

Tatouage robuste aux attaques de désynchronisations

Omar Berrezoug¹ et Marc Chaumont²

LIRMM, UMR CNRS 5506, Université de Montpellier II,
161 rue Ada, 34392 Montpellier cedex 5, France.
Université de Nîmes, Place Gabriel Péri, 30000 Nîmes, France.

Contact : ¹ Omar.Berrezoug@lirmm.fr
² Marc.Chaumont@lirmm.fr

Résumé

Les grandes familles de tatouage d'image robustes aux désynchronisations sont toutes apparues entre 1998 et 2002. Dans cet article nous reprenons un schéma basé contenu apparu en 2002 : le schéma de Bas et al. [2]. Ce schéma présente un fort potentiel vis-à-vis d'attaques plus difficiles qu'une simple attaque de rotation-changement-d'échelle-translation. Malheureusement, ce dernier est très sensible aux attaques à cause de l'utilisation d'un détecteur de points caractéristiques non robuste. Nous proposons donc une amélioration du schéma par utilisation de points caractéristiques multi-échelles : les blobs. De plus, nous renforçons la détectabilité de ces points grâce à un rehaussement d'amplitude. Enfin, nous améliorons la robustesse générale du schéma par un tatouage dans les moyennes fréquences à travers l'utilisation de la transformée en ondelettes. Nos résultats montrent que l'approche proposée est bien plus robuste que l'approche initiale.

Abstract

The different families of image-watermarking robust to the desynchronizations all appeared between 1998 and 2002. In this paper we carry on the content-based approach of Bas and al. [2] appeared in 2002. This scheme presents a strong potential with respect to attacks more difficult than a simple rotation-scaling-translation attack. Unfortunately, this scheme is very sensitive to attacks because it uses non-robust characteristic points. We thus propose an improvement of this scheme by using Gaussian scale-space characteristic points : the blobs. Moreover, we reinforce the detectability of those points by increasing of their magnitude. Lastly, we improve the general robustness of the scheme by embedding in the wavelet transform domain. Our results show that our scheme is more robust than the initial one.

Mots-clés : Tatouage robuste, désynchronisation, tatouage basé contenu, blobs, transformée en ondelettes.

Keywords: robust watermarking, desynchronizations, content-based watermarking, blobs, wavelet transform.

1. Introduction

La plupart des documents numériques échangés aujourd'hui peuvent se retrouver sur Internet. N'importe qui peut les copier, les modifier et les exploiter à son profit. Il est donc indispensable de proposer des solutions de protection des droits d'auteurs face au piratage et à la contrefaçon. Vers les années 90 une solution est apparue, appelée tatouage (en anglais *Watermarking*) [5].

Le tatouage est une technique permettant d'insérer au sein même du document numérique (signal hôte), audio (parole ou musique), visuel (image fixe ou vidéo) voire synthétique (3D) une information. L'information insérée est appelée marque. Cette information peut être un identifiant du créateur, du propriétaire, ou de l'acheteur du document, etc. Dans la Littérature on distingue plusieurs types de tatouage : fragile, 0-bit/multi-bits, robuste, etc [3].

Le tatouage robuste consiste à cacher un signal, appelé marque, dans un signal hôte en respectant les contraintes de perceptibilité. Dans ce cas, il faut pouvoir garantir que l'information insérée puisse résister à des transformations (licites et/ou illicites); l'information doit rester présente même si le document numérique a subi une dégradation. Un bon système de tatouage est donc un système robuste aux attaques non désynchronisantes mais aussi aux attaques désynchronisantes. Une attaque désynchronisante peut se traduire par une déformation géométrique (rotation, translation, changement d'échelle, etc.) globale ou locale [13]. Comme mentionné dans [3], la robustesse contre les transformations géométriques reste l'un des problèmes les plus difficiles du tatouage. La déformation géométrique causera une erreur de synchronisation qui peut nettement détériorer l'extraction et/ou la détection du tatouage.

