Le tatouage de documents numériques
Cours 1

Marc Chaumont

12 novembre 2008

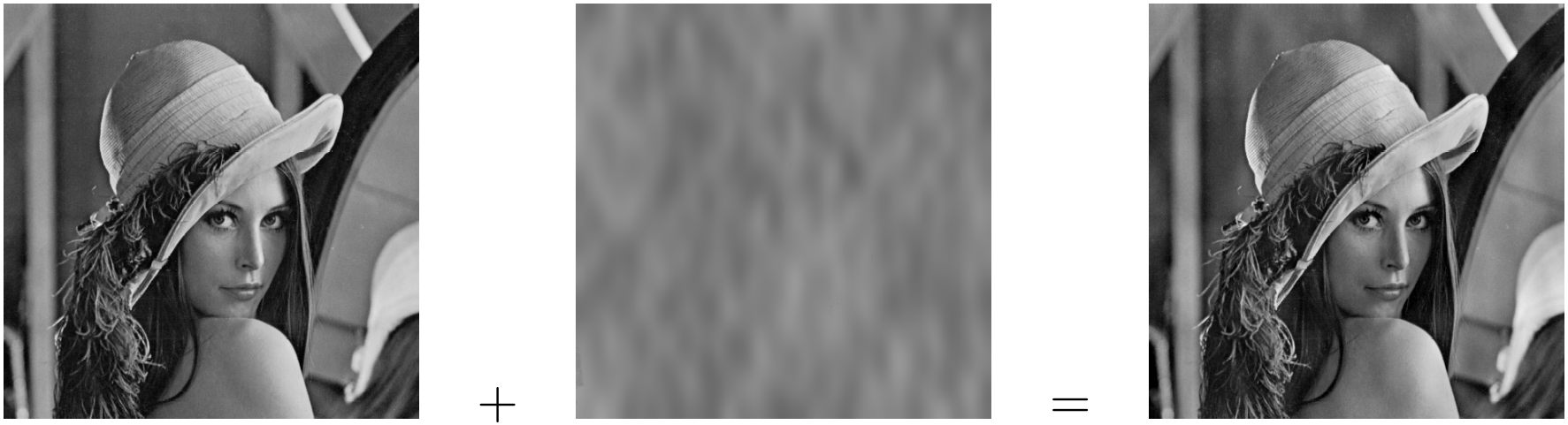
Plan

Tatouage visible et tatouage invisible pour une image

Fig.: Le tatouage visible et invisible sur une image

Note : Nous nous intéresserons dans ce cours au tatouage invisible.

Exemple de système de tatouage : Cox et al. 97



Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal G. Shamoon. "Secure spread spectrum watermarking for multimedia". In Proceedings of the IEEE International Conference on Image Processing, ICIP '97, volume 6, pages 1673- 1687, Santa Barbara, California, USA, 1997.

Définition du tatouage (watermarking)

Le tatouage - Insertion de données cachées

Le tatouage est l'art d'altérer un média (une image, un son, une vidéo ...) de sorte qu'il contienne un message le plus souvent en rapport avec le média et le plus souvent de manière imperceptible.

La stéganographie

la stéganographie est l'art de dissimuler au sein d'un support anodin une information qui bien souvent est sans rapport avec le support. Cette dissimulation se fait de sorte qu'il soit difficile pour un observateur extérieur de se rendre compte qu'il y a eu dissimulation.

La cryptographie

La cryptographie est l'art de rendre indéchiffrable un message et ceci au sus de tout le monde.

La stéganographie : Une histoire ancienne

Exemple de steganographie

Histiée pousse Aristagoras à la rébellion contre Darius le roi Perse. Pour cela, il écrit sur le crâne, d'abord rasé d'un esclave, qu'il envoie à Arisagoras après que les cheveux aient repousés.

Steganographie : Correspondance Sand/Musset 1833

George à Alfred

Je suis très émue de vous dire que j'ai
bien compris l'autre soir que vous aviez
toujours une envie folle de me faire
danser. Je garde le souvenir de votre
baiser et je voudrais bien que ce soit
là une preuve que je puisse être aimée
par vous. Je suis prête à montrer mon
affection toute désintéressée et sans cal-
cul, et si vous voulez me voir aussi
vous dévoiler sans artifice mon âme
toute nue, venez me faire une visite.
Nous causerons en amis, franchement.
Je vous prouverai que je suis la femme
sincère, capable de vous offrir l'affection
la plus profonde comme la plus étroite
en amitié, en un mot la meilleure preuve
que vous puissiez rêver, puisque votre
âme est libre. Pensez que la solitude où j'ha-
bite est bien longue, bien dure et souvent
difficile. Ainsi, en y songeant j'ai l'âme
grosse. Accourez donc vite et venez me la
aire oublier par l'amour où je veux me
mettre.

Réponse de Musset

Quand je mets à vos pieds un éternel hommage
Voulez-vous qu'un instant je change de visage ?
Vous avez capturé les sentiments d'un cœur
Que pour vous adorer forma le Créateur.
Je vous chéris, amour, et ma plume en délire
Couche sur le papier ce que je n'ose dire.
Avec soin, de mes vers lisez les premiers mots
Vous saurez quel remède apporter à mes maux.

Réponse de Sand

Cette insigne faveur que votre cœur réclame
Nuit à ma renommée et répugne mon âme.

Tatouage : Une histoire beaucoup plus récente

Première approches

Les premières approches de tatouage sont plus récentes que pour la stéganographie et peu nombreuses jusqu'aux années 1990.

- 1282 - papier légèrement plus fin à certains endroits (identification ?),
- Présence sur les billets de banque actuels de relief,
- 1954 Premier exemple du monde digital avec insertion d'un message dans une bande sonore à la fréquence 1kHz.

Un exemple moderne pami d'autre

Les images du site web du Musée Hermitage de St. Petersburg sont tatouées pour identifier l'appartenance au musée des images. Un message sur chaque page web indique que ce tatouage est réalisé sur toutes les images. Cette pratique peut donc dissuader la piraterie.

Le tatouage, une science jeune

Year	1992	1993	1994	1995	1996	1997	1998
Publications	2	2	4	13	29	64	103

Table: Number of publications on digital watermarking during the years 1992-1998 according to [PETITCOLAS1999IEEE]

[PETITCOLAS1999IEEE] Fabien A. P. Petitcolas, Ross J. Anderson and Markus G. Kuhn : Information Hiding - A Survey. Proceedings of the IEEE, special issue on protection of multimedia content, 87(7) :1062-1078, July 1999.

Le tatouage, une science jeune

Conférences

- Information Hiding Workshop crée en 1996
- Conférence SPIE "Security and Watermarking of Multimedia Contents" crée en 1999
- ...

Début des années 1990 - Organisations et industries

- The Copy Protection Technical Working Group a testé les systèmes de tatouage pour la protection des DVDs,
- The Secure Digital Music Initiative (SDMI) font du tatouage leur composant central de protection de la musique,
- Projet Européen VIVA et Talisman ont testé les systèmes de tatouage pour l'analyse de diffusion,
- L'ISO étudie l'utilisation du tatouage dans les standard MPEG et JPEG,
- Système de tatouage dans Adobe's Photoshop,
- ...

Tatouage : La motivation première

- facilité de stockage, de copie et de redistribution (disque dur, CD)
- 1993 : Navigateur Web Mosaic et début de l'ère Internet,

Réticence des grands et petits possesseurs et diffuseurs de données numériques envers internet, le DVD , ...

Il faut des solutions pour protéger les ayant droits de ces documents.
Remarque : le cryptage protège tant que le support est crypté mais plus une fois qu'il est en clair.

Les propriétés qu'apporte le tatouage

1. Le tatouage est invisible (cas traité dans ce cours) ; l'esthétique est conservée,
2. Le tatouage est inséparable (cas du tatouage robuste) de son support (à la différence d'un header ou d'un fichier descriptif annexe) ; Un changement de format ne fait pas disparaître le message caché ;
3. Le tatouage subit les mêmes transformations que le support (il est possible d'apprendre sur ces transformations en observant la marque).

Les médias numériques

- texte,
- programme,
- image,
- son,
- vidéo,
- modèle 3D,
- ...

Exemple de tatouage de programmes informatiques Quisquater [99]

Dans un programme codé en assembleur, on peut remplacer certaines séquences d'instructions par d'autres, qui leur sont équivalentes. On peut ainsi modifier la fréquence d'apparition des instructions. Le programme est ensuite compilé. La marque cachée dans le programme est la distribution de fréquences des instructions.

