UNIVERSITÉ MONTPELLIER 2
SCIENCES ET TECHNIQUES

Introduction au tatouage numérique :

Quelques points d'entrée
Usages et applications
Propriétés d'un schéma de tatouage
Evaluation d'un schéma de tatouage
Quelques attaques possibles

Tatouage numérique sans infirmations (1 er génération) :

Grandes classes du tatouage
Schéma d'insertion et détection
Exemples de schémas sans information
Quelques mots sur le tatouage avec information

CONTACT :

Email : kouider@lirmm.fr
http://www2.lirmm.fr/~kouider

Dissimulation de données

Tatouage de documents numériques
(Cours 1)

UNIVERSITÉ MONTPELLIER 2
SCIENCES ET TECHNIQUES

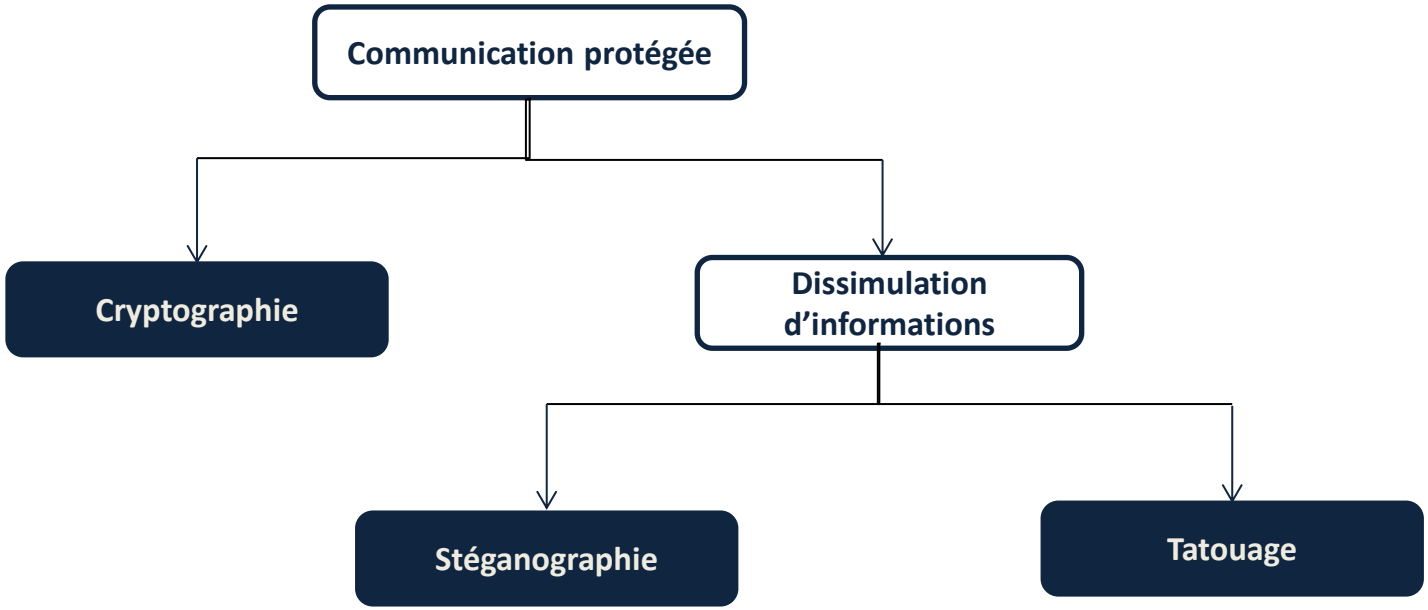
CONTACT :

Email : kouider@lirmm.fr
http://www2.lirmm.fr/~kouider

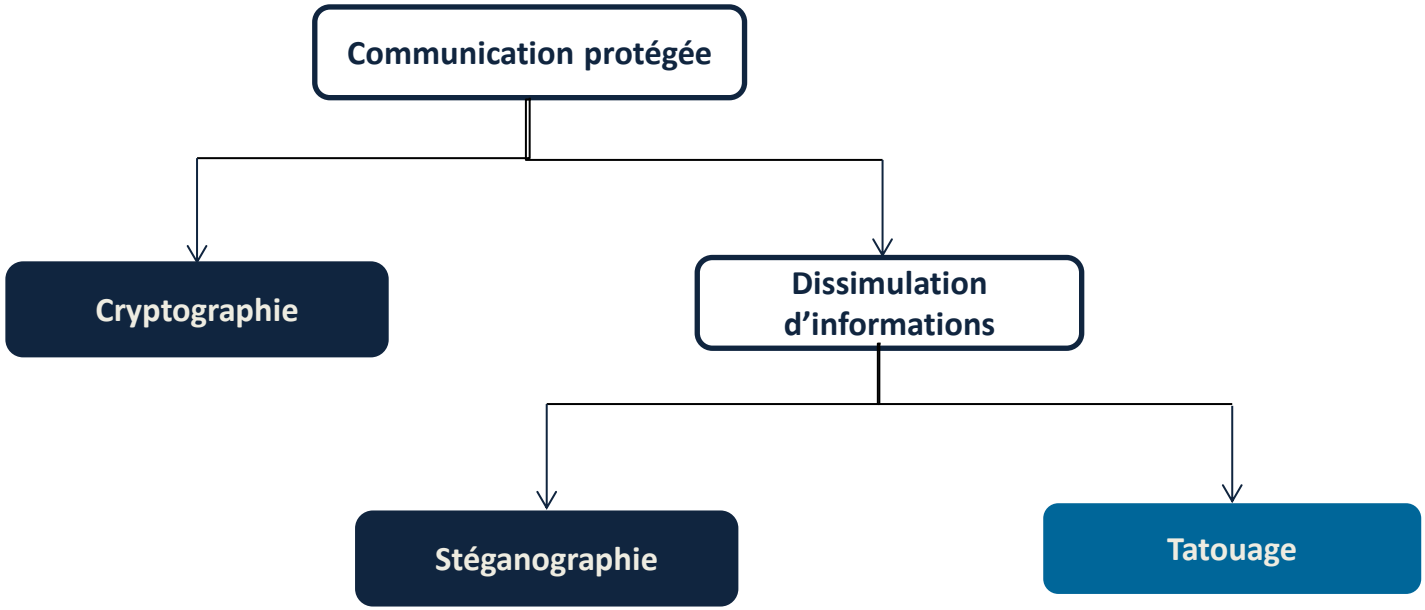
Tatouage de documents numériques

I. Introduction au tatouage numérique

Techniques de protection des données secrètes



Techniques de protection des données secrètes



Définition du tatouage (Watermarking)

Le tatouage numérique

Le tatouage est l'art d'altérer un média (un texte, 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** et **robuste**.