Dans la littérature on trouve trois grandes familles de tatouage robuste aux attaques de désynchronisations :

- le tatouage basé sur le calcul d'espace invariant [8,9,13],
- le tatouage basé sur pattern de synchronisation (insertion d'un template) [6,12],
- le tatouage basé contenu (synchronisation implicite, tatouage de seconde génération) [2,7,11].

Le schéma de Bas et al. [2] présente un fort potentiel vis-à-vis d'attaques plus difficiles qu'une simple attaque rotation-changement-d'échelle-translation. Malheureusement, dans sa forme originale, le schéma de Bas et al. est très sensible aux attaques. Le détecteur de points caractéristiques utilisé (Harris) est en effet non robuste. Dans cet article nous proposons donc une amélioration de ce schéma par utilisation de points caractéristiques plus robustes (blobs), et nous améliorons la robustesse générale par un tatouage (étalement de spectre) dans les moyennes fréquences à travers l'utilisation de la transformée en ondelettes.

Dans la section 2, nous présentons les différentes étapes de notre algorithme de tatouage. La section 3 présente les résultats expérimentaux. Nous concluons en section 4.

2. Algorithme proposé

Dans ce travail nous utiliserons un détecteur de points caractéristiques multi-échelle : les blobs.

2.1. Détection des blobs et amplification

Le blob est une position dans l'image qui localise une région possédant certaines caractéristiques : une zone forte de texture, une zone de contour, etc. Théoriquement, les blobs sont robustes contre la compression avec perte, les rotations, les translations, les changements de luminance, et les changements de contraste. Pour la robustesse au changement d'échelle on utilise la détection dans " l'espace-échelle Gaussien " [10].

Pour obtenir les blobs nous filtrons par convolution l'image originale (I_{orig}) avec des filtres gaussiens appelés LOG-filtre (Laplacian Of Gaussian) de tailles croissantes Eq(1), et d'écart type σ_r :

$$\sigma_r = \{\sigma_r \in \mathbb{R} : \sigma_{min} \leq \sigma_r \leq \sigma_{max}, r \in \mathbb{N} : r \leq R, R \in \mathbb{N}\} :$$

$$I_{LOG(\sigma_r)} = LOG(\sigma_r) * I_{orig}. \quad (1)$$

Le noyau LOG-filtre est calculé suivant l'équation Eq(2) :

$$LOG(x, y, \sigma_r) = \left(\frac{x^2 + y^2}{2 \cdot \pi \cdot \sigma_r^6} - \frac{1}{\pi \cdot \sigma_r^4} \right) \cdot \exp \left(-\frac{(x^2 + y^2)}{2 \cdot \sigma_r^2} \right). \quad (2)$$

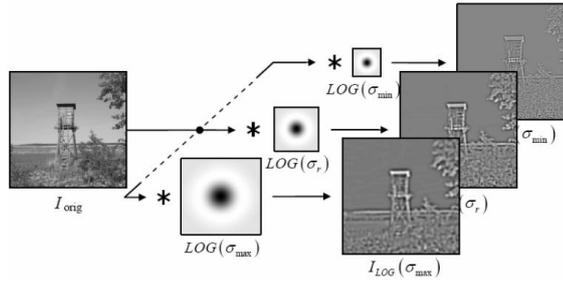
Les images filtrées avec différentes valeurs de σ_r (figure (1)) sont normalisées en utilisant Eq(3) :

$$I_{LOG(\sigma_r)}^* = \sigma_r \cdot I_{LOG(\sigma_r)}. \quad (3)$$

La valeur de σ_{opt} est calculée en comparant toutes les images filtrées ($I_{LOG(\sigma_r)}^*$) en chaque point (x,y) tel que :

$$\sigma_{opt}(x, y) = \operatorname{argmax}_{\sigma_r} \left| I_{LOG(x, y, \sigma_r)}^* \right|. \quad (4)$$

Nous calculons alors une nouvelle image $I_{LOG(\sigma_{opt})}$ de même dimension que l'image originale I_{orig} .