Les grande classes de tatouage

- tatouage robuste,
- tatouage à forte capacité (enrichissement),
- tatouage fragile et semi-fragile,
- tatouage réversible,
- le tatouage sûr.

On s'intéressera principalement au tatouage robuste dans ce cours.

Plan

Les applications possibles

- **contrôle de diffusion** - "broadcast monitoring" : la marque permet d'identifier le support diffusé,

Les applications possibles

- **contrôle de diffusion** - "broadcast monitoring" : la marque permet d'identifier le support diffusé,
  - identification immédiate (à la différence d'une analyse par calcul de signature puis par parcours d'une BD),
  - + pas de problème de droit à l'insertion (dans une "zone" brevetée) ni de perte lorsque l'on change de format (à la différence d'une insertion dans des headers),
  - - le tatouage dégrade le support et nécessite la mise en place de l'insertion et de la détection.

Les applications possibles

- **contrôle de diffusion** - "broadcast monitoring" : la marque permet d'identifier le support diffusé,
- **identification du propriétaire** - "copyright identification" : la marque permet d'identifier l'ayant droit du support,

Les applications possibles

- **contrôle de diffusion** - "broadcast monitoring" : la marque permet d'identifier le support diffusé,
- **identification du propriétaire** - "copyright identification" : la marque permet d'identifier l'ayant droit du support,
  - + bien moins "voyant" que le tatouage visible d'un copyright,
  - + bien plus sûr qu'un tatouage visible de copyright (présent dans un coin ou sur la jaquette pour un CD),
  - - la présence d'une marque n'est pas visuellement identique au fameux symbole © *suivi de la date et du nom de l'ayant droit*, et donc n'a pas actuellement de validité devant une cour de justice.
  - - les systèmes de tatouage ne sont pas exempts d'extraction erronée,
  - - avec un tel système, un utilisateur honnête peut avoir des difficultés à contacter l'ayant droit pour utiliser son œuvre.

Les applications possibles

- **contrôle de diffusion** - "broadcast monitoring" : la marque permet d'identifier le support diffusé,
- **identification du propriétaire** - "copyright identification" : la marque permet d'identifier l'ayant droit du support,
- **preuve de propiété** - "copyright proof" : la marque permet d'identifier un copyright mais pas de prouver que c'est réellement à un ayant droit (un attaquant peut insérer un autre copyright...),

Les applications possibles

- **contrôle de diffusion** - "broadcast monitoring" : la marque permet d'identifier le support diffusé,
- **identification du propriétaire** - "copyright identification" : la marque permet d'identifier l'ayant droit du support,
- **preuve de propiété** - "copyright proof" : la marque permet d'identifier un copyright mais pas de prouver que c'est réellement à un ayant droit (un attaquant peut insérer un autre copyright...),
  - solution chère : utiliser une tierce personne (ex : Office of Copyrights) pour enregistrer le document et le copyright associé. 30$ par document.
  - solution technique : prouver que l'un détient l'original en démontrant qu'un des deux documents est dérivé de l'autre.

Le point de vue Orange (France Telecom) : Gaëtan Le Guelvouit, 11 octobre 2007

use case 1 – marque unique du distributeur

- idée : insérer la marque du distributeur X dans tous les contenus
- les contenus peuvent être marqués à l'avance
- mais à quoi ça sert ?
  - savoir qu'un contenu distribué illégalement provient de X
    → détection d'un problème dans la chaîne
  - savoir qu'un contenu distribué illégalement ***ne*** provient ***pas*** de X
    → se dédouaner vis-à-vis des ayants-droit
- dans ces deux cas, ce n'est pas pas un argument convaincant pour les ayants-droit

Les applications possibles

- **suivi de transaction** - "fingerprinting" : la marque permet d'identifier l'acheteur du support,

Les applications possibles

- **suivi de transaction** - "fingerprinting" : la marque permet d'identifier l'acheteur du support,
  - + bien moins "voyant" que le tatouage visible,
  - + bien plus sûr qu'un tatouage visible,
  - - une structure de traçage qui est complexe à mettre en oeuvre.

Le point de vue Orange (France Telecom) : Gaëtan Le Guelvouit, 11 octobre 2007

use case 2 – une marque par client

- idée : quand un client Bob achète ou loue, son contenu est tatoué avec la marque « Bob »
- aspect répressif : on retrouve un film marqué « Bob » sur un réseau P2P → on peut mettre Bob en prison
- aspect dissuasif
  - hypothèse 1 : le client n'a pas accès au détecteur → il ne sait pas si une attaque sur la marque sera concluante
  - hypothèse 2 : si la fonction de tatouage est déportée chez le client, elle est *suffisamment* sécurisée

mettez-vous à la place de Bob, vous savez que le contenu vous identifie… allez-vous le distribuer ?

Les applications possibles

- **authentification du support** - "authentication" : la présence de la marque permet de savoir si le support est un support non altéré.

Les applications possibles

- **authentification du support** - "authentication" : la présence de la marque permet de savoir si le support est un support non altéré.
  - En crypto on utilise la notion de signature (= résumé crypté du message ; utilisation d'une fonction de "hashing") pour vérifier à la reception l'authenticité du message (comparaison hash reçu et hash calculé)
  - + du tatouage : le message est directement dans le document (pas de risque de perte de la signature)
  - solution tatouage fragile : utilisation de la marque comme authentifiant. Par exemple la signature ("hashing") est calculée sur les bits de poids fort et insérée dans le bit de poids les plus faibles. On peut également être plus précis et calculer des signatures locales au document (blocs) de sorte que la zone dégradée pourra être détectée.
  - tatouage semi-fragile : résistance de la marque à certains traitements comme la compression avec perte.

Les applications possibles

- **contrôle de copie** - "copy control" : la marque indique si l'utilisateur a le droit de copier ou non le document.

Les applications possibles

- **contrôle de copie** - "copy control" : la marque indique si l'utilisateur a le droit de copier ou non le document.

  La solution de défense contre la copie est le cryptage. Trois attaques possibles :

  - force brute : on essaye toutes les clefs (impraticable si la taille des clefs est trop grande)
  - reverse engineering : si le hardware ou le software contient la clef, on peut retrouver celle-ci (exemple : Johansen et al. on fait du "reverse-engineering" d'un lecteur DVD, extrait les clefs de décrypage et ensuite produit un programme nommé DeCSS pour décrypter n'importe quelle vidéo encryptée en CSS (donc les DVDs)).
  - copier un contenu "en clair", décrypté de manière légale (exemple enregistrement par un VCR : Video Cassette Recording).
    Solution de contre-attaque : rendre impossible la copie. Par exemple le système de Macrovision modifie le signal vidéo tel que le VCRs enregistre un signal non regardable alors que le signal sur le téléviseur est parfaitement visualisable. Cette solution est digitale et ne s'applique pas lorsque l'on souhaite enregistrer via un enregistreur de DVD.

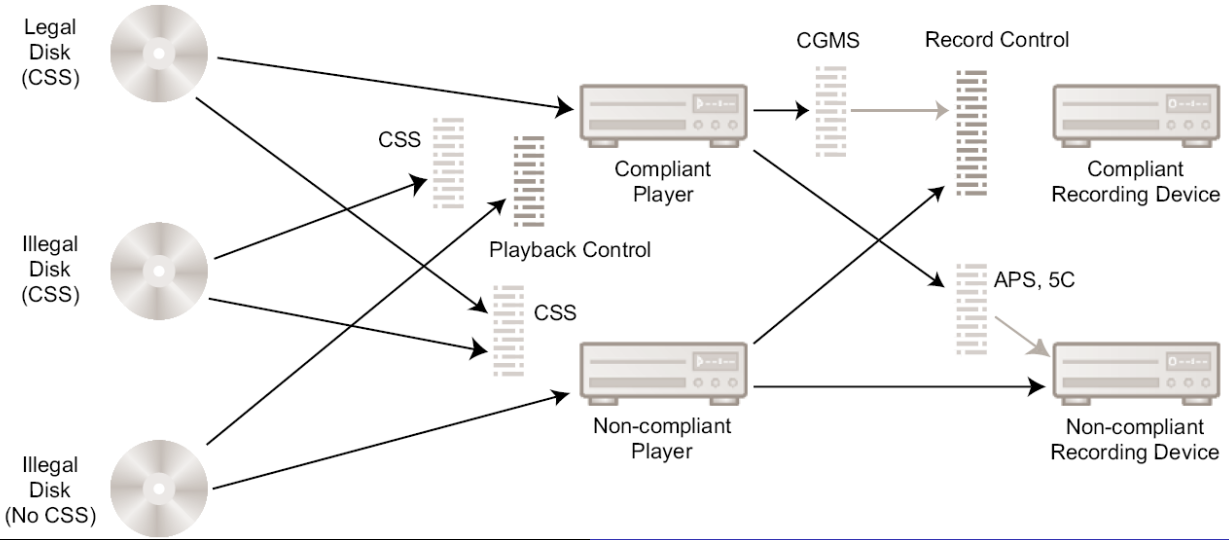
Les applications possibles

- **contrôle de copie** - "copy control" : la marque indique si l'utilisateur a le droit de copier ou non le document.
  - tatouer une info "*never-copy*" sur le document et mettre un détecteur dans tous les appareils de lecture du document.

