents are definitely lost and technology providers have to rework both embedder and retriever.

Obviously, many traditionnal image processings (filtering, lossy compression) can be classified as dewatermarking attacks if they succeed to remove the watermark and some recent attacks[9] already fulfill those requirements.

3. APPROACH FOR STILL IMAGES

Our dewatermarking attack for still images basically exploits self-similarities of the image. Self similarities can be seen like a particular kind of redundancy. Usually correlation between neighbour pixels is taken into account. With self similarities, it is the correlation between different parts (more or less spaced) of the image which is of interest. This idea has already been used with success for fractal compression [4].

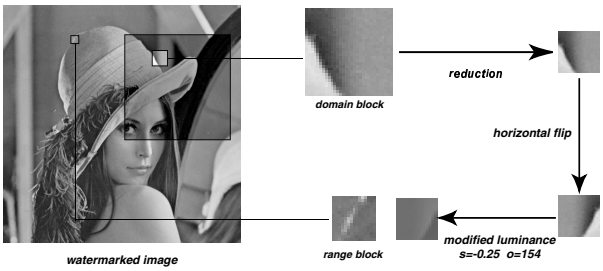

Fig. 1. Self similarities process

The basic idea of the attack consists in substituting some parts of the picture with some other parts of itself (or even from an external codebook) which are, or look, similar. This process is depicted in Figure 1 and explained in the next Subsection. The objective is to approximate, to stir the watermarked signal while keeping clear the cover signal. Even if self similarities can be realized in various transform domain (DCT[1], wavelet), we restrict here our presentation with the attack in the spatial domain.

3.1. Attack in the spatial domain

In the spatial domain, the original image is scanned one block after the other. Those blocks are labelled *range blocks* (block $\mathbf{R_i}$) and have a given dimension n. Each block $\mathbf{R_i}$ is then associated with another block $\mathbf{D_i}$ which looks similar (modulo a pool of possible photometric and geometric transformations) according to a Root Mean Square (RMS) metric defined by the following formula:

$$RMS(f,g) = \frac{1}{n}\sqrt{\sum_{x=1}^{n}\sum_{y=1}^{n}\Big(f(x,y) - g(x,y)\Big)^2} \qquad (1)$$

The block $\mathbf{D_i}$ is labelled *domain block* and is searched in a codebook containing Q blocks $\mathbf{Q_j}$. Those blocks may be blocks from the same image or from an external unwatermarked database. In practice, for a given range block $\mathbf{R_i}$, a window is randomly selected in the image. The blocks belonging to this window provide the codebook. Each block $\mathbf{Q_j}$ is scaled in order to match the dimensions of the range block $\mathbf{R_i}$. A set of $T_\mathbf{k}$ geometrically transformed blocks $T_\mathbf{k}(\mathbf{Q_j})$ is then built (identity, 4 flips, 3 rotations). For each transformed block $T_\mathbf{k}(\mathbf{Q_j})$, the photometric scaling $\mathbf{s}$ and offset $\mathbf{o}$ is computed by minimizing the error between the transformed block $g = T_\mathbf{k}(\mathbf{Q_i})$ and the range block $f = \mathbf{R_i}$ by the Least Mean Square method.

$$R = \sum_{x=1}^{n}\sum_{y=1}^{n}\Big(\mathbf{s}.g(x,y) + \mathbf{o} - f(x,y)\Big)^2 \qquad (2)$$

Eventually, the transformed block $\mathbf{s}.T_\mathbf{k}(\mathbf{Q_j}) + \mathbf{o}$ which has the lowest RMS distance with the range block $\mathbf{R_i}$ is found and the corresponding block $\mathbf{Q_j}$ will be the domain block $\mathbf{D_i}$ associated with the range block $\mathbf{R_i}$. Since the two blocks $\mathbf{R_i}$ and $\mathbf{D_i}$ looks similar, we can substitute $\mathbf{R_i}$ with the transformed version of $\mathbf{D_i}$. As a result, the image will be kindly modified but the watermark signal will be randomly spread through the image and the detector will be unable to retrieve it.

3.2. Additionnal specifications

Self similarities were not designed for attacks. In our case a perfect reconstruction is not expected. In fact we even want to insure a minimal error during the block association so that the watermark is removed. As a result, a threshold τ has been introduced and the rule to associate a domain block

with a range block has been modified. Now, for each range block, we search for the transformed block $\mathbf{s}.T_k(\mathbf{Q_j}) + \mathbf{o}$ which has the lowest RMS distance with the range block $\mathbf{R_i}$ *above the threshold* τ. If all the RMS distances are below the threshold, the block with the greatest distance is kept. In order to have an image dependent threshold, it is chosen in such a way that a given percentage p of the range blocks are not optimally substituted. As a result, two IFS iterations are needed. In the first iteration, the threshold is set to zero and the cumulative histogramme of the errors between the range blocks and the domain blocks is built. The adaptative threshold is then determined in order to interfere with p percents of the substitutions during the second iteration.

This new specification is likely to introduce visible artifacts. In order to prevent this effect, two constraints have been added:

- Only a given part of the domain block is substituted with the range block. In our case, we used a circular mask inscribed in the block.
- Overlapping range blocks have been used. Consequently, specific care must be taken during the reconstruction. A simple substitution is not any more pertinent. Instead the domain blocks are accumulated in a temporary image and, at the end, each pixel value is divided by the number of blocks that contribute to the value of this pixel.