FIGURE 1 – Image filtrée par LOG-filtre avec différents $\sigma_r \in [3, 5]$. Image extraite de l'article [10].

Un blob correspond à une zone de forte amplitude de $I_{LOG(\sigma_{opt})}$, avec une distance suffisante entre les différents blobs. On détermine donc les blobs en triant les valeurs de $I_{LOG(\sigma_{opt})}$ par ordre décroissant, tel que la distance entre les blobs respecte l'équation Eq(5) :

$$2 \cdot [\sigma_{opt}(x_i, y_i) + \sigma_{opt}(x_j, y_j)] \leq d_{ij}, \quad (5)$$

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (6)$$

Pour être sûr de pouvoir détecter les mêmes blobs lors de la phase d'extraction, l'amplitude des blobs est amplifiée par G_{diff} . L'amplification se fait par l'addition d'un négatif ou positif LOG-masque $LOG_{gain}^*(x, y, \sigma_{gain})$, directement sur l'image initiale I_{orig} , Eq(7), figure(2).

Soit $\sigma_{gain}(x, y) = \sqrt{2} \cdot \sigma_{opt}(x, y)$, on a alors :

$$LOG_{gain}^*(x, y, \sigma_{gain}) = \frac{G_{diff}}{K(x, y)} \cdot LOG_{gain}(x, y, \sigma_{gain}), \quad (7)$$

avec

$$K(x, y) = \sigma_{opt}^2 \cdot \sum_{x=x-\Delta x}^{x=x+\Delta x} \sum_{y=y-\Delta y}^{y=y+\Delta y} LOG_{gain}(x, y) \cdot LOG_{opt}(x, y). \quad (8)$$

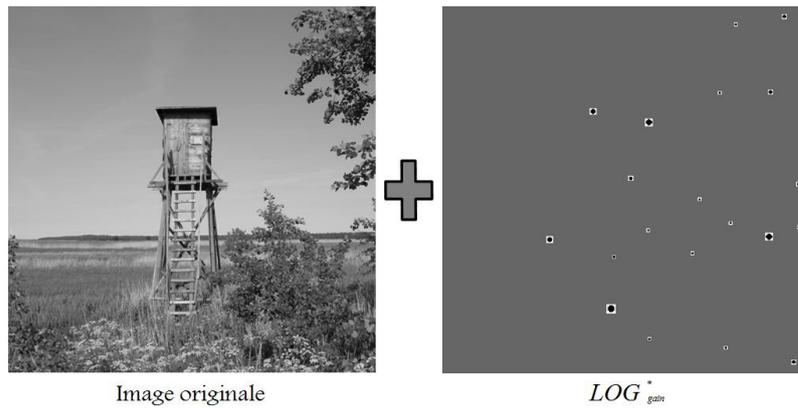


FIGURE 2 – Amplification des blobs.

2.2. Insertion de la signature

La figure(3) montre les différentes étapes d'insertion de la signature :

1- Un vecteur pseudo-aléatoire est généré (signature), à l'aide d'un générateur pseudo aléatoire.