    Problème : cette technologie restrictive n'est pas vendeuse et de plus elle devrait être adoptée par tous les constructeurs. Il faudrait donc une loi incitative mais il serait difficile de la faire accepter par tous les pays. Finalement, seule une pression économique peut faire bouger les chose (cf. "Majors" qui poussent (poussaient ?) aux DRM et également aux systèmes de protection de la HD).

Les applications possibles

- **contrôle de copie** - "copy control" : la marque indique si l'utilisateur a le droit de copier ou non le document.
  - La solution DVD : faire cohabiter des lecteurs et enregistreurs "compliant-tatouage" et des lecteurs et enregistreurs "non-compliant". Par exemple, si le constructeur souhaite construire un lecteur ou enregistreur de DVD, il doit acheter la licence CSS qui lui impose d'introduire un système de détection de tatouage. L'attaque d'un DVD devient alors beaucoup plus difficile.
  - **playback control :** lorsqu'un lecteur "compliant" voit la marque "never-copy", il vérifie l'authenticité du signal vidéo (par exemple par vérification d'encryptage ou bien par vérification de signature) et si le signal n'est pas authentifié la lecture est stoppée.
  - Avec cette solution, l'acheteur a le choix : - d'acheter un lecteur DVD compliant, acheter des DVD legaux et ne pas lire de DVD piratés ou - acheter un lecteur DVD "non-compliant", lire des DVDs piratés et ne pas lire des DVDs légaux.



Les applications possibles

- **contrôle de périphérique** - : Le périphérique réagit en fonction de la marque (le contrôle de périphérique est une catégorie plus large du contrôle de copies).

Les applications possibles

- **contrôle de périphérique** - : Le périphérique réagit en fonction de la marque (le contrôle de périphérique est une catégorie plus large du contrôle de copies).
  - les décodeurs Dolby FM passent en Dolby s'ils détectent un signal spécifique (= marque) inaudible indiquant une émission FM en Dolby.
  - Interaction d'un jouet avec ce qui passe à la télé (la modulation des intensités des parcours des lignes produit un signal visuel et fréquenciel).
  - ...

Les applications possibles

- **contrôle de périphérique** - : Le périphérique réagit en fonction de la marque (le contrôle de périphérique est une catégorie plus large du contrôle de copies).
- **enrichissement** - "enhancement" : la marque contient une information additionelle comme des codes correcteurs du support, des paramètres d'animation d'un clone...

Les applications possibles : résumé

- **contrôle de diffusion** - "broadcast monitoring",
- **identification du propiétaire** - "copyright identification",
- **preuve de propiété** - "copyright proof",
- **suivi de transaction** - "fingerprinting",
- **authentification du support** - "authentication",
- **contrôle de copie** - "copy control",
- **contrôle de périphérique** - "device control",
- **enrichissement** - "enhancement"

Plan

Contraintes (propriétés) et compromis

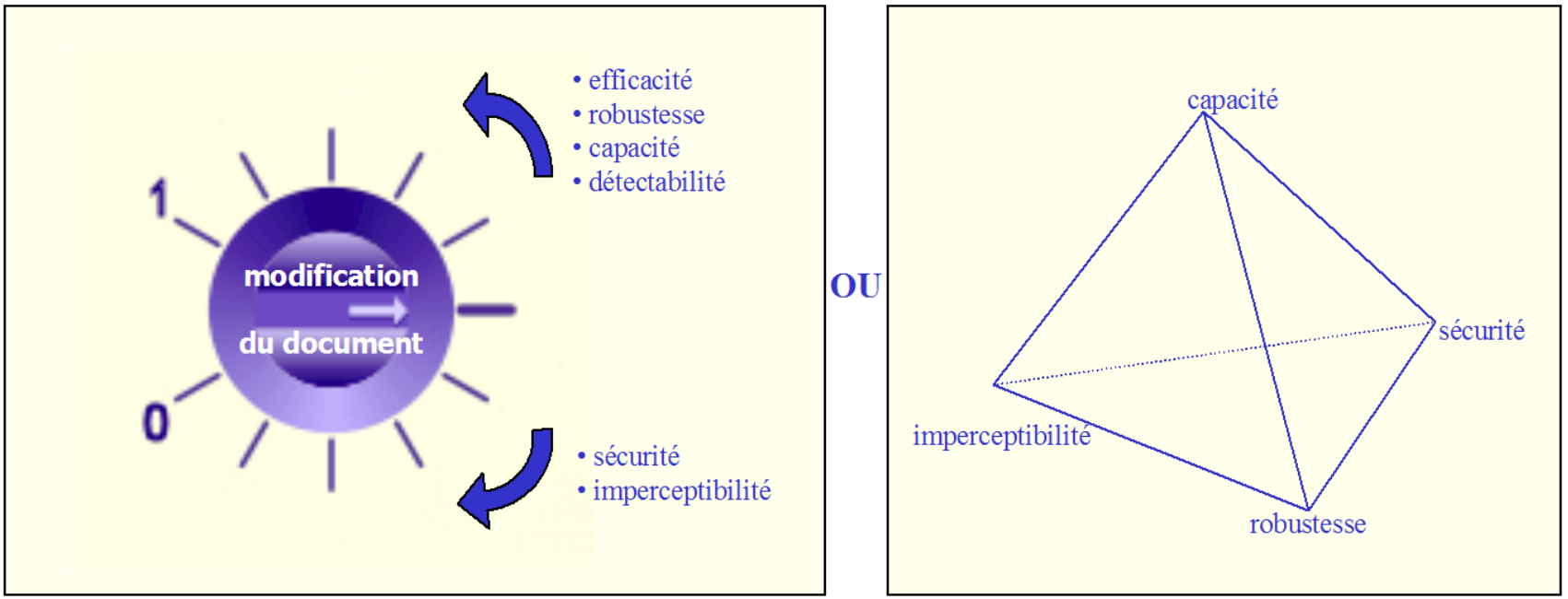


Fig.: **Schémas FAUX** énumérant quelques propriétés et tentant de donner les compromis entre ces quelques propriétés

Propriétés, fonction de l'application

- efficacité (important pour preuve d'appartenance),

Propriétés, fonction de l'application

- efficacité (important pour preuve d'appartenance),
- robustesse (pas besoin pour broadcast monitoring d'être robuste à l'enregistrement VHS)

Propriétés, fonction de l'application

- efficacité (important pour preuve d'appartenance),
- robustesse (pas besoin pour broadcast monitoring d'être robuste à l'enregistrement VHS)
- imperceptibilité (pas le même niveau pour streaming web et HD)

Propriétés, fonction de l'application

- efficacité (important pour preuve d'appartenance),
- robustesse (pas besoin pour broadcast monitoring d'être robuste à l'enregistrement VHS)
- imperceptibilité (pas le même niveau pour streaming web et HD)
- capacité,

Propriétés, fonction de l'application

- efficacité (important pour preuve d'appartenance),
- robustesse (pas besoin pour broadcast monitoring d'être robuste à l'enregistrement VHS)
- imperceptibilité (pas le même niveau pour streaming web et HD)
- capacité, exemple pour l'image :
  - identification : **zero-bits**,
  - protection de copie : 4 à 8 bits pour 5 minutes de vidéo (0.02bits/s),
  - broadcast monitoring : au moins 24 bits par seconde,
  - copyright : à partir de 64 bits,
  - contenu augmenté : plusieurs centaines de bits

Propriétés, fonction de l'application

- efficacité (important pour preuve d'appartenance),
- robustesse (pas besoin pour broadcast monitoring d'être robuste à l'enregistrement VHS)
- imperceptibilité (pas le même niveau pour streaming web et HD)
- capacité,
- sécurité : suppression, insertion non autorisée (forgery) ; (amélioration : pas besoin)

Propriétés, fonction de l'application

- efficacité (important pour preuve d'appartenance),
- robustesse (pas besoin pour broadcast monitoring d'être robuste à l'enregistrement VHS)
- imperceptibilité (pas le même niveau pour streaming web et HD)
- capacité,
- sécurité : suppression, insertion non autorisée (forgery) ; (amélioration : pas besoin)
- détecteur aveugle ou informé (informé possible quand possession de l'original)

