THÈSE DE DOCTORAT DE L'ÉCOLE NATIONAL SUPÉRIEUR DES TÉLÉCOMMUNICATIONS

Spécialité : Signal et Images.

présentée par

Yann BODO

pour obtenir le grade de

DOCTEUR de l'École National Supérieur des Télécommunications

Élaboration d'une technique d'accès conditionnel par tatouage et embrouillage vidéo basée sur la perturbation des vecteurs de mouvement.

soutenue le 29/09/04 devant le jury composé de :

M. Touradj EBRAHIMI	Rapporteur	Professeur / EPFL / Lausanne
M. Benoît MACQ	Rapporteur	Professeur / UCL / Louvain-la-Neuve
M. Jean-Marc CHASSERY	Examinateur	Directeur de Recherche / CNRS / Grenoble
M. Franck DAVOINE	Examinateur	Chargé de recherche / UTC / Compiègne
M. Philippe NGUYEN	Examinateur	Ingénieur de recherche / NEXTAMP / Rennes
M. Nathalie LAURENT	Examinatrice	Ingénieur de recherche / France Telecom R&D / Rennes
M. Jean-Luc DUGELAY	Directeur de thèse	Professeur / EURECOM / Sofia Antipolis

Remerciements

Voila, l'aventure se termine. Le chemin fut long et semé d'embûches, surtout qu'on ne nous avait pas prévenu que l'ennemi était dans notre camp, poukram !

En premier lieu, je tiens à remercier madame Nathalie Laurent et monsieur Christophe Laurent, sans qui cette thèse n'aurait jamais pu aboutir. Ce sont des personnes aux qualités humaines exceptionnelles, leurs remarques éclairées et pertinentes ainsi que leur soutien quotidien m'ont permis de mener à bien cette aventure. Un grand merci à vous deux.

Je tiens à exprimer mes remerciements aux membres du jury :

monsieur Benoît Macq et monsieur Touradj Ebrahimi pour m'avoir fait l'honneur de rapporter cette thèse. Je les remercie d'avoir contribué, par leurs remarques pertinentes, aux nombreuses améliorations de ce manuscrit. Leurs suggestions m'ont été d'une grande aide pour évaluer mon travail d'un autre point de vue.

Monsieur Jean-Luc Dugelay pour avoir accepté de diriger cette thèse.

Monsieur Jean-Marc Chassery pour avoir accepté de présider ce jury.

Monsieur Franck Davoine et monsieur Philipe Nguyen pour avoir accepté d'être examinateurs avec tant d'enthousiasme. Je les remercie pour leur gentillesse et leurs conseils lors de la phase de rédaction.

Je désire aussi remercier monsieur Vincent Marcatté et monsieur Henri Sanson pour m'avoir accueilli au sein du laboratoire HDM, et plus précisément dans l'unité de recherche CIM. Je les remercie de m'avoir accordé tant de liberté dans l'accomplissement de mes travaux. Je tiens à remercier plus particulièrement monsieur Henri Sanson pour la confiance qu'il place en ses thésards.

Un grand merci à Stéphane Pateux pour ses conseils et son aide sur la fin de cette thèse.

J'en profite pour remercier l'ensemble des personnes de l'équipe : Mariette, David, Patrick x 2, et plus généralement l'ensemble des permanents qui ont croisé mon chemin.

Enfin, je salue l'ensemble des thésards de France Télécom avec qui j'ai passé de très agréables moments : Ricardo, Ewa, Elise, Julien, Laurent, Gaspard, Olivier, Muriel, Magalie, Alexandre, Raphaële ainsi que tout ceux que j'oublie !

Bien évidemment l'ensemble des remerciements ne peut se limiter au simple cadre professionnel, je tiens également à remercier Greg, Alex, Violaine, Caro, Solenn ...

Un grand merci à Fif l'américain, "The Colloc" !

Un grand merci à "Nicolas le jardinier", plus connu sous le nom d'Ewen, "the colloc le retour" !

Un grand merci à mister Seb, pour les soirées, son aide quotidienne, pour le permis moto aussi,

sa patience pour les relectures et les présoutenances, la fin est proche, faut tenir bon, t’as fait le plus dur !

Spéciale dédicace à DJ Duval, docteur es pâtisserie option hardtek.

Je tiens aussi à remercier l’ensemble de la troupe parisienne : Seb, Yann, Cécé, Nikoz, Ludo, Vaness, Enora et tout ceux que j’oublie, see you soon.

Un énorme merci aux trois compères de toujours : Nico, Dadou, Mat qui ont toujours répondu présent dans toutes les circonstances.

Un big merci à la miss Anne-Thida, ma conseillère juridique préférée.

Avant de finir, un grand merci à Heretik, FKY, Spiral Tribe, Snot, Slipknot, Prodigy, le monde bédélistique et tous les autres qui m’ont aidé tout au long de la rédaction de ce manuscrit.

Bien évidemment je remercie mes parents et ma famille pour leur confiance et leur soutien sans faille au cours de toutes ces années, et pour m’avoir supporté pendant ces longues études.

J’ai sans doute oublié plein de monde, toutes mes excuses, je sais que vous ne m’en tiendrez pas rigueur.

THE END !

Table des matières

Résumé

L'essor du multimedia et des supports numériques, qui permettent l'obtention d'une copie parfaite, conduisent de plus en plus les fournisseurs de contenus à se poser le problème du piratage des données. La protection et le tatouage des media numériques sont rapidement devenus un axe majeur de recherche dans le domaine du traitement d'images.

La problématique que nous posons dans cette thèse est de proposer une solution globale de protection des contenus vidéo. Pour ce faire, nous proposons deux approches fondées sur une perturbation des vecteurs de mouvement. La première consiste à se servir des vecteurs de mouvement pour embrouiller une séquence. La seconde, quant à elle, utilise aussi les vecteurs de mouvement afin d'insérer une marque invisible et robuste. Ces deux solutions peuvent être ainsi combinée afin de former un système de protection globale d'une vidéo.

Nous proposons dans le chapitre 1, un état de l'art sur les techniques d'embrouillage et les motivations amenant à développer de tels systèmes. Dans le chapitre 2, nous présentons les principes généraux du tatouage ainsi qu'un aperçu des principales techniques utilisées en tatouage vidéo. Ensuite, nous poursuivons au chapitre 3 sur la présentation d'une solution d'embrouillage, basée sur la perturbation des vecteurs de mouvement d'un flux vidéo compressé suivant la norme MPEG4. Nous poursuivons ensuite sur l'élaboration d'un algorithme de tatouage dans le domaine non compressé, basé sur la perturbation locale des vecteurs de mouvement, au sein de la vidéo. Dans le chapitre 4, nous examinons la problématique de l'invisibilité de notre approche, en relation avec les problèmes soulevés au chapitre précédent. Dans cette optique, nous présentons un état de l'art sur les caractéristiques du système visuel humain, pour poursuivre sur un exposé des principales métriques utilisées en vidéo. Nous proposons en fin de chapitre les algorithmes que nous avons mis en place, permettant de minimiser l'impact visuel de notre système de tatouage. Enfin, dans le chapitre 5 nous concluons sur notre approche, et présentons les différentes perspectives d'évolution de notre algorithme.

Abstract

Development of a conditional access technique by video watermarking and video scrambling based on the disturbance of motion vectors.

The recent development of digital multimedia leads the providers to deal with the hacking of multimedia contents. The digital protection of the original media has become a major research issue in the image processing area.

In order to protect a video, three main techniques could be used :

– cryptography ;
– scrambling ;
– and watermarking.

In this PhD-thesis, we focus essentially on scrambling and watermarking techniques.

The first contribution exposed in this PhD-thesis is a novel system of video scrambling. This system use motion vectors for the protection of a digital video in order to disturb the video content before its transmission to an user. Then, to complete our system, we propose to insert an invisible watermark in the digital video contents. This algorithm is also built on a local disturbance of motion vectors.

Before introducing our algorithms of scrambling and watermarking, we first propose in chapter one an overview on scrambling techniques. Then in chapter two, we expose the principles of watermarking, but also an outline of the main techniques used in video watermarking. In chapter three, we present our solution of video scrambling based on a visible disturbance of the motion vectors of a video stream, compressed using an MPEG4 codec. Then in chapter three, we pursue on a novel watermarking algorithm in the uncompressed domain, based on the local disturbance of the motion vectors and we present the drawbacks of the motion vectors perturbation. This algorithm is mainly based on a defined reference grid applied in the insertion rules. In chapter four, we explore the invisibility of our approach, in relation with the several issues raised in the previous chapter. Accordingly, we present a state of the art on the human visual system, and the main quality metrics applied in video. From this, we show how we developed our specific algorithms, which allow minimizing the visual impact of the watermarking procedure. At last, we conclude in chapter five on our approaches, and present the possible evolutions and improvements of our algorithms.

TABLE DES MATIÈRES

Introduction générale

Les applications du traitement d'images sont multiples et interviennent dans de nombreux aspects de la vie courante et professionnelle. Avec l'ère de l'information, de l'internet haut-débit, de l'audiovisuel et du numérique, l'expansion et la circulation des supports multimedia ont beaucoup augmenté, amenant les ingénieurs et les chercheurs à optimiser et à élaborer des techniques de plus en plus complexes afin d'augmenter la qualité, de diminuer la taille et de protéger ces media. Qu'il s'agisse d'image, de son ou de vidéo la problématique est la même. Cependant, les moyens mis en oeuvre pour atteindre ces différents objectifs, bien que proche théoriquement, font appel à des techniques bien distinctes. Ces techniques sont utilisées pour la télévision, pour la vidéo en générale (dvd, cinéma, caméra ...), pour la photographie, dans le monde de l'édition, dans le transport de l'information, dans l'archivage etc. Certains domaines sont plus confidentiels, comme le domaine militaire où ces techniques sont utilisées dans la surveillance, le guidage automatique, la poursuite d'engins ou encore en topographie. D'autres domaines spécialisés font également appel au traitement d'images, on citera notamment l'imagerie aérienne et spatiale (surveillance, météorologie, ...), la médecine (cytologie, tomographie, echographie ...), mais également de nombreux domaines scientifiques (astronomie, robotique mobile, biologie ...).
Le traitement d'images (plus particulièrement le traitement d'images numériques) est un domaine encore jeune. C'est en 1920 que la première image est transmise par câble de New-York à Londres en quelques heures, cependant le traitement d'images voit son origine seulement vers les années cinquante avec l'analyse d'images dans les chambres à bulles (Rayons X, OCR, ...). A cette époque, les images numériques sont de mauvaise qualité et très volumineuses (700*500 pixels sur 8 bits par image). Enfin en 1960, apparaissent trois domaines dominants en traitement numérique d'images :

- la restauration qui consiste à corriger les défauts liés à l'acquisition ;
- l'amélioration qui consiste à rendre l'image de meilleure qualité pour l'affichage ;
- et la compression qui consiste à réduire le volume de l'image.

En 1970, les recherches se focalisent sur l'extraction automatique d'informations. On voit alors l'apparition de la notion de description structurelle ainsi que l'émergence de nouveaux thèmes comme le seuillage, la segmentation, l'extraction de contours, la morphologie mathématique. Des tentatives d'interprétation d'images basées sur l'exploitation de systèmes experts conduisent à des échecs, essentiellement dû à la trop grande complexité des images et à la non prise en compte des aspects perceptuels. Enfin en 1980, le traitement d'images industriel explose réellement grâce au développement de la micro-informatique et de capteurs. On passe de l'image 2D aux modèles

tridimensionnels. On commence à s'intéresser à l'analyse du mouvement et à la vision pour la robotique (mouvement, 3D, détection d'obstacle, trajectoire). En 1990, les transmissions de données connaissent un essor très important supporté par la croissance de l'internet. On s'oriente alors vers la prise en compte de l'observateur dans l'analyse de la scène (passage de l'étude de la vision passive à la vision active). Arrive enfin l'apparition du tatouage, tout d'abord timidement, pour ensuite se développer massivement vers 1995. En effet, avec le développement du tout numérique, s'est posée la problématique de protéger les contenus multimedia. Dans ce contexte le tatouage apparaît comme étant une alternative pouvant s'avérer efficace et complémentaire aux approches de type cryptographique.

Dans cette thèse, nous aborderons les aspects de la protection de la vidéo.
Plus précisément, cette thèse est composées des chapitres suivants :

- chapitre 1 : nous présentons différentes techniques d'embrouillage utilisées dans les systèmes de contrôle d'accès ainsi que les bases de la cryptographie ;
- chapitre 2 : après avoir exposé les principes de l'embrouillage et de la cryptographie, nous poursuivons sur un état de l'art décrivant les principes généraux du tatouage ainsi que sur les algorithmes de base de tatouage vidéo ;
- chapitre 3 : nous proposons dans un premier temps une solution d'embrouillage, inspirée des techniques de tatouage. L'enjeu de notre approche consiste à perturber les vecteurs de mouvement au sein d'un flux MPEG4 en réalisant des variations ajustables et réversibles. Ceci afin de permettre une visualisation dégradée et progressive des vidéos pour la mise en place de services en ligne associés à des achats impulsifs. Dans un second temps, nous présentons de manière détaillée, l'algorithme de tatouage que nous avons élaboré au cours de cette thèse afin de pouvoir concevoir une protection complète, par la mise en oeuvre de ces deux procédés, d'une vidéo. Nous exposerons en premier lieu le principe fondamental de notre méthode, la mise en place d'une grille de référence permettant de partitionner l'espace des vecteurs de mouvement sur lesquels nous appliquerons notre règle d'insertion. Nous poursuivrons en présentant les différentes améliorations successives apportées à cet algorithme ;
- chapitre 4 : après avoir présenté les bases de notre algorithme, nous proposons dans ce chapitre de nous intéresser aux particularités du système visuel humain en présentant les différents aspects de notre perception visuelle et en présentant différentes métriques prenant en compte ces principes pour contrôler la qualité des vidéos. Nous examinerons ensuite la possibilité d'utiliser ces principes pour optimiser les algorithmes de tatouage. Enfin nous exposerons nos travaux concernant l'optimisation de l'invisibilité de notre méthode ;

– chapitre 5 : pour finir, nous proposons dans ce chapitre de présenter les différentes perspectives d'études et d'améliorations de notre algorithme de tatouage.

Dans cette thèse, nous avons proposé différents états de l'art qui dépassent parfois le cadre applicatif nous ayant servi à élaborer nos algorithmes. Cependant, pour des raisons de perspectives et pour une meilleure compréhension du contexte général de cette thèse, il nous est apparu nécessaire de les présenter dans leur globalité.
Le but de cette thèse était d'élaborer un algorithme de protection globale. Nous avons choisi d'utiliser les vecteurs de mouvement comme support pour développer nos algorithmes. En effet, cela nous a permis d'envisager de combiner une solution d'embrouillage et une solution de tatouage utilisant le même support. En marge du tatouage, nous avons donc étudié les principes d'embrouillage afin de proposer une solution complète de protection des contenus vidéos, en combinant une protection a priori et une protection a posteriori. Ces deux éléments sont à ce jour indépendant, mais le support étant le même nous présenterons en perspective la possibilité de corréler ces deux approches.

TABLE DES MATIÈRES

Définitions

Les définitions proposées ici sont très succinctes et sont données à titre indicatif, le lecteur intéressé trouvera plus de détails concernant ces techniques dans le chapitre 1 et 2.

EMBROUILLAGE

Opération destinée à transformer un signal numérique en un signal numérique aléatoire ou pseudo-aléatoire, de même signification et de même débit binaire, en vue d'en faciliter la transmission ou l'enregistrement. *(source : ancien arrêté du 27 avril 1982 (J.O. du 24 juin 1982))*

CRYPTOGRAPHIE ou CHIFFREMENT

Technique des écritures secrètes.

STEGANOGRAPHIE

Méthode visant à camoufler de l'information ou une signature à l'intérieur d'un fichier multimedia (son, image). Par exemple, en introduisant une information sur le bit le moins significatif de chaque couleur d'un pixel d'image. Certes, il y a dégradation des couleurs, mais cela reste généralement imperceptible à l'oeil.

WATERMARKING ou tatouage numérique

Un "watermark" est un bloc d'informations inclus dans une création numérique, invisible en usage normal, contenant des informations d'identification de l'oeuvre, afin de protéger celle-ci. Le "watermarking" se différencie de la stéganographie par la nécessité de répondre à des contraintes de robustesse, et par le fait qu'en "watermarking" l'oeuvre protégée est importante.

TABLE DES MATIÈRES

Chapitre 1

État de l’art sur les méthodes d’embrouillage

1.1 Introduction

Afin de transmettre de manière sécurisée des contenus audio-visuels, de nombreux industriels ont développé des méthodes d'embrouillage faisant appel aux techniques de cryptographie, ainsi qu'aux techniques de traitement du signal. L'intérêt d'utiliser de tels procédés est de rendre économiquement viable les applications d'accès conditionnels, qui permettent la mise en place de systèmes de télévision à péage, dont le déploiement s'est amplifié ces dernières années. D'autres applications peuvent être envisagées pour les systèmes de contrôle d'accès : vidéo-conférence confidentielle, transmissions de fac-similés confidentiels, ou encore transmission et stockage d'images médicales dans une base de données. Cependant les deux problèmes majeurs des approches cryptographiques concernent le chiffrement du contenu vidéo et la complexité de ces approches. Effectivement, interdire la visualisation du média dans un service de type kiosque, empêche de susciter l'envie chez les utilisateurs. Enfin, la complexité de ces approches (par exemple l'algorithme RSA) ne permet pas d'atteindre une exécution en temps réel de l'application, ce qui est une contrainte majeure dans ce contexte.
De ce fait en section 1.2, nous allons poser le cadre dans lequel évolue l'embrouillage, en introduisant les grands principes de la télévision numérique. Ensuite en section 1.3, nous introduisons succinctement le vaste domaine que représente la cryptographie pour enfin donner les définitions fondamentales de l'embrouillage en section 1.4. Une fois le contexte fixé, nous présenterons pour finir en section 1.5 différents systèmes d'embrouillage, afin de mieux en cerner la problématique.

1.2 La télévision numérique

La production des images de télévision sous forme numérique s'est répandue depuis le début des années 90, grâce aux avantages qu'elle apporte aux producteurs de programmes (P. Pirat [122]) :

- fidélité des sources après enregistrement ou duplication ;
- traitements de post-production aisés et sans détérioration de la qualité des images ;
- et enfin transmission sans erreur sur des liaisons numériques de contribution entre studios.

La numérisation, en raison des coûts élevés de matériels qu'elle induit, serait vraisemblablement restée cantonnée dans les sphères de la production sans la double impulsion apportée par les techniques de compression de l'image, les progrès en densité et rapidité des circuits intégrés. Initialement destinés à la réduction du débit des programmes échangés entre studios de télévision sur des liaisons de contribution, de puissants algorithmes de compression de débit ont été développés vers

le milieu de la décennie 1980 dont les principes essentiels ont été retenus par le groupe de standardisation MPEG (pour MPEG1 et MPEG2) pour des applications de moindre qualité : stockage sur CD-ROM à 1,5 Mbit/s et distribution entre 4 et 10 Mbit/s.
Disposant ainsi de signaux numériques à débit modéré, de normes mondiales de compression et de techniques d'intégration pour réaliser des circuits à des coûts abordables, la question de l'acheminement des programmes numériques vers les usagers s'est alors posée. Des procédés permettant la modulation des trains numériques dans les canaux offerts par les supports de diffusion satellite, câble, terrestres hertziens et MMDS (Multipoint Multichannel Distribution System) ont été développés à partir de 1992, sur l'initiative du groupement européen DVB (Digital Video Broadcasting). Ces procédés ont, pour certains, donné lieu à des normes européennes. Ces techniques de modulation numérique sont associées à des techniques de codage canal destinées à augmenter la robustesse des signaux numériques diffusés. La modulation et le codage canal sont adaptés aux caractéristiques du support de diffusion. Ils permettent d'optimiser son utilisation. Le résultat se concrétise aujourd'hui par la distribution de 4 à 10 programmes de télévision comprimés dans un canal.
La télévision numérique bouscule donc le traditionnel paradigme "un programme par canal", en permettant à plusieurs programmes de partager le même canal de diffusion. La première conséquence en est l'introduction dans la chaîne de transmission d'un nouveau maillon, chargé de grouper ou de multiplexer les programmes qui partagent le même canal. La seconde, et non la moindre, est l'apparition d'un nouvel intervenant : l'opérateur de multiplex, qui peut être conduit à gérer l'allocation des canaux aux groupes de programmes, ainsi que les abonnements aux programmes à péage comme le recommande le "Livre blanc sur la télévision numérique terrestre" [2] édité par l'administration britannique.
Ces nouvelles techniques de télévision numérique qui, outre la numérisation, couvrent également la compression (codage source), le multiplexage, le transport, ainsi que la diffusion des signaux de télévision, offrent aux opérateurs ainsi qu'aux téléspectateurs les perspectives suivantes :

- augmentation du nombre de programmes diffusés par l'effet conjugué de la compression des composantes de programme, et une meilleure utilisation du spectre autorisé par les nouvelles techniques de modulation numérique ;
- augmentation de la qualité technique des programmes depuis le format 16/9, jusqu'à la télévision à haute définition ;
- introduction de nouveaux services tels que les programmes à péage et le "pay per view" associés à une grande souplesse dans la composition et la recomposition des bouquets de programmes ;
- introduction de récepteurs audiovisuels portables voire mobiles, offrant une qualité constante

de restitution des programmes ;
- ouverture vers des services multimedia par le mariage de composantes de natures différentes, telles que l'image, le son et les données, dans un même flux d'éléments binaires, qui utilisent les mêmes organes de traitement, transport, diffusion et réception ;
- et enfin, l'introduction de nouveaux services tels que les programmes à la demande (VOD : Video On Demand), la programmation différée (NVOD : Near Video On Demand) et, plus généralement, les services interactifs qui individualisent l'accès des spectateurs aux sources d'information.

Pour résumer, un système de télévision numérique est, comme proposé dans [113], constitué des opérations suivantes :
- génération de signaux numériques compressés ;
- génération de données ;
- constitution d'un multiplex (d'un flux binaire) en combinant plusieurs signaux audios et vidéos (typiquement entre 4 et 10 programmes de télévision) et des données ;
- l'embrouillage et les messages de contrôle d'accès, pour rendre les programmes inintelligibles à toute personne n'ayant pas le droit de les recevoir (par exemple pour restreindre la réception aux abonnés d'un programme payant) ;
- diffusion du multiplex par satellite, câble, hertzien terrestre, MMDS ;
- côté terminal, les fonctions duales : réception du flux binaire diffusé, démultiplexage et désembrouillage, décompression de l'audio et de la vidéo, traitement des données.

Il existe deux grands systèmes concurrents qui diffèrent surtout par les modulations employées :
- le DVB-T en Europe et en Australie ;
- l'ATSC aux États-Unis.

1.2.1 Travaux OFDM sur les techniques de télévision numérique en Europe

L'OFDM (Orthogonal Frequency Division Multiplex) résulte du travail effectué par les militaires américains sur la modulation multiporteuses (MCM : MultiCarrier Modulation). La base du MCM consiste à partager les données en un certain nombre de flots parallèles, qui modulent un grand nombre de porteuses différentes. L'OFDM surmonte les problèmes potentiels associés à l'utilisation d'un grand nombre de porteuses de fréquences proches, sans l'utilisation à la réception de filtres passe-bande à flancs raides, onéreux, et ce, grâce à un traitement numérique (initié en France au CCETT : Centre Commun d'Études de Télédiffusion et de Télécommunication), en générant des porteuses orthogonales (O de OFDM). Les travaux européens sur la télévision

numérique, lancés depuis 1990, ont abouti en 1993 à la création d'une organisation nommée Digital Video Broadcasting Project (DVB Project) rassemblant des diffuseurs, des constructeurs, des régulateurs, etc. Les signataires ont adopté la standardisation des spécifications ISO/MPEG-2, standard mondial pour le codage vidéo, audio et le multiplexage des signaux. DVB a défini des spécifications de diffusion sur câble (DVB-C), satellite (DVB-S), et terrestre (DVB-T). Par ailleurs, le forum DVB a permis de normaliser un algorithme commun d'embrouillage et de désembrouillage, à partir d'un signal en clair et d'une clé secrète. DVB a normalisé également les informations sur les services (SI : Système d'Information), les protocoles pour les services interactifs, les voies de retour par ligne téléphonique et par certains supports physiques (réseaux câblés). L'ensemble des spécifications a été normalisé en 1997, par les organismes de normalisation ETSI (European Telecommunication Standards Institute), pour les normes concernant les signaux et les protocoles, et le CENELEC (Comité Européen de Normalisation en Électronique) pour les normes concernant les équipements. Les organismes français (centres d'études, industriels, opérateurs de contenus, opérateurs de réseaux) ont largement contribué à ces travaux. En France, le 24 juillet 2001, le CSA (Comité Supérieur de l'Audiovisuel) a publié la liste des fréquences identifiées pour la première phase des 29 sites couvrant 50% de la population, et a lancé un appel à candidature pour l'attribution aux opérateurs. Le 1er février 2002, le décret "Must Carry" impose aux cablo-opérateurs de transmettre sur câble ou sur satellite, un contenu identique à la télévision numérique terrestre TVNT.

1.2.2 Situation américaine et contexte historique du 8-VSB

VSB (Vestigial Side Band) est une forme de modulation d'amplitude (AM). L'utilisation de signaux de synchronisation, en dehors des signaux de données, en font un système robuste, permettant de synchroniser l'image, même si les données sont corrompues. Un consortium nommé la "Grande Alliance" a été formé en 1993, pour faire pression sur le FCC (Federal Communication Commission) afin d'adopter le système VSB comme norme digitale pour la télévision standard (STV : Standard TV), et pour la télévision haute définition (HDTV : High Definition TV). Ce consortium était constitué par les groupes AT&T, Centre de Recherche David Sarnoff, Général Instruments, Massachusetts Institute of Technology (MIT), Philips Amérique du Nord, Thomson Électronique grand public et Zénith. Après avoir réalisé divers essais, ces groupes ont recommandé l'utilisation du codage en treillis, une modulation 8-VSB pour la diffusion terrestre et 16-VSB pour une transmission par câble. Le FCC a voté unanimement le 3 avril 1997 l'attribution gratuite de canaux "numériques" à l'ensemble des 1 600 stations de télévision. Afin d'accélérer la transition, le FCC a adopté en 1997 les décisions suivantes :

– les principaux réseaux seront tenus de diffuser un signal numérique avant deux ans (fin

1999), sur les 30 marchés les plus importants, correspondant à 30% des foyers et croissant jusqu'à 50% en l'an 2000 ;
– la transition devra être établie en mai 2002, pour toutes les chaînes commerciales (au nombre de 1200), et en mai 2003, pour toutes les chaînes publiques (au nombre de 400).

Les obligations de transferts s'accompagnent aussi d'attribution de fréquences supplémentaires. Le transfert de l'analogique vers le numérique devrait s'étaler sur une dizaine d'années. Donc, à partir de 2006, ou lorsque 85% des foyers seront numérisés, toutes les émissions analogiques seront terminées, et les fréquences seront alors disponibles. La largeur de bande ainsi libérée sera de 138 MHz. Les diffuseurs devront émettre au moins un programme en clair, avec une obligation de simulcast (diffusion d'un programme sur différents réseaux de transmission) pendant les 6 à 8 premières années. Les opérateurs pourront diffuser des services nouveaux de télévision payante. En outre, la télévision numérique hertzienne sera utilisée pour offrir un service de télévision haute définition. Aujourd'hui, aux États-Unis, 250 émetteurs sont en service et couvrent 85 bassins de diffusion. Enfin, on note que la norme DVB fait son entrée aux USA pour la diffusion directe par satellite.

1.2.3 La diffusion numérique au Japon

Le Japon fait figure de pionnier en matière de nouvelle technologie, c'est tout naturellement, que ce pays a vu rapidement l'émergence de la télévision numérique.
Le standard japonais pour la HDTV se nomme HIVISION. Ce format est composé de 1125 lignes à une fréquence de 60 images par secondes. Ce système utilise une technique appelée MUSE (MUltiple sub-nyquist Sampling Encoding). Cette technique est similaire au DATV (Digitally Assisted TV). Cela permet de réduire l'information qui doit être transmise en envoyant des informations sur le mouvement image par image. Différents taux de rafraîchissement sont utilisés pour les blocs de l'image(1000 pixels), ce taux varie entre 20, 40 ou 80 milliseconde selon l'importance du mouvement[1].
L'opérateur satellite japonnais Space Communications Corp a sélectionné le système Nagravision en coopération avec Matshushita (MITC) pour développer un réseau de télévision numérique au Japon. Ceci représente le premier contrat japonais pour Nagravision[2].
En coopération avec NEC et Toshiba, Matshushita a développé un système d'embrouillage pour le réseau domestique. Parallèlement, matshushita a développé son propre système d'embrouillage

[1] http ://members.aol.com/jimb3d/tv/tv.html
[2] http ://www.nagra.com

appelé M-scrambler pour la télévision privée via satellite. Les services DAB (Digital Audio Broadcasting) ont commencé au Japon en 1993. La diffusion de la télévision numérique via satellite a débuté quant à elle en 1998[3].

Il est aisé de constater, aujourd'hui, que le fossé qui séparait le monde de l'audiovisuel et celui de l'informatique tend à se combler, par la mise en place de systèmes numériques, vers lesquels convergent les activités des industries de l'information, des télécommunications et de la radiodiffusion.
Le rapprochement de ces 3 disciplines à ouvert la porte a de nouvelles possibilités de programme, de service et de personnalisation, mais par conséquent, il est devenu plus facile de violer la propriété intellectuelle des contenus véhiculés. Les moyens informatiques mis à la disposition du grand public, l'augmentation des débits au niveau de l'internet domestique et la sophistication des appareils électroniques grand public, ont vu l'explosion de formats comme divx, ou de systèmes de décodage "software" pirates. Dans ce contexte, il apparaît nécessaire de protéger les contenus mis à la disposition du grand public. Pour résoudre ce problème, il existe deux niveaux de protection, une protection a priori, et une protection a posteriori. La première, correspondant à l'embrouillage et au cryptage, pose un problème : une fois les données reconstruites, cette protection n'est plus efficace. La seconde, est apparue au début des années 90 (d'abord timidement pour se développer au milieu de cette décennie). Il s'agit du tatouage, que nous étudierons en détail dans la suite de cette thèse. En section 1.3 et 1.4, nous allons dans un premier temps, fixer le contexte de l'embrouillage, en présentant quelques définitions et quelques principes de base, auxquels doit répondre un système d'embrouillage. Nous proposerons également une introduction à la cryptographie. Nous présenterons ensuite en section 1.5, les principaux systèmes utilisés actuellement, en mettant en valeur la faiblesse de ces algorithmes, et en présentant plus particulièrement le moyen de "cracker" le système Nagravision, système d'embrouillage très connu en France pour être utilisé par Canal+.

1.3 Introduction à la cryptographie

La cryptographie ou chiffrement est un processus de transcription d'une information intelligible, en une information inintelligible, par l'application de conventions secrètes, dont l'effet est réversible (V. Sedallian et G. Mathias [155]). La loi française définit les prestations de cryptologie de la manière suivante : "toutes prestations visant à transformer à l'aide de conventions secrètes des informations ou signaux clairs en information ou signaux inintelligibles pour des tiers, ou à

[3] http ://www.wtec.org/loyola/satcom/ac_mats.htm

réaliser l'opération inverse, grâce à des moyens, matériels ou logiciels conçus à cet effet".
Il s'agit d'une définition générale qui englobe également la stéganographie, technique particulière qui consiste à cacher un message dans un autre message d'apparence anodine. Les signatures numériques et le chiffrement constituent les deux applications principales de la cryptographie. Les signatures numériques permettent de prouver l'origine des données (authentification) et de vérifier si ces données ont été altérées (intégrité). Le chiffrement aide à maintenir la confidentialité des données et des communications. Enfin, la cryptographie peut aussi bien assurer la protection de données stockées dans un ordinateur, que transmises dans le cadre d'une communication.

1.3.1 Deux grands types de chiffrement

Lorsque l'on parle de cryptographie, on peut distinguer deux grandes familles de protocoles.

Le chiffrement symétrique (ou chiffrement à clés secrètes)

Dans ce type de chiffrement, la même clé est utilisée pour chiffrer et déchiffrer l'information. Le problème de cette méthode est de trouver le moyen de transmettre de manière sécurisée la clé à son correspondant. L'algorithme de chiffrement symétrique le plus populaire est l'algorithme DES (cf. figure 1) qui présente un bon compromis entre complexité et sécurité.

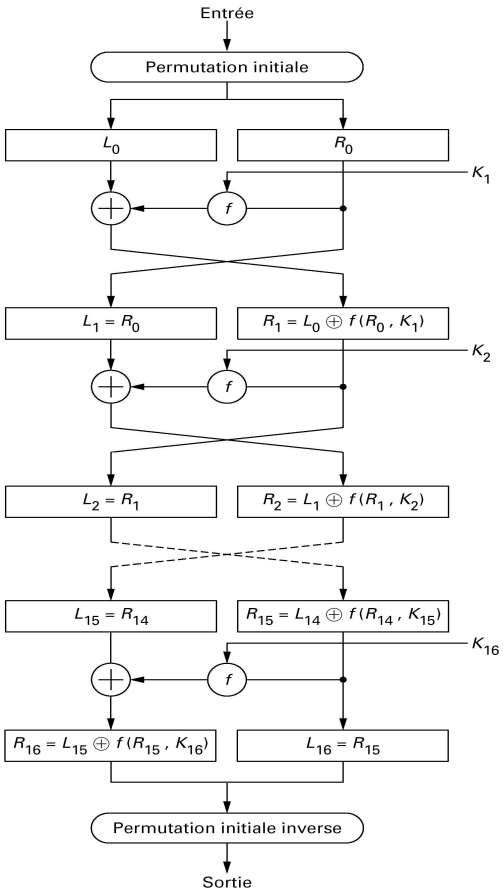


FIG. 1 – Schéma bloc de l'algorithme DES décrit ci-dessous

Sa description peut être trouvé dans FIPS 46-2 (Federal Institute Processing Standards) du NIST (National Institute of Standards and Technology). Le DES utilise des blocs de 64 bits, et une clef formée de 64 bits dont 56 utiles et 8 de parité. La transformation DES est en fait la composition de 16 transformations élémentaires paramétrées par des sous-clefs (K_1, K_2...K_{16}) extraites de la clef initiale. La transformation élémentaire T_i, paramétrée par K_i est la suivante :

- le bloc d'entrée de 64 bits est découpé en deux blocs de 32 bits L_i et R_i ;
- et on applique la relation suivante :

$$L_{i+1} = R_i \; R_{i+1} = L_i \oplus f(R_i, K_i)$$

Cette transformation est quasi involutive et ce, quelle que soit la nature de f. En effet, il suffit d'inverser le rôle de R et L pour trouver la transformation inverse :

$$R_i = L_{i+1}$$
$$L_i = R_{i+1} \oplus f(L_{i+1}, K_i)$$

Il est facile de vérifier que l'algorithme de déchiffrement est identique à celui de chiffrement, à condition d'utiliser la séquence des sous-clefs dans l'ordre inverse (K_{16}, K_{15} ... K_2, K_1).

Le chiffrement asymétrique (ou chiffrement à clés publiques)

Ici, ce n'est pas la même clé qui chiffre et qui déchiffre les messages. L'utilisateur possède une clé privée (appelée encore clé secrète) et une clé publique. Il distribue sa clé publique et garde secrète sa clé privée. Dans ce type d'application, tout le monde peut écrire à l'utilisateur destinataire en utilisant la clé publique, mais lui seul pourra déchiffrer et donc lire le message avec sa clé privée. La cryptographie permet ici d'assurer la confidentialité des données transitant sur un réseau : les données sont uniquement portées à la connaissance des personnes autorisées. Une autre paire de clés ou la même paire est utilisée pour s'assurer de l'identité de l'émetteur d'un message : c'est la question de l'authentification. L'utilisateur chiffre avec sa clé privée son message. Tout le monde peut déchiffrer le message avec la clé publique correspondant à l'expéditeur ainsi identifié. Les méthodes de chiffrement asymétriques reposent sur des calculs mathématiques sophistiqués, utilisant des nombres premiers, gérés par des algorithmes dédiés. Il est facile de multiplier deux nombres premiers, par exemple 127 et 997, et de trouver 126 619. Mais il est plus difficile de factoriser, c'est-à-dire de retrouver 127 et 997 à partir de 126 619.
C'est sur ce principe mathématique que repose le chiffrement asymétrique. Le premier système de chiffrement à clé publique a été proposé en 1978 par R. Rivest, A. Shamir et L. Adleman, trois chercheurs du MIT, qui ont donné leur nom au système baptisé RSA [129]. Cet algorithme est construit à partir d'une fonction à sens unique avec trappe (fonction facile à calculer dans un sens, mais qui est mathématiquement très difficile à inverser sans la clé privée (appelée trappe)). Cette fonction est basée sur le problème de la factorisation des grands nombres entiers. Partant de ce principe, chaque utilisateur va choisir deux grands nombres premiers p et q qu'il garde secrets. Il rend public leur produit N = p * q et un nombre r, premier avec $\phi(N) = (p-1)(q-1)$ qui est l'ordre du groupe multiplicatif des entiers modulo N. L'espace des messages est formé des entiers strictement inférieurs à N.
La fonction de chiffrement est donnée par :

$$X \rightarrowtail X^r (moduloN)$$

le fonction de déchiffrement est donnée par :

$$Y \rightarrowtail Y^s (moduloN)$$

où s est l'inverse de r modulo $\phi(N)$. Cette clef est secrète puisqu'il est nécessaire de connaître $\phi(N)$ pour la calculer.
On a bien :

$$Y^s = (X^r)^s = X^{rs} = X^{k.\phi(N)+1} = X$$

puisque $\phi(N)$ est l'ordre du groupe Z/NZ, par conséquent $X^{\phi(N)} = 1$. Il est facile de voir que la connaissance de $\phi(N)$ est équivalente à celle de la factorisation de N. Si la factorisation de N est connue, $\phi(N)$ se calcule facilement puisque :

$$\phi(N) = (p-1)(q-1)$$

Inversement, si $\phi(N)$ est connu, alors :

$$pq = N$$
$$p + q = N + 1 - \phi(N)$$

Il est donc simple d'en déduire p et q. C'est sur cette méthode RSA que sont fondés de nombreux logiciels de chiffrement et la plupart des logiciels de paiement sécurisé. Pour vérifier l'intégrité du message transmis, ainsi que le caractère exact et complet des données envoyées, on utilise une fonction mathématique qui associe une valeur calculée au message. Lorsque le destinataire reçoit le message, il calcule sa propre valeur et la compare à celle qui lui a été envoyée. L'égalité des deux valeurs assure que les documents n'ont pas subi de modification. La combinaison de procédés d'authentification de l'expéditeur, et de vérification de l'intégrité de son message, permet la création de véritables signatures électroniques, qui s'avèrent en pratique plus difficilement falsifiables que nos procédés de paraphes et signatures manuscrites. La technique informatique permet d'élaborer des outils, générant des clés, et utilisant les systèmes de chiffrement de manière transparente pour l'utilisateur. Le plus célèbre des procédés de chiffrement, et un des plus sûrs d'après les spécialistes, est le logiciel PGP, basé sur le système RSA, inventé par l'américain P. Zimmerman[4]. Cependant le hachage basé sur l'algorithme MD5 a été cassé dans PGP.

1.3.2 Fiabilité des systèmes de chiffrement

Pour que le système soit fiable, il est nécessaire que les clés de chiffrement utilisées soient suffisamment sûres, sachant que les falsifications et atteintes ne sont pas physiquement décelables. Avec les protocoles actuelles, la sûreté d'une clé dépend de sa longueur. Cependant, plus la clé est longue, plus la transaction ou la communication va être lente, en raison du temps nécessaire au logiciel pour effectuer les calculs. Il existe donc un compromis entre sécurité, rapidité et convivialité. Enfin, pour déchiffrer un document sans posséder la clé, il est nécessaire de disposer d'ordinateurs capables d'effectuer un très grand nombre d'opérations par seconde. La fiabilité d'un système dépend donc de la puissance de calcul nécessaire pour casser le protocole. La "dépense" nécessaire pour casser le protocole doit donc être disproportionnée par rapport à la valeur de l'information protégée. Aujourd'hui, une clé de longueur 1024 bits (longueur typiquement utilisée

[4] http ://www.pgpi.com

pour le protocole RSA), nécessiterait plusieurs milliards d'années de calcul pour être cassée. Cependant, ce système dépend de l'état de la technique, qui évolue très rapidement. Un algorithme jugé incassable aujourd'hui ne le sera peut-être plus dans quelques années. Le Challenge RSA [5], lancé en 1997, et qui consistait à casser par force brute (essai de toutes les combinaisons) un message chiffré par un algorithme RC5 à clef de 64 bits, a été remporté le 25 septembre 2002 par le projet Distributed.net [6]. L'opération aura duré cinq années, mais elle prouve que des entités non étatiques peuvent briser les messages de 64 bits. Le projet Distributed.net fédérait des milliers d'internautes, afin d'utiliser le temps libre de leur ordinateur pour tester la totalité des clefs possibles. C'est le 14 juillet 2002 que le PIII-450 d'un internaute de Tokyo a renvoyé la clef. Celle-ci était "0x63DE7DC154F4D039" et produisait le texte clair suivant : "The unknown message is : some things are better left unread". On estime que la longueur de clef d'un protocole symétrique ne doit jamais descendre en dessous de 90 bits, pour que le chiffrement reste sûr. La loi française réglemente, de manière drastique, la création et la diffusion de toute cryptologie dont la clef dépasse 40 bits (tout en ayant autorisé récemment GnuPG, qui utilise des clefs de 128 à 256 bits).

Même si le protocole est incassable, la conception du logiciel peut présenter des failles, qui peuvent être exploitées pour trouver les messages clairs, sans avoir à faire des calculs massifs. Il est en effet fréquent de voir apparaître dans les journaux spécialisés, des failles de système de chiffrement qui sont dûes à la conception du logiciel. Par exemple, une faille de sécurité [2] qui touchait le plug-in Outlook Express de PGP 7.0.3 et 7.0.4 pour Windows a été révélée le 11/07/2002. Ou encore, plus récemment, une faille de sécurité [2] dans PGP 7.1.1 pour Windows, liée à la gestion des noms de fichiers longs, a été découverte et permettrait une attaque à distance. Enfin l'algorithme SSL a été cassé par des chercheurs de l'EPFL [7].

1.3.3 Rôle de la cryptographie dans la société de l'information

Les techniques de cryptographie représentent des enjeux économiques, stratégiques et juridiques considérables. Procédé d'origine militaire, la cryptographie reste considérée comme un enjeu de sécurité intérieure et extérieure, par un certain nombre de gouvernements, malgré le développement des utilisations civiles et commerciales de ces techniques.

Dans un contexte où les échanges d'informations dématérialisées se développent, il est indispensable de pouvoir bénéficier de systèmes sécurisés, afin de protéger les données à caractère personnel ou confidentiel, ou pour assurer la sécurité des transactions financières et commerciales.

[5] http ://www.openpgp.fr.st

[6] http ://www.distributed.net/pressroom/news-20020926.html

[7] http ://www.insatech.net/search.php ?query=&topic=5

En effet, l'utilisation d'un réseau de communication expose les échanges à certains risques, qui nécessitent l'existence de mesures de sécurité adéquates. Il est donc nécessaire d'avoir accès à des outils techniques, permettant une protection efficace de la confidentialité des données, et des communications contre les intrusions arbitraires. Le chiffrement des données est très souvent le seul moyen efficace pour répondre à ces exigences. Les technologies cryptographiques sont ainsi reconnues comme étant des outils essentiels de la sécurité et de la confiance, dans les communications électroniques. Elles vont être amenées à jouer un rôle croissant en matière de protection contre la fraude informatique, de sécurité des données, de protection de la confidentialité des correspondances, de protection du secret professionnel, et du commerce électronique.
Les besoins légitimes des utilisateurs en matière de cryptographie ont été reconnus par la loi du 26 juillet 1996, qui fait référence à la protection des informations et au développement des communications ainsi qu'aux transactions sécurisées. Cependant, la France, invoquant la nécessité de "préserver les intérêts de la défense nationale et de la sécurité intérieure ou extérieure de l'État", a maintenu malgré plusieurs réformes successives, une réglementation contraignante de la cryptographie.

1.3.4 Générateur aléatoire et pseudo-aléatoires

Nous allons ici donner quelques définitions concernant ces générateurs. Ces éléments sont issus du livre de B. Schneier [31]. Ce dernier pose, dans son livre, le problème des générateurs aléatoires et pseudo-aléatoires. En effet, bien souvent, ces derniers sont inclus directement dans le compilateur utilisé, et par conséquent, utiliser un de ces générateurs revient à un simple appel de fonction. En réalité, ces générateurs ne sont, d'un point de vue cryptographique, absolument pas sûrs. Le problème d'un générateur aléatoire est qu'il ne produit pas une suite aléatoire réelle. Il s'avère d'ailleurs impossible d'en produire une, à l'aide d'un ordinateur. Au mieux, il produit un générateur pseudo-aléatoire de suites. De nombreux chercheurs se sont donc posés le problème, et on finit par établir la définition suivante : *la période de la suite générée doit être suffisamment longue de telle manière qu'une suite finie de longueur raisonnable ne soit pas périodique*. En d'autres termes, s'il est nécessaire d'avoir un milliard de bits aléatoires, il ne faut pas choisir un générateur qui a une période de 16000 bits. Ces sous-suites non périodiques relativement courtes, doivent être autant que possible indiscernables d'une suite aléatoire. Par exemple, elles doivent disposer du même nombre de 1 et de 0, et à peu près la moitié des segments (suites de bits ayant la même valeur) doivent être de longueur 1, d'un quart de longueur 2, d'un huitième de longueur 3, etc. On ne doit pas pouvoir les compresser et enfin, les distributions des segments de 0 et des segments de 1 doivent être les mêmes. Ces propriétés peuvent être mesurées empiriquement, et comparées aux prévisions statistiques en utilisant un test du χ^2. De nombreux générateurs ont été

développés dans le monde académique comprenant des tests variés sur leur caractère aléatoire. Tous ces générateurs sont périodiques, mais avec des périodes potentielles de 2^{256} bits (voire plus), ils peuvent donc être utilisés pour les applications les plus exigeantes. Mais le problème des corrélations non désirées et des résultats étranges persiste. Ce sont ces propriétés que le cryptanalyste utilisera pour attaquer le système.

Cependant, il existe des générateurs pseudo-aléatoires que l'on qualifie de cryptographiquement sûrs, pour avoir cette caractéristique, le générateur doit en plus satisfaire la propriété suivante : *il doit être impossible par calcul de prédire quel sera le bit aléatoire suivant, connaissant complètement l'algorithme ou le matériel qui engendre la suite et connaissant tous les bits déjà engendrés.* Les suites pseudo-aléatoires cryptographiquement sûres ne peuvent être comprimées, à moins d'avoir le secret : *le germe utilisé pour initialiser le générateur*. Il est cependant encore possible de casser un tel générateur pseudo-aléatoire.

Enfin, un générateur de suites vraiment aléatoires doit vérifier la propriété suivante : *il ne peut pas être reproduit de manière fiable*. Si on exécute le générateur de suites deux fois, avec exactement les mêmes entrées, on doit obtenir deux suites aléatoires différentes.

Yarrow [88] est un exemple de générateur pseudo-aléatoire considéré comme étant sûr. Il a été prouvé, que ce système est plus robuste que d'autres générateur pseudo-aléatoire. La caractéristique principale du système Yarrow est que ses composantes sont plus ou moins indépendantes. Cela permet à différents systèmes comportant des contraintes différentes, d'utiliser le modèle général de Yarrow. Ce système est basé sur l'utilisation d'une fonction de hachage, ainsi que sur l'utilisation de primitives cryptographiques.

La législation française distingue, d'une part les fonctions d'authentification et d'intégrité des données, soumises à un régime plus libéral, et les fonctions de confidentialité, sur lesquelles l'État entend garder un contrôle étroit. Néanmoins, pour permettre aux utilisateurs de bénéficier de techniques de cryptographie à des fins de confidentialité, la loi a introduit le système dit des «tiers de confiance», ou ce que les anglo-saxons désignent sous le terme de «GAK» pour «Government Access to Keys». L'utilisation de fonctions de confidentialité est libre, à condition que les conventions secrètes soient gérées selon les procédures et par un organisme agréé. C'est en réalité, la principale innovation de la loi du 26 juillet 1996. Or ce système, qui n'existe dans aucun autre pays, soulève de nombreuses questions techniques, juridiques et politiques, soulignées dans plusieurs rapports et documents officiels, et notamment dans un communiqué de la Commission Européenne [43]. Ce communiqué soulève également la question de l'impact de la réglementation du chiffrement sur le marché intérieur européen. Celui-ci introduit aussi le problème de compatibilité entre ces techniques de cryptographie et les principes dégagés par les directives européennes, en matière de protection de la vie privée.

1.4 Définitions caractérisant un service de contrôle d'accès

Un service de contrôle d'accès consiste à fournir à des utilisateurs un contenu d'une qualité moindre (le contenu peut être brouillé complètement ou en partie). Afin d'accéder au contenu dans sa version originale (ou dans une version dont la qualité est identique à l'originale), l'utilisateur doit s'acquitter d'un droit d'accès. La mise en place d'un tel service nécessite l'utilisation d'un système d'embrouillage, et d'un système de transmission de clef (nécessaire au désembrouillage), ces deux systèmes pouvant être corrélés. Nous allons dans un premier temps présenter les principales définitions caractérisant un service de contrôle d'accès, ces définitions étant issues de l'article de S.R. Ely & al. [144], elles sont les suivantes :

- La transparence : Elle correspond à la qualité du médium (ici il s'agit essentiellement du son et de la vidéo), ayant subi les processus d'embrouillage et de désembrouillage. Idéalement, le procédé ne doit pas générer de dégradations perceptibles sur le médium ;
- L'opacité : C'est le degré de dégradation du signal original après le processus d'embrouillage ;
- La sécurité : C'est le point essentiel d'un système d'embrouillage. Elle dépend de ce système et du système de cryptage (les paramètres utilisés pour l'embrouillage sont le plus souvent transmis après avoir été cryptés, afin de pouvoir réaliser le processus de désembrouillage). Au niveau de l'embrouillage, la sécurité repose sur la possibilité de réaliser un désembrouillage, en se basant sur les propriétés de corrélation du signal (par exemple, l'article de M.G. Kuhn [112] présente un moyen de contrer l'embrouillage Nagravision utilisé par canal+). Pour contrer l'éventualité de telles attaques, il est nécessaire de s'assurer qu'il existe un nombre suffisamment grand de permutations du signal embrouillé, et de s'assurer également de l'absence de caractéristiques régulières ou reconnaissables (on peut noter que ce principe est légèrement antagoniste avec celui d'un système, dans lequel on veut conserver une certaine visibilité du contenu original). La sécurité du processus de cryptage est plus facile à assurer. En effet, il existe à ce jour, de nombreux algorithmes dont la sécurité n'est plus à prouver, c'est le cas par exemple des algorithmes RSA ou DES ;
- La complexité et le coût : Le système d'embrouillage doit être simple à utiliser, et sa complexité doit permettre une implémentation temps réel ou quasi temps réel au niveau "hardware" (sur des DSP, Digital Signal Processor, par exemple). Ces contraintes doivent être essentiellement vérifiées au niveau du décodeur ;
- La compatibilité et la robustesse : Les contraintes de compatibilités sont relatives. Il est en effet possible d'implémenter les processus d'embrouillage et de désembrouillage au niveau d'un codeur et d'un décodeur indépendants, qui s'incorporeront dans la chaîne de traitement du signal. Cependant, le signal issu du codeur ou du décodeur ne doit pas être à l'origine d'interférences avec d'autres systèmes de diffusion. Pour être robuste, le signal embrouillé

ne doit pas être plus sensible que le signal original aux bruits du canal de transmission (ce sont essentiellement des contraintes matérielles qui, par exemple, sont importantes dans un système de diffusion terrestre ; il faut en outre conserver le signal de synchronisation entre le flux vidéo et le flux audio). Par exemple, dans un contexte de diffusion terrestre, le procédé utilisé est le "Vestigial Sideband Amplitude Modulation" (VSB-AM), alors que pour une transmission satellite, il s'agit de celui dit de "Frequency Modulation" (FM). Le signal embrouillé est souvent plus sensible aux distorsions apportés par le canal de transmission VSB-AM que le signal normal ;

– La flexibilité : C'est une caractéristique essentielle pour le "business system". Cela correspond à la possibilité de faire évoluer le système d'embrouillage.

Une des applications envisageables pour les systèmes de contrôles d'accès, correspond par exemple à un service de téléchargement d'émission TV, afin d'être enregistrée via un VCR (Video Cassette Recorder), pour une visualisation ultérieure de cette émission. Dans ce contexte, le signal enregistré doit être embrouillé. Ce type d'application s'inscrit dans un ensemble plus large dénommé la télévision à péage. Nous allons maintenant en décrire les principaux composants techniques nécessaires à leur mise en oeuvre.
Dans l'article de L. Brown [100], l'auteur définit quatre éléments essentiels, afin de pouvoir fournir à un client un signal embrouillé, de telle sorte qu'un client n'ayant pas acquitté ses droits ne puisse pas bénéficier du service en question :

– Le système de diffusion : C'est un moyen mis en place pour transmettre le signal jusqu'au client. Il est possible d'utiliser, par exemple, les systèmes suivants : transmission VHF/UHF, MDS (Microwave Distribution Schemes), DBS (Direct Broadcasting by Satellite), ou encore le câble (câble coaxial ou fibre optique) ;
– Le standard de transmission : C'est la technologie utilisée pour combiner la vidéo, le son et éventuellement les métadonnées, afin d'autoriser leur transmission. La solution utilisée est a priori indépendante du système de diffusion ;
– La technologie d'embrouillage : Elle représente la procédure permettant d'embrouiller la vidéo, le son et les métadonnées, composant le signal TV, afin qu'un client ne payant pas les droits d'accès au service ne puisse en bénéficier ;
– La gestion des clefs et leur distribution : C'est le moyen mis en oeuvre pour que la ou les clefs utilisées pour générer le signal embrouillé soient créées et distribuées au client du service.

1.5 Description technique des éléments composant un système de télévision à péage

Nous allons nous intéresser maintenant aux différents éléments techniques, dont la mise en place est nécessaire au bon fonctionnement d'un système de télévision à péage. Nous mettrons plus particulièrement l'accent sur les techniques d'embrouillage actuelles, et utilisées dans les différentes applications existantes.
Dans [144], ainsi que dans [100], les auteurs présentent un aperçu des principales méthodes existantes en brouillage avec la classification suivante :

•La sécurité faible : ces systèmes sont le plus souvent utilisés dans les anciens systèmes du câble.
Les principaux systèmes sont les suivants :

- ”reverse traps and traps” : une trappe qui retire la chaîne payante, est placée avant que le signal vidéo ne soit fourni au client. Cette trappe n'est enlevée que lorsque le client s'est acquitté des droits de visualisation. La chaîne payante peut aussi comporter un signal d'embrouillage qui est éventuellement retiré par la trappe. Cette méthode peut être contrée par l'ajout ou le retrait illégal d'une trappe ;
- suppression ou modification de la synchronisation : les informations de synchronisation sont ici supprimées en brouillant le début de chaque ligne. Il existe plusieurs variantes de cette méthode comme les systèmes basés sur une sinusoïde, les systèmes à impulsions ou encore les systèmes à 3 niveaux. Un signal de référence est transmis via le canal audio, ou sur un canal séparé. Ces systèmes ne sont pas sûrs, et il existe des décodeurs pour les casser [41] ;
- inversion totale de la vidéo : l'intensité du signal vidéo est inversé. Le désembrouilleur n'est fourni qu'aux clients ayant souscrits au service. La construction d'un tel désembrouilleur s'avère relativement facile ;
- inversion aléatoire des lignes constituant la vidéo : des lignes sélectionnées sont inversées. Seul un sous-ensemble de lignes est sélectionné pour effectuer ce traitement. La sélection se fait par des algorithmes, initialisés par une clef qui est transmise soit avec le signal TV soit séparément. Cette méthode est généralement combinée avec un système de suppression de synchronisation et cette information de synchronisation est remplacée par les données de contrôle permettant de décoder le signal. C'est la méthode la plus sûre parmi les systèmes de faible sécurité.

•La sécurité haute : les systèmes mis en place dans cette catégorie sont beaucoup plus complexes mais également beaucoup plus sûrs que les systèmes précédents.
Les principaux systèmes sont les suivants :

- inversion temporelle : une sélection aléatoire de lignes est transmise en ordre inverse. Cette méthode dégrade l'aspect de la vidéo, mais permet toutefois d'avoir une certaine visibilité de son contenu. Ce système peut être utile lorsque le but est d'attirer le téléspectateur mais n'est pas souhaitable dans les systèmes où le signal doit être complètement embrouillé ;
- translation de lignes : la longueur du "padding" au début de chaque ligne est altérée, ce qui a pour effet de déplacer aléatoirement le début de chacune d'elle. Le niveau de translation est déterminé par un générateur pseudo-aléatoire, contrôlé par une clef qui est transmise seulement au client ayant souscrit au service. La translation moyenne sur une image correspond au "line blanking interval", qui permet de ne pas faire varier la taille de l'image . Cette méthode est par exemple utilisé dans le système B-MAC en Australie. Elle possède un haut niveau de sécurité ;
- rotation de lignes avec 1 ou 2 points de coupures (cut and rotate) : Le signal de la partie active de la ligne est coupé, en un ou deux points choisis de façon pseudo-aléatoire. Les deux parties du signal sont ensuite repositionnées sur la ligne ;
- mélange de lignes : Cette méthode s'applique sur un paquet de lignes (32 lignes à 1 trame complète par exemple). Celles-ci sont brassées d'une façon pseudo-aléatoire, c'est à dire qu'elles ne seront pas transmises dans l'ordre du balayage. Cette méthode nécessite un 'buffer' de lignes.

Les séquences de clefs qui contrôlent les algorithmes d'embrouillage ci-dessus, sont créées via un générateur pseudo-aléatoire, ou en utilisant l'algorithme DES. Un des systèmes les plus connus est le système Nagravision [9] utilisé par Canal+. M.G Kuhn [112] en rappelle le principe et en présente les faiblesses. Tout comme d'autres systèmes de brouillage tel qu'EuroCrypt [153, 70] ou VideoCrypt [70], Nagravision envoie un mot de contrôle (ou clef) numériquement crypté, via l'interface radio du décodeur, afin de pouvoir réaliser le désembrouillage. Le mot de contrôle est décrypté dans une "smartcard" et converti en un germe, qui sert à initialiser le générateur pseudo-aléatoire. Ce générateur contrôle ensuite la procédure de désembrouilllage de l'image. Nagravision embrouille le signal vidéo en réalisant une permutation des lignes constituant la vidéo. Nous ne nous attardons pas ici sur l'embrouillage audio qui est plus simple, puisqu'il consiste à inverser le spectre audio puis à le moduler par une sinusoïde.
Il existe 2 méthodes pour 'cracker' ce genre de système :

- Microelectronics a testé des équipements qui peuvent être utilisés pour extraire l'algorithme de décryptage et la clef secrète à partir de la "smartcard". Avec ces données, il est possible

de réaliser des ”smartcard” et des décodeurs pirates [128] ;

– Les propriétés du signal TV peuvent être utilisées pour reconstruire l'image originale ou le germe aléatoire, contrôlant le processus de désembrouillage, qui est ensuite utilisé pour désembrouiller l'image entière, en très haute qualité [154] [48]. Avec cette technique, il n'est pas nécessaire de casser l'algorithme de cryptage, ou la sécurité de la ”smartcard”. Il est possible de l'implémenter sans utiliser un vrai décodeur.

La première attaque peut être utilisée sur des systèmes numériques, tels que DVB, qui crypte un signal MPEG compressé. La seconde attaque n'est pas utilisable pour des systèmes d'accès conditionnel numérique.

Dans [109] M. Pazarci & al. présentent un système d'embrouillage compatible avec les systèmes basés sur la compression MPEG2, comme le système DVB-S/C/T et ATSC (Advanced Television Systems Committee). La dégradation de la vidéo se fait en appliquant une transformation linéaire sur les pixels composant la vidéo, codée par blocs. L'embrouillage se fait préalablement à l'encodage MPEG2. Dans un premier temps, les images sont partitionnées en ”bloc de brouillage” (SB : Scrambling Block) dont la taille est un multiple de la taille des macro-blocs utilisés dans le format MPEG2. Les blocs SB sont synchronisés avec les blocs MPEG2. Le flux RVB d'un bloc SB est codé en utilisant une transformation linéaire des pixels le composant. Cette transformation réalise un mélange aléatoire des éléments structurants :

– luminance ;
– changement de contraste ;
– 'switching' positif ou négatif.

Les paramètres d'embrouillage sont différents pour chaque composante RVB. Ils sont déterminés à partir du GOP précédent. Ces paramètres sont transmis au décodeur de façon sécurisées en les intégrant dans l'image. Ces paramètres passent donc par le biais du cycle encodage puis décodage MPEG-2. Une autre alternative est de transmettre ces paramètres via un flux séparé, dans ce cas, les données sont chiffrées. L'intérêt d'une telle approche est d'être transparente vis à vis du standard MPEG-2 mais aussi d'être indépendante du ”hardware”. Afin de limiter l'augmentation du débit, il est nécessaire d'avoir une synchronisation entre les GOP de l'embrouilleur et ceux de l'encodeur.