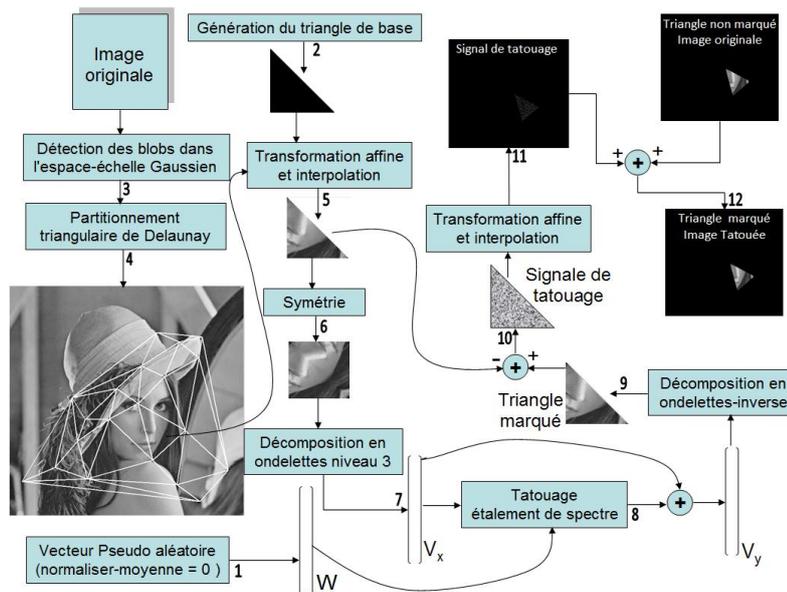


FIGURE 3 – Schéma d'insertion.

Ce vecteur est composé d'une succession de +1 et de -1, d'une taille égale au vecteur signal hôte définit par la suite. Ce vecteur sera modifié pour rendre sa moyenne égale à zéro, et ensuite normalisé. Nous désignons par W ce vecteur ;

2- Un triangle de base est généré (Triangle isocèle rectangle), d'une taille fixe. Nous choisissons comme taille du carré englobant le triangle égale à 128×128 ;

3- Une détection des points caractéristiques est appliquée sur l'image originale. Nous utilisons le détecteur de blobs multi-échelle, présenté dans la section 2.1 ;

4- Un partitionnement triangulaire de Delaunay [4] est appliqué à partir de l'ensemble des blobs. Nous obtenons un ensemble de triangles : $T = \{T_i\}, 0 \leq i \leq N_t$, avec N_t est le nombre de triangle ;

5- Chaque triangle T_i est projeté sur le triangle de base par une transformation affine ;

6- Une opération de symétrie est appliquée sur le triangle de base, pour obtenir "l'image carré" ;

7- Une décomposition en ondelettes (Haar) en niveau 3, est appliquée sur le rectangle englobant. Le signal hôte (V_x) est défini par une sélection des coefficients ondelettes. On sélectionne les coefficients de moyenne fréquence, qui sont à la fois assez robustes, et qui ont une bonne propriété psycho-visuelle. Voir les détails sur la figure (4) ;

8- Un tatouage par étalement de spectre est appliqué sur le signal hôte tel que $V_y = V_x + \alpha W$. avec α la force d'insertion, et V_y le signal hôte tatoué ;

9- Après le remplacement des coefficients marqués dans le domaine ondelette, on fait une décomposition en ondelettes inverse, pour obtenir le triangle marqué ;

10- Le signal de tatouage est obtenu en retranchant le triangle marqué au triangle non marqué ;

11- Le signal de tatouage (sous forme de triangle) est projeté sur l'image avec la transformation affine inverse ;

12- L'image tatouée est obtenue en additionnant chaque triangle non marqué avec le signal de tatouage correspondant.

2.3. Détection de la signature

Le principe de l'algorithme de détection est illustré sur la figure(5). Les sept premières étapes (génération du vecteur, triangle de base, détection de blobs, partitionnement de Delaunay, transformation affine, symétrie, décomposition en ondelettes) sont identiques au schéma d'insertion.

1- Une corrélation normalisée est calculée entre le vecteur provenant de la sélection des coeffi-

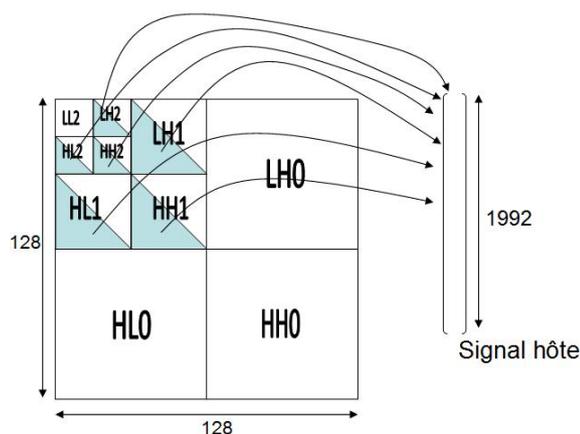


FIGURE 4 – Sélection des coefficients ondelette.