Propriétés, fonction de l'application

- efficacité (important pour preuve d'appartenance),
- robustesse (pas besoin pour broadcast monitoring d'être robuste à l'enregistrement VHS)
- imperceptibilité (pas le même niveau pour streaming web et HD)
- capacité,
- sécurité : suppression, insertion non autorisée (forgery) ; (amélioration : pas besoin)
- détecteur aveugle ou informé (informé possible quand possession de l'original)
- insertion multiple (exemple : copy-once et copy-no-more, tatouage de la succession des vendeurs)

Evaluation des dégradations dûes au tatouage

- c_o le signal à tatouer de moyenne μ_{c_o} et d'écart-type σ_{c_o},
- c_w le signal tatoué,
- w la dégradation dût au tatouage ($w = c_w - c_o$) de moyenne μ_w et d'écart-type σ_w,
- Le PSNR après tatouage, pour une image codée sur 8 bits, est $PSNR = -10log_{10}\frac{|w|^2}{255^2}$ (Les valeurs typiques de PSNR pour des images de bonne qualité varient entre 30 et 40 dB),
- Le Watermark to Content Ratio est $WCR = 10log_{10}\frac{\sigma_w}{\sigma_{c_o}}$
- Il existe également d'autres critères d'évaluation perceptuelle (Watson ...),

Evaluation de la bonne transmission via le tatouage

- Le taux d'erreur binaire est $\mathrm{BER} = \frac{nb\ \ bits\ \ erronées}{nb\ \ bits\ \ total\ \ transmis}$
- Le taux d'erreur message est $\mathrm{MER} = \frac{nb\ \ messages\ \ erronées}{nb\ \ messages\ \ total\ \ transmis}$

Plan

1990 - 1998 : Le tatouage de première génération

Page web de Peter Meerwald

	spatial	fréquentiel	multirésolution
additif	Tirkal [93]	Cox [95]	Kun [97]
	Schmid [94]	Piva [97]	Xia [97]
	Bender [95]	Delaigle [98]	Zhu [98]
	Pitas [96]		Barni [99]
	Hartung [98]		
substitutif	Swanson [96]	Zhao [94]	Kun [98]
	Chen [99]		
	Maes [98]		
	Bas [99]		

... - 1998, La toute première technique ... modifier des bits de poids faible

- choix d'emplacements (clé), puis on écrit un message intelligible dans les bits
- choix d'un motif (clé), puis on incruste ce motif (s'apparente au masque jetable)

mais : ces bits sont détruits dès la moindre manipulation (compression, filtrage, bruit).

1990 - 1998, Exemple Patchwork : Bender 95 puis Pitas 96

Soient : A,B : deux ensembles de n pixels (clé = choix), de luminances $\{a_1, ..., a_n\}, \{b_1, ..., b_n\}$.
Constat :

$$S = \frac{1}{n} \sum_{i=1}^{n} (a_i - b_i) \approx 0$$

1990 - 1998, Exemple du Patchwork de Bender 95 puis Pitas 96

Insertion : on modifie les luminances :

$$\begin{aligned} a'_i &= a_i + C, \\ b'_i &= b_i - C. \end{aligned}$$

On a donc à l'insertion et à la détection :

$$S' = \frac{1}{n}\sum_{i=1}^{n}(a'_i - b'_i) = S + 2C \approx 2C$$

L'introduction du biais dans la statistique permet sachant la clé de retrouver la valeur C insérée.

Plan

1998-2005 : Le tatouage robuste et à plus forte capacité : tatouage de deuxième génération ?

Auparavant, on faisait du tatouage **Sans Information Adjacente** lors de l'insertion.
A partir de 1998-1999, on commence à faire du tatouage **avec Information Adjacente** lors de l'insertion.

Le tatouage de deuxième génération : "Watermarking as Communications with Side Information," I. J. Cox, M.L. Miller, A.L. McKellips, Proceedings of the IEEE, 87(7), pp. 1127-1141, (1999). Copyright (c) 1999 by IEEE

2005-... : Et actuellement ?

On fait encore du tatouage **avec Information Adjacente** lors de l'insertion.
On étudie les problèmes de sécurité (sécurité des schémas actuels, technique d'attaque, nouveaux schémas sûrs, définition de classe de sécurité, formulation théorique des fuites d'information).

ECRYPT (European Network of Excellence for Cryptology) est une initiative d'une durée de quatre ans lancée par des experts européens en cryptologie début des activités 1er février 2004). Les activités du réseau ECRYPT sont divisées en cinq " laboratoires " virtuels dont WAVILA (Watermarking and perceptual hashing virtual lab) pour la lutte contre les contrefaçons et l'abus sur les droits d'auteur.

Plan

Vocabulaire

- signal : par exemple une image. Le signal peut donc être représenté par un vecteur 1D ;
- message : c'est le vecteur binaire qui sera tout d'abord transformé en une marque puis inséré ;
- signal hôte, couverture, document : c'est le signal qui va embarquer (contenir) une marque (filigrane) ;
- signal marqué, signal tatoué : c'est le signal qui a été tatoué ; il embarque une marque ;
- espace d'insertion : c'est un ensemble de coefficients issu du signal hôte ;
- émetteur, codeur : c'est l'algorithme de tatouage ;
- détecteur, extracteur : c'est l'algorithme de détection et/ou d'extraction.

Tatouage basé modèle de communication

Le tatouage est une forme de communication.
On souhaite transmettre un message d'un émetteur (tatouage) vers un récepteur (extraction) et ce message transite à travers un canal (**support hôte** : image, son, vidéo...).

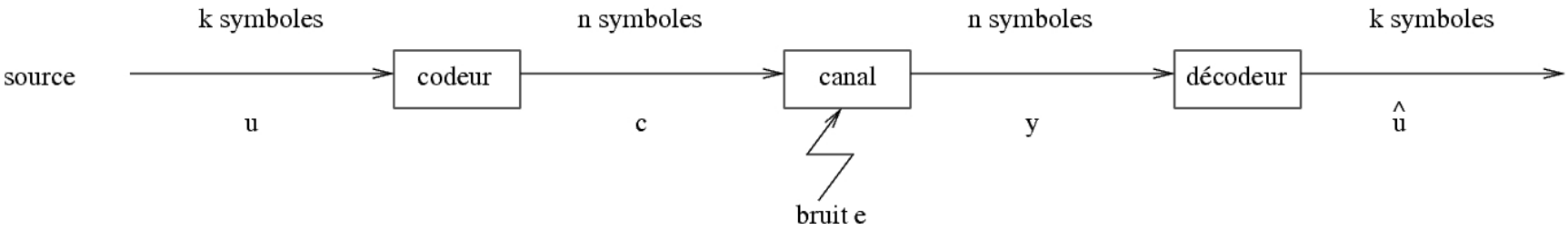
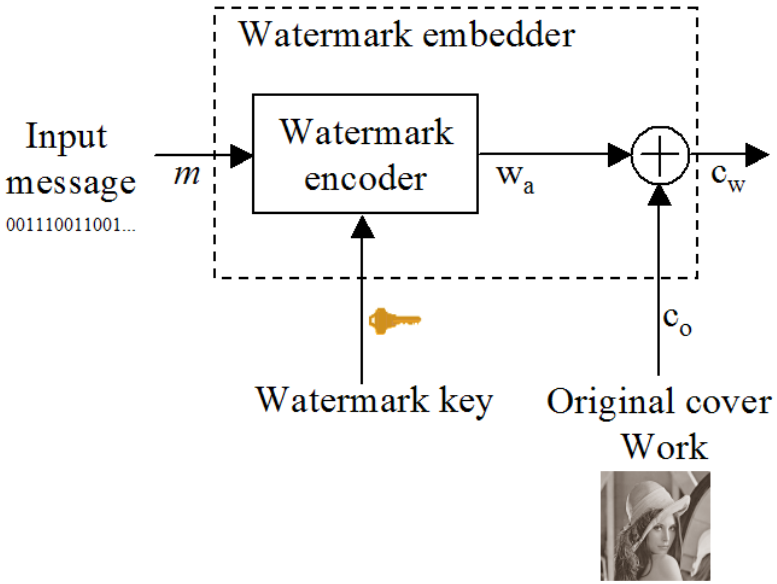


Fig.: Système de communication (message = u, canal = image+attaque, message reçu = $\hat{u}$)

Schéma d'Insertion aveugle

Insertion aveugle :

1. Le message m est transformé ("is mapped") en une marque (pattern) w_a de même dimension que la couverture c_0. Ce "mapping" peut être réalisé en utilisant une clef secrête.
2. La marque w_a est alors ajoutée à la couverture c_0 pour produire le signal tatoué c_w.



