Sécurité de l'information par stéganographie basée sur les séquences chaotiques

Dalia Battikh

Résumé

La stéganographie est l'art de la dissimulation de l'information secrète dans un médium donné (cover) de sorte que le médium résultant (stégo) soit quasiment identique au médium cover. De nos jours, avec la mondialisation des échanges (Internet, messagerie et commerce électronique), s'appuyant sur des médiums divers (son, image, vidéo), la stéganographie moderne a pris de l'ampleur. Dans ce manuscrit, nous avons étudié les méthodes de stéganographie LSB adaptatives, dans les domaines spatial et fréquentiel (DCT, et DWT), permettant de cacher le maximum d'information utile dans une image cover, de sorte que l'existence du message secret dans l'image stégo soit imperceptible et pratiquement indétectable.

La sécurité du contenu du message, dans le cas de sa détection par un adversaire, n'est pas vraiment assurée par les méthodes proposées dans la littérature. Afin de résoudre cette question, nous avons adapté et implémenté deux méthodes (connues) de stéganographie LSB adaptatives, en ajoutant un système chaotique robuste permettant une insertion quasi-chaotique des bits du message secret. Le système chaotique proposé consiste en un générateur de séquences chaotiques robustes fournissant les clés dynamiques d'une carte Cat 2-D chaotique modifiée.

La stéganalyse universelle (classification) des méthodes de stéganographie développées est étudiée. A ce sujet, nous avons utilisé l'analyse discriminante linéaire de Fisher comme classifieur des vecteurs caractéristiques de Farid, Shi et Wang. Ce choix est basé sur la large variété de vecteurs caractéristiques testés qui fournissent une information sur les propriétés de l'image avant et après l'insertion du message. Une analyse des performances des trois méthodes de stéganalyse développées, appliquées sur des images stégo produites par les deux méthodes de stéganographie LSB adaptatives proposées, est réalisée. L'évaluation des résultats de la classification est réalisée par les paramètres: sensibilité, spécificité, précision et coefficient Kappa.

Abstract

Steganography is the art of the dissimulation of a secret message in a cover medium such that the resultant medium (stego) is almost identical to the cover medium. Nowadays, with the globalization of the exchanges (Internet, messaging and e-commerce), using diverse mediums (sound, embellish with images, video), modern steganography is widely expanded. In this manuscript, we studied adaptive LSB methods of stéganography in spatial domain and frequency domain (DCT, and DWT), allowing of hiding the maximum of useful information in a cover image, such that the existence of the secret message in the stégo image is imperceptible and practically undetectable.

Security of the message contents, in the case of its detection by an opponent, is not really insured by the methods proposed in the literature. To solve this question, we adapted and implemented two (known) methods of adaptive stéganographie LSB, by adding a strong chaotic system allowing a quasi-chaotic insertion of the bits of the secret message. The proposed chaotic system consists of a generator of strong chaotic sequences, supplying the dynamic keys of a modified chaotic 2D Cat map.

Universal steganalysis (classification) of the developed methods of stéganography, is studied. On this question, we used the linear discriminating analysis of Fisher as classifier of the characteristic vectors of Farid, Shi and Wang. This choice is based on the wide variety of tested characteristic vectors that give an information about the properties of the image before and after message insertion. An analysis of the performances of three developed methods of steganalysis, applied to the produced stego images by the proposed adaptive methods of stéganography, is realized. Performance evaluation of the classification is realized by using the parameters: sensibility, specificity, precision and coefficient Kappa.

N° d'ordre : 15ISAR 14 / D15 - 14

Institut National des Sciences Appliquées de Rennes
20, Avenue des Buttes de Coësmes • CS 70839 • F-35708 Rennes Cedex 7
Tel : 02 23 23 82 00 - Fax : 02 23 23 83 96

Thèse

2015

Dalia BATTIKH

THESE INSA Rennes
sous le sceau de l'Université européenne de Bretagne
pour obtenir le titre de
DOCTEUR DE L'INSA DE RENNES
Spécialité : Traitement du signal et de l'image

présentée par
Dalia Battikh
ECOLE DOCTORALE : *Matisse*
LABORATOIRE : *IETR*

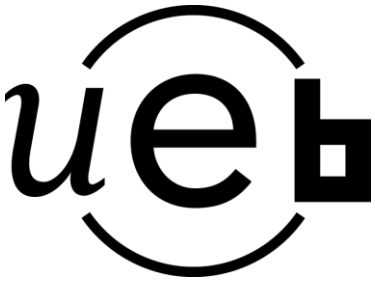
Sécurité de l'information par stéganographie basée sur les séquences chaotiques

Thèse soutenue le 18.05.2015 à Beyrouth (Liban)
devant le jury composé de :

Ahmad Shahin
Professeur + Université Libanaise / Président
William Puech
Professeur + Université de Montpellier II / Rapporteur
Chaouki Diab
Professeur + ISSAE-CNAM-Liban / Rapporteur
Ina Taralova
Maître de Conférences à l'ECN/ Examinatrice
Safwan El Assad
Maître de Conférences HDR + Polytech'Nantes / Co-encadrant
Bassem Bakhache
Maître de Conférences + Université Libanaise / Co-encadrant
Mohamad Khalil
Professeur + Université Libanaise / Directeur de thèse
Olivier Déforges
Professeur + INSA de Rennes / Directeur de thèse