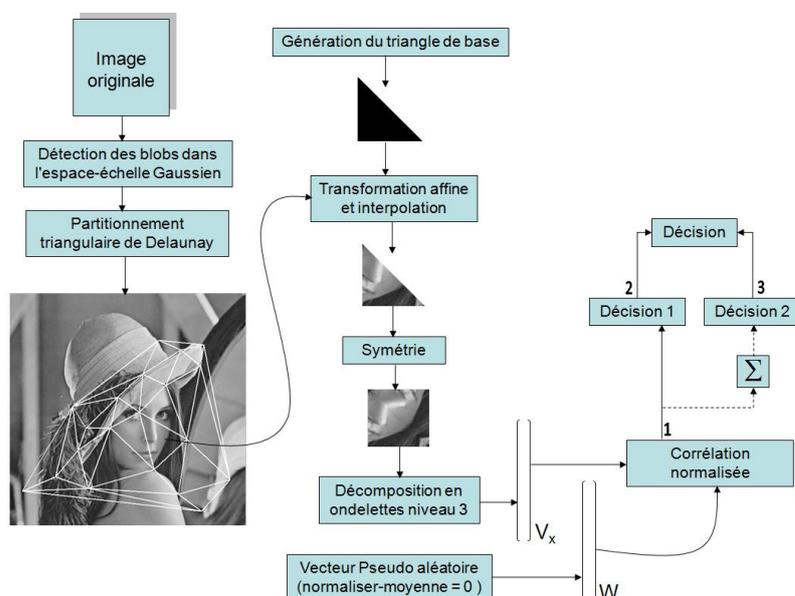


FIGURE 5 – Shéma de détection.

cients d’ondelette, et le vecteur W généré en première étape Eq(9) ;

$$Z_{vw}(V_x, W) = \frac{V_x}{\|V_x\|} \cdot \frac{W}{\|W\|} = \frac{1}{\|V_x\| \|W\|} \sum_{i=1}^N V_x [i] \cdot W [i] \quad (9)$$

Avec N la taille du signal hôte.

2- Une première décision locale est effectuée en fonction de la valeur de corrélation (corrélation normalisée) obtenue pour chaque triangle. Si la corrélation dépasse un certain seuil alors on dit que la signature est présente dans le triangle concerné ;

3- Une décision globale est ensuite effectuée en fonction du nombre de triangles marqués par rapport au nombre de triangles détectés. Si ce nombre dépasse un certain seuil alors on dit que l’image est tatouée.

3. Expérimentations et résultats

Cette partie est divisée en deux sous parties, la première est dédiée aux tests de la robustesse des points caractéristiques utilisés (Blobs), et la deuxième est dédiée à l'évaluation de notre système de tatouage. Les expérimentations ont été réalisées sur un PC Core 2 duo T5750, 3 GB de mémoire, et avec MATLAB 7.6 comme outil de programmation. Pour les tests nous avons utilisé huit images en niveau de gris (512×512) de contenus très différents.

3.1. Robustesse des blobs

Dans cette partie nous testons la robustesse des blobs aux diverses attaques de désynchronisation. Nous avons appliqué aux images originales notre détecteur de blobs, et une amplification de ceux-ci. Ensuite nous avons attaqué les images dont les blobs ont été rehaussés avec diverses attaques de désynchronisation (une rotation de 10° , un changement d'échelle de +12%, un étirement horizontal de 10%, un étirement vertical de 10%, et une compression JPEG avec perte de facteur de qualité 75%), et nous avons réappliqué à nouveau notre détecteur de blobs aux images attaquées figure(6). Pour comparer la robustesse des blobs à l'approche originale, nous avons également utilisé le détecteur de Harris. Les résultats de test sont présentés sous forme d'histogramme figure(7), où l'axe des abscisses représente les différentes images avec les diverses attaques, et l'axe des ordonnées représente le rapport du nombre des triangles stables sur le nombre de triangles détectés (Pourcentage des triangles stables) :

$$\text{Pourcentage des triangles stables} = \frac{\text{Nombre de triangles détectés à la même position}}{\text{Nombre total des triangles détectés}}. \quad (10)$$