Tatouage visible

Tatouage invisible

Tatouage vs Stéganographie vs Cryptographie

Le tatouage numérique

Le tatouage est l'art d'altérer un média (un texte, 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** et **robuste**.

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 hôte**. Cette dissimulation se fait de sorte que la présence même du message soit insoupçonnée. Autrement dit, la dissimulation doit être **indétectable visuellement et statistiquement**.

La Cryptographie

La cryptographie est l'art de rendre **indéchiffrable** un message et ceci au sus de toute personne tierce.

Le tatouage dans l'Histoire

Premières approches

Contrairement à la stéganographie, les premières approches de tatouage sont plus récentes et peu nombreuses jusqu'aux années 1990 :

- 1282 – papier légèrement plus fin à certain endroits pour l'identification.
- Présence sur les billets de banque actuels de filigrane.
- 1954 Premier exemple du monde digital avec insertion d'un message dans une bande sonore à la fréquence 1kHz.

Un exemple parmi d'autre

Actuellement, les images du site web du Musée Hermitage de St. Petersburg sont tatouées pour identifier l'appartenance des images au musée. 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 numérique : 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 numérique : une science jeune

Conférences

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

Le tatouage numérique dans le domaine industrielle

Avènement du tatouage numérique dans le domaine industrielle (Début des années 90) :

- **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 principalement pour la protection de la musique,
- L'ISO étudie l'utilisation du tatouage dans les standard MPEG et JPEG,
- La fonction de tatouage **Digimarc** dans Adobe's Photoshop,
- …

La motivation première du tatouage numérique

- 1993 : Navigateur Web **Mosaic** et début de l'ère **Internet**,
- Facilité de stockage, de copie et de redistribution (**disque dur**, **CD**, **DVD**,…),
- Réticence des grands et petits auteurs, possesseurs et diffuseurs de données numériques envers Internet, CD, DVD, et autres ...

Il faut des solutions pour protéger les ayant droits de ces documents

Note : Contrairement au tatouage le cryptage protège tant que le support est crypté mais plus une fois qu'il est en clair (décrypté).

Intérêt du tatouage numérique

- Le tatouage est invisible (cas du tatouage invisible traité dans ce cours); l'esthétique est conservée,
- 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,
- Le tatouage subit les mêmes transformations que le support (il est possible d'apprendre sur ces transformations en observant la marque).

Les médias numériques

- Textes,
- Programme Informatique,
- Images numériques.
- Programme Informatique,
- Vidéos,
- Modèle 3D,
- …

Différents médias numériques qui peuvent possiblement être tatoués

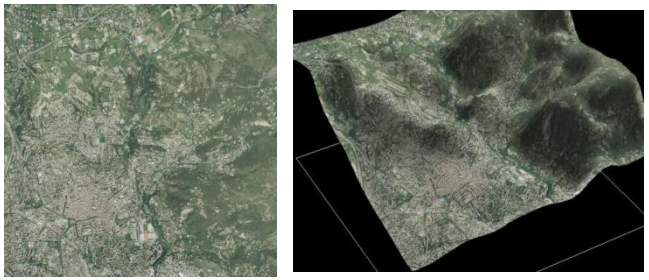
Quelques exemples de tatouage

❑ Tatouage d'un programme informatique :

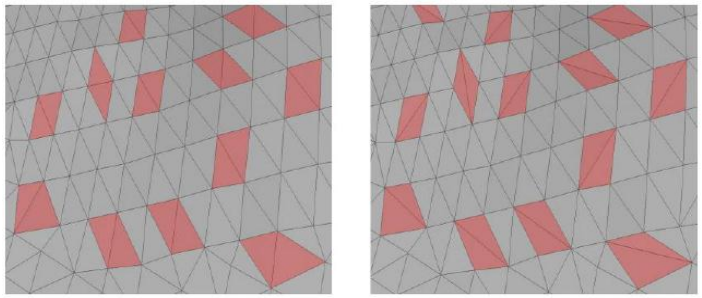
Dans un programme codé en assembleur, on peut remplacer certaines séquences d'instructions par d'autres, qui leur sont équivalentes. On peut ainsi modifier la fréquence d'apparition des instructions. Le programme est ensuite compilé. La marque cachée dans le programme est la distribution de fréquences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

❑ Tatouage d'un modèle 3D :

Pour un modèle 3D on peut insérer a marque du tatouage soit dans la texture du modèle ou bien dans le maillage 3D.



Insertion de marque dans une texture 3D



Maillage 3D tatoué

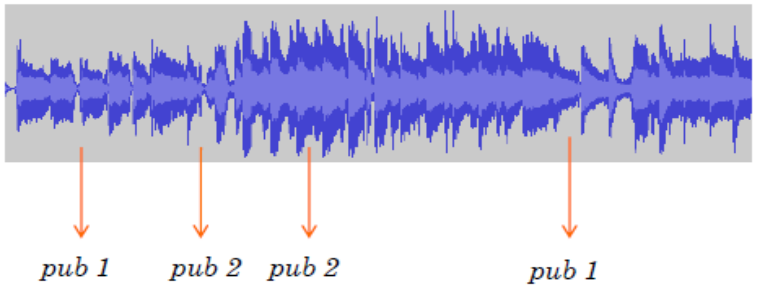
Les applications possibles du tatouage numérique

❑ Contrôle/Surveillance/Analyse de diffusion :

La marque permet d'identifier le support diffusé,

+ Identification immédiate (à la différence d'une analyse par calcul 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 (`a la différence d'une insertion dans des headers),
- Le tatouage dégrade le support et nécessite la mise en place d'un protocole d'insertion et d'extraction

Nombre de pub = 2



Les applications possibles du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification) :**

  La marque permet d'identifier l'ayant droit du support,

FIGURE 2.1
The often-used Lena image in image processing research is a cropped version of a 1972 *Playboy* centerfold. This portion of the original, lost to cropping, identifies the copyright owner.

