Un nouvelle technique pour le tatouage par Dirty Paper Trellis Code (DPTC)

Marc CHAUMONT

Université de Nîmes, Place Gabriel Péri, 30000 Nîmes, France,
Université de Montpellier II, Laboratoire LIRMM, UMR CNRS 5506,
161, rue Ada, 34392 Montpellier cedex 05, France
Marc.Chaumont@lirmm.fr

Résumé – Le schéma de tatouage par codes à papier sale par treillis (Dirty Paper Trellis Code : DPTC), publié en 2004, est un des schémas à forte capacité parmi les plus performants. Il possède cependant deux inconvénients majeurs : sa faiblesse en terme de sécurité et sa complexité en coût de calcul. Nous proposons de traiter ces deux problèmes par l'utilisation d'un espace d'insertion plus sûr et par l'utilisation d'une technique d'insertion plus rapide. L'espace d'insertion est construit par projections des coefficients ondelettes sur des porteuses secrètes. Cela renforce la sécurité, et cela permet d'obtenir de bonnes propriétés psycho-visuelles. L'insertion, quant à elle, repose sur une rotation dichotomique dans le plan de Cox, Miller et Bloom. Cette insertion donne de meilleures performances par rapport aux approches d'insertion rapides précédemment proposées. Quatre attaques différentes sont utilisées pour l'évaluation et les résultats obtenus montrent un bon comportement du schéma en terme de robustesse et de complexité opératoire.

Abstract – Dirty Paper Trellis Codes (DPTC) watermarking, published in 2004, is a very efficient high rate scheme. Nevertheless, it has two strong drawbacks: its security weakness and its CPU computation complexity. We propose a more secure embedding space and a faster embedding. The embedding space is built on the projections of some wavelet coefficients onto secret carriers. It re-enforces the security but also has good psycho-visual properties. The embedding is based on a dichotomous rotation in the Cox, Miller and Boom Plane. It gives better performance than previous fast embedding approaches. Four different attacks are performed and revealed good robustness and rapidity performances.

1 Introduction

Les schémas informés (également appelés schéma à information adjacente) sont apparus en 1998 lorsque le travail de Costa a été redécouvert [1]. On distingue deux grandes catégories de systèmes de tatouage informés multi-bits : les schémas basés codes à lattice, également appelés codes en réseau, et plus couramment appelés schéma basés quantification (DC-QIM [2], SCS [3]...) et les schémas basés treillis (DPTC [4]).

L'algorithme original basé DPTC est connu pour sa bonne robustesse et sa haute capacité d'insertion, mais possède deux grosses faiblesses : l'étape d'insertion informée utilise une approche Monte-Carlo très complexe en coût de calcul et le schéma montre quelques faiblesses de sécurité vis à vis d'attaques par collusion [5]. Dans ce papier nous proposons un schéma DPTC plus sûr et moins complexe.

Lin *et al.* [6] proposent de remplacer l'approche Monte-Carlo par une technique non-optimale mais de faible complexité. Nous proposons une solution encore plus efficace. Nous utilisons le domaine ondelettes qui ne produit pas les « effets de bloc » du domaine DCT (domaine utilisé dans l'approche DPTC initiale). Pour renforcer la sécurité et rendre encore plus difficile l'attaque présentée dans [5], nous réalisons l'insertion du signal de tatouage dans un espace secret. Finalement, puisque notre technique est rapide et que l'espace d'insertion est adapté, nous augmentons la taille du treillis (plus exactement nous augmentons la taille du *livre du code* [1]) ce qui nous permet d'augmenter la robustesse [7] et de réduire la distorsion.