Sécurité de l'information par stéganographie basée sur les séquences chaotiques

Dalia Battikh



En partenariat avec

Table de matières

Liste des tables

Liste des figures

Introduction générale

Le problème d'échange de données secrètes a toujours existé, et ce depuis la naissance des grandes civilisations. La cryptographie offre un moyen efficace pour protéger les données secrètes en les rendant inintelligibles aux personnes non autorisées, cependant, le simple fait de communiquer avec des messages chiffrés attire l'attention. Cela peut être problématique lorsqu'il s'agit d'un canal de communication surveillé par une tierce personne, qui peut, au moindre soupçon, détruire la communication entre les deux parties. Dans un tel cas de figure, une communication, contenant un message secret, entre deux personnes doit sembler normale aux yeux de la personne qui contrôle le canal. Pour ce genre de scénario, la stéganographie représente la solution alternative à la cryptographie. La stéganographie, ou la science de communication secrète, est un procédé permettant de cacher un message secret au sein d'un médium hôte (original, cover) anodin, de telle sorte que le médium résultant (stégo) semble inaffecté (dissimulation indétectable) par l'insertion du message secret. Autrement dit, l'objectif est de rendre difficile, ou impossible, la distinction entre un médium original et un médium modifié par l'insertion d'un message secret.

De manière analogue à la cryptographie, dont la discipline duale est la cryptanalyse visant à décrypter le message chiffré, la stéganographie a également comme discipline duale la stéganalyse. L'objectif principal de la stéganalyse est de détecter la présence d'un message caché, et aussi dans la mesure du possible, d'avoir accès à son contenu. Le concept clé de la sécurité d'un système de stéganographie est son indétectabilité visuelle mais aussi et surtout statistique.

De nos jours, avec le développement de l'Internet et l'explosion des médiums numériques (sons, images, et vidéos) partagés sur les différents réseaux de communication, la stéganographie devient une pratique populaire et accessible à toute personne souhaitant communiquer de façon discrète avec d'autres personnes. La communauté scientifique s'est alors particulièrement intéressée à cette discipline. Les chercheurs ont mis en évidence que la stéganographie appliquée aux médias numériques actuels représente un véritable challenge faisant appel à de nombreuses disciplines : mathématiques, statistiques, traitement du signal, théorie de l'information, et théorie des jeux. Parmi les médiums qui sont très adaptés pour la dissimulation de l'information, on distingue les images numériques. Ce type de médium étant très couramment échangé sur Internet, une grande majorité des travaux de recherche lui est consacrée. Dans ce manuscrit, nous nous intéressons également aux images numériques en tant que médium cover.

La stéganographie possède trois grandes propriétés qui caractérisent son utilisation: la robustesse, l'invisibilité et la capacité. La robustesse assure que l'information secrète ne peut pas être détruite sans dégrader fortement l'image. L'invisibilité vise à ce que l'image stégo ne soit pas perturbée par l'information secrète insérée. La capacité définit la quantité d'information qui peut être intégrée dans le support sans détérioration visible. Ces trois caractéristiques sont en relation étroite et inverse. Par exemple, l'amélioration de la capacité a généralement une influence négative sur l'invisibilité.

L'information secrète peut être dissimulée essentiellement dans deux domaines d'insertion possible : le domaine spatial, et le domaine fréquentiel. Pour le premier, il s'agit, d'effectuer la dissimulation directement dans les bits des pixels de l'image porteuse. La technique LSB (Last Significant Bit) est l'une des techniques spatiales existantes. Elle consiste à cacher un message secret dans les bits de poids faibles des pixels de l'image, de sorte que les distorsions apportées par le processus d'insertion restent imperceptibles. En effet, pour l'œil humain, les variations de la valeur du LSB sont quasiment imperceptibles. Les méthodes d'insertion directe dans ce domaine sont peu coûteuses en termes de temps de calcul puisqu'elles ne nécessitent pas une étape préalable de transformation. A l'inverse, les méthodes de stéganographie spatiale sont en général sensibles aux attaques. Le domaine fréquentiel est l'espace obtenu après une transformée en DFT (Discret Fourier Transform), en DCT (Discret Cosine Transform), ou en ondelettes discrètes DWT (Discret Wavelet Transform). Les techniques stéganographiques fonctionnant dans le domaine fréquentiel encodent les données à travers la fréquence globale de l'image, et cachent l'information dans des zones de l'image qui sont moins sensibles à la compression, au recadrage et aux diverses traitements de l'image. Ceci permet d'avoir une robustesse largement accrue.

Pour la confidentialité du message secret, de nombreux travaux de recherche ont montré l'apport et l'intérêt de l'utilisation des signaux chaotiques. En effet, des caractéristiques importantes des signaux chaotiques, telles que : bonnes propriétés cryptographiques, reproductibilité à l'identique (déterministes), et sensibilité aux conditions initiales et aux paramètres du système, incitent à leur utilisation dans les systèmes de communications pour la sécurité des données. La conception et la réalisation de générateurs de séquences chaotiques est un domaine de recherche relativement récent et il est de plus en plus utilisé. Dans ce contexte, le générateur de séquence chaotique représente un élément fondamental dans toutes applications touchant à la sécurité de l'information, telles que les crypto-systèmes, les systèmes de génération des clés secrètes, les systèmes de tatouage et de stéganographie.