Les résultats obtenus Figure(7) montrent que les blobs sont robustes contre diverses attaques de

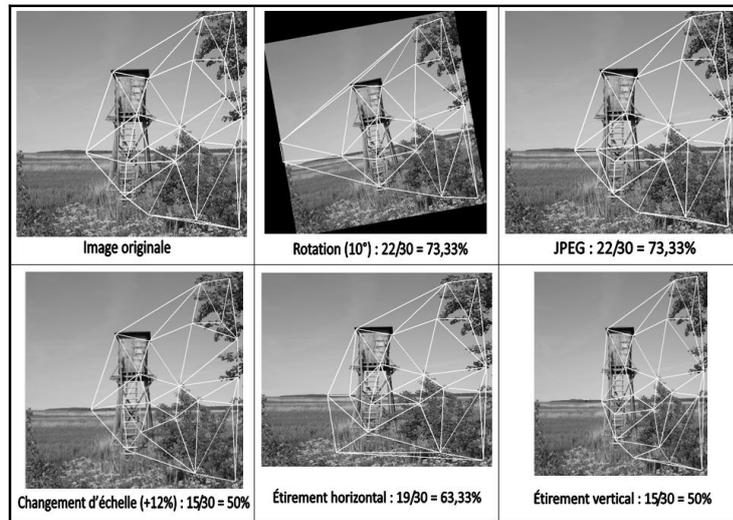


FIGURE 6 – Robustesse des blobs contre diverse attaques.

désynchronisations (rotation, changement d'échelle, étirement H/V, compression JPEG, etc.) avec un nombre de triangle stable supérieur à 40% (voir Eq 10) et une stabilité bien plus élevée que celle des points de Harris.

3.2. Évaluation de notre système de tatouage

Pour évaluer notre système de tatouage, nous avons fixé les paramètres suivants : $G_{diff}=15$, $M(\text{Nombre de blobs détectés})=20$, $\sigma_{min} = 3$, $\sigma_{max} = 5$, la force d'insertion $\alpha = 0.03$, le seuil de corrélation (corrélation normalisée) = 0.03, le seuil de détection (décision globale) = 40.00%. La durée d'insertion est de 131 sec. La durée de détection est de 50 sec.

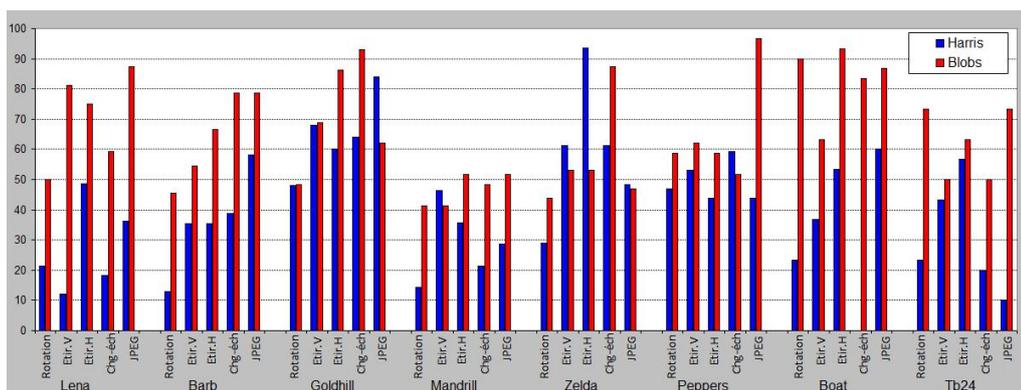


FIGURE 7 – Résultats du comparaison

3.2.1. Avant l'attaque

Le tableau Tab.1 représente le nombre des triangles détectés pour chaque image (avant et après le tatouage), le pourcentage de détection, le PSNR, et les résultats de détecteur. On peut consta-