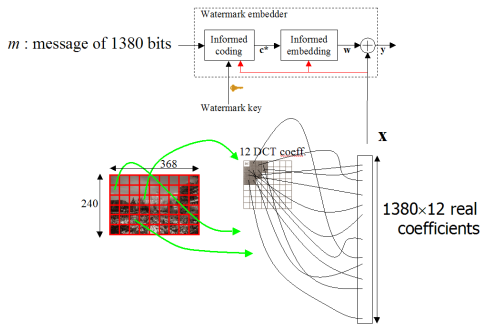


FIGURE 1 – Dirty Paper Trellis Codes appliqués sur une image 240×368

Dans la section 2, nous rappelons le principe du DPTC original [4]. Dans la section 3 nous présentons l'espace d'insertion et l'algorithme d'insertion. Enfin, dans la section 4 nous comparons le DPTC original [4], l'approche de Lin *et al.*[6] et notre approche basée rotation [2].

2 Le schéma DPTC original

Le schéma DPTC original [4] appliqué sur une image $N = 240 \times 368$ est illustré sur la Figue 1.

La première étape du schéma consiste à transformer l'image

1. livre du code : *codebook* en anglais.

2. Ce travail est en partie supporté par le projet VOODDO (2008-2011) de l'ANR (Agence Nationale pour la Recherche) "Contenu et Interaction".

en une représentation spatio-fréquentielle (par transformée DCT) pour obtenir le *signal hôte* $\mathbf{x}$. Dans le schéma original, on applique à l'image une transformée DCT 8×8, puis les douze premiers coefficients ACs de chaque bloc DCT sont extraits et ordonnés pseudo-aléatoirement dans un vecteur $\mathbf{x}$ de taille $12 \times N/64 = 3 \times N/16$.

La deuxième étape du schéma DPTC est le *codage informé*. Le message m à insérer est codé en un mot de code $\mathbf{c}^*$ en prenant en compte le *signal hôte* $\mathbf{x}$. Pour réaliser ce codage, un treillis non déterministe et l'algorithme de Viterbi sont utilisés.

La dernière étape du schéma DPTC est l'*insertion informée*. Cette étape consiste à modifier le *signal hôte* $\mathbf{x}$ pour le « déplacer » dans la région de Voronoï du mot de code $\mathbf{c}^*$. Les auteurs de [4] utilisent une approche Monte-Carlo (avec un critère de robustesse prédéfini) pour effectuer le « déplacement » du *signal hôte* $\mathbf{x}$ dans la région de Voronoï du mot de code $\mathbf{c}^*$. L'approche est itérative et consiste à attaquer (i.e dégrader) le *signal tatoué* $\mathbf{y}$ puis contre-attaquer en regénérant un nouveau *signal tatoué* $\mathbf{y}$. Cette technique nécessite un grand nombre d'appels à l'algorithme de Viterbi. Même avec les optimisations proposées dans [4], la complexité en temps de calcul est élevée et c'est actuellement une forte limitation, que cela soit pour son évaluation intensive ou bien pour son utilisation au sein d'un système logiciel ou matériel [3].

3 Notre schéma DPTC

Dans cette section, nous présentons notre nouvel espace d'insertion et notre nouvelle approche pour l'insertion.

3.1 Le nouvel espace d'insertion

Le travail récent de Bas et Doërr [5] sur la sécurité de l'approche DPTC [4] montre que dans le cadre de Kerckhoffs [8], c'est-à-dire lorsque les algorithmes d'insertion et d'extraction sont publics et donc à la disposition d'attaquants, le *livre du code* $\mathcal{C}$ peut-être retrouvé [4] en observant un grand nombre d'images tatouées (tatouées avec l'utilisation de la même clef secrète). Ce résultat est obtenu sur une version simplifiée de l'algorithme DPTC [5], mais montre néanmoins l'existence d'une certaine faiblesse de sécurité dans l'algorithme. L'espace privé que nous utiliserons dans notre schéma permet de cacher la structure du treillis et devrait rendre plus difficile une attaque à la sécurité du type de celle proposée par Bas et Doërr [5].