En 1996, S. Roche & al. [131] ont proposé un système d'embrouillage permettant un accès multirésolution au contenu embrouillé. Dans ce schéma, ils se basent sur un système de compression à base d'IFS (Iterated Function System) permettant l'accès multirésolution. Le principe des IFS et du codage fractal est le suivant : on représente une image par un ensemble de transformations affines qui, appliquées itérativement, convergent vers un attracteur égal à l'image codée. Nous allons ici présenter brièvement les bases du calcul fractal telles qu'elles sont présentées dans cet article. Les auteurs posent tout d'abord les notations suivantes :

- x,y sont 2 images ;
- x_c représente l'image à encoder ;
- x_0 correspond à l'image initiale du processus itératif ;
- ω correspond à la transformation qui est appliquée à l'image ;
- x_a représente l'attracteur de ω ;
- d est une métrique sur l'espace de l'image.

théorème du collage 1
S'il existe ω *tel que* $d(x_c, \omega(x_c))$ *et*
$d(\omega(x), \omega(y)) \leqslant \sigma.d(x, y)$ *où* $0 < \sigma < 1$
alors $d(x_c, x_a) \leqslant lim_{n\to+\infty} \frac{1-\sigma^n}{1-\sigma}.\epsilon + \sigma^n.d(x_c, x_0)$
avec $x_a = lim_{n\to+\infty}\omega^{on}(x_0)$ *et*
$\omega^{on} = \underbrace{\omega(\omega(...(\omega(x_0))...))}$

Ce théorème ne donne aucun moyen de déterminer ω. Cependant, l'algorithme de A. Jacquin [22] qui introduit le concept d'IFS local, répond à ce problème.
Le problème posé par le théorème du collage est séparé en plusieurs sous-problèmes, où il n'est plus question d'appliquer ce théorème globalement mais localement. Pour ce faire, x_c est partitionnée en blocs sur 2 niveaux. Le premier niveau correspond aux "range blocks" de taille *B*B*, le second niveau correspond aux "domain blocks" de taille *2B*2B* (B est généralement fixé à 4 ou 8 pixels). Pour chaque "range block", l'algorithme cherche le "domain block" qui minimise l'erreur quadratique suivante :

$$\epsilon = \sum_{(i,j)\epsilon R_k} ((\omega_k(D_k))(i,j) - R_k(i,j))^2 \quad (1)$$

où :

- $R_k(i,j)$ désigne la valeur de gris au pixel (i,j) dans le "range block" indexé par k ;
- $(\omega_k(D_k))(i,j)$ représente la valeur de gris du pixel (i,j) dans le "domain block" transformé D_k associé à R_k.

Avant la phase d'appariement (matching), les "domain blocks" sont transformés de la façon suivante :

- sous-échantillonnage par un facteur 2 dans chaque direction ;
- transformations géométriques (8 isométries sont considérées) ;
- "scale & offset" des valeurs de luminance. Ces paramètres sont calculés en utilisant une méthode des moindres carrés, basée sur un critère local décrit précédemment.

Pour chaque "range block" R_k et son "domain block" associé D_k, ces opérations peuvent être transcrites sous forme matricielle :

$$\begin{pmatrix} r_{k1} \\ \vdots \\ r_{kB^2} \end{pmatrix} = [\mathbf{A_k}]_{B^2*4.B^2} . \begin{pmatrix} d_{k1} \\ \vdots \\ d_{k4B^2} \end{pmatrix} + \begin{pmatrix} b_k \end{pmatrix} avec(b_k) = (o_k, \ldots, o_k)^t \quad (2)$$

$$et[\mathbf{A_k}] = \begin{pmatrix} 0 & \cdots & 0 & s_k & s_k & 0 & \cdots & 0 & s_k & s_k & 0 & \cdots & 0 \\ s_k & s_k & 0 & \cdots & \cdots & 0 & s_k & s_k & 0 & \cdots & \cdots & 0 & \\ 0 & \cdots & \cdots & 0 & s_k & s_k & 0 & \cdots & \cdots & 0 & s_k & s_k & \\ 0 & \cdots & \cdots & 0 & s_k & s_k & s_k & s_k & 0 & \cdots & \cdots & 0 & \end{pmatrix} avec -1 \leqslant s_k \leqslant 1. \quad (3)$$

Dans cette représentation, l'espace des "range blocks" est associé aux lignes de la matrice A_k, et l'espace des "domain blocks" est associé aux colonnes de cette matrice. Les 4 s_k non nuls par ligne correspondent aux "scale" des niveaux de gris, et au sous-échantillonnage par moyennage de 4 pixels de D_k. La distribution des s_k autour des 0 représente les 8 isométries. Comme l'ensemble des "range blocks" défini une partition de l'image, l'ensemble des transformations locales ω_k constitue le code de l'IFS local. Dans la phase de décodage, l'attracteur x_a de l'image originale x_c est obtenu à partir du code de l'IFS local, et d'une image initiale x_0 quelconque, par l'algorithme suivant :

- L'image x_0 est partitionnée en un ensemble de blocs carrés.
- Chaque zone R_k de l'image est calculée en prenant le bloc associé D_k dans l'image x_0 et en appliquant les transformations contractantes locales ω_k définies dans l'étape de codage, l'image résultante est appelé x_1.
- L'algorithme itère ce processus pour obtenir x_2 à partir de x_1 jusqu'à obtenir l'image x_a.
- Dans la pratique, moins de 10 itérations sont nécessaires.

Le système de contrôle d'accès est basé ici sur la maîtrise du processus de reconstruction (le processus itératif). Le paramètre de convergence σ (théorème du collage), permet de contrôler la convergence du processus de décodage, et donc la qualité de l'image reconstruite. Ce paramètre n'est pas directement accessible dans l'algorithme de A. Jacquin, c'est pourquoi les auteurs agissent sur le paramètre s_k de la matrice A_k, en modifiant plus ou moins les bits composant s_k (le cas où ce nombre est égal à 8 est présenté). L'image est alors plus ou moins embrouillée. Actuellement, ce processus n'a été développé que dans le cadre de l'image fixe. En outre, il s'inscrit dans un schéma de compression à base de fractale.
La complexité demeure un problème essentiel pour les algorithmes d'embrouillage. W. Kanjanarin

& al. [156] se sont penchés sur ce problème, et par conséquent, sur la rapidité d'exécution des algorithmes de brouillage. Le système développé ici est un système hybride, qui est utilisé conjointement avec des techniques de cryptage classiques. Le schéma proposé a pour but de réduire la complexité de l'algorithme d'embrouillage, tout en maintenant le niveau de sécurité. Ainsi, le temps d'exécution de l'algorithme se trouve réduit. Avant de transmettre le flux vidéo dans le processus de cryptage, les auteurs appliquent une fonction de hachage, ainsi qu'un générateur pseudo-aléatoire à ce flux. Dans ce contexte, ils utilisent une technique de brouillage, qui sépare les données sélectionnées en blocs, puis une méthode de cryptage sur ces blocs.
En 1999, W. Zeng & al. [157] ont proposé un système de brouillage dans le domaine fréquentiel. La mise en oeuvre des systèmes de cryptographie est souvent complexe et coûteuse. Il est nécessaire de prendre certaines précautions, quand à la mise en oeuvre d'un système de brouillage, surtout quand celui-ci s'inscrit dans un système de transmission global, comportant un module de compression. En effet, il faut éviter que les perturbations causées par le système de brouillage apportent une trop grande fluctuation des statistiques de la vidéo, afin de ne pas réduire la potentialité de compression du support embrouillé. La plupart des systèmes d'embrouillage sont des systèmes qui agissent directement dans le domaine spatial. En général, ces méthodes modifient de façon significative les propriétés statistiques du signal vidéo original, entraînant une dégradation de l'efficacité de compression. Ajouté à ce problème de compression, les systèmes agissant dans le domaine spatial permettent l'utilisation d'attaques basées sur les propriétés de corrélation de la vidéo.
Enfin, à l'inverse de l'approche que nous avons développée (utilisant une méthodes d'embrouillage basée sur des principes de tatouage, que nous présenterons au chapitre 3), B. Vassaux & al. [34] ont proposé très récemment un système de tatouage basé sur un système d'embrouillage. La technique de tatouage est basée sur l'étalement de spectre dans le domaine spatial. Cette méthode est ici adaptée, en se basant sur une technique d'embrouillage, afin de protéger des objets contenus dans un flux MPEG4. L'aspect embrouillage évoqué consiste à embrouiller un ensemble de pixels, qui sont alors diffusés dans l'image, celle-ci étant ensuite marquée. Enfin, une opération de désembrouillage est appliquée pour obtenir une image marquée. Ainsi chaque objet contient un pourcentage donné de la marque.

Dans le chapitre suivant, nous présentons un état de l'art sur le tatouage ainsi qu'un bref aperçu des techniques de compression.

1.5. DESCRIPTION TECHNIQUE DES ÉLÉMENTS COMPOSANT UN SYSTÈME DE TÉLÉVISION À PÉAGE

Chapitre 2

État de l’art sur le tatouage vidéo

2.1 Introduction

Dans le chapitre précédent, nous avons présenté les techniques d'embrouillage, ainsi qu'un aperçu succinct des bases de la cryptographie. Ces deux approches permettent une protection a priori d'un medium quelconque. Le problème de ces techniques est que lorsque le medium est désembrouillé ou décrypté, 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. Cette technique permet une protection pérenne du medium qui restera de manière intrinsèque.
Tout comme dans le monde de l'image fixe, ou de l'audio, le tatouage de vidéo consiste à insérer une marque de façon robuste et imperceptible dans le medium [8]. La généralisation grandissante de l'internet et des réseaux haut-débit, permet une transmission et un partage aisé de fichiers vidéos (souvent compressés au format divx). Ainsi nous sommes peu à peu confrontés aux mêmes problèmes que ceux posés par le mp3 et les logiciels de peer-to-peer tels que napster. C'est pourquoi les industriels se posent la problématique de la protection de contenus vidéos dans des contextes de services commerciaux déjà existants ou à venir. Le tatouage apparaît alors comme étant une solution adaptée à ces problèmes.
Une vidéo représente une succession d'images fixes, ce qui peut laisser penser que l'adaptation des techniques de marquage d'images fixes suffit à résoudre le problème de marquage vidéo. Cependant, il n'en est rien, puisque la vidéo engendre de nouvelles contraintes (telles que le temps réel), et des possibilités qui différent de celles de l'image fixe. Le tatouage vidéo représente un sujet d'étude grandissant pour la communauté du tatouage. Cependant les techniques propres à la vidéo n'apparaissent seulement qu'aujourd'hui.
Dans un premier temps, nous présenterons un aperçu des principes de base en compression vidéo, pour ensuite exposer succinctement les principes des formats de compression les plus couramment utilisés. Nous examinerons alors les bases du tatouage, afin de poursuivre sur la présentation des techniques de marquage développées actuellement par les chercheurs, et sur les solutions qui sont actuellement proposées dans le monde industriel.

2.2 Formats de compression

Depuis quelques années, le développement de la compression et des équipements de traitements vidéo ont permis à l'ère digitale de s'épanouir, et de remplacer progressivement l'ère analogique. Le but de développer les formats de compression et les traitements de la vidéo en général, est d'en optimiser le contenu, afin d'en réduire l'espace de stockage, tout en maintenant une excellente qualité. Ce critère de qualité, qui a motivé l'expansion des recherches dans le domaine

de l'évaluation de la qualité des vidéos, sera abordé dans le chapitre 4 afin de mettre en relation le tatouage, et les considérations perceptives que nous avons étudiées. Ceci afin d'améliorer notre algorithme qui sera présenté dans le chapitre suivant. Comme il s'avère nécessaire de bien analyser un système de compression, afin de pouvoir développer un algorithme de tatouage efficace (i.e. imperceptible et robuste à différentes attaques, tout en ayant une capacité suffisante pour pouvoir insérer la quantité d'information nécessaire pour l'application visée), nous nous proposons de décrire brièvement les principes de base de la compression vidéo. Le but d'un système de compression est d'éliminer les redondances spatio-temporelles d'un medium afin d'en diminuer la taille (nous n'évoquerons ici que le cas de la vidéo).
Dans le monde analogique, ces redondances sont exploitées via le codage de la couleur, basé sur la vision et les techniques d'entrelacement. Le monde numérique permet quant à lui d'utiliser de nouvelles méthodes, présentées en section 2.2.2. Le codage de la couleur consiste à déterminer un espace qui se rapproche au mieux des caractéristiques de la vision humaine. De nombreux standards vidéo, tels que le standard PAL, NTSC, ou MPEG introduisent un modèle du système visuel humain pour traiter la couleur. Ces standards prennent en effet en compte la perception non-linéaire de la luminance, l'organisation des canaux de la couleur et les lois de l'acuité visuelle, vis à vis de la chrominance.

2.2.1 Principes élémentaires du codage de la couleur

La théorie des couleurs opposées établit que le système visuel humain décorrèle ses entrées entre des signaux noir-blanc, rouge-vert et bleu-jaune, qui sont traités dans des canaux séparés. De plus, l'acuité visuelle pour la chrominance est inférieure à l'acuité pour la luminance. Afin d'exploiter cet aspect de la vision humaine, les couleurs primaires Rouge, Vert et Bleu (RVB) sont rarement utilisées directement pour le codage ; à la place, on utilise couramment des systèmes de couleurs où les signaux correspondent à des différences, qui se rapprochent du modèle des couleurs opposées proposé par Hering dès 1875 [53]. En vidéo, l'espace résultant de ces considérations est souvent l'espace YUV (ou YC_BC_R), où Y dénote la luminance, U (ou C_B) la différence entre la couleur primaire bleue et la luminance, et V (ou C_R) la différence entre la couleur primaire rouge et la luminance. La faible acuité à la couleur permet une légère réduction du signal de "difference color". En vidéo, il est courant d'employer un sous-échantillonnage pour coder la couleur, les termes couramment utilisés sont les suivants :

- 4 :4 :4 correspond à l'absence de sous-échantillonnage ;
- 4 :2 :2 correspond à un sous échantillonnage de la chrominance par un facteur 2, horizontalement ;
- 4 :2 :0 correspond à un sous-échantillonnage de la chrominance par un facteur 2, à la fois

horizontalement et verticalement. Ce format est celui qui se rapproche le plus de l'acuité visuelle pour les couleurs, lorsque que l'on considère la seule opération de sous-échantillonnage des couleurs ;

– 4 :1 :1 correspond à un sous-échantillonnage horizontal de la chrominance par un facteur 4.

2.2.2 La compression vidéo

Il existe une denrée rare : la bande passante. En effet, la bande passante disponible pour la diffusion de la télévision numérique et ses applications est très limitée. C'est pourquoi, une réelle motivation est née afin de pallier ce besoin : développer de nouvelles technologies de compression pour la diffusion de la télévision numérique et ses applications. Le standard de compression MPEG-2 est à ce jour un des systèmes le plus utilisé au monde. Si un codec permet de diminuer le coût de codage, et donc la bande passante utile, il doit répondre à un certain nombre d'autres exigences :

– accès aléatoire dans la séquence décodée ;
– possibilité de compresser l'information dans plusieurs formats d'image ;
– possibilité d'avoir un débit allant jusqu'à 80 Mbit/s.

La plupart des codecs de compression vidéo sont basés sur des transformées telles que la DCT (Discrete Cosinus Transform), la DWT (Discrete Wavelet Transform), ou encore la transformation fractale. L'architecture d'un codec est généralement composée des étapes suivantes :

– la transformation : afin de faciliter l'exploitation des propriétés statistiques et de la redondance psychovisuelle, les images sont transformées dans un domaine où les bandes de fréquences correspondent (ou s'approchent) au mieux à la sensibilité de l'appareil visuel humain, afin de pouvoir les distinguer plus aisément. Cela est réalisé par exemple par la DCT ou encore par la DWT ;
– la quantification : après la transformation, la précision numérique des coefficients transformés est réduite, afin de diminuer le nombre de bits du flux compressé. Le degré de quantification appliqué aux coefficients est généralement déterminé par la visibilité des distortions résultantes. Les hautes fréquences peuvent être plus fortement quantifiées que les basses fréquences (ceci étant dû à la répartition de l'énergie de l'image majoritairement concentrée dans les basses fréquences). L'étape de quantification est responsable de la perte d'information ;
– le codage : une fois les données quantifiées en un ensemble fini de valeurs, elles peuvent être encodées en exploitant la redondance entre les coefficients. Pour cela, la technique couramment utilisée est le codage entropique (comme le codage de Huffman ou le codage

arithmétique). Le codage entropique se base sur la probabilité d'apparition des symboles. Plus celle-ci sera élevée, plus son expression binaire sera réduite, ce qui permettra de diminuer le volume du flux.

L'une des clefs du codage vidéo est l'exploitation de la redondance qui existe entre les images successives. Ainsi, plutôt que de coder image par image, on code la différence entre les images successives et on réalise des estimations de mouvement.
Il existe de nombreux codecs vidéo, nous pouvons, sans être exhaustif, citer les suivants : MPEG1, MPEG2, MPEG4, MPEG4-AVC (H264), les codeurs de l'avenir (codeurs par ondelettes).
Nous allons maintenant donner quelques détails concernant les codecs MPEG2 et MPEG4, ces derniers nous intéressant plus particulièrement dans le cadre de cette thèse. Dans le cas de la compression MPEG2, l'estimation de mouvement est réalisée grâce à un algorithme BMA (Block Matching Algorithm) que nous détaillerons dans le chapitre 3, consacré à la présentation de notre algorithme de tatouage. L'algorithme dans la norme MPEG2 est fondé sur 2 principes : la compensation de mouvement et la transformée en cosinus discrète. Plus particulièrement, on parle surtout de :

- prédiction temporelle, qui a pour but d'exploiter la redondance temporelle entre différentes images d'une séquence vidéo ;
- transformée en cosinus discrète (DCT), qui effectue une décomposition du signal vidéo dans le domaine fréquentiel. Cette transformation permet d'exploiter la redondance spatiale d'une image ;
- quantification, qui permet de réduire l'information à transmettre ;
- codage à longueur variable (par exemple : codage de Huffman) qui exploite la statistique du signal (un événement rare se codera sur un nombre de bits élevé, et a contrario un événement ayant une probabilité d'apparition forte se codera sur un nombre de bits faible).

En ce qui concerne la compression MPEG4, l'un des principaux objectifs de cette norme est de rendre interactif suivant les souhaits de l'utilisateur, les objets des scènes vidéo. Cette interaction est limitée aux différents scénarios possibles. La conception de cette norme se base sur des objets qui représentent le contenu sonore et vidéo. Ces différents objets, issus de plusieurs scènes différentes, peuvent être recombinés pour former des objets plus complexes. Cette norme organise les objets de façon hiérarchique : le niveau le plus bas est constitué des objets dit "primitifs" comme par exemple les images fixes, les objets vidéo et audio. Ces objets peuvent être utilisés pour représenter des scènes 2D ou 3D. Ces derniers, grâce à des transformations appropriées, peuvent être placés n'importe ou dans une scène vidéo quelle qu'elle soit.

Dans les sections suivantes, nous allons présenter les principes généraux du tatouage, ainsi que les principaux algorithmes de tatouage vidéo.

2.3 Les principes généraux du tatouage

L'arrivée de la vidéo numérique, distribuée à travers les DVD, HDTV et DBS ou encore d'autres media, permet de proposer aux utilisateurs des vidéos de hautes qualités. De plus, la démocratisation de l'internet a démontré le potentiel commercial du marché du multimedia numérique. Cependant, cette venue engendre de nouvelles craintes des propriétaires de contenus, provenant de la disponibilité de l'enregistrement des DVDs, D-VHS et des ordinateurs personnels multimedia. En utilisant ces nouveaux produits, les utilisateurs peuvent réaliser aisément des copies illégales. Le tatouage apparaît donc comme étant une alternative à la protection de la propriété intellectuelle des supports multimedia. Le domaine d'application de cette technologie encore jeune, s'étend du monde 1D (l'audio), jusqu'au monde 3D (et 3D+t), en passant par le 2D et le 2D+t (vidéo). Actuellement, il apparaît que le tatouage seul ne peut pas répondre à une protection suffisamment fiable et complète dans un milieu grand public, où les degrés de liberté (en terme de manipulation des contenus) sont trop élevés. Il semble de plus en plus évident que l'on ne pourra jamais empêcher le piratage grand public, à moins de créer des systèmes complètement propriétaires ; mais dans ce cas, il n'est encore pas assuré que la protection soit totale. La recherche en tatouage a timidement commencé au début des années 90, pour réellement débuter en 1995. Depuis, les concepts fondamentaux du tatouage ont mûrit, et sont maintenant bien connus. Nous allons, dans la suite, en présenter l'essentiel.

2.3.1 Définitions

Que ce soit pour l'image fixe, pour l'audio ou pour la vidéo, le tatouage fait appel aux principes de base suivants :

- robustesse : c'est la capacité que possède un algorithme de tatouage à résister aux attaques extérieures, qu'elles soient bienveillantes ou malveillantes. Pour la vidéo, il peut s'agir d'attaques simples comme le changement de format de compression, le changement de débit ou tout autre traitement classique (il s'agit ici de traitements bienveillants qui ne visent pas forcément à retirer la marque). On peut aussi avoir des attaques plus élaborées, qui ont pour seul but de retirer la marque, comme des attaques statistiques aveugles ou des attaques basées sur la connaissance de l'algorithme utilisé (c.f. [59] et [98]).
- capacité : c'est la quantité d'information que l'on peut cacher au sein du medium. Il paraît

évident que plus on augmente la capacité, plus la marque sera perceptible, et plus la robustesse diminuera (dans le cas où on veut retrouver exactement la marque).

– invisibilité : c'est l'impact que peut avoir la marque sur le medium. Plus le marquage sera fort, plus elle sera visible (et inversement).

Concevoir un algorithme de tatouage revient à trouver le meilleur compromis entre ces trois principes, en fonction de l'application visée. La figure 2 présente une bonne illustration de cette problématique.

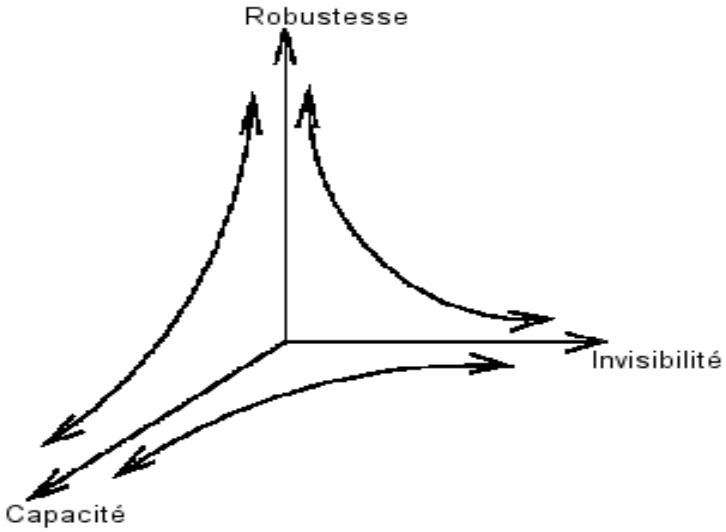


FIG. 2 – Problématique du tatouage

Il existe différents types de tatouage. Par analogie avec la cryptographie, on peut classer un schéma de tatouage selon les catégories suivantes [104] :

– Les schémas privés : Ce système requiert l'image originale. Il en existe 2 types :
  – le système extrait la marque W de la donnée manipulée I', et utilise la donnée originale afin de trouver la localisation de la marque ;
  – le système utilise une copie de la marque pour l'extraction, et répond à la question suivante : est-ce que I' contient la marque ? ($I' \times I \times K \times W \rightarrow 0, 1$), où : W est la marque et K est la clé. Ce type d'algorithme est en général plus robuste puisqu'il nécessite l'accès à un matériel secret. Cependant, de par la nécessité de posséder les informations originales, il est peu pratique d'utilisation.
– Les schémas semi-privés : Dans ce cas, on n'utilise pas l'image originale, on se sert uniquement de la marque et d'une clef ($I' \times K \times W \rightarrow 0, 1$). Le système permet de prouver

l'existence de la marque au sein d'une image.

- Les schémas publics (on parle aussi de tatouage aveugle) : Ils n'utilisent ni l'image originale ni la marque. Seule la clé secrète est nécessaire pour extraire la marque ($I' \times K \rightarrow 0,1$). Il paraît clair que ce système est le plus complexe à mettre en oeuvre, mais aussi le plus pratique à utiliser.
- Les schémas asymétriques : L'extraction de la marque ne nécessite pas la connaissance d'un secret. Ceci implique que tout utilisateur est capable de lire la (ou les) marque(s) du médium sans pouvoir les effacer. Ceci pourrait être réalisé par un marquage sans clé ou par un tatouage avec clé secrète et par une extraction en utilisant la clé publique correspondante (dans un schéma analogue à celui de la cryptographie asymétrique).

Pour une meilleure compréhension de la classification présentée ci-dessus, nous rappelons les définitions suivantes :

- Clé secrète : information connue seulement "d'initiés", permettant d'une part d'insérer la marque dans le contenu, et d'autre part de l'extraire intégralement de ce contenu ;
- Clé publique : information ouverte (à laquelle tout le monde a accès) permettant d'exploiter la marque, mais pas de le modifier (par exemple : information de droit de recopie).

Le principe des clés est fondamental pour asseoir la sécurité des systèmes de tatouage. Un point commun existant avec la cryptographie, selon le "principe de Kerckhoff", est ***qu'un bon système de dissimulation d'information sous-entend la divulgation de la technique de dissimulation elle-même***, la sécurité reposant entièrement sur le caractère secret de la clé.

2.3.2 Les usages

Le tatouage apparaît comme une nécessité pour la protection de contenu multimedia, que ce soit pour l'image, pour la musique ou enfin pour la vidéo. Jusqu'à présent, les systèmes dits propriétaires, ont montré leurs faiblesses. En effet, ce genre de systèmes reposent sur un secret qui, une fois dévoilé, permet de passer outre la protection. C'est par exemple le cas du système CSS de protection des DVD. Deux des principaux intérêts du tatouage, résident dans la protection invisible et dans la conservation d'un medium perceptuellement non dégradé (contrairement aux systèmes cryptant le medium). Donc le support reste parfaitement utilisable. Les applications classiques d'un système de tatouage vidéo sont les suivantes [33] :

- Protection de copyright : 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, quant à elle, l'application la plus courante aujourd'hui. L'objectif est d'incruster une information dans la donnée source, typiquement le copyright du propriétaire, afin de prévenir toute revendication frauduleuse de propriété. Cette signature ne doit être connue que de 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, même si d'autres parties insèrent également une marque ;

– Fingerprinting : cette application est utilisée pour tracer les copies illégales de media (suivi des pirates). Ce type d'application engendre un marquage unique pour chaque copie distribuée (typiquement un numéro de série). Cependant, la distribution de copies composées de différentes marques, peut engendrer des problèmes de collusions (c.f. [65] et [163]). Ainsi, les marques utilisées devront satisfaire un critère de sécurisation de collusions. Enfin, certaines applications de fingerprinting demandent une extraction rapide et de faible complexité, par exemple pour les "WEB crawlers", qui recherchent systématiquement les images piratées. Là encore, un niveau élevé de robustesse est demandé ;
– Protection contre la copie : un souhait des distributeurs de multimedia est l'existence d'un moyen de protection contre la copie, afin d'interdire une circulation de media illégaux. Cependant, cela est difficile à obtenir pour les systèmes ouverts, mais réalisable 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. Un exemple est le système DVD, dans lequel les données contiennent des informations de marquage. Exemple : Il existe deux types de DVD les conformes et les non conformes. Les lecteurs de DVD dits "non conformes" sont basés sur une norme qui ne souscrit pas au traité de protection des DVD. Un DVD sur lequel s'exerce les droits d'auteur, peut être lu et décrypté par un lecteur conforme. Il ne peut pas être lu par un lecteur non conforme, qui n'utilise pas le système CSS (content scrambling system : système de cryptage). Un problème survient lorsque le système CSS est attaqué, et qu'il devient possible de se procurer une copie décryptée de ce DVD. Cette copie peut alors être lue et enregistrée sur du matériel non conforme, mais aussi sur du matériel conforme. L'enregistreur non conforme ne pourra copier qu'une version analogique en raison du système APS (Analog Protection System : perturbe le signal de sortie analogique). L'utilisation du tatouage permet de combler ces failles, en insérant des informations au sein du flux MPEG2. Ces informations contiennent d'une part, des données qui permettent le contrôle de l'enregistrement sur des enregistreurs conformes, et d'autre part, indiquent si le DVD d'origine comprend le système CSS. Ainsi, les lecteurs non conformes sont capables

de lire seulement les DVD illégaux, et les lecteurs conformes, les DVD légaux (recherche de la réduction de la quantité des DVD illégaux) ;

- Contrôle de diffusion : on peut insérer une marque dans une publicité, afin d'en contrôler la diffusion. Cela peut également servir à réaliser une audimétrie ;
- Authentification de données : l'objectif est de détecter toutes modifications éventuelles des données, afin de pouvoir certifier si celles-ci ont été modifiées ou non. On aperçoit ici une problématique de contrôle d'identité de documents. Ce qui peut être obtenu avec un tatouage fragile (c'est à dire qui dispose d'une faible robustesse devant certaines modifications comme la compression). Parmi l'ensemble des applications en tatouage, le marquage pour l'authentification est donc celui qui use du niveau le plus faible de robustesse. Cependant, il faut noter que de nouvelles approches émergent, pour lesquelles les attributs des données, comme en image le moyennage de blocs et les caractéristiques des noeuds, sont marqués et contrôlés par le receveur, qui vérifie si les données contiennent toujours ces attributs. Ces dernières approches demandent donc un niveau de robustesse plus élevé ;
- Indexation : le domaine de l'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 ainsi d'insérer une information (contenant peu de bits) décrivant le contenu de l'image. Cela permet de qualifier sommairement l'image, ou d'insérer un pointeur vers une description plus complète. La société "Digimarc" a présenté un concept d'images "intelligentes". Le tatouage est ici utilisé pour insérer un pointeur vers un lien internet ;
- Sécurité médicale : insertion d'un "identifiant" confidentiel assurant la correspondance entre le patient et la radio, afin d'éviter toutes confusions ;
- Dissimulation de données : stéganographie et transmission de messages secrets.

2.3.3 Les différentes étapes d'un algorithme de tatouage

Nous allons maintenant aborder, de manière plus approfondie, le tatouage. Tout d'abord, il paraît important de séparer les différentes étapes d'un algorithme de tatouage, selon le schéma suivant :

- Dans un premier temps, on commence par formater la marque. Cette étape fait appel à des principes qui viennent aussi bien du monde de la cryptographie (chiffrement de la marque, utilisation de clef...), que du monde du traitement du signal (étalement de spectre, canal de transmission, code correcteur...) ;

– Ensuite vient la phase d'insertion. Chaque domaine possède ses particularités. Dans le monde de l'audio, on possède une bonne connaissance de l'aspect perceptif des bruits, ce qui permet d'optimiser au mieux l'insertion d'une marque robuste. Cette connaissance, étroitement liée aux études de la perception humaine, n'est pas aussi précise dans le monde de l'image fixe ou encore dans celui de la vidéo, ce qui rend plus difficile l'optimisation de la phase d'insertion. L'insertion se fait rarement dans le domaine spatial. On travaille souvent dans un domaine transformé, possédant des propriétés d'invariance ou de dispersion facilitant l'insertion d'une marque robuste et invisible. Il s'agit régulièrement du domaine de Fourier, du domaine des ondelettes, ou encore du domaine DCT, le principe étant de passer dans un domaine fréquentiel ou temps/fréquence.
– Enfin vient la phase de détection. Cette étape est étroitement liée aux précédentes. En effet, pour extraire la marque, le dual de ces phases est souvent utilisé. Dans le cas où la phase de détection n'est pas le dual de la phase d'insertion, on se base sur les propriétés de cette dernière pour réaliser la détection de la marque. Afin d'en améliorer les performances, il est courant d'utiliser des méthodes de préfiltrage qui optimiseront la détection.

Nous allons maintenant décrire quelques éléments constituant ces différente étapes.

Les codes correcteurs

Les codes correcteurs sont présents dans la majorité des systèmes de communication. Leur rôle est d'en augmenter la fiabilité. Lors de la transmission d'information sur un canal de communication, des erreurs peuvent survenir sous forme d'adjonction de bruit au signal initial ou d'inversion de bits. Ces erreurs peuvent être corrigées par des codes correcteurs (si elles ne sont pas trop nombreuses). Ces derniers insèrent une redondance dans l'information transmise, celle-ci pouvant être simple. Par exemple, on peut répéter les bits d'informations un certain nombre de fois, en accord avec le nombre d'erreur que l'on souhaite corriger. Dans ce cas, on augmente le volume d'informations au dépend de la vitesse de transmission. C'est pourquoi les chercheurs ont mis au point des techniques de codes correcteurs d'erreurs plus complexes : il s'agit des codes en blocs linéaires, cycliques, convolutifs ou bien encore plus récemment des turbo codes (mise en série ou en parallèle de plusieurs codes convolutifs). Les codes correcteurs se basent essentiellement sur l'algèbre linéaire. L'utilisation des codes correcteurs en tatouage permet d'augmenter la robustesse de l'algorithme vis-à-vis d'attaques. Dans ce contexte, le support est alors considéré comme étant un canal de transmission, dans lequel l'information qui circule est représentée par la marque. Les attaques sont alors vues comme des erreurs de transmission.

Espaces d'insertion

Un des points clefs dans un algorithme de tatouage est l'espace d'insertion utilisé pour marquer le support multimedia. Il existe de nombreuses transformées, celles qui sont le plus couramment utilisées sont les suivantes :

- La transformée de Fourier discrète (c.f. [118]) : Cette dernière a largement été étudiée en tatouage puisqu'elle offre la possibilité de contrôler les fréquences du signal. Cela permet de choisir de façon adéquate les parties de l'image qui devront être marquées, afin d'obtenir un bon compromis entre visibilité et robustesse. Cette transformation est également utilisée pour fusionner la marque avec le medium, dans la phase de modulation, mais aussi utilisée pour diviser les images en bandes perceptuelles ;
- La transformée en cosinus discrète (c.f. [32]) : Les règles de marquage opérant dans le domaine DCT sont souvent plus robustes à la compression JPEG et MPEG. Le créateur de marques peut ainsi prévenir ces attaques plus facilement. De plus, les études menées concernant les distorsions visuelles sur la source à coder, contribuent à une meilleure prédiction de l'impact visuel d'une marque sur le medium. Enfin, insérer une marque dans le domaine compressé permet de réduire les temps de calcul.
- La transformée de Mellin Fourier (c.f. [93] : La plupart des algorithmes de marquage rencontrent des problèmes lors de l'extraction de la marque, après que le medium ait subi une transformation géométrique affine. Or, la transformée de Mellin Fourier est basée sur la propriété de translation de la transformée de Fourier, qui mentionne que seule la phase est altérée par une translation. En restreignant l'espace de marquage à l'amplitude de la transformée de Fourier, le support de la marque devient insensible à toute translation de l'image. Afin de rendre également le medium insensible aux zooms et aux rotations, il suffit de réaliser un changement de variable exponentiel (LPM : Log polar mapping). En effet, la rotation d'un élément dans le système de coordonnées cartésiennes devient une translation dans le système de coordonnées logarithmiques. De la même manière, le zoom est transformé en translation dans le système de coordonnées polaires. Donc, en utilisant les modifications adéquates, les rotations ainsi que les zooms peuvent être réduits à de simples translations. Donc, la propriété d'invariance par translation peut être utilisée pour construire un espace insensible à toutes opérations de rotations et de zooms, celles-ci sont alors supportées par l'image marquée.
- Le domaine ondelettes (c.f. [46] : On retrouve les mêmes atouts dans la mise en place d'une marque dans le domaine des ondelettes que dans le domaine DCT pour JPEG, avec cependant un avantage supplémentaire provenant de la multirésolution. Cet aspect permet de réaliser en effet, une bonne distribution du message dans le médium en terme de robustesse

et de visibilité.

– La division d'images en bandes perceptuelles (c.f. [84]) : Ce principe consiste à réaliser des processus itératifs, qui permettent de prendre en compte le modèle visuel humain afin de maintenir la marque en dessous d'un seuil de visibilité. Cette hypothèse est encore à l'étude pour la vidéo mais déjà utilisée en audio (une énergie localisée dans une bande de fréquence peut masquer une bande voisine, d'énergie plus faible, c'est le principe de masquage). L'idée de cette méthode repose sur l'hypothèse que le système visuel humain divise le stimulus visuel en plusieurs composantes. Chaque composante étant associée à 3 paramètres :
  – la localisation dans le champ visuel ;
  – la fréquence spatiale (calculée à partir de l'amplitude de la transformée de Fourier) ;
  – l'orientation (calculée à partir de la phase de la transformée de Fourier).

  Ces composantes sont transmises des yeux vers le cortex, à travers différents canaux. L'image est donc divisée en plusieurs canaux, par une technique d'analyse d'image (filtres de Gabor). Puis l'énergie locale est calculée pour chaque canal. Finalement, une fonction de contraste dépendant de la fréquence, de l'orientation, et de la localisation du canal, est calculée. Ce qui permet l'obtention d'un masque psychovisuel tel que : tout signal d'énergie inférieure à ce masque sera invisible.

Préfiltrage de la marque

Un grand nombre d'algorithmes de marquage fonctionnent sous des hypothèses statistiques concernant le signal source. Afin de rendre ces hypothèses aussi viables que possibles durant la phase d'extraction, l'image marquée est alors filtrée (c.f. [147]) avant de débuter l'extraction du message.

Le maximum de corrélation de phase pour la réorientation et le redimensionnement est un bon exemple de ce procédé.

Afin de surmonter aisément les attaques géométriques, telles que les rotations ou les zooms, une estimation a priori de l'attaque (basée sur des transformations géométriques) doit être réalisée, afin de pouvoir appliquer la transformation inverse. L'avantage de cette technique apparaît dans la non restriction de l'espace dans lequel la marque doit être insérée (un espace invariant peut être très petit). Pour ce faire, on peut par exemple, connaissant une partie de l'image originale, réaliser un graphe 3D correspondant au maximum de corrélation de phases (les axes de la grille correspondent au facteur d'échelle horizontal, au facteur d'échelle vertical, et à l'angle de rotation). Ce graphe permet de déterminer le maximum de corrélation de phases. Les coordonnées de ce maximum donnent alors les valeurs du facteur d'échelle horizontal, vertical et finalement de l'angle de

rotation.

Les attaques

Les attaques sont, le plus souvent, des traitements classiques qu'une personne effectue sur le support qu'elle utilise. Les attaques peuvent être des traitements visant soit à brouiller soit à enlever la marque de protection dans la vidéo. Vassaux et al. [34] présentent un survol des différentes attaques en image fixe et en vidéo. La vidéo étant une succession d'images fixes, on peut appliquer la plupart des attaques de l'image fixe à la vidéo. Cependant, certaines attaques couramment utilisées en image fixe ne sont pas applicables à la vidéo, c'est le cas par exemple de l'attaque stirmark [61], qui n'a pas d'intérêt en vidéo, si elle est appliquée sans être adaptée. On peut distinguer 2 grandes familles d'attaques, les bienveillantes et les malveillantes. Nous ne détaillons pas ici les attaques de type géométrique. Ces dernières sont en effet moins pertinentes dans le cadre de la vidéo, et ne correspondent pas aux objectifs fixés dans cette thèse.

- Attaques bienveillantes : Il s'agit de traitement qui n'ont pas initialement pour objectif d'empêcher la détection de la marque. Il peut s'agir des dégradations dues à une compression (MPEG1, MPEG2, MPEG4, MPEG4-AVC, etc.), à un changement de type de compression, à des filtrages (réduction de bruit), à un changement de résolution, au type de codage (progressif ou entrelacé). Un autre traitement couramment utilisé en vidéo est la conversion analogique/numérique, et inversement. Enfin, certaines distorsions géométriques peuvent être utilisées : flip vertical (couramment utilisé pour rendre les publicités non reconnaissables dans une séquence), cropping, perte d'une ligne ou d'une colonne, etc ;
- Attaques malveillantes : De nombreuses attaques initialement développées pour l'image fixe peuvent être facilement adaptées à la vidéo. On peut par exemple appliquer l'attaque Oracle ou des attaques statistiques (collusion ...). Deguillaume et al. [59] proposent une technique de débruitage basée sur un filtrage adaptatif de Wiener. Ils proposent également une attaque ciblée sur la vidéo, qui se base sur ses propriétés temporelles.

Nous allons maintenant donner plus de détails sur différentes attaques.

Diminution du signal

Un pirate dispose d'un large arsenal pour réaliser son attaque, et une brève analyse du processus révèle souvent qu'une opération simple peut suffire à détruire une marque.

- Bruit et surmarquage : Un moyen intéressant pour ajouter du bruit est d'incruster tout simplement une nouvelle marque. Cette opération cause de très légères dégradations. Une

marque publique est a priori plus facile à détruire qu'une marque privée, par surmarquage (i.e. insertion d'une deuxième marque). En effet, en stéganographie une marque privée peut être cachée parmi un ensemble très large de localisations. Ceci permet l'insertion de 2 marques, comportant des clés secrètes différentes, dans des endroits distincts. Une marque publique, peut, quant à elle, être localisée dans un unique emplacement. Il est donc dans ce cas plus probable d'engendrer un conflit d'espace. En image, des programmes de marquage, tel que le "Digimarc Corporation's Picture Marx" prennent en compte ce problème et refusent toute insertion de nouvelle marque ;

– Compression : La technique la plus utilisée en compression vidéo est divx (codec basé sur le standard MPEG4). Généralement, les marques sont insérées dans les hautes fréquences. Cependant, ce sont ces dernières qui disparaissent lors de la compression. De ce fait, les marques doivent être placées sur les composantes perceptuellement significatives, en dépit des risques de distorsion, qui peuvent engendrer des artefacts visibles. Dans notre approche (que nous présenterons aux chapitres 3 et 4), nous avons testé différentes attaques, produites par les codecs divx 5 et MPEG4-AVC ;
– Filtrage : Il existe de nombreux filtres, qui diminuent la précision du signal original. Sans être exhaustif, nous pouvons citer les filtrages de type "blur", moyenneur, sobel, etc. En parallèle des systèmes de codage, nous avons également testé le filtrage "blur" afin d'évaluer la robustesse de notre système.

Attaques spécifiques

En connaissant les détails d'un algorithme de marquage, un pirate peut créer une attaque spécifiquement conçue pour extraire une marque bien précise. En effet, lorsqu'une marque a été insérée dans le domaine de Fourier, le pirate se placera également dans ce domaine pour faciliter sa destruction. Il existe différents types d'attaques spécifiques, que l'on classe généralement dans la catégorie des attaques malveillantes.

– Moyennage (c.f. [66]) : 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 détectable. De même, dans certains cas, le pirate peut extraire la marque : cela peut être réalisé aisément dans les applications vidéos, 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) ;

- ”Protocol attack” (c.f. [61]) : Il s'agit de l'insertion d'une marque supplémentaire par extraction (le pirate soustrait sa marque au lieu de l'ajouter). Ainsi, la fonction de corrélation prouve la présence des 2 marques, on ne peut donc pas déterminer quel parti est le créateur de l'image. En soustrayant sa marque et non en l'ajoutant, le pirate évite le problème d'ordre d'insertion. Cette attaque est basée sur l'inversion d'un processus de tatouage. Un bon moyen de pallier à ce problème, consiste à mettre en oeuvre une méthode d'insertion, utilisant une fonction à sens unique (c.f. [75]). Dans ces processus non inversibles, il est pratiquement impossible de soustraire une marque ;
- ”Oracle attack” (c.f. [95]) : Cette méthode utilise les informations retournées par le détecteur pour extraire la marque, en appliquant de légères variations au contenu jusqu'à ce que le décodeur ne puisse plus retrouver la marque. La solution peut être, éventuellement, de crypter la marque en utilisant une clé, elle même ajoutée à la marque. Dans ce cas, la marque n'est plus déterministe ;
- ”Custom-tailored oracle attack” : L'”oracle attack” ne fonctionne que sur des marques publiques, contrairement au ”Custom-tailored oracle attack”. Le pirate insère sa propre marque N fois dans l'image, en utilisant la même méthode que le propriétaire (algorithme de marquage et de détection connus). L'”oracle attack” est utilisée lors de la mise en oeuvre et permet un affaiblissement progressif de la marque privée originale, qui sera alors finalement détruite. Cette approche repose sur l'hypothèse qu'une marque étant affaiblie lorsqu'une image est modifiée, il en est de même lorsqu'elle subit l'insertion d'autres marques ;
- ”Echo hiding attack” (c.f. [61]) : L'echo hiding essaie de cacher une information dans un signal discret $f(t)$, en introduisant un écho $f(t-\delta t)$ dans le stego-signal $c(t)$: $C(t) = f(t) + \alpha f(t-\delta t)$. L'objet de cette attaque consiste à détecter l'écho, et à l'extraire en inversant tout simplement la formule de convolution. La difficulté est de détecter l'écho sans la connaissance de l'objet original ou des paramètres de l'écho. L'attaque est donc basée sur l'analyse de coefficients cepstraux (proposés aussi pour le marquage audio).
- ”Twin peaks attack” (c.f. [106]) : Cette attaque utilise les caractéristiques de l'image pour détruire la marque (i.e. une image ”cartoonisée” dispose d'un faible nombre distinct de couleurs). Elle permet de tirer avantage de la simplicité des images, pour en extraire des marques, par simple étalement de spectre. Par exemple, on réalise un histogramme de l'image, qui dispose d'un pic d'intensité P. On suppose que cette image est marquée avec un tatouage distribué uniformément, avec une amplitude bi-modale. Dans ce cas, le processus de marquage mappe 50% des valeurs de P à P+d, et les 50 autres de P à P-d. Le pic de l'histogramme original est alors remplacé par 2 pics P-d et P+d (d'où le nom de l'attaque).

En regardant cet histogramme, il est possible de déterminer l'image marquée (en utilisant une prédiction de la marque).

2.4 Le tatouage Vidéo

Le tatouage numérique joue un rôle primordial dans la création d'un système de protection de copie numérique à numérique. Avec l'introduction de l'enregistrement des produits DVD, des enregistreurs D-VHS..., les utilisateurs réalisent aisément des copies parfaites. Ce qui explique qu'un système de protection numérique représente un intérêt non négligeable, qui aidera au maintien du contrôle des copies.

2.4.1 Introduction

Dans un flux vidéo, il est possible de ne tatouer que les images de type intra. De nombreux schémas développés pour les images fixes peuvent être alors appliqués aux séquences vidéos.
Ces dernières présentent cependant d'autres propriétés, qui peuvent être exploitées pour l'insertion de la signature :

- la taille brute d'une séquence vidéo est beaucoup plus importante que la taille d'une image fixe. L'espace d'insertion de la signature en est considérablement augmenté ;
- la dimension temporelle du signal traité peut être utilisée pour l'insertion de la signature. Celle-ci peut, par exemple, être insérée dans le mouvement des différents objets de la séquence.

Les séquences vidéo présentent également des contraintes différentes de celles des images fixes :

- la complexité du schéma de tatouage doit être faible. L'insertion et la détection de la signature doivent pouvoir s'effectuer à la volée, dans la plupart des applications. La contrainte de temps réel s'applique essentiellement à la phase de détection ;
- le mouvement des objets exacerbe souvent la visibilité de la marque : ainsi, une marque "fixe" ajoutée sur un objet en mouvement, sera d'avantage perceptible que si l'objet est statique ;
- le flux vidéo est souvent compressé de manière à réduire la taille brute des séquences. Rappelons qu'un des standards de compression le plus couramment utilisé dans le monde industriel est MPEG2. L'insertion de la signature peut alors directement s'effectuer lors de la compression. L'insertion sur le format décompressé ne doit pas entraîner, après compression, une augmentation significative de la taille des données ;
- la présence de la signature dans la séquence vidéo peut permettre d'autres attaques que celles liées aux images fixes. Si la signature est redondante dans la séquence, elle peut être

estimée en calculant la moyenne des différentes images de la séquence. La signature doit pouvoir être détectée après une perte de synchronisation, produite par la sélection d'une séquence précise ou la perte d'images de la séquence.

La section 2.4.2 présente un aperçu des différents schémas de tatouage appliqués à la vidéo.

2.4.2 Les différents algorithmes de tatouage vidéo

Nous allons, dans cette section, exposer différentes approches utilisées en tatouage vidéo. Comme nous allons le voir dans la suite, les schémas de tatouage sont indirectement liés aux schémas de codage. D'ailleurs, de nombreuses techniques de tatouage sont directement issues des techniques de codage. On retrouve en tatouage les grands principes de la compression (domaine transformé, mouvement, aspects psychovisuels...).

La marque peut être insérée dans différents espaces. Il peut s'agir d'un domaine spatial, d'un domaine transformé, ou encore du domaine temporel. Le domaine spatial présente l'avantage d'être peu coûteux en temps de calcul, puisqu'il n'est pas nécessaire de réaliser des transformations. Cependant, ce domaine ne permet pas de gérer aisément l'invisibilité. En outre, les domaines transformés sont utilisés en tatouage lors de l'insertion du système de tatouage dans un processus de codage ou lors de l'insertion de la marque dans un flux compressé. Le domaine DCT est le plus couramment utilisé. D'autres transformées (telle que la FFT 3D, c.f. [58]), sont plus marginales. Plus récemment, la communauté de traitement d'images a accentué ses recherches sur la transformée en ondelettes (la DWT). Des techniques de compression basées sur cette transformée ont commencé à émerger, et c'est tout naturellement que le tatouage a vu l'émergence de techniques basées sur cette transformée.

La structure d'une vidéo s'avère sensible à un certain nombres d'attaques : le moyennage, la suppression et la permutation d'images. Effectivement, à des fréquences de 25Hz à 30Hz (fréquences utilisées en télévision), ces modifications resteraient probablement invisibles pour l'utilisateur. Un bon marquage doit donc être robuste à ce type d'attaque. Il est donc généralement proposé de distribuer la marque sur plusieurs images consécutives. Cependant, il existe des applications pour lesquelles il est souhaitable de retrouver l'information complète de marquage sur une petite partie de la séquence. Dans ce cas, cette solution n'est pas recevable. Alors qu'il existe un grand nombre de publications sur le marquage des images, il en existe pourtant peu en vidéo. Dans la partie suivante, nous allons résumer quelques méthodes de marquage vidéo.

Techniques de tatouage vidéo provenant de schémas d'images fixes

De nombreux algorithmes de tatouage vidéo sont directement adaptés de techniques appliquées à l'image fixe. Pour un apercu des techniques d'image fixe, le lecteur intéressé pourra se référer

aux articles de G.C. Langelaar *et al.* [71], de N. Nikolaidis *et al.* [116] et de J.J.K. O'Ruanaidh *et al.* [94] qui proposent de bons états de l'art. Nous citerons plus précisément une des principales techniques utilisée en tatouage qui est l'approche basée sur l'étalement de spectre qui a été introduite par I.J. Cox *et al.* [81]. Dans cet article, les auteurs décrivent une méthode pouvant s'appliquer de manière générale à l'audio, à l'image fixe ou encore à la vidéo. Ils supposent que, pour être robuste, un algorithme de tatouage doit insérer la marque dans les composantes significatives du signal. Cependant, un tel procédé dégrade visiblement le signal. Pour éviter trop de distortions, ils proposent d'utiliser une technique d'étalement de spectre qui consiste à insérer un signal à bande étroite dans un signal à large bande.

• C.T. Hsu & J.L. Wu [39] présentent une méthode de marquage de vidéos compressées, qui constitue une extension de leur méthode, mise en oeuvre pour les images. Celle-ci modifie les fréquences moyennes des coefficients DCT en fonction des blocs voisins (spatialement pour les images intras et temporellement pour les images P et B). On force la valeur des coefficients à être inférieure ou supérieure aux coefficients des blocs voisins, suivant la valeur de l'échantillon de la marque que l'on souhaite insérer. Le signal de marquage est basé sur un motif visuel, (tel qu'un logo composé de pixels binaires). L'extraction de la marque nécessite l'utilisation de la vidéo non marquée.

• J. Dittman & al. [83] présentent deux méthodes de tatouage. La première est adaptée de celle de Fridrich [85], qui consiste à insérer un motif, créé par un générateur pseudo-aléatoire, sur une image. L'énergie du motif est essentiellement concentrée dans les basses fréquences, ce qui assure une plus grande robustesse face à la compression, mais rend la marque plus visible. Le principal désavantage de ce système est qu'il nécessite l'image originale. L'information contenue dans la marque, est très pauvre et le système permet seulement de savoir si l'image est marquée ou non. Ces deux problèmes ont été résolus par J. Dittman et al., pour être ensuite adapter à la vidéo.

Génération de la marque :

Le marquage se fait sur des blocs $8 * 8$. Pour chaque bloc, une séquence $8 * 8$ pseudo-aléatoire est générée à partir d'une clef propre à chaque utilisateur. Un automate cellulaire est ensuite appliqué sur cette séquence, afin d'en retirer les hautes fréquences. Cet automate obéit aux règles suivantes : chaque bit est testé en fonction du nombre de $'1'$ dans son voisinage. Si ce nombre est supérieur à 5 alors on lui affecte la valeur $'1'$ et si le nombre est inférieur à 3, on lui affecte la valeur $'0'$. Pour diminuer le taux d'erreur, un code correcteur est utilisé (il s'agit du code $BCH(31, 6, 15)$) ainsi qu'un code de redondance. Le motif M est ainsi obtenu.

Insertion de la marque :

Les blocs $8 * 8$ marqués sont sélectionnés par la technique utilisée dans l'algorithme de E. Koch & J. Zhao [54]. L'insertion se fait dans les valeurs de luminance des images. Pour cacher un $'1'$, on ajoute une valeur k (calculée selon les valeurs des paramètres "$smooth$, $edge$ et $level$") à la valeur de la luminance du pixel, dont la position dans la séquence m correspond également à un $'1'$. On soustrait la valeur k pour cacher un $'0'$. La figure 3 explicite la phase d'insertion.

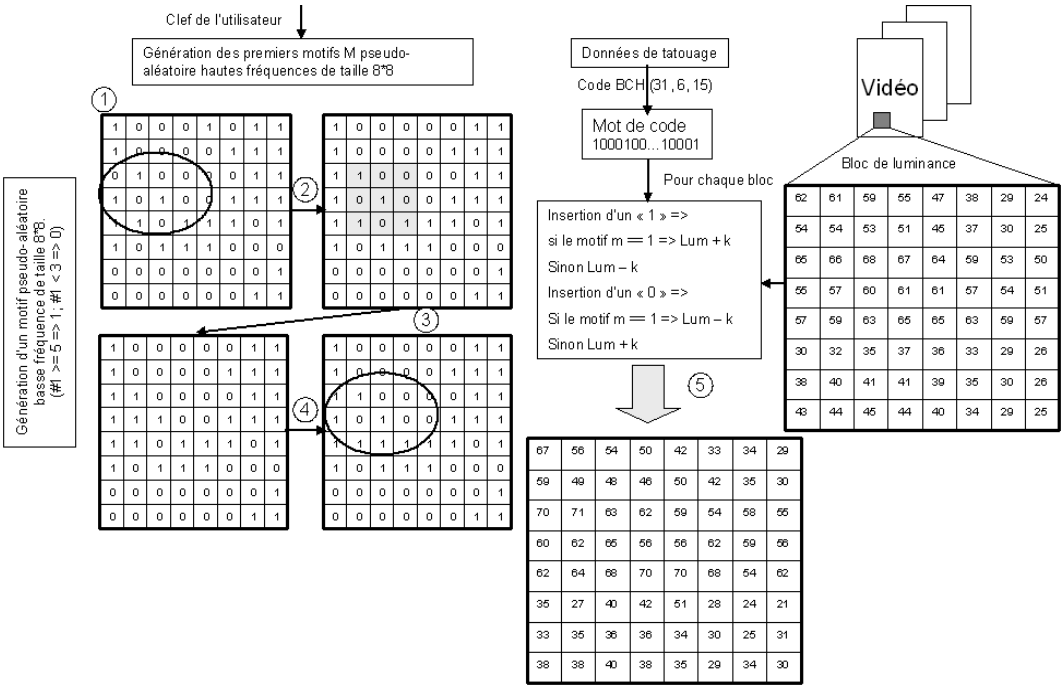


FIG. 3 – Procédure d'insertion de Dittman

Extraction de la marque :

Le motif M généré pour insérer la marque est recréé. La corrélation entre le bloc de luminance et la séquence m est ensuite calculée. Les valeurs moyennes des luminances, $av1$ et $av0$, correspondant à la valeur $'1'$ et la valeur $'0'$ dans la séquence m, sont préalablement déterminées. Si le bloc de luminance et le motif M ne sont pas corrélés, la différence entre les deux valeurs $av1$ et $av0$ est nulle. Mais par le principe d'insertion de la marque, une de ces deux valeurs doit être supérieure à l'autre (de l'ordre de $2 * k$). Ainsi, la valeur du bit masqué est estimée en comparant les valeurs $av1$ et $av0$. Si $av1 > av0$, alors la valeur vaut $'1'$, sinon elle vaut $'0'$. Cette analyse statistique permet de retrouver la marque, sans avoir recours aux images d'origine. Finalement, après avoir retrouvé tous les bits du signal, la marque est décodée avec le code correcteur d'erreurs et le code de redondance. La figure 4 explicite la phase d'extraction.

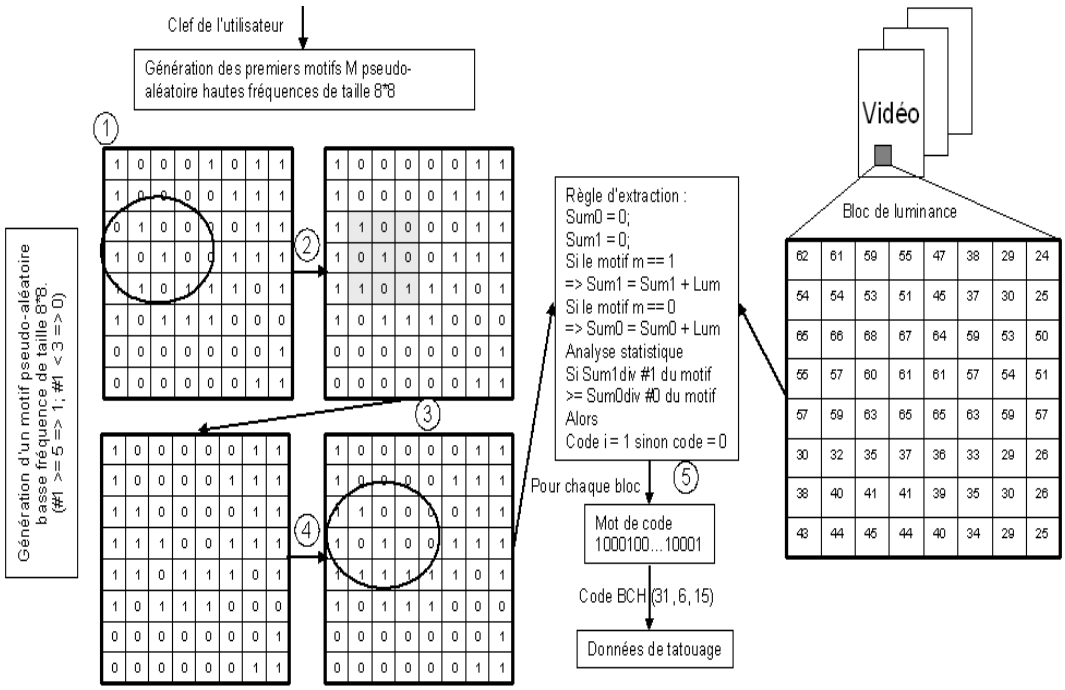


FIG. 4 – Procédure d'extraction de Dittman

Dans cet article, les auteurs présentent également un algorithme d'insertion dans le domaine DCT, basé sur l'algorithme de E. Koch & J. Zhao [54]. Le marquage se fait sur la luminance. L'information de luminance est d'abord transformée dans le domaine DCT (sur des blocs $8 * 8$), 3 coefficients DCT sont alors sélectionnés aléatoirement, puis, selon le bit d'information à insérer, une relation prédéfinie est imposée à ces 3 coefficients. L'extraction se déroule de la même façon, 3 coefficients sont sélectionnés, puis selon leur configuration, un $'1'$ ou un $'0'$ est extrait. L'avantage de cette méthode est la facilité d'intégration dans un schéma de compression de type MPEGx. Mais les blocs sont modifiés indépendamment de leur contenu, ce qui peut conduire à l'apparition d'artefact gênant. En outre, l'algorithme n'est pas robuste aux changements d'échelle et aux rotations.

Génération de la marque :

Le marquage se fait sur des blocs $8 * 8$. Pour chaque bloc, une séquence $8 * 8$ pseudo-aléatoire est générée à partir d'une clef propre à chaque utilisateur. Pour diminuer le taux d'erreur, un code correcteur est utilisé (il s'agit du code $BCH(31, 6, 15)$), ainsi qu'un code de redondance.

Insertion de la marque :

Les blocs sélectionnés à partir de la clef secrète sont d'abord transformés dans le domaine DCT. Deux paramètres sont alors déterminés, un paramètre de lissage ($smooth$), et un paramètre caractérisant les contours ($edge$). Le premier correspond au nombre de coefficients DCT non nuls

après la quantification par la matrice Qm (cf. fig 5). Afin de réduire les artefacts au niveau des contours, un deuxième paramètre est déterminé, il correspond à la somme (en valeur absolue) des 8 premiers coefficients AC suivant le coefficient DC, dans le parcours en zigzag. Ces deux paramètres permettent d'adapter la force de la marque au contenu du bloc.

Bas	16	11	10	16	24	40	51	61	
	12	12	14	19	26	58	60	55	
	14	13	16	24	40	57	69	56	
	14	17	22	29	51	87	80	62	
	18	22	37	56	68	109	103	77	
	24	35	55	64	81	104	113	92	
	49	64	78	87	103	121	120	101	
	72	92	95	98	112	100	103	99	Haut

FIG. 5 – Matrice Qm

$$level = smoothscale * smooth + edgescale * edge + offset \tag{4}$$

Le paramètre *offset* représente la force de marquage de base du tatouage. Les valeurs des paramètres *smoothscale*, *edgescale* et *offset* sont déterminés expérimentalement :
smoothscale $= -10$, *edgescale* $= 0.27$ et *offset* $= 50$
La valeur du paramètre *level* est restreinte à l'intervalle $[0, 5]$. Plus le paramètre de lissage est grand, plus il y a de composantes fréquentielles et, par conséquent, il sera possible de cacher plus d'informations. Un paramètre de quantification supplémentaire est inséré pour déterminer la force finale de marquage. L'algorithme de E. Koch & J. Zhao est appliqué en utilisant une matrice de quantification Qm/Qf. Le coefficient Qf est déterminé à partir de la table présentée sur la figure 6.

Q_f	1	1	2	3	4	4
Niveau/10	0	1	2	3	4	>4

FIG. 6 – Table déterminant la valeur de Qf

Trois coefficients $Y1, Y2$ et $Y3$ sont ensuite sélectionnés parmi les moyennes fréquences, afin d'insérer les bits de la marque. Pour insérer un bit, les 3 valeurs sont modifiées selon le schéma présenté sur la figure 7.