Notre travail dans cette thèse consiste à proposer de nouvelles techniques stéganographiques, spatiale et fréquentielle, dont la sécurité du message secret est basée sur un système de génération de séquences chaotiques de valeurs entières. Il s'agit ensuite d'étudier la performance de ces méthodes et leur robustesse vis à vis de méthodes développées de la cryptanalyse universelle.

Structure de la thèse

Dans le premier chapitre, nous introduisons les concepts et les outils de base qui seront nécessaires à la compréhension de la suite de la thèse. Nous commençons par l'introduction des différents domaines de sécurité de l'information : le tatouage, la cryptographie, et la stéganographie. Puis, nous discutons les caractéristiques d'un système stéganographique et nous définissons les domaines spatial et fréquentiel, ainsi que les différentes transformées fréquentielles DCT, DFT, et DWT. Nous introduisons, ensuite l'essentiel de la théorie du chaos, avec notamment l'un de ses apports dans la réalisation des systèmes de génération de signaux chaotiques (cartes chaotiques de base) ainsi que leurs propriétés, plus

particulièrement celles liées à la sécurité. Dans la dernière partie du chapitre, nous présentons la stéganalyse universelle. Cette dernière utilise la théorie des ondelettes pour extraire les paramètres caractéristiques, et la classification de Fisher pour discriminer les classes cover et stégo.

Le deuxième chapitre porte sur les méthodes stéganographiques spatiales. Les méthodes les plus utilisées dans la littérature sont étudiées, puis après analyse de deux méthodes les plus répandues, nous pointons leur faiblesse vis à vis de la sécurité du message caché. Nous proposons alors un système chaotique permettant de sécuriser les positions des bits du message secret inséré. La comparaison entre les méthodes, avant et après les améliorations proposées, est faite et les résultats obtenus sont discutés.

Le troisième chapitre expose les méthodes de stéganographie fréquentielles: DCT, DWT, et par étalement de spectre. Nous présentons les méthodes les plus utilisées et proposons des modifications afin de rendre ces méthodes plus efficaces. De même, la sécurité de ces méthodes est augmentée par l'utilisation de systèmes chaotiques proposés. Ces derniers permettent de disperser de façon quasi-chaotique le message secret dans les coefficients des transformées. Ceci hausse le niveau de la sécurité ainsi que la capacité des méthodes proposées. Des tests d'imperceptibilité, de qualité, et de capacité sont effectués afin de montrer l'efficacité des améliorations proposées.

Le quatrième chapitre est dédié à la stéganalyse universelle des méthodes stéganographiques spatiales. Cette dernière s'appuie sur un processus d'apprentissage utilisant les paramètres caractéristiques extraits des images originales et des images stégos. Trois méthodes de séganalyse, basées sur les vecteurs caractéristiques de Farid, Shi et Wang et sur le processus de classification de Fisher sont implémentées. L'évaluation des performances de la classification s'appuie sur l'utilisation des paramètres: sensibilité, spécificité, précision et coefficient Kappa.
Nous clôturons ce manuscrit en présentant une conclusion générale à ces travaux, ainsi que les perspectives envisagées.

Chapitre 1

CONTEXTE DE L'ETUDE, DEFINITIONS ET GENERALITES

1. CONTEXTE DE L'ETUDE, DEFINITIONS ET GENERALITES

1.1 Introduction

Les avancées technologiques en informatique et télécommunications ont contribué à soulever une multitude de problèmes liés à la protection (sécurité) de l'information, en permettant ainsi un large développement scientifique et technique en réponse aux défis soulevés. Parmi les questions posées, nous citons : les virus informatiques, l'autorisation d'accès, la protection des droits d'auteurs, et la vérification de l'intégrité des données. A cela il est possible d'ajouter des questions telles que l'authentification, l'accès conditionnel, le tatouage numérique, la signature numérique, la communication secrète et la stéganographie.
Nous nous focalisons dans nos travaux sur la question de la sécurité de la transmission d'informations confidentielles sous forme numérique, basée principalement sur la stéganographie.
L'objectif de la stéganographie est de cacher un message secret dans un média « innocent » tel qu'une image, un son, une vidéo, etc. Cette technique de transmission sécurisée de l'information permet de rendre le message secret indétectable, mais aussi de rendre le secret inintelligible s'il est détecté.
Le but de notre étude est d'étudier et de développer des techniques efficaces de stéganographie dans les domaines spatial et fréquentiel, incluant un procédé robuste de protection du message secret s'il est détecté. Dans cette perspective, la sécurité du message secret est basée sur la génération de séquences chaotiques robustes. Cette nouvelle technique permet de renforcer significativement la sécurité d'un système de transmission de données.
Aussi, afin de tester la robustesse des méthodes de stéganographie développées, nous étudions, développons et testons des méthodes de stéganalyse dites universelles.
Ce chapitre introduit les notions liées aux méthodes et techniques diverses de la sécurité de l'information, ainsi que les outils nécessaires à la compréhension des travaux réalisés dans cette thèse. Nous commençons par rappeler brièvement les diverses techniques les plus répandues de la transmission sécurisée de l'information. Puis, nous rappelons les propriétés pour les domaines spatial et fréquentiel des systèmes de stéganographie. Nous présentons ensuite les principes des systèmes générant des signaux chaotiques et leurs propriétés, en particulier pour celles liées à la sécurité permettant de rendre les systèmes de stéganographie plus robustes

