

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche scientifique

Université SAÂD DAHLAB, Blida
USDB



Faculté des sciences
Département informatique

Mémoire Présenté par :

MORSLI Mohamed Amine
TAIBI Abdelhafid

En vue d'obtenir d'un diplôme de Master 2

Domaine : Mathématique et Informatique

Filière : Informatique

Spécialité : Informatique

Option : ingénierie de logiciel

Sujet :

Protection de contenu multimédia par le
traçage des copies de pirates dans la norme
de compression H.264/AVC.

Organisme d'accueil : Centre de Développement des Technologies Avancées CDTA

Promotrice : AIT SAADI Karima

Soutenance le :

Devant le jury composé de :

M. Président

M. Rapporteur

M. Examineur

---Promotion 2010/2011---

Remerciement

L'écriture des remerciements est un exercice terriblement difficile.

Difficile, car ces mots clôturent un travail qui pourrait être intemporel dans l'absolu. Difficile, car les mots sont écrits et figés dans le temps ; ainsi la crainte de froisser autrui par une omission involontaire est présente. Que ces personnes soient ici remerciées.

Tout d'abord, nous tenons à remercier Allah le tout puissant, de nous avoir donné de la volonté, la force et la foi d'arriver à la finalité de ce modeste travail.

Nous réservons ces lignes en signe de reconnaissance à toute personne qui a contribué directement ou indirectement à la réalisation de ce mémoire :

Nous remercions plus sincèrement nos parents qui nous ont beaucoup soutenus pendant toutes la vie, et qui continueront à nous aider dans tous les projets d'avenir.

Par ailleurs nous souhaitons manifester notre reconnaissance particulièrement à Mme AIT SAADI pour ces conseils précieux, sa patience, son dévouement et le soutien qu'elle nous a témoigné pour achever à terme cette modeste thèse.

Nos remerciements s'adressent de même à l'ensemble des enseignants du département d'informatique de l'université Saad Dahleb, qui nous ont énormément aidés au cours de notre formation.

Nous tenons aussi à remercier l'ensemble du personnel du centre de développement des technologies avancées (CDTA) pour leur disponibilité, et leurs conseils.

Nous remercions également les membres de jury qui nous feront l'honneur de juger notre travail.

Résumé



Résumé

Le fingerprinting ou l’empreinte digitale est une des applications de la technologie du tatouage numérique qui permet le traçage de la distribution des contenus multimédia tout en les protégeant de la redistribution non autorisée. La marque qui représente l’information d’identification unique est insérée dans chaque copie distribuée du signal multimédia et sert comme une empreinte digitale.

Dans cette thèse, nous avons développé une solution de protection du contenu par le traçage des copies des pirates dans la dernière norme de compression H.264/AVC. Le code de Tardos probabiliste qui constitue l’empreinte est inséré dans les signaux vidéo compressés par la norme H.264/AVC en utilisant la technique d’étalement du spectre. L’invisibilité du tatouage est assurée par l’adoption d’une technique d’insertion adaptative en fonction du contenu. Différentes attaques ont été effectuées pour évaluer la robustesse de la technique développée: les attaques par collusion linéaires réalisées dans le domaine de pixels et l’attaque de compression. L’insertion des bits de la marque est effectuée dans le domaine fréquentiel de la norme plus précisément dans les coefficients transformés et quantifiés non-nulle.

Mots clés : Tatouage vidéo, tatouage robuste, protection de contenu, code de Tardos, traçage de contenu, standard H.264/AVC.

Abstract

Digital fingerprinting is a technology for tracing the distribution of multimedia content and protecting them from unauthorized redistribution. Unique identification information is embedded into each distributed copy of multimedia signal and serves as a digital fingerprint.

In this thesis, we developed an approach for active fingerprinting of state of the art video codec H.264/AVC. Tardos probabilistic fingerprinting code is embedded in H.264/AVC video signals using spread spectrum watermarking technique. The invisibility is ensured by the attention adaptation of a technique of inserting adaptive depending on the content of the video. Different attacks have been performed to evaluate the robustness of the developed technique: Linear collusion attacks which have been performed in the pixel domain and compression attack. The embedding has been performed in the non-zero quantized transformed coefficients

Keywords: watermarking of video, robust watermarking, content protection, the Tardos code, content tracking, standard H.264/AVC, fingerprinting.

ملخص

بسم الله الرحمن الرحيم

البصمة الإلكترونية هي تطبيق من تطبيقات تكنولوجيا الوشم الإلكتروني القادرة على تعقب الملفات الإلكترونية (فلم، موسيقى، صورة...) و حمايتها من القرصنة وإعادة التوزيع الغير قانوني. الشيفرة التي تمثل معلومة تحديد استثنائية (وبمناوبة بصمات الأصابع) توضع في نسخة وحيدة للتوزيع. الشفرة المستعملة قادرة على تتبع و تحديد هوية الشخص الذي قام بنسخ و تسريب الفلم.

أنجزنا في هذه المذكرة حلاً لحماية المحتوى و تعقب النسخ المقرصنة من خلال المعيار الأخير لضغط الفيديو H.264. تعتمد شفرة تاردوس Tardos على الاحتمالات الإحصائية، و التي تمثل البصمة الإلكترونية تدرج داخل الفلم المضغوط عن طريق معيار الضغط H.264 باستعمال طريقة عرض الطيف. تمكن تقنية الإدراج المكيف على حساب المحتوى من تمويه الشفرة عن العين المجردة.

نفذت عدة هجومات لتقييم متانة التقنية المطورة، حيث نفذت هجومات التآمر في مجال البكسل ، بالإضافة إلى هجوم الضغط. تم ادراج الشفرة في مجال الترددات، تحديدا في المعاملات المقدرة و الغير معدومة.

الكلمات الرئيسية: الوشم الإلكتروني، ضغط الفيديو باستعمال معيار الضغط H.264، حماية المحتوى، تتبع المحتوى، شفرة تاردوس، البصمة الإلكترونية.

Sommaire



Sommaire

Introduction générale	1
Chapitre 1 : Tatouage numérique.....	5
1. Introduction.....	6
2. Tatouage numérique	7
2.1. Définition.....	7
2.2. Propriétés du tatouage numérique.....	7
2.3. Classification des systèmes du tatouage numérique	9
2.3.1. Domaine d'insertion.....	9
2.3.2. Type de la marque insérée	11
2.3.3. Tatouage selon le type d'extraction	11
2.3.4. Règle d'insertion.....	13
2.4. Les applications du tatouage vidéo	14
2.5. Les attaques.....	16
2.6. Application du tatouage numérique au système de traçage de copies des pirates.....	17
2.6.1. Contraintes des systèmes de traçage de copies des pirates.....	19
2.6.2. Collusion linéaire	21
2.6.3. Collusion non linéaire	22
2. Conclusion	23
Chapitre 2 : La norme de compression H.264/AVC.....	24
1. Introduction.....	25
2. Fonctionnement du codeur H.264/AVC.....	25
2.1. Codage Intra dans H.264/AVC	27
2.2. Codage du résiduel.....	27
2.2.1. Transformation.....	27
2.2.2. La quantification	30
2.2.3. Le balayage de bloc.....	31
2.2.4. Le codage entropique	31

2.2.5.	Calcul du résiduel.....	32
2.2.6.	La prédiction des blocs Intra 4x4.....	32
2.2.7.	La prédiction des blocs Intra 16x16.....	33
2.2.8.	Le filtre de déblocage.....	34
3.	Les profils	34
4.	Conclusion	36
Chapitre 3 : Traçage des copies de pirates		37
1.	Introduction.....	38
2.	Traçage de copie de pirates par approche cryptographique	38
2.1.	Principe du code correcteur d'erreur.....	41
3.	Traçage de copie de pirates à la mode statistique	41
3.1.	Code de Tardos	42
3.1.1.	Principe de Tardos	42
3.1.2.	Propriétés.....	43
4.	Traçage des copies de pirates à la mode de tatouage numérique.....	44
5.	Conclusion.....	45
Chapitre 4 : Implémentation et réalisation.....		46
1.	Introduction.....	47
2.	Etat de l'art de tatouage numérique dans H.264/AVC	47
3.	Système de traçage proposé et développé	48
3.1.	Génération des codes binaires de traçabilité.....	49
3.1.1.	Génération des probabilités	50
3.1.2.	Génération de la matrice de code.....	51
3.1.3.	Accusation	51
3.2.	Processus d'insertion.....	52
3.2.1.	Sélection des trames et les positions pour l'insertion	54
3.2.2.	L'insertion de la marque.....	56
3.3.	Processus d'extraction.....	60
3.4.	L'accusation.....	61

4. Présentation de l'application.....	62
5. Conclusion	66
Chapitre 5 : Tests et résultats.....	67
1. Introduction.....	68
2. Tests de la génération du code de Tardos	71
3. Tests des processus d'insertion et d'extraction.....	72
4. Evaluation de la robustesse face aux attaques	76
5. Conclusion	78
Conclusion générale.....	79
Bibliographie	



Liste des Figures et Tableaux



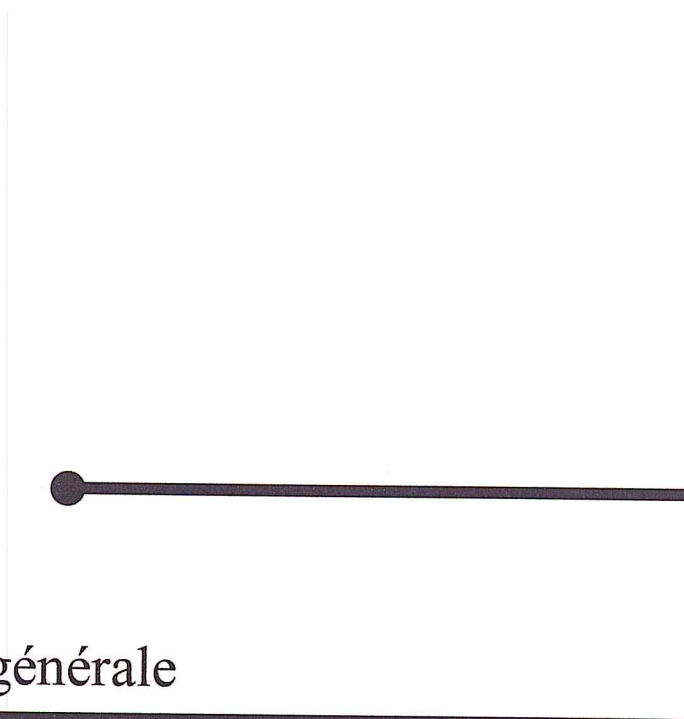
A. Liste des figures

Figure 1.1.	Schéma général d'un système de tatouage d'image.....	7
Figure 1.2.	Caractéristique de la marque.....	8
Figure 1.3.	Classification de tatouage numérique.....	9
Figure 1.4.	Comparaison de la robustesse aux applications de traçage de documents, authentification et l'indexation du document.....	11
Figure 1.5.	Mode d'extraction non-aveugle.....	12
Figure 1.6.	Mode d'extraction aveugle.....	12
Figure 1.7.	Mode d'extraction semi-aveugle.....	13
Figure 1.8.	Schéma de distribution à n utilisateur via un serveur VoD.....	18
Figure 1.9.	Scénario classique de traçabilité.....	19
Figure 1.10.	Collusion en tatouage numérique: Plusieurs utilisateurs rassemblent plusieurs documents tatoués et les combinent pour produire des documents ne contenant plus aucun tatouage.....	21
Figure 1.11.	Attaque de collusion par la moyenne.....	22
Figure 1.12.	Attaque de collusion par couper-coller.....	22
Figure 1.13.	Attaque de collusion non linéaire.....	23
Figure 2.1.	Bloc diagramme du codeur H.264/AVC.....	26
Figure 2.2.	Bloc diagramme du décodeur H.264/AVC.....	26
Figure 2.3.	L'ordre de parcours d'un macrobloc 4x4.....	28
Figure 2.4.	L'ordre de balayage des blocs résiduels dans un macrobloc 4x4..	28
Figure 2.5.	Balayage progressif de pour les blocs 4x4 et 8x8.....	31
Figure 2.6.	Labellisation des échantillons de prédiction 4x4.....	32
Figure 2.7.	Les modes de prédiction intra.....	33
Figure 2.8.	Modes de prédiction des blocs 4x4 de luminance.....	33
Figure 2.9.	Modes de prédiction des blocs 16x16 de luminance.....	34
Figure 3.1.	La séquence pirate y appartient aux ensembles de descendance de trois collusions C , C' et C''	40
Figure 4.1.	Classification des méthodes de tatouage dans la norme H.264/AVC	47
Figure 4.2.	Schéma d'insertion de la marque.....	49
Figure 4.3.	La matrice de code de longueur m pour n utilisateurs.....	51
Figure 4.4.	Schéma du processus d'insertion dans le codeur H.264.....	52
Figure 4.5.	Organigramme général d'insertion.....	53
Figure 4.6.	Organigramme de sélection des macroblocs.....	55
Figure 4.7.	Schéma de tatouage d'un macrobloc.....	56
Figure 4.8.	Organigramme détaillé d'insertion.....	59
Figure 4.9.	Schéma du processus d'extraction.....	60
Figure 4.10.	Organigramme d'extraction.....	61
Figure 4.11.	Fenêtre principale de l'application.....	62
Figure 4.12.	Lecture de la séquence Bus_cif.....	62
Figure 4.13.	Fenêtre de génération de code.....	63
Figure 4.14.	Bouton de compression et décompression.....	63
Figure 4.15.	Fenêtre de compression.....	64
Figure 4.16.	Fenêtre de décompression.....	64
Figure 4.17.	Bouton d'insertion et d'extraction de la marque.....	65
Figure 4.18.	Fenêtre d'insertion de la marque.....	65
Figure 4.19.	Fenêtre d'extraction de la marque.....	66
Figure 5.1.	Liste des séquences vidéos de tests.....	69

Figure 5.2.	Résultat d'insertion de Shahid.....	73
Figure 5.3.	Comparaison des séquences vidéos.....	75
Figure 5.4.	Diagramme de PSNR de chaque copie attaquée à partir de K copies pirates pour différentes attaques.....	77
Figure 5.5.	Diagramme de SSIM de chaque copie attaquée à partir de K copies pirates pour différents attaques.....	78

B. Liste des tableaux

Tableau 1.1.	Comparaison entre le tatouage dans les domaines fréquentiel et spatial.....	10
Tableau 1.2.	Comparaison entre les règles additives et substitutives.....	14
Tableau 2.1.	Taille du pas de quantification dans le codec H.264.....	30
Tableau 2.2.	Les profils de standard H.264/AVC.....	36
Tableau 3.1.	Matrice de code pour 4 utilisateurs.....	43
Tableau 3.2.	Tableau de traitement.....	43
Tableau 5.1.	Paramètre d'entrée utilisé pour le codeur H.264/AVC.....	70
Tableau 5.2.	Paramètre d'entrée utilisé pour la génération du code de Tardos...	72
Tableau 5.3.	Les résultats d'insertion dans les vidéos de tests.....	74
Tableau 5.4.	Nombre de pirates tracés à partir de K copies pirates pour différents attaques	76
Tableau 5.5.	PSNR de chaque copie attaquée à partir de K copies pirates pour différentes attaques	77
Tableau 5.6.	SSIM de chaque copie attaquée à partir de K copies pirates pour différents attaques	78



Introduction générale

Introduction générale

Le développement rapide des systèmes d'information et les réseaux de communication et l'ouverture des technologies de l'information au grand public ont causé une grande souplesse d'échange d'information et offrent une facilité de circulation des documents multimédia (image, vidéo, texte, sons, etc.). Ce phénomène a ouvert de nouveau problématique de protection et de contrôle des données échangées car les documents numériques peuvent être dupliqués, modifiés et transformés illégalement. Pour cela il est nécessaire développer des mécanismes permettant de lutter contre les dépassements et protéger ces documents.

Dans ce contexte, la meilleure solution pour renforcer la sécurité multimédia apparue jusqu'à présent c'est le « tatouage numérique » (ou WATERMARKING). Le principe général de tatouage numérique est de cacher une information visible ou invisible suivant la nature de document permettant d'assurer un service de sécurité (intégrité, traçabilité, copyright). L'une des particularités de watermarking est que la signature est liée de manière intime et résistante aux données. Au contraire de watermarking, la cryptographie assure la sécurité au cours de transfert et si le document atteint la destination, le destinataire peut modifier et transférer le document sans être détecté. D'une manière générale le watermarking est théoriquement indépendant du format du fichier et il est peut être détecté ou extrait même si le document a subi des modifications ou s'il est incomplet.

L'industrie de vidéo a bénéficié du développement de technologie de l'information et connaît une énorme amélioration ces dernières années au point de vue qualité vidéo et sa diffusion. Cette dernière a connu une révolution avec les nouveaux supports de stockage comme DVD et Bleu-Ray, la fibre optique et l'émission par satellite. Et ceci a ouvert les portes pour les chaînes câblées par la fibre optique, les éditions de distribution des films DVD ou Bleu-Ray et les chaînes satellites. Dans le cas de diffusion de vidéo par support de stockage ou par des chaînes câblées, la vidéo peut être traitée et rediffusée sans détecter la source de cette fuite.

Afin de sécuriser le contenu vidéo, la communauté de watermarking propose des solutions de traçabilité du contenu.

C'est dans ce contexte que notre mémoire s'intègre, il s'agit de développer une solution permettant de détecter des copies illicites des vidéos compressées dans la courante norme de compression H.264/AVC.

L'application type de notre travail est la vidéo à la demande sur Internet. Un serveur distribue des copies personnalisées d'une même vidéo à n utilisateurs. Parmi ceux-ci, certains sont malhonnêtes et redistribuent illégalement des copies pirates. Pour ce faire, un identifiant unique sous la forme d'une séquence de m bits est caché dans chaque vidéo distribuée à l'aide d'une technique de tatouage numérique, aussi sont produites n copies du contenu, toutes différentes mais perceptiblement identiques. Cet identifiant permet de tracer la source de copies pirates. L'identifiant dans notre cas est un code appelé code de Tardos. C'est un code de traçabilité basé sur des probabilités.

Ce mémoire regroupe les chapitres suivant :

Chapitre 1 : Tatouage numérique.

Dans ce chapitre nous allons présenter des généralités sur le tatouage numérique, son principe, ses domaines d'utilisation, la classification de ses techniques, ainsi que les caractéristiques. Nous allons aussi abordé dans ce chapitre les différentes stratégies d'attaques pour l'application de la traçabilité (fingerprinting).

Chapitre 2 : La norme de compression H.264/AVC.

Ce chapitre décrit le standard de compression H.264/AVC, en expliquant ses principaux modules jugés nécessaires dans notre travail tels que : la prédiction, la transformation, la quantification et le codage entropique. De plus, la présentation des profils de la norme H.264.

Chapitre 3 : Traçage des copies de pirates.

Une présentation sur les approches de traçage de traitre fait l'objet du troisième chapitre. La traçabilité forte qui est étudiée par les cryptographes et la traçabilité faible élaborée par les mathématiciens. Ce chapitre inclura aussi la contribution de tatouage numérique dans le domaine de traçage des traitres.

Chapitre 4 : Implémentation et réalisation.

Ce chapitre détaillera le système de sécurisation du contenu multimédia par le traçage des copies de pirates en utilisant le tatouage numérique dans la norme de compression H.264/AVC. Le système proposé comporte trois procédures à savoir : la

génération des codes de traçabilité de Tardos, l'insertion et l'extraction des codes dans les vidéos compressées. Suivie par une présentation de l'interface de l'application et les différentes fonctionnalités offertes par le système.

Chapitre 5 : Tests et Résultats.

Le dernier chapitre présentera les différents résultats des tests effectués pour évaluer les performances de la méthode proposée. En commençant par évaluer la capacité d'insertion et l'imperceptibilité, puis l'évaluation de la robustesse des différentes attaques élaborées.

Enfin, Nous terminerons par une conclusion générale en énonçant quelques perspectives.

Le sujet entre dans le cadre des projets de l'équipe « *Biométrie et Sécurité multimédia* » plus particulièrement le projet « *Sécurisation des documents multimédia par tatouage numérique* » initié par le centre de développement des technologies avancés CDTA dans le cadre du protocole de recherche 2011-2013.

Chapitre 1

Tatouage numérique

1. Introduction

Afin de transmettre de manière sécurisée des contenus audio-visuels, de nombreux produits ont développé des méthodes faisant appel aux techniques de cryptographie, ainsi qu'aux méthodes de tatouage numérique. Cependant les deux problèmes major des approches cryptographiques concernent le chiffrement du contenu vidéo et la complexité de ces approches.

Dans le contexte où les échanges d'information dématérialisés se développent, il est indispensable de pouvoir bénéficier de systèmes sécurisés, afin de protéger les données à caractère personnel ou confidentiel. Les techniques cryptographiques sont reconnues comme étant des outils essentiels de la sécurité et de la confiance. Elles jouent un rôle essentiel en métiers de protection contre les fraudes informatiques et de sécurité des données. Le problème de ces techniques est lorsque le document numérique est déchiffré ou désembrouillé, il n'existe plus de protection et ce dernier peut être rediffusé en toute impunité. Pour faire face à ce problème, une nouvelle technique a vu le jour. Il s'agit du tatouage numérique.

Le « watermarking » ou tatouage numérique peut être perçu comme une branche de la stéganographie qui signifie littéralement « écriture cachée ». La stéganographie consiste à cacher, de manière subliminale, un message secondaire dans un message primaire [1].

Le tatouage des données numériques est une discipline qui trouve son origine dans le manque de techniques fiables de protection de ce type de données. En effet, associé à d'autres techniques, cet axe de recherche a pour but de résoudre des problèmes aussi variés que la protection du copyright, le contrôle de copies, le suivi de documents et l'intégrité du contenu des données et aussi le traçage de copies illégales.

Les premiers articles sur le sujet sont apparus au début des années 90. Très vite, de nombreux laboratoires se sont intéressés à ce domaine. Depuis 1995, le nombre de publications et de brevets a fait du tatouage un domaine majeur en traitement d'image [1].

2. Tatouage numérique

2.1. Définition

Le tatouage d'image, que l'on peut sommairement décrire à l'aide de la figure 1.1, consiste à introduire, généralement de manière invisible, une information dans une image, puis à tenter de la récupérer après que l'image ait éventuellement subi des manipulations de natures variées. [1][2]

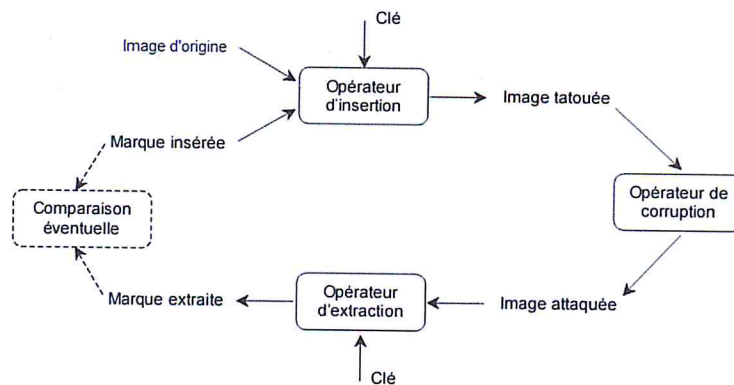


Figure 1.1. Schéma général d'un système de tatouage d'image

2.2. Propriétés du tatouage numérique

Les propriétés du tatouage numérique diffèrent d'une application à une autre selon les besoins de ces applications. Certaines propriétés sont définies pour des applications. Ces propriétés sont [3]:

- **Robustesse** : c'est la capacité que possède un algorithme de tatouage à résister aux attaques extérieures, qu'elles soient bienveillantes ou malveillantes. Par exemple pour la vidéo, il peut s'agir d'attaques simples comme le changement de format, la compression, le changement de débit ou tous autres traitements classiques (il s'agit ici de traitements bienveillants qui ne visent pas forcément à retirer la marque). On peut aussi avoir des attaques ciblées, qui ont pour seul but de retirer la marque, comme des attaques statistiques ou des attaques basées sur la connaissance de l'algorithme utilisé.
- **Invisibilité** : L'imperceptibilité se rapporte au dispositif perceptuel du tatouage. Dans le meilleur des cas, aucune différence perceptible entre le signal tatoué et

l'original ne devrait exister. Une manière simple de réduire la déformation pendant le processus de tatouage. Cependant, ceci le rend facile pour un attaquant de changer l'information de tatouage (marque) sans être remarqué.