Image (512 × 512)	Images non tatouées			Images tatouées		PSNR (db)
	Nombre triangle	Triangle_détecté Image_non_tatouée		Triangle_détecté Image_tatouée		
Lena	32	5/32	15.62%	24/32	75.00%	43.079
Barb	33	3/33	9.09%	25/33	75.75%	42.349
Goldhill	29	2/29	6.89%	13/29	44.82%	43.651
Mandrill	29	1/29	3.44%	17/24	58.62%	42.182
Zelda	32	3/32	9.37%	23/32	71.87%	41.290
Peppers	29	2/29	6.89%	24/29	82.75%	41.311
Boat	30	3/30	10.00%	25/30	83.33%	42.624
Tb24	30	2/30	6.66%	19/30	65.51%	42.813

TABLE 1 – Résultats du tatouage.

ter que lorsque l'image est tatouée, le pourcentage de triangles détectés (Eq 10) est supérieur à 40%. En contrepartie, lorsque l'image n'est pas tatouée, le pourcentage de triangles détectés est inférieur à 20%. Avec un seuil sur le pourcentage de triangles détectés fixé à 40%, les images sont toutes bien classées (l'image est classée comme tatouée ou non tatouée).

On peut également remarquer que certains triangles sont détectés marqués alors qu'ils ne le sont pas (ce sont des faux positifs) et que certains triangles sont détectés non marqués alors qu'ils le sont (ce sont de faux négatifs). La raison principale est le manque de robustesse de l'insertion par étalement de spectre. L'utilisation d'une méthode "à la " Broken Arrows [1], qui est une méthode informée, donnerait vraisemblablement de meilleurs résultats.

En choisissant un tatouage par étalement de spectre (c'est-à-dire non informé) et un PSNR de valeur supérieur à 41 (ce qui implique une bonne qualité visuelle des images tatouées mais une faible force d'insertion et donc une faible robustesse), on peut tout de même conclure que le système de tatouage est fiable (lorsqu'il n'y a pas d'attaque).

3.2.2. Après l'attaque

Les résultats de cette partie sont présentés sous forme d'histogramme figure(8), où l'axe des abscisses représente les différentes images avec diverses attaques, et l'axe des ordonnées représente le pourcentage des triangles détectés marqués.

On remarque une présence de faux positifs et de faux négatifs au niveau triangles et même au

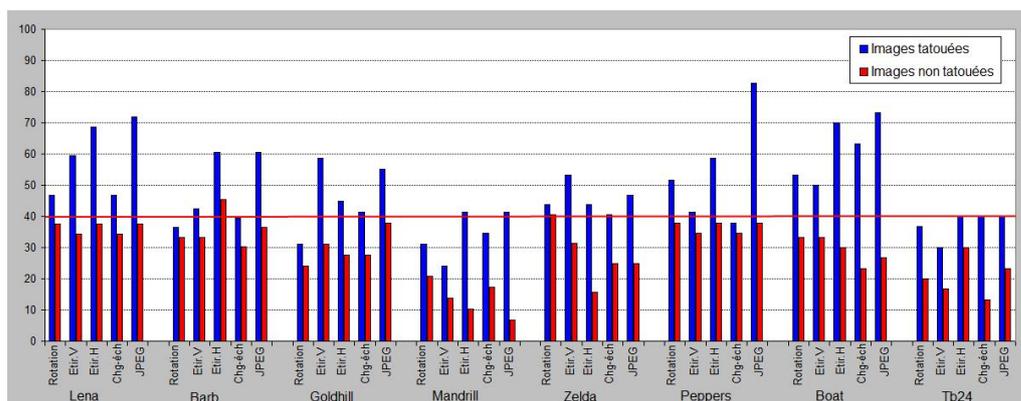


FIGURE 8 – Robustesse de notre système aux attaques.

niveau images. Il y a un faux positif pour l'image Barb, et des faux négatifs pour les images Barb, Goldhill, Mandrill, et Tb24. Encore une fois, les erreurs de détection (i.e classification) sont dues à la méthode de tatouage (étalement de spectre). En effet, si la méthode était plus robuste, plus de 40% des triangles devraient être détectés marqués lorsqu'ils le sont, et toutes les images tatouées attaquées devraient donc être détectées comme tatouées.