Attaque du signal tatoué

Une fois le tatouage réalisé, le média tatoué est **"laché dans la nature"**. Il peut donc subir des dégradations. Ces dégradations sont appelées attaques non-malveillantes (compression, decompression, changement analogique-numérique, filtrage, désynchronisation...) ou attaques malveillantes (un individu essaye volontairement de supprimer la marque). Pour des raisons de simplicité de modélisation, les attaques sont souvent modélisées par l'ajout d'un bruit additif.

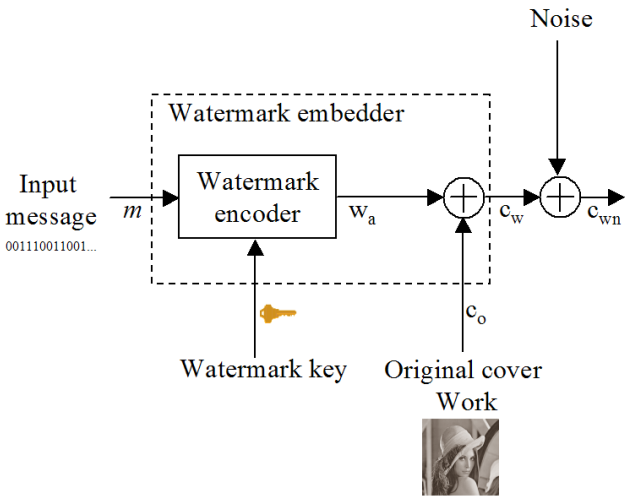


Schéma d'insertion aveugle - détection informée

Détection informée :

1. On retire le signal couverture c_0 du signal marqué attaqué c_{wn} et l'on récupère la marque (filigrane, pattern) bruitée w_n,
2. La marque bruitée w_n est alors décodée grâce à la clef pour obtenir le message m_n.

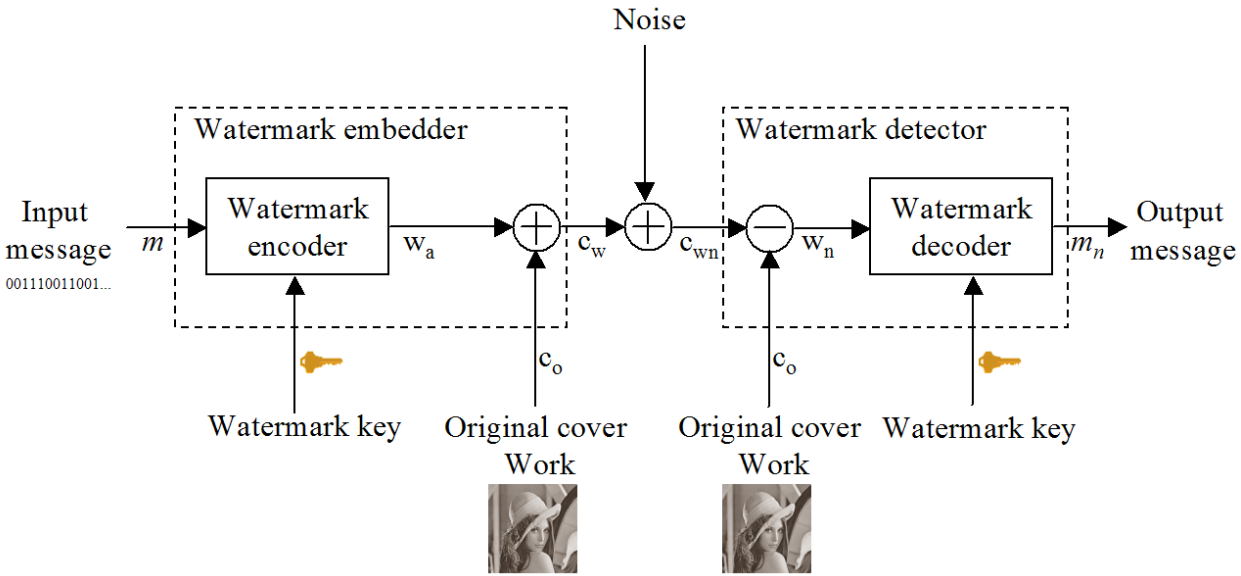
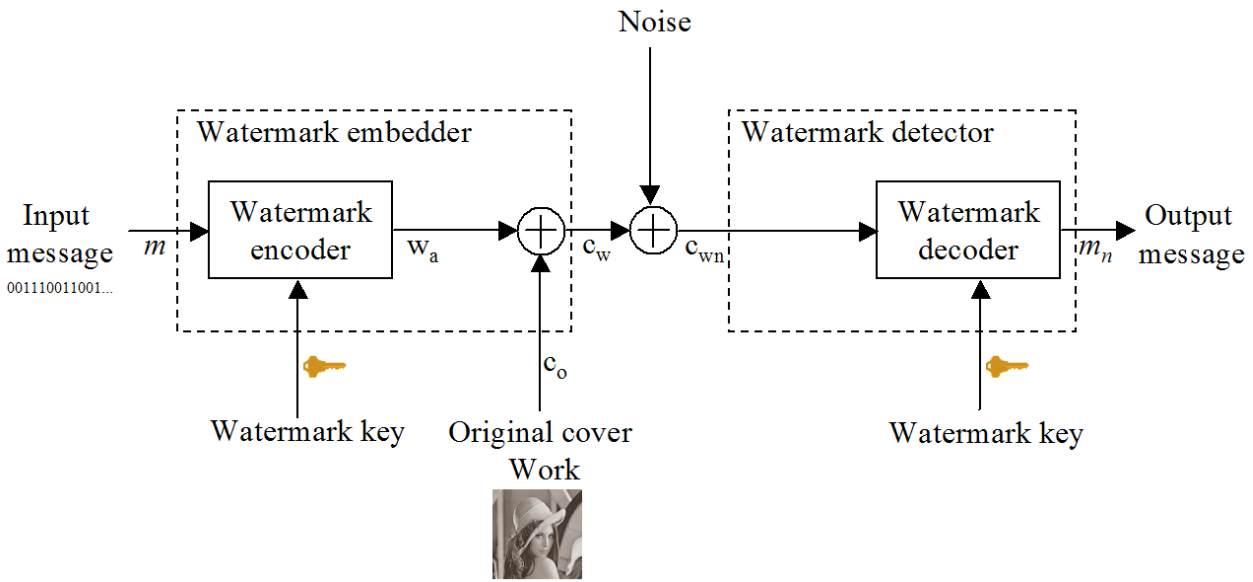


Schéma d'insertion aveugle - détection aveugle

Détection aveugle :

1. La couverture c_0 est inconnue et ne peut donc être retirée. La marque est donc corrompue par la couverture c_0 et par le signal de bruit n.
2. Le signal reçu c_{wn} peut donc être vu comme une version corrompue de la marque w_a



Plan

Insertion aveugle **de 1 bit**

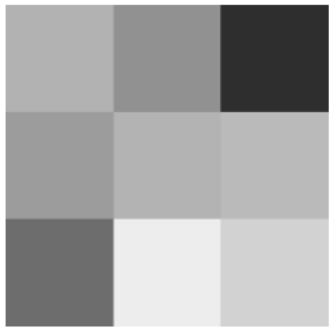


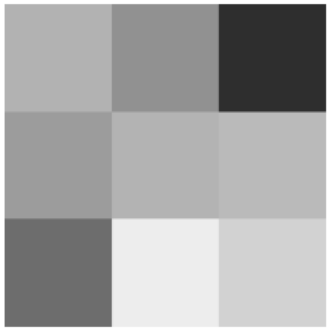
FIG.: Exemple sur cette image monochrome 3×3

$$= (178 \quad 145 \quad 46 \quad 156 \quad 179 \quad 186 \quad 109 \quad 237 \quad 210)^T$$

Insertion aveugle **de 1 bit**

$$
\begin{aligned}
I_w &= I + \alpha \cdot W \\
&= \begin{pmatrix} 178 \\ 145 \\ 46 \\ 156 \\ 179 \\ 186 \\ 109 \\ 237 \\ 210 \end{pmatrix} + \alpha \cdot \begin{pmatrix} -1 \\ -1 \\ -1 \\ -1 \\ 0 \\ 0 \\ 1 \\ -1 \\ 1 \end{pmatrix}
\end{aligned}
$$

Insertion aveugle **de 1 bit**



(a) Image originale

(b) Porteuse insérée

(c) Image tatouée

$$I = (178 \quad 145 \quad 46 \quad 156 \quad 179 \quad 186 \quad 109 \quad 237 \quad 210)^T$$

$$I_w = (175 \quad 142 \quad 43 \quad 153 \quad 179 \quad 186 \quad 112 \quad 234 \quad 213)^T$$

Insertion aveugle **de 1 bit**

- Le message m est un unique bit (0 ou 1).
- Soit w_m générée à partir d'un unique pattern (porteuse) w_r de la même taille que l'image c_o. Ce pattern w_r est généré pseudo-aléatoirement via une clef secrête. On a :

$$w_m = \begin{cases} w_r \ si \ m = 1 \\ -w_r \ si \ m = 0 \end{cases}$$

- La marque est alors définie par $w_a = \alpha w_m$. Le scalaire α permet de contrôler la **force d'insertion** de la marque.
- Finalement, le tatouage est réalisé comme ceci : $c_w = c_o + w_a$.