- **Capacité** : désigne le nombre de bits qui peuvent être insérés dans la donnée à tatouer tout en respectant les autres propriétés. Bien qu'une grande capacité soit désirée, son insertion cause des dégradations en termes de *robustesse* et d'*invisibilité*. Plus la capacité est grande, plus la qualité perceptuelle de la donnée tatouée sera dégradée et moins robuste sera la marque insérée.
- **Sécurité** : L'exigence de sécurité d'un système de tatouage peut différer légèrement selon l'application. La sécurité de tatouage implique qu'il devrait être difficile d'enlever ou changer la marque sans endommager le signal de média. La sécurité de tatouage peut être considérée comme la capacité d'assurer le secret et l'intégrité de l'information de tatouage, et résiste aux attaques malveillantes [4].

Concevoir un algorithme de tatouage revient à trouver le meilleur compromis entre les trois premières propriétés, en fonction de l'application visée, tout en respectant la quatrième propriété qui est la sécurité de la marque. La figure 1.2 présente une bonne illustration de ces caractéristiques.

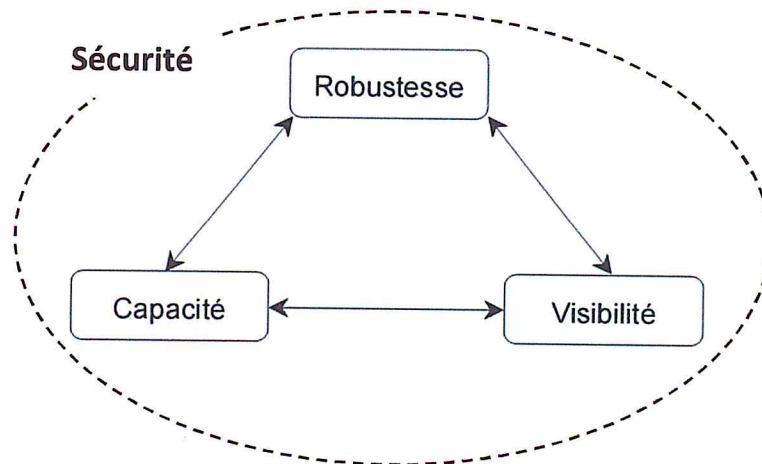


Figure 1.2. Caractéristique de la marque

2.3. Classification des systèmes du tatouage numérique

Les systèmes du tatouage numérique sont classifiés selon plusieurs critères (Figure 1.3). Ces critères sont :

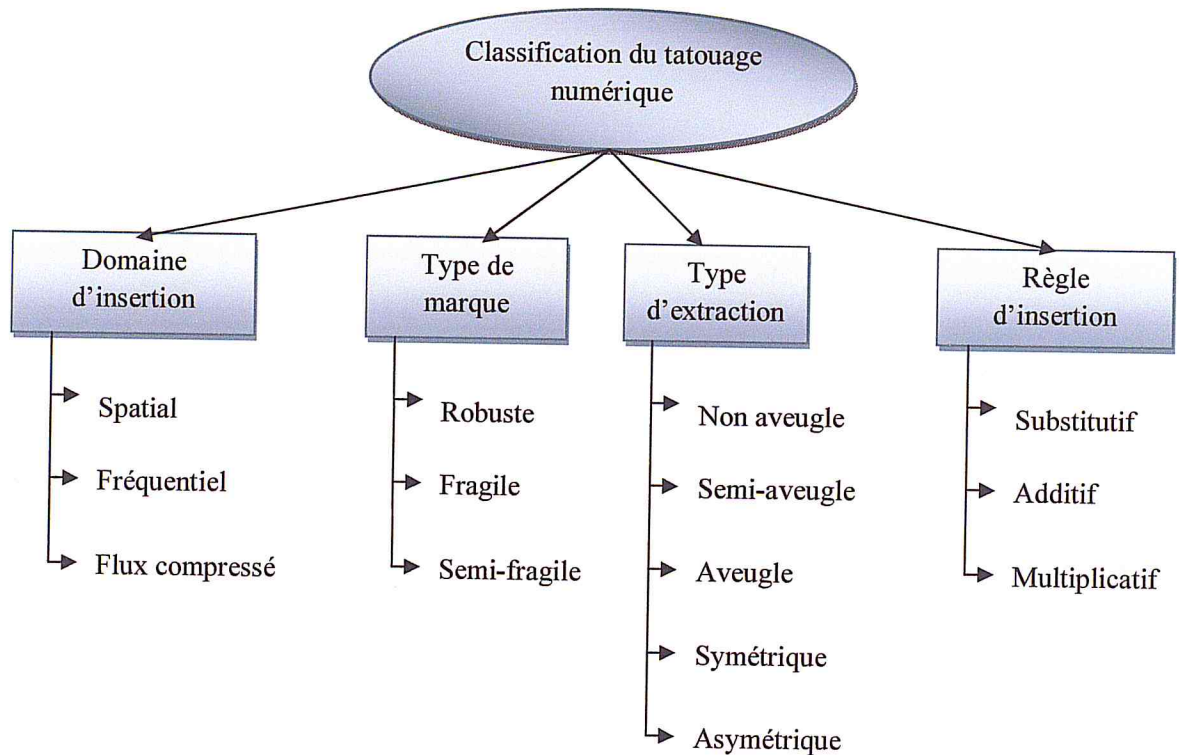


Figure1.3. Classification de tatouage numérique

2.3.1. Domaine d'insertion

La diversité des classifications de tatouage est liée aux choix du domaine d'insertion de la signature. Chaque espace de représentation de document numérique apporte diverse possibilités en termes de performance et de robustesse :

- **Domaine spatial** : les méthodes qui viennent en premier à l'esprit sont celles du domaine spatial, où elles modifient et agissent directement sur les composantes de luminance des pixels. Comme aucun traitement initial n'est requis, ces algorithmes sont très rapides et permettent de travailler en temps réel [3]. Leur problème réside dans leur fragilité face aux attaques.
- **Domaine fréquentiel** : Le tatouage peut être appliqué dans le domaine fréquentiel en appliquant d'abord une transformation comme la transformée de Fourier rapide (FFT), la Transformée en Cosinus Discrète (TCD) ou transformée

en Ondelette Discrète (TOD). D'une façon semblable au tatouage dans domaine spatial, les valeurs des fréquences choisies peuvent être changées de l'image d'origine. Puisque des fréquences seront perdues par compression, le tatouage est appliqué aux fréquences inférieures, ou meilleures encore, appliqués de manière adaptative aux fréquences qui contiennent l'information importante de l'image d'origine. Puisque les bits de la marque insérée dans le domaine fréquentiel seront dispersés au-dessus de l'intégralité de l'image spatiale sur la transformation inverse, cette méthode n'est pas aussi susceptible de la défaite par la culture que la technique spatiale. Cependant, il y a plus une différence ici au niveau de l'invisibilité, puisque le tatouage en effet est appliqué aléatoirement à travers l'image spatiale. Le tableau 1.1 illustre l'étude comparative entre les deux domaines d'insertion [3].

Tableau 1. 1. Comparaison entre le tatouage dans les domaines fréquentiel et spatial

	Domain spatial	Domain fréquentiel
Coût de calcul	Bas	Haut
Robustesse	Fragile	Plus robuste
Qualité perceptuelle	Haut	Bas
Capacité	Haut (dépend de la taille de l'image)	Bas

- **Domain compressé :** Le domaine compressé (flux compressé) est obtenu du domaine fréquentiel, les algorithmes de tatouage sont fragiles et sont appliquées directement sur le flux binaire compressé. La plupart des algorithmes utilisent le code de longueur variable VLC (Variable Length Code) pour l'insertion. Les principaux avantages de ce domaine est que les méthodes sont sans perte d'information et les marques insérées sont invisibles. Au plus, il ya aucune augmentation dans la taille de fichier.

2.3.2. Type de la marque insérée

En plus de la classification du tatouage en fonction du domaine d'insertion, on définit aussi le type de tatouage en fonction de sa résistance aux attaques. Il existe trois types de schémas de tatouages en fonction de leur résistance [5].

- **Tatouage robuste** : il est important d'espérer qu'une marque insérée par tatouage contienne une quantité d'information et soit aussi robuste à une très grande variété de traitement. Il est nécessaire de faire une comparaison entre le niveau de robustesse et la "fonctionnalité" du marquage (figure 1.4).

Le tatouage dédié au traçage de contenu doit être aussi robuste que possible car la signature permet dans ce cas de certifier la propriété de l'image. La robustesse du tatouage dédiée à l'authentification des documents doit être contrôlée. Dans le cas du tatouage pour l'indexation de document, la quantité d'information à insérer est importante plus que la robustesse.



Figure 1.4. comparaison de la robustesse aux applications de traçage de documents, authentification et l'indexation du document.

- **Tatouage fragile** : Un tatouage est fragile si la marque cachée dans l'image est détruite dès que l'image tatouée subira n'importe quelle manipulation. Quand un tatouage fragile est présent dans un signal, nous pouvons impliquer, avec une probabilité élevée, que l'image n'a pas été changée.
- **Tatouage semi-fragile** : Les méthodes ayant recours à un tatouage semi-fragile se distinguent des méthodes fragiles dans la mesure où elles offrent une robustesse accrue face à certaines manipulations d'image.

2.3.3. Tatouage selon le type d'extraction

Il existe plusieurs types de tatouage selon le processus d'extraction de la marque :

- l'extraction non-aveugle (figures 1.5).
- aveugle (figure 1.6).
- semi-aveugle (figures 1.7).

Ces modes spécifient l'information a priori dont dispose le module d'extraction pour la vérification du tatouage. L'utilisation de tel ou tel mode dépendra de l'application visée et des protocoles utilisés [1].

- **Schéma non-aveugle** : le décodeur dispose de l'image tatouée ainsi que de l'image d'origine. Ce contexte est bien évidemment incompatible avec des applications visant à vérifier l'intégrité de l'image, ou à assurer la vérification en temps réel du copyright (problème de temps d'accès à la base de données contenant les informations d'origine) [1].

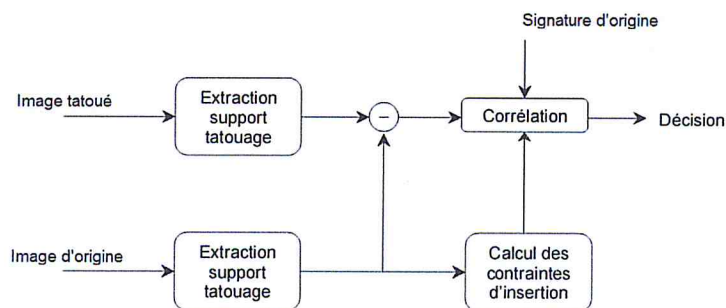


Figure 1.5. Mode d'extraction non-aveugle

- **Schéma aveugle** : il s'agit du seul mode où l'on peut réellement parler d'extraction du tatouage (par opposition à la vérification intervenant dans les deux précédents modes) puisque l'on ne présume ni la connaissance du tatouage, ni la connaissance de l'image d'origine. C'est le mode d'extraction le plus intéressant, mais également le plus difficile à mettre en œuvre [1].

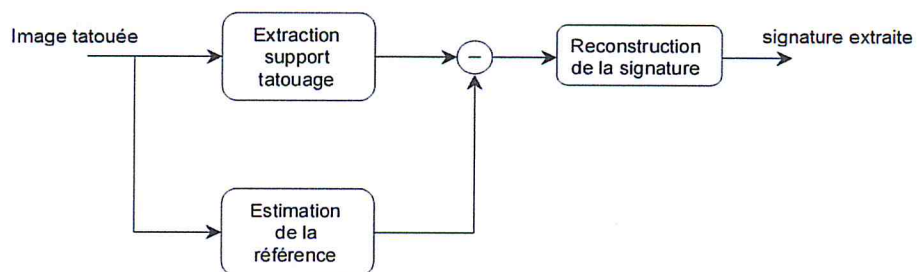


Figure 1.6. Mode d'extraction aveugle

- **Schéma semi-aveugle** : la marque d'origine est supposé connue lors de l'extraction et elle est utilisée le plus souvent via un score de corrélation [1].

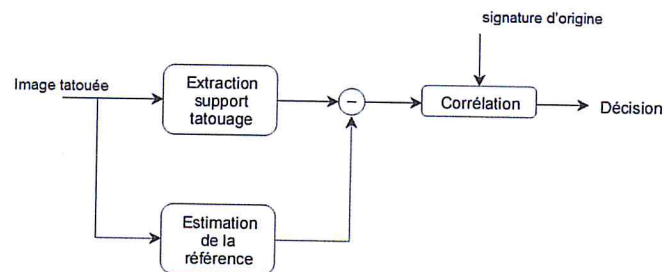


Figure 1.7. Mode d'extraction semi-aveugle

- **Schéma symétrique** : Dans ce schéma, La marque insérée est issue du codage du message à transmettre. Il est dépendant d'une clé. Cette même clé est nécessaire au décodage pour l'extraction du message [6].
- **Schéma asymétrique** : Le tatouage asymétrique repose sur l'utilisation de deux clés : une clé k_I privée pour l'insertion et une clé k_D publique pour la détection. K_D est issue de k_I par une transformation non inversible. N'importe quel utilisateur peut détecter la marque en connaissant K_D , mais seule la connaissance de k_I permet d'enlever ou modifier la marque [6].

2.3.4. Règle d'insertion

- **Règle additive** : Les approches additives constituent une classe particulière de méthode ou la signature est ajoutée à des composantes de l'image. Le principe de ce schéma consiste à sélectionner un certain nombre de composantes de l'image I , la composante extraite forme un vecteur $C(I)$. Une signature est ensuite générée W . la signature est ajoutée aux composantes $C(I)$ pour obtenir les composantes de l'image marquée $C(I_w)$ [7]. Le schéma additif est formulé selon l'équation (1.1):

$$C(I_w) = C(I) + \alpha W \quad (1.1)$$

- **Règle substitutive** : Les approches substitutives peuvent être formulées en 4 étapes :
 - Une sélection des composantes de l'image I selon une clé secrète K pour obtenir un vecteur $C_k(I)$.

- La signature à insérer est obtenue en appliquant une contrainte F sur les composantes sélectionnées $C_k(I)$ en fonction du message à insérer W , la substitution s'effectue comme suit :

$$C_k(I_w) = F(C_k(I), W) \quad (1.2)$$

- L'image marquée est reconstruite à partir des composantes modifiées.

La méthode par quantification est l'approche la plus répandue dans ce type de tatouage [8]. Elle consiste à utiliser les bits de poids faible (LSB) dont la modification n'affecte pas la qualité visuelle.

Une comparaison entre les deux règles d'insertion de la marque se résume dans le tableau 1.2.

Tableau 1.2. comparaison entre les règles additives et substitutives

	additive	Substitutive
Capacité	faible	Maximale
insertion	Addition d'une séquence aléatoire	Substitution de caractéristiques de l'image
détection	corrélation	Analyse de la redondance

- **Règle multiplicative** Dans cette règle, la marque à insérer est multipliée à ces coefficients. Avec la même notation utilisée précédemment, la règle multiplicative est définie par l'équation suivante :

$$Y_i = X_i(1 + \lambda * W_i) \quad \text{telque } i = 1, 2, \dots, N \quad (1.3)$$

2.4. Les applications du tatouage vidéo

Le tatouage apparaît comme une nécessité pour la protection du contenu multimédia, que ce soit pour l'image, pour l'audio ou la vidéo. Jusqu'à présent, les systèmes dits propriétaires, ont montré leurs faiblesses. En effet, ce genre de système

repose sur un secret qui, une fois dévoilé, permet de passer outre la protection. Les applications classiques d'un système de tatouage vidéo sont les suivantes :

- **Protection de copyright [2] [9]:** le tatouage offre une alternative intéressante à la cryptographie, car il permet de protéger l'image, même lorsque celle-ci est diffusée. La protection des droits d'auteur représente l'application la plus courante aujourd'hui. L'objectif est d'insérer une information dans la donnée source afin de prévenir toute revendication frauduleuse de propriété. cette marque ne doit être connue que par la personne ou de l'organisme de tatouage. Elle dépend donc d'une clé secrète, qui permet son insertion et sa détection. Cette application nécessite la mise en place d'un algorithme de tatouage d'un niveau de robustesse très élevé. En effet, celui-ci ne doit pas être ambigu et doit toujours déterminer l'appartenance du medium.
- **Authentification de données [2][10][11] :** l'objectif est de détecter toute modification éventuelle des données, afin de pouvoir certifier si celles-ci ont été modifiés ou non. On aperçoit ici une problématique de contrôle d'identité de document. Ce qui peut être obtenu avec un tatouage fragile. Le marquage pour l'authentification et donc celui qui utilise le niveau le plus faible de robustesse.
- **Protection contre la copie [2]:** un souhait des distributeurs de multimédia est l'existence d'un moyen de protection contre la copie, afin d'interdire une circulation de média piraté. Cependant, cela est difficile à obtenir pour les systèmes ouverts, mais réalisables pour les systèmes fermés ou propriétaires. Dans ces derniers, il est possible d'utiliser des marques spécifiant le statut de la copie de la donnée.
- **Indexation :** le domaine d'indexation des images consiste à classer de manière automatique des images selon leur contenu. Il permet de faciliter une recherche dans une base de données. Les techniques classiques utilisées consistent à effectuer un traitement automatique de l'image, de manière à dégager les composantes essentielles du contenu. Le tatouage d'un document permet d'insérer une information décrivant le contenu de l'image. Cela permet de qualifier sommairement l'image, ou d'insérer un pointeur vers une description plus complète.

Les deux contraintes essentielles du tatouage numérique de documents sont l'invisibilité de la marque et la robustesse lors de la détection de la signature. Néanmoins dans le cas de la protection de contenu par le traçage des copies, la robustesse doit être maximale. La signature doit alors être détectable après divers traitements.

2.5. Les attaques

Les attaques sont, le plus souvent, des traitements classiques qu'une personne effectue sur le support qu'elle utilise. Elles peuvent être des traitements visant soit à brouiller soit à enlever la marque de protection dans la vidéo. On peut distinguer deux grandes familles d'attaques, les bienveillantes et les malveillantes [5] :

- **Attaques bienveillantes** : Il s'agit de traitement qui n'a pas initialement pour objectif d'empêcher la détection de la marque. Il peut s'agir des dégradations dues à une compression, à un changement de type de compression, à des filtrages (réduction de bruit), à un changement de résolution, etc. Un autre traitement couramment utilisé en vidéo est la conversion analogique/numérique, et inversement. Enfin, certaines distorsions géométriques peuvent être utilisées : flip vertical, perte d'une ligne ou d'une colonne, etc[5].
- **Attaques malveillantes** : visent explicitement à rendre le tatouage inopérant. Ces attaques, comme souvent dans le domaine numérique, sont difficile à prouver d'un point de vue juridique. Toutefois, une attaque malveillante qui a réussi devra produire un contenu à la fois lavé de son tatouage et encore exploitable.

Nous allons maintenant citer quelques attaques malveillantes :

- **L'attaque par surmarquage** : consiste à tatouer à nouveau un média déjà tatoué. Pour certains schémas, en particulier si les lieux de tatouages sont fixés, cette attaque peut être très dangereuse. Certains algorithmes de tatouage se protègent en vérifiant, avant de distribuer une clef que le média d'origine n'est pas tatoué. Cette protection n'est utile que si le schéma de tatouage demeure inconnu. En effet, s'il est connu, un pirate peut ajouter une marque de sa fabrication qui invalidera la détection.
- **L'ajout d'un bruit** : le bruit additif et le bruit multiplicatif non corrélatif ont été en grande partie adressés dans la littérature de théorie de traitement des signaux et de théorie de communication [7].

- **Attaque par cropping** : elle consiste à extraire un morceau non tatoué d'un flux média pour le réutiliser. Pour être résistant à ce type d'attaque, le tatouage doit être présent sur tout le média. La même situation se produit dans le domaine fréquentiel du média où la marque doit être partout présente afin d'éviter une destruction par filtrage.
- **L'attaque par recopie** : consiste à recopier une marque obtenue préalablement (par exemple par estimation) sur un média non marqué. Le détecteur validera alors le nouveau média comme étant tatoué.
- **Moyennage**: Lorsqu'on dispose d'un nombre élevé d'images de contenu identique, mais avec des marques différentes (par exemple des numéros de série ou des identifications d'utilisateur), le pirate peut les moyenner, afin de produire une image sans marque détestable. De même, dans certains cas, le pirate peut extraire la marque, cela peut être réalisé aisément dans les applications vidéo, où la même marque est ajoutée à un ensemble d'images successives. En effet, lorsque l'on insère une marque identique sur l'ensemble des images, et si on les additionne ensuite, on sait que l'espérance de la valeur nous permet de déterminer la marque. Afin de résoudre ce problème, il suffit d'insérer une autre marque dépendante de la première (cependant cela peut engendrer des artefacts visibles).

Il existe autres attaque comme Attaque par Minimum, Maximum, MinMax et Median.

2.6. Application du tatouage numérique au système de traçage de copies des pirates

L'intégration des flux audiovisuels dans l'Internet constitue aujourd'hui un enjeu technologique majeur, qui tend à rendre la préservation des droits de propriété des contenus indispensable. Cette nécessité a conduit dès 1993-1995 de nombreux chercheurs à se pencher sur le problème de la sécurisation des données numériques face au piratage et à la contrefaçon, par tatouage robuste, afin notamment de faciliter le développement économique des techniques de communication audiovisuelle en réseaux.

Face au piratage numérique, le tatouage robuste offre une solution répondant aux enjeux de la traçabilité des distributions illégales des copies pirates [2].

L'application type est la vidéo à la demande sur Internet (VoD). Un serveur distribue des copies personnalisées d'un même contenu à n utilisateurs (figure 1.8).

Parmi ceux-ci, certains sont malhonnêtes et redistribuent illégalement des copies pirates. Les ayants droits souhaitent connaître l'identité de ces "sources". Pour ce faire, un identifiant unique sous la forme d'une séquence de m bits est cachée d'une façon imperceptible dans chaque vidéo à l'aide d'une technique de tatouage robuste. Ainsi sont produites n copies du contenu, toutes différentes mais pourtant perceptiblement identiques. Cet identifiant permet de tracer la source des copies pirates. Cependant, il se peut que les pirates soient plusieurs, et qu'ils mélangent leurs copies pour brouiller les pistes.

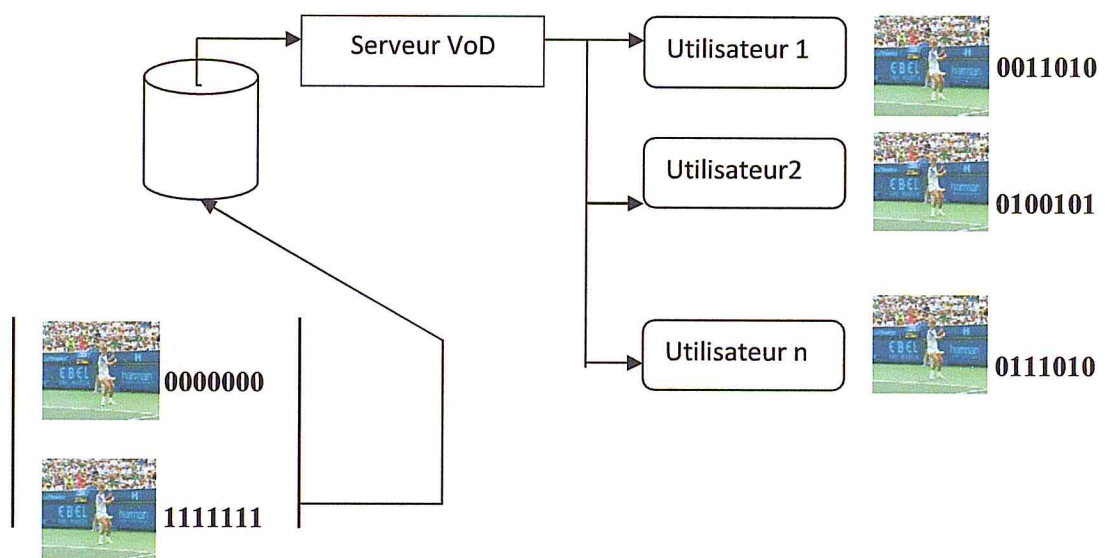


Figure 1.8. Schéma de distribution à n utilisateur via un serveur VoD.