1.2 Transmission sécurisée de l'information

Dans le domaine de la transmission sécurisée de l'information, si l'on reconnaît la cryptographie comme l'art des codes secrets, la stéganographie est l'art de la dissimulation. Alors que la cryptographie consiste en une écriture indéchiffrable d'un message ou d'une information (ainsi rendue secrète), la stéganographie va plutôt s'attacher à cacher un message dans un contenu pour qu'il soit, non seulement indéchiffrable, mais imperceptible. Quant au tatouage, cet autre « principe de camouflage » offre des solutions techniques pour faire face aux problèmes de protection des droits et des copies. La figure 1.1, donné par [Cheddad et al., 2010] résume les différentes techniques de la sécurité de l'information.

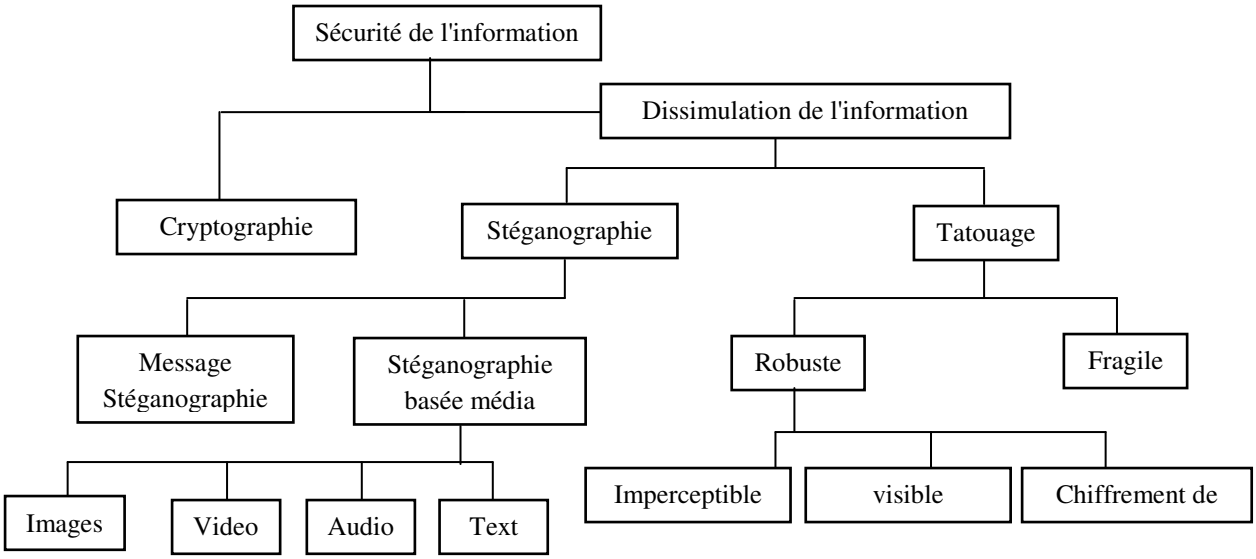


Figure 1.1 : Techniques de la sécurité de l'information

Selon la figure 1.1, trois méthodes principales traitent de la sécurité de l'information : la cryptographie, la stéganographie, et le tatouage numérique. Les deux derniers assurent la dissimulation de l'information.
Nous nous rappelons ci-dessous les différentes notions utilisées dans les différentes techniques liées à la sécurité de l'information :

1.2.1 Cryptographie

C'est la science d'écriture d'un message en code secret afin de préserver sa sécurité et sa confidentialité. Le but est donc de brouiller un message afin de le rendre incompréhensible pour les personnes non autorisées. Le message initial est appelé message en clair et, après chiffrement, message chiffré ou cryptogramme. Le chiffrement et le déchiffrement sont réalisés principalement à partir d'algorithmes, en utilisant des clés secrètes ou publiques.

1.2.2 Tatouage numérique

Le tatouage comprend deux types : le tatouage fragile et le tatouage robuste.
Le tatouage numérique fragile n'est utilisé que pour prouver l'authenticité des documents et l'intégrité des données. La protection de la marque "tatoue" étant très faible, le message qu'il transporte n'est pas vraiment important. La marque fait partie du document et ainsi, lorsque celui-ci est modifié, le marquage l'est également. Ce type de tatouage pose quand même un problème, car même s'il permet de prouver qu'un document a subi une transformation, il ne prouve pas pour autant qui est l'auteur du document.
Le tatouage robuste est plus dur à contourner et doit résister à diverses attaques. Il doit posséder les deux propriétés suivantes :

- la marque doit être très résistante vis à vis des différentes attaques connues telles que : ré-échantillonnage, impression puis scanne, à la compression, à la coupure, aux bruits et aux changements de format.