Les applications possibles du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification) :**

  La marque permet d'identifier l'ayant droit du support,

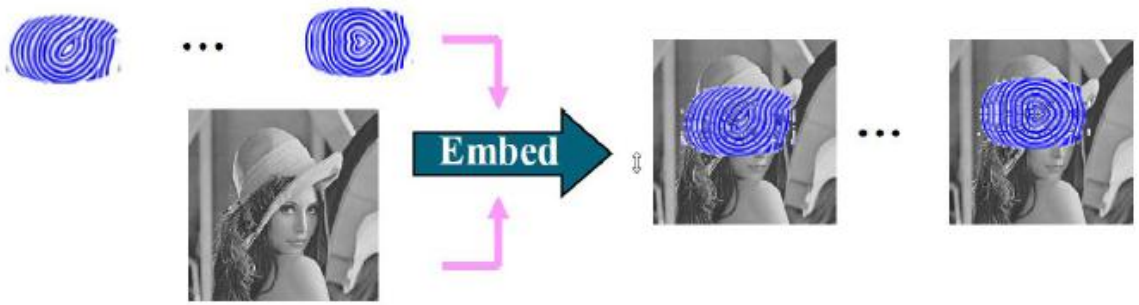
  + Bien moins « voyant» que le tatouage visible d'un copyright,
  + Bien moins sûr qu'un tatouage visible de copyright (présent dans un coin ou sur la jaquette pour un CD),
  - La présence d'une marque n'est pas visuellement identique au fameux symbole © suivi de la date et du nom de l'ayant droit, et 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 difficulté à contacter l'ayant droit pour utiliser son œuvre.

Les applications possibles du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification),**
- **Traçage de traîtres (active fingerprinting) :**

  La marque permet d'identifier l'acheteur du support,

Le vendeur vend à l'acheteur une image qui contient une information identifiant l'acheteur



Les applications possibles du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification) ,**
- **Traçage de traîtres (active 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 œuvre.

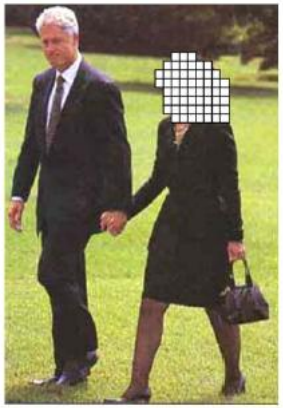
Les applications possibles du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification) ,**
- **Traçage de traîtres (active fingerprinting),**
- **Contrôle d'intégrité (authentication):**

  La présence de la marque permet de savoir si le support est un support non altère,

Image originale (protégée)

Image modifiée



Détection des régions modifiées

Les applications possibles du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification) ,**
- **Traçage de traîtres (active fingerprinting),**
- **Contrôle d'intégrité (authentication) :**

  La présence de la marque permet de savoir si le support est un support non altère,

  En cryptographie on utilise la notion de signature (= utilisation d'une fonction de *hashing*) pour vérifier `a la réception 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),

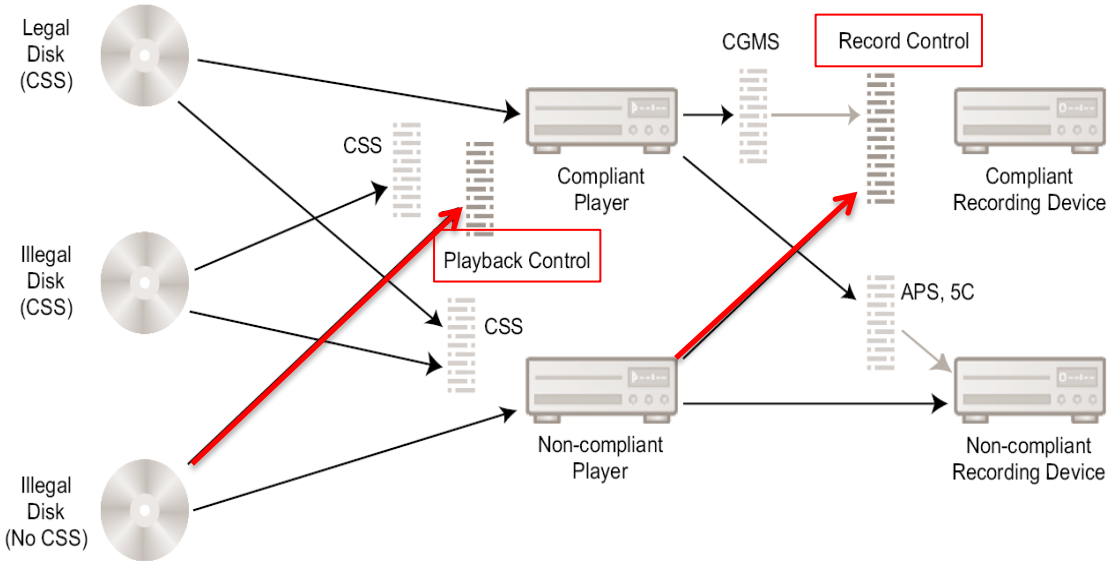
  Solution tatouage fragile : utilisation de la marque comme authentifiant.

  Tatouage semi-fragile : résistance de la marque à certains traitements comme la compression avec perte.

Les applications possibles du tatouage numérique

- Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),
- Identification du propriétaire (copyright identification) ,
- Traçage de traîtres (active fingerprinting),
- Contrôle d'intégrité (authentication),
- Contrôle de copie (copy control) :

  La marque indique si l'utilisateur a le droit de copier ou non le document.



Les applications possibles du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification) ,**
- **Traçage de traîtres (active fingerprinting),**
- **Contrôle d'intégrité (authentication),**
- **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**.

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.

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 du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification) ,**
- **Traçage de traîtres (active fingerprinting),**
- **Contrôle d'intégrité (authentication),**
- **Contrôle de copie (copy control),**
- **Contrôle de périphérique (device control) :**

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

Interaction télévision – jouet robot

Les applications possibles du tatouage numérique

- **Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),**
- **Identification du propriétaire (copyright identification) ,**
- **Traçage de traîtres (active fingerprinting),**
- **Contrôle d'intégrité (authentication),**
- **Contrôle de copie (copy control),**
- **Contrôle de périphérique (device control) ,**
- **Enrichissement (enchancement) :**

  La marque contient une information additionnelle comme des codes correcteurs du support, des paramètres d'animation d'un clone...

Les applications possibles : résumé

- Contrôle/Surveillance/Analyse de diffusion (broadcast monitoring),
- Identification du propriétaire (copyright identification) ,
- Traçage de traîtres (active fingerprinting),
- Contrôle d'intégrité (authentication),
- Contrôle de copie (copy control),
- Contrôle de périphérique (device control) ,
- Enrichissement (enchancement).