Bit	1	1	1	1	0	0	0	0
Y1	H	H	M	M	L	L	M	M
Y2	H	M	H	M	L	M	L	M
Y3	L	L	L	L	H	H	H	H

FIG. 7 – Modification sur les 3 coefficients Y1, Y2 et Y3

Si les changements sont trop importants et que les effets visuels sont trop perceptibles, les coefficients $Y1, Y2$ et $Y3$ sont modifiés en un motif invalide (HLM, LHM, MMM). Les blocs sont ensuite quantifiés de nouveau, et la DCT inverse est appliquée. La figure 8 explicite la procédure d'insertion. *Extraction de la marque :*
L'extraction s'effectue de la même manière que dans l'algorithme de E. Koch & J. Zhao. La séquence vidéo est décodée, et le processus inverse à l'insertion est appliqué : la séquence de position des blocs tatoués est générée et la marque est ensuite extraite. Les 2 premières étapes sont les duales des étapes correspondantes au processus d'insertion. La seconde étape n'est pas nécessaire mais les informations relatives à la force de marquage dans chaque bloc peuvent être utiles. Dans la 3ème étape, à partir des trois coefficients modifiés lors de l'insertion, la séquence qui leur est associée est extraite, et, par conséquent, la valeur du bit caché est retrouvée. La figure 9 explicite la procédure d'extraction.

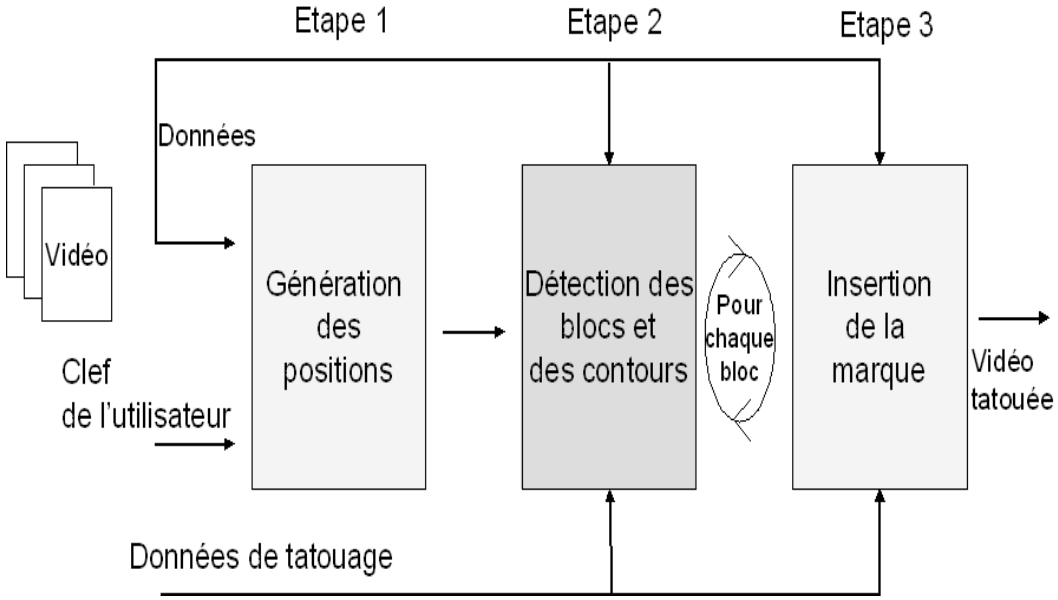


FIG. 8 – Procédure d'insertion de Dittman

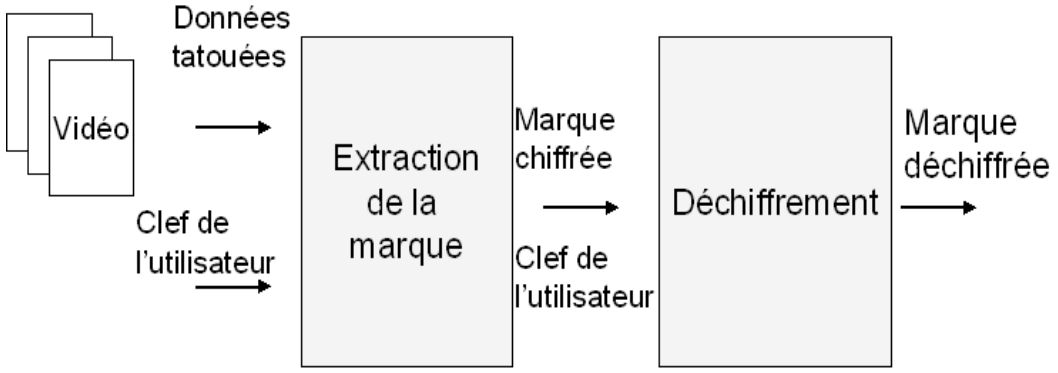


FIG. 9 – Procédure d'extraction de Dittman

• J. J. Chae & al. [87] proposent un algorithme d'insertion dans le domaine DCT. Le principe est d'insérer des images (ou une vidéo) dans une vidéo (insertion d'un volume important d'information). Les auteurs appliquent une DCT sur les blocs 8*8 de l'image à marquer et sur l'image originale. Les blocs sont analysés, pour obtenir un facteur multiplicatif, qui permettra d'adapter la force du marquage. La vidéo originale n'est pas nécessaire pour la détection. L'insertion se fait ensuite selon le schéma présenté sur la figure 10.

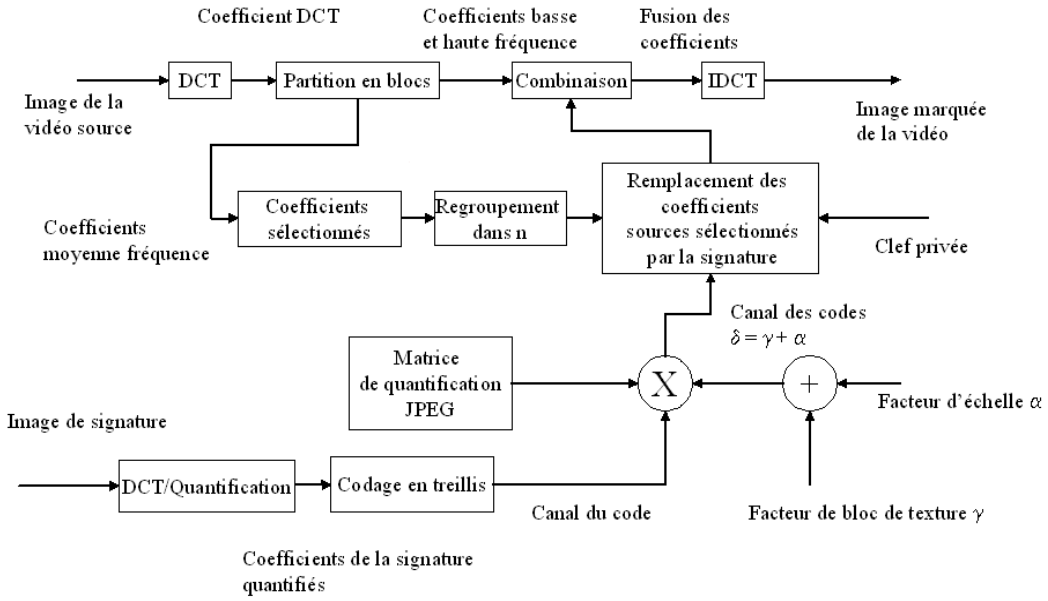


FIG. 10 – Procédure d'insertion de Chae

Technique adaptée à la vidéo

G.C. Langelaar *et al.* [71] et G. Doërr *et al.* [67] ont proposé de bons états de l'art sur le tatouage vidéo. Le premier présente les approches basiques, le second quant à lui présente un bon point de vue sur la problématique actuelle du tatouage.

• F. Hartung et B. Girod [60] proposent deux méthodes d'insertions. La première, dans le domaine non-compressé, basée sur une règle additive dans le domaine spatial. La seconde technique, qui représente une variante de la première, est mise en oeuvre dans le domaine compressé. Une vidéo peut être analysée suivant plusieurs points de vue, le cas d'une succession d'images $2D$ (en occultant la dimension temporelle), ou comme étant un signal $2D + t$, on encore la considérer comme étant un signal $3D$ (dans ce cas la dimension temporelle n'est pas séparée des dimensions spatiales), ou enfin comme un signal $1D$. C'est cette dernière approche que les auteurs ont retenue, comme présenté figure 14.

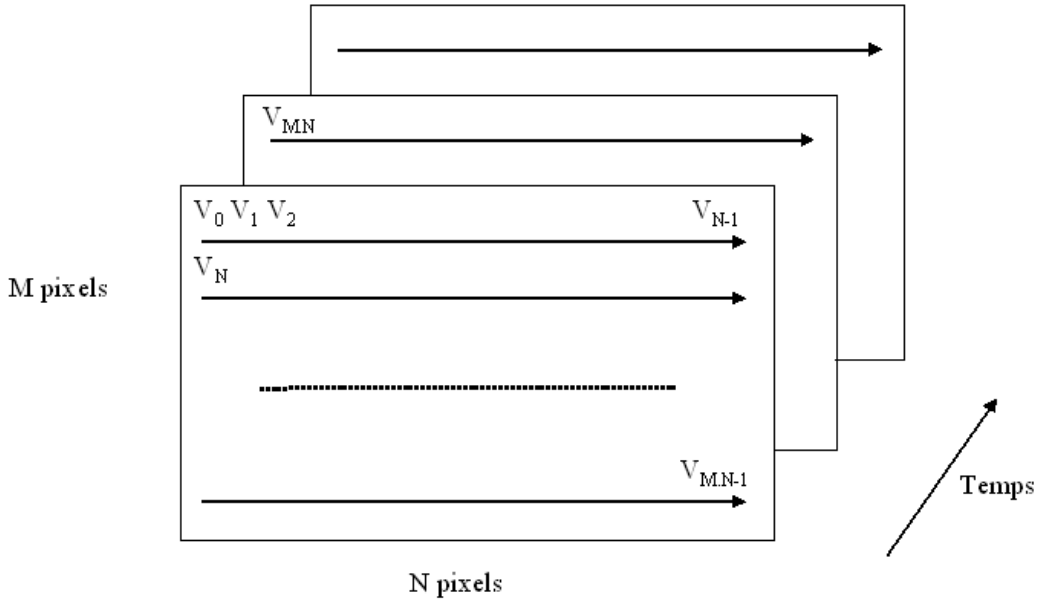


FIG. 11 – représentation d'un signal vidéo en un signal 1D

Génération de la marque :

On souhaite insérer la marque suivante :

$$a_j, \;\; a_j \epsilon \{-1, 1\}, \;\; j \epsilon N \tag{5}$$

Cette marque est tout d'abord sur-échantillonnée d'un facteur cr, afin de créer un signal redondant, permettant de le rendre plus robuste.

$$b_i = a_j, \;\; j.cr \leq i \leq (j+1).cr, \;\; i \epsilon N \tag{6}$$

On module ensuite le signal obtenu par un bruit binaire pseudo-aléatoire, qu'on amplifie ensuite par un facteur α :

$$\begin{cases} p_i, \;\; p_i \epsilon \{-1, 1\}, \;\; i \epsilon N \\ w_i = \alpha_i . b_i . p_i, \;\; i \epsilon N \end{cases} \tag{7}$$

Insertion de la marque :

Le tatouage est ajouté de la façon suivante :

$$\tilde{v_i} = v_i + \alpha_i . b_i . p_i, \;\; i \epsilon N \tag{8}$$

La figure 12 explicite la phase d'insertion de la marque.

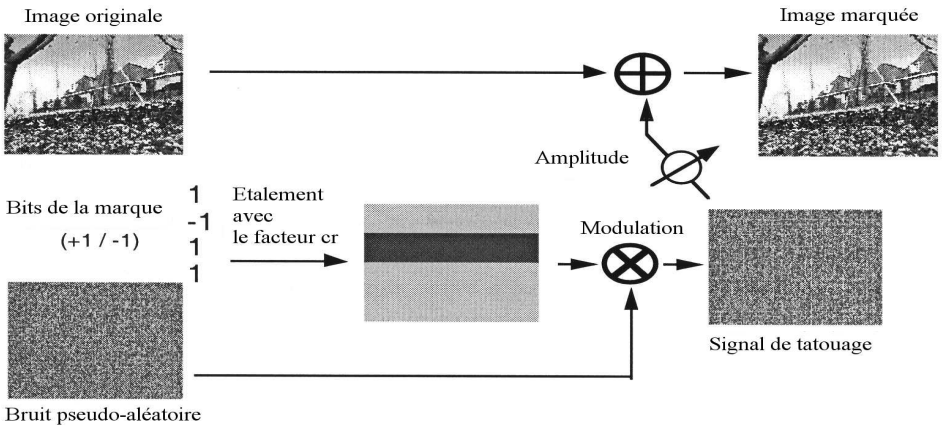


FIG. 12 – Procédure d'insertion de Hartung

Extraction de la marque :

Pour l'extraction, la vidéo originale n'est pas nécessaire. Avant de commencer la détection, l'image est filtrée par un filtre passe bas. Le signal obtenu $\overline{\tilde{v}}$ est démodulé en le multipliant par le bruit utilisé lors de l'insertion de la marque. On calcule ensuite la corrélation entre la marque et le signal obtenu.

$$s_j = \sum_{i=j.cr}^{(j+1).cr-1} p_i.\overline{\tilde{v}} = \sum_{i=j.cr}^{(j+1).cr-1} p_i.\overline{v_i} + \sum_{i=j.cr}^{(j+1).cr-1} p_i.\overline{p_i \alpha_i b_i} \tag{9}$$

on pose :

$$\begin{cases} \sum_1 = \sum_{i=j.cr}^{(j+1).cr-1} p_i.\overline{v_i} \\ \sum_2 = \sum_{i=j.cr}^{(j+1).cr-1} p_i.\overline{p_i \alpha_i b_i} \end{cases} \tag{10}$$

$\sum_1$ et $\sum_2$ correspondent à la contribution du signal vidéo filtré, ainsi qu'à la marque filtrée au calcul de la corrélation. On peut supposer que le filtrage élimine la composante vidéo dans le signal et que son influence sur la marque est négligeable :

$$s_j = \sum_1 + \sum_2 = \sum_{i=j.cr}^{(j+1).cr-1} p_i^2.\alpha_i b_i = a_j \sigma_p^2.cr.moy(\alpha_i) \tag{11}$$

où σ_p^2 est la variance du bruit uniforme.

On peut donc déduire la marque du signe de s_j :

$$signe(s_j) = signe(a_j \sigma_p^2 .cr.moy(\alpha_i)) = signe(a_j) = a_j \quad (12)$$

La figure 13 résume la phase d'extraction.

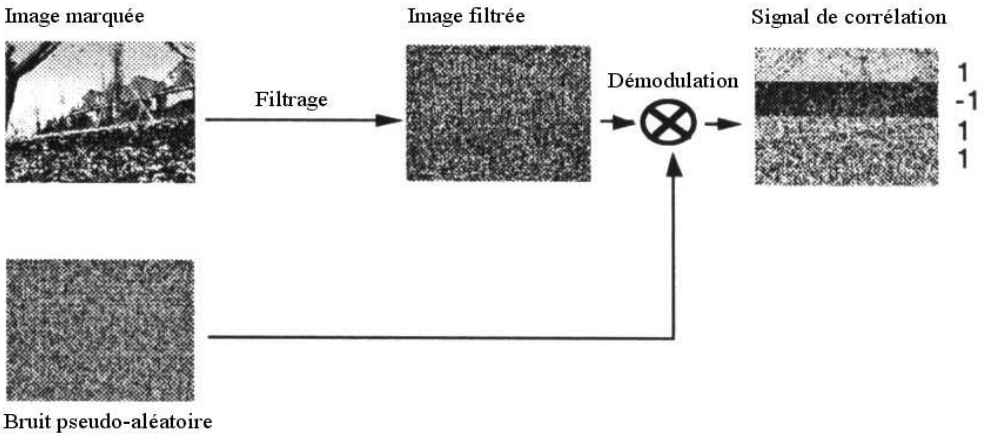


FIG. 13 – Procédure d'extraction

Selon les auteurs, cette méthode peut résister à la plupart des attaques (filtrage, compression, cropping ...) pourvu que les paramètres d'insertion soient choisis judicieusement. Cependant, ils ne développent pas l'aspect d'invisibilité de la marque, on peut donc supposer que le marquage est résistant lorsque la force d'insertion est élevée, mais en contrepartie, les artefacts qui apparaissent deviennent probablement plus nombreux et plus visibles. Pour la détection de la marque, les auteurs font l'hypothèse qu'un filtrage adéquat permet d'enlever l'influence de la vidéo dans le calcul de corrélation mais dans la pratique cette hypothèse n'est pas totalement vérifiée. Cela engendre des erreurs lors de la détection, celles-ci pouvant probablement être compensées par l'utilisation d'un code correcteur d'erreurs.

Dans leurs secondes approches, ils proposent un schéma de marquage qui s'applique directement au flux compressé en ne marquant que les images intra.

Génération de la marque :

Pour chaque image intra, une marque est générée, selon le même principe que dans le schéma proposé dans le domaine non compressé. La vidéo n'est pas considérée comme un signal mono dimensionnel. C'est la marque qui est transformée en un signal $2D$ de même taille que les images de la séquence. Pour chaque bloc $8 * 8$ du train binaire, le bloc $8 * 8$ correspondant au signal de la marque est transformé dans le domaine DCT.

Insertion de la marque :

L'insertion se fait dans le domaine DCT, sur les images intra. Les blocs DCT de l'image et ceux correspondant à la marque sont ajoutés. Afin de ne pas augmenter le débit de la vidéo, seuls les blocs marqués dont la taille est inférieure ou égale au bloc original sont sélectionnés. Les figures 14 et 15 explicitent la procédure d'insertion.

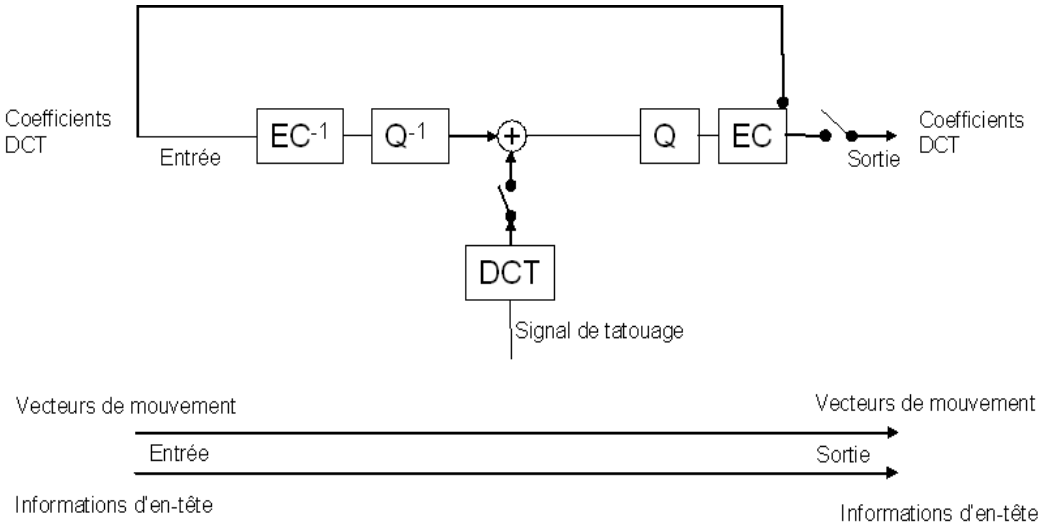


FIG. 14 – Procédure d'insertion

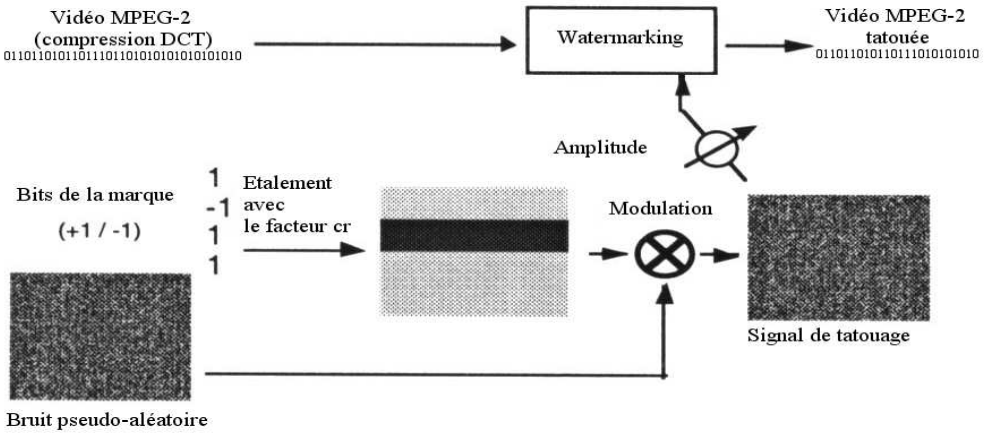


FIG. 15 – Procédure d'insertion

Le fait de modifier les images intra entraîne des perturbations dans les images P et B, on parle d'effet de dérive. Pour compenser cet effet, un signal de compensation de dérive est ajouté. On le déduit facilement, car il correspond à la différence entre la prédiction du mouvement entre l'image k et $k+1$ et la prédiction du mouvement entre l'image k' (image marquée) et l'image $k+1$. La figure 16 explicite cette procédure.

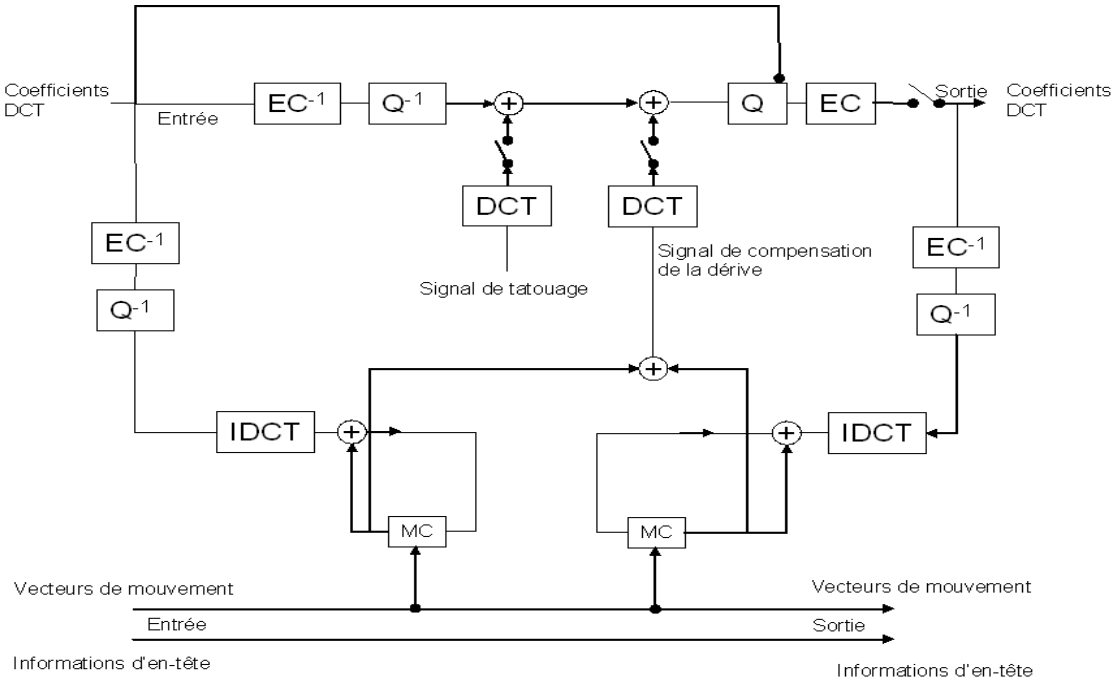


FIG. 16 – Procédure d'extraction

• V. Darmstaeder & al. [152] proposent d'insérer une marque calculée à partir du spectre basse fréquence de séquences vidéos de blocs 8×8. Les blocs sont tout d'abord classés en fonction de leur activité. Les blocs de faible activité ne sont pas marqués. Un motif basse fréquence pseudo-aléatoire est alors ajouté à chaque bloc sélectionné. Selon cette méthode, chaque bloc comprend une information d'un bit, cette information étant répétée sur plusieurs blocs et plusieurs images.

• En 97, G.C. Langelaar & al. [72] proposent 2 types d'insertion de marque pour la vidéo compressée. Nous appellerons la première, méthode de camouflage des données, et la seconde, méthode de marquage. La deuxième méthode est plus complexe, mais également plus robuste. Elle est basée sur la séparation de certaines parties du "bitsream" de la vidéo compressée. Pour chaque bit d'information à insérer, un ensemble de n blocs 8×8 est choisi pseudo-aléatoirement à partir des images de la vidéo, et est divisé pseudo-aléatoirement en deux sous ensembles de même taille.

Le nombre n de blocs varie typiquement entre 16 et 64. Pour chaque sous ensemble, l'énergie des coefficients DCT hautes fréquences est calculée. Afin d'insérer un bit, l'énergie des coefficients hautes fréquences d'un des sous ensembles est réduite en ôtant des coefficients hautes fréquences. Pour faciliter la compréhension de l'approche, des blocs consécutifs sont utilisés plutôt que des blocs aléatoirement choisis. Le bit d'information peut alors être extrait, en sélectionnant le même ensemble de blocs, en le divisant en sous ensembles, et en comparant l'énergie des coefficients hautes fréquences de chaque sous ensemble. Cette méthode nécessite seulement un décodage partiel sans ré-encodage. Cependant, la robustesse de la marque ainsi insérée est limitée, car le ré-encodage augmente le taux d'erreur des bits insérés. De plus, cette méthode ne résiste pas à un nouvel encodage utilisant un autre GOP (group of Picture), différent de celui utilisé lors du marquage, puisque les coefficients DCT sont différents suivant que l'image est codée en I, P ou B.

• N. Checcacci et al. [115] présentent un schéma de tatouage de vidéo testé dans un contexte de communication multimedia sans fil. Le système de marquage est adapté à des schémas de codage tels que MPEG4 ou H263. Ce contexte impose certaines contraintes, propres à ce type de communication, et notamment la nécessité de résister à la perte d'images ou à la désynchronisation temporelle. L'insertion se fait directement sur le flux compressé (MPEG-4) en sélectionnant des paires de coefficients, et peut se faire sur les objets tels qu'ils sont définis dans la norme MPEG4. La vidéo originale n'est pas nécessaire pour la détection. La capacité du système est de 15 bits. Un aperçu de cette approche est donné sur les figures 17 et 18.

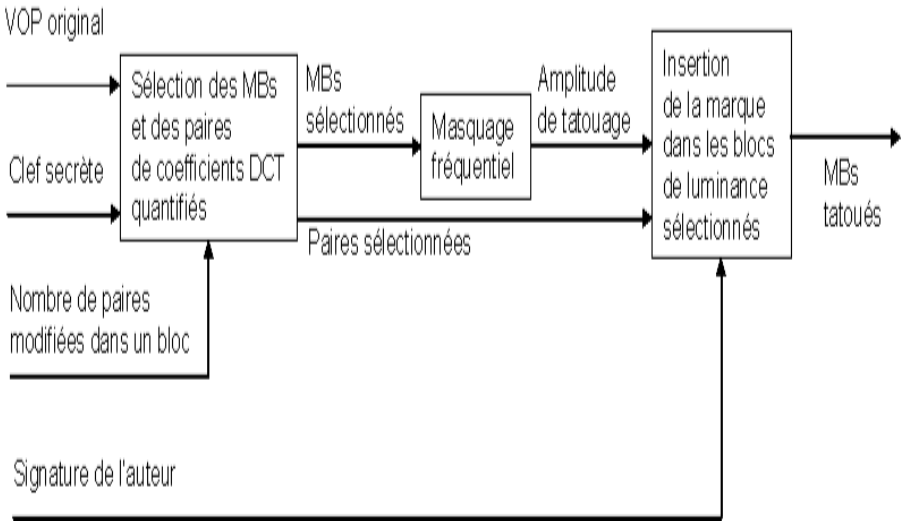


FIG. 17 – algorithme d'insertion de Checcacci

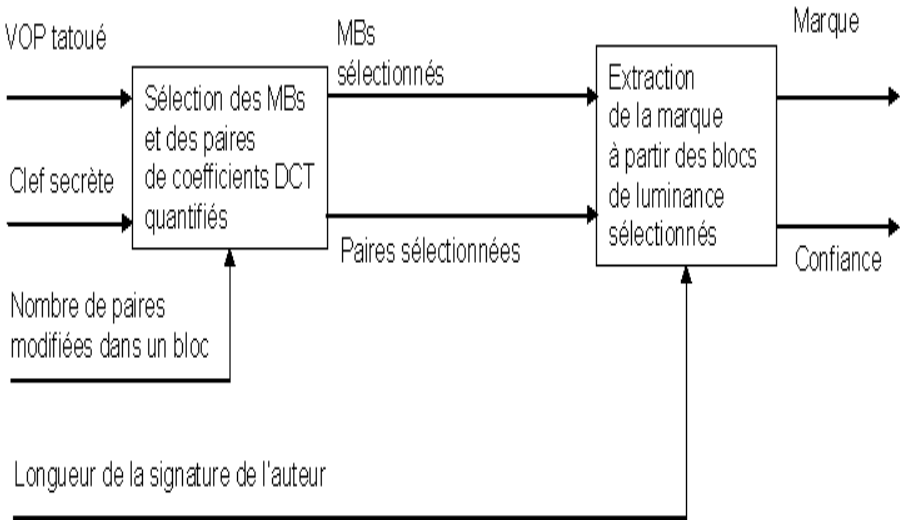


FIG. 18 – algorithme de détection de Checcacci

• M.D. Swanson & al. [111] proposent une méthode multi-échelle de marquage appliquée à de la vidéo non compressée, méthode disposant de propriétés intéressantes. La première étape consiste à segmenter en scènes la vidéo à marquer. Chaque scène est considérée comme une entité à part entière. Un filtrage temporel en ondelettes est alors appliquée sur chaque scène vidéo. La marque utilisée n'est pas une marque arbitraire, mais plutôt un code identifiant le propriétaire. Cette marque est insérée dans chaque composante temporelle et les coefficients marqués sont alors transformés de manière inverse pour obtenir une vidéo marquée. De plus, la marque est constituée de différentes composantes, certaines varient au cours du temps, tandis que d'autres restent fixes puisqu'elles sont insérées dans les coefficients représentant les fréquences temporelles basses. L'existence de ces composantes variables, permet de pallier aux problèmes de moyennage, et d'autoriser la détection d'une image d'une scène sans connaissance de son actuel index. La marque représente de plus un signal dépendant d'une clé afin d'éviter les "deadlock" problèmes. A cela s'ajoute un modèle HVS afin d'exploiter le masquage temporel et spatial. Un apercu de la procédure est présenté sur la figure 22.

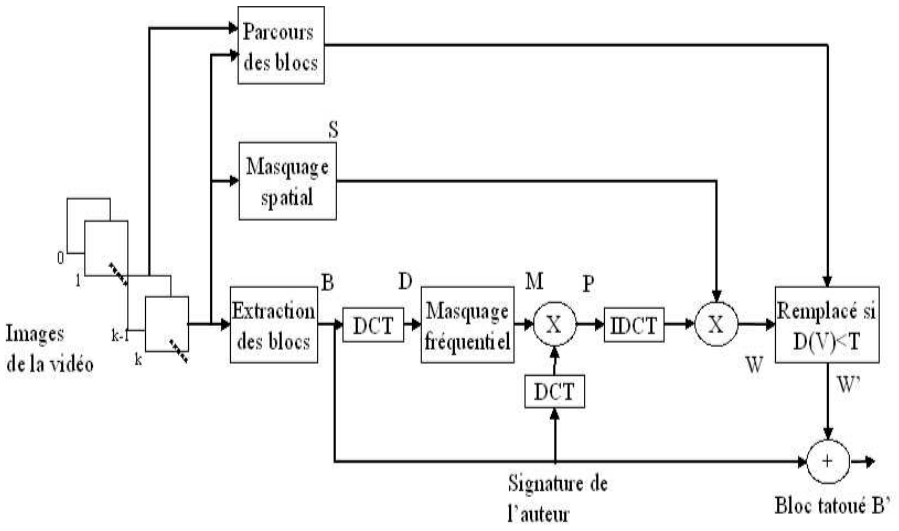


FIG. 19 – Procédure d'insertion de Swanson

• F. Deguillaume & al. [58], considèrent la séquence vidéo comme un signal à 3 dimensions (deux spatiales, une temporelle). La signature est ajoutée après avoir effectué une transformée de Fourier 3D d'un bloc de la séquence. La partie du spectre qui sert d'espace d'insertion correspond aux fréquences moyennes du plan spatial et aux fréquences moyennes de l'axe temporel. Sur cet axe, les basses fréquences sont associées aux objets en faibles mouvements et les hautes fréquences sont associées aux objets en déplacements rapides. L'insertion de la signature s'effectue en ajoutant une séquence aléatoire aux coefficients fréquentiels appartenant à l'espace d'insertion. L'algorithme présenté ici repose sur les propriétés de la transformée de Fourier discrète 3D.

La 3D DFT :

On considère une séquence vidéo comme étant une fonction f(x,y,z) continue à valeurs réelles définies dans un repère cartésien, dont les coordonnées sont des entiers, de dimension $N_x * N_y * N_z$, avec $0 \leq x \leq N_x, 0 \leq y \leq N_y$ et $0 \leq z \leq N_z$.

Sa transformée de Fourier discrète 3D est égale à :

$$F(k_x, k_y, k_z) = \sum_{x=0}^{N_x-1} \sum_{y=0}^{N_y-1} \sum_{z=0}^{N_z-1} f(x,y,z) e^{\frac{-2j\pi x k_x}{N_x} - \frac{-2j\pi y k_y}{N_y} - \frac{-2j\pi z k_z}{N_z}} \tag{13}$$

La transformée inverse est donnée par :

$$f(x,y,z) = \sum_{x=0}^{N_x-1} \sum_{y=0}^{N_y-1} \sum_{z=0}^{N_z-1} (k_x, k_y, k_z) e^{\frac{2j\pi x k_x}{N_x} + \frac{2j\pi y k_y}{N_y} + \frac{2j\pi z k_z}{N_z}} \tag{14}$$

Certaines propriétés de la transformée de Fourier permettent de rendre plus résistant le marquage

face aux attaques géométriques suivantes : translation, rotation, et changement d'échelle. Ces propriétés sont :
- Le changement d'échelle :

$$F(\frac{k_x}{\lambda_x}, \frac{k_y}{\lambda_y}, \frac{k_z}{\lambda_z}) \Leftrightarrow f(\lambda_x.x, \lambda_y.y, \lambda_z.z) \tag{15}$$

λ_x et λ_y sont les facteurs d'échelle spatiale, et λ_z est le facteur d'échelle temporel.

- La rotation :

$$F(k_x.cos(\theta)-k_y.sin(\theta),\ k_x.sin(\theta)+k_y.cos(\theta),\ k_z) \Leftrightarrow f(x.cos(\theta)-y.sin(\theta),\ x.sin(\theta)+y.cos(\theta),\ z) \tag{16}$$

θ représente l'angle de rotation dans le domaine spatial.

- La translation :

$$F(k_x, k_y, k_z).e^{-j(a.k_x+b.k_y+c.k_z)} \Leftrightarrow f(x+a,\ y+b,\ z+c) \tag{17}$$

Génération de la marque :
Soit $b = (b1,\ b2,\ ...,\ bM)^T$ le message à insérer. Pour chaque bit b_i, une séquence pseudo-aléatoire v_i est générée. La famille $v_1,\ ...,\ v_M$ est obtenue à partir de m-séquences ou d'une famille de "gold codes" (utilisés pour leurs propriétés statistiques et cryptographiques intéressantes). Le message encodé peut être obtenu de la façon suivante :

$$w = \sum_{i=1}^{M} b_i.v_i = G_b \tag{18}$$

G_b est une matrice N*M tel que la ième colonne de G_b soit un vecteur pseudo-aléatoire v_i dont les valeurs valent $+1$ ou -1.
Le vecteur w obtenu représente la marque sous la forme d'un signal à spectre étalé. Il sera inséré dans l'amplitude de la DFT 3D.
Insertion de la marque :
Grâce à sa propriété de séparabilité, la DFT 3D peut être considéré comme une DFT 2D dans le domaine spatial suivi d'une DFT 1D temporel. Pour obtenir un compromis entre visibilité et robustesse, le domaine spatial est réduit à une bande de moyennes fréquences délimitée par r_min et r_max. De même, le domaine temporel est restreint à la bande délimitée par d_min et d_max.
Comme le tatouage w possède des valeurs positives et négatives, $w_i \epsilon \{-M, M\}$, l'ajout des w_i aux valeurs d'amplitudes peut les rendre négatives, et par conséquent, la DFT aura des valeurs non

réelles. Pour éviter ce cas de figure, une nouvelle représentation des valeurs positives et négatives est introduite :

$$w_i \rightarrow p_i = \begin{cases} (w_i, 0) & \text{si } w_i \geq 0 \\ (0, -w_i) & \text{si } w_i < 0 \end{cases} \tag{19}$$

En utilisant cette représentation, et en choisissant une paire d'amplitudes pour chaque p_i, la première valeur est modifiée en ajoutant w_i, si $w_i > 0$, et la seconde valeur est modifiée en ajoutant $|w_i|$, si $w_i < 0$. L'emplacement des paires d'amplitudes peut être choisi arbitrairement, mais la seule contrainte est que l'emplacement reste le même lors de l'insertion et de l'extraction.

Extraction de la marque :

Pour extraire la marque, la séquence vidéo est décomposée en une succession d'images fixes, puis est décomposée en blocs de taille fixe. La DFT 3D est alors appliquée sur chaque bloc, et on détermine, après resynchronisation du signal, les paires d'amplitudes. La différence entre les coefficients des paires d'amplitudes permet d'obtenir le signal $w' = w + e$, où w est le signal de la marque et e l'erreur qui représente le bruit que peut contenir l'image (suite à une attaque par exemple). Il ne reste plus qu'à décoder le message. Pour les m-séquences et les "gold codes", le produit scalaire entre v_i et v_j s'écrit :

$$\langle v_i, v_j \rangle = \begin{cases} N & \text{si } i = j \\ -1 & \text{si } i \neq j \end{cases} \tag{20}$$

Pour décoder le message w', on calcule pour chaque i le produit scalaire entre v_i et w' :

$$B_j{}' = \langle w', v_j \rangle = \sum_{i=1}^{M} b_i \langle v_i, v_j \rangle + \langle e, v_j \rangle \tag{21}$$

$$B_j{}' = b_i N - (M-1) + \langle e, v_j \rangle \tag{22}$$

Généralement, M est négligeable devant N. De plus, la distribution de e peut être approximée par une loi normale de moyenne nulle. Le second terme est donc négligeable par rapport à N. Ainsi, chaque bit caché peut être extrait grâce au signe de B'_j :

$$b'_j = signe(B'_j) = signe(b_j) = b_j \tag{23}$$

Cette technique est robuste aux traitements suivants : compression MPEG, changement de proportions et variations de débit du flux vidéo. L'utilisation d'une transformation 3D permet de diffuser la marque sur toute la vidéo et donc de rendre plus difficile les attaques statistiques classiques. De plus, l'approche 3D offre une plus grande largeur de bande pour cacher de l'information.

• T. Kalker & al. [148] en 99, proposent un schéma de tatouage pour la diffusion vidéo qui présente une complexité d'insertion et de détection faible. Ce système est appelé JAWS (Just Another Watermarking System). Il résiste à la compression MPEG2 jusqu'à un débit de 2Mbits/s. L'insertion et la détection s'effectuent dans le domaine spatial du flux décompressé. L'insertion de la signature s'effectue en ajoutant des blocs aléatoires de taille $128 * 128$ sur les différentes images de la séquence.

D'autres transformées comme la transformée de Hadamard ont été utilisées de façon anecdotique [29].

• M. Kutter & al. [103] proposent un algorithme de marquage qui insère la marque dans les vecteurs mouvements d'une vidéo de type MPEG.
Génération de la marque :
La marque générée est une suite binaire de longueur 16 bits ou 32 bits.
Insertion de la marque :
Les vecteurs de mouvement sont extraits d'une séquence compressée. Les différentes étapes de marquage s'appliquent sur l'une des deux composantes des vecteurs de mouvement. Un bloc est sélectionné aléatoirement par image, et son vecteur de mouvement est calculé. Deux bits de la marque sont alors insérés dans chaque composante du vecteur de mouvement.
Soit par exemple V, la composante verticale d'un vecteur de mouvement. Soit $b = 0, 1$, la valeur du bit à cacher.

- si ((V*q+T)mod[2] b alors V' = V + d
- sinon V' = V
- avec T = 2*dim ; dim = taille de la fenêtre de recherche pour l'estimation de mouvement
- d = (2.n+1)/q
- n = 1 si le vecteur de mouvement est le vecteur nul
- n = 0 sinon

- q = facteur de modulation de l'amplitude du vecteur de mouvement
- V' est le vecteur de mouvement marqué

Extraction de la marque :
Le vecteur V' est extrait du flux compressé. La marque est alors extraite de la façon suivante :

$$b = (V' * q + T) mod[2] \tag{24}$$

• J. Zhang & al. [92] ont amélioré cette méthode, en réalisant une sélection de la composante de plus grande amplitude du vecteur de mouvement, pour déterminer la composante à marquer.
La technique proposée dans cet article se base sur la méthode proposée par M. Kutter & al. (présentée ci-dessus). Il s'agit en fait d'une amélioration de l'approche. Le principe reste donc le même : il s'agit d'insérer une marque invisible dans les vecteurs de mouvement d'un flux vidéo, cette insertion s'effectue sur les vecteurs de mouvement ayant la plus grande amplitude.

• D'autres techniques insèrent la marque au sein des objets contenus dans une séquence, c'est le cas des techniques proposées par J. Guo & al. [86], où la marque est insérée dans le domaine DCT sur les objets issus d'une segmentation réalisée sur la vidéo. P. Bas & al. [117] insèrent leur marque dans le domaine spatial. A. Piva & al. [12] proposent eux aussi d'insérer la marque sur des objets, obtenus à partir du flux MPEG4, une DWT étant ensuite appliquée sur ces objets afin d'insérer la marque.

• C-S. Lu & al. [38] proposent un schéma de tatouage qui se base sur les objets contenus dans une vidéo, objets conformes au standard de compression MPEG4.
Génération de la marque :
Il n'y a pas de technique particulière utilisée ici pour la génération. Soit $W(i)$ la séquence binaire définissant la marque à insérer.
Insertion de la marque :
L'insertion se fait dans les objets contenus dans une scène vidéo. Une fois un objet sélectionné, ses deux vecteurs propres sont calculés. Le plus petit rectangle (ayant les mêmes orientations que l'image contenant l'objet) qui contient l'objet à marquer est considéré comme étant une image I. Cette image I est transformée dans le domaine DCT, puis les coefficients présents dans les moyennes fréquences sont sélectionnés. Un nombre $n = |W|$ de coefficients est ainsi sélectionné. Soit $f(i)$ les coefficients sélectionnés, la procédure suivante est utilisée pour insérer la marque :

$$\overline{f(i)} = H(f(i)) \text{ avec H : filtre moyenneur} \tag{25}$$

$$d(i) = f(i) - \overline{f(i)} \tag{26}$$

Si le nombre de $d(i)$ est suffisamment important, ceux-ci peuvent être normalisés selon une distribution gaussienne :

$$d^g(i) = \frac{d(i) - \mu}{\sigma} \tag{27}$$

où μ et σ sont respectivement la moyenne et la variance de l'échantillon $d(i)$. $d^G(i)$ est remplacé par :

$$d^h(i) = w(i).\sigma + \mu \tag{28}$$

Le coefficient DCT est ensuite obtenu à partir de la formule suivante :

$$f^h(i) = \overline{f(i)} + d^h(i) = \overline{f(i)} + w(i).\sigma + \mu \tag{29}$$

La distance MSE est ensuite calculée entre le coefficient $f(i)$ et $f^h(i)$. Si la valeur obtenue dépasse un seuil S alors $f^h(i)$ est calculé à partir de la formule suivante :

$$f^h(i) = \overline{f(i)} + (w(i).\lambda + d^g(i).(1-\lambda).\sigma + \mu \tag{30}$$

où λ est un paramètre qui permet de déterminer le bon compromis entre robustesse et invisibilité.
Détection de la marque :
Pour la détection, le paramètre λ est supposé égale à 1. L'image est filtrée par un filtre moyenneur, les coefficients DCT ainsi obtenu sont de la forme suivante :

$$\overline{f^h(i)} = \frac{1}{l}\sum_{t=1}^{l}(\overline{f(t)} + w(t).\sigma + \mu) = \frac{1}{l}\sum_{t=1}^{l}\overline{f(t)} + \frac{1}{l}\sum_{t=1}^{l}(w(t).\sigma + \mu \approx \overline{f(i)} \tag{31}$$

Dans l'équation ci-dessus, le second terme est proche de 0, car, lors de l'insertion, la marque a une distribution gaussienne de moyenne nulle. Le signal extrait de la vidéo est alors défini comme suit :

$$S^e(i) = f^h(i) - \overline{f^h(i)} \approx f^h(i) - \overline{f(i)} = w(i).\sigma + \mu \tag{32}$$

Le signal ainsi extrait est égal à $d^h(i)$. Si l'image a subi des "attaques", le processus de détection devient :

$$f^a(i) = f^h(i) + a(i) = \overline{f(i)} + (w(i).\sigma + \mu) + a(i) \tag{33}$$

où $a(i)$ est l'effet de l'attaque.
Après filtrage, nous avons :

$$\overline{f^a(i)} = \overline{f^h(i)} + \overline{a(i)} \approx \overline{f(i)} + \overline{a(i)} \tag{34}$$

Finalement, le signal extrait est défini par :

$$S^e(i) = f^a(i) - \overline{f^a(i)} \approx (w(i).\sigma + \mu) + (a(i) - \overline{a(i)}) \tag{35}$$

Enfin, pour détecter la présence de la marque, la corrélation entre $w(i)$ et $s^e(i)$ est déterminée par la relation :

$$\sum_{i=1}^{|W|} w(i).S^e(i) \approx \sigma \sum_{i=1}^{|W|} w^2(i) + \sum_{i=1}^{|W|} w(i).(a(i) - \overline{a(i)}) \quad (36)$$

Techniques de tatouage combinées à d'autres traitements

• A. Bhardwaj & al. [6] présentent un système de tatouage lié à un système d'indexation.

Génération de la marque :

Pour la génération de la marque les auteurs utilisent des codes correcteurs, pour rendre la marque plus robuste. Ils modulent ensuite la marque obtenue par un bruit pseudo aléatoire (gaussien), qui est initialisé par un germe, obtenu à partir des 3 premiers moments de l'image. Ces 3 premiers moments sont calculés afin de premièrement, détecter les changements de scènes qui ont lieu dans la vidéo à marquer (relation avec l'indexation) et deuxièmement, d'extraire des images significatives qui serviront à insérer la marque. Enfin, un facteur de pondération adaptatif est utilisé dans le but de rendre la marque invisible.

Insertion de la marque :

L'insertion se fait dans le domaine compressé, sur les coefficients DCT. La marque est d'abord transformée dans ce domaine, pour ensuite être ajoutée aux coefficients non-nuls des "key-frames" sélectionnées par la méthode d'indexation.

Détection de la marque :

La détection se fait par des mesures de corrélation après avoir filtré l'image (afin d'améliorer les performances de la détection).

• R. Dugad & al. [123] présentent un système de tatouage lié à un système de compression. Deux schémas d'insertion sont ainsi proposés.

Génération de la marque :

La marque est modulée par un bruit uniforme, de la taille de l'image à marquer. La force du marquage est contrôlée par un paramètre qui varie en fonction du type de l'image marquée (I, B ou P).

Insertion de la marque :

Premier schéma : L'insertion se fait dans le domaine compressé, sur les coefficients DCT des images I, B et P. La force d'insertion est différente, selon si l'on marque une image I, B ou P.

Second schéma : L'insertion se fait sur les images I, dans le domaine DWT et non dans le domaine DCT. Le principe d'insertion est le même que dans le premier schéma (mais seule les images I sont marquées).

Détection de la marque :
La détection se fait par corrélation.

• R. Lancini & al. [124] proposent un schéma de tatouage dans le but d'insérer des informations de copyright, ainsi que des informations d'indexation, dans une vidéo. L'algorithme s'exécute dans le domaine spatial. Les auteurs proposent la construction d'un masque global, basé sur la construction de trois masques prenant en compte les aspects de masquage de luminance, de texture et temporels. Enfin, un code correcteur d'erreurs ainsi qu'un motif de synchronisation sont utilisés.

La stéganographie et le tatouage ont des applications multiples. L'une des applications originales correspond à la détection d'erreurs dans la transmission de flux vidéo compressés. En effet, les vidéos compressées sont moins robustes aux erreurs de transmission, qui peuvent se propager sur les images prédites, donnant lieu à des artefacts gênants (c.f. la structure I.B.P des schémas de compression du type MPEG2). Afin d'atténuer ce phénomène, on utilise des systèmes de détection et de correction d'erreurs (les techniques les plus couramment utilisées sont les codes correcteurs d'erreurs du type Reed-Solomon, BCH, Turbo-codes...). Le principal problème de ces systèmes repose sur l'augmentation considérable de la taille des données à transmettre, alors qu'un système de stéganographie a peu d'impact sur la taille des données. En contrepartie, le système stéganographique dégrade légèrement la qualité de l'image, sans que cela soit perceptible. Ces systèmes sont donc utilisés pour la détection d'erreurs associés à des systèmes de correction d'erreurs.

• D.L. Robie &al. [52] proposent un système de détection d'erreurs qui code, dans l'image $n+1$, des informations de l'image n. Pour insérer ces informations, ils utilisent une règle classique :

$$AC_{trans} = \begin{cases} AC_{org} & \text{si } AC_{org} \leq T \\ AC_{org} + I_{stego} & \text{si } AC_org > T \end{cases} \tag{37}$$

où T est un seuil permettant de contrôler l'impact du marquage sur l'image.
Ainsi, ils utilisent les coefficients AC de la DCT pour insérer leurs informations, qui correspondent au nombre de bits de chaque macrobloc (on peut assimiler cette information à une somme de contrôle) ainsi que le coefficient DC final.

• F. Bartolini & al. [57] proposent également un système de détection d'erreurs basé sur la

stéganographie. Le principe est d'imposer une règle à un sous-ensemble de coefficients DCT, sélectionné de façon à ne pas dégrader visuellement l'image. Les coefficients sont modifiés selon la règle suivante :

$$v_i' = \begin{cases} v_i - 1 & \text{si (i>T et i impair et } v_i \text{ pair) ou (i>T et i pair et } v_i \text{ impair)} \\ v_i & \text{sinon} \end{cases} \quad (38)$$

Ainsi, on peut détecter la présence (ou non) de la marque, en vérifiant que les coefficients sélectionnés obéissent à la règle ci-dessus. Si cette dérive est vérifiée, alors, il n'y a pas eu d'erreurs. Sinon, le bloc est corrompu.

Ces deux techniques sont une des applications possibles à la stéganographie. Dans ce contexte, on ne se pose pas de contraintes de robustesse. La mise en oeuvre des techniques proposées est relativement simple, et peut s'adapter à tout type de compression.

Auteurs	Caractéristiques	Avantages	Inconvénients
Hsu & al.	Tatouage du flux compressé Tatouage non aveugle Insertion d'un logo Modification des coefficients DCT	Utilisation d'un motif visuel	Nécessite l'utilisation de la vidéo originale
Dittman & al.	Adaptation à la vidéo de l'algorithme de Fridrich Tatouage aveugle Insertion d'un motif pseudo-aléatoire	Ne nécessite pas la vidéo originale	
Dittman & al.	Adaptation de l'algorithme de Koch&Zhao Marquage de la luminance	Facile à intégrer dans un codec de type MPEG Ne nécessite pas la vidéo originale	Possibilité d'artefacts Non robuste aux changements d'échelle et aux rotations
Chae & al.	Insertion d'image ou de vidéo dans le domaine DCT	Grande capacité Ne nécessite pas la vidéo originale	

FIG. 20 – Récapitulatif des systèmes de tatouage inspirés de l'image fixe

Auteurs	Caractéristiques	Avantages	Inconvénients
Hartung & al.	Tatouage du flux non compressé Tatouage aveugle Signal vidéo 2D+T transformé en signal 1D	Rapide à mettre en œuvre Bonne robustesse	Pas de discussion robustesse/ invisibilité Hypothèse pour l'extraction discutable
Hartung & al.	Tatouage du flux compressé Insertion dans le domaine DCT Tatouage des images Intra	Prise en compte des effets de dérive	
Darmstaeder & al.	Insertion sur des blocs 8*8	Prise en compte de l'activité des blocs	
Langelaar & al.	Insertion d'image ou de vidéo Tatouage du flux compressé	Rapidité de la procédure (utilisation d'un décodage partiel)	Faible robustesse à un réencodage
Checcacci & al.	Insertion dans le flux compressé	Contexte de communication multimédia sans fil Ne nécessite pas la vidéo originale	Faible capacité
Swanson & al.	Méthode multi-échelle Tatouage dans le domaine non compressé	Segmentation en scène Masquage temporel et spatial	
Deguillaume & al.	Insertion dans le domaine fréquentiel	Utilisation d'une transformation 3D Bonne robustesse	Système complexe
Kalker & al.	Insertion dans le domaine spatial du flux décompressé	Complexité d'insertion et de détection faible Grande maturité du système	
Jordan & al.	Tatouage des vecteurs de mouvement	Prise en compte de l'axe temporel exclusivement	Faible robustesse de la règle de parité
Zhang & al.	Amélioration de l'algorithme de Jordan	Prise en compte de l'axe temporel exclusivement	Amélioration non significative
Lu & al.	Insertion de la marque sur des objets conformes au standard MPEG4	Prise en compte du standard MPEG4 et du concept d'objet	

FIG. 21 – Récapitulatif des systèmes de tatouage adaptés à la vidéo.

Auteurs	Caractéristiques	Avantages	Inconvénients
Bhardwaj & al.	Combinaison tatouage / indexation Tatouage dans le domaine compressé	Problématique intéressante	
Dugad & al.	Combinaison tatouage / compression Tatouage dans le domaine compressé	Problématique intéressante	
Lancini & al.	Insertion d'information de copyright et d'indexation Tatouage dans le domaine spatial	Utilisation des principes de masquage Utilisation d'un motif de synchronisation	
Robie & al.	Utilisation du tatouage pour la détection d'erreur	Pas de contrainte de robustesse	
Bartolini & al.	Utilisation du tatouage pour la détection d'erreur	Pas de contrainte de robustesse	

FIG. 22 – Récapitulatif des systèmes de tatouage combinés à d'autres traitements.

2.4.3 Logiciels vidéos

Macrovision et Digimarc ont développé des prototypes susceptibles de protéger les DVDs, le câble, les transmissions satellites, des copies illégales. Le système résultant devrait compléter le système de protection analogique de Macrovision, utilisé dans la génération courante des lecteurs DVD. Les marques numériques sont cachées dans un flux vidéo actif, par un codeur qui analyse chaque image, et qui règle de manière algorithmique la marque, sur une base image par image. Idéalement, le codeur permet de générer plusieurs niveaux de robustesse, en fonction de l'insertion redondante de messages définis arbitrairement, sur les images inter et intra. Le marquage peut être réalisé sur le signal de base, ou dans le domaine DCT. Les implémentations des prototypes de Macrovision et de Digimarc, pour les applications de protection de copies sont maintenant disponibles pour des démonstrations aux clients des OEM (OEM=Original Equipment Manufacturer).
En parallèle, 5 compagnies (groupe GALAXY) se sont alliées pour mettre au point un prototype de protection de contenus. Ces compagnies (Hitachi, Ltd. , IBM Corporation, NEC Corporation, Pioneer Electronic Corporation et Sony Corporation) ont décidé d'unifier leurs approches, pour créer des marques invisibles dans les vidéos. IBM et NEC combinent leurs efforts pour concevoir

un système de marquage robuste et invisible. Quant à Hitachi, Pioneer et Sony , ils se concentrent sur le développement de leur expertise, dans les produits de consommation et des technologies de marquage. En combinant leurs expertises, ces 5 compagnies espèrent réaliser une technique de marquage de performance optimale, aussi bien pour les futurs standards hautes définitions, pour la distribution de cinéma numérique, que pour toutes les applications d'images numériques hautes résolutions. La technique mise en oeuvre est composée de Primary Mark et Copy Mark. Primary Mark est une marque de 8 bits insérée dans une vidéo numérique, composée de la copie du contrôle de l'information (CCI), qui représente soit une copie, soit aucune copie ou soit une copie libre, de l'index d'APS (APS trigger est une suite de bits qui définit la sortie analogique). Suivant la valeur de ces bits, il est possible (ou non) d'enregistrer la source sur un magnétoscope. Les derniers bits sont réservés pour l'utilisation d'un contenu propriétaire. Copy Mark est une autre marque transparente, n'interférant pas avec Primary Mark. Les enregistreurs DVD l'insèrent pour remplacer par exemple le "one copy" par le statut de "no more copy". La technologie "Galaxy" offre un Primary Mark et un Copy Mark robuste et transparent. Ils résistent à la conversion analogique-Numérique et au codage MPEG2.

En parallèle aux activités de recherche des industries et des universités, plusieurs projets internationaux de recherche ont vu le jour, fondés par une communauté européenne, qui souhaite développer des techniques de marquage pratiques. TALISMAN (ACTS project AC019 [1]) souhaite proposer aux fournisseurs de services de l'union européenne, un mécanisme standard de protection de copies, afin de lutter contre la piraterie. Le résultat attendu par TALISMAN est un système de protection des séquences vidéos, par l'intermediaire du marquage et de l'étiquetage. OCTALIS est un projet dont le but est similaire à TALISMAN : mettre en oeuvre une approche globale de protection de copies efficace, et démontrer sa validité pour un grand nombre d'essais sur internet ainsi que sur le réseau EBU (European Broadcasting Union). Le consortium de standardisation internationale est aussi intéressé par les techniques de marquage. L'émergence du standard de compression vidéo MPEG4 (ISO/IEC 14496) dispose d'une structure qui autorise l'intégration d'un système de cryptage et de marquage intéressant.

Cependant, malgré tous les efforts, les systèmes de marquage sont toujours immatures, et un grand nombre de questions restent sans réponse. L'aspect théorique est très fragile, et la plupart des systèmes sont basés sur l'heuristique. Un autre inconvénient provient de la difficulté à comparer les systèmes de marquage. Tant que ces derniers ne seront pas évalués par des tests sophistiqués, le danger de produire des systèmes fragiles et vulnérables reste entier et discrédite le concept.

Dans la suite de ce chapitre, nous allons exposer quelques logiciels de tatouage, disponibles à ce jour.

[1] http ://www.tele.ucl.ac.be/PROJECTS/TALISMAN/

Watercast

Le système Watercast de Philips a été le premier système de marquage vidéo. Ce système est basé sur le marquage vidéo. Des détecteurs ajoutent une identification des données au signal vidéo pour la distribution des DVDs. Ces identifiants sont imperceptibles, et peuvent être utilisés dans une large gamme d'applications (Copyright communication, tracking, triggering ...). L'insertion watercast temps réel assure une transparence complète du marquage, sans perte de qualité vidéo. Le 18 septembre 2001 Philips Digital Networks (entité de Royal Philips Electronics) et Digimarc corporation, ont annoncé leur collaboration pour le développement et la commercialisation de solutions, basées sur le système de marquage vidéo watercast de Philips. Philips va donc continuer à développer et commercialiser le système watercast. Quant à Digimarc, qui agit en tant que société de service, propose l'enregistrement d'identifiant de marque de vidéo ainsi que le développement d'applications. Les deux compagnies se focalisent donc sur le développement de solutions complètes, pour les applications de type : Copyright, tracking, triggering.

AlpVision

AlpVision a mis en place un logiciel de marquage vidéo, pouvant être utilisé pour le tracking, le fingerprinting, le copyright, etc. Ils revendiquent la mise en oeuvre d'une solution de marquage vidéo très robuste, et d'une complexité calculatoire très faible. Cette solution est, de plus, combinée à un procédé statistique. Cette analyse est réalisée sur plusieurs images afin de corriger des erreurs éventuelles, qui proviendraient d'une mauvaise détection de la signature.

VideoMark

Videomark est un logiciel de marquage vidéo. Les marques utilisées représentent des identifiants, appelés clés de tatouage. Chaque copie est caractérisée par un unique identifiant, associé à une clé privée. Videomark supportent 5 niveaux d'insertion, qui sont associés à la force du marquage. Les marques utilisées ne peuvent être extraites, sans la connaissance de la clé privée correspondante.

SysCop

Syscop est une boîte à outils qui permet d'insérer une marque invisible dans un medium. Elle a été développée par MediaSec, une compagnie privée. Syscop propose une interface flexible pour insérer une marque dans un flux vidéo MPEG1 ou MPEG2.

2.5 Conclusion

En résumé, nous pouvons exposer les observations suivantes :
1) Les méthodes proposées couvrent un large champ de complexité, de la plus faible à la plus considérable, incluant les transformées en ondelettes et les modèles HVS. En général, les méthodes les plus complexes semblent insérer des marques avec un niveau de robustesse plus élevé.
2) La plupart des méthodes s'appliquent dans un domaine non compressé. Seules, quelques techniques insèrent directement leur marque dans la vidéo compressée. Pour le marquage dans l'espace compressé, les marques sont insérées dans les coefficients DCT, les vecteurs de mouvements ou dans la structure GOP.
3) Le coût du marquage représente un coût variant de quelques centaines de bits par seconde à quelques bits, pour une vidéo de résolution équivalente au format télévision.
4) Il existe deux types de marquage : les méthodes additives, qui ajoutent une marque au modèle et les méthodes substitutives, qui remplacent des composantes du modèle par une marque.
A ce jour, très peu d'approches prennent en compte l'aspect temporel de la vidéo, la plupart des techniques utilisées étant importées de l'image fixe. Toutefois, certains auteurs exploitent les particularités de la vidéo (transformée 3D avec un axe temporel, vecteur de mouvement ...).

Notre but est de développer un algorithme de tatouage vidéo robuste aux attaques les plus communes, et essentiellement à différents types de codage. Dans les chapitres suivants, nous allons exposer notre algorithme de tatouage basé sur l'insertion d'une marque dans les vecteurs de mouvement, ainsi que la méthode d'embrouillage utilisant le même support que notre algorithme de tatouage.

Chapitre 3

Embrouillage et tatouage : solution globale de protection

3.1 Introduction

Dans ce chapitre, nous allons commencer par présenter une solution d'embrouillage. Ensuite, nous proposons une solution de tatouage, dont le but est de résister à différents types de codage. La combinaison de ces deux approches travaillant sur le même support, les vecteurs de mouvement, nous permet d'élaborer une solution globale de protection des contenus vidéo.