4. Conclusion et perspectives

Dans cet article nous avons présenté un algorithme de tatouage d'image, basé sur le schéma de Bas et al. [2]. L'utilisation des points caractéristiques multi-échelle (blobs), et l'insertion du tatouage dans le domaine ondelettes, ont permis d'obtenir plus de robustesse que le schéma initial. Les résultats obtenus ont confirmé que le partitionnement triangulaire basé sur les blobs est plus robuste que celui basé sur les points de Harris.

Pour améliorer la robustesse du schéma de tatouage aux attaques, nos travaux futurs consisteront à renforcer la robustesse de notre approche en changeant la méthode de tatouage, par exemple en utilisant le principe d'insertion utilisé dans l'algorithme de Broken Arrows [1]. Nous chercherons également à étendre l'approche en proposant une insertion multi-bits avec utilisation de codes détecteurs et correcteurs d'erreurs gérant l'effacement.

Bibliographie

1. T. Furon and P. Bas., *Broken Arrows*, EURASIP Journal on Information Security, vol. 2008, 2008.
2. P. Bas, J. M. Chassery, et B. Macq. *Geometrically invariant watermarking using feature points*. Image Processing, IEEE Transactions on, 11, 9, pp 1014-1028.2002.
3. I. Cox, M. Miller, et J. Bloom. *Digital watermarking*. Morgan-Kaufmann, San Francisco, CA, ISBN : 1-55860-714-5.2002.
4. F. Davoine. *Compression d'images par fractales basée sur la triangulation de Delaunay*. Thèse de l'institut national polytechnique de Grenoble, France.1995.
5. F. Davoine et S. Pateux . *Tatouage de documents audiovisuels numériques*. Traité IC2 - HERMÈS SCIENCE PUBLICATIONS, LAVOISIER. juillet 2003.
6. A. Herrigel, J. J. K. O'ruanaidh, H. Petersen, S. Pereira, et T. Pun. *Secure copyright protection techniques for digital images*. In Proceedings of the International Workshop on Information Hiding. 169-190.1998.
7. M. Kutter, S. Bhattarjee, et T. Ebrahimi. *Towards second generation watermarking schemes*. In Proceedings of the IEEE International Conference on Image Processing '99. Vol. I. IEEE Computer Society Press, Los Alamitos, CA, 320-323.1999.
8. Y. Liu, D. Zheng, et J. Zhao. *A rectification scheme for RST invariant image watermarking*. IEICE Trans. Fundament. (Special Section on Cryptography and Information Security) E88-A, 1.2005.
9. J. O'ruanaidh, et T. Pun. *Rotation, scale, and translation invariant digital image watermarking*. Signal Process. 66, 3, 303-317.1998.
10. M. Schlauweg, D. Profrock, B. Zeibich, et E. Muller. *Self-Synchronizing Robust Texel Watermarking in Gaussian Scale-Space*. Multimedia & Security ACM Workshop MMSEC2008, Oxford, United Kingdom.22-23 September 2008.
11. C. W. Tang, et H. M. Hang. *A feature-based robust digital image watermarking scheme*. IEEE Trans. Sig. Process. 51, 4 (April), 950-959.2003.
12. S. Voloshynovskiy, F. Deguillaume, et T. Pun. *Multibit digital watermarking robust against local nonlinear geometrical distortions*. In Proceedings of the IEEE International Conference on Image Processing (ICIP). IEEE Computer Society Press, Los Alamitos, CA, 999-1002.2001.
13. D. Zheng, Y. Liu, J. Zhao, et A. El Saddik. *A survey of RST invariant image watermarking algorithms*. ACM Comput. Surv. 39, 2, Article 5 , 91 pages DOI = 10.1145/1242471.1242473 http :// doi.acm.org/10.1145/1242471.1242473. June 2007.