Insertion aveugle **de 1 bit**

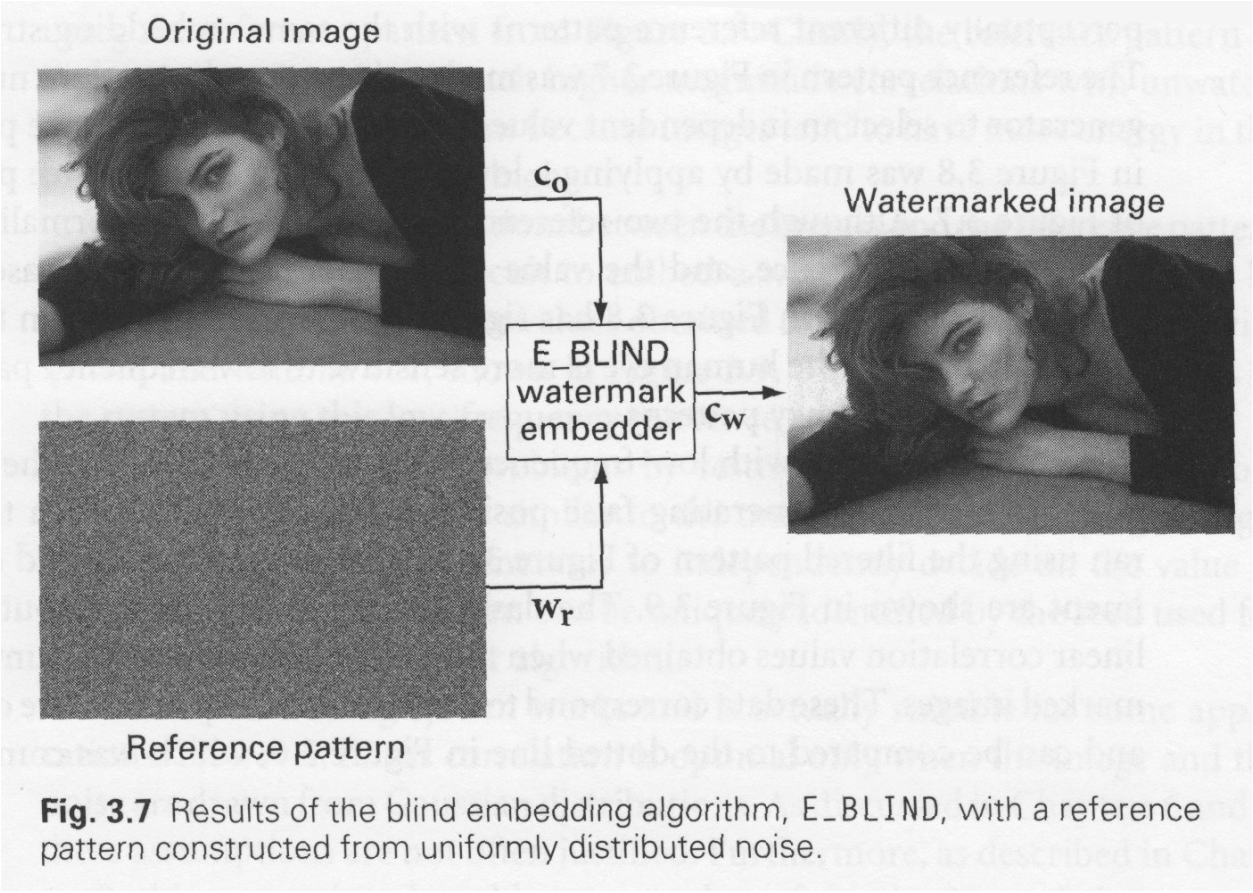


Fig. 3.7 Results of the blind embedding algorithm, E_BLIND, with a reference pattern constructed from uniformly distributed noise.

Fig.: Résultat d'insertion aveugle avec un pattern distribué uniformément

Détection aveugle

Détection aveugle :

Pour détecter la marque, il faut détecter $\pm w_r$ en présence du bruit causé par le signal hôte c_o et le bruit n. La manière optimale pour détecter ce signal en présence de bruit additif Gaussien est de calculer la corrélation linéaire entre l'image reçue c_{wn} et le pattern w_r :

$$z_{lc}(c_{wn}, w_r) = \frac{1}{N} c_{wn}.w_r = \frac{1}{N} \sum_{i=1}^{N} c_{wn}[i].w_r[i]$$

Détection aveugle (justification corrélation linéaire)

Une justification ("intuitive") de l'utilisation de la corrélation linéaire :
Si $c_{wn} = c_o + w_a + n$, alors
$z_{lc}(c_{wn}, w_r) = \frac{1}{N}(c_o.w_r + w_a.w_r + n.w_r)$
Avec l'hypothèse que c_o et n suivent une distribution Gaussienne, $c_o.w_r$ et $n.w_r$ ont de fortes chances d'être de faible amplitude. Au contraire, $w_a.w_r \pm \alpha w_r.w_r$ doit être de forte amplitude. D'où :

$$z_{lc}(c_{wn}, w_r) \approx \alpha w_r.w_r/N \ si \ m = 1$$
$$z_{lc}(c_{wn}, w_r) \approx -\alpha w_r.w_r/N \ si \ m = 0$$

Détection aveugle

Sortie du détecteur :

$$m_n = \begin{cases} 1 & si\ z_{lc}(c_{wn}, w_r) > \tau_{lc} \\ pas\ de\ marque & si\ -\tau_{lc} \leq z_{lc}(c_{wn}, w_r) \leq \tau_{lc} \\ 0 & si\ z_{lc}(c_{wn}, w_r) < -\tau_{lc} \end{cases}$$

Plan

Construction d'un pattern (porteuse) pseudo-aléatoire

```
/*-----------------------------------------------------------------------*
 | MakeRandomPattern -- make a random pattern by drawing pixel values    |
 |                      independently from a Normal distribution and then |
 |                      normalizing to have zero mean and unit variance   |
 |                                                                        |
 | Arguments:                                                             |
 |   seed -- each seed leads to a unique pattern                          |
 |   w -- where to store generated pattern                                |
 |   width -- width of w                                                  |
 |   height -- height of w                                                |
 |                                                                        |
 | Return value:                                                          |
 |   none                                                                 |
 *-----------------------------------------------------------------------*/

void WMTools::MakeRandomPattern( unsigned int seed, double *w, int width, int height )
{
  int i;
  srand(seed); //re-initialisaion de la semance
  for( i = 0; i < width * height; i = i + 1 )
    w[ i ] = RandNormal();
  NormalizePattern( w, width, height );
}
```

Normalisation d'un pattern (porteuse) pseudo-aléatoire

```
/*-------------------------------------------------------------------------*
 | NormalizePattern -- normalize a pattern to have zero mean and unit      |
 |                     standard-deviation                                  |
 | Arguments:                                                              |
 |   w -- pattern to be normalized (changed in place)                      |
 *-------------------------------------------------------------------------*/
void WMTools::NormalizePattern( double *w, int width, int height ) {
  double mean;                          /* mean of pattern */
  double std;                           /* standard deviation of pattern */
  int i;
  const double ESSENTIALLY_ZERO = 10e-10;

  /* subtract out mean */
  mean = 0;
  for( i = 0; i < width * height; i = i + 1 )
    mean = mean + w[ i ];
  mean = mean / (width * height);
  for( i = 0; i < width * height; i = i + 1 )
    w[ i ] = w[ i ] - mean;

  /* normalize standard deviation */
  std = 0;
  for( i = 0; i < width * height; i = i + 1 )
    std = std + w[ i ] * w[ i ];
  std = sqrt( std / (width * height) );
  if( std > ESSENTIALLY_ZERO )
    for( i = 0; i < width * height; i = i + 1 )
      w[ i ] = w[ i ] / std;
```

Quelques mots sur la sécurité des clefs

Principe de Kerckhoffs

La sécurité doit uniquement reposer sur un paramètre inconnu de l'attaquant (la clef secrête $k \in \mathcal{N}$)

A. Kerckhoffs : La Cryptographie Militaire. Journal des Sciences Militaires, vol. 9, pp. 5-38, Jan. 1883.

Note 1 : Seuls les utilisateurs autorisés peuvent réaliser la détection ou le décodage de la marque ; Les utilisateurs autorisés partagent la clef secrète k.