- la marque doit être facilement reconnaissable après extraction et ceci malgré le dommage subi par les différentes attaques. Dans le cas contraire, la marque pourrait être incompréhensible ou avoir changée de sorte qu'elle n'ait plus rien à voir avec celle insérée à l'origine. Ce type de tatouage est utilisé surtout dans les applications de protection de copyright et de contrôle de copies.

1.2.3 Empreinte digitale (fingerprinting) :

Le but de cette application est de pouvoir contrôler et faire le suivi des copies de document. Cela implique de créer une marque originale pour chaque document distribué. Les marques doivent être très robustes, afin de résister aux attaques ayant pour but de les détruire.

1.2.4 Stéganographie

La stéganographie (du grec steganos, couvert et graphein, écriture) est l'art de cacher un message secret au sein d'un autre message porteur (texte, image, son, vidéo...) de caractère anodin, de sorte que l'existence même du secret en soit dissimulée. Alors qu'avec la cryptographie, la sécurité repose sur le fait que le message chiffré soit incompréhensible pour les personnes non autorisées, avec la stéganographie, la sécurité repose sur le fait que la présence même d'un message secret ne sera sans doute pas soupçonnée et détectée.

Il existe deux types de stéganographie :

- la sténographie linguistique,
- la stéganographie technique.

1.2.4.1 Stéganographie linguistique

La littérature sur la stéganographie linguistique, dans laquelle les propriétés linguistiques d'un texte sont modifiées pour cacher l'information, est faible par rapport à d'autres médias (Bergmair, 2007). La raison probable est qu'il est plus facile d'apporter des modifications aux médias non linguistiques dans lesquels le message secret sera indétectable par un observateur. Les différentes formes de la stéganographie linguistique sont :

- Sémagramme

La forme la plus connue en stéganographie linguistique est le sémagramme. De cette manière, le système stéganographique échappe totalement à l'observateur. Alfred de Musset est l'utilisateur le plus connu de ce procédé. Il a entretenu, entre 1833 et 1834, une relation secrète avec Georges Sand au travers de poèmes qu'il lui envoyait.

- Acrostiche

Ce procédé permet de transmettre des données au travers de lettres initiales dans chaque vers de poème et qui, lus de haut en bas, forment un mot ou une expression. Elle a de nombreuses variantes (mot placé dans des vers ou des chapitres,…).

- Ponctuation

L'utilisation de points, hauteur de lettres et virgules par les prisonniers de guerre a également permis de transmettre des messages à leur famille.

- Nulles

Les codes camouflés, aussi appelés les nulles, consistent à marquer d'un signe particulier certaines lettres d'un texte (par des piqûres d'aiguilles sur ou sous les lettres). Il suffit alors de rassembler les lettres marquées pour former un mot.

- Insertion d'erreurs

Mise en valeur de l'information au travers d'erreurs ou de formes de style dans un texte.

Ces différents procédés restent néanmoins difficiles et longs à réaliser et laissent vite suspecter la possibilité d'un message dissimulé. De nombreuses censures ont été ainsi appliquées afin de limiter l'usage de ces techniques.

1.2.4.2 La stéganographie technique

La stéganographie technique regroupe toutes les techniques qui ne jouent pas sur les mots. La stéganographie technique est intéressante car elle permet de dissimuler des données dans plusieurs types de médias.

- Audio

Afin de transmettre de l'information de manière cachée dans du son, différentes techniques existent et se basent sur le fait qu'un son affecte la perception d'un autre :

- un son plus fort peut en cacher un autre,
- un son peut être caché temporairement lorsqu'il est moins fort et qu'il est placé avant ou après un son plus fort.

Il est également possible de cacher des données en utilisant la représentation des notes. Prenons comme exemple un livre de Gaspar Schott, Schola Steganographica, où l'auteur explique que des messages ont été cachés dans de la musique, de sorte qu'une note correspondande à une lettre. J.S Bach, lui, utilisait le nombre d'occurrences de notes qui apparaissait. John Wilkin a même démontré que deux musiciens discutaient au travers de leur musique comme si leurs instruments parlaient.

- Images, vidéo

Enfin, une image ou une vidéo peuvent également contenir un message. Une image est constituée de pixels. Il est possible d'insérer des bits du message secret à l'intérieur sans que ces modifications soient perceptibles à l'œil humain.

Aujourd'hui la stéganographie a énormément évolué et présente un intérêt grandissant. En effet, depuis 1996, cet art est de plus en plus employé, et on lui a porté une attention particulière depuis 2011 puisque différents groupes terroristes sont suspectés d'y avoir eu recours.

1.3 Supports de la stéganographie et leur représentation

1.3.1 Le format d'images numériques

1.3.1.1 Représentation des couleurs

La représentation **YUV** est très utilisée, principalement dans tout ce qui est compression d'image. Y représente la luminance de la couleur, et U et V, la chrominance de cette couleur dans le rouge et le bleu. Cette représentation est utile, car l'œil est plus sensible aux variations de luminances qu'aux variations de chrominance. Séparer ces trois composantes permettra donc de pouvoir dégrader plus les chrominances, tout en conservant mieux la luminance.

La transformation RGB-YUV est un système linéaire [Brun, 2003] :

- Y = 0.299*R + 0.587*G + 0.114*B
- U = -0.169*R - 0.331*G + 0.500*B + 128.0
- V = 0.500*R - 0.419*G - 0.081*B + 128.0

La transformation inverse existe.