Propriétés et caractéristiques d'un tatouage

- **Imperceptibilité (pas le même niveau pour le streaming web et HD),**

Propriétés et caractéristiques d'un tatouage

- Imperceptibilité (pas le même niveau pour le streaming web et HD),
- Robustesse (dépend de l'application),

Propriétés et caractéristiques d'un tatouage

- Imperceptibilité (pas le même niveau pour le streaming web et HD),
- Robustesse (dépend de l'application),
- Capacité (importante pour l'enrichissement de contenu),

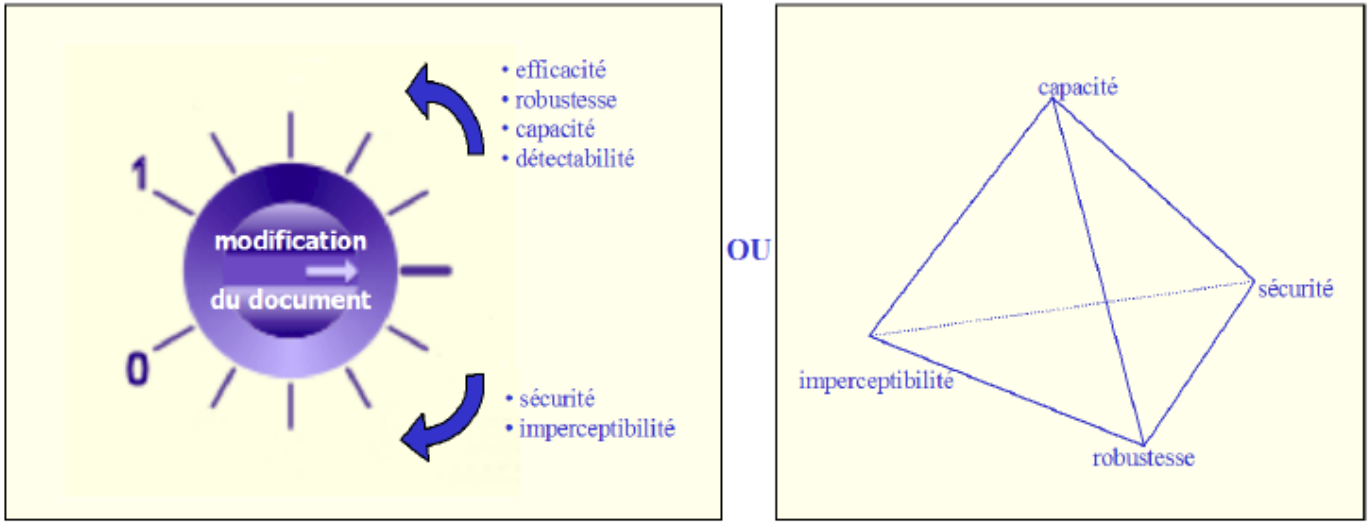
Propriétés et caractéristiques d'un tatouage

- Imperceptibilité (pas le même niveau pour le streaming web et HD),
- Robustesse (dépend de l'application),
- Capacité (importante pour l'enrichissement de contenu),
- Sécurité : le principe de Kerckhoff (la sécurité repose uniquement sur la clé secrète),

Propriétés et caractéristiques d'un tatouage

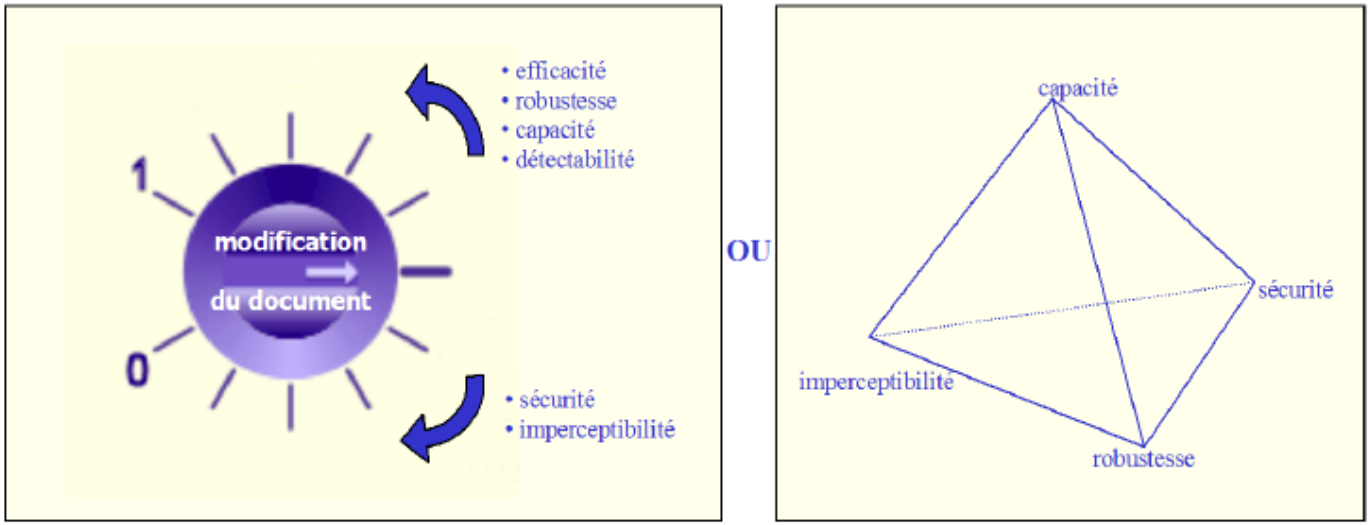
- Imperceptibilité (pas le même niveau pour le streaming web et HD),
- Robustesse (dépend de l'application),
- Capacité (importante pour l'enrichissement de contenu),
- Sécurité : le principe de Kerckhoff (la sécurité repose uniquement sur la clé secrète),
- Complexité (importante pour les applications en temps réel).