Notre nouvel espace d'insertion est obtenu tout d'abord par une transformation ondelettes de l'image, puis par une projection du *signal hôte* $\mathbf{x}$ de dimension N_{wlt} ($\mathbf{x}$ est la concaténation des coefficients des sous-bandes, exceptés les coefficients de la sous-bande LL) sur N_{sec} porteuses (notée $\mathbf{u}_i$ avec $i \in [1, N_{sec}]$). On note que la projection est juste un produit scalaire. Remarquons également que la complexité de la projection peut facilement être réduite en une complexité linéaire avec une approche de Space Division Multiplexing [9]. Le vecteur obtenu $\mathbf{v_x}$ de dimension N_{sec} peut alors être utilisé pour le *codage informé* et *l'insertion informée*.

Cet espace d'insertion est plus sûr que l'original car il permet de disperser le signal de tatouage sur tout le domaine fréquentiel (il n'y a donc pas de super robustesse [10]). De plus, la projection sur les N_{sec} porteuses donne à l'espace d'insertion un aspect Gaussien connu pour ses bonnes capacités de canal [1]. Finalement, le domaine ondelettes a de bonnes propriétés psycho-visuelles et génère des effets moins gênants (ou du moins, moins tranchés) que les effets de bloc du schéma DPTC original [4].

3.2 L'algorithme d'insertion

Le codage informé que nous utilisons est le même que celui du DPTC original, mais il est réalisé à partir du vecteur hôte $\mathbf{v_x}$ (c'est-à-dire à partir de l'espace secret). Le codage-informé permet d'obtenir le mot de code $\mathbf{c}^*$. Pour accélérer l'insertion et conserver un bon compromis robustesse-distorsion, nous proposons une solution non optimale, mais meilleure que l'approche de de Lin *et al.* [6].

Rappelons qu'au décodage, le mot de code le plus corrélé $\tilde{\mathbf{c}^*}$ est obtenu en exécutant l'algorithme de Viterbi sur le treillis « complet ». Ce mot de code $\tilde{\mathbf{c}^*}$ appartient au *livre du code* $\mathcal{C}$ et maximise la corrélation avec le vecteur tatoué-attaqué $\tilde{\mathbf{v_y}}$ tel que :

$$\begin{aligned} \tilde{\mathbf{c}^*} &= \arg_{\mathbf{c}^i} \max_{\forall \mathbf{c}^i \in \mathcal{C}} (\tilde{\mathbf{v_y}}.\mathbf{c}^i) \\ &= \arg_{\mathbf{c}^i} \max_{\forall \mathbf{c}^i \in \mathcal{C}} (||\tilde{\mathbf{v_y}}||.||\mathbf{c}^i||.cos\theta_i), \end{aligned}$$

avec θ_i l'angle entre $\tilde{\mathbf{v_y}}$ et $\mathbf{c}^i$. Sachant que tous les mots de code possèdent la même norme, l'algorithme de Viterbi extrait donc le mot de code $\mathbf{c}^i \in \mathcal{C}$ formant le plus petit angle $|\theta_i|$ avec $\tilde{\mathbf{v_y}}$. En supposant que le bruit d'attaque est un bruit blanc gaussien, l'attaque n'a aucune influence sur la correlation et l'on retrouve, lors de l'extraction, le mot de code $\mathbf{c}^*$ utilisé lors de l'insertion.

Pour insérer le message m lors du codage, il suffit donc de réduire l'angle entre le *vecteur hôte* $\mathbf{v_x}$ et le mot de code $\mathbf{c}^*$, jusqu'à obtenir le plus petit angle (parmi tous les autres angles possibles $\widehat{(\mathbf{v_x}, \mathbf{c}^i)}$).