1.3.1.2 Format d'images

Il faut distinguer les différentes représentations d'une image matricielle. Dans un fichier, pour le stockage et l'échange des données, l'image est le plus souvent compressée et stockée dans un format graphique. Les principaux formats matriciels sont Windows bitmap (BMP), Graphics Interchange Format (GIF), Tagged Image File Format (TIFF), Portable Network Graphics (PNG) et Joint Photographic Experts Group (JPEG).
Chaque format a ses caractéristiques. Pour bien choisir celui qui correspondra à ce que vous souhaitez faire de vos images, il est essentiel de connaître la profondeur des couleurs. Exprimée en bits, elle correspond au nombre de valeurs chromatiques que peut prendre chaque pixel de l'image.

Format BMP (ou *Bitmap*) : est un format universel, non compressé, développé par Microsoft et IBM. Il permet une restitution fidèle des couleurs de l'image d'origine, la contrepartie est le poids élevé du fichier généré.

Format GIF (*Graphics Interchange Format*) : également très répandu dans le web, ce format propriétaire utilise un système de couleurs indexées. Il est ainsi possible de n'utiliser que des valeurs chromatiques spécifiques, ce qui permet d'optimiser au maximum le poids du visuel. En contrepartie, l'apparence de visuels affichant de nombreuses couleurs se trouve fortement dégradée. Ce format permet également de créer des animations image par image.

Format TIFF (*Tag(ged) Image File Format*) : autre format universel, développé par Adobe. C'est un format qui permet l'utilisation de multiples types de compressions, et types de colorimétries (RVB, CMJN (Cyan, Magenta, Jaune et Noir), N&B (Noir et Blanc), etc.). Tout comme le Bitmap, le TIFF non compressé permet une conservation exacte des couleurs au détriment d'un poids de fichier élevé.

Le format PNG (*Portable Network Graphics*) : ce format standardisé par le W3C (World Wide Web Consortium) a été établi pour contourner la License propriétaire existante sur le

format GIF. Le PNG possède donc exactement les mêmes caractéristiques que le GIF, et permet de plus l'enregistrement de 1 à 48bits, et la gestion de la transparence (couches alpha). En revanche il n'est pas possible de faire des animations.

Le format JPEG (*Joint Photographic Experts Group*) : très utilisé dans le web, ce format a été établi par un comité d'experts qui édite des normes de compression pour les images fixes. Ce format compressé altère grandement la qualité des images d'origine, mais permet d'avoir une restitution relativement fidèle des couleurs et un poids de fichier relativement léger.

1.4 Propriétés des systèmes de stéganographie

1.4.1 Capacité

La capacité d'insertion d'un système de stéganographie est défini par la taille en bits du message secret qui peut être intégré dans un média de taille donnée. La capacité d'insertion relative est le rapport entre la taille du message secret à dissimuler et la taille du médium utilisé. Dans le domaine spatial, pour une image numérique, la capacité d'insertion relative peut être exprimée en nombre de bits de message secret insérés par pixel (bpp). Dans le domaine fréquentiel, par exemple insertion dans les coefficients quantifiés d'une image JPEG, la capacité d'insertion relative peut être exprimée par le nombre des bits du message secret à insérer par chaque coefficient DCT quantifié non-nul (bpc) [Kouider, 2013]. Notons que dans ce cas, comme le nombre de coefficients non-nuls dépend du contenu de l'image, la capacité d'insertion est variable d'une image à l'autre.

1.4.2 Sécurité

Toutes les exigences de sécurité pour les systèmes cryptographiques peuvent (doivent) également être considérées pour les systèmes de stéganographie. Cela signifie que la sécurité de l'algorithme de stéganographiene ne doit pas s'appuyer seulement sur l'algorithme, qui devrait être publique, mais sur le caractère secret de la clé. Dans la stéganographie, il ne devrait pas être possible de distinguer une image d'origine d'une image stego si la clé est inconnue.
Par ailleurs, les modifications apportées sur l'image originale afin de pouvoir incorporer le message secret ne devrait pas modifier les propriétés statistiques de l'image. La technique qui étudie la sécurité des systèmes de stéganographie est la stéganalyse.

1.4.3 Robustesse

Elle quantifie la résistance du message dissimulé aux diverses attaques (transformations) apportées au médium stégo.

1.5 Domaines de stéganographie

La stéganographie est divisée en deux domaines, spatial et fréquentiel. Dans le domaine spatial, le message secret est inséré dans les pixels de l'image porteuse, tandis que dans le domaine fréquentiel, les pixels sont transformés en coefficients, et le message secret est inséré dans ces coefficients [Saejung et al., 2013] [Goel et al., 2013].

1.5.1 Domaine spatial

La stéganographie spatiale consiste à faire changer des bits de pixels de l'image pour insérer les bits du message secret. La technique LSB est l'une des techniques la plus simple et la plus répandue. Elle consiste à cacher un message secret dans les bits de poids faible des pixels de l'image, de sorte que les distorsions apportées par le processus d'insertion restent non perceptibles. La raison est que pour l'œil humain, les variations de la valeur du LSB sont quasiment imperceptibles. L'insertion de bits de message secret peut être faite séquentiellement ou de façon pseudo aléatoire. La stéganographie par substitution de LSB [Cheddad et al., 2010] et la stéganographie par correspondance de LSB [Zhang et al., 2009] sont des exemples de techniques de stéganographie dans le domaine spatial.