Contraintes et compromis



Schémas FAUX énumérant quelques propriétés et tentant de donner les compromis entre ces quelques propriétés

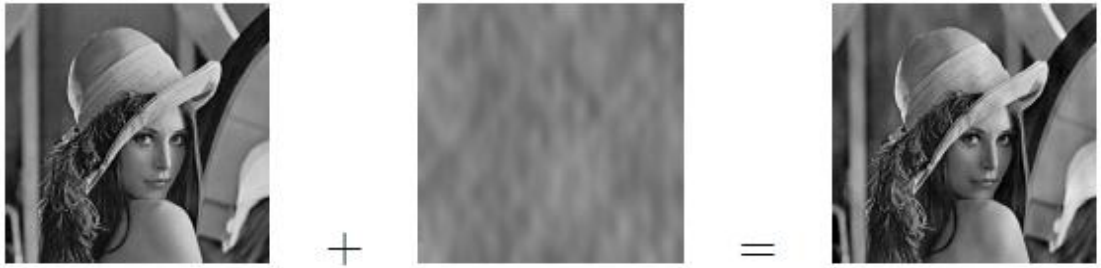
Contraintes et compromis



Schémas FAUX énumérant quelques propriétés et tentant de donner les compromis entre ces quelques propriétés

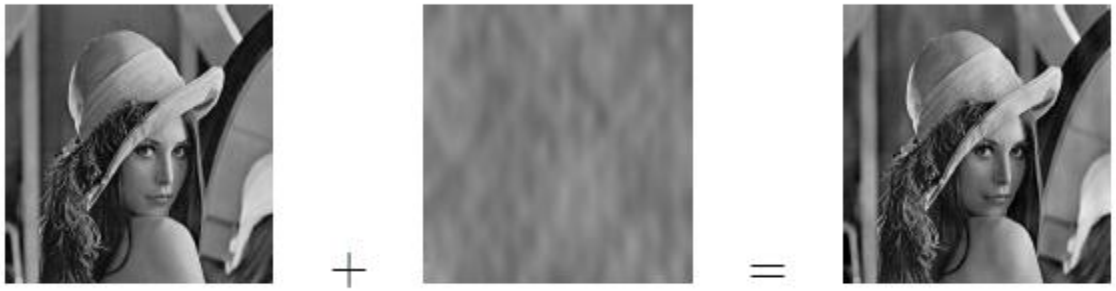
En résumé, lors la conception d'un schéma de tatouage, il faut prendre en compte l'application visée, les éventuelles contraintes de sécurité et la nature du document hôte

Evaluation des dégradation dues au tatouage



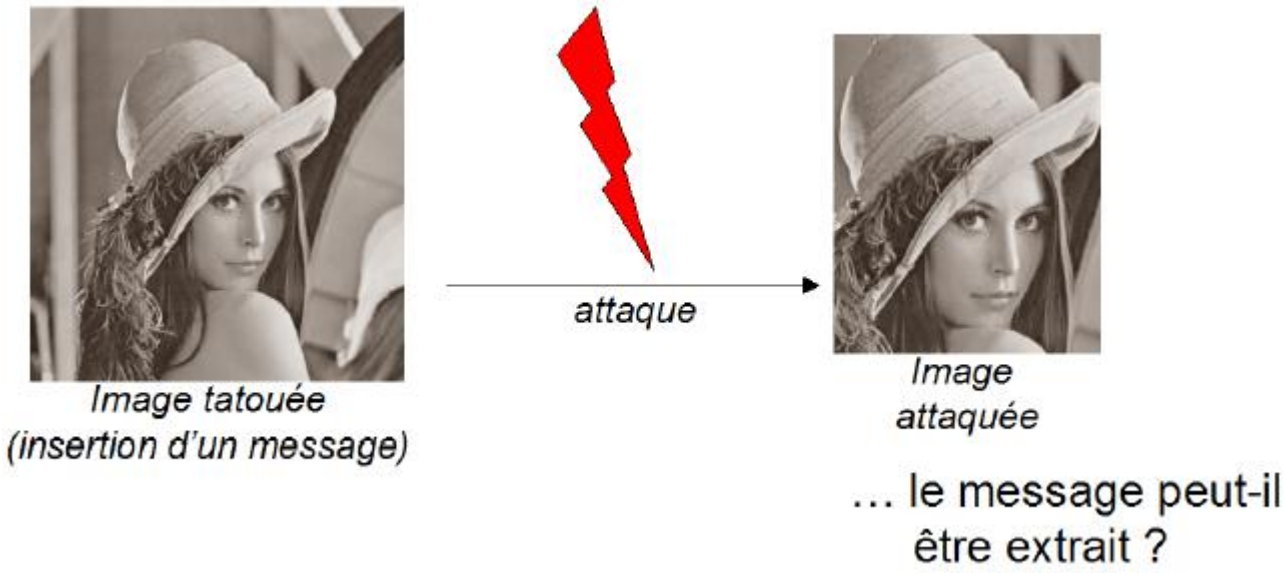
- c_o le signal à tatouer de moyenne μ_{c_o} et d'écart-type σ_{c_o},
- c_w le signal tatoué,
- w la dégradation dût au tatouage ($w = c_w - c_o$) de moyenne μ_w et d'écart-type σ_w,
- Le PSNR après tatouage, pour une image codée sur 8 bits, est $PSNR = -10log_{10}\frac{|w|^2}{255^2}$ (Les valeurs typiques 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 de données en tatouage



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

Attaques possibles sur un tatouage



Attaques possibles sur un tatouage

Attaques sur la robustesse (peuvent être involontaires)	Attaques sur la robustesse (attaques malicieuses)
- **Attaques d'effacement :** bruit, compression avec perte, rehaussement de contraste, lissage, les transformations valumétriques, filtrage, débruitage, etc… - **Attaques désynchronisantes :** rotation, translation, changement, d'échelle, decoupage, etc…	Estimation de la clé secrète et des paramètres secrets : **Cadres d'attaques :** - Le pirate observe uniquement des contenus tatoués, - Le pirate observe des paires de contenus originaux et tatoués, - Le pirate observe des paires de marques et de contenus tatouées associés, - Le pirate observe uniquement des contenus tatoués, mais il sait que le marque caché est toujours la même. **Quelques attaques :** - Attaque cryptographique (attaque sur la clé), - Attaque de protocole (insertion d'une ou plusieurs autres marques), - Attaque de sensibilité (par apprentissage).