La solution de tatouage vidéo que nous avons développée se base sur l'insertion invisible d'une marque dans l'espace engendré par les vecteurs obtenus par un estimateur de mouvement. Pour ce faire, nous utiliserons un algorithme de "block-matching" classique décrit en section 3.2.
La problématique de développer un système de tatouage réside essentiellement dans la définition d'une règle d'insertion, qui autorise un déplacement du support de la marque dans un voisinage de son origine, tout en conservant l'information du tatouage. Nous allons donc chercher à définir une sphère, afin de pouvoir anticiper les légères variations que pourraient subir les vecteurs de mouvement après une attaque (c.f. figure 23).

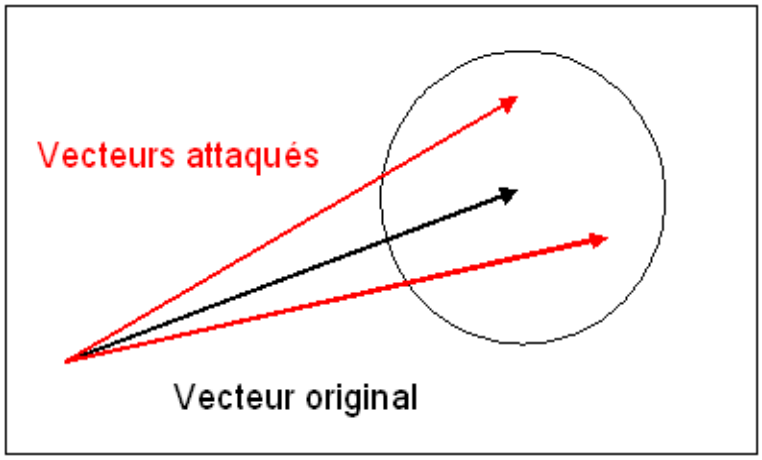


FIG. 23 – Illustration de la problématique du tatouage

La section 3.3 décrira quant à elle les bases de notre approche de tatouage, dont les différentes améliorations mises en oeuvre seront proposées ultérieurement.

3.2 Méthode d'embrouillage basé sur des techniques de tatouage

Les algorithmes de tatouage sont apparus dans le but de proposer une protection a posteriori des contenus multimedia. Ces algorithmes décrivent des méthodes permettant d'insérer des informations, invisibles et robustes, dans des données numériques. Nous proposons ici une nouvelle application du tatouage dans un contexte d'embrouillage. Nous avons développé une technique

que nous appelons "waterscrambling", dans laquelle une vidéo est embrouillée efficacement en perturbant un sous-ensemble de vecteurs de mouvement, extraits de l'analyse vidéo. L'intérêt de cette approche repose sur la capacité d'embrouiller une vidéo, tout en maîtrisant le degré de visibilité du contenu original. En effet, par cette technique, nous pouvons maîtriser la dégradation de la vidéo. De plus, en utilisant des techniques de tatouage, nous pourrons hybrider notre approche d'embrouillage, avec une approche de tatouage classique, afin de réaliser une protection globale, à la fois a priori et a posteriori du médium.

3.2.1 Introduction

Jusqu'à présent, les systèmes de tatouage étaient développés dans le but de protéger un médium en y insérant, par exemple, un copyright. Notre approche est légèrement différente. En effet, nous utilisons des techniques inspirées du tatouage afin d'insérer une marque visible dans le médium, ce qui nous permet d'embrouiller son contenu. Le système, que nous avons conçu, se trouve à la frontière entre un système de contrôle d'accès, et un système de tatouage classique. Le contexte applicatif dans lequel notre travail se situe correspond à un service de mise à disposition de contenus vidéos sur l'Internet. Par cette approche, le fournisseur de contenus n'a pas besoin de posséder un système de sécurité pour stocker les vidéos, la protection étant intégrée dans celles-ci. En outre, nous proposons un système qui permet de contrôler la qualité de la vidéo embrouillée, afin d'assurer une certaine visibilité du contenu original.
Ce contrôle d'embrouillage est autorisé par la génération de petites perturbations, pouvant être pondérées, sur les vecteurs de mouvement. Il existe quelques travaux sur l'insertion de marques invisibles sur les vecteurs de mouvement [145] et [92]. Ces deux articles traitent d'un système de tatouage vidéo basé sur l'insertion d'une marque sur les vecteurs de mouvement. En conséquence, la marque insérée est invisible. Les auteurs ne s'intéressent pas à la dégradation perceptible de la vidéo et donc ne réalisent pas l'embrouillage de celle-ci. Ces deux articles seront développés au chapitre suivant. Les techniques utilisées ne sont pas réversibles, la règle employée n'étant pas bijective. Ces approches ne permettent donc pas la reconstruction de la vidéo originale. A contrario, un système d'embrouillage doit permettre la reconstruction de la video originale, il nous a donc fallu déterminer une nouvelle règle d'insertion de la marque, qui sera détaillée dans la section suivante. En outre, le système développé ne doit pas changer de façon significative la distribution statistiques des éléments servant de support à l'embrouillage, et la procédure de désembrouillage doit pouvoir s'exécuter en temps réel.

3.2.2 Procédure d'embrouillage

Dans ce contexte, les contraintes sont différentes des contraintes habituelles en tatouage. On ne se pose pas ici le problème de robustesse face à des attaques de type désynchronisation, ou dégradation de la marque, seules les attaques d'extraction de celle-ci sont à envisager du point de vue du tatouage. En effet, le système développé ici, a pour but de brouiller la vidéo. Donc, toutes les attaques auront pour but de reconstruire au mieux la vidéo originale, celles-ci ne devront pas, par conséquent, apporter une dégradation supplémentaire à la vidéo. Par exemple, il ne devra pas être possible à un utilisateur, ayant payé pour obtenir une clef $K1$ de décodage, de transmettre cette clef à un tiers qui pourra à son tour reconstruire la vidéo, sans avoir payé. La clef de décodage devra donc être constituée de 2 parties propres à chaque utilisateur, une partie publique, que l'utilisateur transmettra au fournisseur (afin de chiffrer le code permettant de désembrouiller la vidéo), et une partie privée stockée secrètement par l'utilisateur (qui permet de déchiffrer le code au niveau du décodeur). Dans cette architecture, on peut ainsi identifier d'autre type d'attaques, qui ne sont pas liées au schéma de marquage. En effet, un pirate aura différentes approches à considérer avant de pouvoir reconstruire la vidéo. Soit il essaie d'extraire la marque, par des techniques protocolaires et des techniques d'estimation, basées sur des connaissances a priori du schéma de marquage, soit il essaie de récupérer le code permettant de brouiller la vidéo. Pour cela, deux approches sont possibles ; une approche de cryptanalyse qui consistera à casser l'algorithme de chiffrement du code, soit une approche plus "informatique" qui consistera à identifier la technique utilisée pour stockée la clef privée d'un utilisateur. Ainsi, le pirate pourra effectuer des recherches exhaustives, en testant, par exemple, l'ensemble des codes possibles pour le brouillage ou l'ensemble des combinaisons de vecteurs de mouvement possibles. Enfin, une approche basée sur la corrélation entre des blocs voisins, pourrait éventuellement permettre de reconstruire une version moins dégradée de la vidéo originale.
Nous allons maintenant présenter l'algorithme d'embrouillage que nous avons développé. Celui-ci a pour but de répondre aux attentes d'un système tel que présenté sur la figure 24.

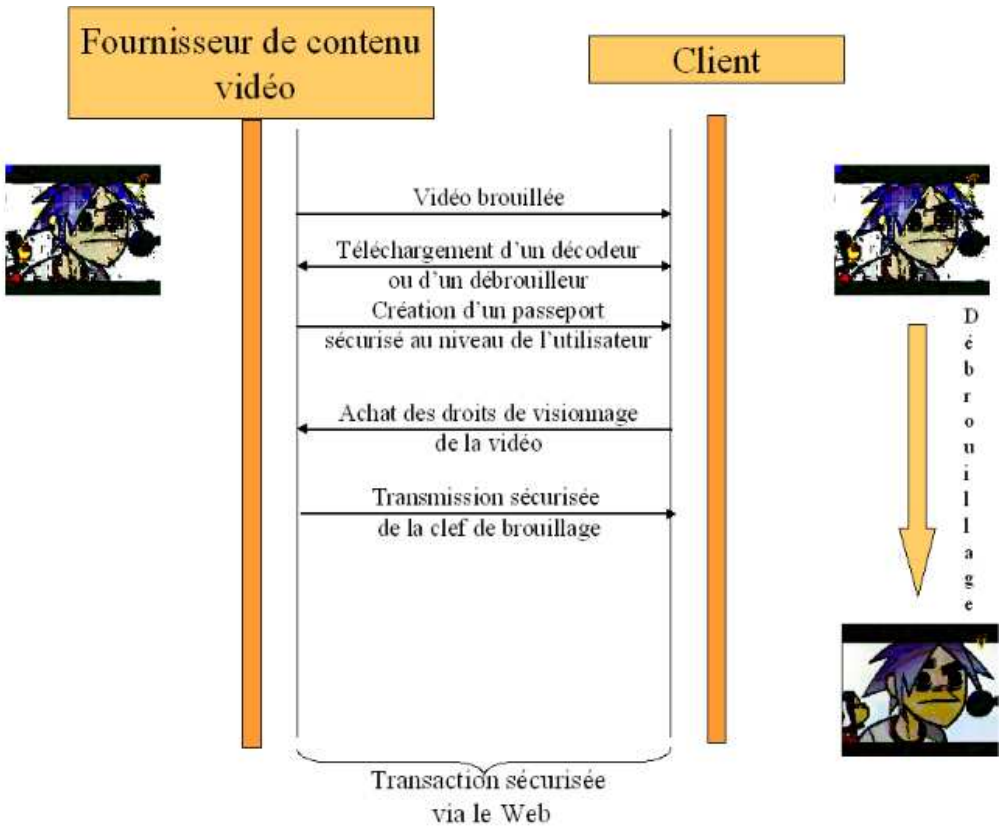


FIG. 24 – Scénario de contrôle d'accès

Au niveau de l'encodeur, le signal vidéo d'entré est compressé au format MPEG4. La première étape consiste donc à extraire les vecteurs de mouvement du flux vidéo qui seront perturbés pour embrouiller le signal. Deux approches sont possibles ; la première consiste à utiliser un analyseur syntaxique détectant et extrayant les vecteurs de mouvement. Dans ce cas, le module de "waters-crambling" est un module indépendant. La deuxième approche consiste à intervenir directement pendant la phase de compression. Dans ce cas, le module supplémentaire doit être compatible avec le module de compression.

Pour embrouiller la vidéo, nous générons une marque visible, définie par un vecteur binaire $W\epsilon\{1,-1\}^N$ (où N correspond à la taille de la marque), qui est ajoutée au sous-ensemble de vecteurs de mouvement. Afin d'accroître la robustesse, nous pouvons appliquer une permutation $\sigma_f(W)$ sur W pour chaque image f. Premièrement, comme présenté dans [81], et par analogie aux techniques d'étalement de spectre, la marque est diffusée sur de nombreuses fréquences de façon à ce que l'énergie ajoutée à chaque fréquence soit faible. Nous extrayons ensuite pour chaque image f correspondant à une image P ou B, l'ensemble de m_f vecteurs de mouvement appelé $V_f = \{\vec{v_f^i}, 1 \leq i \leq m_f\}$. Rappelons que les images P et B correspondent aux images prédites

et bi-directionnelle du schéma de compression. Ensuite, un sous-ensemble $\widetilde{V}_f$ ($\widetilde{V}_f \subset V_f$) de k_f vecteurs est utilisé pour insérer la marque $\sigma_f(W)$. La sélection d'un sous-ensemble nous permet de réguler les dégradations apportées par l'insertion de la marque visible. Cette insertion se fait selon la formule suivante :

$$\forall \vec{d_f} = (d_f^x, d_f^y)^T \in \widetilde{V}_f, \vec{d_f^w} = \vec{d_f} + \Phi(\alpha, \sigma_f(W), K_{\sigma_f(W)}) \tag{39}$$

où

- $\vec{d_f}$ correspond à un vecteur appartenant à $\widetilde{V}_f$;
- α correspond à la force de la marque, qui peut varier selon le contenu de la vidéo ;
- Φ correspond à une fonction qui dépend de W et de K, K est une clef secrète de "waters-crambling" qui peut être utilisée pour renforcer la sécurité.

Afin de déterminer l'ensemble $\widetilde{V}_f$ de vecteurs de mouvement, nous utilisons la clef de "waters-crambling" $K_{\sigma_f(W)}$ pour initialiser un générateur pseudo-aléatoire qui nous permet de déterminer les k_f indices k_f^i ($1 \leq i \leq m_f$) correspondant aux indices des vecteurs de mouvement de V_f :

$$\widetilde{V}_f = \{\vec{d_f^j}, j \in \{k_f^i\}_{1 \leq i \leq m_f}\}. \tag{40}$$

Pour améliorer la sécurité, la clef $K_{\sigma_f(W)}$ est modifiée pour chaque nouvelle vidéo à embrouiller. Cette clef représente l'état initial du générateur pseudo-aléatoire. Notre système de "waterscrambling" utilise cette clef de la même façon qu'un algorithme de chiffrement symétrique i.e. la clef doit être partagée entre le fournisseur de contenus et l'utilisateur, afin de permettre la réalisation de la procédure inverse de "waterscrambling". Par conséquent, un canal sécurisé doit être établi entre les deux protagonistes pour transmettre cette clef de manière sécurisée.

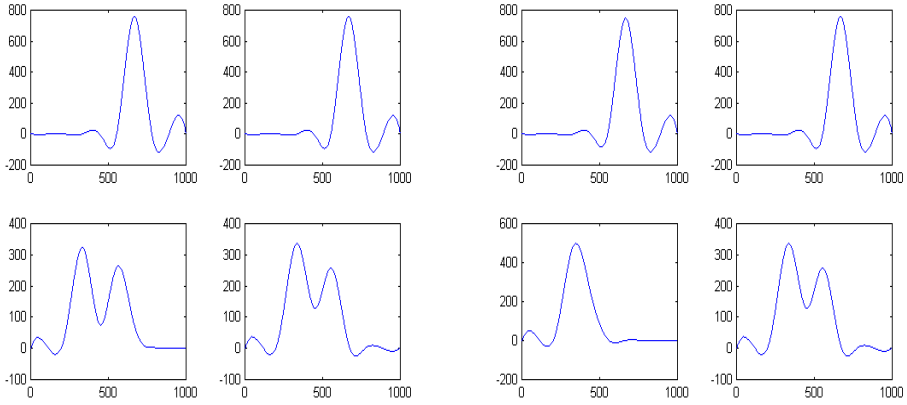
Le processus inverse de "waterscrambling" doit aussi connaître la force α utilisée par le fournisseur pour embrouiller la vidéo, la clef $K_{\sigma_f(W)}$ est donc composée de deux parties : les 7 premiers bits de la clef représentent la force α et les autres bits correspondent à la clef utilisée pour initialiser le générateur pseudo-aléatoire.

La règle d'insertion suivante est ensuite utilisée :

$$\forall \overrightarrow{d_f} = (d_f^x, d_f^y)^T \in \widetilde{V_f}, \overrightarrow{d_f^W} = \begin{cases} d_f^{W,x} = \begin{cases} d_f^x + \alpha \times \Upsilon(\sigma_f^i(W), K_{\sigma_f^i(W)}) & \text{si} \sigma_f^i(W) = +1 \\ d_f^x & \text{sinon} \end{cases} \\ d_f^{W,y} = \begin{cases} d_f^y + \alpha \times \Upsilon(\sigma_f^i(W), K_{\sigma_f^i(W)}) & \text{si} \sigma_f^i(W) = -1 \\ d_f^y & \text{sinon} \end{cases} \end{cases} \tag{41}$$

où $\sigma_f^i(W)$ correspond à la $i^{\text{ème}}$ composante du vecteur $\sigma_f(W)$, W étant le tatouage visible. Nous pouvons voir que W est diffusée dans l'image en insérant un bit par vecteur de mouvement sélectionné dans $\widetilde{V_f}$.

Enfin, pour étaler l'effet du "waterscrambling", nous pouvons insérer la marque visible dans un domaine transformé plutôt que dans le domaine spatial. Pour cela, nous réalisons deux DCT 1D, la première sur les composantes x et la seconde sur les composantes y d'un vecteur global V : $V = (V^x, V^y)^T \in \Re^{2k_f}$ avec $V^x = (d^x_{f_1}, d^x_{f_2}, ..., d^x_{f_{k_f}})$ et $V_y = (d^y_{f_1}, d^y_{f_2}, ..., d^y_{f_{k_f}})$.
En travaillant dans un domaine transformé (ici un domaine fréquentiel), nous pouvons contrôler l'énergie globale ajoutée aux vecteurs de mouvement, en ne perturbant par exemple que les moyennes fréquences ou les hautes fréquences. Nous pouvons également mieux contrôler la perturbation de la distribution statistique des vecteurs de mouvement, évitant ainsi une augmentation trop importante du taux de compression. Afin d'atteindre ce but, nous pouvons définir une fonction Υ, qui correspond à une déformation pseudo-homothétique de la distribution des vecteurs de mouvement. (cf. Figure 25). Les figure 25(a) et figure 25(b) montrent un exemple de la distribution des vecteurs de mouvement. La première colonne (respectivement la seconde) de ces figures illustre la distribution des vecteurs de mouvement embrouillés (respectivement originaux). Les colonnes montrent deux distributions : la première ligne (respectivement la seconde) correspond à la distribution de l'amplitude des vecteurs de mouvement suivant l'axe des abscisses (respectivement l'axe des ordonnés). Comme nous pouvons le voir, protéger une vidéo avec une force $\alpha = 20$ ne modifie pas significativement les distributions, ce qui permet de maintenir un taux de compression proche de celui obtenu sur la séquence originale, tout en dégradant suffisamment la qualité de la vidéo. De la même manière, pour une force $\alpha = 100$, la distribution de l'amplitude des vecteurs suivant l'axe des ordonnés est très affectée, et par conséquent, le taux de compression s'en trouve lui aussi dégradé. Bien que la plupart des codeurs vidéo expriment les vecteurs de mouvement par une approche différentielle, ces courbes montrent néanmoins que le coût de codage ne varie pas de façon significative. En effet, les vecteurs de mouvement sont légèrement modifiés par le schéma de "waterscrambling", et ainsi restent dans une zone restreinte. Par conséquent, en gardant approximativement la même distribution des vecteurs de mouvement, nous nous assurons que le coût de codage restera proche de celui de la vidéo originale compressée. Dans le cas où le taux de compression doit rester le même, nous devons choisir la fonction Υ adéquate. La figure 26 présente les variations du taux de compression suivant la force de "waterscrambling" utilisée. Comme nous l'avons déjà mentionné, nous pouvons noter qu'une force $\alpha = 100$ augmente le taux de compression d'environ 35%. Cependant, une force $\alpha = 20$ suffit en général à dégrader la qualité de la vidéo, tout en gardant une visibilité du contenu original. Dans ce cas, l'augmentation du taux de compression est de seulement 10%, ce qui reste acceptable.



(a) (b)

FIG. 25 – Modification de la distribution des vecteurs de mouvement après l'application du waterscrambling : (a) waterscrambling avec une force $\alpha = 20$ et (b) waterscrambling avec une force $\alpha = 100$ (1) (resp. 2) correspond à la distribution des vecteurs de mouvement embrouillés suivant l'axe des abscisses (resp. correspond à la distribution des vecteurs de mouvement originaux suivant l'axe des abscisses) (3) (resp. 4) correspond à la distribution des vecteurs de mouvement embrouillés suivant l'axe des ordonnés (resp. correspond à la distribution des vecteurs de mouvement originaux suivant l'axe des ordonnés)

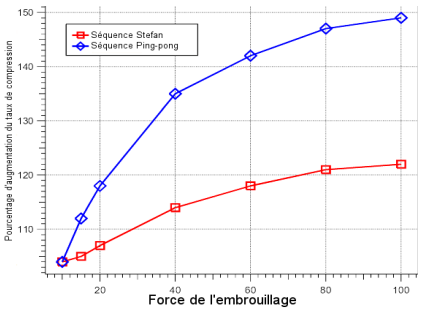


FIG. 26 – Variation du taux de compression en fonction de la force d'embrouillage utilisée sur les séquences Stephan et Ping-pong

Une fois la marque visible insérée, une approche de tatouage classique peut être utilisée. La combinaison d'une approche d'embrouillage et de tatouage a été précédemment présentée dans [68]. Les auteurs proposent deux alternatives. La première est d'insérer le tatouage avant d'appliquer la procédure d'embrouillage. De cette façon, lors de la phase de décodage, seule la protection apportée par l'embrouillage est supprimée, la marque invisible quant à elle, reste de façon permanente dans le contenu ainsi protégé. La deuxième alternative consiste à n'insérer le tatouage qu'au moment du décodage, au niveau du client. Ainsi, les deux protections n'ont pas à cohabiter et les perturbations résultantes de la présence de ces deux protections ne sont pas à prendre en compte. Dans notre approche, les deux cas peuvent être envisagés. En effet, nous pouvons ajouter une marque invisible W' sur les mêmes vecteurs de mouvement sélectionnés pour l'embrouillage. La procédure de tatouage peut être exécutée dans le domaine compressé, ou dans le domaine non-compressé. Dans notre approche, que nous présenterons au chapitre 3 et 4, nous avons choisis de travailler dans le domaine non-compressé, afin d'éviter de prendre en compte les effets de dérive qui peuvent survenir dans le domaine compressé. Ainsi, les procédures de "waterscrambling" et de tatouage sont appliquées dans un même système, mais de façon différente.

3.2.3 Procédure de désembrouillage

Notre procédure de "waterscrambling" inverse est incluse dans un schéma de décompression MPEG, cependant, tout comme pour la procédure de "waterscrambling", nous pouvons utiliser un analyseur syntaxique afin d'avoir un module de "waterscrambling" inverse indépendant. Le "waterscrambling" inverse consiste à extraire les vecteurs de mouvement embrouillés, et à les traiter, afin de reconstruire la vidéo originale. Pour réaliser la reconstruction, il est nécessaire de récupérer la clef $K_{\sigma_f(W)}$ afin d'initialiser le générateur pseudo-aléatoire, et la force α pour régler le "de-waterscrambling". Ainsi, nous pouvons extraire la marque visible de chaque image f en

appliquant la fonction inverse de Φ avec :

$$\forall \vec{d_f} \in \widetilde{V}_f, \vec{d_f} = \vec{d_f^W} + \Phi^{-1}(\alpha, \sigma_f(W), K_{\sigma_f(W)}). \tag{42}$$

3.2.4 Résultats expérimentaux

Le système de "waterscrambling" a été testé sur différentes vidéos. Dans cette section, nous montrons les résultats obtenus sur la séquence "Stefan" et "Ping-pong". Les figures 27 et 28, montrent l'évolution de la dégradation en fonction de l'amplitude de la force α. La première ligne montre quelques images de la séquence originale ($\alpha = 0$), la seconde ligne montre des images embrouillées avec $\alpha = 40$, et la dernière ligne montre les résultats avec $\alpha = 60$. Comme nous pouvons le voir, la force α permet de régler le degré de visibilité de la vidéo. Plus la force α est importante, plus la vidéo est dégradée.

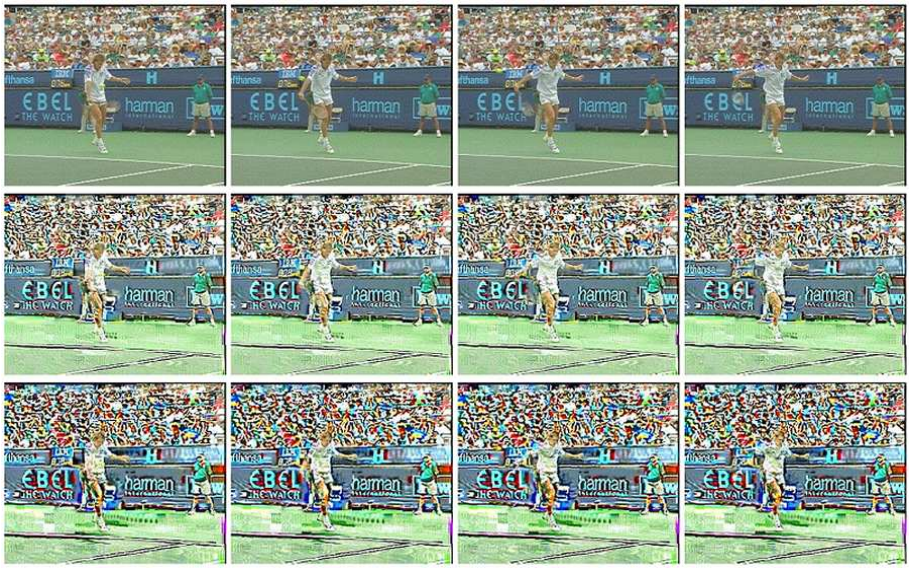


FIG. 27 – Résultats du Waterscrambling sur la séquence Stefan avec les forces suivantes α . La première ligne représente la séquence originale($\alpha = 0$), la seconde ligne représente les résultats avec $\alpha = 40$ et la dernière ligne utilise $\alpha = 60$

FIG. 28 – Résultats du Waterscrambling sur la séquence ping-pong avec les forces suivantes α . La première ligne représente la séquence originale($\alpha = 0$), la seconde ligne représente les résultats avec $\alpha = 40$ et la dernière ligne utilise $\alpha = 60$

Nous avons également testé la robustesse de cette approche vis-à-vis d'une attaque dont le but est d'améliorer la qualité de la vidéo embrouillée, sans avoir connaissance de la clef utilisée dans le système présenté ci-dessus.
Cette attaque consiste à réaliser une recherche exhaustive globale des variations appliquées sur les vecteurs de mouvement, afin de tenter une restitution de la vidéo originale. Pour tester ces variations, nous utilisons le même algorithme servant à la procédure d'embrouillage. Nous utilisons donc des DCT 1D sur les composantes des vecteurs de mouvement.
Deux approches peuvent être testées :

1. à partir des blocs prédits, chacun composés d'un vecteur de mouvement embrouillé, et d'un bloc d'erreur (calculé à partir du vecteur de mouvement original), on peut essayer de corriger les vecteurs de mouvement embrouillés, en retrouvant un bloc intra dans l'image intra correspondante se rapportant au bloc prédit de l'image que l'on veut reconstruire. Pour ce faire, nous projetons le bloc prédit dans l'image intra correspondante, et nous examinons dans une fenêtre de recherche prédéfinie un bloc intra approprié qui maximise le PSNR avec le bloc prédit courant ;
2. à partir des blocs intra localisés dans les images P ou B (qui ne sont pas embrouillés), on peut essayer de retrouver le bloc original associé au vecteur de mouvement dans un voisinage du bloc prédit, en augmentant la corrélation entre les blocs intra non embrouillés et le bloc

embrouillé prédit.

L'implémentation de ces deux attaques nous a amenés aux conclusions suivantes :

- la première attaque génère de nombreux artefacts sur les frontières des blocs (i.e. des effets de bloc). Ces artefacts peuvent être expliqués par le fait qu'au niveau de l'attaque, seuls les blocs d'erreurs sont disponibles pour retrouver le bloc intra original, correspondant au bloc prédit. Ainsi, on peut seulement retrouver une approximation du bloc original et une approximation du vecteur de mouvement original. Ces erreurs d'approximation causent des artefacts fortement visibles, qui s'amplifient dans le temps, en raison de l'effet de dérive ;
- la seconde attaque possède deux inconvénients. Le premier peut s'expliquer par le fait qu' il y a, en général, peu de blocs intra dans une image prédite. Par conséquent, nous sommes seulement capables de corriger de manière approximative les vecteurs de mouvement des blocs prédits, localisés dans le voisinage immédiat des blocs intra, mais nous ne sommes pas capable d'assurer la correction des vecteurs de mouvement lorsque que l'on s'éloigne du bloc intra initial. Nous pouvons expliquer ceci par l'erreur d'approximation qui augmente dans le domaine spatial à chaque fois que l'on considère un nouveau bloc prédit à traiter. Le deuxième inconvénient correspond à l'accumulation de la même erreur d'approximation sur l'axe temporel, générant ainsi un effet de dérive.

Lors de nos expériences, la première attaque a mieux reconstruit la vidéo que la deuxième. Afin de donner un aperçu de l'impact visuel de cette attaque, nous avons tracé sur la figure 29 la courbe de PSNR des différentes vidéos. Cette figure montre le PSNR de la vidéo compressée, de la vidéo embrouillée et non attaquée et de la vidéo embrouillée puis attaquée. Comme nous pouvons le voir, l'attaque permet d'augmenter légèrement le PSNR qui reste néanmoins très nettement inférieur au PSNR de la séquence originale compressée prouvant ainsi l'innefficacité de l'attaque.

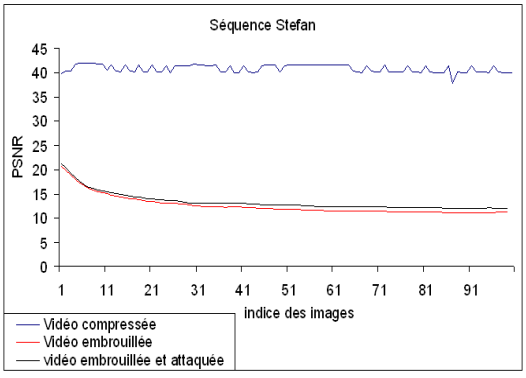


FIG. 29 – PSNR des différentes vidéo afin de mettre en évidence le gain obtenu par l'attaque testée

Afin d'être plus robuste à ce type d'attaque, nous pouvons crypter les images intra et ensuite

stocker la clef de façon sécurisée avec le décodeur. Pour lire la vidéo, les images intra peuvent être décryptées au niveau du décodeur, mais ne doivent pas être montrées à l'utilisateur. De cette façon, il n'y a plus d'image de référence pouvant servir de support à une telle attaque.

3.3 Algorithme de ”block-matching”

Notre algorithme se base sur l'insertion d'une marque dans l'espace engendré par les vecteurs de mouvement issues d'une vidéo. La première étape consiste à obtenir un sous-ensemble de ces vecteurs, qui serviront au tatouage. Pour ce faire, plusieurs approches sont envisageables, puisqu'il existe différents types d'estimateurs de mouvement, dont nous allons donner un bref aperçu dans la section suivante.
L'estimateur de mouvement utilisé dans cette thèse, est un algorithme de ”block-matching”. Celui-ci étant utilisé dans les normes MPEGx, nous pouvons espérer, en perspective, marquer un flux compressé issu de ces normes, même si la transposition de notre schéma à ce domaine nécessitera quelques adaptations.
Cependant, afin de faciliter la lecture de ce document, nous nous proposons d'effectuer quelques rappels de base nécessaire à la compréhension des estimateurs de mouvement utilisés dans les vidéos.

3.3.1 Estimation et compensation de mouvement pour la prédiction temporelle

La prédiction temporelle constitue le fondement de la réduction de débit dans la problématique du codage vidéo. Ce domaine reste très ouvert, et même si des techniques ont été utilisées dans un cadre normatif comme MPEGx ou H26x, il existe des méthodes alternatives qui pourraient sans doute émerger dans l'avenir. Notamment les codeurs à base d'ondelettes 2D+t, ou ceux à bases d'ondelettes de deuxième générations, ou encore les codeurs à base de bandelettes.
L'ensemble de ces approches repose sur des concepts de base issus de la cinématique.

Concepts de base de l'estimation du mouvement cinématique

La base de nombreuses techniques d'estimation de mouvement est la **contrainte du flot optique**. Elle suppose que l'intensité d'un objet reste constante, ou varie de façon prédictible le long de sa trajectoire au cours du temps. Cette contrainte forte étant posée, on peut donc exprimer au point $p(x, y)$ **l'équation du flot optique (EFO)** :

$$\frac{dI(p,t)}{dt} = 0$$

$$\frac{\delta I(p,t)}{\delta x}\frac{dx}{dt} + \frac{\delta I(p,t)}{\delta y}\frac{dy}{dt} + \frac{\delta I(p,t)}{\delta t}\frac{dt}{dt} = 0$$

En parallèle à ce modèle, un autre outil, correspondant à la **différence d'image déplacée** est fréquemment employé afin de déterminer le mouvement entre deux images. Plus connu sous le terme DFD (Displaced Frame Difference), celle-ci modélise la variation de l'intensité entre un pixel p d'une image I_t, et son correspondant par un mouvement $\overrightarrow{d}$ dans l'image $I_{t+\delta t}$:

$$DFD(p, \overrightarrow{d}) = I(p - \overrightarrow{d}(p,t,\delta t), t+\delta t) - I(p,t)$$

Bien que la résolution de l'EFO en chaque pixel soit possible, la DFD est un outil très puissant dans le contexte discret de l'image. En effet, on peut montrer que sous certaines conditions, le respect de la contrainte du flot optique est équivalent à l'annulation de la DFD en tous points du domaine.
Pour résoudre le problème de l'estimation de mouvement, on peut, quelle que soit la méthode, appliquer deux stratégies : une estimation avant ou une estimation arrière. Ce choix dépend en fait du modèle de mouvement ou encore de la méthode choisie.

- Une estimation **Avant** (cf. figure 30) modélise le mouvement entre I_t et $I_{t+\delta t}$, par une déformation de l'image I_t. On attribue ainsi un vecteur de mouvement à chaque pixel de l'image I_t.
- Une estimation **Arrière** (cf. figure 30) procède de manière inverse, et modélise le mouvement entre $I_{t+\delta t}$ et I_t. Elle attribue ainsi un vecteur de mouvement à chaque pixel de $I_{t+\delta t}$. La différence principale vis à vis de l'estimation de type **avant**, est qu'elle permet une compensation "complète" de $I_{t+\delta t}$. Toutefois, si des objets apparaissent sur $I_{t+\delta t}$ (encore appelée zone de découvrement), les vecteurs de mouvement associés aux pixels seront dénués de sens physique. Cependant, la plupart des codeurs vidéo utilisent cette approche, puisque dans le cadre d'un schéma de codage, "la prédiction de l'intégralité de l'image est plus importante que l'obtention d'un mouvement réaliste".

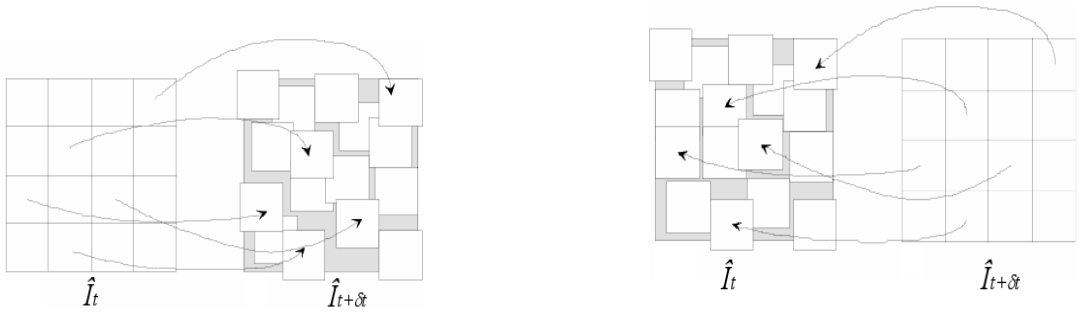


FIG. 30 – Différence entre une estimation-compensation vers l’avant et une estimation-compensation vers l’arrière dans le cas des méthodes basées blocs. On note que la reconstruction de I_t est ”complète” dans la cas arrière.

Dans notre approche, la méthode utilisée est une prédiction de type **Avant**, afin d’éviter de gérer les problèmes dûs aux zones de découvrement.

Les techniques utilisées pour l’estimation de mouvement dans les séquences vidéos peuvent être classées de 2 manières différentes :
En quatre familles selon le mode de description ou de représentation du champ de mouvement :
– Les méthodes de calcul d’un champ dense de mouvement, où un vecteur est attribué à chaque pixel de l’image ;
– Les méthodes calculant un vecteur de déplacement par bloc ;
– Les méthodes visant au calcul d’un mouvement complexe de type polynomial, pour différentes régions de l’image ;
– Les méthodes permettant le calcul d’un champ, défini par un modèle d’éléments finis.
Ou alors, en trois familles, selon la méthode de calcul des paramètres de mouvement :
– Les méthodes d’appariement ou de ”matching” qui recherchent localement les meilleurs paramètres au sens d’un critère donné ;
– Les méthodes par transformée, qui transforment les déplacements en un déphasage dans l’espace de Fourier ;
– Les méthodes différentielles, qui optimisent les paramètres du modèle de mouvement en cherchant à minimiser un critère (une erreur quadratique). Ces méthodes supposent que le critère à optimiser dispose de certaines propriétés mathématiques (par exemple, la dérivabilité).

Les méthodes énumératives et locales de mise en correspondance

Les méthodes d’estimation et de compensation de mouvement basées blocs sont les plus répandues dans les codeurs vidéo actuels, comme MPEGx et H26x. Elles sont également utilisées dans des applications d’interpolation temporelle de trames.

Ces méthodes s'appuient sur un partitionnement de l'image en blocs, et supposent que chaque bloc se déplace selon une translation. De cette manière, tous les pixels d'un bloc se voient attribuer le même vecteur de mouvement. Celui-ci permet, dans le cas d'une estimation arrière, de maximiser la ressemblance du contenu du bloc à l'instant $t + \delta t$, avec le contenu de son correspondant à l'instant t. La précision de ces méthodes peut-être pixelique ou subpixelique. Cependant, il faut noter que si un mouvement pixelique permet une compensation par simple recopie du contenu d'un bloc à l'instant t dans un bloc à l'instant $t + \delta t$, la compensation par un mouvement subpixelique nécessite l'interpolation de chaque valeur de colorimétrie du bloc.

Nous allons maintenant décrire plus en détail l'algorithme de **Block-Matching**.

Algorithme Block-Matching

Le BMA est généralement utilisé en estimation arrière. Pour ce faire, un partitionnement régulier en bloc de l'image $I_{t+\delta t}$ est réalisé. Afin d'estimer le vecteur de mouvement d'un bloc B entre $I_{t+\delta t}$ et I_t, le BMA utilise une fenêtre de recherche F. Celle-ci est placée dans l'image I_t, et centrée sur le bloc B, déplacé par un vecteur nul. On recherche alors le bloc B' de F, dont les valeurs minimisent l'erreur de ressemblance avec B.
Le principal facteur qui permet d'expliquer le quasi monopole de ces méthodes, est qu'il s'agit d'approches locales, à faible complexité, l'utilisation de blocs permettant une intégration simple avec les algorithmes de codage d'images de type JPEG.
Ce type d'estimation se décompose en deux phases : **La recherche (Searching) et l'évaluation (Matching)**. De nombreux algorithmes proposent des méthodes afin d'optimiser l'étape de recherche comme [40, 36]. Cependant, la méthode la plus efficace dans un schéma de codage est le "Diamond Search" détaillé dans [164]. Il conduit souvent à l'estimation d'un mouvement de plus faible amplitude, et donc plus facile à coder. L'étape d'évaluation, quant à elle, consiste à déterminer un critère représentant une erreur de reconstruction. Ce critère peut-être le MSE (Mean of Square Error), le MAD (Mean of Absolute Difference), ou la SAD (Sum of Absolute Difference). La SAD est souvent retenue pour sa faible complexité.
Pour plus de détails sur l'estimation de mouvement en générale, le lecteur intéressé pourra se référer aux travaux d'A. Buisson [7].

3.4 Algorithme de tatouage : première étape

L'algorithme que nous allons décrire ici est, comme nous l'avons annoncé, un système de tatouage vidéo basé sur le marquage des vecteurs de mouvement. Comme nous avons pu le voir dans l'état de l'art, il existe à ce jour très peu d'algorithmes utilisant la même approche. Rappelons que dans cette thèse, nous nous sommes placés dans le domaine non-compressé. L'approche proposée pourrait éventuellement être adaptée dans le domaine compressé. Cependant, ce domaine apporte des difficultés et des contraintes supplémentaires, qu'il faudrait étudier plus précisément, afin de s'assurer de la robustesse et de l'invisibilité de l'algorithme. Nous allons maintenant étudier en détail l'algorithme que nous avons développé.

3.4.1 Sélection déterministe ou pseudo-aléatoire

Notre procédure de tatouage consistant à marquer un sous-ensemble de vecteurs de mouvement déterminés par le "BMA", nous devons au préalable définir une procédure de sélection. Elle peut se faire soit de façon déterministe, soit de façon pseudo-aléatoire (c.f. figure 31). Afin de choisir judicieusement notre procédure de sélection, nous avons étudié l'impact du tatouage sur le PSNR des vidéos (c.f. figure 32). Or, comme nous pouvons le constater, pour les deux types de sélection, les variations du PSNR ne sont pas significatives. Cependant, les effets de clignotement temporel étant plus importants dans l'approche déterministe, nous préférons réaliser une sélection pseudo-aléatoire, en prenant garde à la plus grande sensibilité de cette sélection vis à vis des attaques protocolaires. En effet, dans ce cas, les blocs marqués possèdent alors une référence non marquée dans les images voisines. Ce qui peut permettre d'élaborer une attaque ciblée consistant à reconstruire les blocs marqués à partir des références dans les images voisines. Nous reviendrons sur cette attaque dans le chapitre 5.

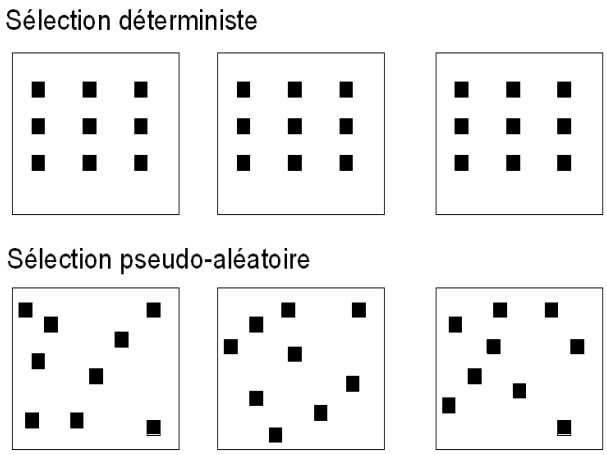


FIG. 31 – Sélection déterministe ou pseudo-aléatoire des blocs pour l'insertion de la marque

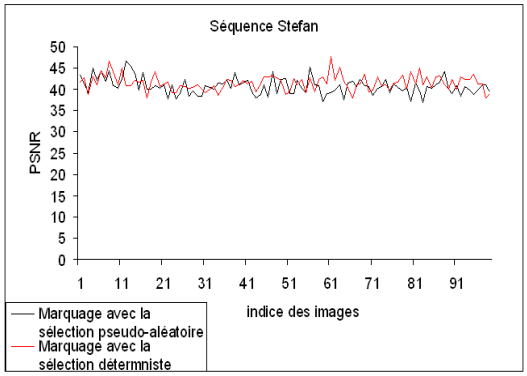


FIG. 32 – PSNR des vidéos marquées pour les deux types de sélection

Le "BMA" nous donne un ensemble V_f de N vecteurs de mouvement. De par la procédure de sélection S choisie, nous n'exploiterons qu'un sous-ensemble $\tilde{V}_f$ de k vecteurs de mouvement. Les vecteurs de mouvements de ce sous-ensemble seront notés d_f avec $\tilde{V}_f = d_f \epsilon V_f, S(d_f) = \delta$.

3.4.2 Règle d'insertion

Après avoir sélectionné pseudo aléatoirement les k vecteurs d_f qui nous serviront de support au tatouage, nous pouvons appliquer la règle d'insertion définie par :

$$\forall \vec{d_f} = (d_f^x, d_f^y)^T \in \widetilde{V}_f, \vec{d_f^W} = \widetilde{\Phi}(\vec{d_f}, \sigma_f(W), K_{\sigma_f(W)}) \tag{43}$$

où :

- $\widetilde{\Phi}$ est une fonction non inversible, définie dans la suite de cette section ;
- $\sigma_f(W)$ représente une permutation de la marque permettant d'augmenter la robustesse de notre algorithme ;
- $K_{\sigma_f(W)}$ est une clef générée à partir de la marque à insérer et de l'indice de l'image en cours. Cette clef nous permet d'initialiser le générateur pseudo aléatoire utilisé pour la sélection décrite ci-dessus ;
- f est l'indice de l'image en cours.

Afin d'augmenter la robustesse de cette approche, la règle d'insertion mise en oeuvre respecte une structure spatiale basée sur la construction d'une grille de référence $\mathcal{G}$, comme illustré sur la figure 33. Cette grille rectangulaire est générée dans le domaine cartésien et est associée au référentiel $(O, \overrightarrow{i}, \overrightarrow{j})$. Cela représente une partition en bloc de l'espace d'insertion, indirectement lié au support compact de l'image. Il en résulte un ensemble de blocs E, chaque bloc étant de taille $H \times K$ (dans le cas présenté ici, $H = K = 7$). Dans la suite, les points de référence seront notés R_i, et ceux-ci correspondent à l'intersection entre chaque bloc.

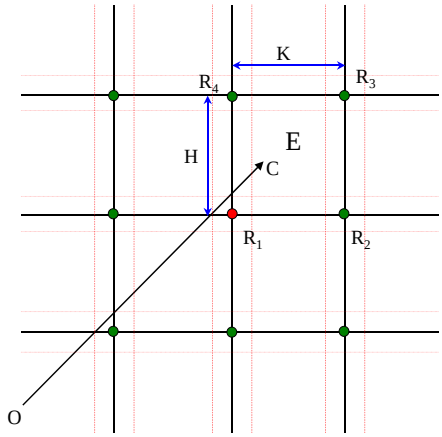


FIG. 33 – Construction d'une grille de référence afin d'insérer une marque sur les vecteurs de mouvement

Chaque vecteur de mouvement appartenant à $\widetilde{V}_f$ est tout d'abord projeté sur la grille $\mathcal{G}$. Cette phase permet de déterminer le point de référence rattaché à ce vecteur. Cette grille, de par sa structure, peut être vue comme une quantification de l'espace d'insertion, qui nous permet ainsi de définir une "sphère". Celle-ci permettra d'anticiper les légères variations sur les vecteurs de mouvement, sans pour autant perdre l'information de tatouage. Dans un premier temps, nous utilisons une grille carrée dont la géométrie est présentée sur la figure suivante. La figure 33 illustre

ce processus : l'extrémité de ce vecteur projeté $\vec{OC}$ appartient à un bloc E de $\mathcal{G}$. On peut donc en déduire trivialement les 4 points d'intersection R_1, R_2, R_3 et R_4. Le point de référence du vecteur de mouvement est celui qui est le plus proche de l'extrémité de ce vecteur, en accord avec la distance L^2. Dans l'exemple de la figure 33, le point de référence de $\overrightarrow{d_f}$ est R_1.
Ensuite, afin d'insérer la marque, le vecteur de mouvement est modifié (c.f. Figure 34), en construisant dans chaque bloc E un élément rectangulaire de taille $h \times k$ (zone Z_1), où $h = H - \delta_1$ et $k = K - \delta_2$. δ_1 et δ_2 sont choisis tels que les deux zones Z_1 et Z_2 couvrent la même surface et que $Z_1 \cup Z_2 = E$ (dans le cas présenté ici, δ_1 = δ_2 = 5). Ces deux zones Z_1 et Z_2 supportent la règle d'insertion : Z_1 est associée au bit -1 et Z_2 au bit $+1$.

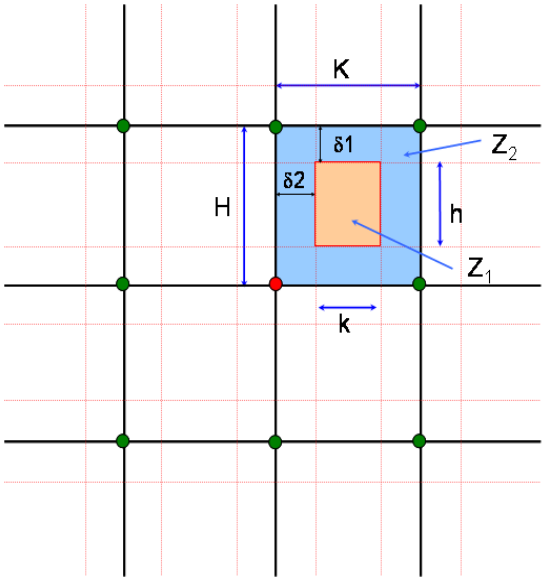


FIG. 34 – Partition en blocs pour insérer la marque

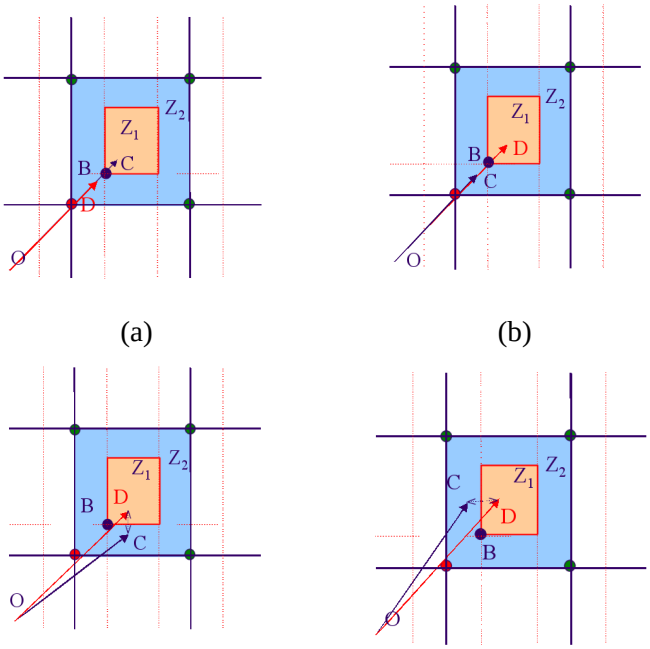


FIG. 35 – Calcul des vecteurs marqués

Ensuite, si nous considérons que $\vec{d_f} = \vec{OC}$ correspond au vecteur à marquer, et que W_i est le bit à insérer, le vecteur marqué $\vec{d_f^W}$ étant calculé de la façon suivante :

- si $W_i = -1$ et si $\vec{d_f}$ est dans la zone correcte (i.e. dans la zoneZ_1), alors $\vec{d_f^W} = \vec{d_f}$; sinon, une symétrie centrale de centre B doit être appliquée. On obtient alors le vecteur $\vec{d_f^W} = \overrightarrow{OD}$ (cf. Figure 35(a)) ;
- si $W_i = +1$ et si $\vec{d_f}$ est dans la zone correcte (i.e. dans la zone Z_2), alors $\vec{d_f^W} = \vec{d_f}$; sinon, comme la zone Z_2 n'est pas compacte, trois possibilités se présentent pour calculer le vecteur $\vec{d_f^W}$:
  - $\vec{d_f^W}$ est donné par une symétrie centrale de centre B (cf. Figure 35(b)) ;
  - $\vec{d_f^W}$ est donné par une symétrie axiale parallèle à l'axe y, et passant par B (cf. Figure 35(c)) ;
  - $\vec{d_f^W}$ est donné par une symétrie axiale parallèle à l'axe x, et passant par B (cf. Figure 35(d)).

Le choix de la symétrie doit permettre de minimiser les distorsions de $\overrightarrow{d_f}$. En fait, après avoir calculé $d_x = C_x - B_x$ et $d_y = C_y - B_y$ (avec $B = (B_x, B_y)^T$ et $C = (C_x, C_y)^T$), les symétries sont choisies telles que :

- si $d_x \leq \delta_2$ et $d_y \leq \delta_1$, la symétrie centrale est appliquée ;

– si $d_x \leq \delta_2$ la symétrie axiale parallèle à l'axe x est appliquée ;
– si $d_y \leq \delta_1$, la symétrie axiale parallèle à l'axe y est appliquée.

Enfin, nous réalisons une compensation de mouvement avec les vecteurs marqués. Cette étape peut être réalisée sur l'ensemble des blocs constituant l'image, ou seulement sur les blocs marqués (le reste de l'image étant complété par les blocs originaux). La seconde approche nous permet d'éviter des artefacts pouvant être généré par l'estimateur de mouvement, et dans un même temps, de rendre notre approche plus robuste. La figure 36 présente le PSNR obtenu avec ces deux alternatives.

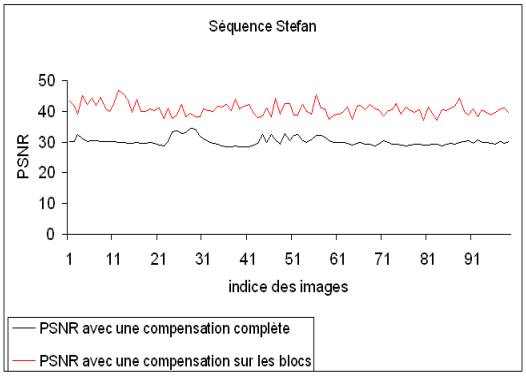


FIG. 36 – Comparaison des PSNR entre une compensation complète de l'image et une compensation réalisée seulement sur les blocs marqués

3.4.3 Mise en oeuvre de l'aspect hiérarchique, de la redondance et de l'étalement temporel

Dans cette section, nous nous intéressons à l'amélioration de la robustesse de notre algorithme. Pour ce faire, nous nous proposons d'élaborer différentes stratégies. La première consiste à appliquer une procédure hiérarchique. La deuxième à insérer une notion de redondance. Enfin la dernière consiste à réaliser un étalement temporel.
Afin de générer une redondance et d'augmenter la robustesse de notre système au niveau du support de la marque, nous pouvons appliquer une procédure hiérarchique sur les vecteurs de mouvement comme illustré sur la figure 37.

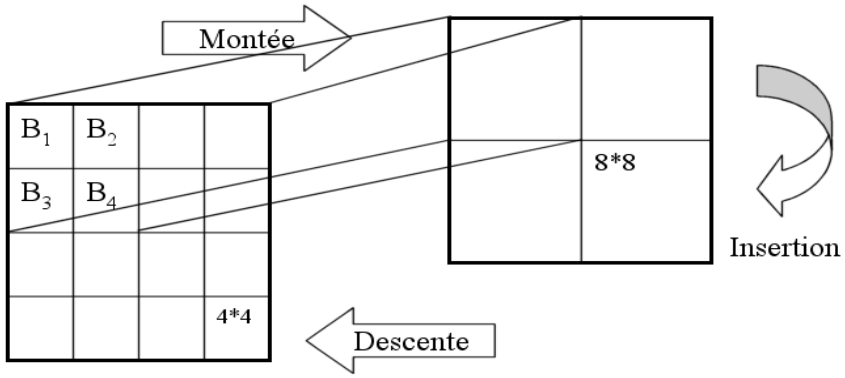


FIG. 37 – Approche hiérarchique : la procédure de ”montée” consiste à calculer par moyennage les vecteurs de mouvement pour le bloc 8×8. La procédure de descente consiste à répercuter les variations calculées en appliquant la règle d'insertion sur les vecteurs de mouvement des blocs 8×8 au niveau des vecteurs associés au blocs 4×4

Cette figure illustre le cas où nous réalisons cette procédure sur un niveau. Il est possible d'envisager d'appliquer cette méthode sur un nombre plus important de niveau, dans ce cas il est important d'étudier l'impact que peut avoir le marquage sur des blocs de taille trop importante. Une autre approche envisageable est de réaliser l'estimation de mouvement sur des niveaux sub-pixelliques, et d'appliquer ensuite cette procédure sur deux ou trois niveaux, ainsi la redondance créée serait plus importante et par conséquent l'algorithme de tatouage serait plus robuste. Dans la suite, pour plus de clarté, nous décrirons cette procédure pour deux niveaux.
Comme décrit en section 1.2, nous calculons une estimation de mouvement entre deux images sur des blocs de taille $N \times N$ (soient les B_j de la figure 37), et chacun de ces B_j sont associés aux vecteurs de mouvement ${d_{f,j}}^i$. Puis on initialise le calcul de la pyramide par les vecteurs issus du ”block-matching”. Les vecteurs de mouvement du niveau supérieur (notés ${D_f}^i$), sont obtenus par simple calcul de la moyenne des vecteurs ${d_{f,1}}^i, {d_{f,2}}^i, {d_{f,3}}^i$ et ${d_{f,4}}^i$ associés aux 4 blocs concaténés B_1, B_2, B_3, B_4 comme décrit ci-dessous :

$$ {D_f}^i = \frac{1}{4} \sum_{j=1}^{4} {d_{f,j}}^i \tag{44} $$

où :

- j représente l'indice du bloc à marquer du niveau supérieur ;
- i représente l'indice du bloc du niveau supérieur en cours de traitement ;

– f représente l'indice de l'image en cours.

Une fois les vecteurs du niveau supérieur déterminés par ce procédé (phase de remontée), on applique la règle d'insertion sur ces derniers. Les modifications réalisées sur le niveau le plus élevé seront ensuite répercutées sur le niveau le plus bas (phase de redescente). Cette approche nous permet de créer un signal de marquage redondant lors de la phase d'insertion, le schéma de tatouage est par conséquent, plus robuste.

Il existe d'autres approches pour créer un signal redondant. L'approche la plus intuitive est de répéter tout simplement un certain nombre de fois le bit à insérer. Pour ce faire, nous utilisons un code par répétition/accumulation, qui nous permet d'avoir un gain en robustesse contre d'éventuelles déformations locales, comme pourrait produire une attaque du type "Stirmark" (rappelons que cette attaque n'est pas applicable telle quelle sur une vidéo). Lors de la détection, il suffira de déterminer le bit majoritaire dans chaque sous séquence de la marque. La forme de la marque à insérer est alors différente, comme présenté sur la figure 38. Cette technique permet de contrer d'éventuelles attaques locales de faibles envergures qui provoqueraient la mauvaise détection d'un bit. Enfin, il serait possible d'utiliser des techniques de cryptographie afin d'améliorer la robustesse contre les attaques malveillantes associées (analyse statistique [66]...).

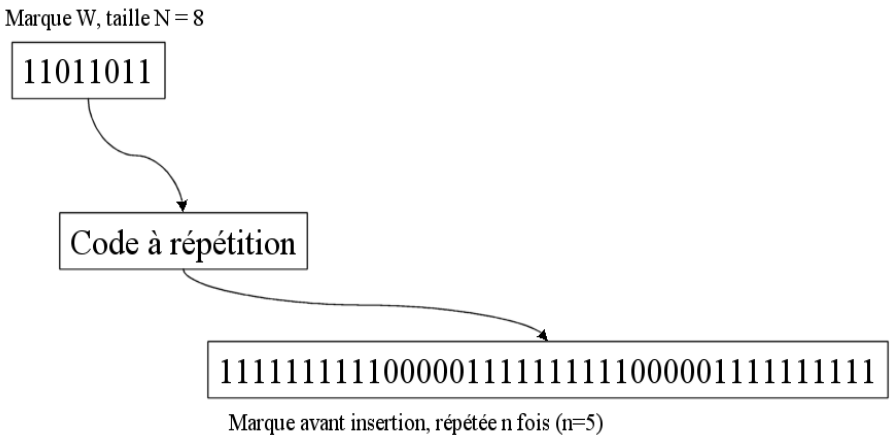


FIG. 38 – création d'un signal redondant

Une approche plus élaborée consiste à utiliser des techniques de code correcteur d'erreurs comme les turbo-codes [4] (ces codes sont connus pour être les plus performants).

Ajoutée à cette redondance, nous pouvons également réaliser un étalement temporel, afin de donner à chaque bit un poids qui est statistiquement le même. Pour cela, nous générons une marque continue qui sera insérée sur l'ensemble de la vidéo comme le montre la figure 39.

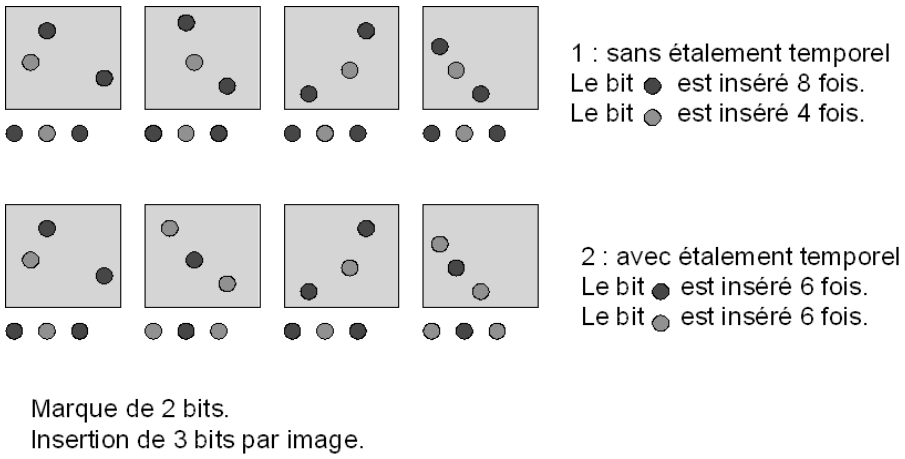


FIG. 39 – Etalement temporel

Cette approche permet d'éviter un mauvais comportement de la détection inhérente à une pondération biaisée des bits composant la marque. Cependant, on remarquera que l'utilisation de l'étalement temporel affaiblira la robustesse de notre algorithme vis à vis des attaques de sous-échantillonnage temporel.
Enfin, nous pourrions utiliser un masque psychovisuel afin de déterminer les localisations spatiales où le marquage sera susceptible d'être le plus transparent possible (ces aspects seront étudiés dans le chapitre 4).

3.4.4 Détection

La phase de détection de la marque correspond à un processus dual à celui de l'insertion. La détection est réalisée par une mesure de corrélation entre la marque extraite et la marque insérée (schéma de tatouage semi-aveugle). Pour une marque de 8 bits, le seuil de détection de la marque sera égal à 0.875 ce qui correspond à $7/8$. Cependant, afin de bien comprendre notre procédure de détection, nous nous proposons d'étudier de manière plus approfondie cette formule de corrélation.

Limite du produit de corrélation et probabilité d'erreur

La formule de corrélation utilisée lors de la détection est la suivante :

$$C_t = \frac{(t-1)C_{t-1} + (1 - \frac{d_H(\tilde{W},W)}{N})}{t}$$

où :

– C_t représente le score de corrélation à l'image f_t ;

– t représente l'indice de l'image en cours ;
– d_H est la distance de Hamming ;
– $\tilde{W}$ est la marque originale que l'on cherche à détecter ;
– W et la marque extraite au niveau de l'image f_t.