Pour réduire l'angle entre $\mathbf{v_x}$ et $\mathbf{c}^*$, nous exprimons d'abord les deux vecteurs dans le plan de Miller, Cox et Bloom (*abrév.* plan MCB) [11]. La figure 2 illustre ce plan. Le plan MCB est défini par une base ortho-normalisée ($\mathbf{v_1}$, $\mathbf{v_2}$) telle que le $\mathbf{v_x}$ et $\mathbf{c}^*$ appartiennent à ce plan (algorithme de Gram-Schmidt) :

$$\mathbf{v_1} = \frac{\mathbf{c}^*}{||\mathbf{c}^*||}, \qquad \mathbf{v_2} = \frac{\mathbf{v_x} - (\mathbf{v_x}.\mathbf{v_1})\mathbf{v_1}}{||\mathbf{v_x} - (\mathbf{v_x}.\mathbf{v_1})\mathbf{v_1}||}$$

Dans le plan MCB, les coordonnées 2D du *vecteur hôte* $\mathbf{v_x}$ sont : $\mathbf{v_x^{2D}}(1) = \mathbf{v_x}.\mathbf{v_1}$, $\mathbf{v_x^{2D}}(2) = \mathbf{v_x}.\mathbf{v_2}$, et les coordonnées 2D du mot de code $\mathbf{c}^*$ sont : $\mathbf{c^*_{2D}}(1) = ||\mathbf{c}^*||$, $\mathbf{c^*_{2D}}(2) = 0$.

Une rotation du *vecteur hôte* $\mathbf{v_x^{2D}}$ d'un angle θ dans le plan

3. Sur un PC à 3Ghz, en fonction du seuil de robustesse, l'insertion peut prendre 30 min à 1 heure 30 sur une image 256×256.

4. Plus exactement, ce sont les coefficients attachés aux arcs du treillis qui peuvent être assez bien estimés.

5. il n'y a pas de ré-ordonnancement pseudo-aléatoire des coefficients DCTs.

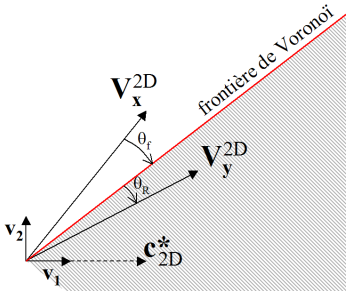


FIGURE 2 – Insertion basée rotation dans le plan de Miller, Cox et Bloom

MCB est telle que :

$$\begin{aligned}\mathbf{v_y^{2D}}(1) &= cos\theta.\mathbf{v_x^{2D}}(1) - sin\theta.\mathbf{v_x^{2D}}(2),\\ \mathbf{v_y^{2D}}(2) &= sin\theta.\mathbf{v_x^{2D}}(1) + cos\theta.\mathbf{v_x^{2D}}(2).\end{aligned}$$

Si nous réduisons la valeur absolue de l'angle entre le *vecteur hôte* $\mathbf{v_x}$ et le mot de code $\mathbf{c}^*$ dans le plan MCB, cela augmente la corrélation $\mathbf{v_x}.\mathbf{c}^*$. Avec une approche dichotomique sur l'angle de rotation, on peut rapidement trouver une frontière de Voronoï. L'algorithme consiste à itérer une dizaine de fois les instructions suivantes :

1. tourner $\mathbf{v_x}$ et obtenir $\mathbf{v_y}$ dans le plan MCB,
2. exécuter l'algorithme de Viterbi sur le treillis « complet » et tester si $\mathbf{v_y}$ appartient ou non à la région de Voronoï de $\mathbf{c}^*$ i.e. vérifier si le vecteur décodé est égal ou non à $\mathbf{c}^*$,
3. modifier l'angle de rotation en fonction du résultat d'appartenance à la région de Voronoï region et retourner en 1.

Une fois que l'angle frontière θ_f dans le MCB est trouvé, on améliore la robustesse d'insertion en pénètrant à l'intérieur de la région de Voronoï avec un angle θ_R. Notre insertion informée est donc une rotation du *vecteur hôte* $\mathbf{v_x}$ d'un angle orienté égal à $\max(\theta_f + \theta_R, (\widehat{\mathbf{v_x}, \mathbf{c}^*}))$. La figure 2 illustre $\mathbf{v_x}$, $\mathbf{v_y}$, θ_f et θ_R dans le plan MCB.