1.5.2 Domaine fréquentiel

Le message est inséré dans les coefficients transformés de l'image, ce qui a pour effet d'apporter plus de robustesse contre les attaques. La stéganographie fréquentielle est une technique essentielle de dissimulation de l'information secrète : de nos jours, la plupart des systèmes de stéganographie opèrent dans le domaine fréquentiel. La stéganograhie fréquentielle va ainsi permettre de cacher l'information dans des zones de l'image moins sensibles à la compression, au recadrage et aux divers traitements de l'image, [Raja et al., 2004].
Dans la suite nous rappelons les principes des différentes techniques de transformation dans le domaine fréquentiel.

1.6 Techniques fréquentielles

1.6.1 Transformée de Fourier

La transformée de Fourier est un outil important de traitement d'images, utilisée pour décomposer une image suivant ses composantes en sinus et cosinus. La sortie de la transformation représente l'image dans le domaine fréquentiel, tandis que l'image d'entrée est dans le domaine spatial équivalent. Chaque coefficient de l'image dans le domaine de Fourier représente une fréquence particulière contenue dans l'image. La transformée de Fourier est utilisée dans une large gamme d'applications telles que l'analyse, le filtrage, la reconstruction ou la compression d'images.

Pour une image dans le domaine spatial de taille M×N, sa transformée de Fourier est donnée par l'équation :

$$F(k,l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n) e^{-i2\pi(\frac{km}{M}+\frac{ln}{N})} \quad (1.1)$$

Où *f (m, n)* est la valeur du pixel de l'image se trouvant dans la ligne *m* et la colonne *n*, et *F (k, l)* est la valeur du coefficient correspondant dans l'espace de Fourier se trouvant dans la ligne *k* et la colonne *l*. Cette transformée est parfois utilisée pour coder des fichiers audio [Mazumde et Hemachandran, 2013].

L'inverse de la transformée de Fourier est donné par l'équation :

$$f(m,n) = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} F(k,l) e^{i2\pi(\frac{km}{M} + \frac{ln}{N})} \quad (1.2)$$

1.6.2 Transformée en cosinus discrète (DCT)

La transformée en cosinus discrète est similaire à la transformée de Fourier mais elle n'a pas de composante imaginaire. Pour une image de taille M×N la **DCT** est donnée par :

$$DCT_{kl} = \alpha_k \alpha_l \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} C_{mn} \cos\frac{\pi(2m+1)k}{2M} \cos\frac{\pi(2n+1)l}{2N} \quad (1.3)$$
$$0 \le k \le M-1 \quad 0 \le l \le N-1$$

$$\alpha_k = \begin{cases} 1/\sqrt{M} & pour\ k = 0 \\ \sqrt{2/M} & 1 \le k \le M-1 \end{cases} \qquad \alpha_l = \begin{cases} 1/\sqrt{N} & pour\ l = 0 \\ \sqrt{2/N} & 1 \le l \le N-1 \end{cases}$$

Où DCT_{kl} sont les coefficients DCT de la ligne k et la colonne l et C_{mn} sont les valeurs des pixels de l'image d'origine de la ligne m et la colonne n [Goel et al., 2013] [Mazumde et Hemachandran, 2013].

Pour passer du domaine fréquentiel au domaine spatial, il suffit d'appliquer la transformation inverse de la cosinus discrète (fonction IDCT).

1.6.3 Transformée en ondelettes DWT

La transformée en ondelettes est un outil très utilisé en traitement d'images dans des domaines tels que le tatouage numérique [Kaewkamnerd et Rao, 2000], le débruitage [Benabdellah et al., 2003], la compression [Elbasi et Eskicioglu, 2006], la segmentation d'images, l'analyse de texture, la détection de contours ou encore l'extraction d' information manuscrite [Douzi et al., 2001]. La transformée en ondelettes, en comparaison avec la transformée de Fourier, offre une bonne localisation espace-fréquence. La représentation à différents niveaux de résolutions permet l'extraction du contenu principal de l'image en un nombre restreint de coefficients, tout en localisant précisément les discontinuités

Principe de la transformée en ondelettes

La transformée de Fourier permet d'analyser le comportement fréquentiel d'un signal mais perd toutes les informations relatives au temps. De ce constat est venue l'idée d'utiliser une transformée de Fourier « à court terme », principe développé par GABOR en 1946. Cette transformée consiste à considérer le signal autour d'un temps *t*., et de réaliser une analyse par une fenêtre glissante $g(u-t)$ centrée sur cette instant *t* , et en appliquant sa transformée de Fourier définie par l'équation (1.4) suivante :

$$\int x(u) g(u-t) e^{-i2\pi vu}\, du \quad (1.4)$$

Avec

$$x(u) \in L^2(\mathbb{R})$$

Le glissement de cette fenêtre le long du signal permet de mesurer le contenu spectral au cours du temps. Cette méthode d'analyse par la transformée de Fourier à court terme a toutefois quelques inconvénients.