Si l'on considère le cas de mesure hard (on mesure directement chacun des bits de W), on peut modéliser le processus par un canal binaire symétrique, avec une probabilité d'erreur P_e.
On peut alors estimer la probabilité d'observation des différentes distances de Hamming :

$$P\left[d_H = k\right] = C_N^k (P_e)^k (1-P_e)^{N-k}$$

Remarquons que la variable aléatoire d_H a pour espérance $P_e N$. Etant donné que la formule récursive de calcul du produit de corrélation (qui n'est rien d'autre qu'un débruitage de la quantité $(1 - \frac{d_H(\tilde{W},W)}{N})$), le produit de corrélation va par conséquent avoir la limite suivante :

$$\begin{aligned}
\lim_{t\to\infty} C_t &= \lim_{t\to\infty} \frac{(t-1)C_{t-1} + (1 - \frac{d_H(\tilde{W},W)}{N})}{t} \\
&= \lim_{t\to\infty} \frac{1}{t}\left[(t-1)\left[\frac{(t-2)C_{t-2} + (1 - \frac{d_H(\tilde{W},W)}{N})}{t-1}\right] + (1 - \frac{d_H(\tilde{W},W)}{N}\right] \\
&= \lim_{t\to\infty} \frac{1}{t}\left[(t-2)C_{t-2} + 2\left[(1 - \frac{d_H(\tilde{W},W)}{N}\right]\right] \\
&\ldots \\
&= \lim_{t\to\infty} \frac{C_1}{t} + \frac{t-1}{t}\left[(1 - \frac{d_H(\tilde{W},W)}{N}\right] \\
&= E\left(1 - \frac{d_H(\tilde{W},W)}{N}\right) = \frac{1 - E\left(d_H(\tilde{W},W)\right)}{N} \\
\lim_{t\to\infty} C_t &= 1 - P_e
\end{aligned} \tag{45}$$

Il apparaît donc que les produits de corrélation ne tendent pas vers 1. En revanche, si la probabilité d'erreur sur le canal est très faible, ils tendent vers 1, et dans ce cas, il n'y a pas lieu de répéter les bits de la marque pour améliorer l'extraction.

Lien entre probabilité d'erreur de décodage, nombre de répétition et bruit

Nous venons de voir que le produit de corrélation donnait la probabilité d'erreur P_e pour chaque bit transmis, mais que ceci ne nous donnait pas pour autant la probabilité d'erreur sur les bits utiles b_i, répétés t fois.

Nous allons donc chercher à exprimer cette probabilité d'erreur sur les bits utiles, lors de l'utilisation de t répétitions. Tout d'abord, nous pouvons considérer le cas où un seul bit est transmis, puisqu'aucun lien n'existe entre les différents bits insérés (i.e. on n'utilise pas de codes correcteurs).
Par exemple, après t tirages, nous avons observé k_0 fois le bit correct et k_1 fois le mauvais. Pour déterminer le bon bit au niveau du "décodeur", nous pouvons considérer celui qui est apparu le plus grand nombre de fois, ou alors, donner une marge de robustesse et prendre un seuil de garde du type :
$k_O \geq \eta.t$, où en général $\eta \geq \frac{1}{2}$.
Tout d'abord, nous pouvons exprimer la probabilité pour un tirage d'observer k_0 fois le bit correct par la relation :

$$P\left[C_t = k_0\right] = C_t^{k_0}(P_e)^{t-k_0}(1-P_e)^{k_0}$$

Si on utilise un seuil η pour définir le bit correct, la probabilité d'erreur au décodage est alors :

$$P_{e,bit}(\eta, t) = \sum_{k_1=t-\eta.t}^{t} C_t^{k_1}(P_e)^{k_1}(1-P_e)^{t-k_1}$$

Nous allons maintenant donner un encadrement sur la probabilité d'erreur pour les bits utiles, à partir de la formule ci dessus.

Minorant de la probabilité d'erreur

Pour cela, on ne retient que le premier terme de la somme, soit :

$$P_{e,bit}(\eta, t) \geq C_t^{t-\eta.t}(P_e)^{t-\eta.t}(1-P_e)^{\eta.t}$$

Nous préciserons ensuite un équivalent de cette fonction pour $t \to \infty$.

Majorant de la probabilité d'erreur

Pour cela, prenons une suite géométrique majorante de la somme utilisée, pour la probabilité d'erreur. On a, en posant $A_k = C_t^k(P_e)^k(1-P_e)^{t-k}$:

$$A_k = \frac{1-P_e}{P_e}\frac{k+1}{n-k}A_{k+1} = \frac{1-P_e}{P_e}\frac{\frac{k}{n}+\frac{1}{n}}{1-\frac{k}{n}}A_{k+1} \tag{46}$$

et donc :

$$A_k \geq \frac{1-P_e}{P_e}\frac{\frac{k}{n}}{1-\frac{k}{n}}A_{k+1}$$

Comme la fonction $\frac{x}{1-x}$ est une fonction croissante, on peut alors dire que :

$$A_k \geq \frac{1-P_e}{P_e}\frac{\frac{k_0}{n}}{1-\frac{k_0}{n}}A_{k+1}$$

si $k \geq k_0$.

D'où en notant $\rho = \left[\frac{1-P_e}{P_e}\frac{\frac{k_0}{n}}{1-\frac{k_0}{n}}\right]^{-1}$, avec $k_0 = t - \eta.t$, nous obtenons une suite géométrique majorante pour les A_k :

$$A_k < \rho^{(k-k_0)} \times A_{\eta.t}$$

Nous pouvons vérifier qu'en général $\rho < 1$.
En majorant alors la somme utilisée dans la probabilité d'erreur par la somme de la série géométrique à l'infini, on a :

$$P_{e,bit}(\eta, t) \leq A_{t-\eta.t}\frac{1}{1-\rho}$$

soit encore :

$$P_{e,bit}(\eta, t) \leq \frac{1}{1-\rho}C_t^{t-\eta.t}(P_e)^{t-\eta.t}(1-P_e)^{\eta.t}$$

Equivalent de probabilité d'erreur

On vient de donner un encadrement sur la probabilité d'erreur de décodage d'un bit utile, et par celui-ci, nous pouvons observer que la tendance de cette probabilité d'erreur est la même que celle de :

$$B = C_t^{t-\eta.t}(P_e)^{t-\eta.t}(1-P_e)^{\eta.t}$$

En utilisant les équivalents et la formule de Stirling, on a :

$$n! \sim \sqrt{2\pi}n^{(n+\frac{1}{2})}e^{-n}$$

d'où :

$$\log n! \sim n\log n - n\log e$$

D'où encore après développement :

$$\log C_t^{t-\eta.t}(P_e)^{t-\eta.t}(1-P_e)^{\eta.t} \sim (t-\eta.t)\left[\log P_e - \log(1-\eta)\right] + \eta.t\left[\log(1-P_e) - \log\eta\right]$$

soit enfin :

$$\log B \ - tD_{KL}(Pe||1-\eta)$$

Où $D_{KL}(Pe||1-\eta)$ est la distance de Kullbak-Leibler, entre les lois binomiales de paramètres P_e et $(1-\eta)$ (qui par conséquent est une valeur positive).

On peut alors donner une "tendance" pour la probabilité d'erreur, pour les bits utiles qui varient alors en :

$$P_{e,bit}(\eta, t) \propto e^{-t.D_{KL}(Pe||1-\eta)}$$

Soit une décroissance géométrique assurée dans le temps.

Remarque : en pratique, on prend $\eta = \frac{1}{2}$, ce qui garantit d'avoir la plus grande distance de Kullbak-Leibler possible, et donc, d'avoir la décroissance en probabilité la plus rapide (la vitesse maximal est en e^{-t}). Également, si on utilise des codes correcteurs, on peut alors avoir des vitesses de décroissance de l'erreur de probabilité encore plus grandes.

3.4.5 Autres formes de grille

L'utilisation d'une grille de référence de forme rectangulaire (dans notre implémentation, nous utilisons le cas particulier d'une grille carrée) reste l'approche la plus intuitive. De par sa structure, elle est facilement implémentable et se rapproche de la partition en blocs utilisée classiquement dans les systèmes de compression. Cependant, la décorrélation entre l'espace de référence ainsi créé et l'espace des images sur lesquelles sont appliquées les vecteurs de mouvement, nous permet d'envisager d'autres formes de grilles. En effet, suivant l'application visée, certains types de grille pourraient s'avérer plus robuste. La grille carrée présente l'avantage de proposer une répartition relativement homogène de l'espace dans lequel va pouvoir se déplacer un vecteur marqué, sans pour autant perdre l'information de marquage. Cependant, dans le cas d'attaques spécifiques, il est possible d'adapter la géométrie de la grille afin d'anticiper de plus grandes déformations. C'est pourquoi, nous avons examiné le comportement de deux autres types de grille : une grille angulaire et une grille circulaire, et nous mettrons en perspective, dans le chapitre 5, l'utilisation d'une grille hexagonale dont nous présenterons rapidement les caractéristiques.

Nous allons commencer par présenter les caractéristiques d'une grille angulaire. Tout comme dans le cas d'une grille rectangulaire, nous effectuons une partition de l'espace dans lequel évolue les vecteurs de mouvement, en définissant deux zones différentes, auxquelles nous pourrons rattacher le bit $+1$ (zone Z_2) ou le bit -1 (zone Z_1). La règle va alors se définir par rapport à l'angle du vecteur de mouvement $\vec{OC}$ (c.f. figure 40). Puis, en fonction de la position de ce dernier, mais également en fonction du bit à insérer, nous effectuerons des modifications (s'il y a lieu) sur l'angle du vecteur. Cette étape aura une répercution sur les composantes en x et en y du vecteur (cf figure 41). On peut noter que dans ce cas, la direction du vecteur de mouvement marqué changera, mais son amplitude restera la même. Ainsi, on peut supposer que ce type de grille sera plus robuste à des attaques qui agiront seulement sur la amplitude des vecteurs de mouvement. C'est le cas

par exemple de l'application d'un changement d'échelle spatiale. Cependant, ce type d'attaque s'accompagne d'une désynchronisation des blocs rendant impossible une détection fiable, c'est pourquoi nous n'avons examiné que les attaques basiques qui seront définies en section 3.4, en accord avec le but que l'on s'est fixé (à savoir être robuste à différents types de codage).

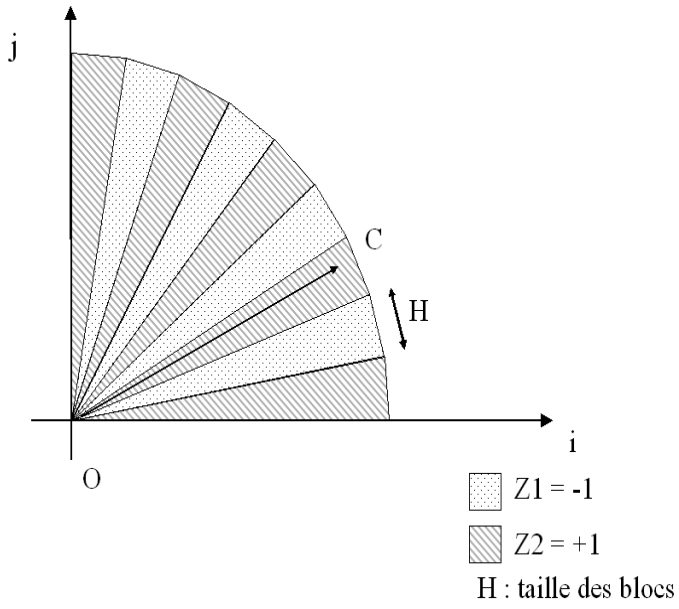


FIG. 40 – Partition en blocs de l'espace des vecteurs selon un découpage angulaire

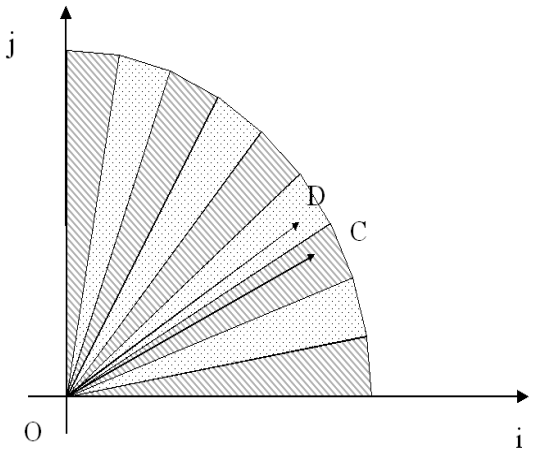


FIG. 41 – Insertion d'un bit par rapport à la grille angulaire

Dans le cas des grilles angulaires, les références sont représentées par les intersections entre les blocs triangulaires constituant la partition de l'espace d'insertion. Tout comme le cas des grilles

rectangulaires, nous plaçons le vecteur à marquer sur cette grille, puis en fonction de sa localisation et du bit à insérer, nous effectuons une symétrie par rapport à l'axe de référence.
La grille circulaire suit toujours le même principe. On réalise une partition de l'espace (comme montré sur la figure 42), puis on agit sur le module du vecteur de mouvement pour insérer la marque. Dans le cas d'une grille circulaire, seule la norme du vecteur sera modifiée nécessitant de faire varier les deux composantes du vecteur, cependant celui-ci gardera sa direction originale (cf. figure 43). Les références seront ici représentées par les cercles définissant la partition de notre espace d'insertion. La symétrie à appliquer sera une symétrie axiale dont le centre est définie par l'intersection entre le vecteur de mouvement et son "cercle" de référence.

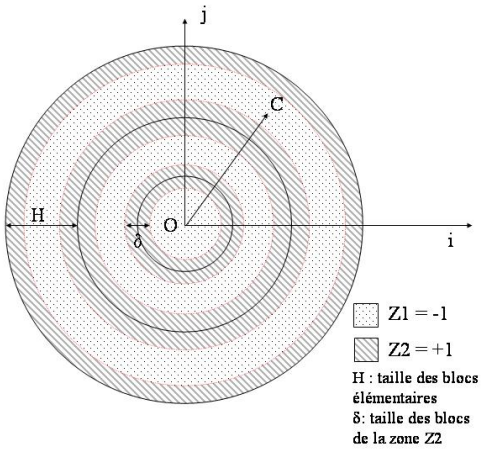


FIG. 42 – Partition en blocs de l'espace des vecteurs selon un découpage circulaire

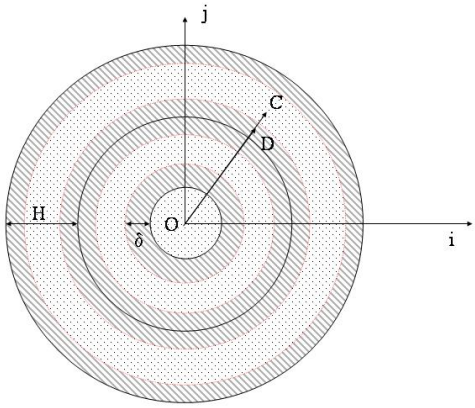


FIG. 43 – Insertion d'un bit par rapport à la grille circulaire

Pour chaque grille, nous déterminons les paramètres afin que les deux zones définies possèdent la même surface. Si ce critère paraît évident pour les grilles carrées (ou angulaire), le cas de la grille circulaire est légèrement différent. En effet, lorsque l'on examine la structure de cette grille, on peut s'apercevoir que cela revient à se placer dans un espace rattaché à l'amplitude du vecteur de mouvement, et il est alors important que les zones produisent une partition de cet espace homogène. Pour cela, il ne faut pas que la surface couverte par les deux zones soient toujours équivalente, car sinon les zones n'auraient plus la même surface "utile".
Nous avons testé trois types de grilles (carrée, angulaire et circulaire). Toutefois, il est possible d'utiliser d'autres types de grilles (par exemple hexagonale, rectangulaire, ou logarithmique...). Parmi ces autres possibilités, la grille hexagonale semble être plus particulièrement intéressante. En effet, de par sa structure, la répartition des zones semble être plus homogène que dans le cas des grilles carrées usuelles. De plus, il est possible d'utiliser ce type de grille avec une estimation de mouvement, réalisée sur une partition hexagonale de l'image. Par conséquent, on peut présupposer une meilleure robustesse du système.

3.5 Résultats

Afin d'étudier le comportement des différentes grilles élaborées dans les différentes variantes (étalement temporel, approche hiérarchique, sélection pseudo-aléatoire, etc.), nous avons testé la robustesse de notre système contre des attaques de traitement vidéo classique orientées essentiellement vers des attaques de type compression, à savoir :

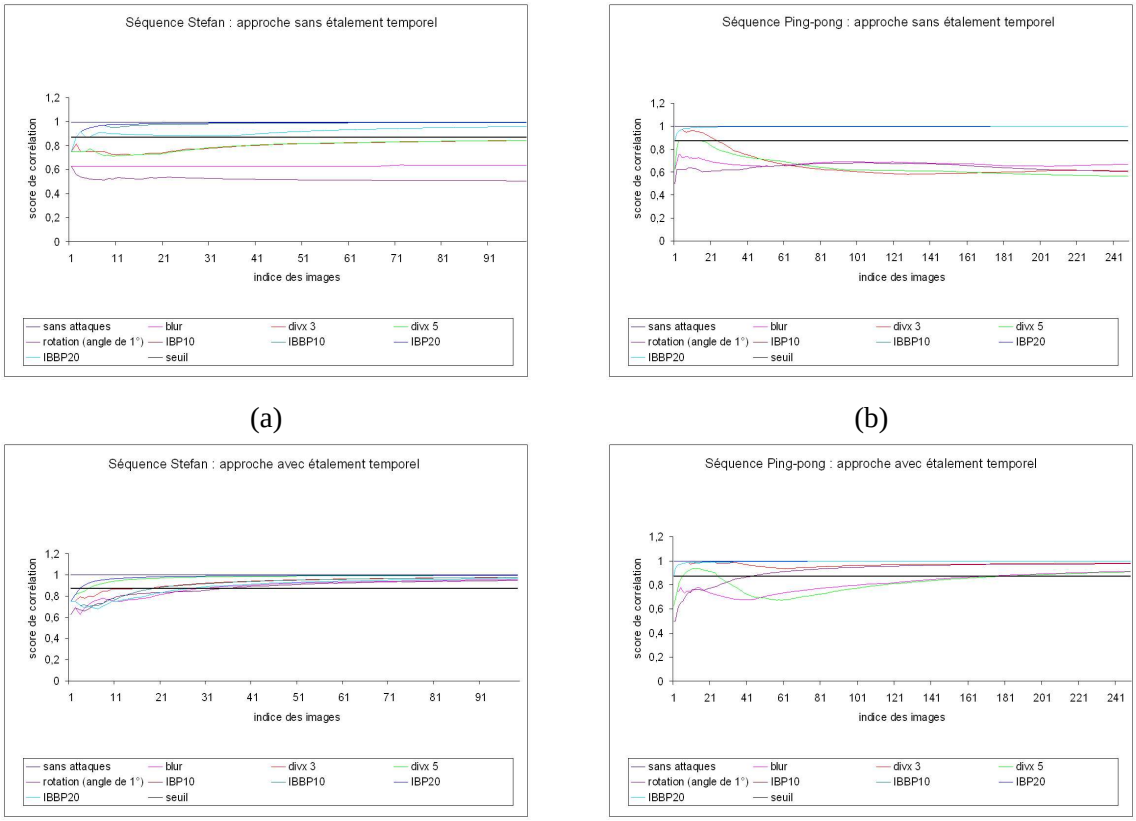
- Un filtrage blur, avec un noyau de taille 3 ;
- Une faible déformation géométrique, caractérisée par une rotation d'un angle de 1 degré ;
- Trois types de compression : le format divx (version 5 (respectivement 3), avec des taux de compression de 1 :54 (respectivement 1 :69)), et le nouveau standard émergeant H264. Dans ce cas, nous avons testé différents paramètres de compression, en utilisant les formats IBP et IBBP, accompagnés d'un pas de quantification de 10 et de 20, ce qui correspond aux taux de compression suivants : IBP10, 1 :15 ; IBBP10, 1 :15 ; IBP20, 1 :50 ; IBBP20, 1 :49.

Lors de nos tests, nous avons utilisé des vidéos au format $YUV420$. La réalisation des attaques fait intervenir des conversions YUV/RVB et AVI/RVB, rajoutant ainsi de légères dégradations. Nous présentons de manière détaillée les résultats sur les séquences "Stefan" et "Ping-pong". Nous conclurons en présentant les résultats récapitulatifs sur ces deux séquences, et sur les séquences "Lord of the ring" et "Tigre et dragon".
Dans l'ensemble des tests, nous avons utilisés la sélection pseudo-aléatoire, et lorsque ce n'est pas

précisé, nous n'utilisons pas l'approche adaptative. Le panel d'attaques que nous avons testé est relativement restreint, il serait donc intéressant de l'élargir à d'autres attaques : attaques spécifiques, autres traitements classiques (suppression d'images, changement de la fréquence,etc). En fonction de ces résultats, il faudrait alors envisager des solutions comme l'insertion d'un motif de resynchronisation spatial et temporel ou l'ajout d'une sous-séquence de resynchronisation au sein de la marque.

3.5.1 Étalement temporel

Afin de mettre en avant l'intérêt de l'utilisation de l'étalement temporel, nous présentons sur les figures 44 (a), 44 (b), 44 (c) et 44 (d) une comparaison de l'approche classique et de l'approche avec l'étalement temporel. Les résultats montrent le mauvais comportement de la détection lorsque l'on ne réalise pas l'étalement. Cela peut s'expliquer, comme nous l'avons déjà mentionné ci-dessus, par le fait que les bits insérés n'ont pas un poids égal lors de la détection. Une solution pourrait être donc de pondérer les bits lors de cette procédure. Cependant, les résultats de la détection seraient biaisés. Il est donc préférable d'appliquer l'étalement temporel. Il faut cependant rappeler, que lorsque l'on applique l'étalement temporel, notre algorithme devient plus sensible au sous-échantillonage temporel. On peut noter toutefois que l'approche sans étalement temporel présente une bonne robustesse vis à vis des compressions H264, mais que pour les autres attaques la marque n'est pas bien détectée, et ce pour les deux vidéos étudiées ici, c'est pourquoi dans la suite nous utiliserons l'approche avec l'étalement temporel.



(a) (b)

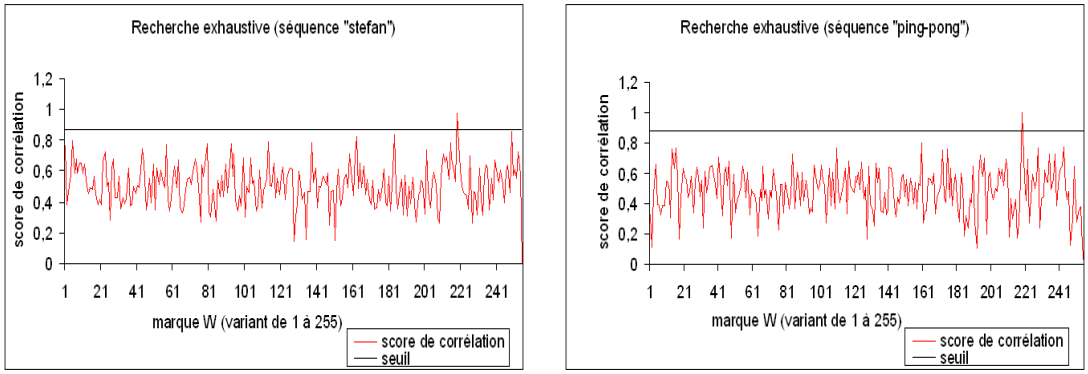
(c) (d)

FIG. 44 – Résultats de corrélation pour la séquence ”Stefan” (a) (respectivement (c)) et la séquence ”Ping-pong” (b) (respectivement (d)) avec l’approche sans étalement temporel (respectivement avec l’étalement temporel)

Dans la suite des résultats, les approches testées utilisent l’étalement temporel.

3.5.2 Recherche exhaustive

Tout d’abord, nous allons examiner le problème de fausse détection (le problème de fausse alarme ne sera pas étudié, en effet, une vidéo non marquée contiendra, en raison de la structure de notre algorithme, une configuration qui conduira à la détection d’une marque quelconque). Nous présentons, pour cela, les résultats de détection pour une recherche exhaustive des différentes marques. La figure 45 (a) présente les résultats pour la séquence ”Stefan”, et la figure 45 (b), pour la séquence ”Ping-pong”. Comme nous pouvons le voir sur ces figures, seule la marque insérée (ici 219) conduit à une détection qui dépasse le seuil de corrélation fixé. On peut donc en déduire que notre algorithme présente une bonne stabilité vis à vis des problèmes de fausse détection.



(a) (b)

FIG. 45 – Recherche exhaustive des marques pour la séquence ”Stefan” (a) et la séquence ”Ping-pong” (b)

3.5.3 Grille carrée

Dans cette section, nous avons souhaité montrer l’intérêt d’utiliser une sélection pseudo-aléatoire. Pour ce faire, nous présentons des résultats comparant les différents types de sélection présentées précédemment, mise en oeuvre dans le cadre de l’utilisation d’une grille de référence carrée. Sur les figures 46 (a) et 46 (b), nous utilisons une approche avec sélection déterministe, alors que, sur les figures 46 (c) et 46 (d), nous présentons les résultats avec une sélection pseudo-aléatoire. Nous pouvons voir sur ces figures, que l’approche utilisant une sélection pseudo-aléatoire, nous permet d’obtenir de meilleures résultats, en plus de nous permettre de diminuer les effets de clignotement.

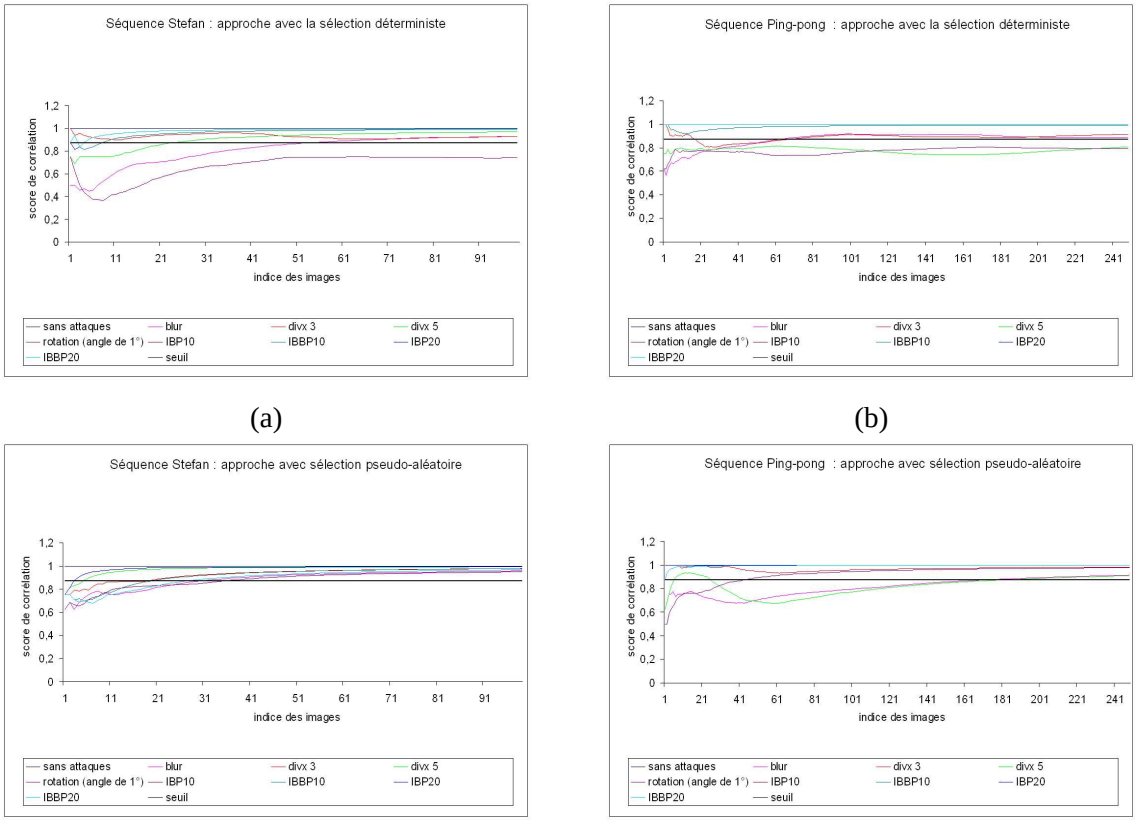
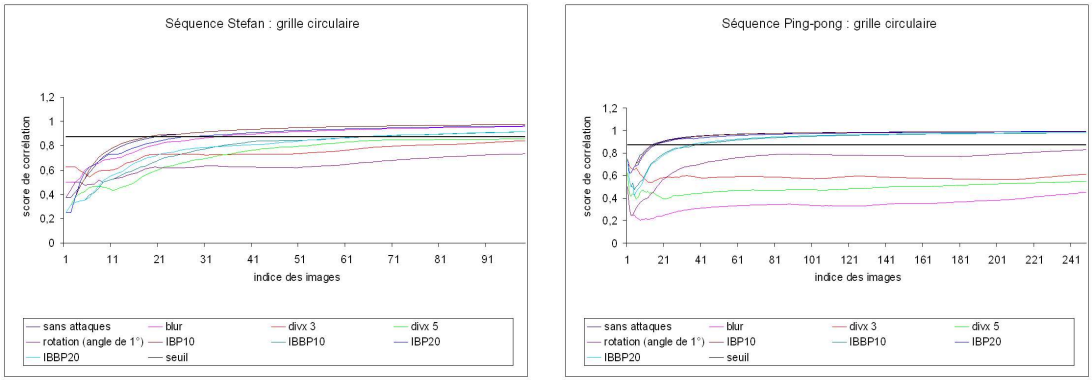


FIG. 46 – Résultats de corrélation pour la séquence ”Stefan” avec une sélection déterministe ((a) et (b)) ou une sélection pseudo-aléatoire ((c) et (d))

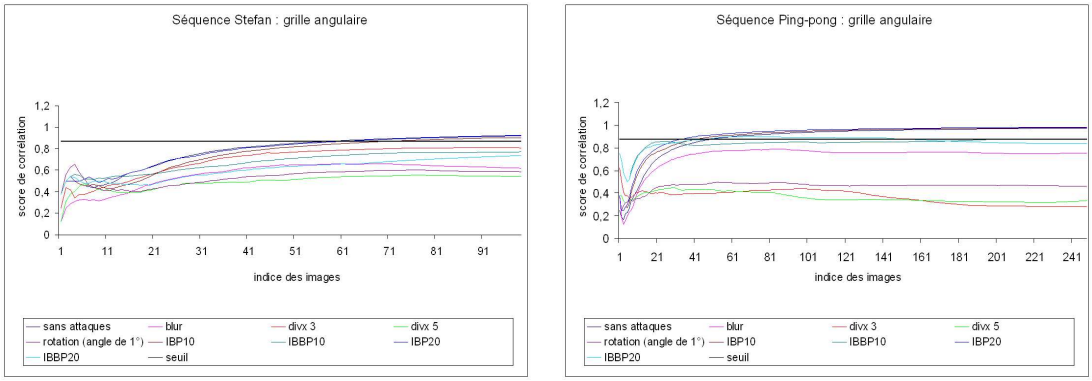
3.5.4 Grille circulaire et grille angulaire

Enfin, nous avons également réalisé une base de tests pour les différents types de grille présentées en section 3.3.5. Nous pouvons noter que les résultats concernant les grilles circulaires et angulaires offrent de moins bons résultats que ceux de la grille carrée. En effet, au premier abord, la géométrie de ces grilles semblent être moins bien adaptée à l'estimateur que nous avons utilisé, qui est basé sur une partition en blocs de l'image. Cependant, à ce jour, nous n'avons pas testé d'autres estimateurs. Nous pouvons supposer que le comportement de notre algorithme de marquage avec les différents types de grilles devrait être différent, et les résultats concernant les grilles angulaires et circulaires pourraient être améliorés. En effet, il y a une certaine dépendance entre l'estimateur de mouvement, le type de grille utilisé et la robustesse de notre algorithme.



(a) (b)

FIG. 47 – Résultats de corrélation pour la séquence ”Stefan” et la séquence ”Ping-pong” avec une grille circulaire



(a) (b)

FIG. 48 – Résultats de corrélation pour la séquence ”Stefan” et la séquence ”Ping-pong” avec une grille angulaire

3.5.5 Optimisations de l’algorithme

Jusqu’à présent, nous avons présenté différentes approches permettant d’optimiser globalement notre algorithme. L’utilisation de la sélection pseudo aléatoire nous permet de renforcer la robustesse de notre approche et de minimiser les effets de clignotement. Enfin le choix des grilles carrées semble être mieux approprié à notre contexte. L’étalement temporel a montré quant à lui son efficacité, cependant, comme nous l’avons noté, il faudra à l’avenir se pencher sur le problème de synchronisation temporelle qui représente une faille de cette approche.

Toutefois, de nombreuses optimisations sont encore possibles. En effet, celles-ci peuvent se situer tant au niveau de la formation de la marque, qu’au niveau de l’insertion ou encore au niveau de la

détection.
En ce qui concerne la création de la marque, il est courant d'utiliser des codes correcteurs : les codes par répétition/accumulation, les codes convolutifs ou encore les turbo-codes (mise en parallèle ou en série de plusieurs codes convolutifs). Dans cette thèse, nous avons utilisé le code par répétition/accumulation, qui consiste à répéter N fois le même bit à l'insertion afin de pouvoir corriger de légères déformations locales. Au niveau de la détection, on réalise alors un vote majoritaire afin de déterminer le bit le plus probable. Ce type de code correcteur d'erreurs est loin d'être optimal. Cependant, sa complexité est très faible et ne pénalise que très peu le temps d'exécution de notre algorithme, ce qui est primordial en vidéo. Pour accroître la sécurité, il est également possible d'utiliser des techniques issues de la cryptographie. On peut, par exemple, moduler la marque par un processus pseudo-aléatoire, afin de rendre l'extraction difficile voire impossible sans la clef utilisée pour initialiser le générateur. Pour ce faire, nous nous sommes orientés vers une sélection pseudo-aléatoire, réalisée grâce à un générateur (nous avons utilisé celui présent dans le compilateur C++ de Visual C++, il est possible d'envisager d'utiliser d'autres générateur tel que "Yarrow" décrit au chapitre 1 afin de s'assurer une meilleure robustesse), initialisé dans notre cas, par une combinaison de la marque et du numéro de l'image traitée.
Toutefois, cette approche, de par la sélection pseudo-aléatoire des vecteurs de mouvement, ne prend pas en considération l'impact visuel engendré par le marquage des vecteurs de mouvement. De ce fait, un certain nombre d'artefacts apparaissent sous forme d'effets de bloc. Pour pallier ce problème, nous avons développé une approche adaptative dont le masque se base sur un critère de PSNR. Cette approche est décrite dans la section suivante.

3.5.6 Approche adaptative

Comme nous l'avons déjà mentionné, la première approche ne prend pas en compte le problème de l'invisibilité du marquage. De ce fait, des artefacts peuvent apparaître au sein de la vidéo. Ceux-ci sont visibles sous l'aspect d'un effet de bloc dans les zones où est insérée la marque (c.f. figure 49). Dans cette section, nous nous intéressons donc particulièrement à la mise en oeuvre de variantes ayant pour but de réduire ces artefacts, en améliorant l'invisibilité de notre schéma, l'estimateur de mouvement gagnant ainsi en stabilité. En effet de trop grandes perturbations pourraient désorienter cet estimateur conduisant au calcul d'un vecteur de mouvement erroné, et par conséquent à une mauvaise détection. Dans cette section, l'approche présentée réalise un contrôle a priori de l'impact de notre système. Nous nous intéresserons dans le chapitre 4 à l'étude d'un contrôle a posteriori.

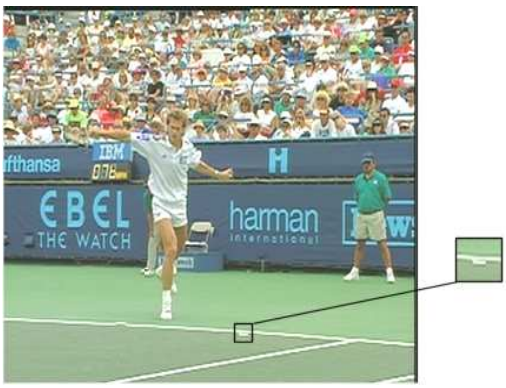


FIG. 49 – Illustration d'un effet de bloc dû au tatouage

Mise en place de la fenêtre de recherche

Afin de minimiser l'impact visuel engendré par le marquage des vecteurs de mouvement de notre précédent schéma, nous allons autoriser l'utilisation d'une phase d'insertion "exploratrice". En effet, comme décrit en section 3.3.2, la phase d'insertion consiste à appliquer si nécessaire des symétries visant à placer le vecteur à marquer dans la zone correspondant au bit à insérer. Nous rappelons que deux types de symétries sont employées (axiale ou centrale) selon la localisation du vecteur de mouvement. Afin de minimiser l'impact visuel, nous pouvons explorer les pixels, appartenant à la zone adéquate, dans le voisinage du point de référence comme le montre la figure 50. Puis, nous pouvons déplacer ce point de référence comme présenté sur la figure 50, afin d'élargir la zone de recherche.

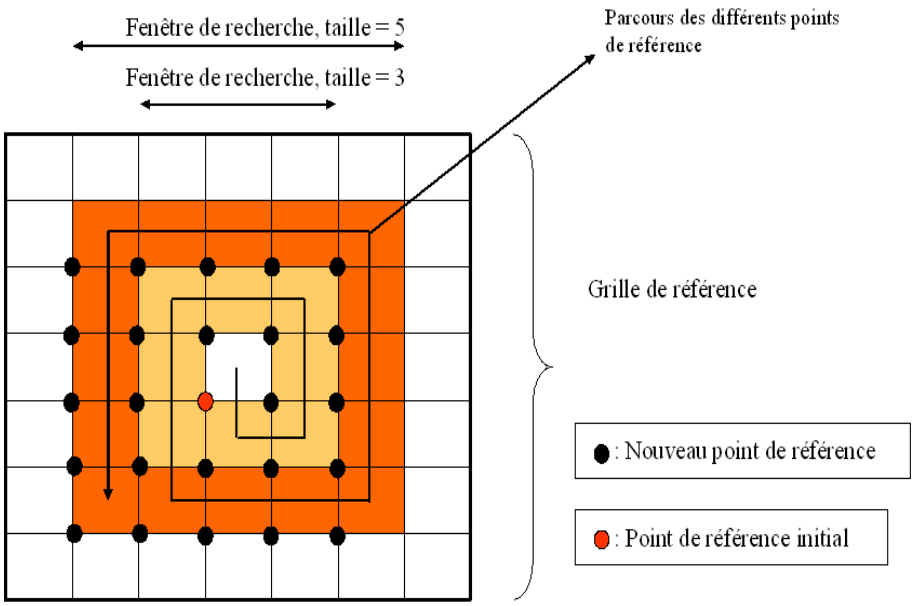


FIG. 50 – Parcours des vecteurs candidats dans un voisinage du point de référence initial

Au final, le vecteur sélectionné sera celui qui permettra de maximiser un critère de PSNR (c.f. figure 51).

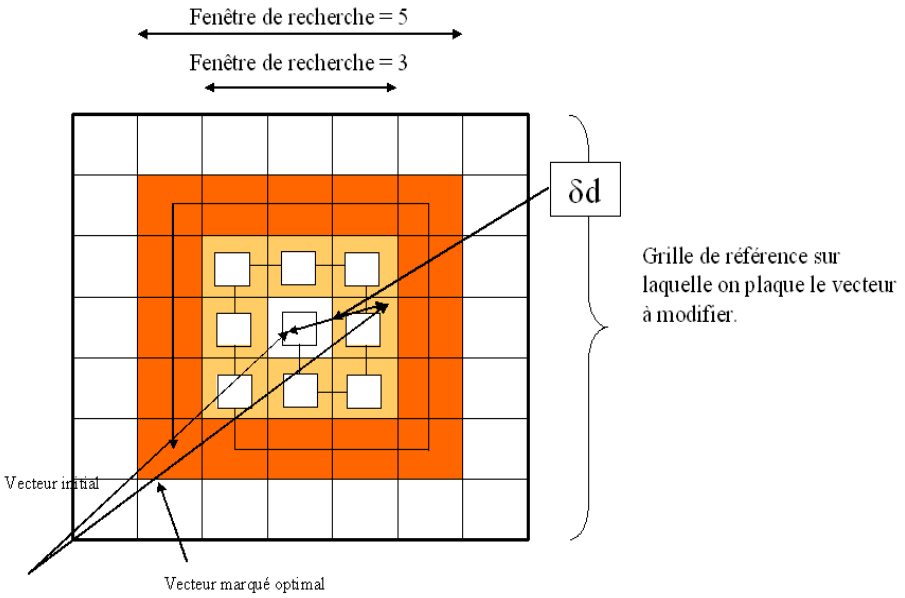
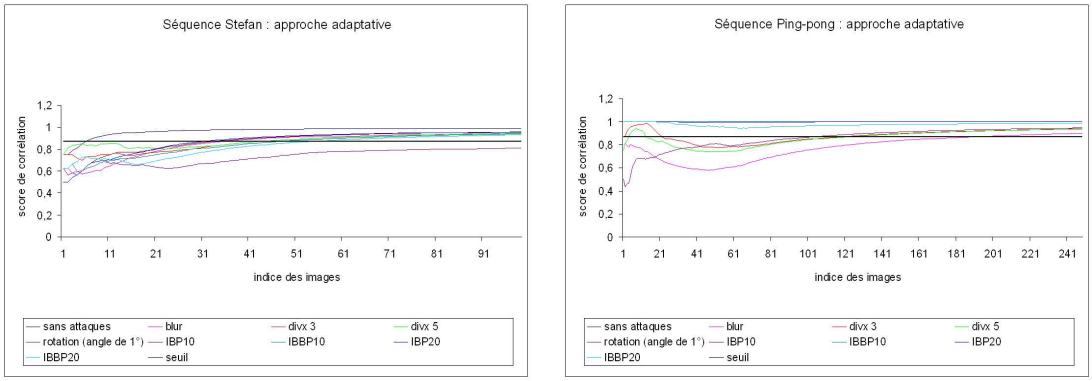


FIG. 51 – Détermination du meilleur vecteur candidat suivant un critère de PSNR

Les figures 52 (a) et 52 (b) présentent les résultats concernant cette approche.



(a) (b)

FIG. 52 – Résultats de corrélation pour la séquence ”Stefan” et la séquence ”Ping-pong” avec l’approche adaptative

Comme nous pouvons le voir, la détection se fait moins rapidement. En effet, avec cette approche, la probabilité qu’un vecteur marqué se trouve proche de l’intersection entre les deux zones est plus forte, ce qui fragilise légèrement la robustesse de notre algorithme. Cependant, comme présenté sur la figure 53, le gain en PSNR est non négligeable et l’approche reste toutefois valide.

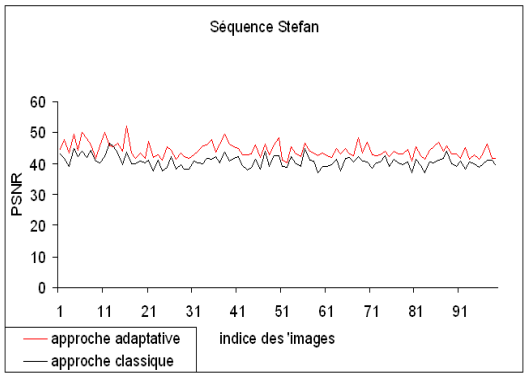


FIG. 53 – Comparaison des PSNR entre l’approche de base et l’approche adaptative sur la séquence ”Stefan”

3.5.7 Mise en place d’une zone intermédiaire entre les deux zones, 0 et 1

Enfin, une dernière amélioration de l’approche de base concerne la phase de détection. Les attaques réalisées sur une vidéo marquée engendrent des modifications au niveau des attributs marqués (ici les vecteurs de mouvement). Suivant les attaques, ces modifications peuvent être mineures (i.e. elles restent dans la sphère de quantification) ou majeures (i.e. elles sortent de la

sphère). Dans le second cas, l'algorithme de détection ne pourra extraire le bit adéquat contrairement au premier cas. Notre amélioration a consisté à définir une troisième zone, au niveau de la grille de référence. Cette zone intermédiaire se trouve entre les deux zones précédemment définies (zones Z_1 et Z_2). Celles-ci voient leur taille légèrement diminuée (c.f. figure 54). L'idée de mettre en place une zone "d'incertitude" vient de la logique floue. En effet, dans ce domaine, une variable n'est plus binaire. Il existe trois possibilité, 1, -1 et une valeur réelle appartenant à $[-1, 1]$ (zone d'incertitude). Dans notre cas, nous appliquons cette théorie lors de la détection d'un bit. Si le vecteur est dans la zone Z_1 ou Z_2, alors il n'y a pas d'ambiguïté. Par contre, s'il se trouve dans la zone intermédiaire (zone d'incertitude), c'est à dire entre les deux zones, la valeur du bit sera donnée par une fonction de type sigmoïde (c.f. figure 56). Plus le vecteur sera proche de la zone "dure", plus sa valeur sera proche de celle affectée à cette zone. En revanche, s'il se trouve au milieu, sa valeur sera moyenne et n'affectera pas la décision de la détection. Enfin, il serait possible d'étendre ce principe, dans le cas où la marque ne serait plus binaire (c.f. figure 55). Cette approche est appliquée en utilisant la version adaptative de notre algorithme.

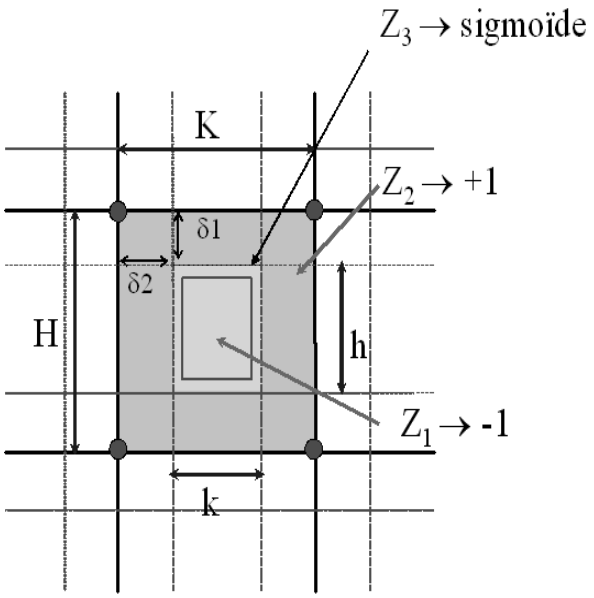


FIG. 54 – Mise en place d'une zone intermédiaire

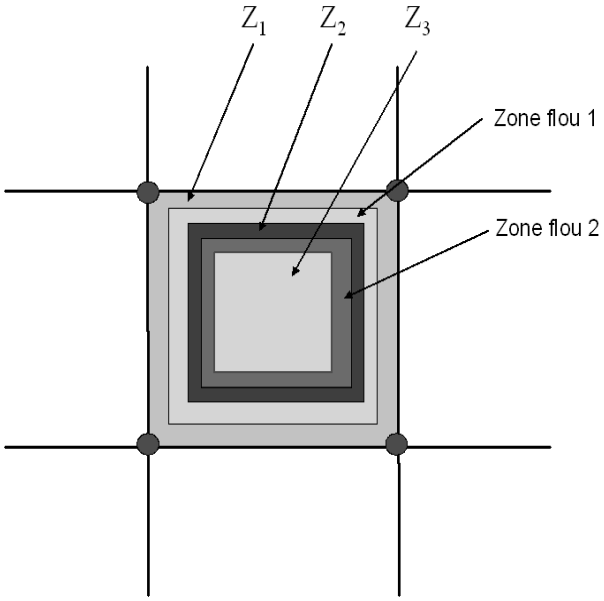


FIG. 55 – Extension à plusieurs zones

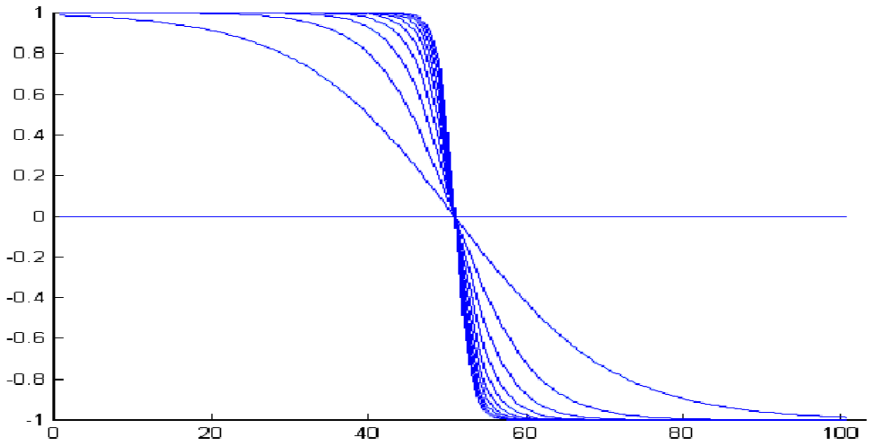
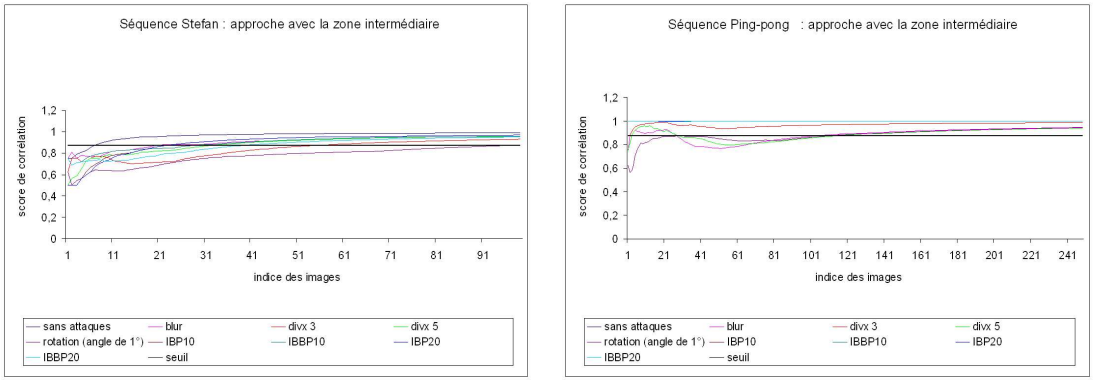


FIG. 56 – Exemple de fonction sigmoïde

Nous présentons les résultats de cette approche sur les figures 57 (a) et 57 (b). La version de l'algorithme utilisé est celui avec l'approche adaptative et la sélection pseudo-aléatoire. Sur la figure 58, nous présentons les résultats pour une attaque entre l'approche sans et avec la zone intermédiaire. Comme nous pouvons le constater, le seuil est dépassé dès la 43ème image avec la zone intermédiaire, alors que pour la version sans, il faut 59 image. On obtient donc une vitesse de convergence plus rapide pour l'approche présentée ici, elle est donc par conséquent plus robuste.



(a) (b)

FIG. 57 – Résultats de corrélation pour la séquence ”Stefan” et la séquence ”Ping-pong” avec la zone d’incertitude

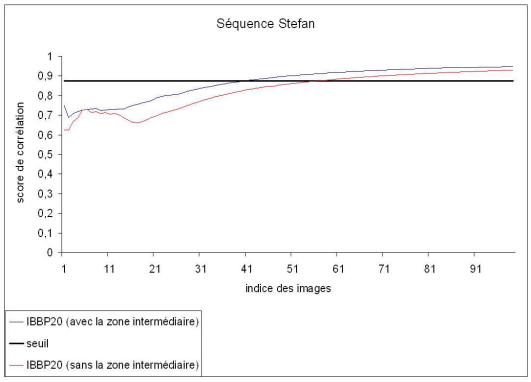
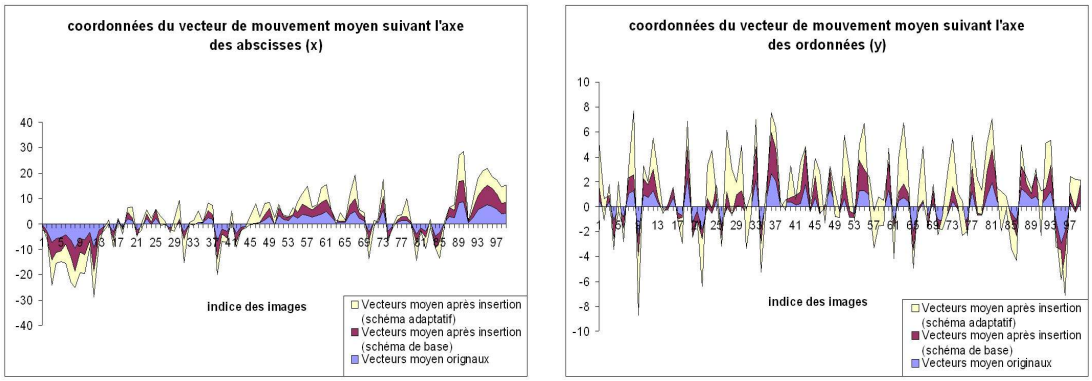


FIG. 58 – Comparaison de la vitesse de détection entre l’approche avec et sans zone intermédiaire

3.5.8 Décorrélation entre le PSNR et la variation des vecteurs

Enfin, il est intéressant de noter que dans notre approche, les variations effectuées sur les vecteur de mouvement ne sont pas directement liées à la dégradation de la séquence. En effet, l’approche adaptative conduit à réaliser de plus grandes variations sur les vecteurs de mouvement (c.f. figures 59 (a) et 59 (b)). Cependant, le PSNR est meilleur dans la version adaptative, par rapport à la version de base (c.f. figure 53). La force de marquage ne correspond pas dans notre cas à l’ampleur des variations que l’on est amené à réaliser sur les vecteurs de mouvements, mais sur le nombre de ces derniers servant de support à la marque.



(a) (b)

FIG. 59 – variation des vecteurs de mouvement suivant la composante en x (a) (respectivement en y (b))

3.5.9 Résultats récapitulatifs

Nous présentons ici un récapitulatif des résultats pour les quatre séquences présentées en section 3.4. Comme nous pouvons le voir sur la figure 97, les résultats concernant la séquence ”lord of the ring”, sont légèrement moins bons que pour les autres séquences. Cette séquence étant une bande annonce, il y a beaucoup de changement de scène, ce qui a pour conséquence de déstabiliser l’estimateur de mouvement et donc la procédure de détection. Pour le reste des séquences les résultats sont relativement homogènes et nous amène à conclure que notre algorithme présente une robustesse satisfaisante. On peut remarquer que l’attaque posant le plus de problème est la rotation. Malgré le faible angle utilisé, pour certaines approches, la détection ne se fait pas correctement. Notre algorithme ne prenant pas en compte les problèmes de synchronisation spatiale, l’application d’attaques suffisamment fortes visant à désynchroniser les blocs tatoués conduira à la mise en échec de notre système. Nous présenterons au chapitre 5, les méthodes qu’il est possible d’envisager afin de pallier ce problème.

Enfin, nous pouvons conclure que les différentes améliorations que nous avons présentées dans ce chapitre ont montré leur efficacité, et que notre système présente une bonne robustesse vis à vis des attaques de type compression.

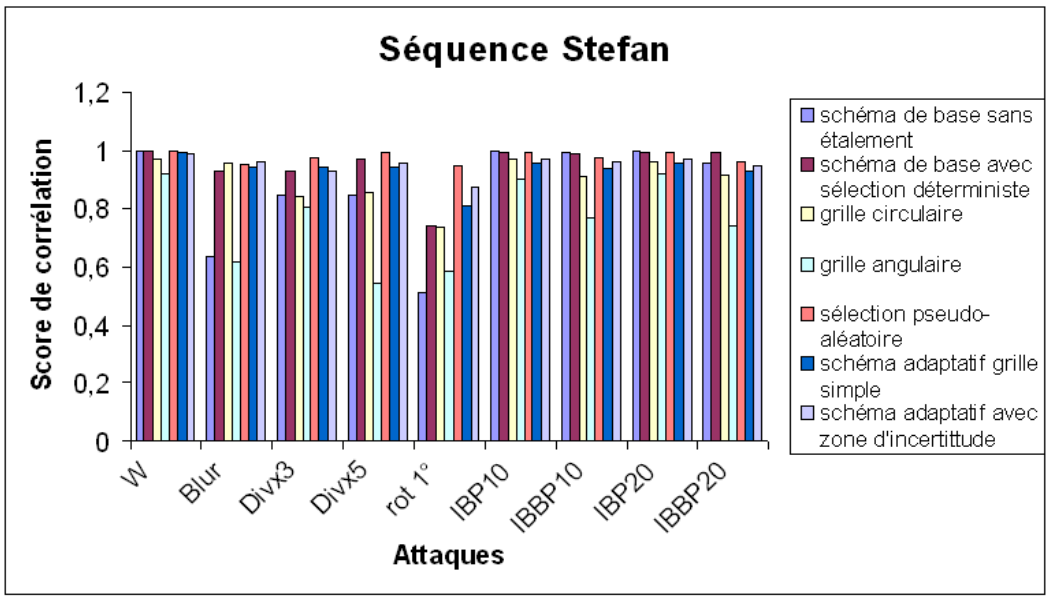


FIG. 60 – Résultats récapitulatifs pour la séquence ”Stefan”

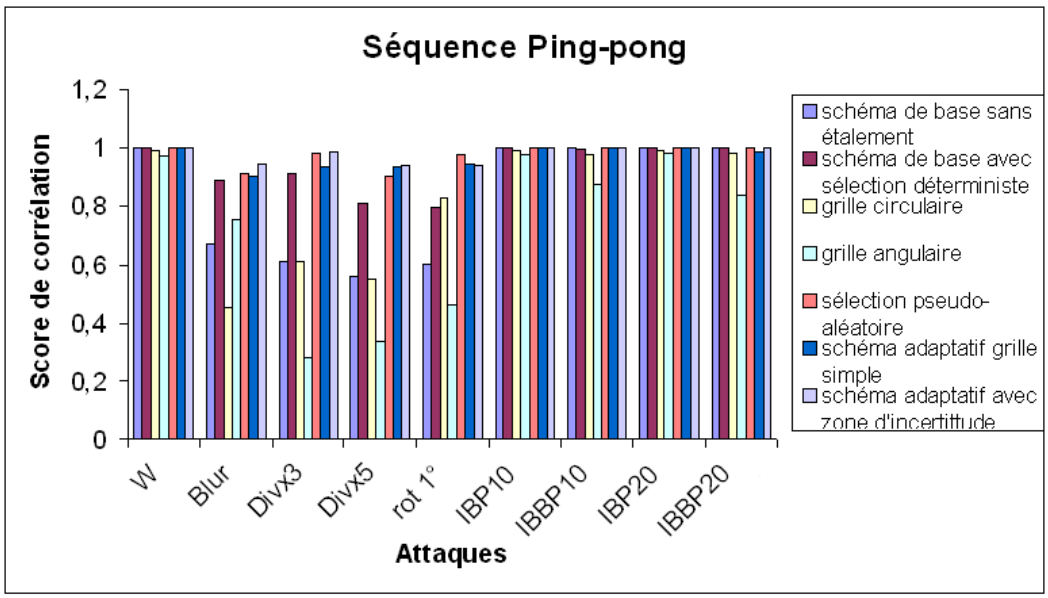


FIG. 61 – Résultats récapitulatifs pour la séquence ”Ping-pong”

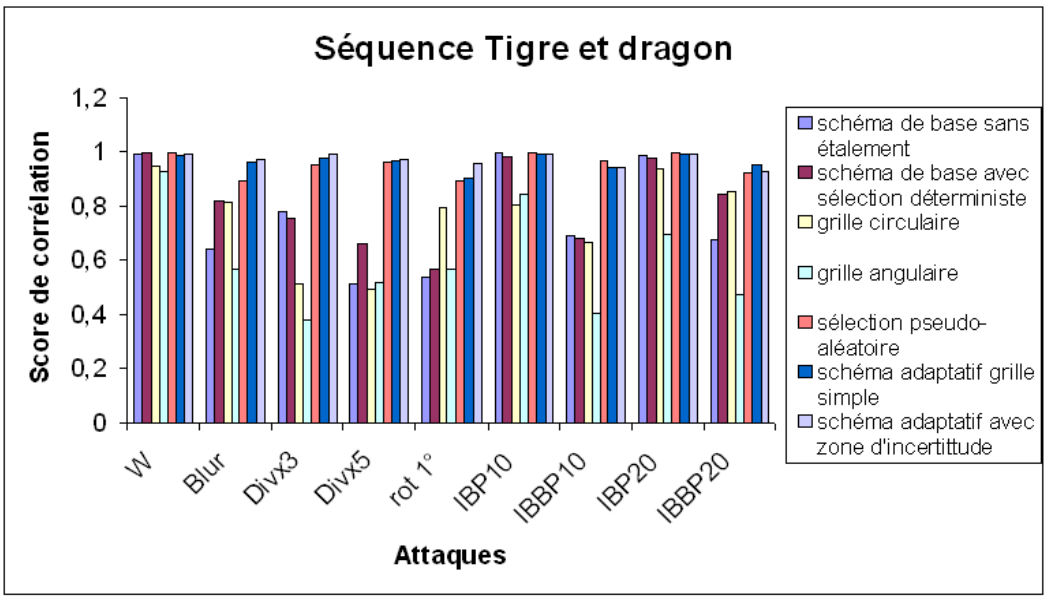


FIG. 62 – Résultats récapitulatifs pour la séquence ”Tigre et dragon”

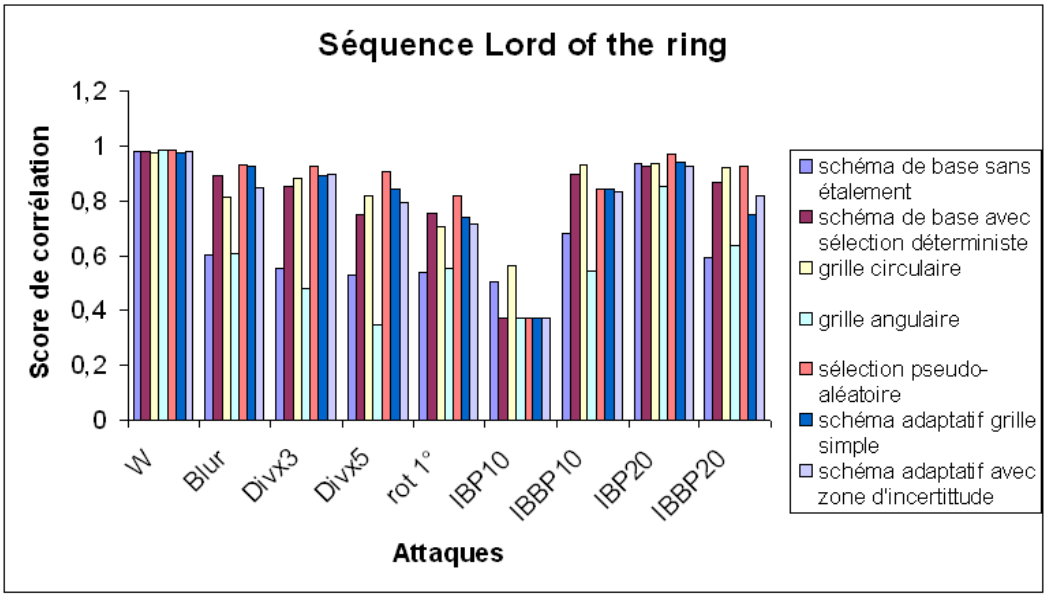


FIG. 63 – Résultats récapitulatifs pour la séquence ”Lord of the ring”

Chapitre 4

Aspects psychovisuels en tatouage vidéo

4.1 Introduction

Initialement, très peu de schémas de tatouage prenaient en compte les caractéristiques du système visuel humain. Cependant, de par la proximité entre le monde du tatouage et le monde de la compression, et afin d'optimiser le compromis invisibilité/robustesse, il est apparu nécessaire de tenir compte des particularités perceptuelles. Le système visuel humain est un appareil très complexe, en effet R.A. Young [126] estime que 80 à 90% de la totalité des neurones servent à la vision. De nombreux auteurs présentent les caractéristiques physiologiques de l'appareil visuel humain, la structure de l'oeil, les canaux de transmission jusqu'au cerveau, et la zone cérébrale d'analyse des informations visuelles. Nous ne rentrerons pas ici dans ces détails, le lecteur intéressé pourra se reporter aux références suivantes [136], [108], [42], [102], ainsi qu'au site internet "Webvision"[1]. L'obtention d'une modélisation fidèle et complète de l'appareil visuel humain s'associe aujourd'hui à la "quête du graal". Contrairement à l'audio, où les aspects perceptifs sont relativement bien maîtrisés, il est beaucoup plus difficile de les modéliser en image fixe ou en vidéo. Malgré cette difficulté d'élaborer un modèle parfait de la vision humaine, il existe un certain nombre de principes qui permettent d'optimiser les systèmes de compression ainsi que les systèmes de tatouage, en utilisant au mieux les caractéristiques de l'oeil humain.