Dans la traçabilité des documents numériques, il est possible de dissimuler l'identité de l'acheteur des documents, afin de remonter à la source d'une vente illicite (figure 1.9). Citons par exemple l'identification, parmi les votants des *Academy awards*, de ceux qui ont divulgué leurs copies personnelles des films en compétition [sw1].

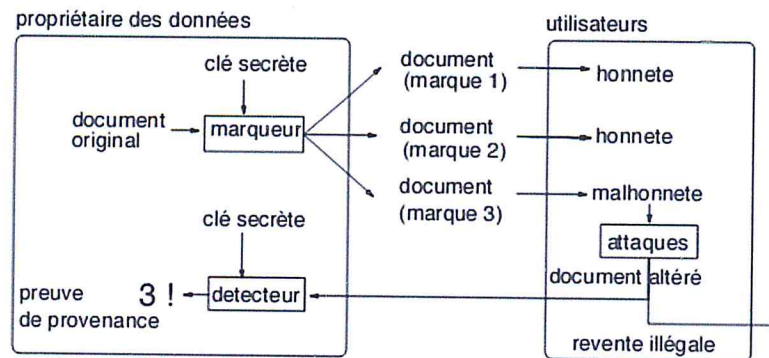


Figure 1.9. Scénario classique de traçabilité

Dans la figure 1.9 le schéma de traçage de copies consiste à dissimuler dans chaque copie distribuée un identifiant unique (numéro de série) afin de garder l'identité de chaque utilisateur. L'insertion de la marque fait appelle au processus de tatouage de documents numérique évoqué. Si toute distribution illégale trouvée par le propriétaire, la source de la fuite est vite trouvée. Mais certains utilisateurs malhonnêtes se réunissent afin de créer une copie pirate à partir de leurs copies légales en utilisant des traitements ciblées. L'identité des utilisateurs malhonnêtes n'est pas simple à identifier, le propriétaire de document numérique récupère la marque piratée de la copie illégale et il trace la provenance de copie en utilisant des processus mathématiques ou cryptographique.

2.6.1. Contraintes des systèmes de traçage de copies des pirates

En plus des contraintes usuelles telles que la robustesse, l'invisibilité, la capacité et la sécurité qui sont imposées dans les systèmes de sécurité basés sur le tatouage numérique, l'application de traçage des copies des pirates exige que le code (marque) à insérer doit être un code anti collusion afin d'éviter toute attaque par collusion.

2.6.1.1. Stratégie d'attaque par collusion

La collusion est une stratégie d'attaque connue depuis un certain temps en cryptographie. Une clique d'utilisateurs malicieux se rassemble et met en commun ses informations/connaissances sur le système de protection, pour générer des données non protégées. Ce type de comportement a été mentionné pour la première fois lorsque des protocoles ont été mis au point pour diviser un secret entre plusieurs individus sans qu'aucun d'entre eux n'ait accès à l'ensemble du secret [12]. Un exemple typique est le partage de secret pour contrôler des actions critiques telles que l'ouverture de la porte

d'un coffre-fort particulier à la banque. Le client et le responsable de la banque ont tous les deux une clé et les deux sont nécessaires pour ouvrir le coffre. Si une partie du secret (clé) manque, la porte du coffre reste fermée. A plus grande échelle, plusieurs clés contenant une partie du secret sont distribuées et il est nécessaire de rassembler au moins k clés différentes pour avoir accès à l'intégralité du secret. Dans ce contexte, les attaquants sont un groupe de n utilisateurs qui cherchent à construire de fausses clés ou à reconstruire l'intégralité du secret quand bien même $n < k$. On retrouve aussi cette problématique de la collusion dans des schémas de distribution dynamique de clés [13] pour les sessions d'audio/vidéo conférences, vidéo à la demande, etc.

En tatouage numérique, les attaques par collusion ont été mentionnées pour la première fois dans le contexte du suivi de copies [14]. Dans ce cas, les fournisseurs de contenus veulent distribuer un faible nombre de contenus à une très large audience. Ils désirent par conséquent avoir les moyens de pister une copie pirate jusqu'à la personne à l'origine de cette fuite. Dans ce but, au lieu de distribuer par exemple le même film à tous les consommateurs, des copies sensiblement différentes sont assignées à chacun d'entre eux. Ainsi, chaque consommateur a une copie unique portant son propre tatouage. Si un utilisateur isolé rend sa copie disponible sur Internet, on peut alors l'identifier en utilisant le tatouage.

Par conséquent, les attaquants (pirates) sont tentés de se regrouper pour combiner leurs différentes copies afin de générer un nouveau document qui ne contiendrait plus de tatouage comme illustré dans la figure 1.10. Il existe principalement deux stratégies de collusion en tatouage:

1. Soit les documents sont analysés pour estimer certaines propriétés du signal de tatouage qui pourraient être utilisés dans un second temps pour retirer
2. Le signal de tatouage, soit les documents sont combinés pour estimer directement le document d'origine non tatoué.

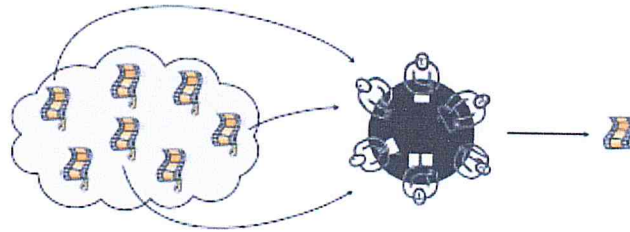


Figure 1.10 Collusion en tatouage numérique: Plusieurs utilisateurs rassemblent plusieurs documents tatoués et les combinent pour produire des documents ne contenant plus aucun tatouage.

Il y a plusieurs types d'attaques de collusion qui peuvent être employées contre des codes de traçabilité de multimédia. La méthode la plus simple consiste à synchroniser les signaux de médias et de faire leur moyenne, qui est un exemple de l'attaque linéaire de collusion. Une autre attaque de collusion, désignée sous le nom de l'attaque de couper coller (copy-and-paste) consiste à former un nouveau signal en collant des parties de chacun de signant des utilisateurs légaux.

D'autres attaques peuvent utiliser des opérations non linéaires, telles que prendre le maximum ou la médiane des valeurs des composants correspondants de différentes copies [2].

2.6.2. Collusion linéaire

La collusion linéaire est l'une des attaques de collusion les plus faisables qui peuvent être utilisées contre des codes de traçabilité. Donnée plusieurs copies différemment marquées du même contenu, les pirates combinent linéairement toutes les copies pour produire une copie étendue. La figure 1.11 montre un exemple de la collusion par l'établissement d'une moyenne à partir des trois empreintes des trois pirates.

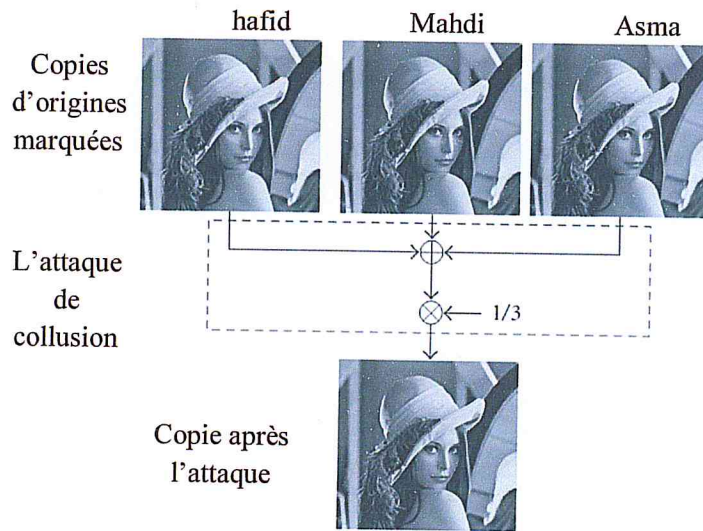


Figure 1.11. Attaque de collusion par la moyenne

La figure 1.12 montre un exemple de l'attaque de couper-coller avec deux pirates.

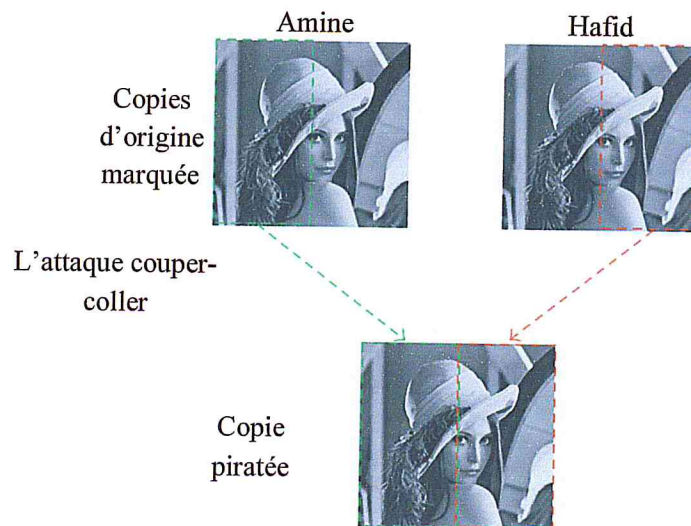


Figure 1.12 Attaque de collusion par couper-coller

2.6.3. Collusion non linéaire

La collusion linéaire en faisant la moyenne est un moyen simple et efficace pour une coalition d'utilisateurs pour atténuer les codes insérés dans le média [2]. Cependant, la moyenne n'est pas la seule forme d'attaque par collusion disponible pour une coalition d'adversaires. En fait, pour chaque composante du média, les pirates peuvent déduire n'importe quelle valeur entre le minimum et le maximum des valeurs

correspondantes, et ils sont confiants que la fausse valeur qu'ils obtiennent sera dans la fourchette « différence juste perceptible » (just-noticeable-different) puisque chaque copie de ses codes devrait avoir une qualité perçue élevée.

Une classe importante d'attaques par collusion non linéaire est basée sur le maximum, minimum, MinMax et la médiane des copies des pirates.

Figure 1.13 montre pour un pixel à la $n^{\text{ième}}$ rangée et à la $m^{\text{ième}}$ colonne dans l'image, les attaquants prennent les valeurs 172, 173, et 176 à partir de chacune de leur copie et produisent une copie pirate en donnant au pixel à la ligne n et à la colonne m le maximum qui correspond à la valeur 176, ou la moyenne des trois pixels qui est de l'ordre de 173.

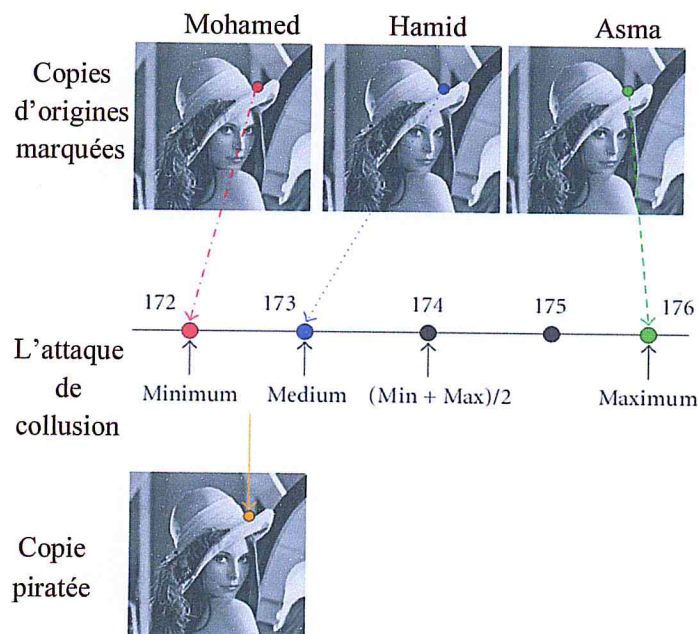


Figure 1.13. Attaque de collusion non linéaire

2. Conclusion

Nous avons présenté dans ce chapitre, une introduction générale au tatouage numérique, avec les définitions de base, les principaux concepts, les différentes applications ainsi que les différentes attaques visant cette discipline.

Dans le prochain chapitre, nous donnerons une présentation générale sur la norme de compression vidéo H.264/AVC.

Chapitre 2

La norme de compression H.264/AVC

1. Introduction

La toute dernière norme de compression vidéo H.264, est appelée à devenir la norme vidéo de référence au cours des prochaines années. Elle a déjà été intégrée avec succès dans des gadgets électroniques tels que les téléphones mobiles et les lecteurs vidéo numériques. Pratiquement, dans toutes les applications utilisant la vidéo, le H.264 offre de nouvelles possibilités en termes de réduction des frais de stockage et de renforcement de l'efficacité globale.

Le H.264 (également connue sous l'appellation MPEG-4 Partie 10/AVC « Advanced Video Coding ») est une norme ouverte sous licence, compatible avec les techniques de compression vidéo les plus efficaces d'aujourd'hui. Un encodeur H.264 peut réduire la taille d'un fichier vidéo numérique de plus de 80 % par rapport à la norme Motion JPEG et de 50 % par rapport à la norme traditionnelle MPEG-4 Partie 2, sans que la qualité d'image ne soit compromise. L'importance de ces gains rend le H.264 extrêmement utile pour les applications vidéo.

2. Fonctionnement du codeur H.264/AVC

Le schéma d'un codeur de type H.264 est présenté sur la **figure 2.1**. L'image à l'entrée F_n du codeur est partitionnée en blocs de pixels de taille 16x16 appelées macroblocs. Ces macroblocs sont alors prédits soit à l'aide de prédictions spatiales (Intra prédiction ou prédiction I) soit à l'aide de prédictions temporelles appelées compensation de mouvement (MC) ou Inter prédiction (prédiction P).

Pour la première image, seule la prédiction spatiale est possible dans la mesure où il n'y a pas d'autres images auxquelles se référer pour la prédiction temporelle. Pour les autres images, on met donc en compétition ces deux modes de prédictions pour chaque macrobloc et on choisit le meilleur mode Inter/Intra. Le résiduel D_n résultant de cette prédiction est ensuite transformé (T) à l'aide d'une DCT puis quantifié (Q) et codé à l'aide d'un codage entropique sans perte.

Notons aussi que ce résiduel est dé-quantifié (Q^{-1}) et dé-transformé (T^{-1}) afin de stocker l'image décodée uF_{n-1} pour l'utiliser lors du codage temporelle des prochaines images à coder. Afin de réduire les effets de bloc, uF_{n-1} est filtré par filtre de déblocage.

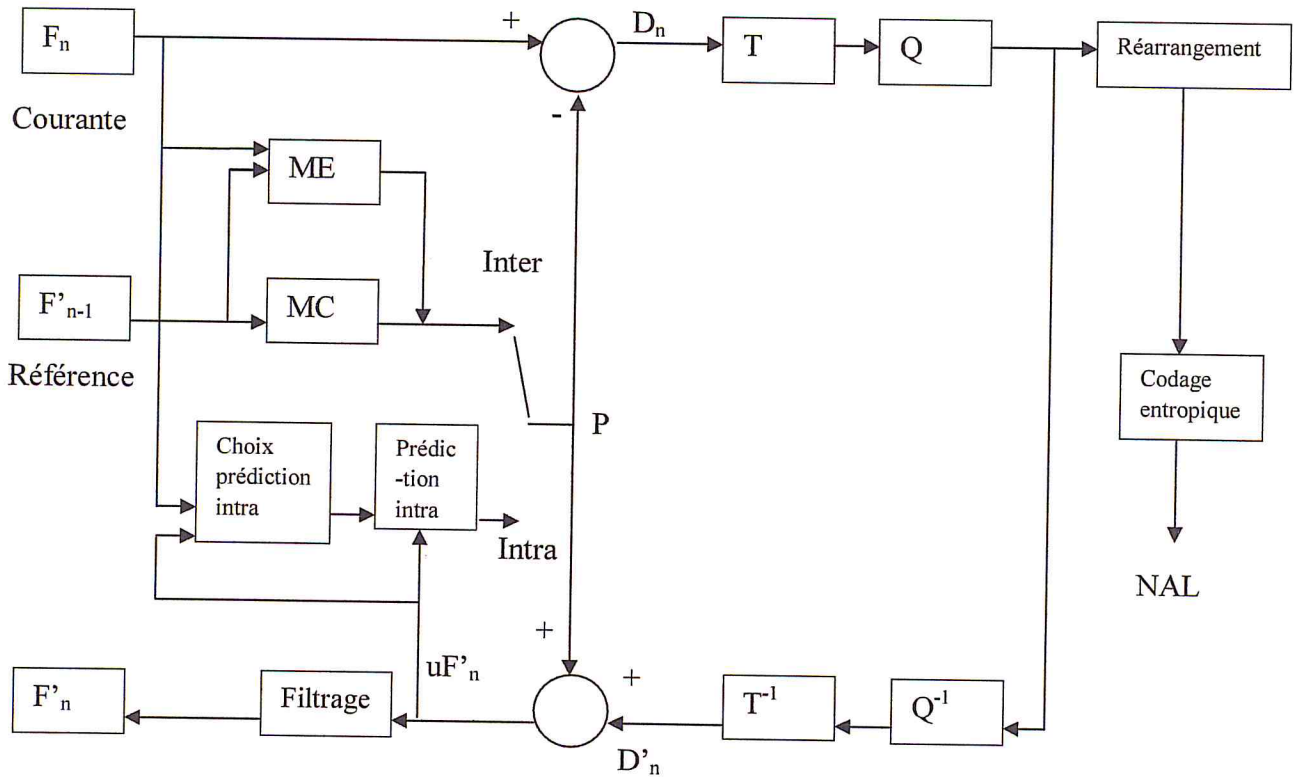


Figure.2.1. Bloc diagramme du codeur

Le même processus de codage est utilisé pour le décodage (Figure.2.2). Les macroblocs fournis par le NAL sont décodés et réarrangés pour obtenir les coefficients X à partir desquels sont obtenus les même D'_n que le codeur. En utilisant les informations contenues dans le bitstream, le décodeur crée une prédiction P , l'ajoute à D'_n et obtient uF'_n . Après filtrage, l'image décodée F'_n est retrouvée.

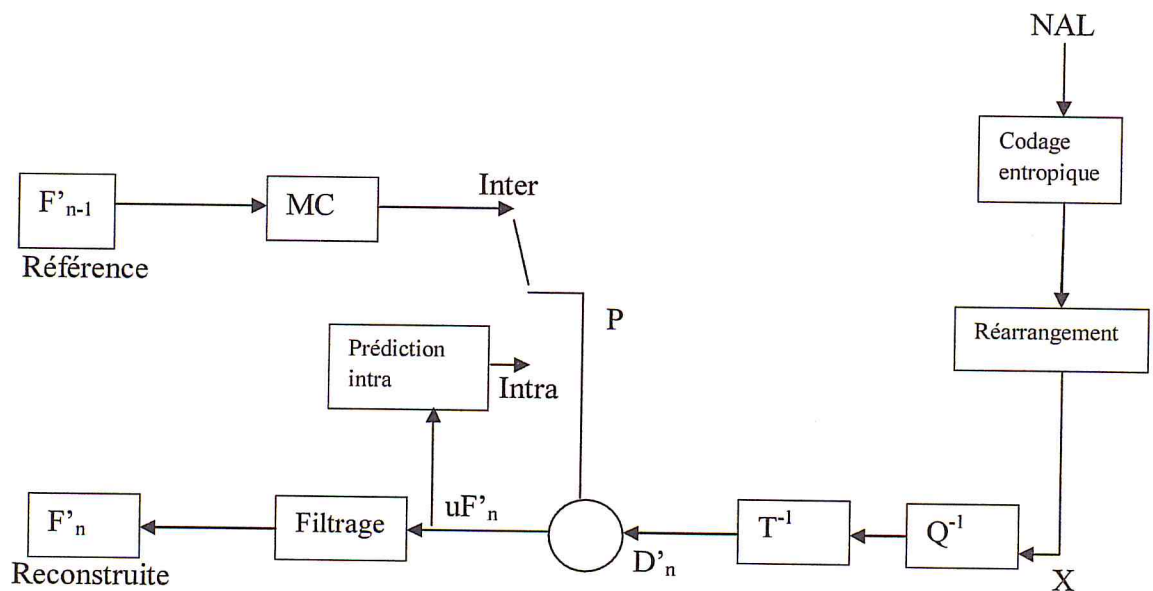


Figure2.2. Bloc diagramme du décodeur

Toutes les notions concernant le codage Intra sont plus précisément détaillées dans la partie suivante.

Pour le codage Inter, retenons qu'il s'agit d'un codage temporel qui utilise la compensation de mouvement pour exploiter les redondances temporelles entre les images. Ainsi, les macroblocs sont prédits à partir d'images dites de référence préalablement codées qui peuvent être antérieures ou postérieures à l'image à coder. Les prédicteurs utilisés sont représentés à l'aide d'un vecteur mouvement et d'une image de référence qui seront codées et envoyées.

De nombreux articles décrivent plus précisément la norme H.264, on citera T. Wiegand et al [15], J. Ostermann et al [16], R. Schâfer et [17].

2.1. Codage Intra dans H.264/AVC

Dans cette partie nous allons décrire précisément le fonctionnement du codage Intra, aussi appelé l'Intra prédiction. Tout d'abord nous définirons ce qu'est un résiduel et comment il est codé puis nous verrons comment il est calculé.

Le résiduel est la différence entre le prédicteur et le macrobloc à encoder. Le prédicteur, noté \hat{p} est calculé à partir de blocs déjà codés se trouvant au-dessus et à gauche du bloc à prédire. On notera e le résiduel et p le pixel à coder, l'expression du résiduel est alors donnée par la *formule 2.1*:

$$e(x, y) = p(x, y) - \hat{p}(x, y) \quad (2.1)$$

C'est ce résiduel e qui est transmis au décodeur qui le décode puis recalcule le prédicteur \hat{p} , on peut alors retrouver le pixel initial p à l'aide de la *formule 2.2* :

$$p(x, y) = \hat{p}(x, y) + e(x, y) \quad (2.2)$$

Afin de saisir l'intérêt de la prédiction, nous allons détailler le codage de ce résiduel e .

2.2. Codage du résiduel

2.2.1. Transformation

H.264/AVC utilise trois transformations selon le type de données résiduelles qui doivent être codées.

La *figure 2.3* montre l'ordre dans lequel les macroblocs sont parcourus. Si un macrobloc est codé en mode intra 4x4, le bloc contenant les composantes DC continues de chaque bloc de luminance 4x4 est labélisé « -1 » et transmis en premier. Ensuite, les blocs résiduels de luminance [0-15] sont transmis dans le même ordre (*figure 2.4.a*). Puis, les blocs contenant les composantes continues de la chrominance (bloc labélisé 16 pour Cb et 17 pour Cr) sont envoyés (*figure 2.4.b*) et (*figure 2.4.c*)

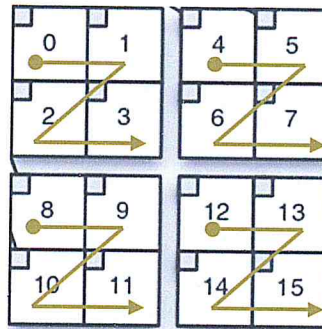


Figure 2.3. l'ordre de parcours d'un macrobloc 4x4

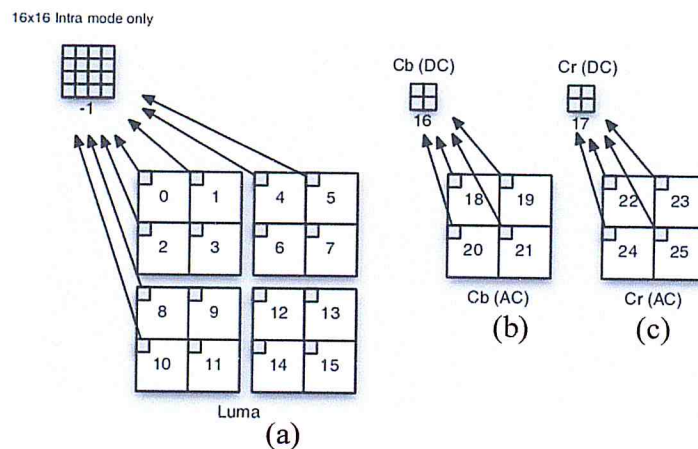


Figure 2.4. L'ordre de balayage des blocs résiduels dans un macrobloc 4x4

2.2.1.1. La transformation DCT 4x4

Cette transformation est appliquée sur les blocs 4x4 de données résiduelles. Le cosinus discret transformant (DCT) opère X, un bloc d'échantillons de 4x4, typiquement échantillons d'image ou valeurs résiduelles après la prédiction, pour créer Y, un bloc de 4x4 de coefficients. L'action du DCT et de son inverse, l'IDCT peut être décrite en termes de matrice A.