Note 2 : Nous étudierons dans ce cours uniquement le tatouage symétrique (la clef secrète est appelée privée et la clef est la même à l'insertion et à l'extraction/décodage).

Hypothèse de base de la sécurité :

On doit supposer que l'attaquant connait le (ou a accès au) code du schéma de tatouage.

Comment utiliser les clefs ?

1. La clef sert de "germe" (seed) à un Générateur de Nombres Pseudo-Aléatoire (GNPA) (pseudo random generator number : PRNG),
2. Un appel au GNPA produit une séquence uniforme de nombres aléatoires associés à la clef,
3. Ces nombres peuvent alors être utilisés pour produire un secret. La distribution peut également être modifiée.

Remarque :

- Les PRNGs cryptographiquement sûrs sont lents (BBS, ISAAC),
- Ne jamais utiliser srand/rand du C (ni ceux de Matlab),
- Il vaut mieux utiliser le PRNG MT19937 pour le tatouage (que l'on trouve dans les librairies C LIBIT et GSL).

Insertion d'un message m de taille 1 bit

```
/*-------------------------------------------------------------------------*
 | E_BLIND -- embed a watermark by simply adding a message pattern         |
 | Arguments:                                                              |
 |   c -- image to be watermarked (changed in place)                       |
 |   width -- width of img                                                 |
 |   height -- height of img                                               |
 |   m -- one-bit message to embed -> m=1 or m=0                                  |
 |   alpha -- embedding strength                                           |
 |   wr -- reference pattern (width x height array of doubles)             |
 |                                                                         |
 | Return value:                                                           |
 |   none                                                                  |
 *-------------------------------------------------------------------------*/
void WME_BLIND::E_BLIND( unsigned char *c, int width, int height,
                         int m, double alpha, double *wr ) {
  /* Allocate memory for the pattern */
  double *wm = new double [ width*height ];  /* pattern that encodes m */

  /* Encode the message in a pattern */
  WMTools::ModulateOneBit( m, wr, wm, width, height ); //Recopie wr dans wm fois + ou -1

  /* Scale and add pattern to image (with clipping and rounding) */
  WMTools::AddScaledPattern( c, width, height, alpha, wm );

  /* Delete the memory for the pattern */
  delete [] wm;
}
```

Création de la marque w_m par modulation du message m et d'un pattern w_r

```
/*-----------------------------------------------------------------------*
 | ModulateOneBit -- encode a one-bit message by either copying or negating |
 |                   a given reference pattern                           |
 |                                                                       |
 | Arguments:                                                            |
 |   m -- message to be encoded                                          |
 |   wr -- reference pattern                                             |
 |   wm -- where to store resulting message pattern                      |
 |   width -- width of wm                                                |
 |   height -- height of wm                                              |
 |                                                                       |
 | Return value:                                                         |
 |   none                                                                |
 *-----------------------------------------------------------------------*/
void WMTools::ModulateOneBit( int m, double *wr, double *wm, int width, int height )
{
  int i;                                /* index into patterns */

  if( m == 0 )
    for( i = 0; i < width * height; i = i + 1 )
      wm[ i ] = -wr[ i ];
  else
    for( i = 0; i < width * height; i = i + 1 )
      wm[ i ] = wr[ i ];
}
```

Ajout de la marque au signal hôte

```
/*-------------------------------------------------------------------------*
 | AddScaledPattern -- scale and add a pattern to an image with clipping   |
 |                     and rounding                                        |
 |                                                                         |
 | This multiplies w by alpha to obtain the added pattern, and adds        |
 | it to c, clipping and rounding each pixel to an 8-bit integer.          |
 |                                                                         |
 | Arguments:                                                              |
 |   c -- image to which to add pattern (changed in place)                 |
 |   width -- width of image                                               |
 |   height -- height of image                                             |
 |   alpha -- scaling factor                                               |
 |   w -- pattern to scale and add (width times height array of doubles)   |
 |                                                                         |
 | Return value:                                                           |
 |   none                                                                  |
 *-------------------------------------------------------------------------*/

void WMTools::AddScaledPattern( unsigned char *c, int width, int height,
                                double alpha, double *w )
{
  int i;                                /* pixel index */

  for( i = 0; i < width * height; i = i + 1 )
    c[ i ] = ClipRound( (double)c[ i ] + alpha * w[ i ] );
}
```

Détection par corrélation linéaire

```
/*-------------------------------------------------------------------------*
 | D_LC -- detect watermarks using linear correlation                      |
 | Arguments:                                                              |
 |   c -- image                                                            |
 |   width -- width of img                                                 |
 |   height -- height of img                                               |
 |   tlc -- detection threshold                                            |
 |   wr -- reference pattern (width by height array of doubles)            |
 | Return value:                                                           |
 |   decoded message (0 or 1), or NO_WMK if no watermark is found          |
 *-------------------------------------------------------------------------*/
int WMD_LC::D_LC( unsigned char *c, int width, int height, double tlc, double *wr ) {
  double lc;                            /* linear correlation */
  int m;                                /* decoded message (or NO_WMK) */

  /* Find the linear correlation between the image and the reference pattern */
  lc = WMTools::ImgPatInnerProduct( c, wr, width, height ) / (width * height);

  /* Decode the message */
  if( lc > tlc )
    m = 1;
  else if( lc < -tlc )
    m = 0;
  else
    m = NO_WMK;

  return m;
}
```

Rappel : Produit scalaire

```
/*-------------------------------------------------------------------------*
 | ImgPatInnerProduct -- get the inner product of an image and a pattern   |
 |                                                                         |
 | Arguments:                                                              |
 |   c -- image                                                            |
 |   w -- pattern                                                          |
 |   width -- width of both patterns                                       |
 |   height -- height of both patterns                                     |
 |                                                                         |
 | Return value:                                                           |
 |   inner product of c and w                                              |
 *-------------------------------------------------------------------------*/
double WMTools::ImgPatInnerProduct( unsigned char *c, double *w,
                                    int width, int height )
{
  double product;                   /* inner product of c and w */
  int i;                            /* index into patterns */

  product = 0;
  for( i = 0; i < width * height; i = i + 1 )
    product = product + c[ i ] * w[ i ];

  return product;
}
```

Plan

Insertion sur 4000 images avec m=0 et m=1

le pattern (porteuse) est obtenu avec un distribution uniforme normalisé et α est mis à 1.

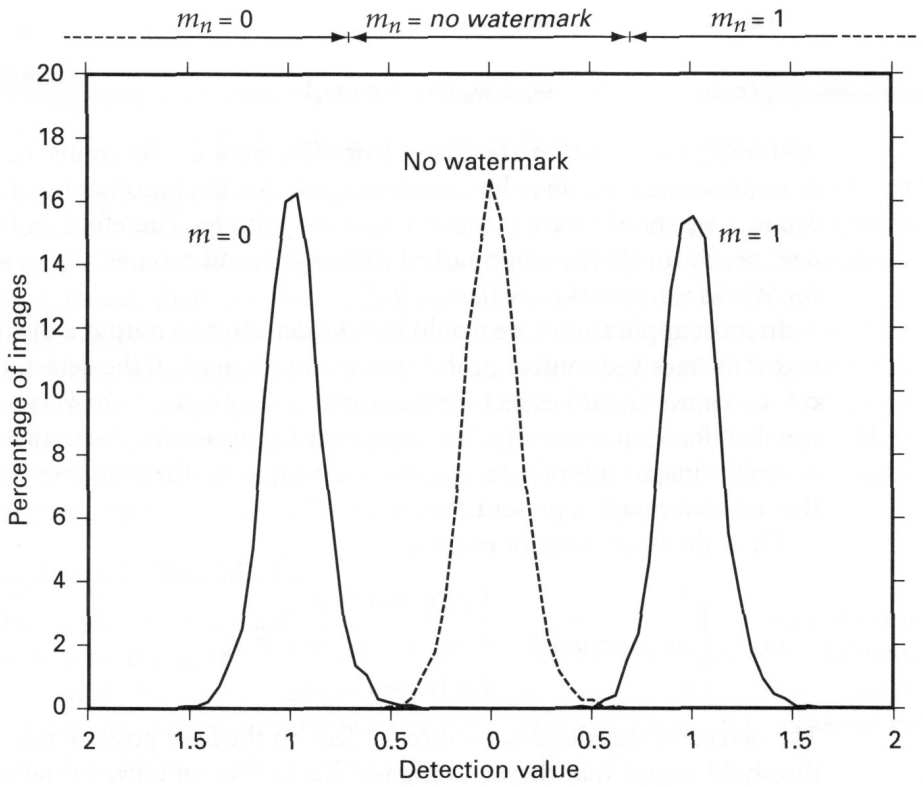


Fig. 3.6 Distributions of linear correlation values resulting from `System 1` (`E_BLIND/D_LC`) with a white noise reference pattern. The left-hand curve is the distribution when the embedded message was 0. The right-hand curve is the distribution when it was 1. The dashed curve is the distribution when no watermark was embedded. The legend at the top of the graph shows how the linear correlation decoder (`D_LC`) maps correlation values into messages.