Dans un premier temps, nous allons présenter un aperçu des propriétés de l'oeil du point de vue du traitement d'images, et non du point de vue physiologique. Nous présenterons essentiellement les propriétés les plus couramment utilisées en image fixe ainsi qu'en vidéo. Ensuite, nous décrirons les différents artefacts auxquels on peut être confronté en traitement d'images, pour poursuivre sur la présentation des différentes méthodes qui permettent d'évaluer la qualité ainsi que la similarité d'une image ou d'une vidéo, en tenant compte des aspects perceptuels. Nous poursuivrons sur l'utilisation de ces méthodes dans le contexte du tatouage. Pour finir nous détaillerons les approches testées dans notre système afin de contrôler les dégradations apportées par notre algorithme basé sur la perturbation de vecteurs de mouvement.

4.2 Caractéristiques du système visuel humain

L'avènement de l'ère numérique et l'explosion des performances informatiques ont donné aux ingénieurs plus de liberté et de flexibilité dans le traitement de l'information. De nombreuses mesures de qualité sont apparues, et initialement ces méthodes sont des métriques purement mathématiques telles que les mesures MSE (Mean of Square Error) ou SNR (Signal to Noise Ratio). Elles s'appliquent pixel à pixel et ne prennent en compte ni les relations de voisinage qui

[1] http ://webvision.med.utah.edu/

peuvent exister entre les pixels d'une même zone, ni les caractéristiques du système visuel humain. Cependant de par leur facilité d'utilisation, elles sont très couramment utilisées et offrent une première approximation de la qualité d'une image, en donnant une mesure de similarité. Or, le système visuel humain présente des limitations qui, bien exploitées, permettent de cacher de l'information de façon imperceptible dans un medium, qu'il s'agisse de l'image fixe ou de la vidéo. La problématique qui se pose aujourd'hui est de déterminer un moyen automatique pour mesurer la qualité et la similarité d'un support, en élaborant un modèle qui puisse se rapprocher au mieux des tests subjectifs. Dans la suite de ce chapitre, nous nous focaliserons sur l'image fixe et sur la vidéo. Bien qu'il ait été démontré que l'audio joue un rôle dans la perception visuelle [130], nous avons choisi de ne pas aborder ce domaine. Une bibliographie sur le sujet de la vision est présente sur le site web d'Ahumada [2].

4.2.1 Caractéristiques bas-niveau

Le système visuel humain a été étudié dès 1704, époque à laquelle Newton a posé les fondations de l'étude moderne de la perception de la couleur dans son "Traité d'optique" [80] (Marr [47] présente dans son ouvrage un historique détaillé de l'étude de la vision). Même si les premiers travaux sur le sujet remontent au 18ème siècle, il a fallu attendre l'émergence du calcul numérique pour voir se réaliser les progrès les plus significatifs. Ceux-ci ont, de fait, été motivés par la nécessité de représenter numériquement des images et d'en optimiser les traitements. Très vite, il est apparu que certains phénomènes pouvaient induire en erreur notre perception, et mettre à jour l'imperfection de notre vision. En se basant sur ces propriétés, les scientifiques ont pu améliorer le rendu des images numériques, et optimiser les systèmes de compression qui intègrent à présent des processus complexes, visant à ne garder que les informations perçues par notre système visuel. Les travaux effectués dans ce domaine ont tout d'abord été majoritairement concentrés sur l'image fixe (T. Eude [146] a réalisé un bon aperçu de quelques systèmes d'évaluation de la qualité des images fixes). Ce n'est que récemment que l'étude de la vidéo a été abordée. F.X.J. Lukas et Z.L. Budrikis [64] ont été les premiers en 1982, à proposer une métrique basée sur un modèle spatio-temporel du système visuel humain. Depuis, de nombreux chercheurs se sont penchés sur le problème et de nombreuses métriques ainsi que de nombreux modèles du système visuel humain ont vu le jour.

Initialement, les aspects psychovisuels de la perception humaine ont été étudiés dans le cadre de la compression d'image fixe ou de vidéo.

Les systèmes de compression sont majoritairement basés sur des schémas intervenant sur une partition en blocs de l'image. Classiquement, les blocs sont de taille 8×8 ou 16×16 [1], et plus

[2] http ://vision.arc.nasa.gov/personnel/al/bib.htm

récemment, le système H264 [150] exploite des tailles de blocs moins conventionnelles. Il est donc nécessaire dans ce contexte d'élaborer des systèmes de mesures qui prennent en compte cet aspect de partitionnement en blocs qui combiné à de forts taux de compression, génère des discontinuités dans l'image ou la vidéo, celles-ci étant visibles par l'oeil humain (cf figure 64).

FIG. 64 – Artefacts de compression : à gauche l'image originale, à droite l'image fortement compressée avec JPEG (taux de compression de 80%)

Considérons la structure générale d'un modèle du système visuel humain présentée par la figure 65 :

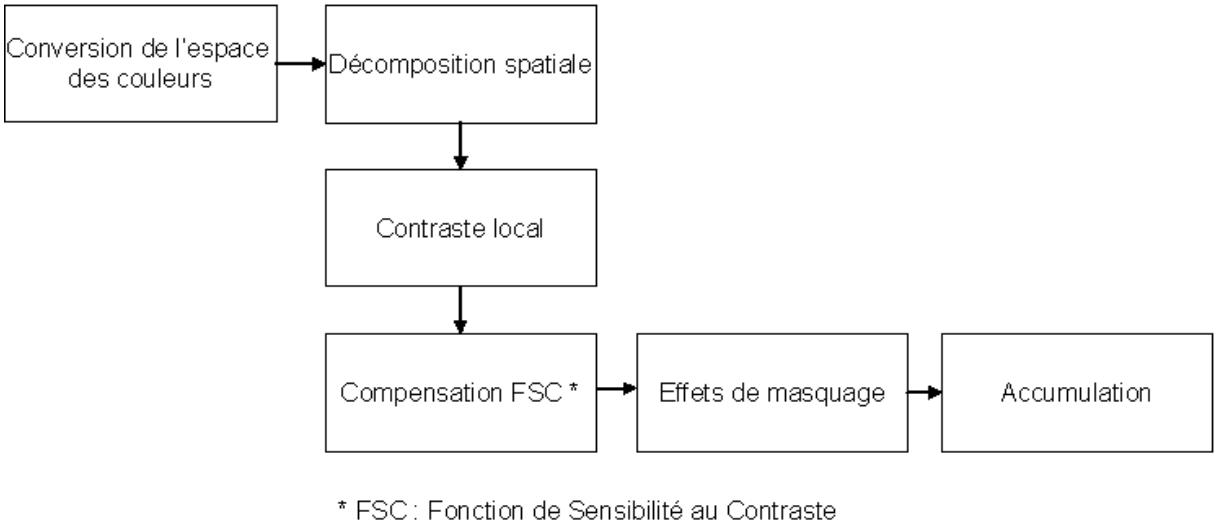


FIG. 65 – Modèle générique du système visuel humain

- l'image d'entrée, généralement donnée en RGB, est convertie dans un espace colorimétrique qui permet une bonne prédiction des distortions des couleurs, ainsi qu'une meilleure exploitation des redondances ;
- l'étape suivante consiste à décomposer l'image suivant ses fréquences spatiales orientées. Il est ici important d'appliquer une décomposition qui se rapproche au mieux du système visuel humain ;
- la troisième étape consiste à mesurer le contraste local, ce calcul peut être réalisé par des processus non-linéaires qui se basent sur la décomposition spatiale de l'image ;
- la quatrième étape prend en compte la diminution de la sensibilité au contraste pour les hautes fréquences. Cette propriété est modélisée par la fonction de sensibilité au contraste. Cette étape permet ensuite d'appliquer le modèle de masquage à toutes les fréquences spatiales avec les mêmes paramètres ;
- la cinquième étape, celle de masquage, permet de prédire la quantité de bruit de quantification qu'il est possible d'occulter dans le signal original, grâce à la présence de forts contrastes ou de zones à fortes activités ;
- la sixième et dernière étape, consiste à accumuler toutes les données suivants les différentes fréquences spatiales, afin d'obtenir une valeur unique qui caractérisera la différence perceptuelle entre l'image (ou la vidéo, dans ce cas on considère aussi les fréquences temporelles) originale et la version compressée.

M. Nadenau [108] montre les aspects du système visuel humain appliqués dans le cadre de la compression d'image fixe. S. Winkler [136], quand à lui, présente leurs utilisations dans le cadre

du tatouage d'image fixe, ainsi qu'un aperçu de métriques appliquées à la vidéo. Nous nous baserons par la suite sur ces travaux pour présenter les propriétés essentielles de l'oeil.

Sensibilité au contraste

L'étude de la perception humaine ne se limite pas seulement à la compréhension de l'appareil optique que représente l'oeil : Entre la vision de l'image qui s'imprime sur la rétine et son interprétation au sein du cerveau humain, le chemin est long et fait intervenir des processus complexes qui sont, à ce jour, loin d'être totalement maîtrisés. Tout d'abord, l'oeil est sensible au contraste, et même si le système visuel humain est capable d'appréhender des petites différences de luminance, il existe une limite en dessous de laquelle les différences ne sont plus perçues. Cette limite dépend de la luminance L_0 du fond sur lequel se trouve les composantes de l'image. Cette dépendance est appelée "sensibilité au contraste", et peut être caractérisée mathématiquement par la loi de Weber-Fechner. Cette loi peut être formalisée par le contraste de Weber :

$$C^W = \triangle L / L_0 \tag{47}$$

Cette définition est utilisée dans le cas de stimuli constitués de petits motifs, $\triangle L$ représentant "l'offset" de luminance et L_0 la luminance d'un fond uniforme. Dans le cas de motifs périodiques avec une déviation symétrique variant entre L_{min} et L_{max}, le contraste de A. Michelson [10] est utilisé :

$$C^M = \frac{L_{max} - L_{min}}{L_{max} + L_{min}} \tag{48}$$

Ces deux définitions ne sont pas équivalentes, mais elles représentent néanmoins une bonne approche dans le cas de stimuli simples. Lorsque ces derniers sont plus complexes, ces définitions ne sont plus applicables, c'est le cas pour les motifs de Gabor [56]. Il est évident que pour des images naturelles, ces deux définitions ne sont pas utilisables. Ceci est dû au fait qu'un point très clair ou très sombre déterminera le contraste pour l'image entière, alors que la perception humaine du contraste varie en fonction de la luminance locale moyenne. Pour pallier ce problème, E. Peli [55] propose la définition d'un contraste local :

$$C_j{}^P(x,y) = \frac{\psi_j * I(x,y)}{\phi_j * I(x,y)} \tag{49}$$

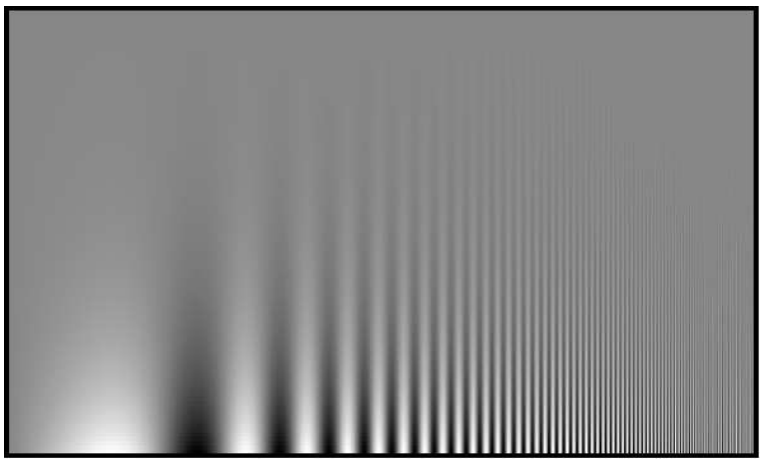


FIG. 66 – diagramme de sensibilité au contraste de Campbell-Robson

où ψ_j est un filtre passe-bande d'un banc de filtre au niveau j, et ϕ_j est le filtre passe-bas correspondant. Cette définition du contraste est bien définie si certaines contraintes sont respectées par les noyaux des filtres. Sous l'hypothèse que ϕ et l'image sont des fonctions à valeurs réelles positives, qu'elles sont intégrables, et que ψ est intégrable, alors ${C_j}^P(x,y)$ est bien définie si le support de ψ est inclus dans le support de ϕ. Dans ce cas, $\phi_j * I(x,y) = 0$ implique que ${C_j}^P = 0$. J. Lubin [89] propose une version modifiée de la définition de E. Peli. Cette définition du contraste est intégrée dans une métrique basée sur un modèle multi-canal du système visuel humain :

$$ {C_j}^L(x,y) = \frac{\big(\phi_j - \phi_{(j+1)}\big) * I(x,y)}{\phi_{(j+2)} * I(x,y)} \tag{50} $$

Les résultats d'expériences psychophysiques d'analyse du contraste avec les motifs de Gabor [56], montrent que cette définition du contraste est relativement bien adaptée.

La figure 66 présente le diagramme de sensibilité au contraste de Campbell-Robson [63]. La fonction de sensibilité spatiale au contraste apparaît comme étant l'enveloppe visible du motif modulé.

Perception de la lumière

Une autre caractéristique importante du système visuel humain est la perception de la lumière. Cette perception est non-linéaire, et cette propriété doit être prise en compte dans les systèmes de compression qui ne doivent pas s'appliquer par conséquent linéairement. La loi de puissance

définie par la formule suivante :

$$B = a_l L^{p_l} - B_0 \tag{51}$$

est considérée comme une bonne description de la luminance perçue par le système visuel humain [74]. B correspond à l'intensité lumineuse perçue par un observateur, en présence d'un stimulus de luminance L. B_0 est un offset constant d'intensité lumineuse perçue. L'exposant p_l a une valeur approximative de $1/3$. Cette loi est utilisée dans l'espace de couleur $L * a * b$.

Perception de la couleur

Nous avons évoqué les cas du contraste et de la luminance, mais afin d'être plus exhaustif, nous allons maintenant exposer brièvement la perception de la couleur.
La couleur est un concept intrinsèque à l'être humain, elle lui permet de faire la distinction entre deux sensations colorées. Physiquement, les variations de couleur correspondent à des variations de longueur d'ondes. C'est en 1878 que Ewald H. Hering introduit l'idée des couleurs opposées. Plus récemment, il a été montré que les informations de couleur provenant des trois récepteurs différents constituant la rétine, étaient combinées mutuellement pour former un canal achromatique et deux canaux chromatiques. Cette combinaison constitue la base pour la définition d'un espace de couleurs opposées. Pour plus de détails sur la couleur, le lecteur intéressé pourra se reporter aux références suivantes [69] et [44].

Les trois caractéristiques que nous venons de présenter sont à la base des effets de masquage que nous allons décrire dans la suite. Ces effets peuvent être ingénieusement utilisés dans un système de tatouage afin d'en optimiser la force et l'invisibilité. La définition de modèle mathématique, que ce soit pour la perception du contraste, de la lumière ou encore de la couleur, s'approchant au mieux du système visuel humain, permet d'en exploiter les défauts. Ainsi, en codage, elles permettront de ne garder que l'information essentielle nécessaire à une bonne perception du médium. En tatouage, elles permettent de se servir des informations non indispensables pour y insérer la marque. Nous verrons à la fin de ce chapitre, l'utilisation d'une mesure locale du contraste que nous avons utilisée afin de pondérer un masque visant à éliminer les blocs marqués provoquant de trop grande dégradation.

Les effets de masquage

La sensibilité dans les zones homogènes a été étudiée dans les précédents paragraphes. Cependant, en pratique, les images à traiter sont dites naturelles, et composées de nombreux petits

détails, de zones texturées, de zones homogènes et de formes variées. Il est donc nécessaire dans ce contexte de décrire la sensibilité au contraste dans le cadre des fréquences spatiales. L'effet de masquage est un terme général qui dénote l'effet de masque que peut produire un élément A sur un élément B. Les effets de masquage peuvent être de différentes natures, par exemple un masquage spatial (masquage de contraste ou masquage d'entropie) ou un masquage temporel.
Généralement, cette description est donnée par la fonction de sensibilité au contraste (FSC, ou CSF en anglais : *Contrast Sensibility Function*). Le système visuel humain est moins sensible dans les hautes fréquences. Ce phénomène se traduit dans la pratique par une moindre sensibilité au bruit dans les zones texturées, comme l'illustre l'image 67, où une même quantité de bruit a été ajoutée dans deux zones de même taille, l'une homogène, l'autre texturée.

FIG. 67 – Masquage d'activité.

Le masquage de contraste :

Le masquage de contraste fait référence au cas où les deux signaux sont de même nature (le signal proprement dit et le signal de masquage). Par exemple, lorsqu'ils sont tous les deux des motifs sinusoïdaux, et quand ces deux signaux sont superposés, le troisième motif ainsi obtenu peut soit être différent des deux premiers signaux (et dans ce cas il n'y a pas de masquage), soit

être similaire au signal de masquage (et dans ce cas le phénomène peut être observé). Cet effet dépend de la fréquence des deux signaux et de leur orientation spatiale (cf figure 68).

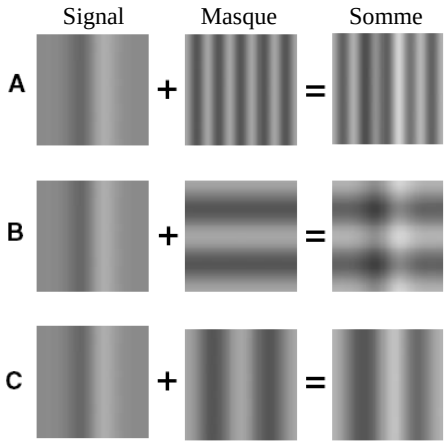


FIG. 68 – Masquage de contraste.

Le masquage d'entropie (ou masquage d'activité) :

Le masquage d'entropie est lié au masquage de contraste. En effet, plus il y a d'information dans une zone (et donc plus l'entropie est grande), plus il y a de variation de contraste (en général). Tout comme pour le contraste, un signal sera moins perceptible dans une zone fortement texturée (donc avec une forte entropie) que dans une zone homogène (cf figure 67). La différence entre les modèles pour le masquage de contraste, et ceux pour le masquage d'entropie, se situe dans le support spatial. Le masquage de contraste est souvent appliqué de façon précise. Cela signifie que le contraste à la position (x, y) indique la quantité d'erreurs de quantification que l'on peut appliquer au même endroit, sans qu'elle soit perceptible. Le masquage d'entropie considère, quant à lui, la totalité du voisinage local pour établir la visibilité du signal de distortion. Parfois, les deux effets ne sont pas dissociables, et sont combinés dans le même modèle de masquage.

Adaptation de l'oeil :

Enfin, il existe également un phénomène correspondant à l'adaptation de l'oeil au stimulus qui lui est envoyé, et plus particulièrement, aux motifs perçus. Par exemple, l'adaptation à un motif d'une fréquence donnée peut provoquer une légère diminution de la sensibilité au contraste autour de cette fréquence. De nombreuses études ont été réalisées sur cet aspect et ont démontrées que le contenu de l'image (fréquence, distribution de la couleur, motif) influençait la sensibilité de l'oeil.

Nous reviendrons sur l'aspect temporel dans la section suivante, consacrée à la vidéo même si certains aspects temporels sont aussi présents au niveau de l'image fixe.
Les phénomènes que l'on vient de décrire peuvent être qualifiés de bas-niveau, en effet ils ne font que très peu intervenir l'interprétation cérébrale dans la perception du contenu de l'image.

En résumé, l'activité locale permet d'envisager un marquage plus ou moins fort, sans que ce dernier soit perceptible. En effet, les zones texturées offriront un meilleur masquage au marquage que les zones homogènes. Les zones à fort contraste sont essentiellement les contours, où il sera possible aussi de cacher plus d'information sans dégrader la qualité de la vidéo. Ces aspects peuvent être mis en oeuvre grâce à une segmentation intelligente de l'image ou encore grâce à l'utilisation de points saillants. Nous reviendrons sur ce dernier point dans le chapitre 5.

4.2.2 Caractéristiques haut-niveau

Nous allons aborder dans cette section un aspect plus psychologique que physiologique de la perception humaine, il s'agit de phénomènes souvent qualifiés de haut-niveau. Nous avons vu précédemment que le système visuel humain a ses restrictions, mais la perception d'une image ne se limite pas à la simple visualisation d'une image puis à sa transmission. Au bout de cette chaîne, déjà complexe, se trouve le cerveau humain qui va interpréter ce que l'oeil capte et ce que le circuit neuronal lui transmet. Cet aspect est sans doute l'un des plus connus du grand public. En effet, l'artiste Escher élabore dans ses oeuvres, des scènes impossibles et bien souvent déroutantes. En pratique, ces éléments sont rarement intégrés dans les modèles perceptuels, mais servent cependant à l'élaboration de tests subjectifs. C'est en 1934 que l'artiste suédois Oscar Reuterswäld dessine pour la première fois un triangle impossible (dessin encadré sur la figure 69).

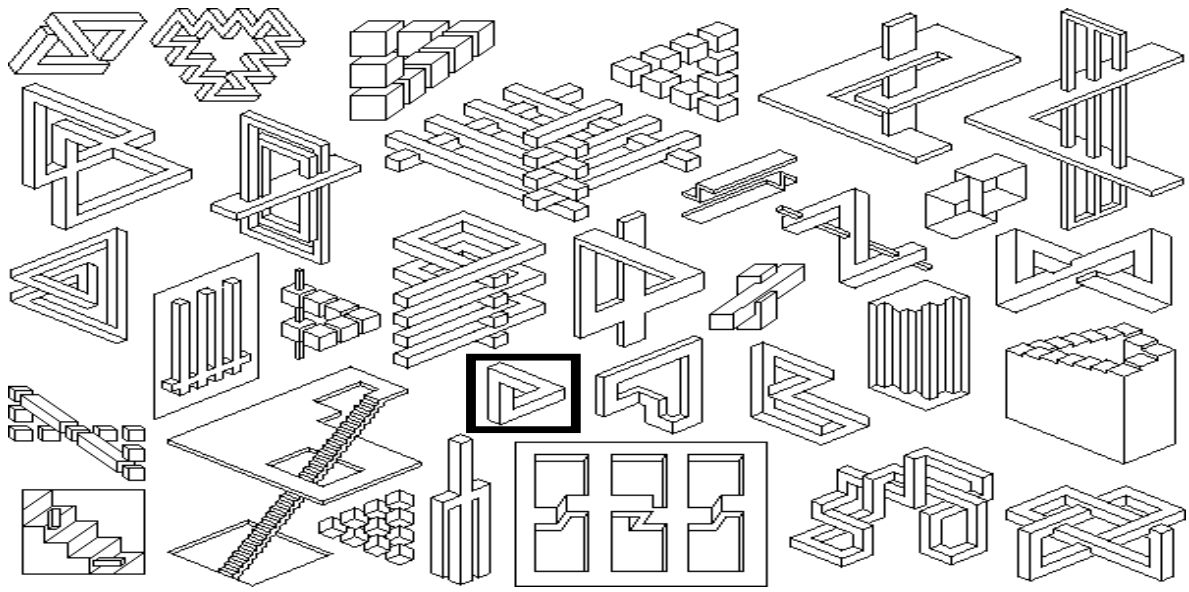


FIG. 69 – Figures impossibles d'Oscar Reuterswäld

Plus tard, M.C. Escher [3] exploite cet effet dans de nombreuses oeuvres. La figure 70 présente un exemple des travaux d'Escher où les mécanismes psychophysiques influencent notre perception : On associe le haut au ciel, et par conséquent, les poissons s'y trouvant sont considérés comme représentant le ciel, a contrario dans le bas, on associe les oiseaux à l'eau. Il nous semble effectivement plus naturel que les poissons nagent dans l'eau et que les oiseaux volent dans le ciel. Certains dessins paraissent ambiguës et peuvent être interprétés de différentes façons. Ces oeuvres mettent en valeur le fait que la vision est un processus actif, qui essaie d' extraire la sémantique de ce que nous voyons. Les modèles basés uniquement sur l'information de stimulus ne peuvent pas prédire de tels effets.

[3] http ://www.worldofescher.com/

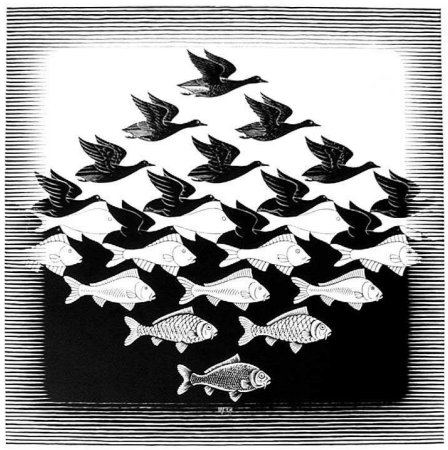


FIG. 70 – Sky & Water I, 1938

Un autre exemple est l'illusion de distortion (cf. figure 71). Ces images montrent qu'il existe un lien entre les processus bas-niveau et les processus haut-niveau. Nous n'allons pas donner une liste exhaustive des illusions d'optique, et pour plus de détails, le lecteur pourra se référer au site web "*Lightness Perception and Lightness Illusions*" [4] ainsi qu'à l'article de M. Livingston [105].

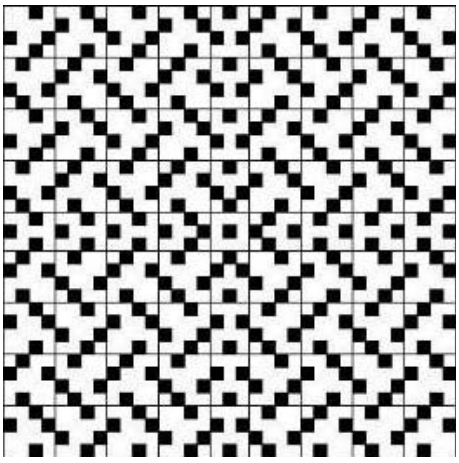


FIG. 71 – Les lignes paraissent courbes alors qu'en réalité elles sont droites

On peut donc voir dans cette section, que l'aspect psychologique joue un rôle important dans la perception visuelle et conduit à penser qu'elle reste subjective. Pour cette raison, arriver à modéliser efficacement les mécanismes de la perception est une tache difficile, empêchant d'aboutir à un modèle générique et parfait. Il nous apparaît à ce jour difficile d'inclure de telles

[4] http ://www-bcs.mit.edu/gaz/

considérations dans un schéma de tatouage.
En vidéo, nous retrouvons les mêmes caractéristiques que pour l'image fixe, mais comme nous allons le voir dans la section suivante, l'axe temporel joue un rôle important dans la perception visuelle.

4.2.3 Caractéristiques intrinsèques à la vidéo

Une vidéo peut être vue comme une succession d'images fixes, variant temporellement. Cependant, le fait que le système visuel humain soit limité dans la perception spatiale et temporelle, nous permet de suivre une vidéo dont la fréquence est de 25Hz sans s'apercevoir qu'elle ne représente qu'une suite discontinue d'images fixes. C'est pourquoi il apparaît naturel de considérer les propriétés de perception visuelle de l'image fixe pour le cas de la vidéo. Toutefois, il convient de ne pas négliger l'importance de l'axe temporel. Des artefacts invisibles sur une image figée peuvent devenir visibles et gênants lorsque que l'on ajoute la dimension temporelle. L'artefact peut alors apparaître sous la forme d'un clignotement. En contrepartie, lorsqu'une image dégradée apparaît brièvement (à savoir pour une vidéo de fréquence 25 Hz, l'image n'apparaît que 40ms), il a été constaté qu'un artefact pourra ne pas être perçu. C'est d'ailleurs une des caractéristiques propre à la vidéo. Il existe en outre un autre aspect temporel qui peut aussi bien être lié à l'image fixe qu'à la vidéo, il s'agit de la persistance rétinienne. Pour appréhender simplement ce phénomène, nous pouvons prendre un exemple de la vie courante. Lorsque nous sommes dans un environnement fortement éclairé et que nous passons dans un endroit sombre, notre perception initiale de l'environnement est limitée, puis avec le temps notre oeil s'accommode au faible éclairage, et nous pouvons dès lors distinguer cet environnement. Inversement lorsque nous passons d'un lieu sombre à un milieu fortement éclairé, nous sommes initialement ébloui. L'accommodation fait intervenir différents processus physiologiques, que nous ne décrirons pas ici, et pour plus de détails le lecteur pourra se référer à [136] et [108]. Il y a essentiellement trois formes de masquages temporels : celui dû au changement de scène, le masquage par le mouvement des objets et enfin celui modélisé par la fonction temporelle de sensibilité au contraste. Le masquage temporel dû au mouvement est complexe à modéliser : l'effet de masquage temporel d'un objet en mouvement dépend du fait que l'attention soit ou non focalisée sur cet objet. Un changement de scène intervient lorsqu'il y a un important changement dans tout le contenu de l'image. La fonction temporelle de sensibilité au contraste, quant à elle, peut être vue comme une extension de la fonction spatiale de sensibilité au contraste. Dans le cas qui nous intéresse, cet effet peut intervenir lorsqu'il y a des discontinuités fortes dans l'intensité lumineuse d'une séquence vidéo. Ce phénomène a tout d'abord été étudié dans le cadre de la télévision, par A.J. Seyler et Z.L. Budrikis [26] [27].

Ces auteurs ont établi que le seuil de visibilité pouvait durer quelques centaines de millisecondes, après une transition du sombre vers le lumineux ou inversement. D'autres travaux ont confirmé par la suite ce phénomène [35]. Il a été démontré par D.H. Kelly [51], [49], [50] que les distortions d'une période de 4 à 8Hz sont plus visibles. Plus récemment, W.J. Tam et al. [158] ont examiné la visibilité des artefacts de codage MPEG2 après un changement de scène ; il est apparu un effet de masquage significatif au niveau de la première image de la scène suivante. Il est intéressant de souligner que le masquage temporel n'apparaît pas seulement après une discontinuité, mais peut également se manifester antérieurement à celle-ci. Ce phénomène peut s'expliquer par la variation dans la latence des signaux neuronaux du système visuel humain [24].

Après avoir présenté les principales caractéristiques du système visuel humain, nous nous proposons maintenant d'examiner leur utilisation, tout d'abord dans les métriques de qualité puis au sein du tatouage.

4.3 Méthodes de mesure de la qualité d'une vidéo

4.3.1 Artefacts visuels

M.S. Moore et al. [114] proposent une étude de la détectabilité et de la nuisance d'artefacts pour une compression de type MPEG2.
Il existe en effet, différents types d'artefacts qui peuvent survenir dans une vidéo après une compression. Dans [110], M. Yuen & al. en présentent une liste exhaustive :

- l'effet de bloc est une distortion qui laisse apparaître une structure de blocs dans l'image, sous forme de discontinuités aux frontières, entre les blocs. Cet effet est dû à la quantification indépendante des blocs (de taille 8*8 par exemple) et/ou à une quantification trop forte, soit sur le coefficient DC, soit sur les coefficients AC (perte de détails hautes-fréquences), soit sur les deux. Cela arrive par exemple dans les schémas de codage basés sur la transformée DCT, appliquée sur une partition en blocs de l'image ;
- un effet de flou peut apparaître sous la forme d'une perte de détails et d'une réduction de l'intensité des contours. Ceci est dû à la suppression des coefficients de hautes fréquences par une quantification grossière ;
- des tâches de couleurs peuvent apparaître entre deux zones ayant une différence de chrominance importante. Ce phénomène est dû à la suppression des hautes fréquences des coefficients de chrominance. A cause du sous-échantillonage, ce phénomène peut s'étendre à tout un macrobloc ;
- le codage DCT peut aussi avoir pour conséquence de faire apparaître au niveau des lignes

de perspectives un effet d'escalier, ceci étant principalement dû au fait que la DCT est plus adaptée aux lignes verticales et horizontales. Les lignes avec d'autres orientations nécessitent les coefficients de hautes fréquences (ex : diagonales). Une quantification forte fera apparaître des lignes de perspectives en dent de scie ;

- apparition de faux contours dans l'image reconstruite. Cet artefact est introduit par une variation de l'intensité et de la couleur dans une zone de l'image où, à l'origine, les transitions ne sont pas brutales. Cette dégradation est très difficilement détectable. Néanmoins, elle peut être atténuée en filtrant l'image par un passe bas ;
- le mouvement en dent de scie, qui peut être dû aux faibles performances de l'estimateur de mouvement ;
- l'estimation de mouvement est réalisée sur la composante de luminance et ces mêmes vecteurs sont utilisés pour la chrominance. Cela peut conduire à une mauvaise correspondance de la couleur sur un macro bloc ;
- le bruit dit "*de Mosquito*". C'est un artefact temporel qui est dû à une erreur de quantification entre pixels adjacents. Ces dégradations sont introduites par la DCT et elles se manifestent par une dispersion des niveaux des ruptures d'homogénéité de l'image. Elles entraînent également un phénomène de flou. Ce bruit se produit surtout autour des objets artificiels, tels que le texte ou les formes géométriques, il est également visible sur les figures humaines ;
- aliasing : apparaît quand le contenu de la scène est sous-échantillonné (en dessous de la borne de Nyquist) que ce soit spatialement ou fréquentiellement.

Certains artefacts sont exclusivement liés à la structure en blocs de certains codecs vidéos, d'autres sont plus généraux. Enfin, certains artefacts sont moins gênants que d'autres, et ils dépendent étroitement de l'appréciation des utilisateurs (certains seront plus gênés par un blur et d'autres par les effets de blocs).

4.3.2 Métriques : généralités

Dans cette section, nous allons essentiellement nous concentrer sur les méthodes d'évaluation destinées à mesurer les effets de blocs. Ceux-ci sont dûs à une quantification trop importante des coefficients constituant les différents blocs de l'image partitionnée, système couramment utilisé en compression, tout comme dans notre schéma de tatouage. Les modèles perceptuels sont étudiés dans différents contextes. Tout d'abord, celui de la compression où le plus souvent le but est d'évaluer les effets de bloc. Ensuite, celui de l'indexation où les modèles perceptuels sont utilisés pour discriminer les textures, pour de la classification ou pour de *l'image retrieval*. Ces approches sont le plus souvent basées sur des métriques s'appliquant à la couleur ou aux textures, qui nous intéressent moins ici, dans la mesure où notre système de tatouage se base sur une estimation

de mouvement sur des blocs de luminance de taille 4*4. De par ces aspects, nous sommes plus proches du monde de la compression que du monde de l'indexation. Le lecteur intéressé pourra cependant se reporter aux références suivantes, traitant de métriques basées sur la couleur ou les textures [162], [96] et [11].

Lorsqu'il s'agit d'évaluer la qualité d'une image ou d'une vidéo, deux mesures universelles sont utilisées. Il s'agit des métriques MSE et $PSNR$. Ces deux formules mathématiquement liées sont employées couramment de par leur simplicité d'utilisation et de par la facilité d'implémentation. Ces deux méthodes sont des mesures pixel à pixel. La métrique MSE correspond à la différence moyenne quadratique entre la luminance d'une image I et $\widehat{I}$:

$$MSE = \frac{1}{TXY}\sum_{t}\sum_{x}\sum_{y}[I(t,x,y) - \widehat{I}(t,x,y)]^2 \tag{52}$$

Pour une image de taille $X * Y$ et pour une séquence composée de T images. La différence moyenne par pixel est alors donnée par la racine carrée de l'erreur quadratique moyenne : $RMSE = \sqrt{MSE}$. Le PSNR est une mesure donnée en décibels, et se calcule à partir de la mesure MSE, définie par la formule suivante :

$$PSNR = 10log\frac{m^2}{MSE} \tag{53}$$

avec m correspondant à la valeur maximale que peut prendre un pixel (typiquement pour un pixel codé sur 8 bits cette valeur vaut 255). La métrique MSE mesure la différence entre deux images, alors que le PSNR mesure la fidélité entre deux images. A.M. Eskicioglu et al. [28] présentent une étude de ces deux métriques statistiques ainsi que les métriques de différence moyenne, différence maximum, erreur absolue, etc. Il est montré dans cet article que certaines de ces métriques sont bien corrélées avec les réponses d'observateur humain, pour des systèmes de compression de données. Mais elles ne sont pas adaptées pour une évaluation de différentes techniques.

Il existe deux principaux types de modèles du système visuel humain :

- Les modèles simple-canaux considèrent le système visuel humain comme un simple filtre spatial, dont les caractéristiques sont définies par la fonction de sensibilité au contraste.
- Les modèles multi-canaux considèrent que chaque bande de fréquence spatiale est traitée par des canaux indépendants. A.B. Watson [13] introduit la "*cortex transform*", une pyramide multi-résolution qui simule les fréquences spatiales et qui correspond à l'orientation de simples cellules du cortex. A.B. Watson & al. [19], proposeront plus tard une décomposition alternative au premier modèle. Il s'agit d'une pyramide orientée orthogonalement qui s'applique à un réseau hexagonal.

Pour un historique plus complet sur les métriques psychovisuelles, le lecteur intéressé pourra se reporter à la thèse de S. Winkler [136].

4.3.3 Métriques pour la vidéo

Nous n'exposerons pas dans cette section une liste exhaustive des métriques existantes en vidéo, mais seulement les plus importantes d'entres elles. Nous ne présenterons pas non plus les métriques dédiées à l'image fixe, le lecteur intéressé pourra se référer à l'article d'A.J. Ahumada [23] qui propose un aperçu des différentes méthodes pour évaluer la qualité d'une image fixe et monochrome.

La plupart des travaux concernant l'optimisation des systèmes de compression, basés sur des considérations perceptuelles, sont majoritairement orientés vers la mesure des distortions spatiales. Cependant, devant la nécessité d'élaborer des métriques adaptées à la vidéo, de nouvelles métriques basées sur des modèles de perception spatio-temporelle du système visuel humain ont vu le jour.

T.N. Pappas & al. [151] présentent les critères objectifs basés sur les modèles perceptuels pour l'évaluation de la qualité d'une image fixe.

Beaucoup de métriques partagent la même structure : phase de calibration, filtrage linéaire pour différentes fréquences spatiales et pour différentes orientations, ajustage à la sensibilité du contraste, et mécanismes non-linéaires tenant compte des effets de masquage :

- calibration : une image peut avoir subi différentes transformations (*conversion to densities*, correction gamma, ...) et provenir de différents appareils, avant d'être affiché pour observation par l'oeil humain. De nombreuses métriques nécessitent que l'image d'entrée soit convertie en luminance avant de rentrer dans le modèle HVS.
- registration : c'est la correspondance point-à-point entre deux images, qui est nécessaire pour que la métrique ait un sens. Sinon, il est possible de changer la valeur de la métrique en effectuant une simple translation sur l'image, ce qui ne change pas pour autant l'image elle-même, mais modifie cependant la valeur de la métrique.
- l'affichage : différents matériels d'affichage peuvent provoquer différents effets sur la qualité perçue par un utilisateur. La métrique de Watson prend par exemple en compte la taille en pixel de l'image et la taille réelle en centimètre obtenue sur l'écran.

Dans [159], W.Y. Zou et al. exposent la méthodologie pour évaluer la qualité d'une vidéo, en décrivant le protocole à mettre en place pour des tests subjectifs. Ils en montrent les limitations, et proposent des améliorations pour ces tests. Enfin, ils présentent le lien entre les métriques subjectives et objectives. Les tests subjectifs jouent un rôle important dans les environnements de la télévision analogique et numérique. Cependant, l'évaluation subjective est très coûteuse en temps et en ressources (salle de tests normalisée, nombre important d'observateurs ...), ce qui a motivé le développement des métriques objectives, corrélées aux caractéristiques perceptuelles du système visuel humain, qui permettent d'automatiser les tâches d'évaluation de la qualité. Nous

ne nous étendrons pas ici sur les approches subjectives. Nous recherchons dans un premier temps à déterminer des algorithmes automatiques pour la prise en compte des aspects psychovisuels. En outre, la mise en place de tests subjectifs demande une infrastructure lourde et coûteuse, et cela nous apparaît un peu prématuré à ce stade de développement de notre algorithme.

Selon S. Wolf et al. [141], afin de couvrir un large panel d'applications, une métrique de qualité pour la vidéo doit :

- produire des résultats qui simulent des réponses subjectives ;
- fonctionner pour un large ensemble de qualité, du bas débit jusqu'au très haut-debit ;
- être performante et pouvoir être facilement implémentable sur un PC.

Dans un premier temps, nous allons exposer quelques métriques spatiales qui ont le plus souvent pour but de détecter les effets de blocs. Ensuite nous poursuivrons par la présentation de métriques prenant en compte l'axe temporel. Enfin, nous aborderons l'utilisation des aspects perceptifs au sein des algorithmes de tatouage.

Métriques spatiales

Le modèle de A.B. Watson pour la vidéo [18], [21], [14] est destiné aux systèmes de compression basés sur la transformation DCT. La métrique de Watson utilise la mesure de l'erreur perceptuelle basée sur la transformée DCT. C'est une métrique qui est une extension d'un système développé précédemment par les auteurs [17], [20], [15], [16]. L'erreur de quantification pour chaque coefficient de chaque bloc est pondérée par la sensibilité visuelle correspondante, pour chaque fonction DCT de base, pour chaque bloc. Cette sensibilité est déterminée par trois facteurs : sensibilité au contraste, masquage de la luminance, masquage de contraste.
La sensibilité au contraste est traitée de la façon suivante :
Le seuil de luminance t_{ij} de chaque fonction DCT de base a été mesuré expérimentalement par H.A. Peterson et al.[76]. A.J. Ahumada et al. [25] réalisent une approximation de ces mesures et les ont étendues en fournissant une formule qui permet l'extrapolation à différents types d'écran, à d'autres fréquences spatiales (différentes tailles de pixels, différentes distances de visualisation) ainsi qu'à différentes directions de couleur. Cette formule est la suivante :

$$log_{10}t_{ij} = log_{10}\frac{T_{min}}{r_{ij}} + k(log_{10}f_{ij} - log_{10}f_{min})^2 \tag{54}$$

avec :

$$r_{ij} = r + (1 - r)cos^2\theta_{ij} \tag{55}$$

où t_{ij} est le seuil de luminance de chaque fonction DCT de base, f_{min} est la fréquence où le seuil est le plus petit, T_{min} est la luminance moyenne au niveau f_{min}, k détermine la pente de la parabole, et θ_{ij} est un paramètre angulaire qui sera défini plus loin.
Les paramètres T_{min}, k et f_{min} sont donnés par les équations suivantes :

$$T_{min} = \begin{cases} \dfrac{L}{S_0}, \text{si } L > L_T \\ \dfrac{L}{S_0}(\dfrac{L_T}{L})^{1-a_t}, \text{si } L \leq L_T \end{cases} \tag{56}$$

où $L_T = 13.45cdm^2$, $S_0 = 94.7$ et $a_t = 0.649$.

$$k = \begin{cases} k_0, \text{si } L > L_k \\ k_0(\dfrac{L}{L_k})^{a_k}, \text{si } L \leq L_k \end{cases} \tag{57}$$

où $L_k = 300cdm^2$, $k_0 = 3.125$ et $a_k = 0.0706$.

$$f_{min} = \begin{cases} f_0, \text{si } L > L_f \\ f_o(\dfrac{L}{L_f})^{a_f}, \text{si } L \leq L_f \end{cases} \tag{58}$$

où $L_f = 300cdcm^2$, $f_0 = 6.78cyclesdeg$ et $a_f = 0.182$. La fréquence spatiale f_{ij} associée à la fonction de base $DCT(i,j)$ s'exprime de la façon suivante :

$$f_{ij} = \frac{1}{16}\sqrt{(I/W_x)^2 + (j/W_y)^2} \tag{59}$$

où W_x et W_y sont respectivement la taille horizontale et verticale d'un pixel en degré d'angle visuel. La magnitude de l'effet de sommation/obliquité, qui correspond à la sommation imparfaite des deux composantes de Fourier présentent dans les fonctions de bases, est déterminée par $0 < r < 1$ (la valeur recommandée est de 0.7) et le paramètre angulaire θ_{ij} est donné par :

$$\theta_{ij} = arcsin\frac{2.f_{i0}.f_{0j}}{{f_{ij}}^2} \tag{60}$$

L'effet de masquage de la luminance :

L'équation (61) décrit le seuil pour les fonctions de bases DCT comme une fonction de luminance moyenne d'un écran. Mais les variations de luminance moyenne locale dans l'image produisent

des variations non négligeables dans le seuil DCT. Ce phénomène est appelé masquage de la luminance. Watson a proposé d'approximer cet effet de masquage en calculant les seuils de luminance par bloc :

$$t_{ijk} = t_{ij}(\frac{c_{00k}}{\bar{c}_{00}})^{a_t} \tag{61}$$

où c_{00k} est le coefficient DC du bloc k, $\bar{c}_{00}$ la luminance moyenne de l'écran, et a_t représente le degré de masquage (la valeur recommandée étant de 0.65).

L'effet de masquage de contraste :
La visibilité d'un motif peut être réduite par la présence d'un autre motif au sein de l'image. Ce masquage est plus fort lorsque les deux composants sont de mêmes fréquences spatiales, de même orientation, et de même localisation. Selon le modèle de A.B. Watson, le seuil de masquage est défini par :

$$m_{ijk} = Max[t_{ijk}, |c_{ijk}|^{w_{ij}} {t_{ijk}}^1 - w_{ij}] \tag{62}$$

où m_{ijk} représente le seuil et w_{ij} le degré du masquage de contraste (A.B. Watson recommandant de prendre $w_{00} = 0$ et $w_{ij} = 0.7$ pour tous les autres coefficients).

L'erreur perceptuelle de chaque fréquence et de chaque bloc peut être exprimée de la façon suivante :

$$d_{ijk} = \frac{e_{ijk}}{m_{ijk}} \tag{63}$$

où e_{ijk} représente l'erreur de quantification. Pour obtenir l'erreur perceptuelle totale, il est nécessaire de regrouper les erreurs selon la fréquence et l'espace. Pour réaliser cette opération, il est possible d'utiliser la sommation de Minkowski :

$$d(i,\widetilde{i}) = \frac{1}{N^2}[\sum_{i,j}(\sum_{k} d_{ijk}^{\beta_s})^{\frac{\beta_f}{\beta_s}}]^{\frac{1}{\beta_f}} \tag{64}$$

A.B. Watson recommande de fixer $\beta_s = \beta_f = 4$.

Nous avons utilisé le modèle de Watson dans le but de localiser les zones marquées par notre algorithme de tatouage. Ce modèle calcule l'erreur perceptuelle en utilisant les facteurs de sensibilité au contraste, de masquage de luminance et de masquage de contraste. Pour ce faire nous utilisons une vidéo marquée et la vidéo originale correspondante, puis nous appliquons les principes du modèle de Watson afin d'extraire les zones marquées. Comme nous pouvons le voir sur

la figure 72, ce modèle peut nous permettre d'identifier les zones marqués, mais cela nécessite la vidéo originale. Cependant, cette métrique pourrait nous permettre d'élaborer une attaque efficace dont nous reparlerons au chapitre 5.

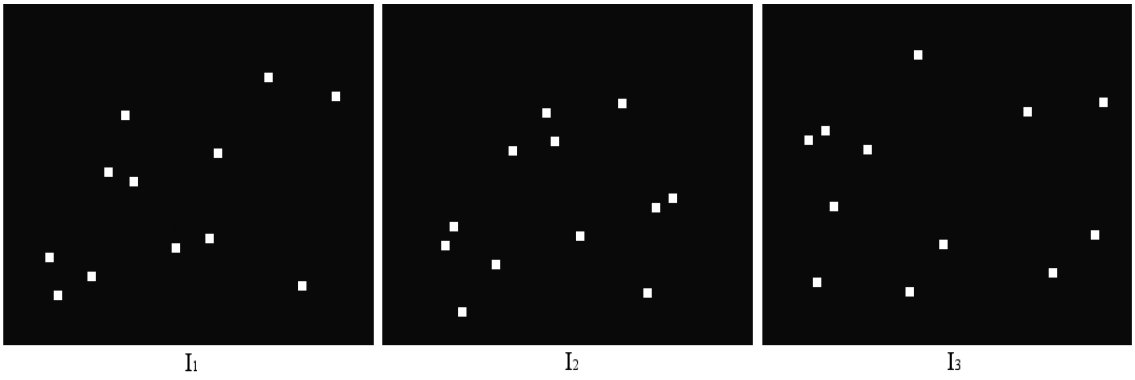


FIG. 72 – Utilisation du masque de Watson sur la détection des zones marquées. Les carrés blancs représentent les blocs détectés par le modèle de Watson et correspondent exactement aux localisations des blocs marqués par notre algorithme de tatouage

S. Winkler et al. [137] présentent, eux aussi, des métriques pour évaluer la qualité d'une vidéo, dans le contexte de schémas de compression basés blocs comme MPEG2. En compression, le principal artefact que l'on peut rencontrer est l'artefact de bloc. Visuellement, une structure en bloc peut apparaître au sein de l'image provoquant des discontinuités gênantes. Les schémas de compression tels que H.261, H.263, H.264, MPEG1, MPEG2, MPEG4 réalisent des DCT sur des blocs 8*8 et quantifient séparément chaque coefficient des différents blocs. Il existe certaines métriques qui prennent en compte cette structure de bloc : [142], [107], [166] , cependant ces techniques nécessitent d'avoir accès à la version originale de l'image ou de la vidéo. Dans cet article, l'auteur teste trois métriques basées bloc qui ne nécessitent pas la version originale du support testé.
La métrique de T. Vlachos [149] utilise un algorithme, basé sur une "cross-correlation" d'images sous-échantillonnées. La structure d'échantillonnage est déterminée de façon à ce que chaque sous-image contienne un pixel spécifique des blocs 8*8. Quatre sous-images, correspondants aux pixels des quatre coins de chaque bloc, sont générées. Quatre sous-images supplémentaires sont générées à partir des quatre pixels voisins du coin supérieur gauche de chaque bloc. Enfin, la "cross-correlation" du premier ensemble de sous-images est normalisée par la "cross-correlation" du second ensemble de sous-images pour donner une mesure de l'effet de bloc.
La métrique de Wang-Bovik-Evans [165] modélise l'image contenant les effets de blocs par une image sans ces effets, en interférant celle-ci par un signal bloc pur. Ils appliquent une FFT 1D sur le signal en colonne et en ligne représentant la différence entre les deux éléments cités ci-dessus.

Cela donne une estimation du spectre de puissance moyen horizontal et vertical. Les pics de ce spectre dûs à la structure en bloc 8*8 sont identifiés par leur localisation fréquentielle. Le spectre de puissance de l'image sans effet de bloc est approximé en appliquant un filtre médian sur la courbe précédente. La mesure des effets de bloc est alors calculée en effectuant la différence entre ces spectres de puissance. La figure 73 présente un résumé de cette métrique.

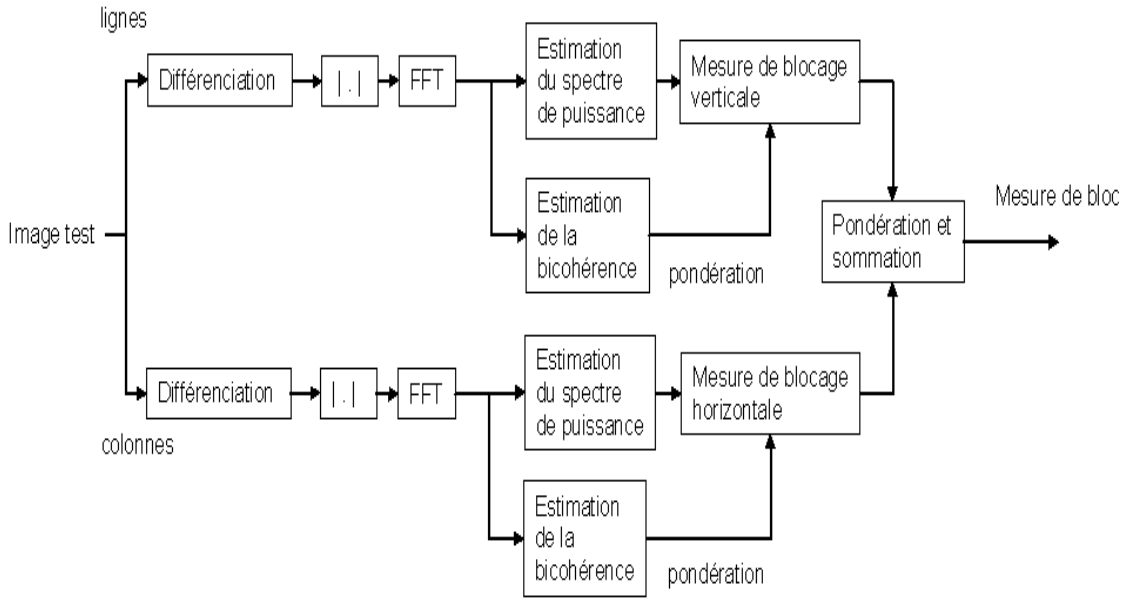


FIG. 73 – Métrique de Wang-Bovik-Evans

La métrique de Wu-Yuen [77] mesure la différence horizontale et verticale entre les lignes et les colonnes aux frontières de chaque bloc 8*8. Une pondération prenant en compte les effets de masquage de luminance et de textures est dérivée de la déviation standard et moyenne des blocs adjacents à chaque frontière. La mesure résultante est normalisée par la même mesure moyenne calculée sur les lignes et colonnes internes.
S. Winkler conclut que toutes ces métriques ne prennent pas en compte l'aspect temporel de la vidéo puisqu'elles s'appliquent image par image, mais qu'elles représentent un grand pas vers une mesure plus élaborée.

La métrique de Wu-Yuen [77] a été améliorée par la suite, par S. Suthaharan dans [132]. L'auteur prend en compte les caractéristiques du système visuel humain, en se basant sur la mesure de distortion L_∞, ce qui permet d'aboutir à une métrique plus performante que les métriques GBIM (*Generalized Block-edge Impairment Metric*) et IBIM (*Improved Block-edge Impairment Metric*).

Dans [133], Suthaharan a récemment présenté une métrique de distorsion pour la vidéo dénommée PS-BIM (*Perceptually Significant Block-edge Impairment Metric*). Cette métrique utilise les effets de masquage de la luminance, et la différence visible des blocs dans les régions perceptuelles de l'image.

Dans [142], S.A. Karunasekera et al. présentent une métrique pour les effets de blocs dans une image compressée. Le modèle se base sur la sensibilité du système visuel humain aux artefacts de contours horizontaux et verticaux.

D'autres métriques évaluent des artefacts différents des effets de bloc. C'est le cas de celle présentée par P. Marziliano & al. [121]. Cette métrique sans référence, est basée sur l'analyse des contours, afin d'évaluer les dégradations dues au flou. Cette métrique est validée par des tests subjectifs.

Métriques spatio-temporelles

Devant la nécessité de développer des métriques dédiées à la vidéo, il apparaît naturel de prendre en compte l'axe temporel dans leur élaboration.

Dans [139] et [3], S. wolf et al. proposent un système de mesure objective basé sur la perception humaine, pour la vidéo et les signaux télévisuels. Leur système est composé de deux sous-systèmes, un pour la source originale et un pour la vidéo dégradée. En extrayant un ensemble de caractéristiques qui peuvent être utilisées pour prédire les changements perceptuels de la qualité de la vidéo, une estimation objective de la qualité du système peut être établie en comparant les caractéristiques de la source et de la vidéo dégradée. Le système est basé à la fois sur des caractéristiques spatiales et temporelles en utilisant un filtre de Sobel pour le domaine spatial et l'image de différences de mouvement pour l'analyse de l'axe temporel. Afin d'obtenir une valeur unique pour la qualité, une formule composée d'un ensemble de caractéristiques spatiales et temporelles est déterminée en utilisant deux tests subjectifs [140]. L'architecture générale du système est présentée sur la figure 74.

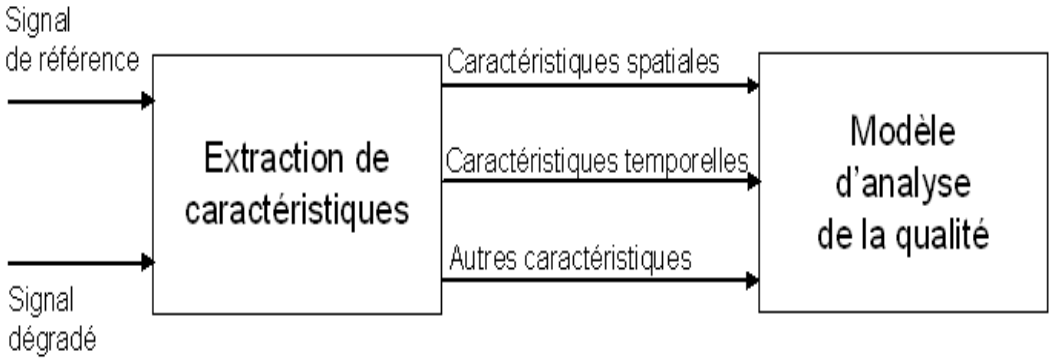


FIG. 74 – Métrique ITS

Cette métrique présente cependant des limitations, comme l'ont présenté A. Basso et al [5], dans le cadre de l'étude du codec MPEG2. Cette métrique est destinée aux séquences de faibles débits, qui ont généralement plus d'artefacts que les séquences MPEG2. Elle n'est pas capable de bien détecter les artefacts dûs au codage DCT, comme les effets de blocs et le bruit de mosquito. De ce fait, ils présentent leur métrique (MPQM : *Moving Pictures Quality Metric*) qui est mieux adaptée au codage MPEG2.

Par la suite, S. Wolf et al. [141] ont pris en compte la remarque de l'article de A. Basso sur les limitations de la métrique ITS.
Ils présentent un résumé du modèle exposé précédemment. Un aperçu de leur métrique est proposé sur la figure 75. Seule la luminance est traitée. Ils réalisent dans un premier temps un filtrage horizontal et vertical des contours. Le résultat de ces filtrages est ensuite divisé en régions spatio-temporelles, desquelles sont extraites les caractéristiques de la séquence. L'activité spatiale est donc quantifiée comme une fonction d'orientation angulaire. Les processus perceptuels sont alors appliqués sur ces caractéristiques, pour enfin être regroupés, afin d'obtenir une mesure de la distorsion pour chaque région spatio-temporelle. Ces mesures sont ensuite regroupées afin d'obtenir une mesure de la qualité de la vidéo.

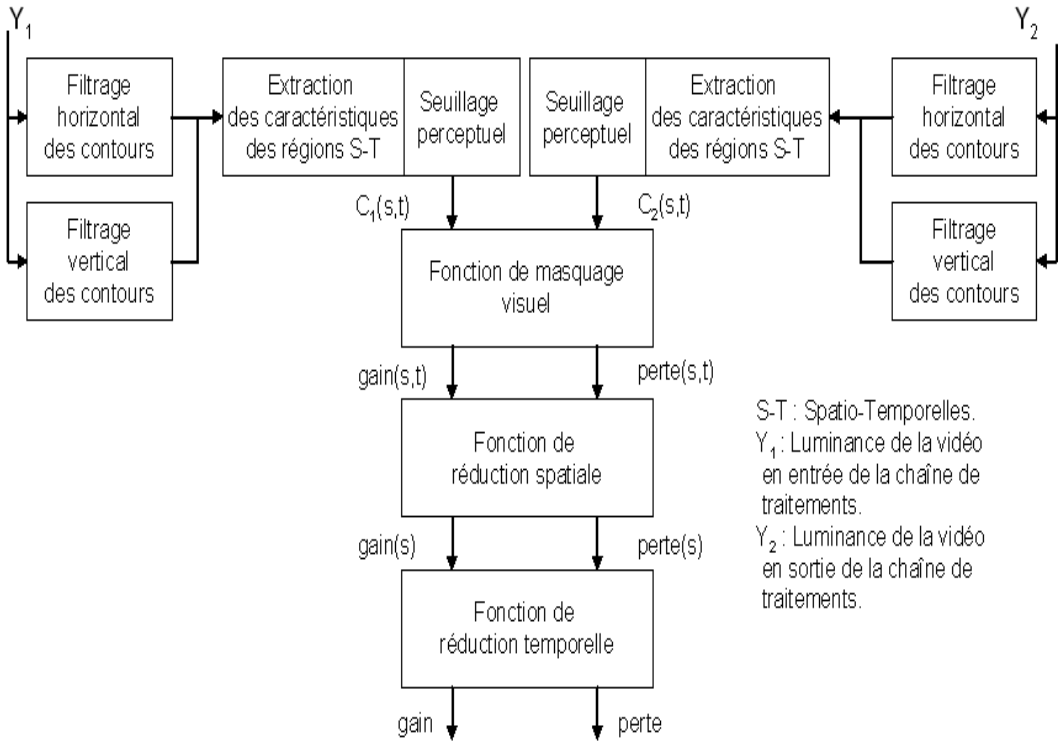


FIG. 75 – Métrique de déformation spatio-temporelle pour les services de surveillance de la qualité de système vidéo numérique

S. Winkler [135] propose une métrique pour la vidéo, qui est une extension de la métrique développée dans [135] pour l'image fixe, celle-ci étant basée sur les travaux de C.J. Van Den Branden Lambrecht [119], [42]. La structure générale de ce système est présentée sur la figure 76.