Le DCT vers l'avant (FDCT) d'un bloc témoin de $N \times N$ est donné par :

$$Y = A * X * A^T \quad (2.3)$$

Et l'inverse DCT (IDCT) par :

$$X = A^T * Y * A \quad (2.4)$$

Avec X est une matrice des échantillons, Y est une matrice des coefficients et A est une matrice de transformation 4x4. Les éléments de A sont :

$$A = \begin{pmatrix} a & a & a & a \\ b & c & -c & -b \\ a & -a & -a & a \\ c & -b & b & -c \end{pmatrix} \quad (2.5)$$

Avec :

$$a = 1/2$$

$$b = \sqrt{1/2} \cos \pi/8 = 0.6532 \dots$$

$$c = \sqrt{1/2} \cos 3\pi/8 = 0.2706 \dots$$

La matrice Y peut être factorisée à la forme équivalente suivant

$$Y = (C * X * C^T) \otimes E$$

$$= \left(\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & d & -d & 1 \\ 1 & -1 & -1 & 1 \\ d & -1 & 1 & -d \end{pmatrix} (X) \begin{pmatrix} 1 & 1 & 1 & d \\ 1 & d & -1 & -1 \\ 1 & -d & -1 & 1 \\ 1 & 1 & 1 & -d \end{pmatrix} \right) \otimes \begin{pmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{pmatrix} \quad (2.6)$$

$C * X * C^T$ est le cœur de la transformation, E est une matrice des facteurs de graduation et le symbole \otimes indique la multiplication terme à terme et non matriciel. Les constantes a et b sont en tant qu'avant; et d est approximé par $1/2$.

2.2.1.2. La transformation de Hadamard 4x4

Si un macrobloc est codé en mode intra 16 x16, chaque bloc résiduel est d'abord transformé par la transformation DCT 4x4. Les DC (coefficients continus des blocs)

sont placés dans une matrice 4×4 et transformés par une transformation de Hadamard 4×4 tel que :

$$Y = \left(\left(\begin{matrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{matrix} \right) (W) \left(\begin{matrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{matrix} \right) \right) \quad (2.7)$$

W est le bloc 4×4 de coefficients DC et Y est le bloc après transformation.

Au décodeur, la transformation inverse de Hadamard est donnée par :

$$W = \left(\left(\begin{matrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{matrix} \right) (Z) \left(\begin{matrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{matrix} \right) \right) \quad (2.8)$$

Avec Z la matrice de quantification.

2.2.2. La quantification

H.264/AVC utilise une quantification scalaire dont l'opération de base est :

$$Z_{ij} = \text{round} \left(\frac{Y_{ij}}{Q_{step}} \right) \quad (2.9)$$

Avec Y_{ij} le coefficient de la transformation, Q_{step} la taille du pas de quantification et Z_{ij} le coefficient quantifié. Un total de cinquante-deux (52) valeurs de Q_{step} sont supportés par la norme, et indexées par un paramètre de quantification QP.

Les valeurs de Q_{step} correspondant à chaque QP sont montrées dans le **tableau 2.1** suivant :

Tableau 2.1. Taille du pas de quantification dans le codec H.264

QP	0	1	2	3	4	5	6	7	8	9	10	11	12	...
Q_{step}	0,625	0,6875	0,8125	0,875	1	1,125	1,25	1,375	1,625	1,75	2	2,25	2,5	
QP	...	18	...	24	...	30	...	36	...	42	...	48	...	51
Q_{step}		5		10		20		40		80		160		224

Q_{step} double pour chaque incrément de 6 dans QP et augmente de 12,5% pour chaque incrément de 1 dans QP. Les valeurs de QP peuvent être différentes pour la

luminance et la chrominance. Les deux paramètres varient entre 0 et 51 mais le paramètre Qp de la chrominance est dérivé de paramètre QP de luminance.

L'opération de base de quantification inverse est donné par :

$$Y'_{ij} = Z_{ij} \cdot Q_{step} \quad (2.10)$$

2.2.3. Le balayage de bloc

Les coefficients transformés des blocs sont balayés, c.-à-d. converti en coefficients quantifiés, avant le codage d'entropie. L'ordre de balayage est prévu pour grouper l'ensemble des coefficients significatifs, c.-à-d. coefficients quantifiés différents de zéro. Des coefficients différents de zéro tendent à être groupés autour de DC gauche supérieur. Dans ce cas-ci, un ordre de balayage de zigzag peut être le plus efficace, représenté sur la *figure 2.5*, des blocs du 8×8 et de 4×4. Après balayage

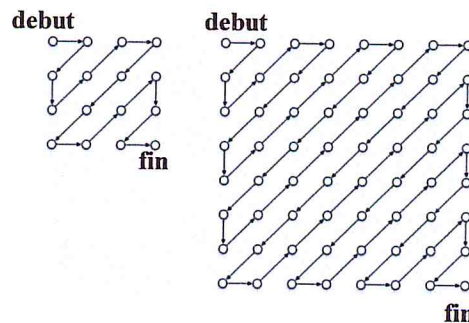


Figure 2.5. Balayage progressif pour les blocs 4×4 et 8×8

2.2.4. Le codage entropique

Après la quantification, il faut coder le résiduel. Dans H.264, deux méthodes de codage entropique sont proposées :

- Le Context Adaptive Variable Length Coding (CAVLC) qui est une méthode de faible complexité basée sur l'utilisation d'un codage de taille différentes suivant le contexte, chaque composante (résiduel, mode, vecteur mouvement,..) à coder dispose ainsi de son propre contexte.
- Le Context Adaptive Binary Arithmetic Coding (CABAC) qui est un codage arithmétique plus complexe que CAVLC, plus d'information sur ce codage qui est une nouveauté très importante de H.264 est dans l'article [18].

A la suite du codage entropique, les coefficients sont transmis dans un seul flux (bitstream) avec les autres informations éventuelles pour une prédiction Intra ou Inter image (mode, vecteur mouvement..).

2.2.5. Calcul du résiduel

Le codage Intra est un codage spatial contrairement au codage Inter qui est un codage temporel. C'est à dire que pour prédire un bloc on utilise seulement les blocs de l'image courante qui ont déjà été codées.

Dans H.264, pour la luminance, trois tailles de blocs ont été définies : 16x16, 8x8, 4x4. Il y a quatre prédictions possibles pour les blocs de tailles 16x16 et neuf pour les blocs de tailles 8x8 et 4x4.

2.2.6. La prédiction des blocs Intra 4x4

Pour la prédiction en mode Intra, les standards précédents travaillaient généralement avec des blocs de tailles 8x8. C'est donc une nouveauté de descendre jusqu'à des blocs de taille 4x4, ce qui engendre bien évidemment, une augmentation de la quantité de calculs.

Neuf modes de prédiction différents sont utilisés dans la norme H.264/AVC. Un mode DC, telle que tous les échantillons du bloc courant 4x4 sont prédits par le moyen de tous les échantillons voisins à gauche et juste au-dessus du bloc courant qui ont été déjà reconstruits. En plus de ce mode, huit modes de prédiction dont chacun a une direction spécifique.

La **figure 2.6** présente la manière dont les pixels sont labellisés et la figure 2.7 les directions de huit modes associés.

M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

Figure 2.6. Labellisation des échantillons de prédiction 4x4

Avec a, b, c, p sont des pixels du bloc courant, et M, A, B, L sont des pixels des blocs voisins.

Le mode 0 (prédiction verticale) et le mode 1 (prédiction horizontale) sont montrés explicitement sur la **figure 2.7**. Par exemple, si le mode de prédiction verticale

est appliqué, tous les échantillons au-dessous de l'échantillon A sont prédits par l'échantillon A.

Tous les échantillons ci-dessous d'échantillon B sont prédits par l'échantillon B et ainsi de suite. Les échantillons des autres modes de prédiction sont calculés à partir des échantillons de A à M. Si les pixels voisins manquent (cas du premier bloc de la trame), des valeurs par défaut sont utilisées [19].

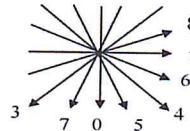


Figure 2.7. Les modes de prédiction intra

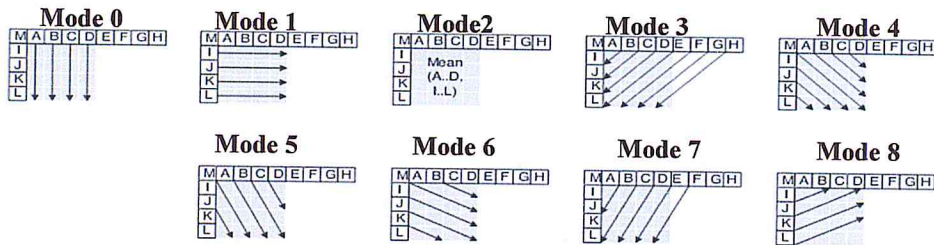


Figure 2.8. Modes de prédiction des blocs 4x4 de luminance

Les flèches indiquent la direction de la prédiction de chaque mode *Figure 2.8*. Pour les modes de 3 à 8, les pixels prédits sont calculés par une formule appliquée aux pixels [A-M]. Une fois calculés, les neuf modes sont évalués par une Somme de Différence Absolue (SAD) donnée par l'équation 2.11 . Le mode fournissant la valeur minimale du SAD est retenu pour la prédiction du bloc.

$$SAD(u, v) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} |C_{i,j} - R_{i+u,j+v}| \quad (2.11)$$

Où $C_{i,j}$ est la valeur du pixel de la trame courante

$R_{i+u,j+v}$ Celle du bloc de référence.

Les composantes du vecteur de déplacement sont (u, v).

2.2.7. La prédiction des blocs Intra 16x16

La prédiction par blocs 4x4 demande des calculs importants et n'est pas toujours justifiée pour des régions de faible variation. C'est pourquoi la prédiction intra image peut se faire par blocs 16x16. Cette alternative à l'avantage d'être évidemment plus

rapide et moins coûteuse. Quatre modes de prédiction sont présentés dans la norme H.264/AVC dont les trois premiers (mode 0, mode 1 et mode 2) sont semblables aux modes de prédiction Intra 4×4 et un mode plan (mode 3) tel que la prédiction est une fonction linéaire entre les échantillons voisins gauche et au-dessus afin de prédire les échantillons courants (les échantillons sont calculés par moyenne des valeurs obliques des deux sens de direction). Les quatre modes sont représentés sur la *figure 2.9* [19].

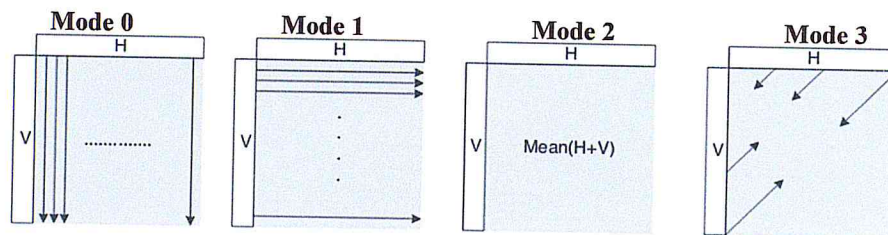


Figure 2.9. Modes de prédiction des blocs 16x16 de luminance

2.2.8. Le filtre de déblocage

Un défaut du codage axé sur le bloc est la visibilité de la structure en blocs. Les bords sont en général reconstitués avec moins de précision que les pixels intérieurs, la pixellisation est l'un des artefacts les plus visibles des méthodes de compression actuelles. Pour cette raison, la norme H.264/AVC définit un filtre de «déblocage» adaptatif en boucle où la puissance du filtrage est contrôlée par les valeurs de plusieurs éléments syntaxiques. La pixellisation est réduite sans affecter outre mesure la clarté du contenu et la qualité subjective est considérablement améliorée.

En même temps, le filtre réduit le débit binaire de 5 à 10% tout en produisant la même qualité objective que la vidéo non filtrée. Plusieurs paramètres, seuils et également des caractéristiques locales de l'image elle-même contrôlent la force du processus de filtrage.

La compression d'une séquence vidéo par la norme H.264 suivant les différents types du codage entropique produit un flux vidéo compressé qu'on appelle le *bitstream*.

3. Les profils

Un profil définit un ensemble d'outils de codage ou d'algorithme qui peuvent être utilisés pour générer un flux compatible. Tous les décodeurs conformes à un profil

spécifique doivent prendre en charge toutes les fonctionnalités de ce profil. Les codeurs ne sont pas tenus d'utiliser un ensemble particulier de fonctionnalités prises en charge dans un profil, mais doivent fournir des flux compatibles. Le standard H.264/AVC définit sept profils chacun supportant un ensemble spécifique de fonctions de codage et spécifiant la conformité d'un couple codeur/décodeur (**Tableau 2.2**) [sw2] :

- Le profil de base (Baseline profile) : Ce profil vise typiquement des applications avec une résolution réduite, comportant seulement des images I et P, avec une précision de compensation au 1/4 de pixel. ce profil est très utilisé dans les applications mobiles et de visioconférence.
- Le profil principal (Main profile) : Inclus le module de prédiction, les trames I, le codage inter utilisant des trames P et B, le codage entropique CAVLC et CABAC. Ce profil permet typiquement la meilleure qualité au coût d'une complexité plus élevée (essentiellement due aux trames B et au CABAC).
- Le profil étendu (Extended profile) : Inclus tous les outils de la norme H.264/AVC comme le profil de base excepté le CABAC, il inclut aussi la possibilité de changer de flux entre les bitstreams codés (slices SI et SP) et améliore la résilience d'erreur (Data partitioning).
- Le profil élevé (high profile) : Pour la diffusion et le stockage sur disque, en particulier pour la télévision haute définition (ce profil a été adopté pour les disques HD DVD et Blu-ray ainsi que pour la télévision numérique haute définition).
- Le profil élevé 10 (High 10 profile) : Ce profil va au-delà des applications grand public et s'appuie sur le profil High ajoutant jusqu'à 10 bits de précision par pixel.
- Le profil élevé 4 :2 :2 (High 4 :2 :2 profile) : Le profil principal pour les applications professionnelles, il s'appuie sur le profil High 10 ajoutant le support pour la quantification 4:2:2 jusqu'à 10 bits par pixel.
- Le profil élevé 4 :4 :4 (High 4 :4 :4 profile) : Ce profil s'appuie sur le profil High 4:2:2 en ajoutant le support pour la quantification 4:4:4, jusqu'à 12 bits par pixel et en plus le support pour un mode sans perte efficace.

Tableau2.2. Les profils de standard H.264/AVC

Description	Base	Etendu	Principal	Elevé	Elevé 10	Elevé 4 :2 :2	Elevé 4 :4 :4
Tranches I et P	✓	✓	✓	✓	✓	✓	✓
Tranches B	✗	✓	✓	✓	✓	✓	✓
Tranches SI et SP	✗	✓	✗	✗	✗	✗	✗
Image de Références Multiples	✓	✓	✓	✓	✓	✓	✓
Filtre anti-blocs	✓	✓	✓	✓	✓	✓	✓
Codage CAVLC	✓	✓	✓	✓	✓	✓	✓
Codage CABAC			✓	✓	✓	✓	✓
Ordonnancement flexible des macroblocs (FMO)	✓	✓	✗	✗	✗	✗	✗
Ordonnancement arbitraire des tranches (ASO)	✓	✓	✗	✗	✗	✗	✗
Format 4:2:0	✓	✓	✓	✓	✓	✓	✓
Format monochrome (4:0:0)	✗	✗	✗	✓	✓	✓	✓
Format 4:2:2	✗	✗	✗	✗	✗	✓	✓
Format 4:4:4	✗	✗	✗		✗	✗	✓
Pixel 8 Bit	✓	✓	✓	✓	✓	✓	✓
pixel 9 et 10 Bit	✗	✗	✗	✗	✓	✓	✓
pixel 11 et 12 Bit	✗	✗	✗	✗	✗	✗	✓

4. Conclusion

Nous venons de présenter dans ce chapitre le principe de fonctionnement du codeur H.264/AVC afin de comprendre le processus de codage et les traitements utilisés afin de pouvoir déterminer les positions d'insertion.

Le prochain chapitre, nous donnerons les principaux codes de traçabilité utilisés pour l'application de traçage des vidéos piratés.

Chapitre 3

Traçage des copies de pirates

1. Introduction

L'apparition de la télévision en 1952 a donné naissance à la diffusion du signal vidéo. Ce dernier connaît une grande évolution en passant du signal analogique au signal numérique. Les lecteurs DVD, la télévision numérique ou par satellite, la transmission vidéo par Internet, tous ces exemples apparus dans le contexte de la diffusion payante nécessitant une protection. La communauté cryptographique est la première à étudier la problématique de diffusion payante [20], en transmettant à un ensemble d'utilisateurs un contenu chiffré, via un canal public (internet, CD-ROM, réseau câble, etc...) de telle sorte que seuls les utilisateurs autorisés puissent déchiffrer ce contenu. Hélas, cette technologie a connu des limites devant l'évolution de l'ère numérique qui a déclenché une facilité de piratage et de partage des documents numériques [sw3] : parmi les utilisateurs figurent certains ayant des attentions malveillantes, ils donnent à d'autres utilisateurs non autorisés la possibilité de bénéficier de ces contenus chiffrés. Le traçage des copies pirates de ces utilisateurs (traçage des traitres) est une solution pour lutter contre ce problème [20].

Dans ce chapitre, nous allons donner un bref aperçu des diverses approches liées au traçage des traitres, les contributions et les solutions apportées par les trois diverses communautés ; la cryptographie, les statistiques et le tatouage numérique.

2. Traçage des copies de pirates par l'approche cryptographique

La communauté cryptographique est la première à étudier le sujet [21]. La problématique de traçage est similaire à un problème de gestion de clés secrètes. Imaginons que les utilisateurs malhonnêtes « cassent » leurs décodeurs afin de trouver la clé secrète, et créent des décodeurs pirates. Si tous les utilisateurs ont une clé propre et unique, il est facile à partir d'un décodeur pirate de retrouver l'identité des traitres, en revanche le contenu distribué diffusé aux abonnés est chiffré et transmis n fois. A l'inverse, si tous les utilisateurs partagent la même clé, le contenu est chiffré et transmis qu'une seule fois, mais il est impossible d'identifier les traitres. Chor et al. [20] proposent des schémas attribuant un jeu de clé propre à chaque décodeur afin de minimiser le nombre de chiffrements et de transmissions. Par conséquent permettre de retrouver les traitres. De nos jours, avec l'évolution de l'ère numérique, et les bandes passantes qui augmentent de plus en plus et devient de moins en moins chers, les pirates

pensent à décrypter le contenu et de le retransmettre plutôt que de fabriquer des décodeurs pirates. D'où un saut du traçage de décodeurs au traçage de contenus.

Les auteurs dans l'article [20] ont introduit le concept de la collusion : cette dernière est une stratégie bien connue dont le principe se réfère essentiellement à un ensemble de clients malveillants qui rassemblent leurs connaissances individuelles sur la protection du système, afin d'obtenir un contenu multimédia non protégé.

Boneh et Shaw sont les premiers à faire le lien avec le problème soulevé dans l'article [20], en définissant un modèle mathématique de la collusion connu sous le nom *marking assumption*. Ce dernier est défini par [8] :

- Un contenu qui est constitué d'une suite de symboles, par exemple un alphabet binaire.
- Dans ce contenu, il y a m positions peu importants dont la modification de leurs contenus n'affecte pas significativement le contenu global.
- Le mot de code est une séquence binaire de longueur m identifiant un utilisateur, sera cachée dans le contenu aux emplacements sélectionnés.
- L'emplacement du mot de code n'est pas connu par les pirates, c'est en comparant leurs copies qu'ils distinguent des différences dévoilant certains emplacements.
- Ils créent une copie pirate en mélangeant symbole par symbole de leurs copies.
- Le processus d'accusation connaît ces emplacements et extrait de la copie pirate une séquence pirate de m symboles.

Ainsi, la règle de ce modèle est que là où les mots de code des pirates ont tous le même symbole, ce dernier se retrouve forcément dans la séquence pirate.

Parmi les 2^m séquences possibles, le code n'en retient que n . La collusion C est l'ensemble des c mots de code des pirates. Une notion utile est l'ensemble des descendants, $desc(C)$ qui est l'ensemble de toutes les séquences pirates réalisables à partir des c mots de code en suivant les règles de la *marking assumption*. En binaire, cet ensemble est de taille $2^{m'}$, où m' est le nombre d'emplacements dans les mots de code des pirates où leurs c symboles ne sont pas tous égaux.

La raison pour laquelle, une terminologie des codes **anti-collusion** s'est mise en place [22]. Elle classe les codes en quatre catégories suivant certaines propriétés : "frameproof", "secure frameproof", "identifiable parents property" (IPP) et "traceable".

- *Frameproof* : To frame en anglais veut dire produire des fausses preuves pour qu'innocent soit accusé à tort. Un code est *c-frameproof* s'il est impossible pour une collusion de taille au plus c de recréer le mot de code d'un autre innocent :

$$desc(C) \cap X = C \tag{3.1}$$

- *Secure frameproof* : un code est *c-secure frameproof* si aucune collusion de taille c ne peut créer une séquence pirate qu'un autre groupe de c personnes aurai pu créer :

$$y \in desc(C) \cap desc(C') \Rightarrow C \cap C' \neq \emptyset \tag{3.2}$$

L'accusation se produit comme suit : Si la séquence pirate y appartient à deux ensembles de descendance $desc(C)$ et $desc(C')$, on ne peut pas décider laquelle des collusions la produit. Cependant, cette propriété assure que les deux collusions C et C' ont une intersection non vide. De cette manière, on accuse le ou les utilisateurs communs. Mais avec cette catégorie de code, un autre problème surgit (figure 3.1):

$$y \in desc(C) \cap desc(C') \cap desc(C'') \quad \text{Alors que} \quad C \cap C' \cap C'' = \emptyset \tag{3.3}$$

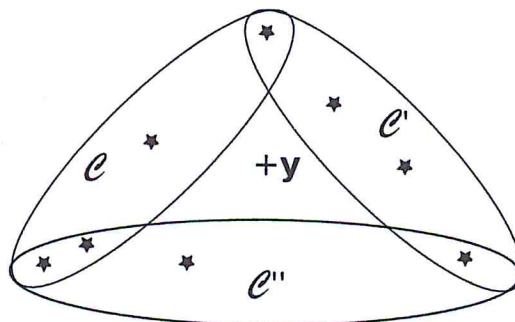


Figure 3.1 : La séquence pirate y appartient aux ensembles de descendance de trois collusions C , C' et C''

Les mots de code sont figurés sous forme d'étoiles. Qui accusez-vous?

Pour y remédier, il faut une propriété encore plus remarquable.

- *identifiable parents property* (IPP) et « *traceable* » : la traçabilité forte donnée par les code IPP ou « *traceable* » assure une accusation rigoureuse. Un code est *c*-« *traceable* » si le mot de code le plus proche au sens de Hamming de la séquence pirate est celui d'un traître.