4 Resultats

Les expérimentations ont été réalisées sur les 100 premières images de la base de données de BOWS-2 [6] avec des images redimensionnées en 256×256 [7]. Ces images sont des images en niveau de gris codées sur 8 bits et sont des photos personnelles.

La structure du treillis possède 128 (resp. 64) états avec 128 (resp. 64) arcs par états pour l'algorithme de Lin *et al.* **basé cône** et notre algorithme **basé rotation** (resp. pour l'algorithme **DPTC original**). Les étiquettes des arcs de sortie sont tirées d'une distribution Gaussienne et il y a 12 coefficients par arc. La capacité d'insertion est la même que dans l'article sur les DPTC [4], c'est-à-dire 1 bit pour 64 pixels. Le nombre de bits insérés est donc de 1024.

Pour l'algorithme de Lin *et al.* basé cône, et notre algorithme basé rotation, nous utilisons le même espace d'insertion. La transformée ondelettes est une 9/7 Daubechies avec $l = 3$ niveaux de décomposition. Exceptée la sous-bande LL, toutes les autres sous-bandes sont utilisées pour former le *signal hôte* $\mathbf{x}$. Avec des images 256×256, la taille de l'espace ondelettes est de $N_{wlt} = 64\,512$ coefficients. Sachant que la capacité est de $1/64$ *bits par pixel* et que le nombre de coefficients de sortie pour un arc est $N_{arc} = 12$ coefficients, la taille de l'espace privé est donc de $N_{sec} = 1024 \times 12 = 12\,288$ coefficients. Les porteuses $\mathbf{u}_i$ sont construites à partir de séquences pseudo-aléatoires bipolaires normalisées.

Quatre attaques à la robustesse ont été expérimentées : l'attaque par ajout de bruit Gaussien, l'attaque par filtrage, l'attaque valumétrique et l'attaque par compression jpeg. Le Taux d'Erreur Binaire (Bit Error Rate : BER) est calculé à partir du message extrait et il est égal au nombre de bits erronés divisé par le nombre total de bits insérés. Le BER est calculé pour chaque attaque. Trois algorithmes sont en compétition pour un PSNR d'insertion proche de 42.6 dB : le **DPTC original** (dans l'espace DCT) avec un PSNR moyen d'insertion = 42.6 dB, l'algorithme de Lin *et al.* **basé cône** (dans l'espace ondelette secret) avec la robustesse positionnée à un bruit de puissance $n^2 = 1$ (ce qui correspond à $R_t = 1$ dans le papier [6]) et un PSNR moyen d'insertion = 34.2 dB (remarquons qu'il est impossible d'augmenter le PSNR de Lin *et al.* Leur technique est très sous-optimale lorsque l'on expérimente avec des images réelles) et notre algorithme **basé rotation** (dans l'espace ondelette secret) avec un angle de pénétration de $\theta_R = 0.1$ radian et un PSNR moyen d'insertion = 42.4 dB.

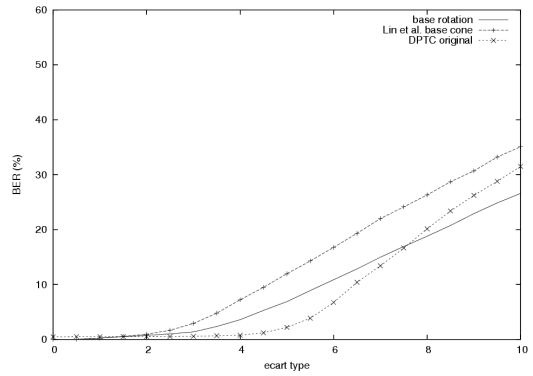
Sur la figure 3 nous donnons les différents résultats de BER pour les quatre attaques. D'emblée, nous écartons les résultats de l'approche de Lin *et al.* puisqu'il est impossible d'avoir un PSNR moyen supérieur à 34.2 dB. L'algorithme de Lin *et al.* ne peut donc pas être utilisé comme substitut rapide à l'algorithme DPTC original, puisqu'il n'est pas capable d'atteindre un PSNR raisonnable de 42 dB. La comparaison est donc uniquement réalisée entre l'algorithme DPTC original et notre algorithme basé rotation.