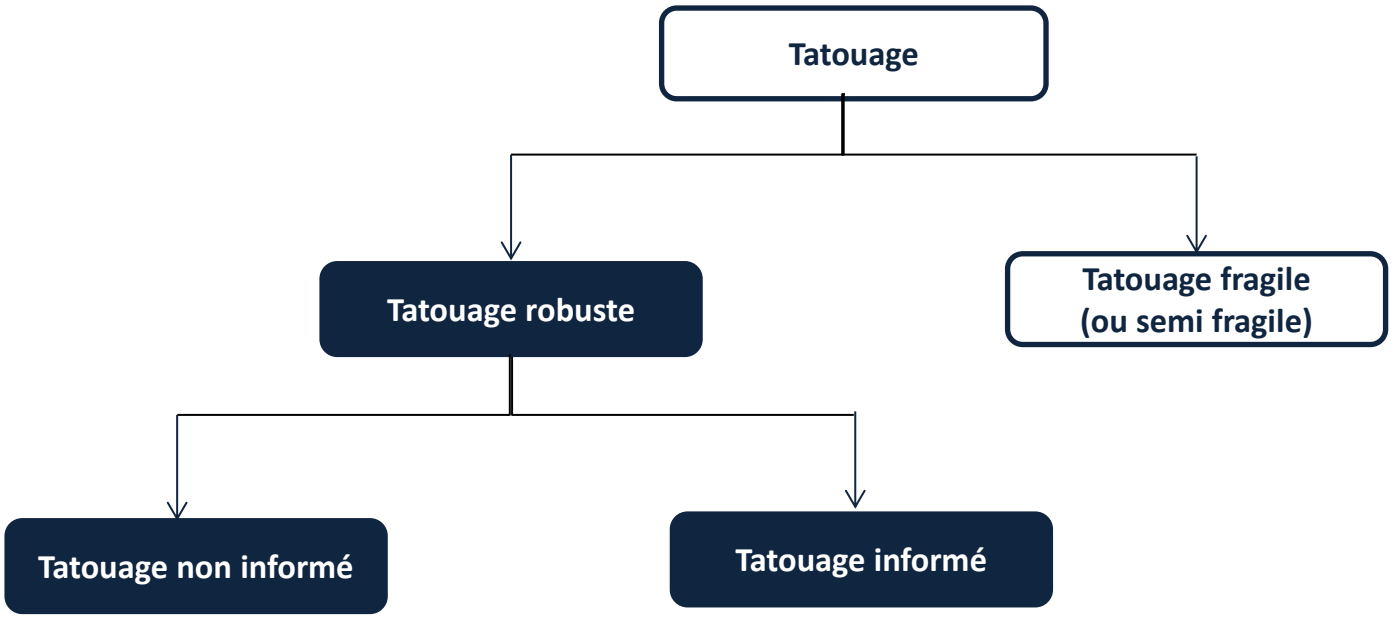
CONTACT :

Email : kouider@lirmm.fr
http://www2.lirmm.fr/~kouider

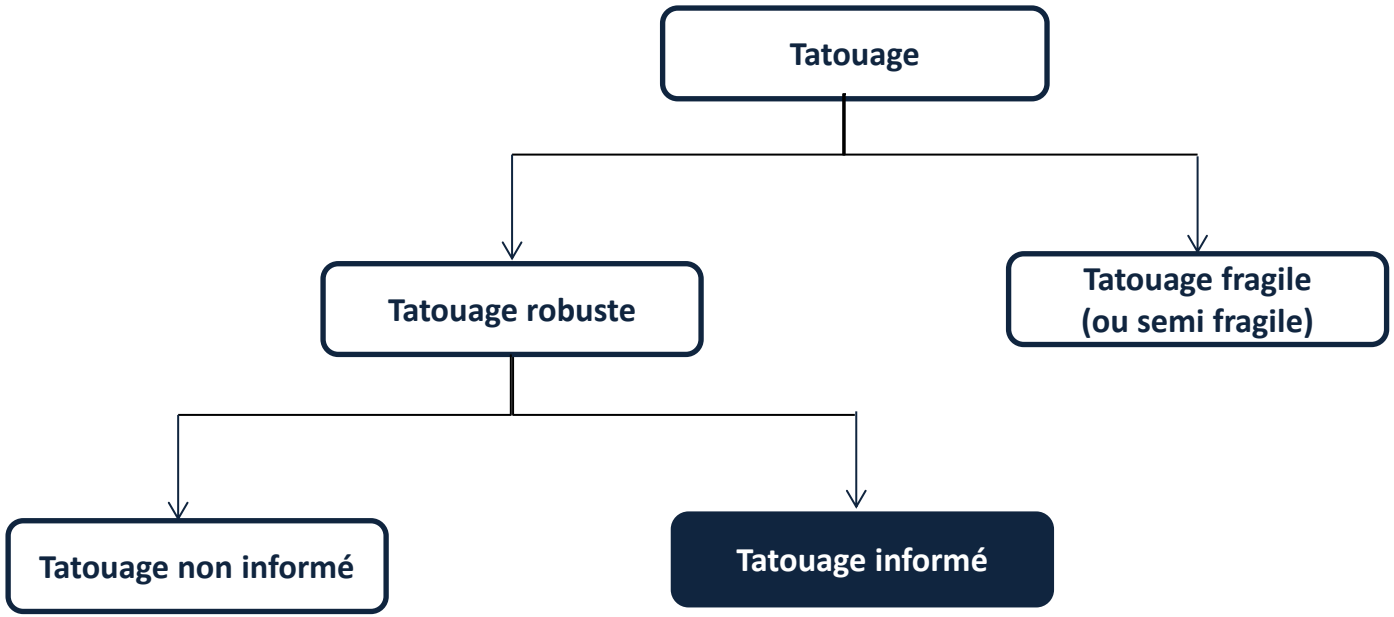
Tatouage de documents numériques

II. Tatouage numérique sans infirmations (1 er génération)

Les grandes classes du tatouage



Les grandes classes du tatouage



Le tatouage du première génération (1990 - 1998)