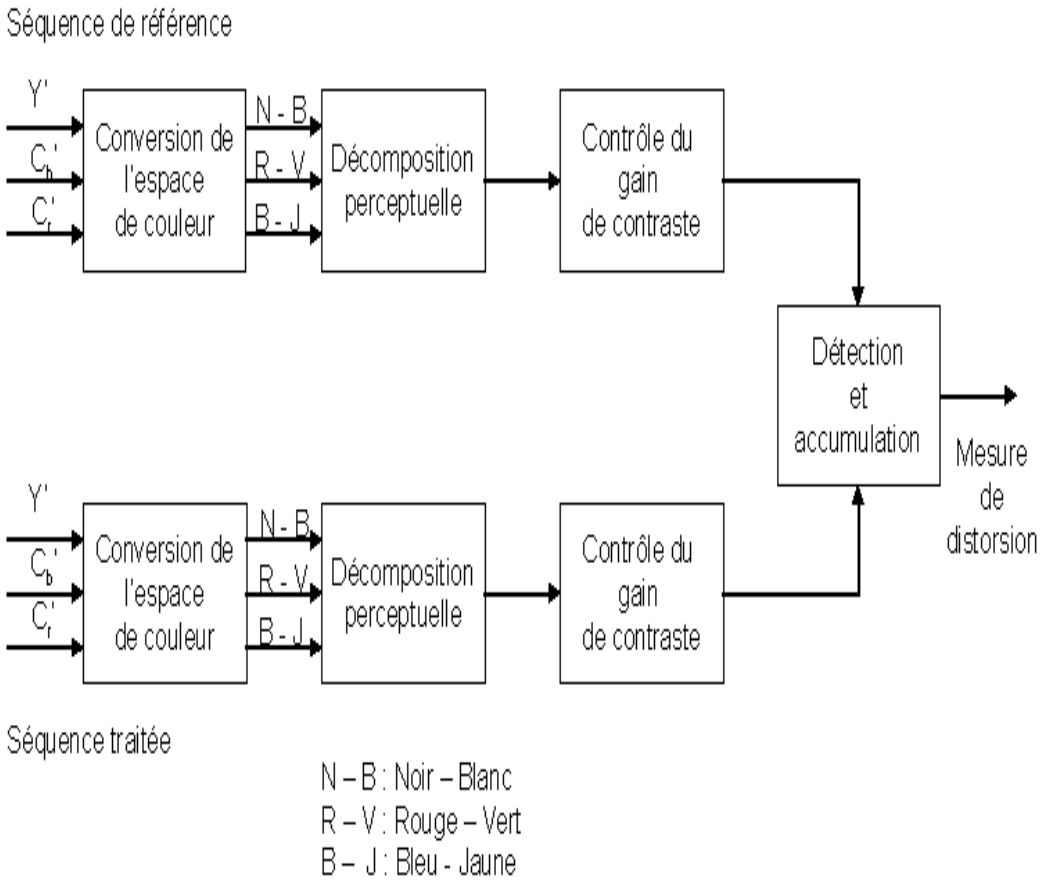


FIG. 76 – Schéma de la métrique de Winkler

Les séquences sont tout d'abord transformées dans un espace de couleurs opposées. Une décomposition spatio-temporelle est alors appliquée. Le résultat est ensuite pondéré en fonction de la sensibilité au contraste. Enfin les résultats sont combinés dans une mesure de la distorsion (pour de plus amples détails, le lecteur pourra se référer à l'article de Winkler cité ci-dessus).

Dans [99], K.T. Tan et al. proposent un modèle de mesures objectives en deux étapes, pour les vidéos codées en respectant le standard MPEG2. Ils réalisent premièrement une pondération de la distorsion de la vidéo codée, en fonction de la réponse du système visuel humain. La dégradation perceptuelle trame par trame dans l'image décodée vis-à-vis d'une image de référence est calculée. Ce calcul inclut un filtrage spatial passe-bas, un filtrage de Sobel pour dériver les coefficients de masquage, et un masquage spatial sur l'erreur brute entre images de référence et images compressées. Ensuite, l'émulation cognitive prend place au sein d'une simulation du traitement

haut-niveau de l'information visuelle. Ceci inclut la réponse temporelle, très faible, des observateurs humains aux changements de qualité d'image, et un comportement asymétrique vis-à-vis des changements de qualité d'image, de mauvais à bon, et vice et versa. Avec ce modèle, les auteurs ont été capables de modéliser très précisément la qualité subjective variant dans le temps, de séquences vidéos enregistrées avec la méthode ITU-RSSCQE.

Dans [143], S.J.P. Westen et al. proposent un nouveau modèle pour la prédiction des distortions visibles dans une séquence d'images. Ce modèle est une extension d'un modèle spatial avec une fonction de sensibilité spatio-temporelle au contraste, ainsi qu'une estimation du mouvement de l'oeil. Selon cet article, le mouvement de l'oeil a une grande importance dans la sensibilité spatio-temporelle du système visuel humain. Cette importance a notamment été démontrée par B. Girod dans [30]. La structure générale de ce système est présentée sur la figure 77.
Ce modèle prend en entrée la séquence originale ainsi que la séquence décodée. Le signal passe ensuite dans un filtre FSC, afin de prendre en compte le mouvement de l'oeil. Ce signal filtré subit alors une compensation de mouvement. Le signal est ensuite décomposé en bandes de fréquences spatiales et en orientations. Pour chaque bande de fréquence et d'orientations, les auteurs appliquent un convertisseur qui permet de modéliser l'effet de masquage au niveau des bandes de fréquences et des orientations. Enfin, les différences entre les réponses des bandes de fréquences et des orientations sont combinées pour obtenir une mesure locale des distortions.

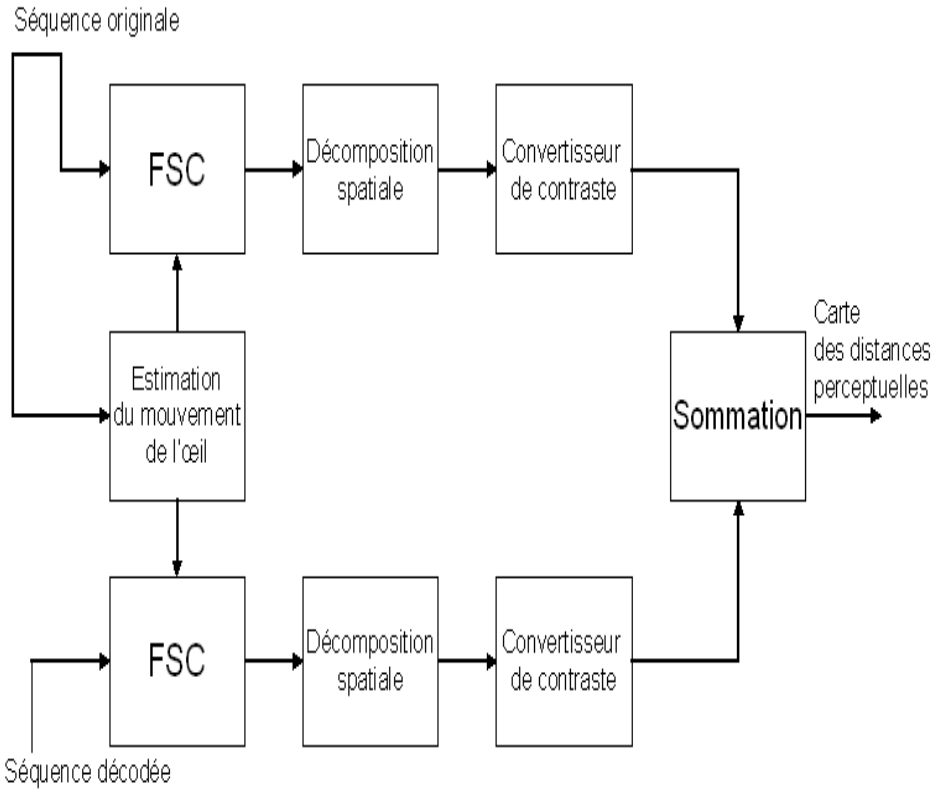


FIG. 77 – Modèle spatio-temporel du système visuel humain pour la compression vidéo

sensisibilité spatio-temporel au contraste :

$$C(x,y,t) = \frac{A.E(x,y,t)}{k + I(x,y,t)} \tag{65}$$

Les termes k et A sont des constantes. Les termes $E(x,y,t)$ et $I(x,y,t)$ sont respectivement les termes d'excitation et d'inhibition, donnés par les équations suivantes :

$$E(x,y,t) = R_e(x,y,t) \bigotimes L(x,y,t) \tag{66}$$

et

$$I(x,y,t) = R_i(x,y,t) \bigotimes L(x,y,t) \tag{67}$$

où $R_e(x,y,t)$ est la réponse impulsionnelle d'excitation et $R_i(x,y,t)$ la réponse impulsionnelle d'inhibition. $L(x,y,t)$ correspond à la luminance de l'écran au pixel de position (x,y) à l'instant t et $a \bigotimes b$ correspond au produit de convolution. Les fonctions de réponse impulsionnelle sont données par :

$$R_e(x,y,t) = U_e(x,y,t).V_e(t) \tag{68}$$

et

$$R_i(x, y, t) = U_i(x, y, t).V_i(t) \tag{69}$$

où $U_e(x, y, t)$ et $U_i(x, y, t)$ sont des fonctions de diffusion spatiale de points. $V_e(t)$ et $V_i(t)$ sont des fonctions de réponses impulsionnelles temporelles. Les fonctions de diffusion spatiale de points sont des gaussiennes. Les réponses impulsionnelles sont des fonctions exponentielles du second ordre.
Il apparaît donc important de prendre en compte les aspects temporels tant au niveau des mouvements de l'oeil qu'au niveau de la vidéo. En effet, un simple modèle spatial ne suffira pas à limiter les effets de clignotement. Cependant, comme nous pouvons le voir dans cet article, ce type de modèle représente un premier pas vers l'élaboration de modèles plus complexes et mieux adaptés à la vidéo.

Dans [90] J. Lubin présente le modèle de Sarnoff pour déterminer le JND (*Just Noticeable Difference*). Le modèle général est décrit dans cet article, ainsi qu'une étude des performances du système pour différentes applications vidéo.

Même si les aspects de la perception visuelle ne sont pas encore parfaitement maîtrisés, certains travaux de recherche comme ceux effectués par S. Winkler aboutissent à une maturité suffisante pour une commercialisation de produit d'évaluation de la qualité relativement fiable, comme les solutions proposées par la société Genista [5] qui se basent en partie sur les travaux de S. Winkler [136], [135], [137], [121] et [138]. En effet, ce dernier a proposé dans ces différentes publications des études approfondies du système visuel humain et la mise en oeuvre de ses caractéristiques dans le cadre de méthodes d'évaluation de la qualité efficaces. En outre, il a proposé des évaluations de différentes méthodes afin de pouvoir comparer les méthodes subjectives et objectives et ainsi fournir des solutions mieux adaptées à l'évaluation de la qualité.

4.3.4 Les modèles perceptuels en tatouage

Les métriques d'évaluation de la qualité ont récemment été utilisées afin de réaliser de la steganalyse, technique consistant à extraire ou à détecter une marque dans un medium, sans disposer du détecteur et ce, dans le but de passer outre le schéma de tatouage, en brouillant ou en extrayant la marque du support multimedia. Dans [79] et [78], I. Avcibas et al. utilisent des métriques psychovisuelles afin de distinguer les images tatouées des images non-tatouées et également de

[5] www.genista.com

distinguer différentes techniques de tatouage.

Certains auteurs considèrent qu'un schéma de tatouage, pour être efficace, doit insérer les informations de tatouage dans les composantes perceptuelles du medium, sans pour autant laisser apparaître des distortions trop visibles. La problématique qui se pose alors est assez complexe. Dans [82], I.J. Cox et al. insistent sur la nécessité d'utiliser les aspects psychovisuels pour optimiser les schémas de tatouage.
Les méthodes les plus couramment utilisées sont le PSNR et le wPSNR (PSNR pondéré) pour mesurer la similarité. D'autres approches visent à calculer un seuil (JND : Just Noticeable Difference) qui permet de déterminer la quantité maximum de distortion qu'il est possible d'introduire, sans que cela soit perceptuellement visible.
On peut adopter deux positions vis-à-vis du problème de la visibilité. La première approche vise à établir un contrôle a priori. Dans ce cas, on détermine le seuil JND qui va définir la limite à ne pas dépasser pour rester invisible. L'autre approche cherche à réaliser un contrôle a posteriori de la qualité de la vidéo, afin de déterminer si les dégradations apportées par le schéma de tatouage sont perceptibles ou non.
Actuellement, le PSNR est souvent utilisé comme critère perceptuel afin de minimiser l'impact d'un système de tatouage. Cependant le contrôle se fait en général a priori, c'est à dire que l'utilisation des aspects psychovisuels est réalisée préalablement à l'insertion de la marque, afin de déterminer un facteur (qui représente la force de tatouage) qui va permettre de moduler l'impact du marquage.

Dans [127]et [37], les auteurs abordent la problématique de la perceptibilité des systèmes de marquage. Le principe est d'optimiser le compromis invisibilité/robustesse, en augmentant la force du marquage dans les zones moins sensibles (classiquement les zones texturées), et en diminuant la force dans les zones sensibles (classiquement, ce sont les contours et les zones homogènes). Pour arriver à cette pondération du marquage, le contenu de l'image est souvent classé en trois zones : textures, contours et zones homogènes. Ensuite, pour chaque zone un seuil est établi, correspondant au niveau de perceptibilité du schéma de marquage. Dans le cas de la vidéo, il faut prendre en compte l'information de mouvement pour établir un masque. Dans ce contexte, un autre type d'artefact peut survenir : la distorsion temporelle, qui le plus souvent a pour conséquence de faire apparaître des clignotements ou des effets de dérives. Pour éviter ce genre de problème, une première solution consiste à séparer les zones dynamiques (ou à fort mouvement), des zones statiques (ou à mouvement faible) et d'adopter une politique de marquage différente pour ces deux zones. Le problème de la perception visuelle (en comparaison à la perception auditive) est qu'il

n'existe pas de vrais modèles mathématiques permettant de s'adapter au mieux aux propriétés du système visuel humain, ce qui complique la détermination des seuils de perception.

Dans notre contexte, il paraît difficile d'utiliser les principes de ces approches, car les artefacts visibles générés par notre système de tatouage sont essentiellement temporels et provoquent des clignotements. Une des particularités de notre approche est que les distortions perceptibles ne sont pas directement liées aux variations que l'on applique sur les vecteurs de mouvements, par conséquent il est inutile d'établir le JND, et préférable de réaliser un contrôle a posteriori. En outre, il est nécessaire de se baser sur des approches sans référence afin d'avoir un schéma de tatouage qui ne soit pas privé (afin que la vidéo originale ne soit pas nécessaire lors de la phase de détection). Enfin, nous avons vu que la méthode de watson permet d'extraire les blocs qui ont été marqué, mais le problème essentiel qui se pose à nous est l'effet temporel de clignotement qui nécessite une analyse du mouvement (global et locale) afin d'adapter la stratégie d'insertion au mouvement de la séquence. Nous verrons cependant dans les perspectives qu'il est envisageable d'utiliser le modèle de watson afin de réaliser une attaque ciblée de notre algorithme.

4.4 Solution proposée

Dans cette partie, nous allons présenter les différents travaux que nous avons effectués sur l'invisibilité de notre approche, décrite dans le chapitre 3. L'objectif de ce chapitre concerne l'élaboration d'un masque prenant en compte des critères de qualité, afin de diminuer les distorsions créées par notre algorithme. Les artefacts auxquels nous sommes confrontés sont tout d'abord les effets de blocs, classiques en compression, qui sont dûs ici à la structure de notre estimateur de mouvement qui réalise ses calculs sur des blocs de taille $4 * 4$. Le second type d'artefact auquel nous sommes confrontés est un effet de clignotement qui survient au niveau des blocs marqués. Le principal problème qui se pose dans la conception d'un masque est qu'il doit être le même à l'insertion et à la détection, par conséquent, il est impératif d'utiliser un modèle sans référence (c'est à dire qui ne nécessite pas d'avoir la vidéo originale pour déterminer le critère de qualité nous servant à la construction du masque, et ce afin de rester dans le cadre d'un algorithme de tatouage semi-aveugle). Un autre problème, corrélé au premier, est qu'en tatouage, entre la phase d'insertion et de détection, la vidéo peut avoir subi différents traitements, qui peuvent conduire à une mauvaise détection.
Nous avons travaillé en premier lieu sur la suppression des effets de blocs. Pour cela, nous avons mis en place un masque basé sur un critère de PSNR. Il a été exposé précédemment que ce critère n'est pas très adéquat d'un point de vue perceptuel, Winkler [138] a même récemment démontré

la faible corrélation du PSNR vis à vis des procédures subjectives. En outre, ce critère ne permet pas de prendre en compte l'aspect temporel d'une vidéo. Cependant, il représente une première étape intéressante, par sa facilité d'utilisation. Afin d'obtenir le même masque à l'insertion et à la détection, différentes approches ont été testées. Dans un premier temps, le masque est calculé sur la version tatouée de la vidéo, un processus itératif permet la convergence vers un masque stable pour les deux phases. Dans un second temps, nous avons réalisé différents préfiltrages avant de calculer le masque, les filtres utilisés étant les suivants : blur gaussien (avec un noyau de taille $5 * 5$), sobel (combinaison de deux filtres de sobel, vertical et horizontal, pour obtenir un gradient de l'image), Min, et Max. Enfin, nous nous sommes orientés vers un réglage du seuil automatique basé sur des caractéristiques locales, nous servant à pondérer le calcul du PSNR. Les méthodes servant à déterminer la pondération et le seuil peuvent varier : mesure de la luminance moyenne locale, mesure de la variance, mesure de contraste, gradient..., nous n'avons testé ici que la mesure de contraste, nous présenterons brièvement les autres approches dans le dernier chapitre.
Les systèmes de tatouage actuels réalisent en général le contrôle de qualité a priori, c'est à dire que l'utilisation des aspects psychovisuels se fait préalablement à l'insertion de la marque, afin de déterminer un facteur (qui représente la force de tatouage) qui permettra de moduler l'impact du marquage. Dans notre cas, le contrôle se fait a posteriori. Ce contrôle nous permet d'obtenir une bonne synchronisation du masque entre la phase d'insertion et de détection. Il serait toutefois préférable d'adopter un système hybride, combinant notre approche pour les zones à fort mouvement, et une approche d'images fixes pour les zones statiques ou à faibles mouvements.

4.4.1 Mise en place d'un masque basé sur un critère de PSNR

Nous allons maintenant décrire le masque que nous avons défini précédemment. Le calcul de ce masque intervient après avoir simulé une première insertion. Nous réalisons ensuite un processus itératif, qui permettra de déterminer le masque afin de ne garder que les blocs dont le PSNR sera supérieur à un seuil donné. La figure 78 présente plus en détails le procédé d'insertion avec la prise en compte du masque.

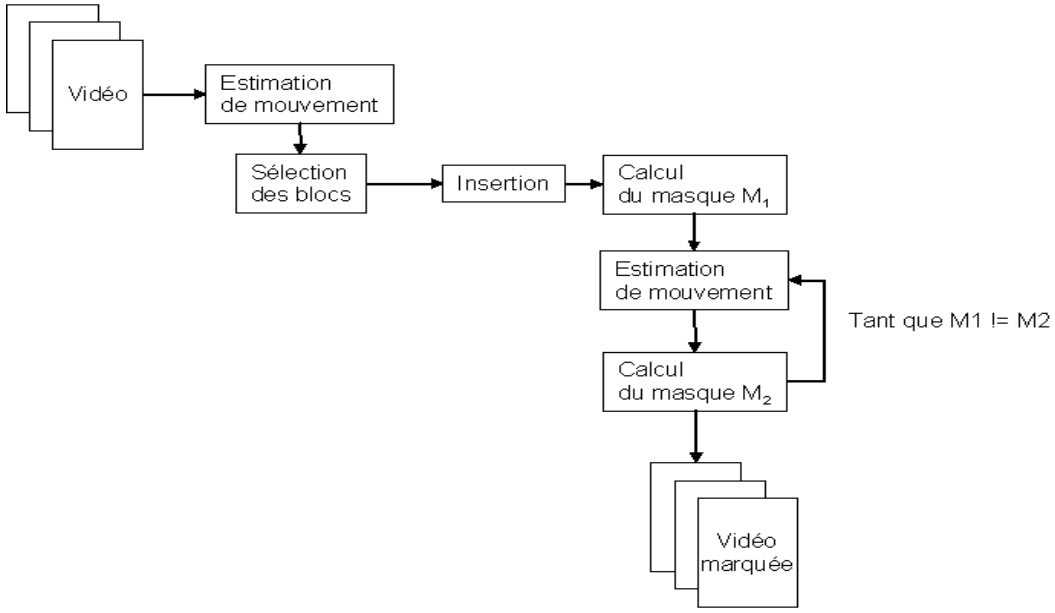


FIG. 78 – Insertion d'une marque avec la prise en compte d'un masque basé sur un critère de PSNR

Dans nos expériences, nous avons utilisé un seuil égal à 20 associé à une sélection pseudo-aléatoire de 12 blocs. L'expérience nous a montré qu'au delà de ce seuil on ne conservait pas suffisamment de blocs pour que l'algorithme converge rapidement vers la bonne marque, cependant sur des vidéos plus longue, il est envisageable de prendre un seuil plus élevé. En dessous de ce seuil, l'augmentation du PSNR était trop faible. Ce seuil peut paraître faible, cependant sur des blocs 8×8, l'impact d'un pixel erroné est plus forte.
Sur la figure 79, nous présentons le principe du calcul du masque.

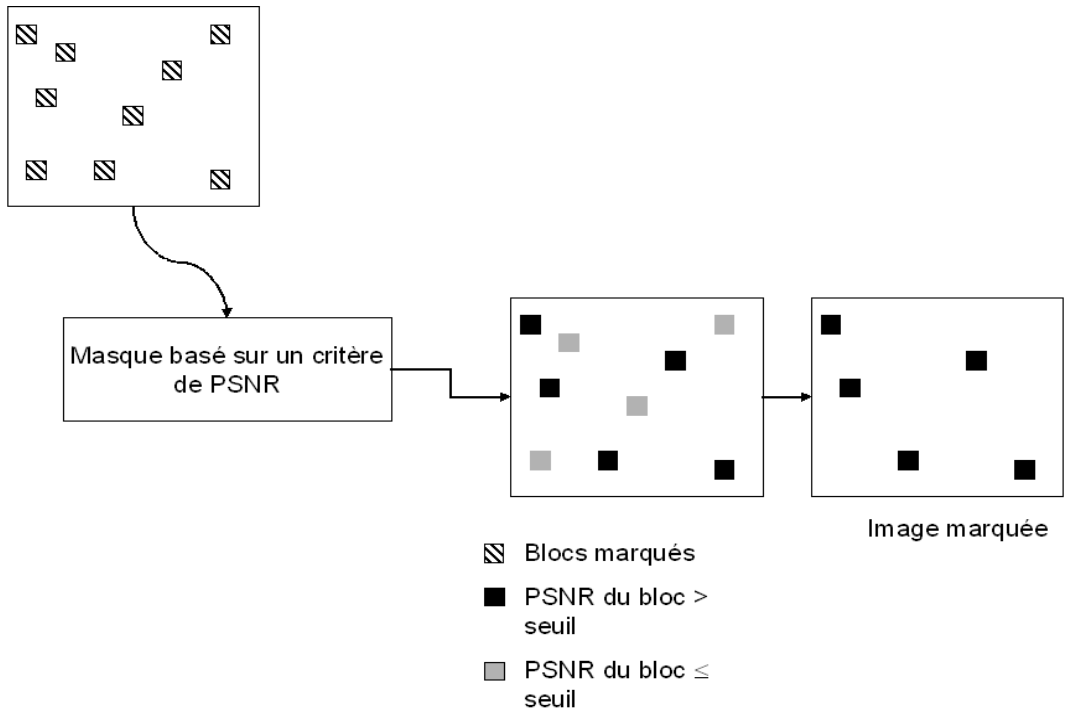
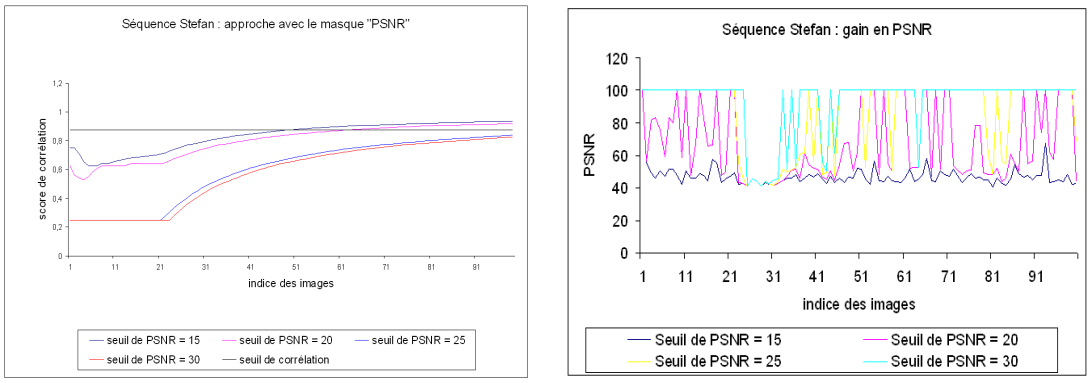


FIG. 79 – Détermination du masque basé sur un critère de PSNR

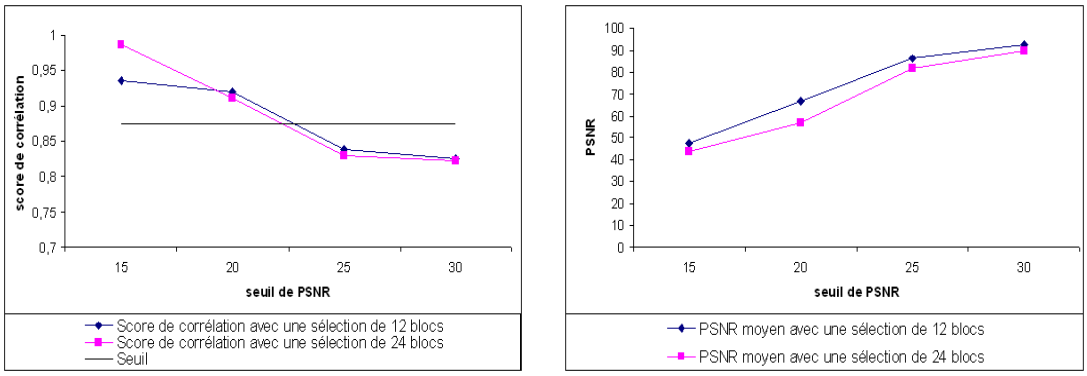
La figure 80 (b) montre l'évolution du PSNR suivant le seuil choisi. La figure 80 (a) montre quand à elle les scores de corrélation obtenus avec les différents seuils sur la séquence "Stefan" marquées et sans attaque.



(a) (b)

FIG. 80 – (a) Evolution du score de corrélation en fonction du seuil de PSNR choisi (b) Evolution du PSNR en fonction du seuil choisi

Comme nous pouvons le voir sur ces différentes courbes, le PSNR évolue inversement à la robustesse (i.e. à la vitesse de convergence de la détection). Il apparaît qu'un seuil de 20 présente un bon compromis entre invisibilité et robustesse. En effet, lorsque l'on veut une vitesse élevée de convergence afin de pouvoir détecter la marque sur des séquences ne faisant que quelques secondes, il est nécessaire de sélectionner suffisamment de bloc. En revanche, sur des vidéos plus longue, la vitesse de convergence ne doit pas être aussi rapide, ce critère nous permet d'envisager l'utilisation de seuils plus élevés. Une étude plus poussée permettrait sans doute d'affiner ce seuil. Dans les résultats, concernant le PSNR, certaines valeurs ont été mise artificiellement à 100. Cela correspond aux cas où le PSNR est infini. Dans ce cas, aucun bloc n'est marqué dans l'image traitée et par conséquent, l'image marquée correspond à l'image originale.
Sur les figures 81 (a) et 81 (b), nous présentons une comparaison de cette approche en fonction du nombre de bloc utilisés pour la sélection. Cette comparaison se fait sur la séquence "Stefan" comportant 100 images. Comme nous pouvons le voir, en augmentant le nombre de blocs potentiellement marqués, nous avons une faible diminution du PSNR, ce qui est logique. En revanche, nous pouvons noter que les scores de corrélation ne sont pas toujours supérieurs lorsque l'on augmente le nombre de blocs. Nous n'avons pas à ce jour d'explication à ce comportement illogique.



(a) (b)

FIG. 81 – (a) Evolution du score de corrélation en fonction du seuil de PSNR choisi et du nombre de blocs sélectionnés (b) Evolution du PSNR en fonction du seuil choisi et du nombre de blocs sélectionnés

Sur les figures 82 (a), 82 (b), 82 (c), et 82 (d), nous présentons les résultats de détection pour les séquences "Stefan" et "Ping-pong". Nous comparons les résultats de l'approche avec la prise en compte du masque et l'approche adaptive présentée au chapitre 3.

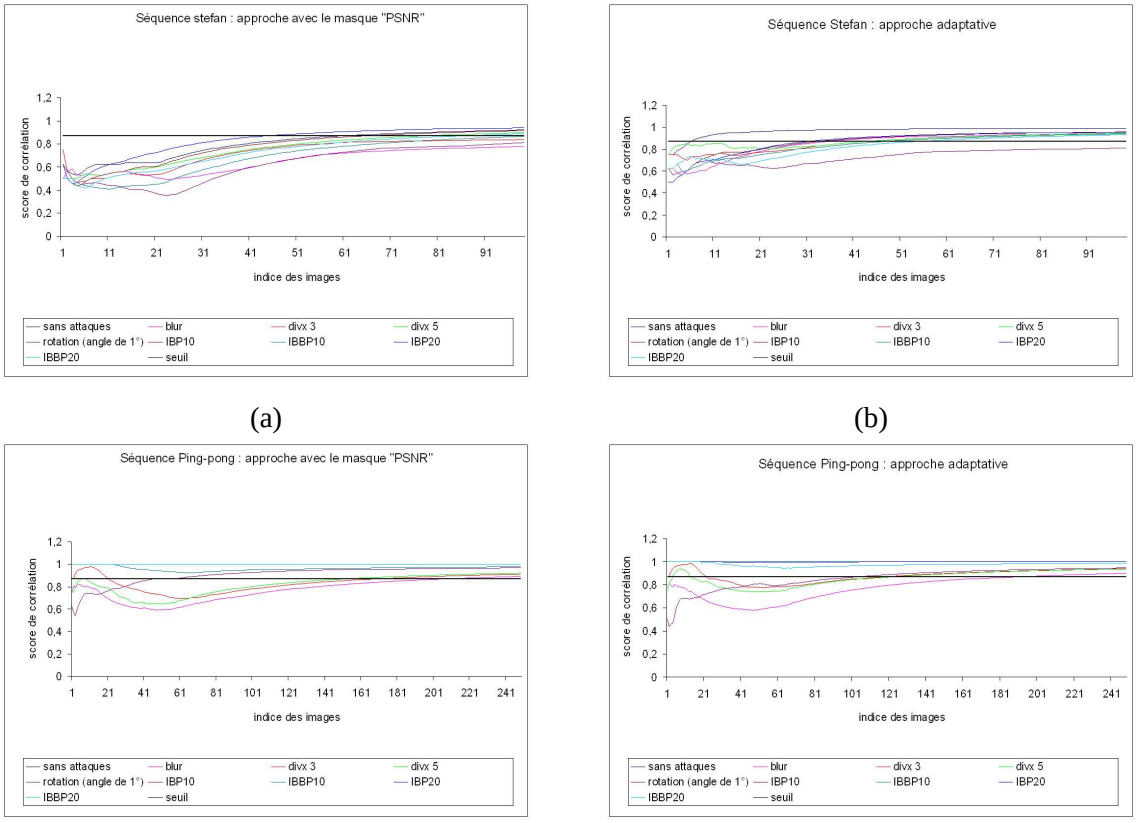
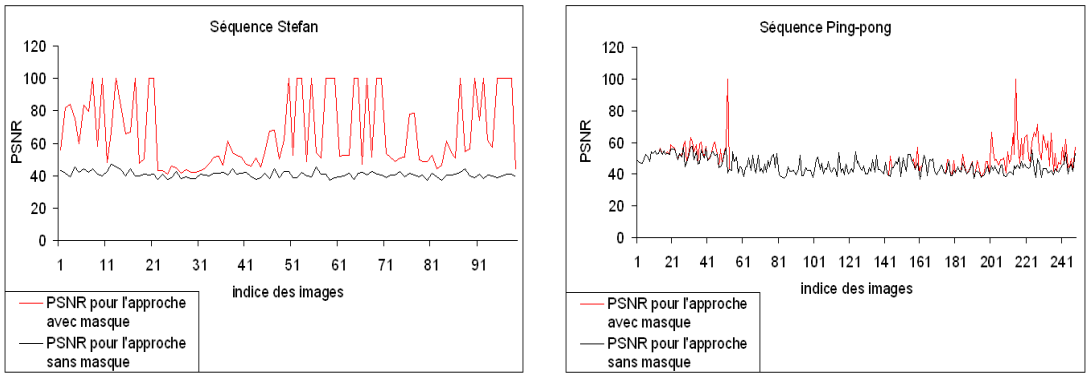


FIG. 82 – Résultats de corrélation pour la séquence ”Stefan” et la séquence ”Ping-pong” avec la prise en compte du masque, (a) et (c), et sans la prise en compte du masque, (b) et (d)

Enfin, sur les figures 83 (a) et 83 (b), nous présentons les courbes de PSNR obtenues avec l’utilisation du masque.



(a) (b)

FIG. 83 – Courbes de PSNR pour la séquence ”Stefan” et la séquence ”Ping-pong” avec la prise en compte du masque

L'usage d'un masque ralentit naturellement la vitesse de convergence de la courbe de corrélation. En effet, l'utilisation du masque conduit à ne pas sélectionner l'ensemble des blocs obtenus par la technique décrite en section 3.3.1. Cependant, nous sommes toujours capable de détecter la bonne marque, et les courbes de détection semblent pouvoir converger au-delà du seuil, mais le nombre d'images utilisées est trop faible. Pour pallier ce problème, il suffirait de réaliser la simulation de l'insertion sur un plus grand nombre de blocs ou de réaliser non pas un masque a posteriori, mais un masque a priori. Enfin, nous pouvons noter que l'augmentation du PSNR est légèrement inférieure pour la séquence ”Ping-pong” que pour la séquence ”Stefan”. Or, le seuil utilisé est le même pour les différentes vidéos testées, il apparaît donc nécessaire d'adapter ce seuil en fonction de critères propres à la vidéo à marquer, critères devant s'adapter à notre schéma de tatouage.

4.4.2 PSNR pondéré par une mesure locale du contraste

Afin d'adapter le masque en fonction du contenu de la vidéo, nous avons pondéré ce dernier par une mesure locale du contraste. Pour ce faire, nous avons utilisé le facteur de contraste local défini dans le mémoire de R. Pastrana [125]. Le masque initial ne s'adapte pas au contenu de la vidéo. Or afin d'optimiser l'insertion il est nécessaire de prendre en compte les caractéristiques de la vidéo à marquer. Pour ces raisons nous avons choisi d'établir un critère local. Ce critère s'adapte à l'architecture de notre algorithme. En effet, il est calculé sur des blocs de taille équivalente à ceux utilisés pour l'insertion, et nous permet d'adapter le seuil utilisé dans la section précédente en le pondérant.
Ce facteur est le suivant :

$$f_c = a.log(u+1); 0 \leq u \leq b \tag{70}$$

où : $u =\mid p_c - \bar{p} \mid$;
avec, a = constante, $b = 2^{n-1}$ et n = nombre de bits par pixel
$a = 105$ est calculé pour avoir une plage de données de sortie entre 0 et 255 (cette valeur permet en outre de visualiser plus clairement la réponse de la métrique locale).
p_c est la valeur du pixel central dans la fenêtre $3 * 3$.
$\bar{p}$ est la moyenne des voisins de p_c.
Le facteur de contraste sur un bloc de taille $m * n$ est donné par :

$$f_{c_{global}} = \frac{\sum_{j-2}^{m-1} \sum_{i-2}^{n-1} f_c(i,j)}{(m-2).(n-2)} \tag{71}$$

Sur les figures 84 (a), 84 (b), 84 (c), et 84 (d), nous présentons les résultats de détection pour les séquences ”Stefan” et ”Ping-pong”. Nous comparons les résultats de l’approche avec la prise en compte du masque pondéré et l’approche avec le masque non pondéré présenté dans la section précédente.

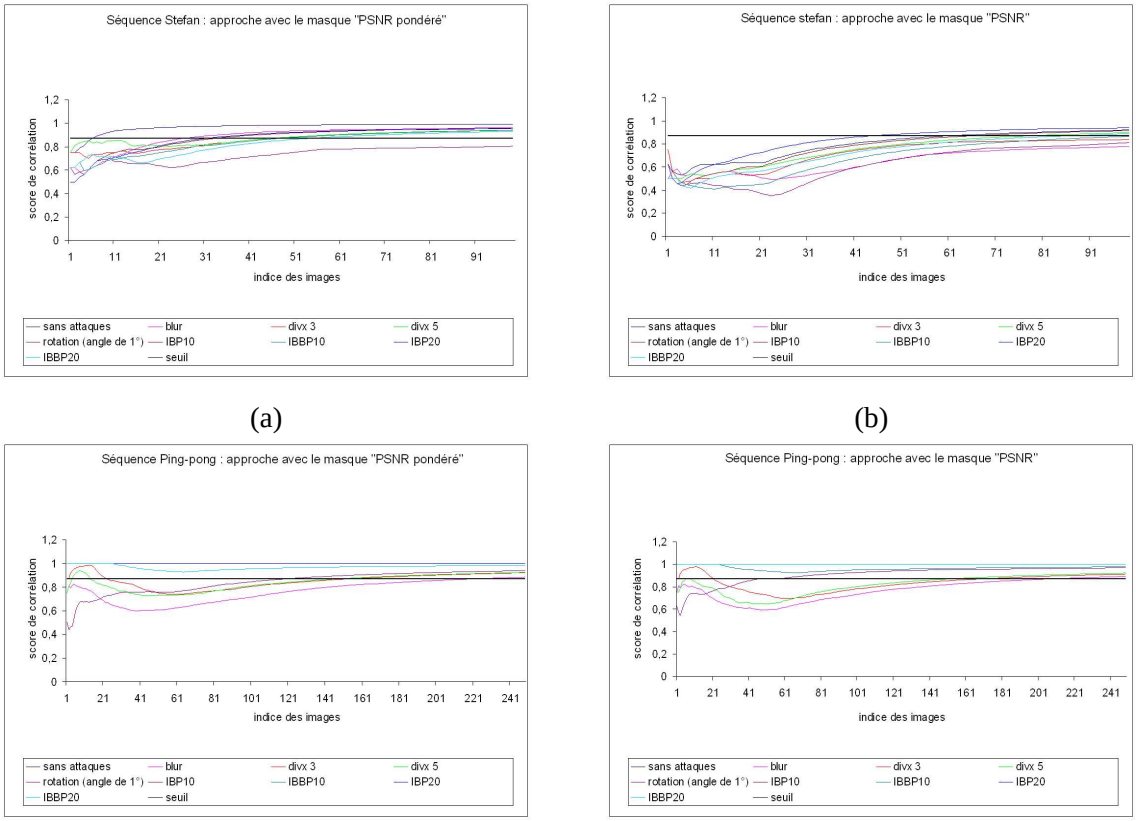
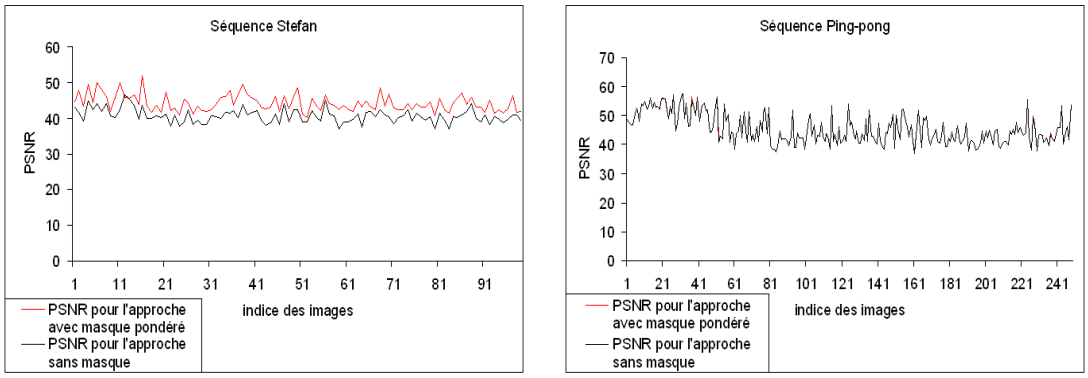


FIG. 84 – Résultats de corrélation pour la séquence ”Stefan” et la séquence ”Ping-pong” avec la prise en compte du masque pondéré, (a) et (c), et avec le masque non pondéré, (b) et (d)

Enfin, sur les figures 85 (a) et 85 (b), nous exposons les courbes de PSNR obtenues avec cette approche.



(a) (b)

FIG. 85 – Courbes de PSNR pour la séquence ”Stefan” et pour la séquence ”Ping-pong” avec la prise en compte du masque pondéré

On peut noter que l'augmentation du PSNR est supérieure pour la séquence ”Stefan” et nulle pour la séquence ”Ping-pong” par rapport à l'approche classique, mais amoindri vis à vis du masque non pondéré. Ce résultat est logique, car l'utilisation du masque pondéré conduit à sélectionner plus de vecteur de mouvement. Les scores de corrélation sont cependant plus élevés que pour l'approche non pondérée. Il apparaît donc que la pondération n'est pas suffisante. Soit le critère n'est pas suffisamment approprié, soit le paramétrage n'est pas adéquat. Concernant les deux approches que nous venons de décrire, le paramétrage du masque est à effectuer en fonction du compromis recherché, c'est à dire, entre invisibilité, robustesse et capacité. Il serait nécessaire d'approfondir l'étude de celui-ci afin d'améliorer cette approche. Afin de rendre le masque plus stable entre l'insertion et la détection lors d'utilisation d'attaques sur la vidéo, nous avons examiné l'utilisation de préfiltrages sur la vidéo avant de calculer le masque.

4.4.3 Les différents préfiltrages

Afin de ”robustifier” le masque, nous avons décidé de réaliser des préfiltrages sur l'image à marquer avant de réaliser le calcul du masque. Cette nouvelle procédure est illustrée sur la figure 86.

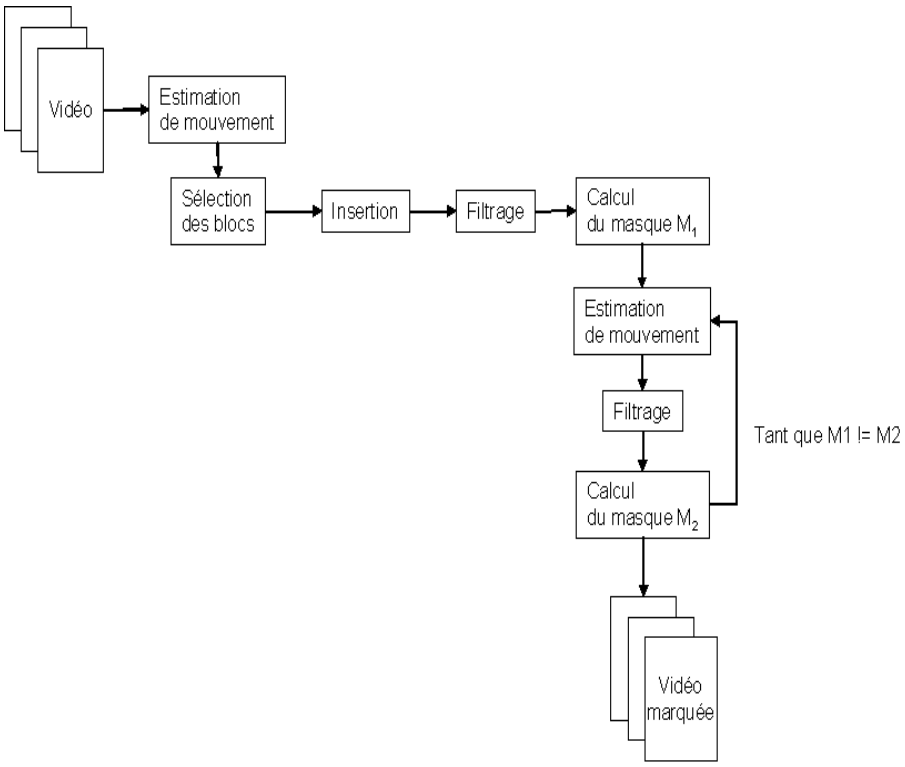
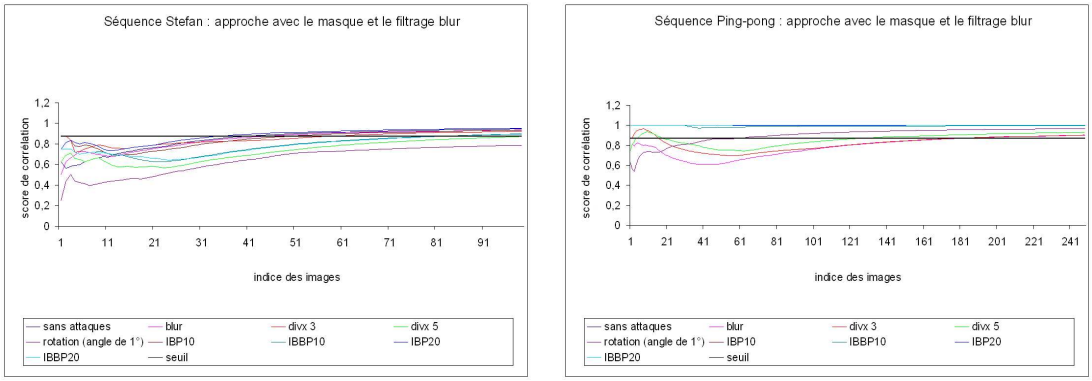


FIG. 86 – Calcul du masque basé sur un critère de PSNR avec l'utilisation d'un préfiltrage.

Le but de ce procédé est d'obtenir un "résumé" de l'image que l'on puisse retrouver après l'application de traitements tels que différents codages.

Filtre blur

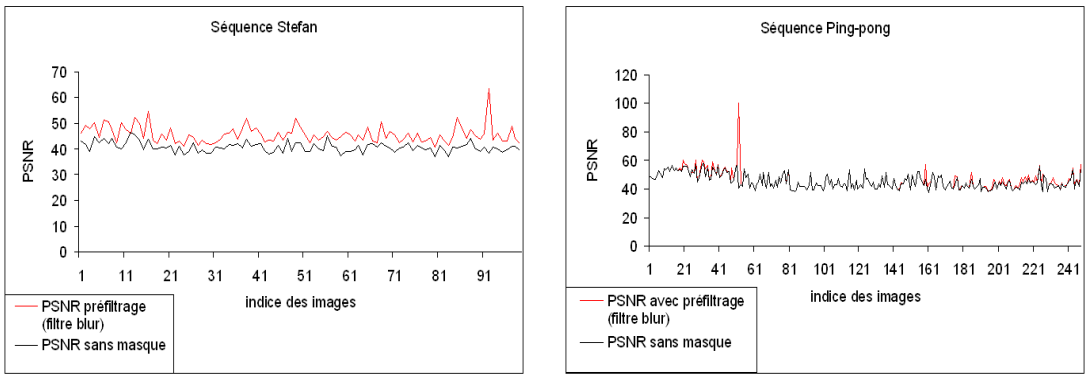
En premier lieu, nous avons utilisé un filtre blur, afin d'obtenir une version de la vidéo légèrement floue, aspect qui peut survenir lors d'une compression trop forte. Sur les figures 87 (a) et 87 (b), nous présentons les résultats de détection pour les séquences "Stefan" et "Ping-pong".



(a) (b)

FIG. 87 – Résultats de corrélation pour la séquence ”Stefan” et pour la séquence ”Ping-pong” avec le masque et un préfiltrage blur

Enfin, sur les figures 88 (a) et 88 (b), nous exposons les courbes de PSNR obtenues avec cette approche.

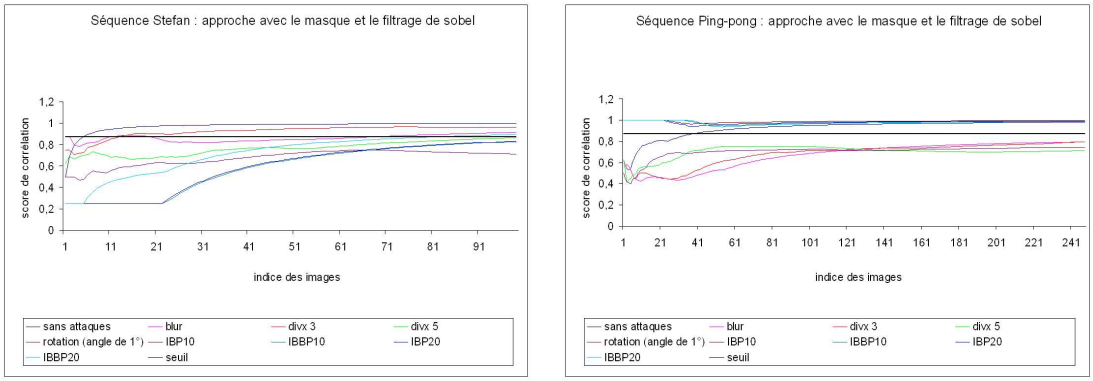


(a) (b)

FIG. 88 – Courbes de PSNR pour la séquence ”Stefan” et pour la séquence ”Ping-pong”

Filtre de Sobel

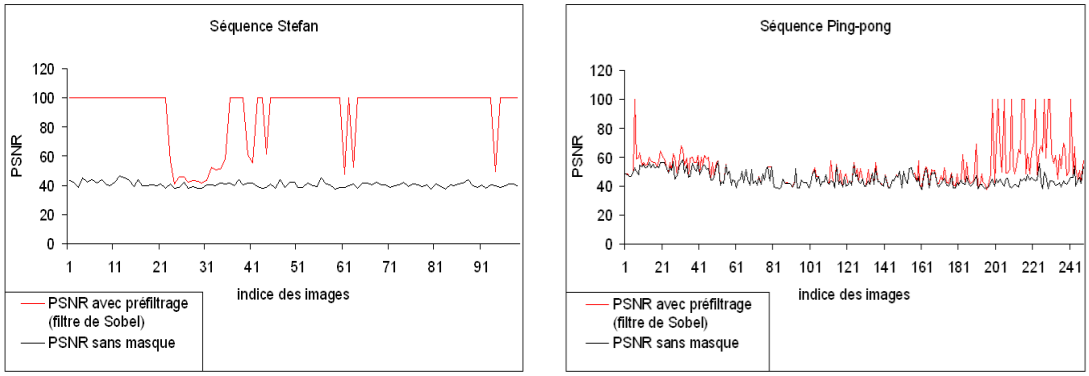
Nous avons ensuite testé un filtre nous donnant un gradient de l’image. Pour cela nous avons utilisé un filtrage de Sobel. Sur les figures 89 (a) et 89 (b), nous présentons les résultats de détection pour les séquences ”Stefan” et ”Ping-pong”.



(a) (b)

FIG. 89 – Résultats de corrélation pour la séquence ”Stefan” et pour la séquence ”Ping-pong” avec le masque et un préfiltrage de Sobel

Enfin, sur les figures 90 (a) et 90 (b), nous exposons les courbes de PSNR données par cette approche.

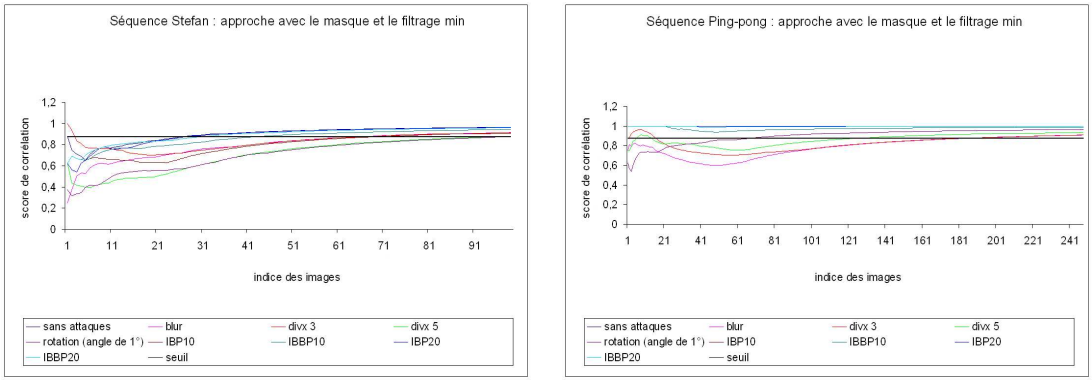


(a) (b)

FIG. 90 – Courbes de PSNR pour la séquence ”Stefan” et pour la séquence ”Ping-pong”

Filtre Min

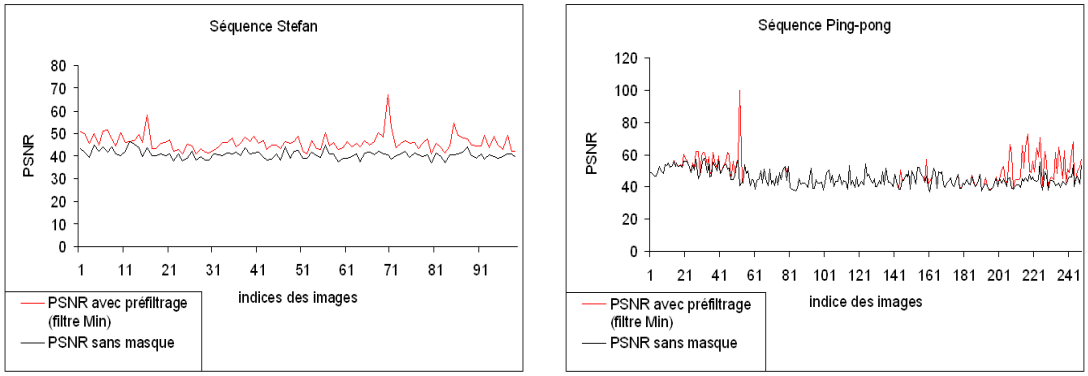
Enfin, nous avons testé deux filtres de type morphologique. Un filtre ”Min” affecte au pixel traité, la valeur minimale des pixels dans un voisinage prédéfini de celui-ci. Nous avons utilisé ici une taille de filtre de $5*5$. Sur les figures 91 (a) et 91 (b), nous présentons les résultats de détection pour les séquences ”Stefan” et ”Ping-pong”.



(a) (b)

FIG. 91 – Résultats de corrélation pour la séquence ”Stefan” et pour la séquence ”Ping-pong” avec le masque et un préfiltrage Min

Enfin, sur les figures 92 (a) et 92 (b), nous exposons les courbes de PSNR obtenues avec cette méthode.

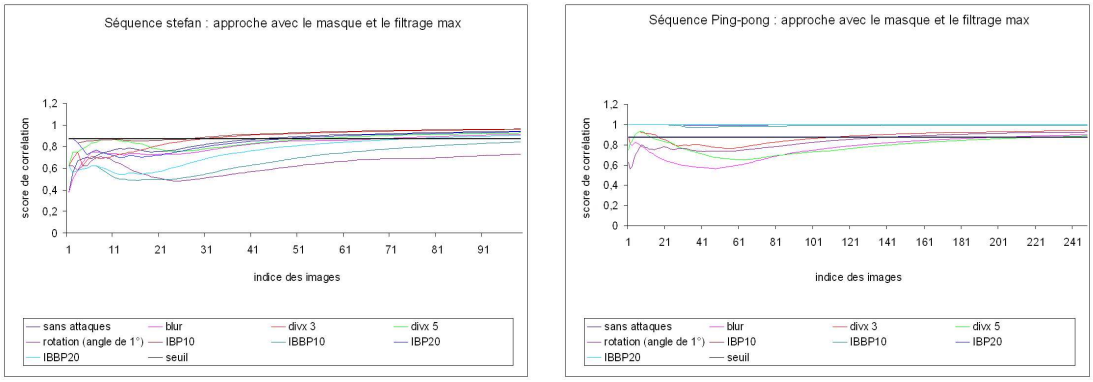


(a) (b)

FIG. 92 – Courbes de PSNR pour la séquence ”Stefan” et pour la séquence ”Ping-pong”

Filtre Max

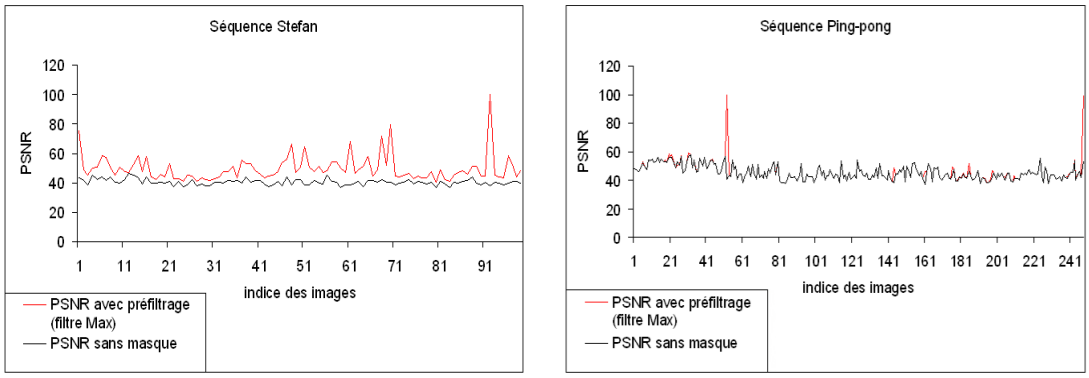
A contrario, ce filtre affecte au pixel traité la valeur maximale des pixels dans un voisinage prédéfini de celui-ci. De la même manière, nous avons utilisé ici une taille de filtre de $5 * 5$. Sur les figures 93 (a) et 93 (b), nous présentons les résultats de détection pour les séquences ”Stefan” et ”Ping-pong”.



(a) (b)

FIG. 93 – Résultats de corrélation pour la séquence ”Stefan” et pour la séquence ”Ping-pong” avec le masque et un préfiltrage Max

Enfin, sur les figures 94 (a) et 94 (b), nous exposons les courbes de PSNR obtenues avec cette approche.



(a) (b)

FIG. 94 – Courbes de PSNR pour la séquence ”Stefan” et pour la séquence ”Ping-pong”

Discussion

En examinant ces différentes courbes, nous pouvons remarquer que le niveau de robustesse est légèrement inférieur à l’approche sans masque. Ceci peut s’expliquer par les mêmes raisons exposées précédemment, la prise en compte du masque diminuant le nombre de blocs marqués. En revanche, il semble que l’augmentation en PSNR ne soit pas très significative dans les cas où l’on utilise un préfiltrage. Cependant, tout comme pour le masque avec la pondération, il est nécessaire d’adapter les filtrages en fonction de l’application visée. Les filtrages blur, min et max, donnent

des résultats relativement proches. L'effet ce ces filtres étant proche, ces résultats sont logiques. Les meilleurs résultats concernant l'augmentation du PSNR sont donnés par le filtrage de Sobel. Avec ce filtrage le nombre de blocs sélectionnés est plus faible, cela s'explique aisément. En effet, ce filtrage nous permet d'obtenir un gradient de l'image qui est alors plus sensible aux variations, et par conséquent lors du calcul du masque, les blocs sélectionnés devront être très proches des blocs originaux. Un aperçu récapitulatif des résultats est présenté sur les figures 95, 96, 97 et 98.