Cependant la traçabilité forte est une propriété contraignante; par exemple si $c > 2$, il faut de grand alphabet q -aire ($q > c^2$) et de très longue séquences. Par contre la traçabilité faible propose un changement de stratégie en admettant que l'accusation peut donner des erreurs.

Le théorème le plus connu en traçabilité forte est le théorème des *codes correcteurs d'erreur* [20].

2.1. Principe du code correcteur d'erreur

Si X est un code correcteur d'erreur (n mot de code de longueur m), de distance minimale $d > m(1 - c^{-2})$ alors X est un code *c*-*traceable*.

Considérons le mot de code X d'un traître donné, en mélangeant leurs mots de code ils forment une séquence pirate Y qui peut s'écrire comme $Y = X + e$ ou e est l'erreur commise.

Le mot de code d'un traître est celui qui a le plus d'erreurs. L'algorithme de décodage du code correcteur d'erreur enlève les erreurs et retrouve le mot de code X du traître. Si les traîtres partagent les risques, il y a au plus $m(1 - c^{-1})$ erreurs. La condition sur la distance minimale assure qu'aucun innocent est plus proche de Y .

Cependant, décoder autant d'erreurs n'est pas facile pour tous les codes correcteurs. Il faut employer de codes redondants ou des concaténations de code produisant des mots de code très longs.

3. Traçage de copie de pirates par l'approche statistique

Devant la complexité énorme des codes à traçabilité forte, les cryptographes ont relâché les contraintes et toléré des erreurs d'accusation. C'est la traçabilité faible, il y a deux types d'erreurs :

- La probabilité ε_1 d'accuser à tort des innocents.
- La probabilité ε_2 de rater des pirates.

Le code est utile si on sait borner ces erreurs et si les bornes sont faibles, la probabilité ε_1 est la plus critique, pratiquement de l'ordre de 10^{-6} , et la probabilité ε_2 est beaucoup plus grande, de l'ordre de 10^{-1} [21].

Le code le plus connu pour la traçabilité par l'approche statistique est le code de Tardos [23].

3.1. Code de Tardos

Gabor Tardos est un maître en probabilité, statistique et calculs combinatoires. En 2003 un de ses collègues lui expose la problématique de traçage de traitre. Spécialement ; un résultat non-constructif de la communauté cryptographique [24] : la borne inférieure la plus fine sur la longueur d'un code binaire est en :

$$\theta (c^2 \log \varepsilon_1 n^{-1}) \tag{3.4}$$

Avec c est le nombre possible de pirate, n est le nombre total d'utilisateur et ε_1 est la probabilité d'erreur.

Tardos est le premier à exhiber un code qui peut atteindre cette borne inférieure. Il publie dans une conférence en 2003 [23] [sw4]. Deux ans après, Philipse et al. [25] adoptent les résultats de Tardos.

3.1.1. Principe de Tardos

Tardos n'a jamais donné son raisonnement, Il construit un tableau de n lignes (nombre de suspects qui est le nombre d'utilisateurs) et m colonnes (nombre de réponses aux questions posées). L'idée est de conserver un tableau de traçage de traitre : si le $j^{\text{ième}}$ utilisateur a le même symbole que celui présent dans la séquence pirate à l'emplacement i alors la case (j, i) du tableau reçoit un '1' sinon '-1'. Autrement dit, l'utilisateur qui a le même symbole que la séquence pirate est plus proche d'être un pirate et un différent symbole signifie l'innocence de l'utilisateur. Cependant, l'accusation de piraterie ne peut pas se faire en reposant sur un seul symbole, donc il est nécessaire de calculer les scores.

Soit l'exemple suivant qui a comme séquence pirate de longueur $m=5$

$$Y = 0 1 1 0 1$$

Soit le nombre d'utilisateur $n= 4$ chacun possède une séquence binaire unique figurant dans le tableau 3.1.

Tableau 3.1. Matrice de code pour 4 utilisateurs

utilisateur	i=0	i=1	i=2	i= 3	i=4
Utilisateur1	0	0	1	0	1
Utilisateur2	0	1	0	1	1
Utilisateur3	1	0	1	1	1
Utilisateur4	1	0	0	0	0

Le processus d'accusation repose sur la création du tableau 3.2 de traitre de taille 4 lignes et de 5 colonnes.

Tableau 3.2. Tableau de traitre

	Y ₀ =0	Y ₁ =1	Y ₂ =1	Y ₃ =0	Y ₄ =1	SOMME
Utilisateur1	+1	-1	+1	+1	+1	+3
Utilisateur2	+1	+1	-1	-1	+1	+1
Utilisateur3	-1	-1	+1	-1	+1	-1
Utilisateur4	-1	-1	-1	+1	-1	-3

En calculant la somme, l'utilisateur qui a le plus grand score est accusé de trahison (utilisateur 1 dans l'exemple) et le pirate est poursuivi.

3.1.2. Propriétés

- Si m est relativement grand, le score d'un innocent est distribué comme une Gaussienne centrée sur zéro de variance m alors que le score de coupable est distribué comme Gaussienne centré sur $2m/c\pi$ de variance $\approx m$. Cela veut dire que plus la valeur de m est grande où plus la valeur de c est petite, plus les deux distributions sont éloignées : et ainsi la distinction des pirates des innocents est évidente.

- La création du code est souple. L'ajout d'un nouvel utilisateur se fait en générant simplement un nouveau mot de code avec une longueur m de l'ordre :

$$m \approx 20 c^2 \log(1/\varepsilon_1) \quad (3.5)$$

- L'accusation est flexible. Après le calcul des scores. Le pirate est l'utilisateur qui a le plus grand score, si on veut minimiser le risque d'accuser un innocent. Ou alors, pour assurer plus d'un pirate, on calcule un seuil Z dépendant de ε , tel que l'utilisateur est accusé si son score est supérieur à Z .
- L'inconvénient du code de Tardos est son accusation exhaustive. Il est facile de savoir si un utilisateur est coupable ou non. Par contre pour chercher les coupables, il faut calculer tous les scores, soit une accusation de $O(n)$.

4. Traçage des copies de pirates par l'approche de tatouage numérique

La communauté de tatouage numérique ou « fingerprinting » est une communauté de traitement du signal, qui modélise un contenu multimédia (vidéo, un clip audio etc.) par une séquence binaire où certains bits sont flippés. La modélisation du problème par les cryptographies ou les mathématiciens par la modification de certain bit peut provoquer une dégradation en niveau de la qualité de la vidéo, pour cela, le système élaboré pour la traçabilité doit assurer l'invisibilité et la robustesse des mots de code insérer dans les séquences vidéo.

Une contrainte liée à l'application de tatouage est le tatouage de vidéo diffusée sur internet via un serveur. Ce dernier comporte une grande complexité parce que les fichiers sont diffusés plusieurs fois en même temps, par contre, cette contrainte n'est pas valide dans le système de diffusion via un support de stockage.

En revanche, certains utilisateurs ont des attentions malhonnêtes, en créant des copies pirates à partir de leurs copies d'origine, en utilisant des attaques.

Il existe 3 types d'attaque en multimédia qui sont :

- L'échange des blocs.
- La fusion des blocs qui est une attaque critique qui a pour but de créer des blocs pirates à partir des blocs de copie d'origine.
- Post-traitement sont les traitements bienveillante comme le filtrage ou la compression, ces attaques n'ont pas but de détruire la marque ou de déjouer système de traçage.

Pour cela, la technique de tatouage choisit pour le système de traçabilité doit satisfaire certain critère.

5. Conclusion

Le traçage des traitres a connu une évolution ces dernières années, en passant de la traçabilité forte à la traçabilité faible. Dans ce chapitre, nous avons éclairé ces deux approches en expliquant le principe général des codes de traçabilités pour chaque approche, en arrivant à l'approche de tatouage numérique.

Dans le prochain chapitre, nous allons exposer la stratégie de traçage des copies de pirates des vidéos compressées par le codec H.264/AVC.

Chapitre 4

Implémentation et réalisation

1. Introduction

Dans les précédents chapitres nous avons présenté une vue d'ensemble de ma technologie de tatouage numérique, le standard de compression vidéo H.264/AVC et les différentes approches dédiées au traçage des traites. Dans ce chapitre nous appliquons ces connaissances pour développer une méthode permettant de sécuriser des vidéos distribuer à la demande ou graver et distribuer sur des supports tel que les DVD. Solution permettant de détecter les sources de redistribution illégales en utilisant le tatouage numérique et le code de Tardos comme un code de traçabilité.

Un exemple d'application de tatouage numérique de la vidéo distribuée est le film Harry Potter 7 et les Reliques de la Mort qui est victime de son succès avec le téléchargement illégal proposé sur des serveurs de partage de fichiers comme Rapidshare, ou Megaupload, même dans les applications peer to peer qui sont souvent des fausses vidéos en streaming. Cependant, le vrai Harry Potter 7 épisode 1 de la Warner Bros dont les 36 premières minutes du film circule sur internet, possède un tatouage numérique. Cette empreinte numérique sur le film Harry Potter 7 permet à la Warner Bros d'identifier le pirate qui a partagé ce nouvel épisode d'Harry.[sw5]

2. Etat de l'art de tatouage numérique dans H.264/AVC

Différentes techniques de tatouage ont été proposées pour les codecs vidéo, mais peu de travaux ont été effectués pour la dernière norme de compression vidéo H.264/AVC. En terme de domaine dans lequel la marque est insérée, les techniques de tatouage dans la norme H.264/AVC peuvent être classées selon trois domaines d'insertion (figure 4.1) : domaine spatial (pixel), domaine transformé (au cours de la compression) et domaine compressé (tatouage dans le flux compressé).

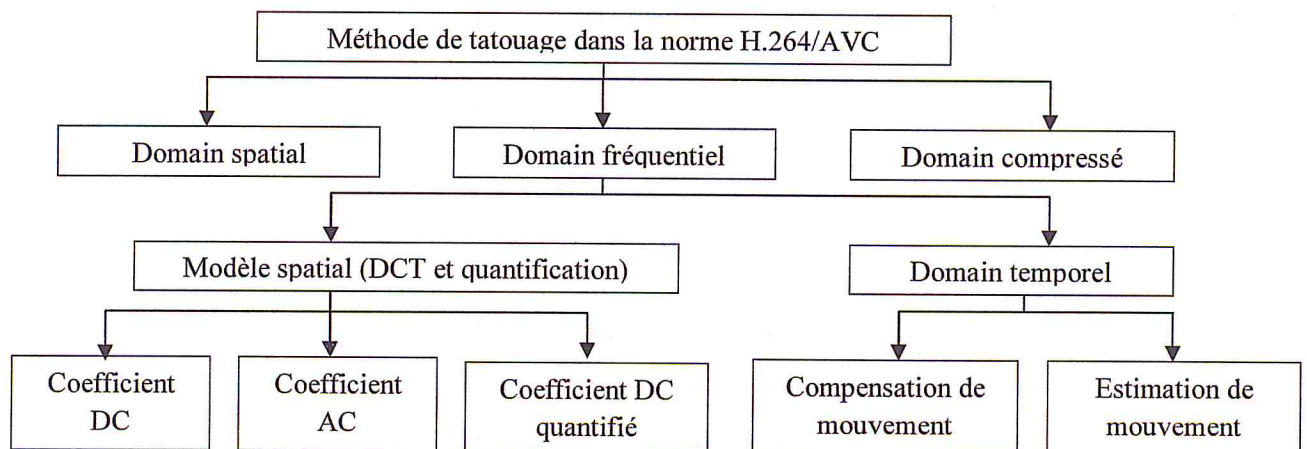


Figure 4.1. Classification des méthodes de tatouage dans la norme

Quelques travaux récemment édités se sont concentrés sur l'insertion de la marque aux cours de la compression. Deux classes d'insertion sont possibles dans ce cas : insertion d'une marque robuste dans le modèle spatial [26] (les opérations de la transformation et la quantification) pour l'application de copyright et l'insertion d'une marque fragile dans les vecteurs de mouvements pour l'application l'authentification.

Les méthodes de tatouage dans le domaine spatial sont à leur tour classifiées selon les modules d'insertion [26] : l'insertion peut se faire après la transformation DCT 4x4 en utilisant les coefficients de moyennes fréquences AC ou bien les coefficients continus DC [26]. Elle peut se faire aussi après l'application de la quantification des coefficients.

Le domaine temporel regroupant les modules de compensation de mouvement et l'estimation de mouvement de la norme H.264 est pris comme domaine d'insertion surtout pour authentifier le contenu vidéo. Principalement, deux technologies ont été utilisées: le tatouage numérique [27][28] et la signature numérique externe [26][29]. Les techniques de tatouage numérique ont été proposées pour assurer un service d'intégrité. Elles sont basées sur l'utilisation d'un tatouage fragile, par opposition au tatouage robuste classiquement utilisé pour la protection des droits d'auteur. Le principe de ces approches est d'insérer une marque (généralement prédéfini et indépendant des données à protéger) dans la vidéo d'origine de telle manière que les moindres modifications apportées à la vidéo se répercutent également sur la marque insérée. Pour vérifier l'intégrité de la vidéo, il suffit alors de vérifier localement la présence de cette marque.

Pour l'application de traçage des copies des pirates une seule méthode basée sur le tatouage numérique est publié dans la littérature c'est celle de Shahid et al. [30]. Ces derniers ont proposé une approche basée sur l'insertion du code de Tardos dans la vidéo au cours de processus de compression H.264/AVC. Plus précisément ce code est caché dans les coefficients transformés quantifiés.

3. Système de traçage proposé et développé

Le système de traçage des copies de pirates développé entre dans le contexte de la diffusion à la carte de document multimédia, e.g vidéo à la demande : chaque utilisateur reçoit une version personnalisée de la vidéo compressé par H.264/AVC contenant un identifiant personnel inséré grâce à une technique de tatouage robuste. Ainsi si une

copie est rediffusée telle quelle de manière illégale, on est en mesure de remonter à l'utilisateur malhonnête.

Les marques insérées dans les vidéos diffusées sont des mots de code de Tardos. Notre choix pour les codes de Tardos s'explique par leur simplicité et leur longueur optimale [23].

Le principe de la méthode découle de l'approche proposée par Z. Shahid [30]. Le schéma d'insertion comprend deux étapes (figure 4.2) :

- *Etape 1* : La génération des codes binaires de Tardos.
- *Etape 2* : L'insertion du code dans chaque copie vidéo compressée par H.264/AVC.

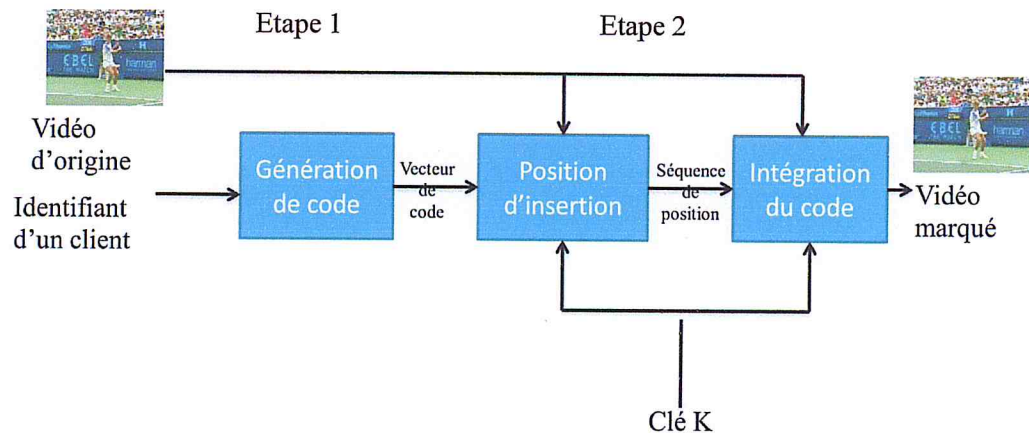


Figure 4.2. schéma d'insertion de la marque

L'extraction de la marque se fait au niveau de décodeur.

3.1. Génération des codes binaires de traçabilité

Contexte : un serveur multimédia distribue des copies d'un même contenu numérique à n acheteurs différents. Ces copies sont personnalisées et contiennent chacune un message qui les identifie, caché à l'aide d'une technique de tatouage robuste et invisible.

Des utilisateurs malhonnêtes, appelés « colluders », utilisent leurs copies pour créer un faux qu'ils vont redistribuer de façon illégal. L'objectif est alors pour le distributeur de contenus de retrouver l'identité de ces colluders par un processus d'accusation en utilisant la marque extraite de la copie pirate. Pour que cette accusation soit efficace, le schéma doit s'appuyer un bon code anti-collusion.

Notre choix des codes, s'est porté sur les codes traçants binaires de Tardos reposants sur des probabilités. Ce sont des codes très intéressants par leur ordre de longueur optimale et faciles à implémenter, ils permettent un bon contrôle des probabilités d'erreur d'accusation (risque d'accuser un innocent, risque de rater des pirates).

La construction du code de Tardos regroupe trois étapes principales qui sont [23]:

- Génération de probabilité.
- Génération de la matrice de code.
- Accusation.

Ces processus de génération dépendent des paramètres :

- n le nombre total d'utilisateurs.
- c le nombre de pirates (*colluders*) prévus parmi les utilisateurs.
- m la longueur du code de traçabilité (marque) que chaque utilisateur possède.

Le calcul de la longueur du code m est défini suivant les formules suivantes :

$$m = 100 c^2 k \quad (4.1)$$

Avec $k = \log(1/\varepsilon) \quad (4.2)$

Où ε est la probabilité d'erreur commise en accusant un innocent.

3.1.1. Génération des probabilités

Pour un code de longueur m , on génère d'une façon aléatoire des probabilités $p(i)$ ou $0 \leq i \leq m$ selon une fonction de densité de probabilité $f(p)$ donnée par :

$$f(p) = \frac{1}{\pi \sqrt{p(1-p)}} \quad (4.3)$$

Avec $p \in [0,1]$.

$p_i \in [t, 1-t]$ tel que $t = \frac{1}{300c}$. Si t prend la valeur 10^{-3} donc la valeur maximale de $p = 0,999$ et la valeur minimale $p = 0,001$.

Où

$$p_i = \sin^2 r_i \tag{4.4}$$

$$r_i \in [t', \pi/2 - t'] \quad 0 < t' < \pi/4 \tag{4.5}$$

$$\sin^2 t' = t \tag{4.6}$$

Pour un utilisateur j , un vecteur de probabilité p_j de longueur m est généré : $p_j = \{p_{j1}, p_{j2}, \dots, p_{jm}\}$.

3.1.2. Génération de la matrice de code

Pour un nombre d'utilisateur n , une matrice de probabilité S de dimension $(n \times m)$ est générée (figure 4.3). Pour générer la matrice S , m nombres réels $p_i \in [0,1]$ sont tirés au hasard selon la fonction de densité de probabilité $f(p)$.

Chaque élément (i,j) de la matrice S est ensuite indépendamment tiré en suivant la probabilité :

$$Prob[S(i,j)=1] = p_j \tag{4.7}$$

Chaque ligne de la matrice S est le code d'empreinte d'un utilisateur.

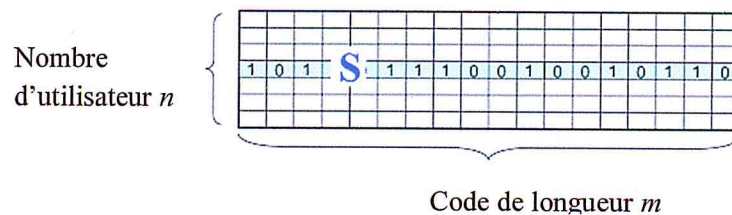


Figure 4.3. La matrice de code de longueur m pour n utilisateurs

Chacun de ces n mots est caché dans la copie délivrée à l'utilisateur associé.

3.1.3. Accusation

Afin de savoir si l'utilisateur j est impliqué dans la production du faux, on calcul un score d'accusation A_j . Ce dernier est donné par :

$$A_j = \sum_{i=1}^m Y_i * U_{ji} \tag{4.8}$$

$$U_i = \begin{cases} \sqrt{\frac{1-p_i}{p_i}} & \text{si } S_{ii} = 1 \\ -\sqrt{\frac{p_i}{1-p_i}} & \text{si } S_{ii} = 0 \end{cases} \quad (4.9)$$

Avec Y_i est l’empreinte extraite d’une copie pirate.

L’utilisateur j est considéré comme pirate si A_j est supérieur à un seuil Z défini par :

$$Z = 20 * c * k \quad (4.10)$$

3.2. Processus d’insertion

Après la génération des codes traçants binaires, un code est inséré dans chaque copie compressée par le standard H.264/AVC (figure 4.4)

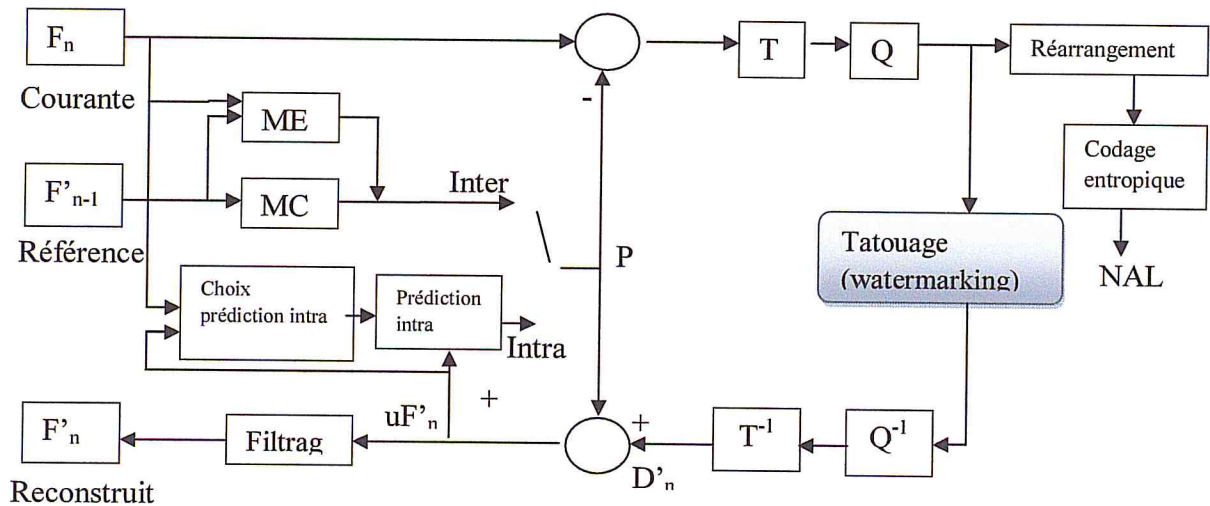


Figure 4.4. Schéma du processus d’insertion dans le codeur

Le schéma de tatouage proposé appartient à la classe de tatouage additif et le code est caché d’une façon invisible dans la vidéo au cours de sa compression.

Le domaine compressé est considéré comme plus pratique dans le système de traçage développé car la vidéo est habituellement stockée dans un format compressé avant qu’elle ne soit transmise sur les réseaux.

Un autre point important concernant le tatouage vidéo est la sélection de la composante de couleur appropriée pour l’insertion. Habituellement, les images d’une

vidéo sont divisées en composante de luminance Y, et deux composantes de chrominances Cr, Cb ou U, V au cours de codage. Selon différentes règles de ré-échantillonnage de couleur, il est possible pour le taux Y:U:V qu'il soit réglée à 4:2:2 ou 4:2:0. Dans ces circonstances, la seule composante inchangée est la composante de luminance Y. ainsi, pour l'insertion des codes dans la vidéo, nous préférons exploiter la composante Y comme signale hôte.

Les étapes de l'algorithme d'insertion se résument comme suit:

- Sélection des positions d'insertion (macroblochs).
- Calcul de la force de marquage pour le macrobloc courant.
- Insertion de la marque (tatouage).

La figure 4.5 présente un organigramme général de processus d'insertion.