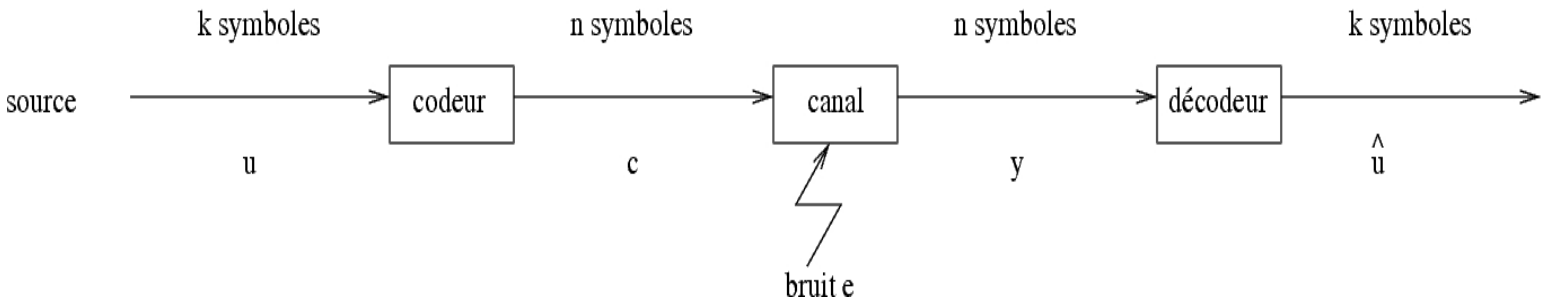
	spatial	fréquentiel	multirésolution
additif	Tirkal [93]	Cox [95]	Kun [97]
	Schmid [94]	Piva [97]	Xia [97]
	Bender [95]	Delaigle [98]	Zhu [98]
	Pitas [96]		Barni [99]
	Hartung [98]		
substitutif	Swanson [96]	Zhao [94]	Kun [98]
	Chen [99]		
	Maes [98]		
	Bas [99]		

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...).



Système de communication (message = u, canal = image+attaque, message reçu = û) message reçu = û)

Schéma d'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.

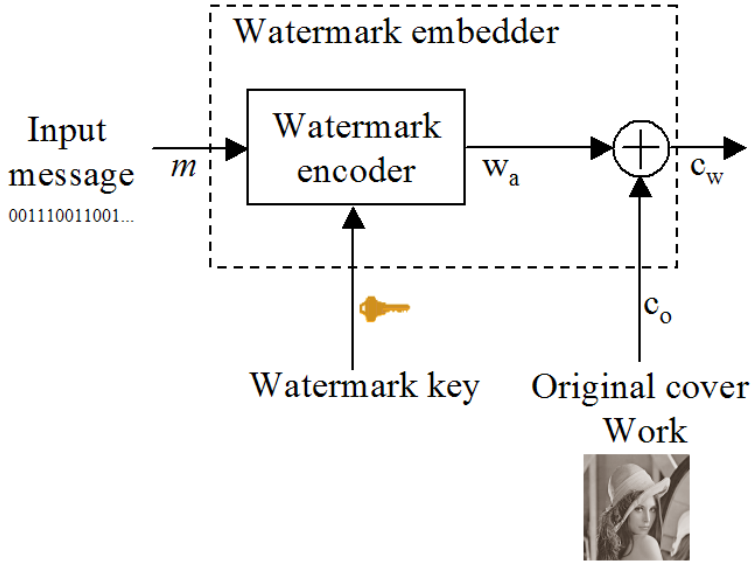


Schéma de détection informé

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.

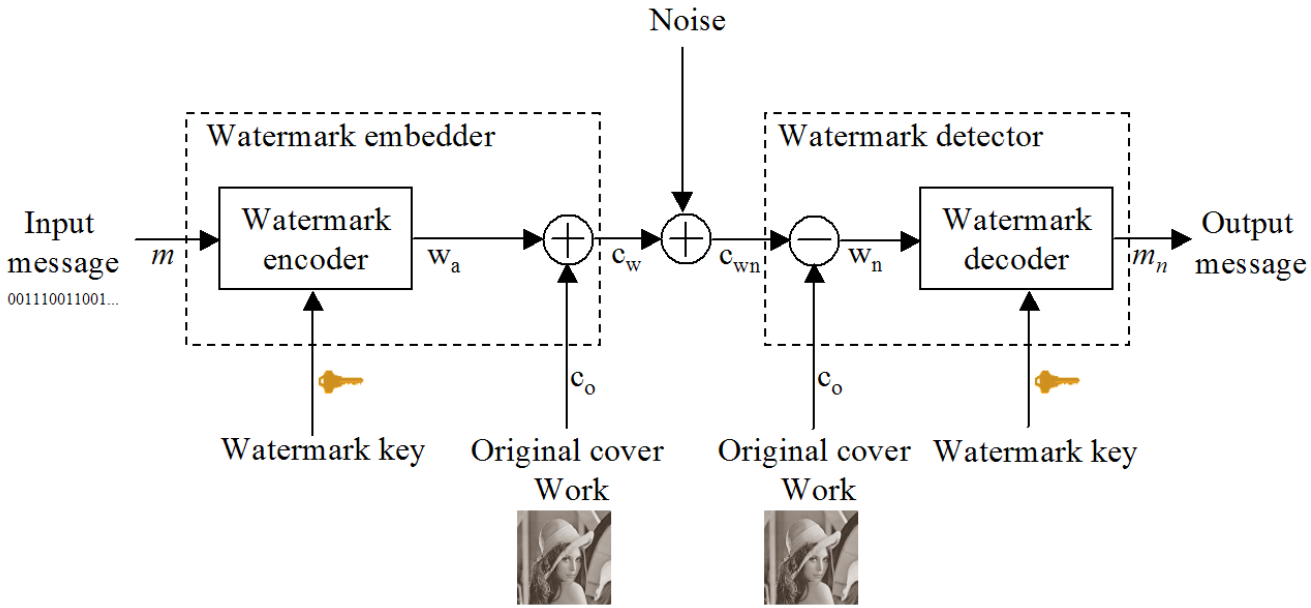
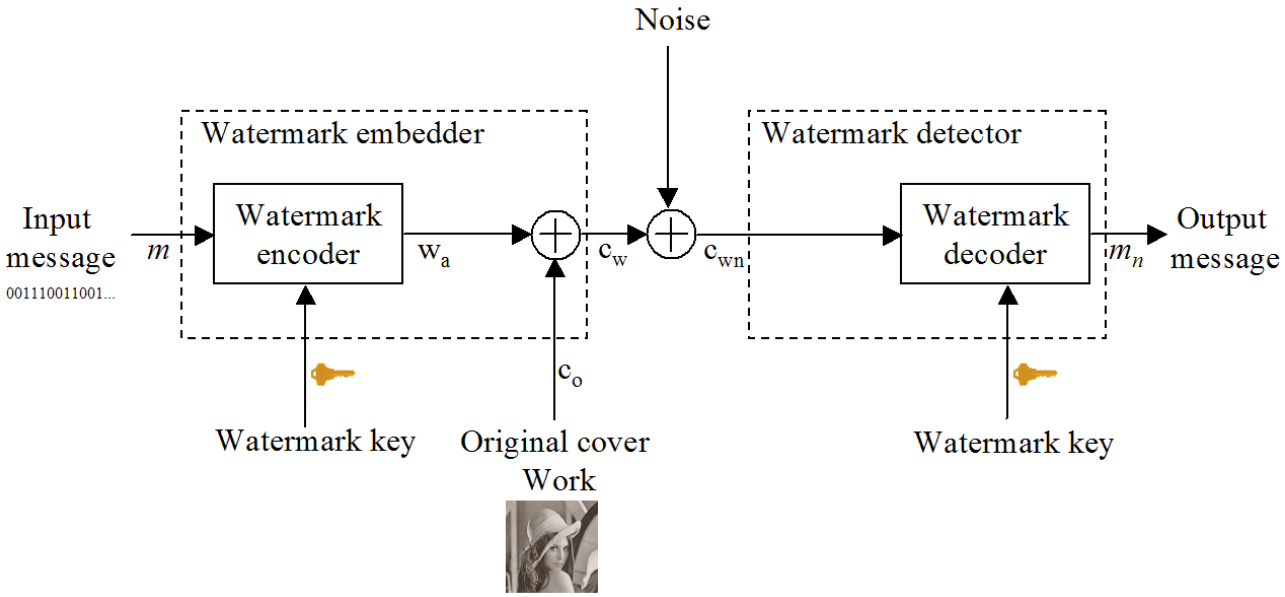