4. EXPERIMENTAL RESULTS

This attack has been tested with three publicly available watermarking tools[3] that offer quite the same capacity (a few bits). A wide range of colour images have been tested, even if we only report the results with *lena* in this article. Moreover, we made the assumption that the attacker knows in which colour channel is embedded the watermark. Indeed, even if this hint is kept secret, it is quite easy to guess.

Our attack has been tested against D******* in a first experiment. The watermark seems to be mainly embedded in the V channel of the HSV colour space. We find out that around 60% of the block associations need to be disturbed in order to remove the watermark in quite all the images. This results in a quite good image quality as it can be seen in Figure 2. Visually one can notice that the textured areas are a little bit affected. The PSNR (resp. wPSNR)[1] is equal to 40.32 dB (resp. 53.90 dB) between the original image and its watermarked version, while it is equal to 35.67 dB (resp. 51.54 dB) between the watermarked image and its attacked version. As a result, we can call this attack a success.

In a second experiment, S***I** has been put to the test. The watermark is strongly embedded in the B channel

[1]The PSNR and the wPSNR are computed on the Y channel of YUV colour space.

(a) Original image

(b) D*******

(c) Attacked image

Fig. 2. Attack against D*******.

of the RGB colour space. In order to face the strength of the watermark, we need to disturb 99% of the block associations. It results in a strong degradation of the blue channel. But this degradation is quite invisible since the human eye is less sensible to the blue channel as it can be seen in Figure 3 which shows the luminance of the attacked image. The PSNR (resp. wPSNR) is equal to 49.05 dB (resp. 59.73 dB) between the original image and its watermarked version, while it is equal to 46.52 dB (resp. 59.24 dB) between the watermarked image and its attacked version. Once again, the attack is a success.

In the last experiment, we tested S***S***. The water-

(a) S***I**

(b) Attacked image

Fig. 3. Attack against S***I**.

(a) S***S*** (b) Attacked image

Fig. 4. Attack against S***S***.

mark seems to be mainly embedded in the channel Y of the colour space YUV. It has been determined experimentally that 92% of the block associations have to be disturbed in order to remove the watermark. This results in strong visible artifacts as can be seen in Figure 4. This time, the attack is a failure. However, the authors strongly believe that an attack based on self-similarities in the wavelet domain should work.

5. CONCLUDING REMARKS

We have described in this paper an efficient dewatermarking attack, which we expect to be widely used in the near future to test watermarking algorithms. It fulfills the requirements specified earlier and succeeds in trapping two of the three investigated watermarking schemes. However, even if all the proposed attacks have a common root (self similarities), the parameters of the attack differ (attacked colour channel, percentage p). It seems indeed difficult to build a generic dewatermarking attack. This is due to the high specialization of the watermarking technologies. Defeating one watermarking algorithm does not mean the others will be defeated. For example, a simple averaging filter of width 5 usually removes the watermark inserted by D*******. On the other hand, it will leave the watermarked inserted by S***I** or S***S*** unaffected! Anyway, having a pool of dedicated attacks is not completely useless.

Recently some researchers found some exciting results in steganalysis[6]. The authors showed that it is possible to predict if an image has been watermarked and by which technology. So now we have a toolbox containing multiple simple attacks optimized for a single technology in one hand, and an oracle which is able to say which watermarking technology has been used in the other hand. Combine those two items together and you obtain a very powerful tool for attackers. We can now make a straightforward analogy with an antivirus software. For any new incoming watermarking technology (the virus), the attackers only have to design a simple dewatermarking attack (the antivirus) and to update the oracle. As a result, if an attacker does not want to get caught, he just has to keep his system up to date.

6. REFERENCES

[1] K.-U. Barthel, J. Schüttemeyer and P. Noll "A new image coding technique unifying fractal and transform coding", in *IEE on Image Processing*, Austin, USA, November 13-16 1994.

[2] Certimark, http://vision.unige.ch/certimark

[3] D*******, http://www.d*******.com
S***I**, http://www.a********.com
S***S***, http://www.s*********.com

[4] Y. Fisher, *Fractal Image Compression: Theory and Application*, editor Springer-Verlag, New York, 1995.

[5] M. Kutter, F. Jordan and F. Bossen, "Digital signature of color images using amplitude modulation", in *Proceedings of Electronic Imaging*, San Jose, USA, February 1997.

[6] N. Memon I. Avcibas and B. Sankur, "Steganalis of watermarking techniques using image quality metrics", in *Proceedings of SPIE Security and Watermarking of Multimedia Contents III*, San Jose, USA, January 22-25 2001, vol. 4314.

[7] P. Moulin and J. O'Sullivan, "Information theoric analysis od information hiding", *Preprint*, September 1999.

[8] F. Petitcolas, R. Anderson and M. Kuhn, "Attacks on copyright marking systems", in *Proceedings of Information Hiding*, Portland, USA, April 15-17 1998.

[9] K. Tsang and O. Au, "A review on attacks, problems and weaknesses of digital watermarking and the pixel reallocation attack", in *Proceedings of SPIE Security and Watermarking of Multimedia Content III*, San Jose, USA, January 22-25 2001, vol. 4314.

[10] S. Voloshynovskiy, A. Herrigel, N. Baumgaertner and T. Pun, "A stochastic approach to content adaptive digital image watermarking", in *Third International Workshop on Information Hiding*, Dresden, Germany, September 29 - October 1 1999.