Pour les comparaisons, nous regardons uniquement le BER inférieur à 10%. Avec ce critère, l'algorithme basé rotation obtient des résultats identiques ou proches du DPTC original, pour l'attaque par ajout de bruit Gaussien et l'attaque par filtrage Gaussien. Les deux approches diffèrent fortement pour l'attaque jpeg et l'attaque valumétrique. Pour l'attaque par compression jpeg, le DPTC original est très robuste alors que l'algorithme basé rotation ne l'est pas. Les résultats de performances sont opposés pour l'attaque valumétrique, puisque l'algorithme basé rotation est bien plus robuste que l'algorithme DPTC original.

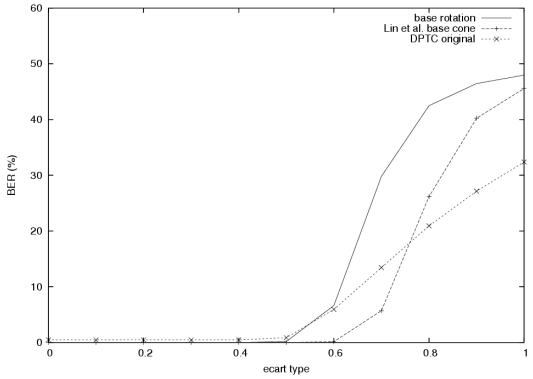
Nous pouvons conclure que notre approche basée rotation est actuellement la meilleure approche (sachant qu'elle est meilleure que l'approche référence de *Lin et al.*) pour réduire la

6. la base de données de BOWS2 est téléchargeable à l'adresse http ://bows2.gipsa-lab.inpg.fr/.

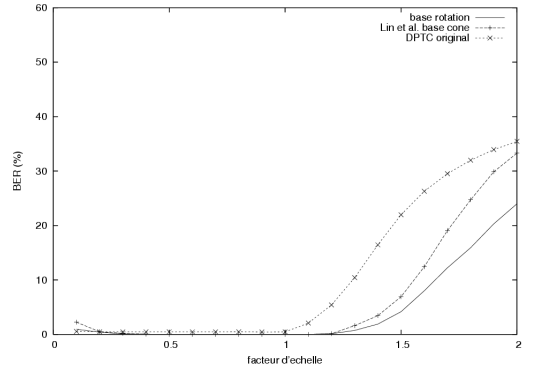
7. le sous-échantillonnage d'images a été réalisé avec le programme xnview et utilisant l'interpolation de Lanczos.



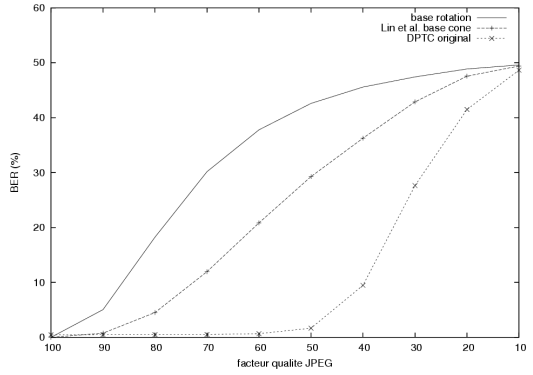
a) Attaque par bruit Gaussien



b) Attaque par filtrage Gaussien



c) Attaque Valumétrique



d) Attaque Jpeg

FIGURE 3 – BER pour les attaques sur l'algorithme DPTC original, l'algorithme de Lin *et al.* basé cône et notre algorithme basé rotation.

complexité de l'algorithme DPTC original. De plus, notre approche garantit une bonne sécurité via l'utilisation d'un espace d'insertion sûr [12].