Schéma de détection aveugle

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



Tatouage par substitution du bit de poids faible (LSB)

Choix de quelques pixels

Chacun des pixels va être modifié pour « embarquer » un bit

Tatouage par substitution du bit de poids faible (LSB)

- 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).

Le patchwork de 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 :

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

Le 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.

Tatouage par étalement de spectre (Spread Spectrum)

La technique provient du monde des télécommunications. Un message m est composé d'un ensemble de symboles $m[i]$ ($m[i]$ vaut bien souvent 0 ou 1). Chaque symbole $m[i]$ est transmis à travers un signal appelé **porteuse** et noté u_i. Une porteuse est un signal pseudo-aléatoire (obtenu par un GNPA) pouvant être composé de 0 et de 1 ou bien distribué suivant une loi Gaussienne normale $\mathcal{N}(0,1)$. On peut également contraindre les porteuses à être orthogonales ($\forall i$, $\forall j$, $u_i.u_j = 0$).

Tatouage par étalement de spectre (Spread Spectrum)

Insertion :

- w_i : un vecteur (porteuse) de la taille du signal hôte N,
- m : un message composé de N_c bits.
- s : une fonction (appelée modulation) $0, 1 \rightarrow \mathbb{R}$. Par exemple $s(m[i]) = \gamma(-1)^{m[i]}$ avec γ un facteur réglant l'ampleur de la distortion.
- La marque est alors $w = \sum_{i=1}^{N_c} w_i.s(m(i))$
- l'insertion est alors $c_w = c_o + w$

Détection :

- Soit c_{wn} un signal tatoué attaqué. Le message extrait est $\hat{m}[i] = sign(c_{wn}.u_i)$ avec :

$$sign(x) = \begin{cases} 0 \ si \ x > 0 \\ 1 \ si \ x \leq 0 \end{cases}$$

Insertion et détection d'un unique bit 0 ou 1

Tatouage par étalement de spectre (Spread Spectrum)

Insertion aveugle de 1 bit

FIG.: Exemple sur cette image monochrome 3×3

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

Tatouage par étalement de spectre (Spread Spectrum)

Insertion aveugle de 1 bit

$$I_w = I + \alpha \cdot W$$

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

Tatouage par étalement de spectre (Spread Spectrum)

Insertion aveugle de 1 bit

(a) Image originale (b) Porteuse insérée (c) Image tatouée

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

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

Tatouage par étalement de spectre (Spread Spectrum)

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 : $c_w = c_o + w_a$.

Tatouage par étalement de spectre (Spread Spectrum)

Détection aveugle de 1 bit

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 :

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

Tatouage par étalement de spectre (Spread Spectrum)

Détection aveugle de 1 bit

Sortie du détecteur :

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

Code source extrait du livre de Cox, Miller et Bloom

Construction d'un patern (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 );
}
```

Code source extrait du livre de Cox, Miller et Bloom

Normalisation du patern (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 clés (comment utiliser une clé ?)

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).

Code source extrait du livre de Cox, Miller et Bloom

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;
}
```

Code source extrait du livre de Cox, Miller et Bloom

Création de ma marque W_m par modulation du message m et d'un patern 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 ];
}
```

Code source extrait du livre de Cox, Miller et Bloom

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 ] );
}
```

Code source extrait du livre de Cox, Miller et Bloom

Détection de la marque 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;
}
```

Code source extrait du livre de Cox, Miller et Bloom

Rappel : Produit scalaire

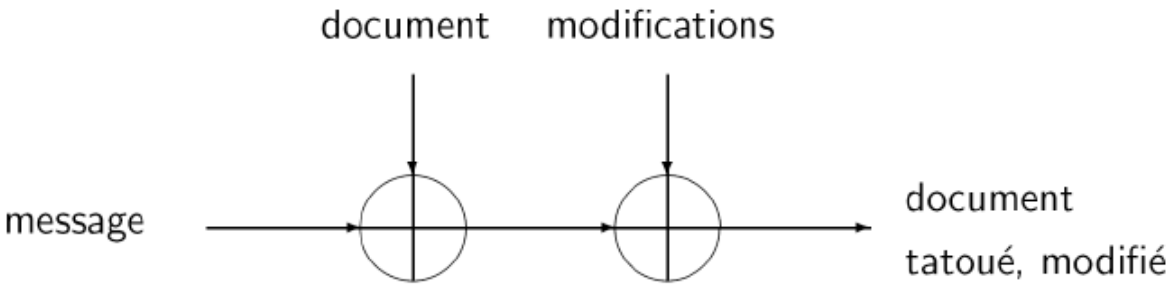
```
/*-------------------------------------------------------------------------*
 | ImgPatInnerProduct -- get the inner product of an image and a pattern   |
 |                                                                         |
 | Arguments:                                                              |
 |   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;
}
```

Quelques 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.

Quelques mots sur le tatouage avec information adjacente (2ème génération)

1998 : on effectue en fait une transmission . . .



Quelques mots sur le tatouage avec information adjacente (2ème génération)

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.