Insertion sur 4000 images avec m=0 et m=1

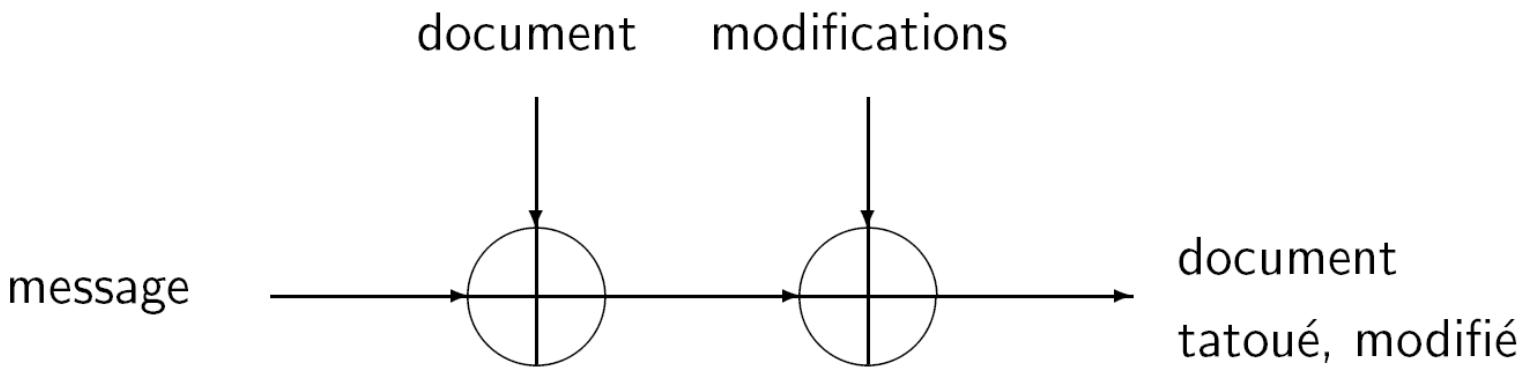
Commentaires :

- Le détecteur peut se tromper pour certaines images tatouées ; On verra ultérieurement l'évaluation de l'efficacité (faux positif) de la robustesse (faux négatif) ;
- Pour le moment, nous n'avons pas pris en compte les attaques ;
- Pour le moment, nous n'avons pas pris en compte l'invisibilité de l'insertion ?
- La technique donnée ici est une technique de première génération (1990-1998) où il n'y a pas de prise en compte de l'information adjacente lors de l'insertion. Les capacités d'insertion (première génération) sont donc très faibles.

Plan

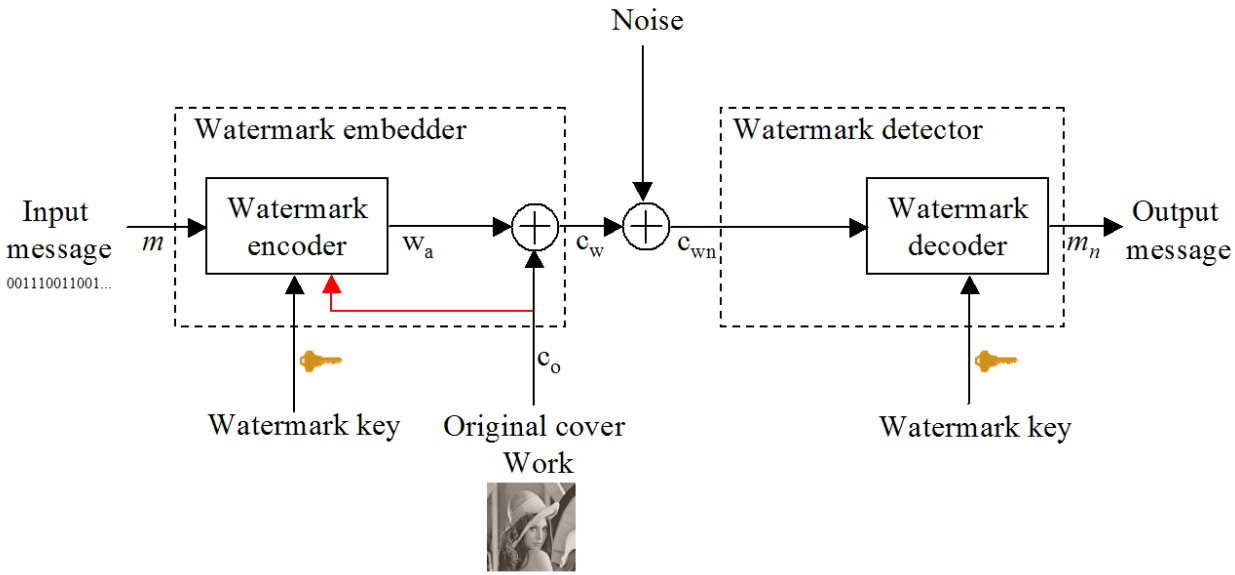
Le modèle de télécomunication utilisé jusqu'en 1998

1998 : on effectue en fait une transmission . . .

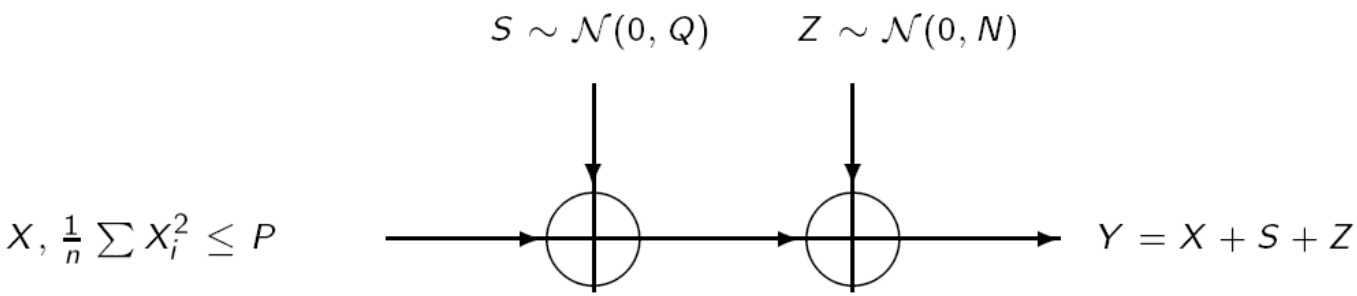


Insertion informée

Puisqu'à l'insertion le signal hôte est connu, il est possible d'exploiter cette connaissance pour améliorer l'éfficacité de l'algorithme. Le codeur examine donc c_o avant de générer la marque w_a. Plusieurs études des communications ont montré que pour certains types de canaux, l'utilisation de l'information de bord permettait de supprimer son interférence.



Writing on a dirty papers - [Costa 83]



Si S est inconnu de l'émetteur et du récepteur, alors

$$C = \frac{1}{2}\log(1 + \frac{P}{N+Q})$$

Si S est connu de l'émetteur, alors

$$C = \frac{1}{2}\log(1 + \frac{P}{N})$$

Construction de codes

Plutôt que de lutter contre le document (bruit), il faut l'exploiter au maximum.

La démonstration de Costa donne une stratégie de construction, qui a amené à une nouvelle génération d'algorithmes de tatouage : Eggers [00 :SCS,QIM], Chen [01 :QIM], Miller [02], Furon [02], Le Guelvouit [03][1], Eggers [03], Miller [04] .