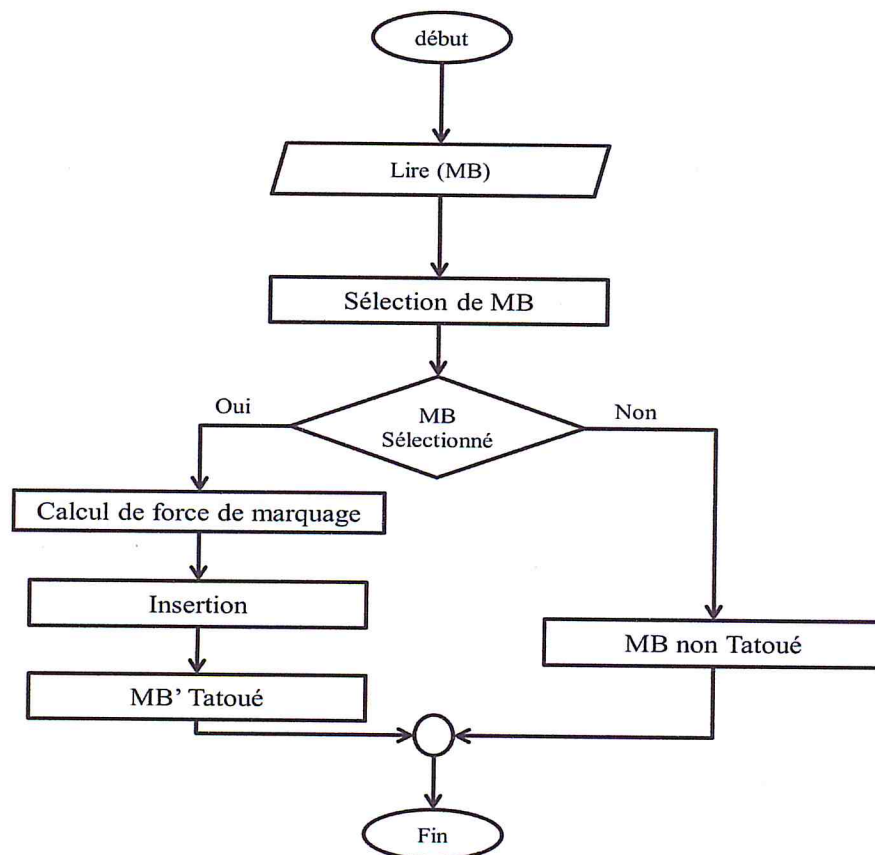


Figure 4.5 Organigramme général d'insertion

3.2.1. Sélection des trames et les positions pour l'insertion

En plus de la sélection du domaine d'insertion et la composante du signal, le choix du type de la trame appropriée entre trame I, trame P ou trame B pour cacher les marques est également un élément crucial.

Habituellement, une vidéo classique consiste à un certain nombre de GOP. Chaque GOP est composé d'une trame I et plusieurs trames P et B. le standard code la première trame d'une séquence avec un codage intra-trame (trame I) ce qui signifie que le codage ne se réfère qu'à la trame courante (prédiction spatiale).

A la différence, les trames P sont codées en inter-trame. C'est-à-dire, le codage de cette trame dépend des images précédemment codées de type I ou P. par contre pour une trame B, le codage réfère à la plus proche précédent et succédant la trame P.

Par conséquent, seules les trames de type I peuvent détenir des informations complètes. La raison pour laquelle nous avons sélectionné les trames I et P pour insérer les codes.

Dans la norme H.264/AVC. Les trames de type I contiennent des macroblocs codés en mode intra_16x16 ou en mode intra_4x4. Le mode intra_16x16 effectue la prédiction et le codage résiduel sur un macrobloc entier de luminance de taille 16x16 et est bien adapté pour le codage des zones homogènes [31]. Par contre le mode intra_4x4 est basé sur la prédiction de chaque bloc de luminance séparément et est bien adapté pour le codage des parties d'image comprenant des détails significatifs.

Pour limiter les distorsions d'insertion, la marque est insérée dans les blocs de type intra_4x4 et inter_4x4 au niveau des coefficients continus DC. Dans le domaine DCT, la composante DC est plus adapté à l'insertion de la marque que les coefficients AC [32].

La composante DC représente la moyenne du bloc et elle a une plus grande capacité de perception, après l'insertion de la marque, elle ne cause pas de changement évident de la qualité visuelle de l'image d'origine, d'autre part le traitement du signal et les interférences de bruit ont de faible influence sur la composante DC que sur les composantes AC.

Afin de minimiser au mieux les distorsions et assurer l'invisibilité, nous avons adopté un mécanisme d'insertion adaptative selon les étapes suivantes (figure 4.6).

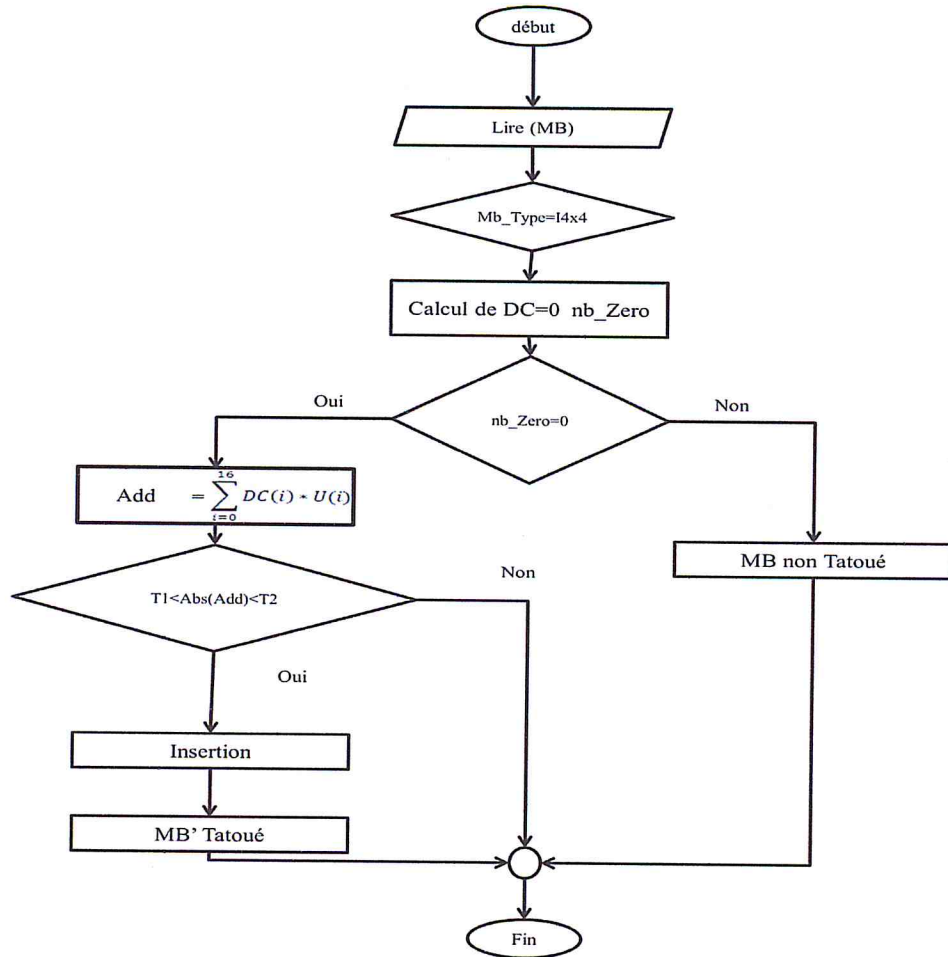


Figure 4.6 Organigramme de sélection des macroblocs

- Le calcul de la somme des coefficients DC quantifiés d'un macrobloc k avec la clé U_i qui est une séquence Gaussien, $U_i \in \{-1,1\}$.

$$add_{(k)} = \sum_{i=1}^{16} dc_{ij} * U_i \quad (4.11)$$

- Le macrobloc dont la valeur de l'un de ses DC est égale à zéro est écarté, la sélection du macrobloc pour l'insertion doit vérifier la condition suivante :

$$T_1 < |add_{(k)}| < T_2 \quad (4.12)$$

Avec T_1 et T_2 sont des seuils de synchronisation prédéfinis.

3.2.2.L'insertion de la marque

La figure 4.7 schématise le processus d'insertion. L'insertion des codes s'effectue selon la formule suivant :

$$Y = X + \alpha U_i w(i) \tag{4.13}$$

et

$$w(i) = (-1)^{S(i,j)} \tag{4.14}$$

Où

- Y est le vecteur des coefficients DC tatoués.
- X représente le vecteur des coefficients DC d'origine d'un bloc 4x4.
- U_i est la séquence Gaussien(clé).
- $S(i, j)$ est le $i^{\text{ème}}$ bit du code de Tardos du $j^{\text{ème}}$ utilisateur.
- α est la force de marquage, elle dépend du contenu vidéo.

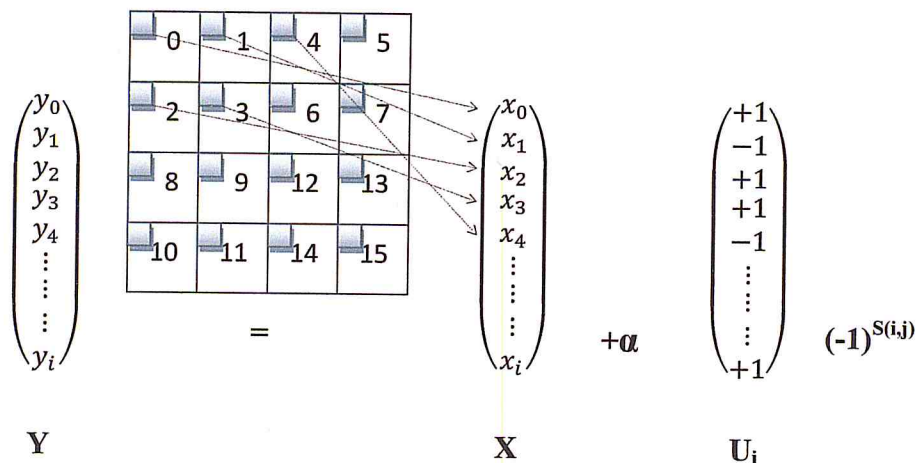


Figure 4.7 Schéma de tatouage d'un macrobloc

L'introduction de la séquence gaussienne U_i dans l'opération d'insertion est justifiée par le fait que beaucoup de techniques de tatouage robuste existantes dans la littérature offrant la robustesse [33] et la capacité [34] sont basées sur la méthode d'étalement de spectre. D'autre part, il a été confirmé par de nombreux travaux que le tatouage basé sur l'étalement de spectre est résistant face à de nombreuses attaques, quand les marques sont de distribution gaussienne et sont statistiquement indépendantes, particulièrement aux attaques de collusion [35]. L'idée de base de cette

stratégie est que le caractère aléatoire inhérent à ces marques rend la probabilité d'accuser un utilisateur innocent faible.

L'insertion est effectuée selon la relation qui existe entre le signe de Add définie dans la formule (4.11) et $w(i)$ l'empreinte.

$$diff(k) = sign(Add_k) \quad (4.15)$$

$$\text{avec } sign(a) = \begin{cases} -1 & \text{si } a < 0 \\ +1 & \text{si } a > 0 \end{cases} \quad (4.16)$$

L'insertion est comme suit :

Cas 1 Si $diff(k) = w(i)$, les coefficients DC du $k^{\text{ième}}$ macrobloc reste inchangés.

Cas 2 Si $diff(k) \neq w(i)$, l'insertion est effectuée selon la formule (4.13).

Afin d'augmenter la capacité, l'insertion est faite d'une manière adaptative selon le contenu, la force de marquage α n'est pas fixe comme dans le cas de la méthode de Shahid [30] mais elle dépend du contenu de la trame. Celle-ci est calculée comme suit :

- Calcul de la moyenne $D(i)$ de tous les coefficients DC différents de zéro dans un macrobloc [36]:

$$D(i) = \frac{1}{ki} \sum_{j=1}^{ki} |DC_{ij}| \quad (4.17)$$

Avec ki est le nombre de $DC(i,j)$ différent de zéro du $i^{\text{ième}}$ macrobloc.

- Calcul de la moyenne de la trame \bar{D} , $D = \{ D(i), i=1,2,\dots, N \}$ tel que N est le nombre de macroblocs parcourus dans la trame.

$$\bar{D} = \frac{1}{N} \sum_{i=1}^N D(i) \quad (4.18)$$

La force de marquage α est donnée par :

$$\alpha = \begin{cases} T - (D(i) - \bar{D}) & \text{si } S(i,j) = 0 \\ T + (D(i) - \bar{D}) & \text{si } S(i,j) = 1 \end{cases} \quad (4.19)$$

Tel que T est défini par :

$$T = \lambda T_{min} + (1 - \lambda)T_{max} \text{ avec } 0 \leq \lambda \leq 1 \quad (4.20)$$

Avec

$$T_{min} = \min\{|D(1) - \bar{D}|, \dots, |D(N) - \bar{D}|\} \quad (4.21)$$

$$T_{max} = \max\{|D(1) - \bar{D}|, \dots, |D(N) - \bar{D}|\} \quad (4.22)$$

Après l'insertion, pour une valeur de Y_i nulle, le décodeur ne peut pas extraire exactement le bit de la marque, pour cela cette dernière est forcée à prendre le signe du coefficient DC d'origine:

$$Y_i = \text{sign}(DC_i). \quad (4.23)$$

Le processus d'insertion est achevé si les deux conditions suivantes sont assurées avec les nouveaux coefficients quantifiés :

$$T_1 < |add| < T_2 \quad (4.24)$$

$$\text{sign}(add) = w(i) \quad (4.25)$$

La figure 4.8 représente un organigramme détaillé des étapes à suivre pour l'insertion de la marque.

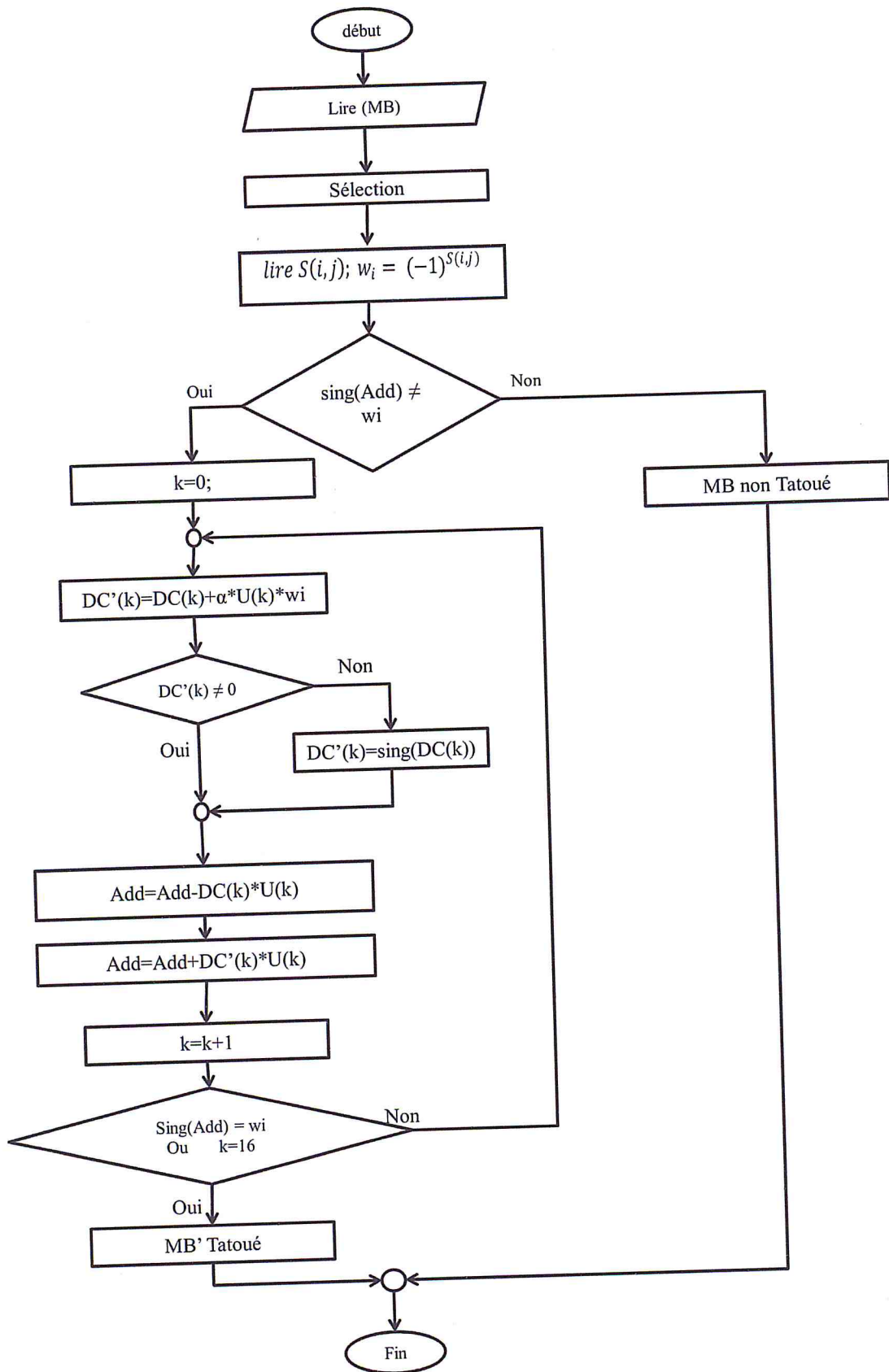


Figure 4.8 Organigramme détaillé d'insertion

3.3. Processus d'extraction

L'extraction de la marque s'effectue au niveau du décodeur H.264/AVC d'une manière aveugle (figure 4.9). (C.-à-d. le décodeur n'a pas besoin de la vidéo d'origine pour extraire les bits du code).

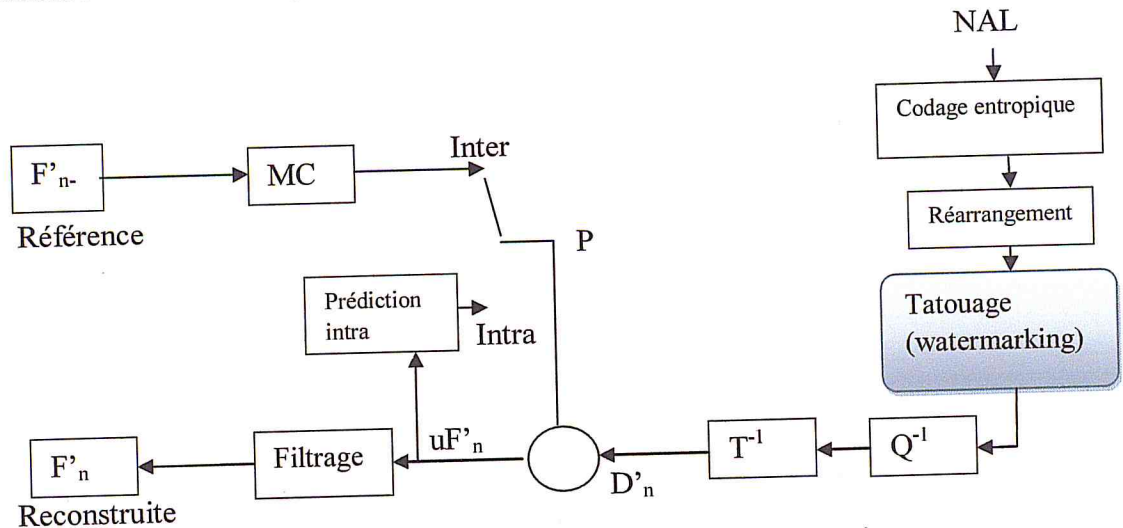


Figure 4.9 Schéma du processus d'extraction

La valeur de (*add*) est calculée selon la formule (4.11).

Le processus d'extraction suit les étapes suivantes:

- La sélection des positions d'extraction des macroblocs tatoués selon la clé.

Cette opération opère exactement comme celle de l'insertion.

- Extraction de la marque.

Après le décodage entropique des coefficients quantifiés *DC* sont notés par *Z*. Le bit de la marque est extrait de *Z* par une corrélation linéaire de *Z* et la clé *U_i* de longueur *l*:

$$\tilde{S}(i, j) = \begin{cases} 0 & \text{si } add > 0 \\ 1 & \text{si } add < 0 \end{cases} \quad (4.26)$$

L'organigramme dans la figure 4.10 donne les étapes à suivre pour l'extraction des bits de la marque.

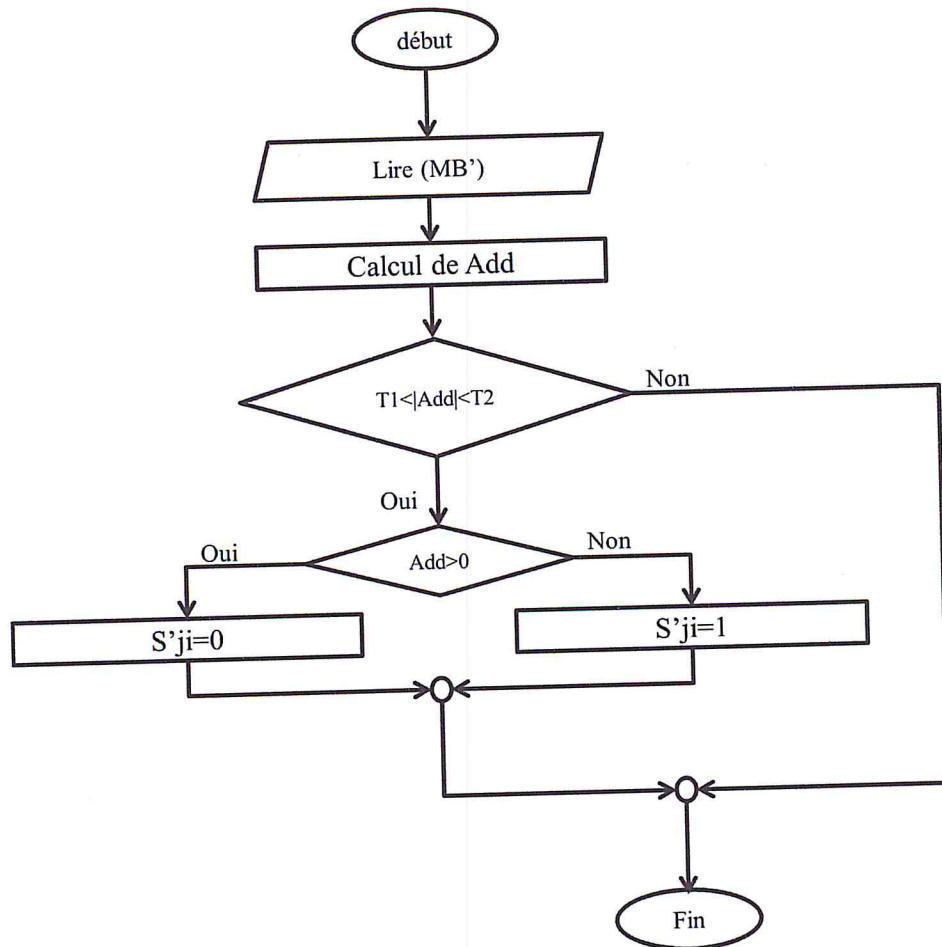


Figure 4. 10. Organigramme d'extraction

3.4. L'accusation

Lorsque un vendeur désire tracer la provenance d'une vidéo, il doit extraire le mot de Tardos insérer dans la vidéo et ensuite lancer son processus d'accusation. Pour cela, il calcul un score d'accusation A_j selon la formule (4.8) pour chaque utilisateur j .

L'utilisateur est considéré comme un pirate (colluders) si son score est supérieur à un seuil Z calculé selon l'équation (4.10).

$$Z = 20 * c * k \quad (4.10)$$

4. Présentation de l'application

Nous allons présenter la fenêtre principale de l'application et les différents outils permettant la communication avec le logiciel. La figure 4.11 montre la structure de la fenêtre principale.