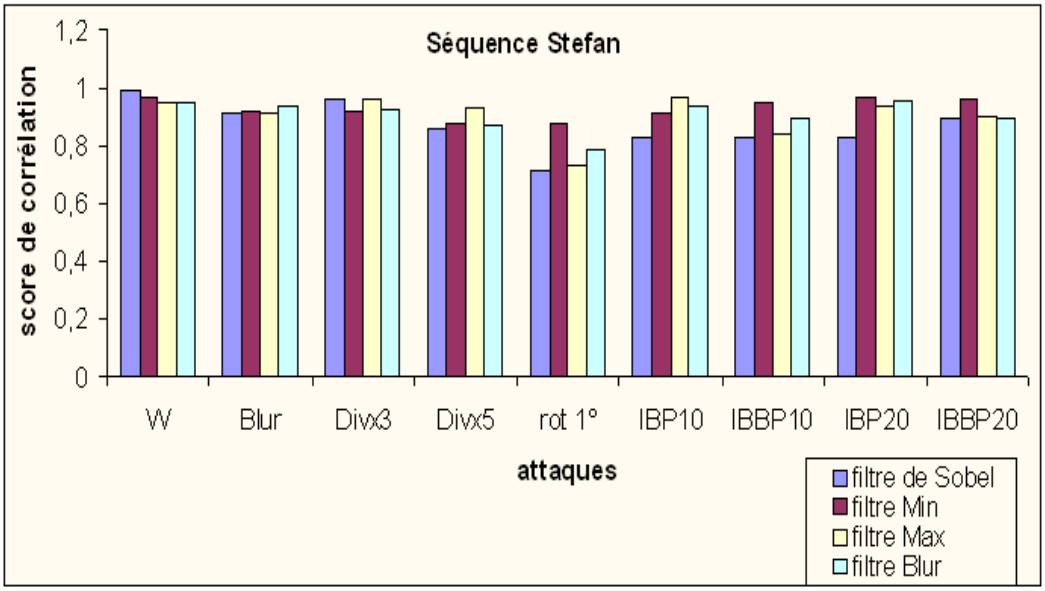


FIG. 95 – Résultats récapitulatifs pour la séquence "Stefan"

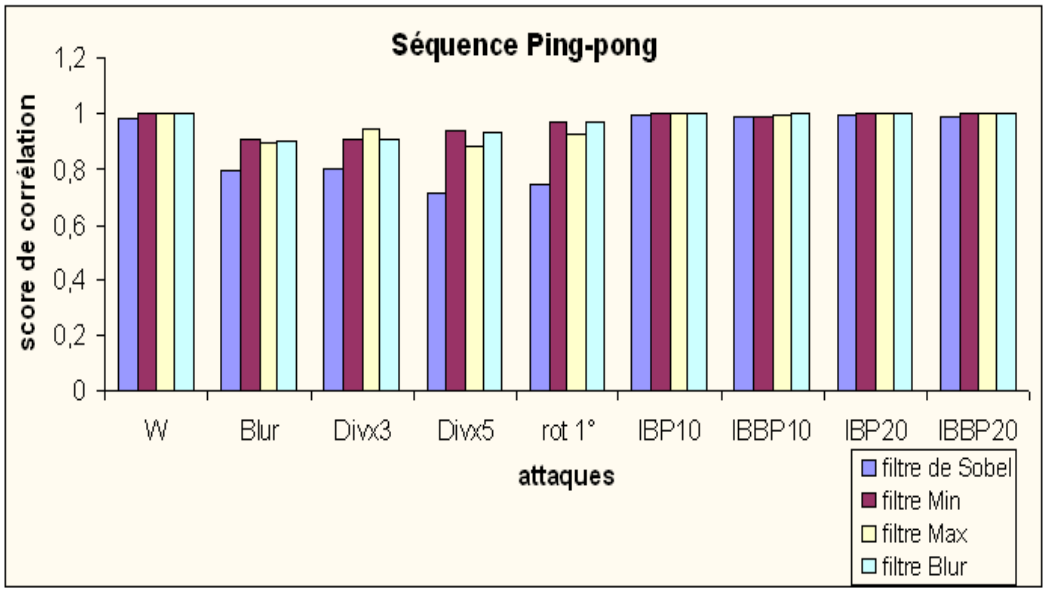


FIG. 96 – Résultats récapitulatifs pour la séquence ”Ping-pong”

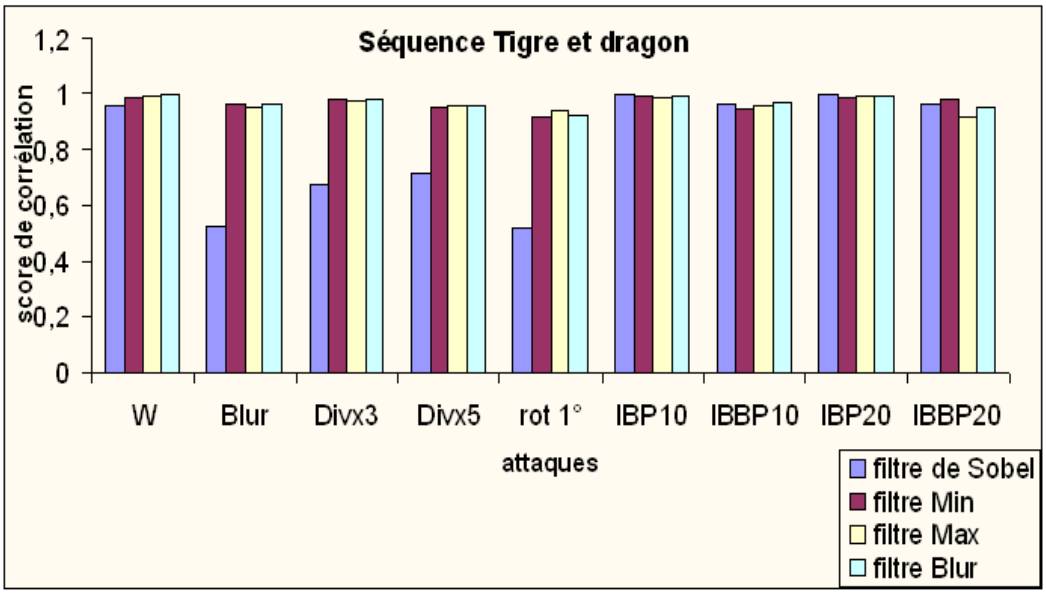


FIG. 97 – Résultats récapitulatifs pour la séquence ”Tigre et dragon”

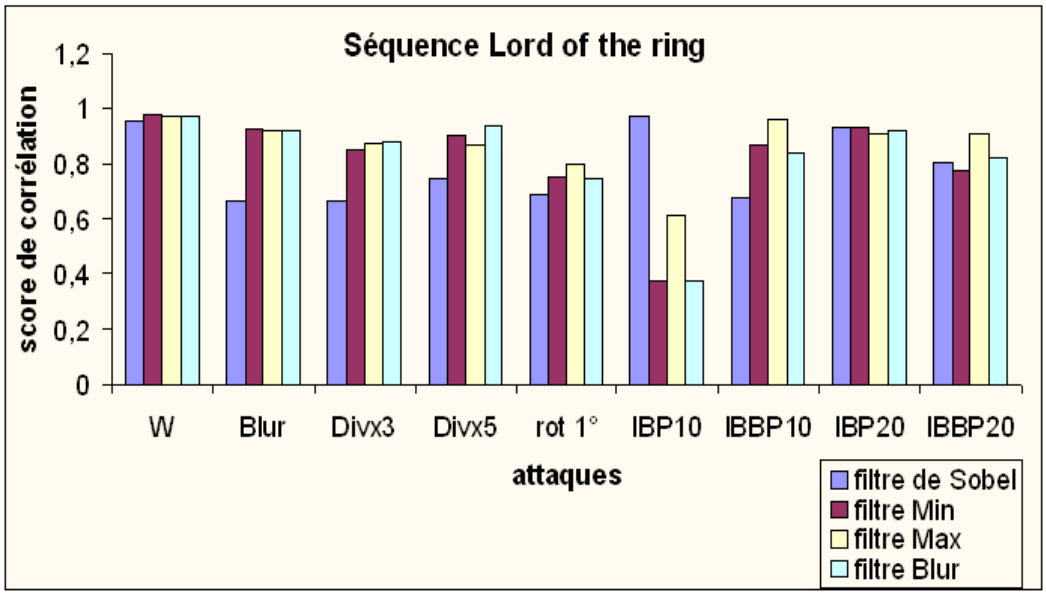


FIG. 98 – Résultats récapitulatifs pour la séquence ”Lord of the ring”

Il serait intéressant de combiner cette approche de filtrage avec la pondération du masque et l'utilisation des points saillants (que nous exposerons dans le chapitre suivant) afin de déterminer des zones de marquage qui restent stable entre l'insertion et la détection, dans le cas où des attaques ont eu lieu sur la vidéo.

4.4.4 Conclusion

Rappelons que nous effectuons un masquage a posteriori, basé sur un critère de PSNR. Ce dernier est reconnu pour ne pas être suffisamment représentatif du système visuel humain, mais plutôt pour être une bonne mesure de similarité entre deux images.
Nous présentons sur la figure 99 un récapitulatif de la vitesse de convergence des différentes approches présentées dans ce chapitre. Nous comparons ces approches avec l'approche adaptative présentée au chapitre 3.

Vitesse de convergence	Séquence Stefan							Séquence Ping-pong						
Approches / Attaques	Approche adaptative	Masque PSNR	Masque PSNR pondéré	Masque avec filtre de Sobel	Masque avec filtre Min	Masque avec filtre Max	Masque avec filtre Blur	Approche adaptative	Masque PSNR	Masque PSNR pondéré	Masque avec filtre de Sobel	Masque avec filtre Min	Masque avec filtre Max	Masque avec filtre Blur
Sans attaque	6	63	6	5	27	44	41	1	1	1	40	1	1	1
Blur	36	-	27	11	66	70	45	198	217	230	-	188	203	191
Divx3	45	-	47	13	67	31	59	123	177	159	-	186	112	188
Divx5	45	81	47	-	97	56	-	127	159	153	-	127	228	133
Rotation 1°	-	-	-	-	-	-	-	107	48	124	-	68	142	63
IBP10	36	68	34	-	70	25	50	1	1	1	1	1	1	1
IBBP10	49	-	49	-	43	-	82	1	1	1	1	1	1	1
IBP20	32	46	32	-	28	49	35	1	1	1	1	1	1	1
IBBP20	56	83	31	77	84	89	56	1	1	1	1	1	1	1

FIG. 99 – Résultats récapitulatifs des vitesses de convergence pour les différentes approches traitées dans ce chapitre

Comme nous pouvons le voir, sur ce tableau, l'approche qui converge le plus vite pour la séquence "Stefan" semble être celle avec l'utilisation d'un préfiltage de Sobel, cependant comme nous pouvons le remarquer cette approche est moins robuste que l'approche adaptative. Pour la séquence "Ping-pong", l'approche avec le préfiltrage Min et l'approche adaptative présente des vitesses de convergence très proche. Si on prend en compte la vitesse de convergence et la robustesse, l'approche adaptative reste plus efficace. Cependant, l'intérêt majeur de mettre en place un masque est d'augmenter l'invisibilité de notre algorithme.

L'approche que nous avons décrite a montré son intérêt en terme d'invisibilité. En effet, le gain en PSNR est significatif, même si cela se fait au détriment d'une légère perte en robustesse. En outre, le fait d'utiliser le masque a posteriori diminue le nombre de blocs marqués, ce qui a pour conséquence de diminuer également la vitesse de convergence de la corrélation, cependant les tests ont été réalisé sur des vidéos contenant très peu d'image (250 pour la séquence Ping-pong, ce qui correspond à $10s$ de vidéo), sur des vidéos "de taille usuelle" (par exemple un film de $1h30$), cette diminution de la vitesse de convergence ne serait pas significative. Pour pallier ce problème, deux solutions sont envisageables. La première consiste à examiner plus de blocs lors de l'insertion,

l'utilisation du masque nous assurant que les dégradations ne devraient pas être plus gênantes que dans l'approche sans masque. La deuxième solution consisterait à déterminer un masque a priori. Dans ce cas, il est nécessaire de réaliser une segmentation spatiale, afin de déterminer les zones susceptibles d'accueillir le marquage. Le principal problème de cette approche est la stabilité du masque entre l'insertion et la détection. La détermination des zones de marquage pourrait se faire, par exemple, par l'utilisation de points saillants, assurant non seulement une segmentation de l'image, mais aussi une certaine stabilité permettant de retrouver un masque proche, voire identique au niveau de la détection, et ainsi de retrouver les zones d'insertion. Enfin, les résultats concernant le masque pondéré, n'ont pas montré d'améliorations significatives. Cependant un paramètrage adéquat ou l'utilisation d'autres critères de pondération pourrait amener à de meilleurs résultats.

Chapitre 5

Conclusions et perspectives

5.1 Introduction

Nous allons dans cette section, présenter les différentes perspectives de notre solution de tatouage vidéo, ainsi que les conclusions, que nous pouvons extraire de nos expérimentations.

5.2 Conclusion des tests

Dans un premier temps, nous avons développé une approche qui s'est avérée être suffisamment robuste à des attaques spécifiques telles que la compression Divx (version 3 et 5), la compression H264, un filtrage blur, et une rotation de 1°. Cependant, l'insertion de la marque engendrant parfois des perturbations visuelles au niveau de la vidéo, nos travaux se sont ensuite essentiellement concentrés sur l'amélioration de l'invisibilité de notre algorithme.
A ce jour, certains artefacts restent cependant perceptibles. De ce fait, le développement de notre approche reste ouvert, et nous laisse entrevoir un grand nombre de perspectives quant à l'amélioration générale de notre système dont nous proposons un aperçu dans ce chapitre.

5.3 Perspectives

Chaque brique de notre algorithme pouvant être sujette à des optimisations, nous allons exposer ici les différentes approches envisageables, afin de rendre notre méthode plus performante, sans pour autant présenter une liste exhaustive de ces améliorations.

5.3.1 Prétraitement de la marque

En premier lieu, il serait intéressant d'approfondir l'étude de la formation de la marque à insérer et notamment d'étudier d'autres types de code correcteur d'erreur comme les turbo-codes.

5.3.2 Estimateur de mouvement

Dans cette section, nous présentons tout d'abord les améliorations et variantes qu'il est possible d'apporter à un algorithme de block matching. Comme nous l'avons présenté brièvement dans le chapitre 3, il existe différents types d'estimateur de mouvement. Nous avons utilisé un "BMA", qui est à ce jour, un des plus répandu dans les standards de compression. Cependant, il existe différentes approches d'estimation de mouvement comme les estimations de mouvement continu à base de maillages ou d'ondelettes. Une étude comparative sur l'insertion d'un tatouage sur des vecteurs de mouvement issus de techniques différentes s'avérerait intéressante. En effet, elle nous permettrait peut être d'adapter notre choix en fonction de l'application visée. Il serait

par exemple intéressant d'utiliser un estimateur de mouvement continu, basé sur les techniques de maillage. Ce type d'estimateur aurait pour conséquence de diffuser les perturbations apportées par le marquage sur l'ensemble de l'image. On peut alors supposer que les artefacts dûs aux effets de blocs disparaîtraient, pour n'obtenir seulement que de légères perturbations locales. Enfin, il serait intéressant de réaliser une analyse du choix des paramètres de l'estimation de mouvement (taille des blocs, précision pixélique , ...) et de la géométrie des grilles afin de déterminer l'espace optimal d'insertion.

Estimation de mouvement

On pourrait tout d'abord réaliser l'estimation de type "arrière". Ensuite, nous pourrions effectuer des estimations bidirectionnelles. On pourrait également réaliser des estimations, non pas sur des images contiguës, mais sur des images séparées par un delta donné (ce delta devant être relativement faible pour que les images aient des contenus proches). En générant de façon pseudo-aléatoirement ces deltas entre les images et en combinant adroitement les différentes possibilités d'estimations, cela pourrait permettre d'augmenter la robustesse de notre algorithme, le but de notre estimation de mouvement n'étant pas d'être optimale.

Variation du bloc source et du bloc cible

Lors de l'insertion, nous ne faisons varier que le bloc cible des vecteurs de mouvement. Il serait intéressant d'étudier la répercution de modifications réalisées à la fois sur le bloc source et sur le bloc cible. Ceci aurait pour conséquence de diminuer les distorsions de blocs lors de la réalisation de la compensation de mouvement avec les blocs marqués.

Les extensions du Block Matching

L'estimateur de mouvement mis en oeuvre correspond à un estimateur pixelique. Il serait intéressant d'examiner une estimation sur des niveaux subpixeliques, qui permettrait de générer un nombre de niveau plus important dans la pyramide décrite en section 3.3.3 et ainsi d'accroître la redondance du marquage.
L'un des principaux défauts des méthodes basées blocs, concerne la limitation du mouvement à une simple translation. Ces méthodes ne peuvent pas prendre en compte les déformations de type affine. Afin de résoudre ce problème, des améliorations du BMA ont déjà été proposées. Elles reposent sur une modification de la forme des éléments polygonaux et/ou du modèle de mouvement utilisé. La forme des blocs définit un certain nombre de degrés de liberté (en fonction du nombre

de sommets du polygone). On peut donc utiliser, suivant le nombre de noeuds des éléments polygonaux, des modèles de mouvement plus ou moins complexes, comme des transformations affines, bilinéaires ou encore perspectives.

Les maillages actifs

Les algorithmes d'estimation de mouvement par maillage reposent sur la mise en œuvre de méthodes par éléments finis, qui permettent de modéliser un champ dense à partir des déplacements estimés aux nœuds d'un maillage. Le champ intérieur à une maille est déterminé par une fonction d'interpolation.

Afin d'estimer au mieux les déplacements de ces nœuds, plusieurs approches sont possibles. La complexité et la qualité, de ces dernières, varient fortement d'une méthode à l'autre.
De plus, il existe une dépendance de ces estimateurs avec la construction du maillage. En effet, afin de réaliser un maillage, deux approches sont possibles : soit par échantillonnage régulier de la surface, soit par positionnement des nœuds de manière à s'adapter aux contenus. Dans [91] et [161], des grilles régulières constituées de mailles quadrangulaires et triangulaires sont utilisées. Cette approche présente l'avantage, dans le cadre d'un schéma de codage vidéo, de ne pas avoir à coder les positions des nœuds. Par opposition, dans [62], les auteurs utilisent une estimation de mouvement basée maillage, afin de réaliser une interpolation temporelle de trame. Ils positionnent les nœuds des mailles triangulaires de manière à obtenir des arêtes parallèles aux contours, et une plus forte densité de nœuds dans les régions présentant beaucoup de détails. Dans ce but, des opérateurs de détection de contours et de squelettisation de l'image sont employés.

Une fois la régularité du maillage définie en fonction de l'application visée, on doit choisir également une méthode de détermination des déplacements des nœuds d'un maillage. Il en existe une multitude, on citera par exemple la méthode par échantillonnage d'un champ dense, celui-ci étant estimé par une méthode de type pel-récurcive (comme dans [62]).

Toutefois, dans la littérature, les méthodes précédemment citées sont souvent suivies par des techniques itératives, visant à optimiser localement la position des nœuds, en se basant sur un critère de minimisation de la DFD dans leurs zones d'influence (comme dans [97]). Le premier algorithme itératif de ce type est sans doute celui de [73], qui propose une technique appelée CGI pour "Control Grid Interpolation". Cependant, celle-ci est basée sur une recherche via un BMA.

Une évolution de cette méthode est décrite dans [161] et propose un algorithme appelé ”Hexagonal Matching”. Ce procédé permet d'optimiser localement le mouvement d'un nœud. De plus, afin d'éviter l'apparition de trop fortes dégénérescences du maillage, un terme de régularisation est introduit. Celui-ci, appelé ”Shape-Preserving Energy”, est minimal lorsque les triangles d'un hexagone conservent leur forme. Cependant, les déplacements estimés par ces méthodes ne correspondent pas à une minimisation globale de la DFD, et en interdisant les dégénérescences, les zones d'occlusions ne peuvent pas être déterminées.

Enfin, la dernière famille de méthodes, est celle visant à minimiser globalement la DFD, ce qui nécessite la résolution d'un problème de type moindres carrés. On rencontre principalement deux approches, les descentes de gradients (comme dans [101]) et les descentes du deuxième ordre de type Gauss-Newton, sur lesquelles sont basés les travaux de [120].

On notera cependant que les méthodes basées maillages actifs présentent des défauts relatifs aux dégénérescences des maillages. En effet, l'application des paramètres de mouvement, estimés sans contraintes, peut conduire à des maillages à mailles fortement déformées (étirements, tassements, et retournements). Différentes techniques ont été proposées afin d'estimer les paramètres de déplacement, tout en contrôlant la topologie et la géométrie du maillage. P. Lechat [120] présente une correction a posteriori du mouvement, suivant une technique décrite dans [160], ainsi qu'une méthode de détermination des paramètres de mouvement sous une contrainte de non-retournement. Celle-ci utilise un lagrangien augmenté, afin de contrôler l'évolution de la compacité des mailles au cours du procédé d'estimation de mouvement. On peut également citer les travaux présentés par [134] dans lesquels la déformation du maillage actif est controlée par trois termes d'énergie. Ces énergies sont relatives à un critère de mouvement (basé sur la DFD), et sur un critère spatial, caractérisant la position des arêtes vis à vis des contours détectés dans l'image de référence, ainsi qu'à un critère de régulation temporelle. Ce dernier est très spécifique à l'application de suivi d'objets et de segmentation spatio-temporelle. Cependant, l'algorithme proposé étant déterministe, le déplacement des noeuds est limité par une fenêtre de recherche. De plus, l'auteur utilise un remaillage, afin de contrôler l'aspect du maillage, qui sera le support d'une nouvelle étape d'estimation.

5.3.3 Analyse du mouvement

L'analyse spatiale ne suffit pas à éviter l'ensemble des artefacts engendrés par le tatouage au sein d'une vidéo. Il est nécessaire d'analyser les caractéristiques du mouvement. En effet, certains artefacts n'apparaissent que lors de changements brusques dans l'orientation du mouvement

général de la séquence. Comme nous avons pu le constater au chapitre 4, les images n'apparaissant que très brièvement (40ms). Les défauts spatiaux (artefacts de bloc) ne sont donc pas permanents, et ils seront par conséquent moins importants. En revanche, les clignotements sont particulièrement gênant, il s'avère nécessaire d'appliquer une stratégie d'insertion adaptée à la régulation de ce type d'artefact. Pour ce faire, il serait utile de réaliser une segmentation temporelle. En effet, une étude sur le mouvement moyen d'une séquence peut être suffisant à diminuer considérablement la visibilité du marquage. Pour cela, il suffit d'examiner la dérivée première de ce vecteur moyen, un changement de signe correspondra à un changement d'orientation du mouvement global. Dans ce cas il est préférable de ne pas marquer l'image.

5.3.4 Grille hexagonale

Afin d'optimiser la grille de référence, il pourrait être intéressant d'étudier des grilles de type hexagonal qui présente a priori une répartition plus homogène des deux zones Z_1 et Z_2.

5.3.5 Les points saillants

La sélection des vecteurs de mouvement repose au final sur un processus de sélection pseudo-aléatoire, qui ne prend pas en compte le contenu de la vidéo. Cette approche n'est pas optimale, il serait donc nécessaire d'envisager des approches de type points saillants afin de localiser des zones synchronisables entre la détection et l'insertion et suffisamment robustes pour pouvoir garder le même ensemble d'insertion à la détection.
La détection des points saillants dans les images a suscité un grand nombre de recherches depuis de nombreuses années. Historiquement, les points saillants étaient caractérisés par les coins des objets. Cette caractérisation provient du domaine de la vision par ordinateur. Aujourd'hui, bon nombre de chercheurs s'accordent à dire que les coins ne représentent pas une information pertinente pour représenter une image (car trop focalisés sur la texture).
Une approche qui nous semble intéressante, concerne la détection des points d'une image localisés dans des régions présentant des contours. Les contours permettent de générer une représentation du contenu de l'image en adéquation avec le système visuel humain. Cette approche est basée sur la théorie des ondelettes qui permet une analyse multi-résolution de l'image. Comparée aux approches traditionnelles, les points saillants détectés avec cette méthode permettent une représentation beaucoup plus fidèle de l'image en étant localisés dans les zones où l'information est pertinente. De cette façon, les zones ainsi localisées resteront stables à l'insertion et à la détection même si de légères dégradations sont survenues sur la vidéo.

5.3.6 Attaques

Il serait nécessaire d'expérimenter l'algorithme sur d'autres types d'attaques et d'étudier l'élaboration de systèmes de protection supplémentaires si nécessaire. Il serait plus particulièrement intéressant d'étudier le problème de la scalabilité temporelle et spatiale qui sont de plus en plus présentes dans les nouveaux codecs [45].

Enfin, il est possible de créer une attaque basée sur le modèle de Watson, qui permet de détecter les artefacts de blocs et ainsi de localiser les zones marquées (c.f. chapitre 4). En raison de la sélection pseudo-aléatoire, il est possible de trouver un équivalent aux blocs marqués dans les images voisines de l'image en cours de traitement, ainsi, en remplaçant les blocs marqués par ces blocs équivalents (par exemple, en maximisant un critère de ressemblance tel que le PSNR), il serait alors possible de désynchroniser la détection et de provoquer une mauvaise détection. A ce jour, nous n'avons pas trouvé de solutions à cette attaque, un des axes fondamentaux à étudier, consiste donc à élaborer un système de protection capable de mettre en défaut l'attaque que nous venons de décrire.

5.3.7 Corrélation des approches d'embrouillage et de tatouage

Actuellement, les algorithmes d'embrouillage et de tatouage, que nous avons élaborés, ne sont pas directement corrélés. Il est possible d'appliquer en premier lieu le tatouage puis l'embrouillage. Il serait intéressant de corréler ces deux approches afin qu'elles se déroulent au cours de la même procédure. Cependant dans ce cas, si on réalise la tatouage dans le domaine compressé, il faudra prendre en compte les effets de dérive dûs aux légers défauts apportés par le tatouage. Dans un premier temps, il faudrait étudier la stabilité de la combinaison de ces deux approches dans le domaine non compressé. Étant donné, que la procédure d'embrouillage est réversible et nous permet de reconstruire la vidéo dans sa version originale, on peut penser qu'en appliquant tout d'abord la procédure de tatouage, et ensuite la procédure d'embrouillage, on conservera les informations de tatouage. La principale difficulté réside dans l'adaptation du tatouage au domaine compressé.

5.4 Conclusion générale

Le tatouage représente une alternative à la protection de la propriété intellectuelle des supports multimedia courant. Le domaine d'application de cette technologie encore jeune s'étend du monde 1D (l'audio) au monde 3D (3D+t) en passant par la 2D et la 2D+D (vidéo). Cependant, bien qu'au début de son apparition le tatouage apparaissait comme étant la solution à tous les problèmes de protection de contenus face aux multiples attaques que ces derniers pourraient subir, la communauté de tatouage s'est aperçue progressivement que, créer un algorithme de tatouage assurant une protection absolue représentait une mission impossible. En effet, nous savons aujourd'hui que le tatouage seul ne peut pas répondre à une protection suffisamment fiable dans un milieu grand public ou les degrés de liberté en terme de manipulation des contenus sont trop élevés. C'est pourquoi beaucoup d'industriels s'orientent vers l'élaboration de solutions spécifiques dépendant de l'application visée, essentiellement destinées au domaine professionnel. Il semble de plus en plus évident que l'on ne pourra jamais empêcher le piratage grand public, à moins de créer des systèmes exclusivement propriétaires, mais dans ce cas il n'est absolument pas assuré que la protection soit efficace.

Dans cette thèse, nous nous sommes intéressés à l'élaboration d'un nouvel algorithme de tatouage en essayant de prendre en compte les principaux axes de développement d'un tel système. Après avoir introduit l'exploitation des vecteurs de mouvement dans le cadre de l'élaboration d'un système d'embrouillage, nous avons poursuivi sur l'exploitation de ces derniers dans le cadre du tatouage. Nous avons commencé par élaborer une règle d'insertion nous assurant une robustesse essentiellement à différents types de codage. Par la suite, nous avons examiné différents axes afin d'améliorer notre méthode. Pour ce faire, nous avons examiné le formatage de la marque, l'optimisation de la détection, et enfin nous avons étudié les aspects perceptuels afin de minimiser les artefacts associés au marquage des vecteurs de mouvement.
Comme nous avons pu le voir dans le chapitre 5, les perspectives restent nombreuses. Cependant, l'algorithme présenté dans cette thèse possède un niveau de robustesse intéressant, de plus un grand nombre de perspectives peuvent être étudiées en fonction de l'application visée. Un algorithme robuste aux différents traitements applicables sur une vidéo nous semble difficile à réaliser. Il est important de fixer un contexte bien précis dans l'élaboration d'un algorithme de tatouage, afin d'obtenir un niveau de robustesse satisfaisant. Le tatouage fait preuve d'une maturité grandissante, aussi bien sur le plan théorique que sur le plan pratique. Il reste toutefois de nombreux aspects à approfondir. La combinaison de systèmes de protection (embrouillage, tatouage, cryptographie...) nous semble être une bonne solution afin de réaliser un système complet et sûr.

Glossaire

- APS : Analog Protection System
- ATSC : Advanced Television System Committee
- BCH : Bose Chaudhuri Hocquenghem
- BMA : Block Matching Algorithm
- CCETT : Centre Commun d'Études de Télédiffusion et de Télécommunication
- CENELEC : Comité Européen de Normalisation en Électronique
- COFDM : Codage OFDM
- CSA : Comité Supérieur de l'Audiovisuel
- CSF : Contrast Sensibility Function
- CSS : Content Scrambling System
- DCT : Discrete Cosinus Transform
- DBS : Direct Broadcasting by Satellite
- DES : Data Encryption Standard
- DFD : Displaced Frame Difference
- DSP : Digital Signal Processor
- DVB (S-T-C) : Digital Video Broadcasting (Satellite-Television-Cable)
- DVD : Digital Video Disc
- DWT : Discrete Wavelet Transform
- EBU : European Broadcasting Union
- EFO : Equation du Flot Optique
- ETSI : European Telecommunication Standards Institute
- FCC : Federal Communication Commission
- FFT : Fast Fourier Transform
- FIPS : Federal Institute Processing Standards
- FM : Frequency Modulation
- GAK : Government Access to Keys
- GBIM : Generalized Block-edge Impairment Metric
- GOP : Group Of Picture
- GPA : Générateur Pseudo-Aléatoire
- HDTV : Hi Definition TV
- HVS : Human Visual System
- IBIM : mproved Block-edge Impairment Metric

- JAWS : Just Another Watermarking System
- JND : Just Noticeable Difference
- IFS : Iterated Function System
- LPM : Log Polar Mapping
- MAD : Mean of Absolute Difference
- MCM : MultiCarrier Modulation
- MDS : Microwave Distribution Schemes
- MIT : Massachusetts Institute of Technology
- MMDS : Multipoint Multichannel Distribution System
- MPEG : Moving Picture Expert Group
- MPQM : Moving Pictures Quality Metric
- MSE : Mean of Square Error
- NIST : National Institute of Standards and Technology
- NTSC : National Television System Committee
- NVOD : Near Video On Demand
- OCR : Optical Character Recognition
- OEM : Original Equipment Manufacturer
- OFDM : Orthogonal Frequency Division Multiplex
- PGP : Pretty Good Privacy
- PS-BIM : Perceptually Significant Block-edge Impairment Metric
- PSNR : Peak Signal to Noise Ratio
- RSA : Ronald **R**ivest, Adi **S**hamir et Leonard **A**dleman
- RVB : Rouge, Vert, Bleu
- SAD : Sum of Absolute Difference
- SB : Scrambling Block
- SI : Système d'information
- SNR : Signal to Noise Ratio
- SSL : Secure Sockets Layer
- STV : Standard TV
- TVNT : Télévision Numérique Terrestre
- VCR : Video Cassette Recorder
- VOD : Video On Demand
- VSB-AM : Vestigial Sideband Amplitude Modulation

Bibliographie

[1] Information technology- generic coding of moving pictures and associated audio information. Technical report, Standard MPEG-2 : ISO/IEC 13818.

[2] Digital Terrestrial Broadcasting : The Government's Proposal. August 1995.

[3] A. A. Webster, C.T. Jones, M.H. Pinson, S.D. Voran, and S. Wolf. An objective video quality assesment system based on human perception. *SPIE Human Vision, Visual Processing, and Digital Display IV*, 1913(San Jose, CA), Feb. 1993.

[4] A. Ambroze, G. Wade, C. Serdean, M. Tomlinson, J. Stander, and M. Borda. Turbo code protection of video watermark channel. *IEE Proc. Vis. Image Signal Processing, Vol.148, No.1, February 2001, 54-58*.

[5] A. Basso, I. Dalgiç, F. A. Tobagi, and C.J. van den Branden Lambrecht. Study of mpeg-2 coding performance based on a perceptual quality metric. *Proc. PCS'96*, pages 263–268, Mars 1996.

[6] A. Bhardwaj, T.P. Pandey, and S. Gupta. Joint indexing and watermarking of video using color information. *MMSP 2001, October 3-5, 2001, CANNES, FRANCE*.

[7] A. Buisson. Implémentation efficace d'un codeur vidéo hiérarchique granulaire sur une architecture à processeurs multimedia. 2002.

[8] A. H.Tewfik and M.D. Swanson. Data hiding for multimedia personalization, interaction and protection. *IEEE Signal Processing Magazine*, pages 41–44, 1997.

[9] A. Kudelski. Method for Scrambling and Unscrambling a Video Signal, 20 December 1994 1994.

[10] A. Michelson. *Studies in optics*. PhD thesis, University of Chicago, 1927.

[11] A. Mojsilovic, J. Hu, and R.J. Safranek. Perceptually based color texture features and metrics for image retrieval. *IEEE International Conference on Image Processing*, pages 588–592, 1999.

[12] A. Piva, R. Caldelli, and A. De Rosa. A dwt-based object watermarking system for mpeg-4 video streams. *International Conference on Image Processing*, 3 :5 –8, 2000.

[13] A.B. Watson. The cortex transform : Rapid computation of simulated neural images. *Computer Vision, Graphiccs, and Image Processing*, 39(3) :311–327, 1987.

[14] A.B. Watson. Perceptual components architecture for digital video. *Journal of the Optical Society of America A*, 7(10) :1943–1954, 1990.

[15] A.B. Watson. Perceptual optimization of dct color quantization matrices. *IEEE International Conference on Image Processing*, 1, 1994.

[16] A.B. Watson. Image data compression having minimum perceptual error. *US Patent 5,426,512*, 1995.

[17] A.B. Watson. Image data compression having minimum perceptual error. *US Patent 5,629,780*, 1997.

[18] A.B. Watson. Toward a perceptual video quality metric. *SPIE Conference on Human Vision and Electronic Imaging, San Jose*, 3299 :139–147, 1998.

[19] A.B. Watson and A.J. Ahumada, Jr. A hexagonal orthogonal-oriented pyramid as a model of image representation in visual cortex. *IEEE Transactions on Biomedical Engineering*, 36(1) :97–106, 1989.

[20] A.B. Watson, G.Y. Yang, J.A. Solomon, and J. Villasenor. Visibility of wavelet quantization noise. *IEEE Transactions on Image Processing*, 6(8) :1164–1175, 1997.

[21] A.B. Watson, J. Hu, J.F. McGowan III, and J.B. Mulligan. Design and performance of a digital video quality metric. *Human Vision, Visual Processing, and Digital Display IX, Proc. SPIE*, 3644 :168–174, 1999.

[22] A.E. Jacquin. Image Coding Based on Fractal Theory of Iterated Contractive Image Transformations. *IEEE Transactions on Image Processing*, 2(1) :18–30, 1992.

[23] A.J. Ahumada, Jr. Computational image quality metrics : A review. *J. Morreale, ed., Society for Information Display International Symposium Digest of Technical Papers*, 24 :305–308, 1993.

[24] A.J. Ahumada, Jr., B.L. Beard, and R. Eriksson. Spatio-temporal discrimination model predicts temporal masking function. *Proceedings of SPIE Human Vision and Electronic Imaging, San Jose*, 3299 :120–127, 1998.

[25] A.J. Ahumada, Jr. and H.A. Peterson. Luminance-model-based dct quantization for color image compression. *Human vision, Visual Processing and Digital Display III, Proc. SPIE*, 1666 :365–374, 1992.

[26] A.J. Seyler and Z.L. Budrikis. Measurements of temporal adaptation to spatial detail vision. *Nature*, 184 :1215–1217, 1959.

[27] A.J. Seyler and Z.L. Budrikis. Detail perception after scene changes in television image presentations. *IEEE Transactions on Information Theory*, 11(1) :31–43, 1965.

[28] A.M. Eskicioglu and P.S. Fisher. Image quality measures and their performance. *IEEE Transactions on Communications*, 43(12) :2959–2965, Dec 1995.

[29] A.T.S. Ho, J. Shen, A.K.K. Chow, and J. Woon. Robust digital image-in-image watermarking algorithm using the fast hadamard transform. *IEEE International Symposium on Circuits and Systems (ISCAS) 2003, Bangkok, Thailand*, 25-28 May 2003.

[30] B. Girod. Eye movements and coding of video sequences. *SPIE Visual Communications and Image Processing*, 1001, 1988.

[31] B. Schneier. *Applied Cryptography - Second Edition*. John Wiley & Sons, 1996.

[32] B. Tao and B. Dickinson. Adaptive watermarking in the dct domain. *in Proc. Int. Conf. Image Processing (ICIP), Lausanne, Switzerland, Sept. 96*.

[33] B. Vassaux, P. Nguyen, S. Baudry, P. Bas, and J-M. Chassery. A survey on attacks in image and video watermarking. *SPIE, Seattle, Juillet 2002*.

[34] B. Vassaux, P. Nguyen, S. Baudry, P. Bas, and J-M. Chassery. Scrambling-based Watermarking for MPEG-4 Video. In *Eusipco*, Toulouse, 2002.

[35] B.G. Breitmeyer. Visual masking : An integrative approach. *New York : Oxford University Press*, 1984.

[36] C. Chok-Kwon and P. Lai-Man. Hybrid search algorithm for block motion estimation. *Proceedings of IEEE International Symposium on Circuits and Systems, ISCAS*, vol 4 :pp 297–300, 1998.

[37] C. De Vleeschouwer, J-F. Delaigle, and B. Macq. Invisibility and application functionalities in perceptual watermarking - an overview. *Proceedings of the IEEE*, 90 :64–77, 2002.

[38] C-S. Lu and H-Y.M. Liao. Video object-based watermarking : a rotation and flipping resilient scheme. *International Conference on Image Processing*, 2 :483 –486, 2001.

[39] C. T. Hsu and J.L. Wu. Digital watermarking for video. *in Proc. Of DSP'97, Santoniri, Greece*, July 97.

[40] C. Tae-Sun and K. Jong-Nam. Real-time video coding. *IEEE Transactions on Consumer Electronics*, vol 45(n. 2) :pp 417–426, 1999.

[41] Cabletronics. *Descrambler Manual*. 1986.

[42] C.J. van den Branden Lambrecht. *Perceptual Models and Architectures for Video Coding Applications*. PhD thesis, EPFL, 1996.

[43] Communication de la Commission au Parlement européen, au Conseil, au Comité économique et social et au Comité des régions. Assurer la sécurité et la confiance dans la communication électronique. *COM (97) 503*, 8 octobre 1997.

[44] D. Alleysson. *Le traitement du signal chromatique dans la rétine : un modèle de base pour la perception humaine des couleurs*. PhD thesis, Université Joseph Fourier - Grenoble 1, 1999.

[45] D. Delannay, C. De Roover, and B. Macq. Temporal alignement of video sequences for watermarking systems. *SPIE/IST 15th Electronic Imaging, Santa Clara, USA, January 20-24, Proc. Vol. 5020, , pp. 481-492*, 2003.

[46] D. Kundur and D. Hatzinakos. Digital watermarking using multiresolution wavelet decomposition. *Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing, Seattle, Washington, vol. 5, pp. 2969-2972, May 1998.*

[47] D. Marr. *Vision*. W.H. Freeman and Company, 1982.

[48] D. Raychaudhuri and L. Schiff. Unauthorized Descrambling of a Random Line Inversion Scrambled TV Signal. *IEEE Transactions on Communications*, pages 31,(6) :816–821, 1983.

[49] D.H. Kelly. Flicker fusion and harmonic analysis. *Journal of the Optical Society of America*, 51 :917–918, 1961.

[50] D.H. Kelly. Flickering patterns and lateral inhibition. *Journal of the Optical Society of America*, 59 :1361–1370, 1961.

[51] D.H. Kelly. Visual response to time-dependant stimuli. *Journal of the Optical Society of America*, 51 :422–429, 1961.

[52] D.L. Robie and R.M. Mersereau. Video error correction using steganography. *EURASIP Journal on Applied Signal Processing, v.2002 n.2, p.164-173, February 2002.*

[53] E. Hering. Zur lehre vom lichtsinne. *Wien. Math. Nat. Kl., 70 :169, 1875.*

[54] E. Koch and J. Zhao. Towards robust and hidden image copyright labelling. *IEEE Workshop on Nonlinear Signal and Image Processing*, 1995.

[55] E. Peli. Contrast in complex images. *Journal of the Optical Society of America*, 7(10) :2032–2040, 1990.

[56] E. Peli. In search of a contrast metric : Matching the perceived contrast of gabor patches at different phases and bandwiths. *Vision Research*, 37(23) :3217–3224, 1997.

[57] F. Bartolini, A. Manetti, A. Piva, and M. Barni. A data hiding approach for correcting errors in h.263 video transmitted over a noisy channel. *IEEE Multimedia Signal Processing Workshop 2001 (MMSP'01), Cannes, France.*

[58] F. Deguillaume, G. Csurka, J.J.K. O'Ruanaidh, and T. Pun. Robust 3d dft video watermarking. *Electronic Imaging / Session : Security and Watermarking*, 1999.

[59] F. Deguillaume, G. Csurka, and T. Pun. Countermeasures for unintentional and intentional video watermarking attacks. *Ping Wah Wong and Edward J. Delp eds., IS&T/SPIE's 12th Annual Symposium, Electronic Imaging 2000 : Security and Watermarking of Multimedia Content II, Vol. 3971 of SPIE Proceedings, San Jose, California USA, 23-28 January 2000. (Paper EI 3971-33)*.

[60] F. Hartung and B. Girod. Watermarking of uncompressed and compressed video. *Signal processing*, 66 :283–301, 1998.

[61] F.A.P. Petitcolas, R.J. Anderson, and M.G. Kuhn. Attacks on copyright marking systems. pages 218–238, 1998.

[62] F.C.M. Martins. Real-time frame rate adaptation based on warping of edge-preserving meshes. *Proceeding of IEEE International Conference on Image Processing, ICIP*, 1999.

[63] F.W. Campbell and J.G. Robson. Application of fourier analysis to the visibility of gratings. *Journal of Physiology (London)*, 197 :551–566, 1968.

[64] F.X.J. Lukas and Z.L. Budrikis. Picture quality prediction based on a visual model. *IEEE Trans. Comm.*, 30 (7) :1679–1692, 1982.

[65] G. Doerr and J.L. Dugelay. New intra-video collusion attack using mosaicing. *Proc of VCIP, Lugano, Switzerlan, July 2003*.

[66] G. Doerr and J.L. Dugelay. Secure background watermarking based on video mosaicing. *In Security, Steganography and Watermarking of Multimedia Contents VI, Proceedings of SPIE 5306, 2004*.

[67] G. Doër and J-L. Dugelay. A guide tour of video watermarking. *Signal processing : Image communication*, 18(4) :263–282, 2003.

[68] G. Girod and F. Hartung. Watermarking and Apparatus for Compressed Digital Video, Sept. 1998.

[69] G. Sharma. Digital Color Imaging. *IEEE Transactions on image processing*, 6(7) :901–932, July 1997.

[70] G.C. Langelaar. Overview of Protection Methods in Existing TV and Storage Devices. Technical Report Document-ID : SMS-TUD-609-1, 26 February 1996 1996.

[71] G.C. Langelaar, I. Setyawan, and R.L. Lagendijk. Watermarking digital image and video data. a state-of-the-art overview. *IEEE Signal Processing Magazine*, 17(5) :20 –46, 2000.

[72] G.C. Langelaar, J.C.A. van der Lub, and R. L. Lagendijk. Robust labeling methods for copy protection of images. *in Proc. Electronic Imaging, San Jose, CA, vol. 3022, pp 298-309*, Feb. 97.

[73] G.J. Sullivan and R.L. Baker. Motion compensation for video compression using control grid interpolation. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, vol 4 :pp 2713–2716, 1991.

[74] H. Bodmann, P. Haubner, and A. Marsden. A unified relationship between brightness and luminance. *CIE Proceedings Kyoto Session 1979, Paris*, pages 99–102, 1980.

[75] H. Min-Shiang, C. Chin-Chen, and H. Kuo-Feng. A watermarking technique based on one-way hash functions. *IEEE Transactions on Computer Electronics, Vol 45, 286(8), May 1999*.

[76] H.A. Peterson, H. Peng, and W.B. Pennebaker. Quantization of color image components in the dct domain. *Human vision, Visual Processing and Digital Display II, Proc. SPIE*, 1453 :210–222, 1991.

[77] H.R. Wu and M Yuen. A generalized block-edge impairment metric for video coding. *IEEE Signal Processing Letters 4(11) :317-320*, 1997.

[78] I. Avcibas. *Image Quality Statistics and Their Use in Steganalysis and Compression.* PhD thesis, Bogaziçi University, 2001.

[79] I. Avcibas, N. Memon, and B. Sankur. Steganalysis using image quality metrics. *IEEE Transactions on Image Processing*, 12 :221–229, February 2003.

[80] I. Newton. *Traité d'Optique*. Gauthier-Villars. Paris (1955), reproduction fac-similé de l'édition de 1722 : traduction de M.Coste.

[81] I.J. Cox, J. Killian, F.T. Leighton, and T. Shamoon. Secure Spread Spectrum Watermarking for Multimedia. *Transactions on Image Processing*, 6(12) :1673–1687, 1997.

[82] I.J. Cox and M.L. Miller. A review of watermarking and the importance of perceptual modeling. *Proceedings of Electronic Imaging*, February 1997.

[83] J. Dittman, M. Stabenau, and R. Steinmetz. Robust mpeg video watermarking technologies. *ACM Multimedia*, 1998.

[84] J. F. Delaigle, C. De Vleeschouwer, F. Goffin, and B. Macq. Low cost watermarking based on a human visual model. *Lecture Notes in Computer Science*, 1242, 1997.

[85] J. Fridrich. Methods for data hiding. *Center for Intelligent Systems & Department of Systems Science and Industrial Engineering*, 1997.

[86] J. Guo and P. Shi. Object based video watermarking scheme using inertia ellipse and shape adaptive dct. *MMSP 2002, St. Thomas, US Virgin Islands*, December 9-11, 2002.

[87] J. J. Chae and B. S. Manjunath. Data hiding in video. *Proceedings of 6th IEEE International Conference on Image Processing (ICIP'99), Kobe, Japan*, pages 311–15 vol.1, 24-28 Oct. 1999.

[88] J. Kelsey, B. Schneier, and N. Ferguson. Yarrow-160 : Notes on the Design and Analysis of the Yarrow Cryptographic Pseudorandom Number Generator. In Springer Verlag, editor, *Proc. of the 6^{th} Annual Int. Workshop*, volume 1758 of *Lecture Notes in Computer Science*, page 13, Kingston, August 1999.

[89] J. Lubin. A visual discrimination model for imaging system design and evaluation. *E. Peli(ed.), Vision Models for Target Detection and Recognition*, World Scientific Publishing :245–283, 1995.

[90] J. Lubin. A human vision system model for objective picture quality measurements. *International Broadcasting Convention*, 447 :498–503, 1997.

[91] J. Nieweglowski, T. Moisala, and P. Haavisto. Motion composated video sequence interpolation using digital image warping. *Proceedings of IEEE International Conference on Acoustic Speech, and Signal Processing, ICASSP*, vol 5 :pp 205–208, 1994.

[92] J. Zhang, J. Li, and L. Zhang. Video watermarking technique in motion vector. *Proceedings of XIV Brazilian Symposium on Computer Graphics and Image Processing*, pages 179–182, 2001.

[93] J.J.K. O'Ruanaidh and T.Pun. Rotation, scale and translation invariant spread spectrum digital image watermarking. *Signal Processing, Vol.66, No.3, May 1998, pp.303-317.*

[94] J.J.K. O'Ruanaidh, W.J. Dowling, and F.M. Boland. Watermarking digital images for copyright protection. *Vision, Image and Signal Processing*, 143 :250–256, 1996.

[95] JP.M.G. Linnartz and M. van Dijk. Analysis of the sensitivity attack against electronic watermarks in images. *IHW'98 - Proc. of the International Information hiding Workshop, April. 1998.*

[96] J.S. Payne, L. Hepplewhite, and T.J. Stonham. Perceptually based metrics for the evaluation of textural image retrieval methods. *IEEE International Conference on Multimedia Computing and Systems*, 2 :793–797, 1999.

[97] K. C. Ming, H. C. Heh, and H. M. Hsiung. A new mesh-based motion compensation algorithm for very low bit rate coding. *Proceedings of IEEE International Conference on Image Processing, ICIP*, vol 2 :pp 639–643, 1999.

[98] K. Su, D. Kundur, and D. Hatzinakos. A content-dependent spatially localized video watermark for resistance to collusion and interpolation attacks. *in Proc. IEEE Int. Conf. on Image Processing, 2001, vol. 1, pp. 818821*.

[99] K.T. Tan, M. Ghanbari, and D.E. Pearson. An objective measurment tool for mpeg video quality. *Signal Processing*, 70 :279–294, 1998.

[100] L. Brown. Comparing the Security of Pay-TV Systems for Use in Australia. *Australian Telecommunication Research Journal*, 24 N˚2 :1–8, 1990.

[101] M. Dudon, O. Avaro, and C. Roux. Triangular active mesh for motion estimation. *IEEE Transactions on Signal Processing : Image Coding*, vol 10 :pp 21–41, 1997.

[102] M. Kunt, G. Granlund, and M. Kocher. *Traitement numérique des images*, volume 2. Presses Polytechniques et Universitaires Romandes, 1993.

[103] M. Kutter, F. Jordan, and T. Ebrahimi. Proposal of a watermarking technique for hiding/retrieving data in compressed and decompressed video. JTC1/SC29/WG11 M2281, ISO/IEC Document, 1997.

[104] M. Kutter and F.A.P. Petitcolas. A fair benchmark for image watermarking systems. *To in E. Delp et al. (Eds), in vol. 3657, proceedings of Electronic Imaging '99, Security and Watermarking of Multimedia Contents, San Jose, CA, USA, 25-27 January 1999. The International Society for Optical Engineering*.

[105] M. Livingstone and D. Hubel. Segregation of form, color, movement, and depth : anatomy, physiologie, and perception. *Science*, 240, 1988.

[106] M. Maes. Twin peaks : The histogram attack to fixed depth image watermarks. *IHW'98 - Proc. of the International Information hiding Workshop, April. 1998.*

[107] M. Miyahara and K. Kotani. Block distortion in orthogonal transform coding - analysis, minimization and distortion measure. *IEEE Transaction on Communications*, 33(1) :90–96, 1985.

[108] M. Nadenau. *Integration of human color vision models into high quality image compression*. PhD thesis, EPFL, 2000.

[109] M. Pazarci and V. Dipçin. A MPEG2-transparent Scrambling Technique. *IEEE Transactions on Consumer Electronics*, 48(2) :345–355, 2002.

[110] M. Yuen and H. R. Wu. A survey of hybrid mc/dpcm/dct video coding distortions. *Signal Processing*, 70(3) :247–278, 1998.

[111] M.D. Swanson, B. Zhu, and A.H. Tewfik. Multiresolution scene-based video watermarking using perceptual models. *IEEE Journal on Selected Areas in Communications*, 16(4) :540–550, 1998.

[112] M.G. Kuhn. Analysis of the Nagravision Video Scrambling Method. Technical report, University of Cambridge, 23 august 1998.

[113] Ministère de l'économie et des finances. Présentation de la télévision numérique. Technical report, http ://www.industrie.gouv.fr/observat/innov/tele/hertzien/hertzien1-13.htm, 09/07/1999.

[114] M.S. Moore, J. Foley, and S.K. Mitra. Detectability and annoyance value of mpeg2 artifacts inserted into uncompressed video sequences. *Proceedings of the SPIE, Human Vision and Electronic Imaging V, San Jose*, 3959 :99–110, 2000.

[115] N. Checcacci, M. Barni, F. Bartolini, and S. Basagni. Robust video watermarking for wireless multimedia communications. *Proceedings IEEE Wireless Communications and Networking Conference (WCNC 2000)*, pages 1530–1535 vol.3.

[116] N. Nikolaidis and I. Pitas. Digital image watermarking : an overview. *IEEE International Conference on Multimedia Computing and Systems*, 1 :1 –6, 1999.

[117] P. Bas and B. Macq. A new video-object watermarking scheme robust to object manipulation. *International Conference on Image Processing*, 3 :526 –529, 2001.

[118] P. Bas, N. Le Bihan, and J-M Chassery. Color image watermarking using quaternion fourier transform. *Proc of ICASSP , 2003, Hong Kong, China*.

[119] P. J. Lindh and C.J. van den Branden Lambrecht. Effcient spatio-temporal decomposition for perceptual processing of video sequences. *Proceedings of the International Conference on Image Processing*, 3 :331–334, 1996.

[120] P. Lechat. *Représentation et codage d'objet vidéo, par maillage 2D déformables.* PhD thesis, Université de Rennes 1, 1999.

[121] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi. A no-reference perceptual blur metric. *Proceedings of the International Conference on Image Processing*, 3 :57–60, 2002.

[122] P. Pirat. Les bases techniques de la télévision numériques. Technical report, 2000.

[123] R. Dugad and N. Ahuja. A scheme for joint watermarking and compression of video. *IEEE Int. Conf. on Image Proc., v.2, pp.80-84, Sept. 2000*.

[124] R. Lancini, F. Mapelli, and S. Tubaro. A robust video watermarking technique in the spatial domain. *Video/Image Processing and Multimedia Communications 4th EURASIP-IEEE Region 8 International Symposium on VIPromCom*, pages 251 –256, 2002.

[125] R. Pastrana. Restauracion de imagenes digitales : Eliminacion de ruido por medio de un filtro geometrico. *Tesis de Licenciatura, Facultad de Ciencias Fisico Matematicas, Colegio de Electronica. BUAP*, 1996.

[126] R.A. young. Oh say, can you see ? the physiologie of vision. *Proceedings of SPIE Human Vision, Visual Processing and Digital Display*, 1453 :99–123, 1991.

[127] R.B. Wolfgang, C.I. Podilchuk, and E.J. Delp. Perceptual watermarks for digital images and video. *Proceedings of the IEEE*, 87 :1108–1126, 1999.

[128] R.J. Anderson and M.G. Kuhn. Tamper Resistant - a Cautionary Note. *Second USENIX Workshop on Electronic Commerce Proceedings, Oakland*, pages 1–11, 1996.

[129] R.L. Rivest, A. Shamir, and L. Adleman. A Method for Obtaining Digital Signatures and Public Key Cryptosystems. *Communications of the ACM*, 21(N°2) :120–126, 1978.

[130] S. Rihs. The influence of audio on perceived picture quality and subjective audio-video delay tolerance. *MOSAIC Handbook*, pages 183–187, 1996.

[131] S. Roche, J-L. Dugelay, and R. Molva. Multi-resolution Access Control Algorithm Based on Fractal Coding. *International Conference on Image Processing*, 3 :235–238, 1996.

[132] S. Suthaharan. l_∞-based distortion measure for blocking artifacts in digital video. *Proceedings of the 1998 Second IEEE International Caracas Conference on Devices, Circuits and Systems*, pages 235–238, 1998.

[133] S. Suthaharan. Perceptual quality metric for digital video coding. *Electronics Letters*, 39(5) :431–433, 2003.

[134] S. Valette. *Modèles de maillages déformables 2D et multirésolution surfaciques 3D sur une base d'ondelettes*. PhD thesis, INSA Lyon, 2002.

[135] S. Winkler. A perceptual distortion metric for digital color images. *Proceedings of the International Conference on Image Processing*, 3 :399–403, 1998.

[136] S. Winkler. *Vision models and quality metrics for image processing applications*. PhD thesis, EPFL, 2000.

[137] S. Winkler, A. Sharma, and D. McNally. Perceptual video quality and blockiness metrics for multimedia streaming applications. *Proc. 4th International Symposium on Wireless Personal Multimedia Communications*, pages 553–556, 2001.

[138] S. Winkler, E. Drelie Gelasca, and T. Ebrahimi. Toward perceptual metrics for video watermark evaluation. *in Proc. SPIE Applications of Digital Image Processing, vol. 5203, pp. 371-378, San Diego, CA, August 5-8*, 2003.

[139] S. Wolf. Features for automatic quality assesment of digitally transmitted video. *NTIA Report 90-264, US Department of Commerce*, June 1990.

[140] S. Wolf and A. Webster. Objective and subjective video performance testing of ds3 rate transmission channels. *ANSI T1A1 contr. No. T1A1.5/93-060*, April 1993.

[141] S. Wolf and M.H. Pinson. Spatial-temporal distortion metrics for inservice quality monitoring of any digital video system. *Proceedings of SPIE International Symposium on Voice, Video, and Data Communications*, 1999.

[142] S.A. Karunasekera and N.G. Kingsbury. A distortion measure for blocking artifacts in images based on human visual sensitivity. *IEEE Transactions on Image Processing*, 4(6) :713–724, 1995.

[143] S.J.P. Westen, R.L. Lagendijk, and J. Biemond. Spatio-temporal model of human vision for digital video compression. *Proceedings of Electronic Imaging*, 1997.

[144] S.R. Ely and S.R. Shuttleworth. Conditional Access Scrambling Techniques for Terrestrial UHF Television Broadcasts. In IEE, editor, *International broadcasting convention 1988*, volume 293, pages 318–322, London, 1988.

[145] T. Ebrahimi, M. Kutter, and F. Jordan. Proposal of a Watermarking Technique for Hiding/Retrieving Data in Compressed and Decompressed Video. *ISO/IEC Document, JTC1/SC29/WG11 (M2281)*, 1997.

[146] T. Eude and A. Mayache. An evaluation of quality metrics for compressed images based on human visual sensitivity. *Proceedings of Fourth International Conference on Signal Processing*, pages 779–782, 1998.

[147] T. Kalker and A.J.E.M. Janssen. Analysis of spomf detection. *IEEE-ICIP'99, October 1999, Proceedings 6th ICIP, 1999, Vol. 1, pp. 316-319.*

[148] T. Kalker, G. Depovere, J. Haitsma, and M. Maes. A video watermarking system for broadcast monitoring. *SPIE 3657, Security and Watermarking of Multimedia Content*, pages 103–112, 1999.

[149] T. Vlachos. Detection of blocking artifacts in compressed video. *Electronics Letters 36(13) :1106-1108*, 2000.

[150] T. Wiegand. Joint final committee draft of joint video specification. Technical report, ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC, 2002.

[151] T.N. Pappas and R.J. Safranek. *Perceptual criteria for image quality evaluation*. Handbook of Image and Video Processing (A. C. Bovik, ed.), p. 669-684, academic press edition, 2000.

[152] V. Darmstaeder, J.F. Delaigle, D. Nicholson, and B. Macq. A block based watermarking technique for mpeg2 signals : Optimization and validation on real digital tv distribution links. *in Proc. European Conf. Multimedia Applications, Services ansd Techniques-ECMAST'98, Berlin, Germany*, May 1998.

[153] V. Lenoir. Eurocrypt, a Successful Conditional Access System. *IEEE Transactions on Consumer Electronics*, 37(3) :432–436, 1991.

[154] V. Mangulis. Security of a Popular Scrambling Scheme for TV Pictures. Technical report, RCA Review, September 1980.

[155] V. Sedallian and G. Mathias. Les problèmes posés par la legislation française en matière de chiffrement. *Droit de l'informatique et des télécoms*, 98/4, 1998.

[156] W. Kanjanarin, P. Supasirisun, and T. Amornraksa. Access Limited Coding for Digital Video Streams. In *SCI*, Orlando, 2001.

[157] W. Zeng and S. Lei. Efficient Frequency Domain Selective Scrambling of Digital Video. In *ACM Multimedia*, volume 1, pages 285–294, Orlando, 1999.

[158] W.J. Tam and L. Stelmach. Visual masking at video scene cuts. *Proceedings of SPIE Human Vision, Visual Processing and Digital Display*, 2411 :111–119, San Jose, 1995.

[159] W.Y. Zou and P.J. Corriveau. Methods for evaluation of digital television picture quality. *SMPTE Technical Conference*, 1996.

[160] Y. Altunbasak and A. Murat Tekalp. Closed-form connectivity-preserving solutions for motion compensation using 2-D meshes. *IEEE Transactions on Image Processing*, 6(9) :1255, sep, 1997.

[161] Y. Nakaya and H. Harashima. Motion compensation based on spatial transformations. *IEEE Transactions on Circuits and Systems for Video Technology*, vol 4(n. 3) :pp 339–356, 1994.

[162] Y. Rubner and C. Tomasi. Texture metrics. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pages 4601–4607, 1998.

[163] Y. Wu and R. Deng. Adaptive collusion attack to a block oriented watermarking scheme. *Information and Communications Security, 5th International Conference, Huhehaote, China, October 10-13, 2003*.

[164] Z. Shan and M. Kai-Kuang. A new diamond search algorithm for fast block-matching motion estimation. *IEEE Transactions on Image Processing*, vol 9(n. 2) :pp 287–290, 2000.

[165] Z. Wang, A.C. Bovik, and B.L. Evans. Blind measurement of blocking artifacts in images. *Proc. ICIP, vol. 3, pp. 981-984, Vancouver, Canada*, 2000.

[166] Z. Yu, H.R. Wu, S. Winkler, and T. Chen. Vision model based impairment metric to evaluate blocking artifacts in digital video. *Proceedings of IEEE, vol. 90, no. 1, pp. 154-169, January*, 2002.

Articles et brevets

Conférences internationales :

- ACM Multimdédia 2002, Juan-les-pins (FRANCE)
  ”A scrambling method based on disturbance of motion vector”
  Yann Bodo, Nathalie Laurent, Jean-Luc Dugelay.
- ICIP 2003, Barcelonne (ESPAGNE)
  ”Watermarking Video, Hierarchical Embedding in Motion Vectors”
  Yann Bodo, Nathalie Laurent, Jean-Luc Dugelay.
- EUSIPCO 2004, Vienne (AUTRICHE)
  ” A comparative study of different modes of perturbation for Video Watermarking based on Motion Vectors”
  Yann Bodo, Nathalie Laurent, Jean-Luc Dugelay.

Conférence nationale :

- CORESA 2003, Lyon (FRANCE)
  ”Tatouage vidéo par marquage hiérarchique des vecteurs de mouvement”
  Yann Bodo, Nathalie Laurent, Jean-Luc Dugelay.

Journal :

- EURASIP Journal on Applied Signal Processing, special issue : ”Multimedia Security and Rights Management”
  ”Video Waterscrambling : Towards a Video Protection Scheme Based on the Disturbance of Motion Vectors”
  Yann Bodo, Nathalie Laurent, Christophe Laurent, Jean-Luc Dugelay.

Brevets :

- Brevet 1 (2001) : ”Procédés de brouillage et de débrouillage de signal vidéo, système, codeur, décodeur, serveur de diffusion, support de données pour la mise en oeuvre de ce procédé.”
  Yann Bodo, Nathalie Laurent, Christophe Laurent.

– Brevet 2 (2002) : ”Procédé de tatouage d’un signal vidéo, système et support de données pour la mise en oeuvre de ce procédé, procédé d’extraction du tatouage d’un signal vidéo, système pour la mise en oeuvre de ce procédé.”
  Yann Bodo, Nathalie Laurent.
– Brevet 3 (2002) : ”Procédé de tatouage d’une séquence d’images vidéo à sélection adaptative de la zone d’insertion du tatouage, procédé de détection d’un tatouage, dispositifs, support de données et programmes d’ordinateur correspondants.”
  Yann Bodo, Nathalie Laurent.
– Brevet 4 (2003) : ”Contrôle a posteriori de l’impact visuel d’un schéma de watermarking vidéo basé sur le marquage hiérarchique de vecteurs de mouvements.”
  Yann Bodo, Nathalie Laurent.
– Brevet 5 (2003) : ”Procédé et dispositif de détection du tatouage d’un signal numérique.”
  Yann Bodo, Nathalie Laurent, Sébastien Brangoulo.