5 Conclusion

Dans ce papier, nous présentons un nouvel algorithme de codes à papier sale par treillis (Dirty Paper Trellis Code : DPTC). Par rapport à l'algorithme original, nous utilisons le domaine ondelette au lieu du domaine DCT. La sécurité est améliorée par l'ajout d'un espace d'insertion secret. Cet espace secret est obtenu en projetant les coefficients ondelettes sur des porteuses orthogonales. Cette projection garantit également (pendant la rétro-projection) une dispersion du signal de tatouage sur tous les coefficients ondelettes. Nous présentons également une nouvelle approche d'insertion basée rotation. L'objectif est de déplacer le *vecteur hôte* dans la région de Voronoï du mot de code représentant le message. Le déplacement consiste à faire faire une rotation au *vecteur hôte* par rapport à l'axe défini par le mot de code. L'angle de rotation utilisé pour déterminer la frontière de Voronoï est obtenu de manière dichotomique. Le *vecteur hôte* est ensuite « tourné » de sorte qu'il y ait une pénétration dans la région de Voronoï d'un angle prédéfini. Les résultats sont meilleurs que l'approche rapide de Lin *et al.*. et sont bons en comparaison de l'approche originale [4]. De plus, notre approche est de bien plus faible complexité calculatoire que l'approche originale [4].

Références

[1] M. Costa, "Writing on dirty paper," *IEEE Transactions on Information Theory*, vol. 29, no. 3, pp. 439–441, 1983.

[2] B. Chen and G. Wornell, "Quantization index modulation : A class of provably good methods for digital watermarking and infromation embedding," *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1423–1443, 2001.

[3] J. J. Eggers, R. Bäuml, R. Tzschoppe, and B. Girod, "Scalar Costa Scheme for Information Embedding," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 1003–1019, 2003.

[4] M. L. Miller, G. J. Doërr, and I. J. Cox, "Applying Informed Coding and Informed Embedding to Design a Robust, High Capacity Watermark," *IEEE Transactions on Image Processing*, vol. 13, no. 6, pp. 792–807, 2004.

[5] P. Bas and G. Doërr, "Evaluation of an Optimal Watermark Tampering Attack Against Dirty Paper Trellis Schemes," in *10th ACM workshop on Multimedia and Security, MM&Sec'2008*, Oxford, United Kingdom, Sept. 2008, pp. 227–232.

[6] L. Lin, I. J. Cox, G. Doërr, and M. L. Miller, "An Efficient Algorithm for Informed Embedding of Dirty Paper Trellis Codes for Watermarking," in *IEEE International Conference on Image Processing, ICIP'2005*, Genova, Italy, Sept. 2005, vol. 1, pp. 697–700.

[7] C. Wang, G. Doërr, and I. J. Cox, "Toward a Better Understanding of Dirty Paper Trellis Codes," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'2006*, Toulouse, France, May 2006, vol. 2, pp. 233–236.

[8] A. Kerckhoffs, "La Cryptographie Militaire," *Journal des Sciences Militaires*, vol. IX, pp. 5-38 Jan. 1883, pp. 161-191, Feb. 1883.

[9] M. Chaumont, "A Fast Embedding Technique For Dirty Paper Trellis Watermarking," in *8th International Workshop on Digital Watermarking, IWDW'2009*, University of Surrey, Guildford, United Kingdom, Aug. 2009.

[10] S. Craver, I. Atakli, and J. Yua, "How we broke the BOWS watermark," in *IS&T/SPIE 19th Annual Symposium on Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX edited by Edward J. Delp III, Ping Wah Wong, SPIE'2007*, Jan. 2007, vol. 6505, pp. 1–8.

[11] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, chapter 5, pp. 142–143, in Multimedia Information and Systems. Morgan Kaufmann, 2nd edition edition, Nov. 2007.

[12] T. Furon and P. Bas, "Broken Arrows," *EURASIP Journal on Information Security*, vol. 2008, 2008.