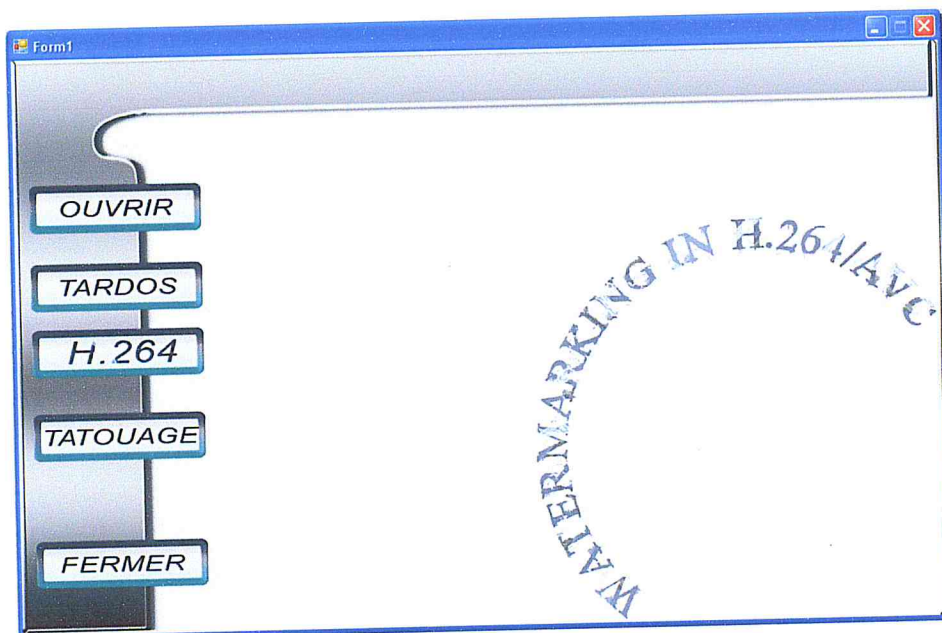


Figure 4. 11. Fenêtre principale de l'application

La barre des boutons apparait sous forme de barre verticale, elle se compose des boutons suivants :

- **OUVRIR** : permet de visualiser une vidéo en format YUV. La figure 4.12 montre un exemple de la visualisation de la vidéo «Bus » de résolution CIF

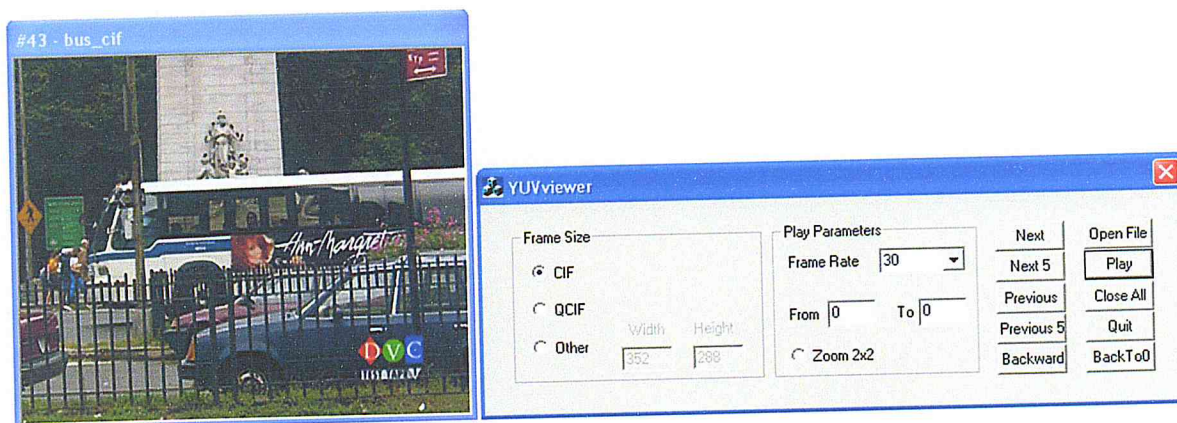


Figure 4. 12. Lecture de la séquence Bus_cif

- **TARDOS** : bouton qui donne accès a une nouvelle fenêtre pour générer l'ensemble des codes (figure 4.13).

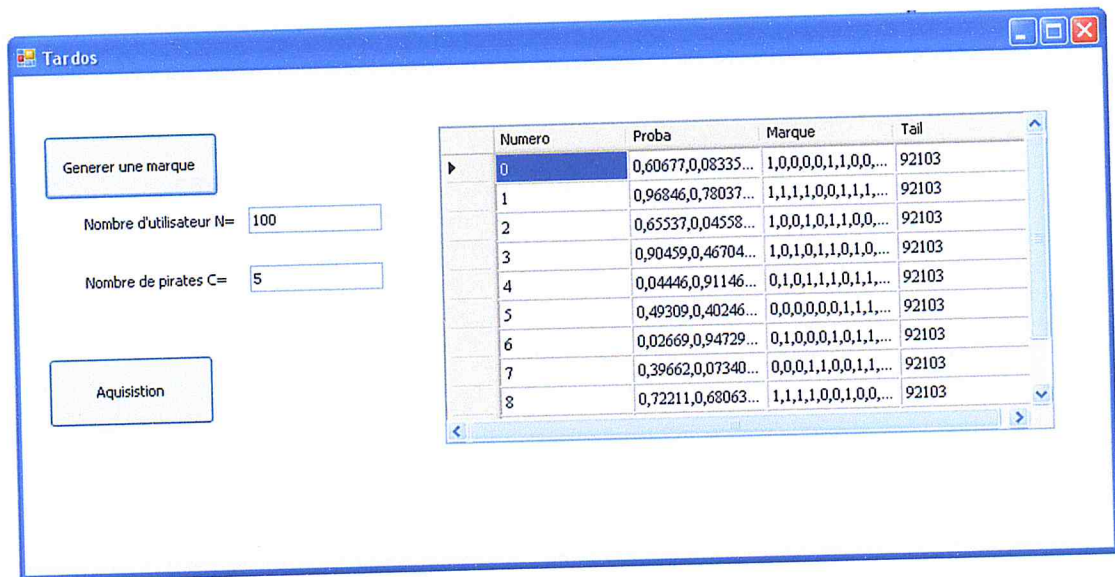


Figure 4. 13. Fenêtre de génération de code.

Cette fenêtre permet à l'utilisateur de générer des codes selon les paramètres n et c . L'ensemble des codes est visible sur un tableau dans la fenêtre.

- **H.264** : ce bouton permet une compression et décompression des séquences vidéo avec le codec H.264/AVC (figure 4.14)



Figure 4. 14. Bouton de compression et décompression

Pour la compression une fenêtre apparaît permettant de choisir la vidéo ainsi que les paramètres de compression comme la figure 4.15 le montre.

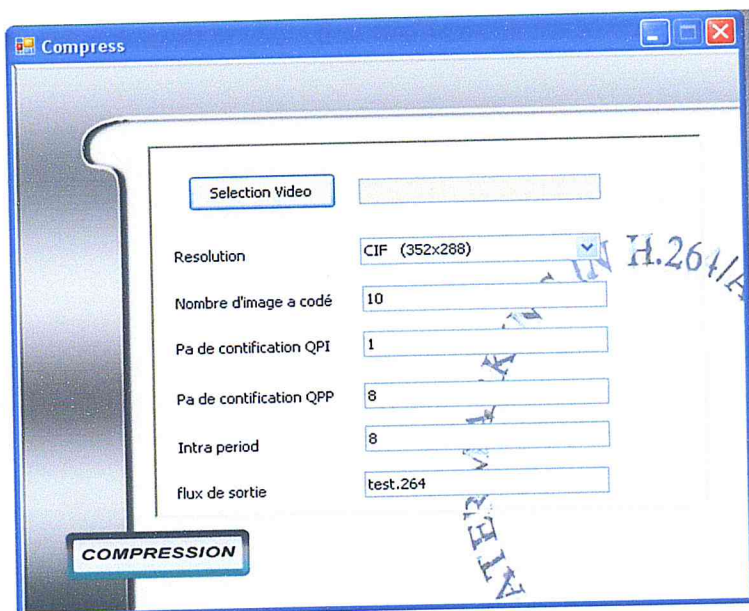


Figure 4. 15. Fenêtre de compression

Pour la décompression, une autre fenêtre (figure 4.16) apparaît qui permet de sélection un flux compressé pour une décompression.

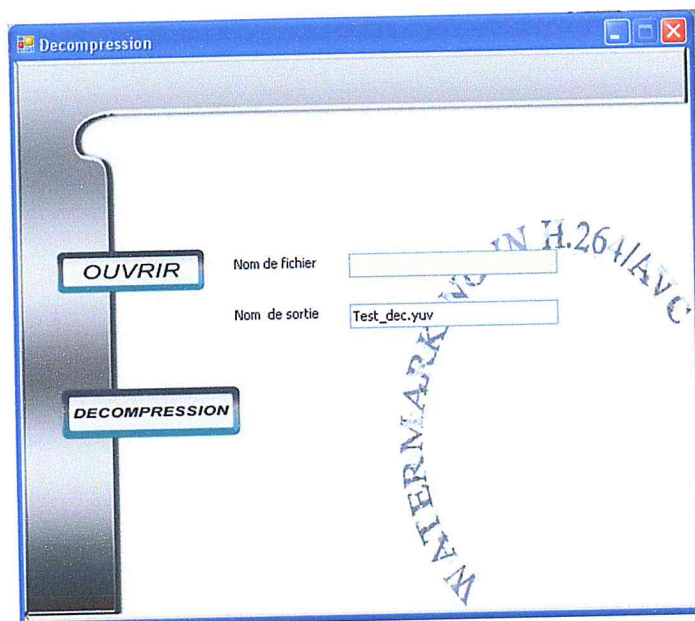


Figure 4. 16. Fenêtre de décompression

TATOUAGE : il permet les opérations d'insertion et d'extraction de la marque (empreinte digitale) dans la vidéo sélectionnée par l'utilisateur (figure 4.17).

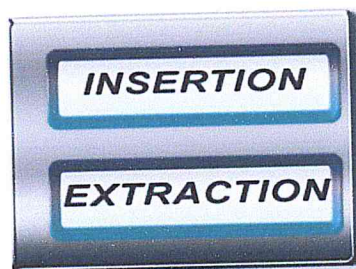


Figure 4. 17. Bouton d'insertion et d'extraction de la marque

Pour l'insertion, l'utilisateur sélectionne la vidéo à tatouer ainsi que les paramètres de compression (figure 4.18). L'utilisateur peut visualiser l'ensemble des codes générés, afin de sélectionner le processus d'insertion pour un seul utilisateur (une marque) ou tous les utilisateurs.

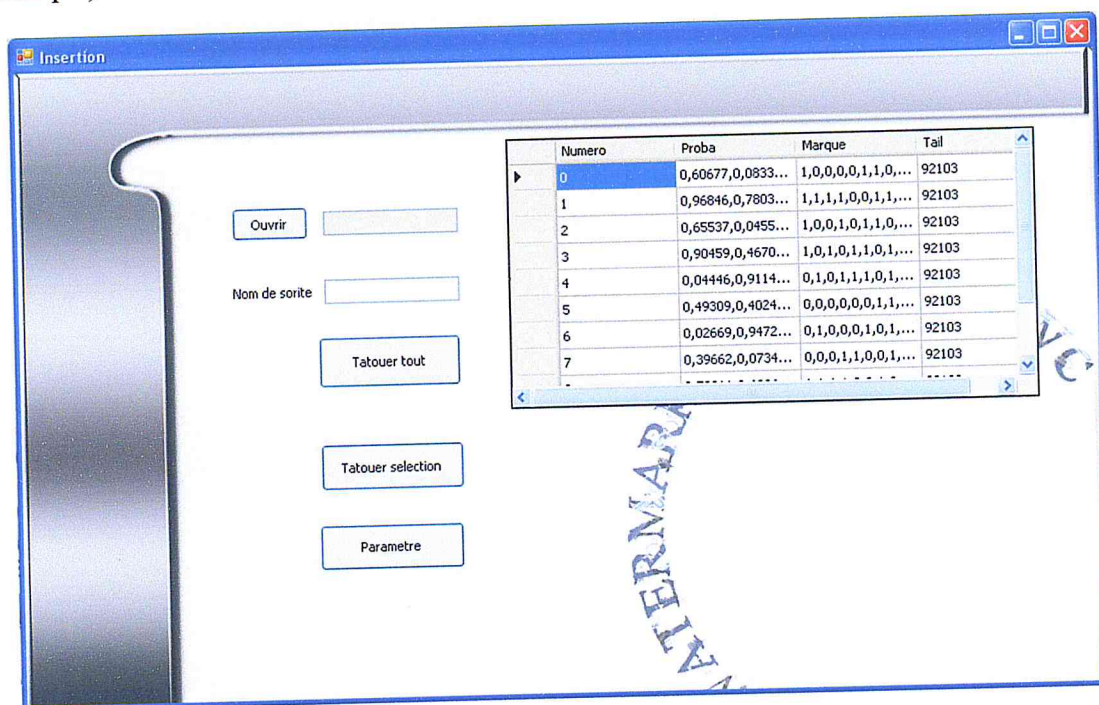


Figure 4. 18. Fenêtre d'insertion de la marque

Pour l'extraction de la marque, la décompression de la vidéo tatouée est nécessaire (figure 4.19).

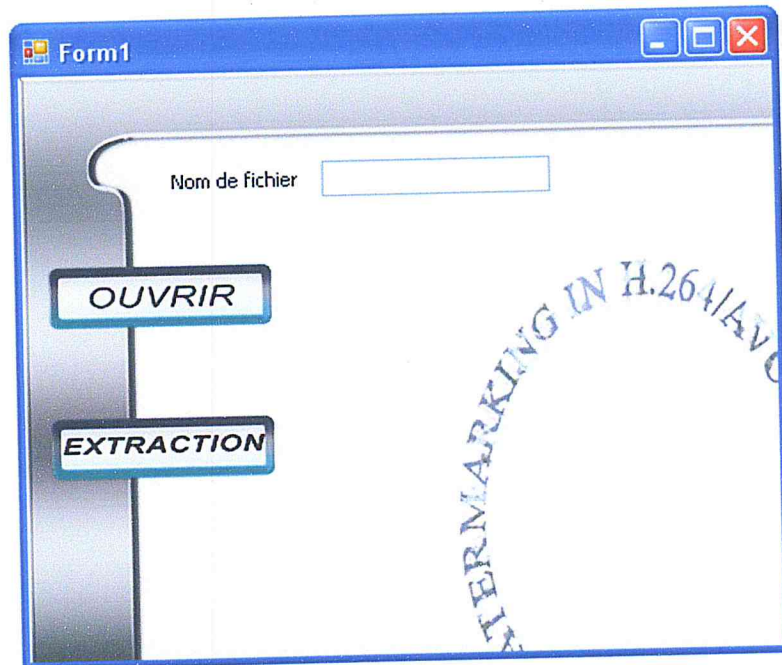


Figure 4. 19. Fenêtre d'extraction de la marque

- **FERMER** : ce bouton permet une fermeture complète de l'application

5. Conclusion

Nous avons présenté dans ce chapitre une approche de tatouage des codes traçants, constituée d'un code de traçabilité de Tardos et une technique robuste d'insertion. Cette méthode a pour objectif d'identifier des utilisateurs impliqués dans la création de fausses copies des vidéos compressées par la norme H.264/AVC dans le contexte de la vidéo à la demande. L'objectif est alors pour le distributeur de contenus vidéo de retrouver l'identité des pirates par un processus d'accusation utilisant la marque extraite de la copie pirate. Pour que cette accusation soit efficace, le schéma doit s'appuyer sur une technique de tatouage robuste et un code anti-collusion. Pour se faire, différentes étapes sont suivies pour concevoir ce schéma. Une première étape qui consisté à la génération du code de traçabilité de Tardos, suivit par une étape qui consiste à insérer la marque dans les blocs de type Intra 4x4 et Inter 4x4, précisément dans les coefficients DC quantifiés non nuls. La dernière étape qui constitue l'extraction du code au cours du décodage entropique afin de procéder au processus d'accusation.

Dans le prochain chapitre nous donnerons les différents tests et expérimentations effectués du système réalisé afin d'évaluer la méthode proposée.

Chapitre 5

Tests et résultats

1. Introduction

Afin d'évaluer les performances de notre méthode de tatouage vidéo développée pour le traçage des copies de pirates de la vidéo compressée par codec H.264/AVC. Deux types attaques sont appliqués :

- Les attaques malveillantes, c'est les attaques de collusion. Ces dernières ont toutes été réalisées dans le domaine spatial. Les disposent tous d'une version différente de la même vidéo et créent une nouvelle vidéo en utilisant une stratégie particulière. Chaque pixel de la nouvelle vidéo est obtenu en utilisant l'une des stratégies suivantes : calcul de la moyenne (AVG), calcul du Minimum (Min), Maximum (Max), calcul de la valeur Mediane (Med), le calcul de la valeur MinimumMaximum (MinMax) et calcul du modifié Négative (ModNeg).

Les fonctions de collusion de ces attaques sont données par [30][37] :

$$Z_{avg}(j) = \sum_{K \in S_c} \frac{Y_k(j)}{k} \quad (5.1)$$

$$Z_{min}(j) = \min\{Y_k(j)\}_{K \in S_c} \quad (5.2)$$

$$Z_{max}(j) = \max\{Y_k(j)\}_{K \in S_c} \quad (5.3)$$

$$Z_{med}(j) = \text{med}\{Y_k(j)\}_{K \in S_c} \quad (5.4)$$

$$Z_{minMax}(j) = (Y_{min}(j) + Y_{max}(j))/2 \quad (5.5)$$

$$Z_{modNeg}(j) = Y_{min}(j) + Y_{max}(j) - Y_{med}(j) \quad (5.6)$$

avec Z le pixel manipulé.

- L'attaque bienveillante constituée par l'opération de compression.

Les tests appliqués dans ce chapitre sont réalisés sur les séquences vidéos de format CIF (résolution 352 x 288 pixels) et sur les séquences vidéos de format 4CIF (résolution 704 x 576 pixels) et des séquences de format 1080P (résolution 1920 x 1080 pixels) à 30 images/seconde et en mode d'échantillonnage 4:2:0. Les tests sont effectués avec un pas de quantification $Q_p=10$. Les séquences de test de format CIF sont les suivantes (figure 5.1) [sw5] :



Figure 5.1. Liste des séquences vidéos des tests.

Ainsi deux séquences de tests de format 4CIF ont été utilisées :

- « City » avec 598 images dans la vidéo.
- « Soccer » avec 298 images dans la vidéo.

Et une vidéo de format 1080p appelé full HD

- « Sunflower » qui contient 498 images.

Ce travail est réalisé sur un CPU INTEL Core 2 Quad : 2,40 Ghz, 2 Go de RAM. L'environnement de programmation utilisé pour la réalisation des procédures d'insertion et d'extraction du système de traçage des copies de pirates est Microsoft Visual Studio 2008 .nous avons utilisé l'implémentation de JM 10.1 de H.264/AVC [sw7] pour implémenter le mécanisme de traçabilité proposée.

Pour comparer et analyser les vidéos d'origines et les vidéos traitées nous avons utilisé le logiciel « YUV Tools » et « MSU Video Quality Measurement ».

Les procédures d'insertion et d'extraction sont testées sur le High Profil de la version JM 10.0 de la norme H.264/AVC [sw7].

Les paramètres d'entrée du codeur sont regroupés dans le tableau 5.1 :

Tableau 5.1. Paramètre d'entrer utilisé pour le codeur H.264/AVC

Paramètre de compression	valeurs	Description
FramesToBeEncoded	200	Le nombre d'images codées dans la vidéo
FrameRate	30	Le nombre d'image par seconde
ProfileIDC	100	Le profile High (élevé)
NumberReferenceFrames	10	Le nombre des images e référence utilisées pour la prédiction Inter (1-10)
IntraPeriod	20	La période des images
QPIslice	10	Pas de quantification pour les trames I
QPPslice	10	Pas de quantification pour les trames P
BFrames	0	L'utilisation des trames B

Les performances de la méthode développée ont été évaluées en termes de qualité vidéo en se basant sur le calcul du PSNR (Peak Signal to Noise Ratio) et le SSIM (Structur Similarity Image Measurement).

- PSNR [sw8]

Le PSNR est le critère le plus usuel pour l'évaluation de la qualité. Il est donné par la formule (5.7) suivante :

$$PSNR(dB) = 10 \log_{10} \left(\frac{Imax^2}{\frac{1}{N*M} \sum_{i=1}^N \sum_{j=1}^M [I_0(i,j) - I_d(i,j)]^2} \right) \quad (5.7)$$

Où

I_0 : l'image tatouée.

I_d : l'image d'origine.

N, M : les dimensions de l'image.

$Imax$: la valeur maximale de l'intensité d'un pixel ($Imax = 255$ avec un codage sur 8 bits).

- SSIM [sw8]

De la même façon que le PSNR, le SSIM est une mesure de similarité entre deux images numériques. L'idée de SSIM est de mesurer la similarité de structure entre les deux images, plutôt qu'une différence pixel à pixel comme le fait par exemple le PSNR. L'hypothèse sous-jacente est que l'œil humain est plus sensible aux changements dans la structure de l'image.

La métrique SSIM est définie par la formule (5.8) suivante :

$$SSIM(I, I') = \frac{(2\mu_I\mu_{I'} + C_1)(2cov_{II'} + C_2)}{(\mu_I^2 + \mu_{I'}^2 + C_1)(\sigma_I^2 + \sigma_{I'}^2 + C_2)} \quad (5.8)$$

Avec :

μ_I La moyenne de I .

$\mu_{I'}$ La moyenne de I' .

σ_I^2 La variance de I .

$\sigma_{I'}^2$ La variance de I' .

$cov_{II'}$ La covariance de I et I' .

$C_1 = (K_1L)^2$, $C_2 = (K_2L)^2$ deux variables destinées à stabiliser la division quand le dénominateur est très faible.

L est la dynamique des valeurs des pixels, elle est égale à 255 pour des images codées sur 8 bits.

$K_1=0.01$ et $K_2=0.03$ par défaut.

2. Tests de la génération du code de Tardos

Selon les paramètres d'entrée du code de Tardos, le tableau 5.2 montre la longueur de la marque selon les paramètres prisent pour la génération de l'ensemble des codes :

Tableau 5.2. Paramètre d'entrer utilisé pour la génération du code de Tardos

N nombre d'utilisateur	C nombre de pirate prévu	ϵ la probabilité d'erreur	M la longueur de message (marque)
100	20	10^{-1}	92 103
100	20	10^{-3}	276 311
10000	10	10^{-5}	115 129

La longueur du code augmente quadratiquement avec le nombre de pirates logarithmiquement avec la diminution de la probabilité d'erreur ϵ , par exemple pour $c=10$, $n= 10^4$ et $\epsilon= 10^{-5}$ la longueur $m=115\ 129$ bits.

Cela fait une complexité de calcul de l'accusation. Car le nombre d'appels à la fonction U est extrêmement grand aussi un cout de stockage de l'ordre des Gigabits est nécessaire si l'on souhait stocker la matrice S.

3. Tests des processus d'insertion et d'extraction

Pour évaluer le processus d'insertion et d'extraction par rapport aux critères d'invisibilité et de capacité (tableau 5.3), nous avons testé l'insertion dans les séquences vidéo selon les paramètres d'évaluation suivants :

- PSNR la qualité visuelle subjective.
- Capacité qui est le nombre de bit insérer dans les images I.
- BitRate taux de compression.

Dans le travail de Shahid [30], l'auteur a inséré que 10 bits par trames, et il n'a pas vraiment assuré l'invisibilité. La vidéo illustrée sur la figure 5.2 est une capture prise de la vidéo test publié par l'auteur [sw9].



Vidéo d'origine



Vidéo tatoué



Vidéo d'origine



Vidéo tatoué

Figure 5.2 résultat d'insertion de Shahid

Aussi l'auteur, dans son travail a utilisé une force de marquage fixe, alors que dans notre approche, elle est variable selon le contenu. Dans ce cas nous avons augmenté considérablement la capacité en la comparant à celle obtenue par Shahid [30]. Le tableau 5.3 illustre ce résultat pour les différents vidéos tests, ainsi pour la capacité totale d'insertion. Ainsi pour insérer une marque de longueur $m=92103$, l'auteur à utilisé une séquence vidéo d 9211 images soit 10 bits par image. Par contre dans notre méthode, nous n'avons utilisé que 2096 images pour une marque de même longueur.

Tableau 5.3. Les résultats d'insertion dans les vidéos de tests

résolution	séquence	Avant l'insertion		Après l'insertion			
		PSNR	BitRate	Capacité d'insertion		PSNR	BitRate
				Trame I	Toute la vidéo		
CIF	Bus	49.554	2302396	24	10810	47.728	2302401
	Forman	49.742	2026565	45	23566	47.445	2026577
	Paris	50.298	2372880	36	26150	47.863	2372898
	Stefan	51.596	2364119	48	3328	50.400	2364159
	Soccer	52.298	1900352	40	10225	50.298	1900361
	Crew	52.345	1911891	69	18136	51.860	1911897
	Football	52.277	2029106	22	5960	50.123	2029106
	City	52.181	2128583	21	6258	49.569	2128576
4 CIF	City	52.168	7778336	137	81926	52.168	7778328
	Soccer	52.416	6827717	154	45892	52.416	6827704
1080p	Sunflower	52,580	34040504	2239	111022	52,580	3404101 6

Au point de vue qualité vidéo subjective, il est impossible que l'œil humain peut distinguer la différence entre une vidéo d'origine et celle tatouée. Figure 5.3 illustre les résultats.

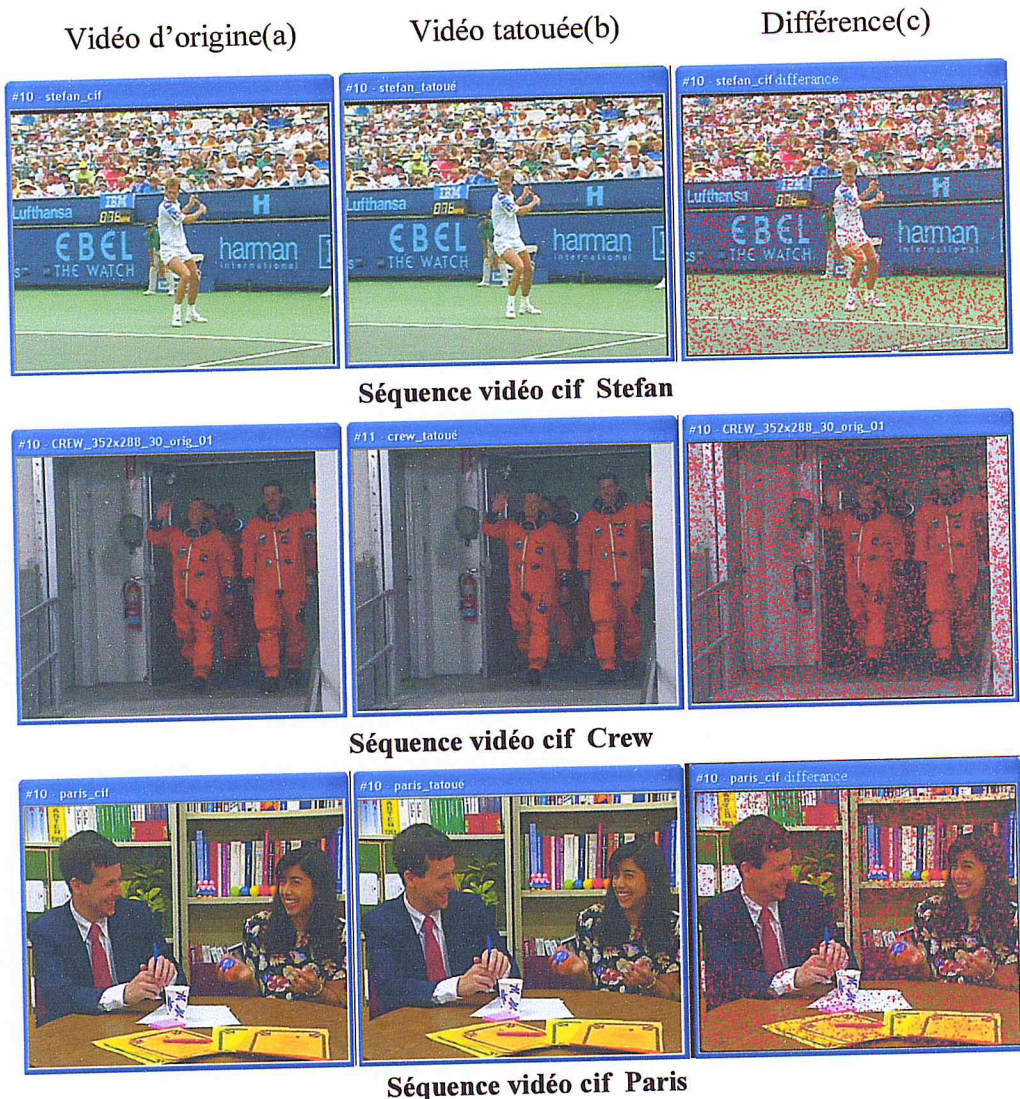


Figure 5.3. Comparaison des séquences vidéos

Dans les vidéos illustrées par la figure 5.3. Il est clair qu'il y a aucune différence en qualité subjective entre la trame d'origine et celle tatouée. Mais en utilisant le logiciel YUVTools, cette différence est claire, elle figure en point rouge dans la partie (c) de la figure 5.3.

Pour les vidéos 4CIF, HD (720p) et Full HD (1080p ou 1080i). La résolution est très grande donc la capacité d'insertion est très grande aussi. Par exemple, dans la vidéo **Sunflower** la capacité d'insertion d'une seule trame est **2239 bits**, et la capacité d'insertion dans toute la vidéo est **111022 bits**, celle-ci est une très grande capacité à insérer dans une vidéo de 498 images.

Pour la perceptibilité subjective, il est impossible de voir la différence entre la vidéo d'origine et celle tatouée sans utiliser un logiciel de comparaison.

4. Evaluation de la robustesse face aux attaques

Dans le test de robustesse, l'objectif est d'étudier le comportement du système face aux attaques de collusion et attaque par compression. Un scénario de test est calculé à partir des paramètres d'entrée suivantes :

Pour le nombre d'utilisateur $n = 100$, $c=20$ pirates et la probabilité d'erreur $\epsilon=10^{-1}$. La longueur m du message à insérer est de $m = 100 * 20^2 * \ln(10) = 92103$.

Cette marque de longueur de 92103 bits doit être insérée dans les vidéos CIF. Comme les vidéos de format CIF mises à notre disposition pour les tests, ne nous permettent pas d'insérer cette quantité, nous avons créé une nouvelle vidéo CIF de taille plus grandes appelées TousLesVideos (environ 2100 images) en répétant les vidéos tests autant de fois pour augmenté la taille.

Pour appliquer les attaques citées précédemment un logiciel est développé.

Le tableau 5.4 montre le nombre de pirates qui ont été successivement détectés après l'analyse de la vidéo pirate pour les différentes stratégies de collusion et pour le nombre variable de pirates. La stratégie la plus efficace est l'attaque ModNeg. Pour toutes les autres attaques, quelle que soit le nombre de pirate, le processus d'accusation retrouve plus que la moitié des traîtres. Sachant que dans une application de traçage de traîtres, le plus important est de déterminer au moins un traître, d'où nous pouvons conclure que l'approche fonctionne extrêmement bien.

Tableau 5.4 nombre de pirates tracés à partir de K copies pirates pour différents attaques

k	Nombre de pirates détectés pour chaque attaque					
	AVG	Min	Max	Médian	MinMax	ModNeg
2	2	2	2	2	2	2
5	4	5	5	5	5	3
8	8	8	8	8	8	7
11	9	10	10	10	9	8
14	12	13	13	14	12	10
17	15	15	15	16	14	13
20	16	15	15	16	14	13

D'après le **tableau 5.5** est la **figure 5.4**, il est évident que la qualité visuelle des vidéos attaquées n'a pas vraiment changé sachant que le PSNR des vidéos tatouées est de 47,83046 et pratiquement toutes les stratégies d'attaque mènent à un PSNR proche.

Tableau 5.5. PSNR de chaque copie attaquée à partir de K copies pirates pour différentes attaques

k	PSNR de chaque copie attaquée					
	AVG	Min	Max	Médian	MinMax	ModNeg
2	47,55232	47,51205	47,47991	47,77731	47,57850	45,35241
5	48,20606	47,09799	46,74872	48,17524	48,01760	45,82563
8	48,27959	46,60656	46,47176	48,23365	47,87766	45,45262
11	48,38360	46,57443	46,44868	48,29153	47,93604	45,36529
14	48,23243	46,49823	46,76548	48,30896	47,76963	45,48956
17	48,32434	47,01971	46,65081	48,26510	48,05366	45,29658
20	48,25537	46,66303	46,56021	48,30896	47,83352	45,57541

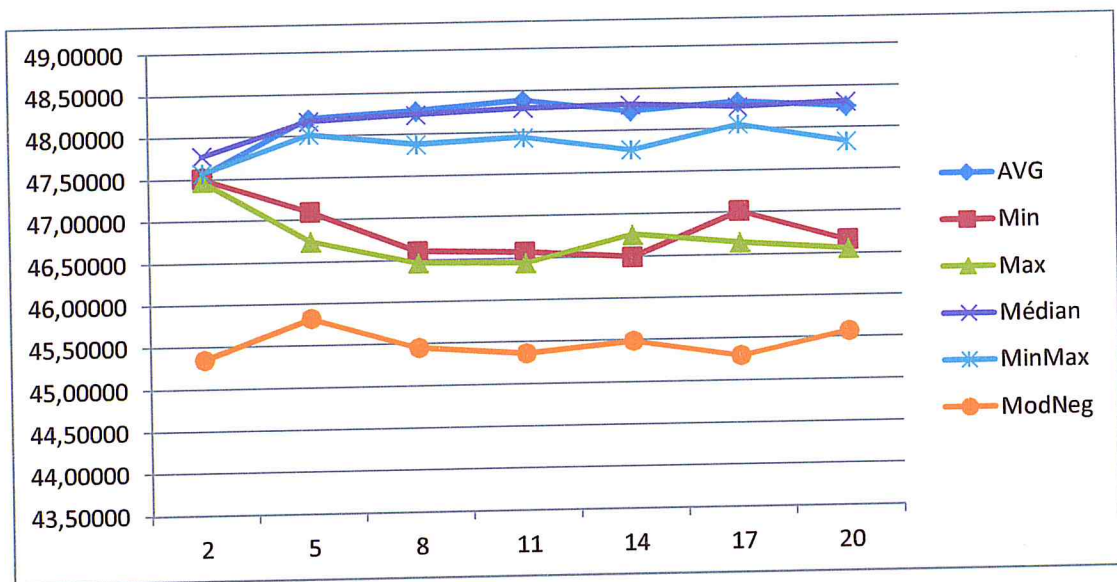


Figure 5.4. Diagramme de PSNR de chaque copie attaquée à partir de K copies pirates pour différentes attaques

Le **tableau 5.6** et la **figure 5.5** indiquent la similarité des vidéos, à partir de ces données il est évident que les vidéos sont presque similaires car la valeur de SSIM de la vidéo compressé non marquée est 0.99612 donc même avec ces attaques les vidéos reste en bonne qualité.

Tableau 5.6. SSIM de chaque copie attaquée à partir de K copies pirates pour différents attaques

k	SSIM de chaque copie attaquée					
	AVG	Min	Max	Médian	MinMax	ModNeg
2	0,99556	0,99553	0,99552	0,99555	0,99559	0,99526
5	0,99565	0,99550	0,99549	0,99561	0,99561	0,99515
8	0,99558	0,99548	0,99547	0,99564	0,99561	0,99513
11	0,99563	0,99549	0,99547	0,99564	0,99561	0,99515
14	0,99566	0,99549	0,99548	0,99563	0,99560	0,99514
17	0,99559	0,99550	0,99549	0,99564	0,99562	0,99516
20	0,99566	0,99548	0,99547	0,99563	0,99560	0,99514

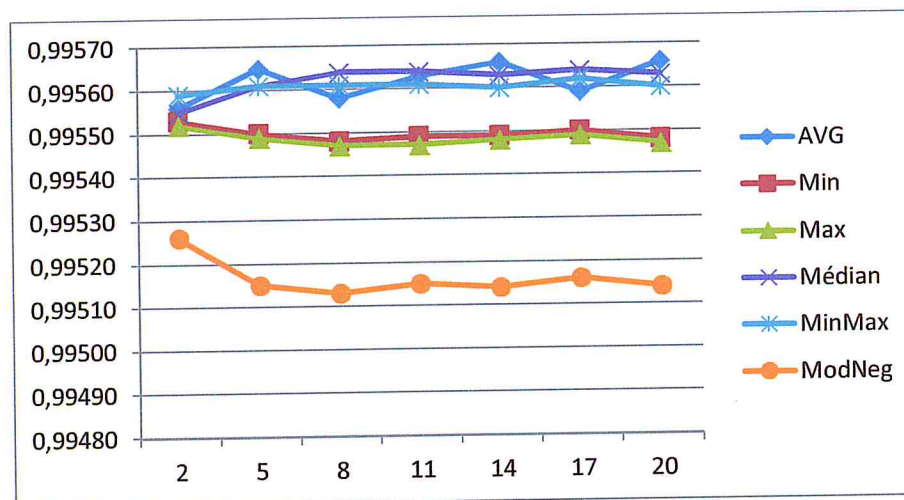


Figure 5.5. Diagramme de SSIM de chaque copie attaquée à partir de K copies pirates pour différents attaques

Pour l'attaque par compression, les tests effectués en appliquant cette attaque ont montré l'efficacité de système développé et résistance de la marque par l'extraction et la détection de l'utilisateur.

5. Conclusion

Dans ce chapitre nous avons exposé nos différents tests et expérimentations effectués ainsi que les résultats obtenus. Notre approche de traçage de copies des pirates dans le standard de compression H.264/AVC a prouvé son efficacité de traçage dans toutes les vidéos testées, nous avons augmenté la capacité d'insertion, et nous avons amélioré la visibilité ainsi que la robustesse.

Conclusion générale

Conclusion générale

L'étude que nous avons réalisée sur le traçage des copies des pirates des contenus dans la norme de compression H.264/AVC par le tatouage numérique se décompose en cinq parties.

La première expose l'apport de la technologie du tatouage numérique dans la sécurité du contenu multimédia. Le chapitre 2 se consacre à la présentation du standard de compression vidéo H.264/AVC, plus précisément, certains modules jugé nécessaire pour la réalisation de notre application. Le chapitre 3 propose un résumé non exhaustif des approches dédiées au traçage des traitres alors que le chapitre 4 donne une présentation détaillé de la réalisation du système de sécurisation de contenu pour le traçage des copies de pirates en utilisant un tatouage invisible et robuste.

Ce travail a mis en lumière d'un système de traçage de traitres par tatouage vidéo au sein de H.264/AVC. Le code de Tardos a été utilisé pour assurer le traçage des traitres d'une vidéo pirates obtenu par collusion.

Le système de tatouage et le code ont été conçus pour être réalistes dans le cas d'un très petit nombre d'utilisateurs ($n=100$). Aucune des stratégies de collusion n'a mis en défaut le processus d'accusation.

Il est évident que l'intégration d'un système de traçage de traitres dans une vidéo compressée n'est pas encore un problème totalement résolu. Cela dit, la combinaison de tatouage de code de Tardos par l'étalement de spectre au sein de H.264/AVC fonctionne bien et permet de mettre en lumière le travail qu'il reste à faire pour obtenir des systèmes plus matures.

Comme perspective à ce travail, nous proposons de changer le code de Tardos, par le code de Tardos amélioré, ainsi les composantes de chrominance dans le processus d'insertion.

Bibliographie



Bibliographie

- [1]. J.Chassery , " Tatouage d'image : Gain en robustesse et intégrité des images," thèse de doctorat de l'université de France, février 2003.
- [2]. M. WU, W. Trappe, Z. Janz Wang, K.J Ray Liu , " Multimedia Fingerprinting Forensics For Traitor Tracing, " EURASIP Book Series on Signal Processing and Communications, Vol.4, edition 2005.
- [3]. M. El-Gayyar , "Watermarking Techniques Spatial Domain Digital Rights Seminar," Media Informatics University of Bonn Germany, mai 2006.
- [4]. C.T. Li and F.M. Yang, "One-dimensional Neighborhood Forming Strategy for Fragile Watermarking," In Journal of Electronic Imaging, vol. 12, no. 2, pp. 284-291, edition 2003.
- [5]. P.Bas , "Methode de tatouage d'image fondées sur le contenu ," thèse de doctorat Institut National Polytechnique de Grenoble. Octobre 2005.
- [6]. T. Furon, " Application du tatouage numérique à la protection de copie," Ecole Nationale Supérieure des Télécommunications, Paris, mars 2002.
- [7]. G.Chareyron , "TATOUAGE D'IMAGES: UNE APPROCHE COULEUR," thèse de doctorat de l'université Jean Monnet Saint-Etienne, décembre 2005.
- [8]. D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," In Springer-Verlag, editor, *Crypt'95*, pp 452-165, 1995.
- [9]. B. Yasmina et B. Razika, "Protection de copyright dans la norme de compression vidéo H.264/AVC," thèse d'ingénieur en informatique, Université SAAD DAHLAB de Blida, Algérie, juin 2007.
- [10]. G. Said, "Authentification du contenu H.264/AVC le tatouage numérique et la signature numérique," thèse d'ingénieur en informatique, Université SAAD DAHLAB de Blida, Algérie, juin 2009.
- [11]. Y.M. Amine, "Le tatouage fragile dans les vecteurs de mouvement de la norme H.264," these d'ingernier en informatique, université SAAD DAHLAB Blida, Algérie, juin 2008.
- [12]. P. Menezes, V. Oorschot, and S. Vanstone, " Handbook of Applied Cryptography," CRC Press, 1996.
- [13]. Eskicioglu, "Multimedia security in group communications: Recent progress in key management, authentication and watermarking," ACM Multimedia Systems, Special Issue on Multimedia Security, vol. 9(3), pp. 239-248, September 2003.
- [14]. M. Wu, W. Trappe, J. Wang, and R. Liu, "Collusion-resistant fingerprinting for multimedia. IEEE Signal Processing Magazine," vol. 21(2), pp. 15-27, March 2004.
-

- [15]. T. Wiegand, G.J.Sullivan and A. Luthra, " Overview of the H.264/AVC Video Coding Standard," IEEE transactions on circuits and systems for video technology, vol. 13, no. 7, juillet 2003
- [16] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer and T. Wedi, "Video coding with H.264/AVC: Tools, Performance, and Complexity," IEEE Circuits And Systems Magazine, 2004.
- [17] R. Schäfer, T. Wiegand and Hh Schwarz, "The Emerging H.264/AVC Standard," EBU technical review, Jan 2003
- [18]. O. Rioul, "Codage entropique à longueur variable," ENST/COMELEC journal, 2003.
- [19]. M. G.J avier, "Optimisation des performances d'un encodeur suivant la norme Advanced Video Coding pour une machine vectorielle", thèse magistère, université libre de Bruxelles, 2006.
- [20]. B.Chor, A. Fiatand M. Naor, "Tracing traitors," In Springer-Verlag, ed Proc. Of Advances in cryptology, CRYPTO'94, Vol. 839, pp. 257–270, Springer-Verlag 1994.
- [21]. T.Furon, "Traçage de traître," Thomson Security Lab, Cesson-Sévigné, France. 2009
- [22]. D.R. Tinsonand R.Wei, "Combinatorial properties and construction of traceability schemes and frameproof codes," SIAM Journal on Discrete Mathematics, vol. 1, pp. 41–53, edition 1998.
- [23]. G. Tardos, "Optimal probabilistic fingerprint codes," In : Proceedings of the 35th annual ACM symposium on theory of computing, San Diego, CA, ACM, pp. 116–125, USA 2003.
- [24]. C.Peikert, A. Shelat and A. Smith, "Lower bounds for collusion-secure fingerprinting codes," In : Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), Baltimore,MY, pp.472–479, USA 2003.
- [25]. B. Skoric, T. Vladimirova, M. Celik and J. Talstra, "Tardos fingerprinting is better than we thought," IEEE Tran. on IT, vol. 54, edition 2008.
- [26]. K. Ait Saadi, A. Bouridane and A. Guessoum, "Combined Fragile Watermark and Digital Signature for H.264/AVC video Authentication," EUSIPCO 2009, Scotland, Glasgow 2009.
- [27]. G. Qiu, P. Marziliano, A. T. S. Ho, D. J. He and Q. B. Sun, "A hybrid watermarking scheme for H.264/AVC video," in Proc. 17th Int. Conf. Pattern Recogn., U.K., vol. 4, pp. 865-868, 2004.
- [28]. S. Ueda, H. Shigeno and K. I. Okada, "NAL Level Stream Authentication for H.264/AVC," IPSJ Transactions on Database, vol. 48, no. 2, , pp. 635-643, 2007.
-

- [29]. N. Ramaswamy and K. R. Rao, "Video Authentication for H.264/AVC using Digital Signature Standard and Secure Hash Algorithm," NOSSDAV'06, Rhode Island, USA, May 2006.
- [30]. Z. Shahid, M. chaumont and W. Puech, "Spread Spectrum-Based Watermarking For Tardos Code-Based Fingerprinting For H.264/AVC Video". ICIP 2010 SHAHID CHAUMONT PUECH Fingerprinting H264.
- [31]. J. Sullivan, P. Topiwala and A. Luthra, "The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions," SPIE Conference on Applications of Digital Image Processing XXVII, Special Session on Advances in the New Emerging Standard: H.264/AVC, 2004
- [32]. G. Zhu and N. Sang, "Watermarking algorithm research and implementation based on DCT block," World Academy of Science, Engineering and Technology, vol. 45, pp. 38–42, 2008.
- [33] I. Cox, J. Kilian, F. Leighton, and T. Shamoan, "Secure Spread Spectrum Watermarking for Multimedia," IEEE Transactions on Image Processing, vol.6, pp.1673–1687, 1997.
- [34]. P. Moulin and J. O'Sullivan, "Information Theoretic Analysis of Information Hiding," IEEE Transactions on Information Theory, vol.49, pp.563–593, 2003.
- [35]. F. Hartung, J. Su, and B. Girod, "Spread Spectrum Watermarking: Malicious Attacks and Counter attacks," in Proc. SPIE: Security and Watermarking of Multimedia Contents, pp. 147–158, 1999.
- [36]. T.T. Lu, W. Lun Hsu, and P. Chi Chang, "Blind Video Watermarking for H.264," in IEEE CCECE, pp. 2353-2356, May 2006.
- [37]. H. Vicky Zhao, M. Wu, Z.J. Wang, and K.J. Ray Liu, "Forensic Analysis of Non linear Collusion Attacks for Multimedia Fingerprinting," IEEE transactions on image processing, vol. 14, no. 5, pp 646-661, may 2005.
-
-

Sites web

- [Sw1]. <http://www.msnbc.msn.com/id/4037016/> visualisé en mars 2011.
- [Sw2]. <http://fr.wikipedia.org/wiki/H.264> visualisé en mars 2011.
- [Sw3]. <http://www.tdf.fr> (site officiel de TDF) visité en juin 2011.
- [Sw4]. <http://www.youtube.com/watch?v=L668OHqeuUM> vidéo visualisé en mars 2011.
- [Sw5]. <http://www.onsebuzz.com/divers/harry-potter-7-identification-du-piratage-grace-a-un-tatouage-numerique,201011191397.html>. Visité en juin 2011.
- [Sw6]. <http://www.vdocapture.com/WhatisCIF4CIFQCIFD1.htm>. visité en avril 2011.
- [Sw7]. <http://iphome.hhi.de/suehring/tml/>. Visité en février 2011.
- [Sw8]. http://compression.ru/video/quality_measure/info_en.html#start. Visité en juillet 2011.
- [Sw9]. http://www.lirmm.fr/~chaumont/publications/ICIP-2010_SHAHD_CHAUMONT_PUECH_watermarking_H264_Tardos_732_632_2450.yuv
vidéo téléchargé en mai 2011.
-
-