• Si l'on considère une fenêtre temporelle large, on pourra constater une bonne résolution fréquentielle contre une mauvaise résolution temporelle.
• Dans le cas contraire, on constatera une résolution temporelle précise contre une mauvaise résolution fréquentielle.

C'est pour cela que Morlet a introduit la transformée en ondelettes, conçue initialement pour être adaptative. Cette transformée permet de déterminer les différentes composantes fréquentielles d'un signal donné, ainsi que leur localisation spatiale ou temporelle.

Par définition, les ondelettes sont des fonctions générées par dilatations et translations à partir d'une fonction appelée ondelette mère Ψ de moyenne nulle ($\int_{-\infty}^{+\infty} \Psi(t)dt = 0$). Ainsi, la décomposition en ondelettes fait intervenir deux paramètres qui sont le facteur d'échelle *s* et le facteur de translation *u* :

$$\Psi_{u,s} = \frac{1}{\sqrt{s}}\left(\frac{t-u}{s}\right) \tag{1.5}$$

L'ondelette $\Psi_{u,s}$ est déplacée pour être centrée sur *u* : c'est donc le point autour duquel l'analyse se fait. Le paramètre d'échelle *s* permet d'obtenir, à partir d'une ondelette mère, des ondelettes comprimées (support réduit) ainsi que des ondelettes dilatées (support étendu). Les ondelettes comprimées sont utilisées pour déterminer les composantes de haute fréquence tandis que les ondelettes dilatées permettent de déterminer les composantes de basse fréquence.

La transformé continue d'ondelette d'un signal *x* à l'échelle *s* et à la position *u* est donnée par :

$$W_x(u,s) = \int_{-\infty}^{+\infty} x(t)\frac{1}{\sqrt{s}}\Psi^*\left(\frac{t-u}{s}\right)dt \tag{1.6}$$

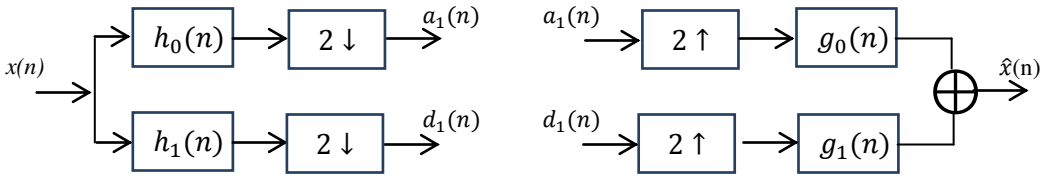
L'inverse de la transformé en ondelettes est donnée par l'équation (1.7) suivante :

$$x(t) = \frac{1}{\sqrt{C_\Psi}}\int\int W_x(u,s)\frac{1}{\sqrt{s}}\Psi^*\left(\frac{t-u}{s}\right)du\frac{ds}{s^2} \tag{1.7}$$

La transformée discrète en ondelettes est dérivée de la version continue. Elle utilise un facteur d'échelle et une translation discrétisés. Nous parlons de la transformée en ondelettes discrète dyadique lorsque le facteur d'échelle est égal à 2^i.
De nombreux types de la transformée en ondelettes ont été proposés dans la littérature [Daubechies, 1992] [Chappelier, 2005]. On peut citer les plus utilisées, comme les ondelettes de : Morlet, Sombrero, Haar, Meyer, Daubechies, Redgelette, Countourlet et Curvelettes, etc. Les deux premières sont des ondelettes continues tandis que les dernières sont des ondelettes discrètes.

Transformée en ondelettes discrète

L'analyse multi-résolution permet l'analyse d'un signal en différentes bandes de fréquences, pour une vue à différentes échelles. Le principe est d'analyser le signal à hautes fréquences, pour prélever les détails, puis d'analyser le signal à une résolution deux fois moins fine, et réitérer l'opération en grossissant l'échelle d'un facteur deux, jusqu'à obtenir une description complète du signal. L'un des éléments fondamentaux de l'analyse multi-résolution est l'introduction d'une matrice de dilatation *D* qui définit le processus de lissage lors d'un changement de résolution.
Nous présentons dans la figure 1.2 et la figure 1.3 l'algorithme de décomposition/synthèse rapide (DWT) d'un signal *x(n)* tel qu'il a été proposé par S. Mallat [Mallat, 1997]. Le calcul de l'approximation passe-bas et des coefficients d'ondelettes à l'échelle *l* se résume à la convolution (filtrage) des coefficients de l'approximation passe-bas à l'échelle *l – 1* suivie d'une opération de décimation suivant D : $h_0(n)$ représente le filtre passe-bas, $h_1(n)$ est le filtre passe-haut, $2\downarrow$ représente l'opération de décimation d'un facteur 2 et $2\uparrow$ représente l'interpolation qui consiste à intercaler un zéro entre deux échantillons.
En suivant le même raisonnement que pour la décomposition, on obtient que la reconstruction de l'approximation passe-bas à l'échelle *l* se résume à la convolution des coefficients de l'approximation passe-bas et des coefficients d'ondelettes à l'échelle *l* +1 précédée d'une opération d'interpolation suivant D. Comme pour l'analyse, ce processus peut se réitérer permettant ainsi de reconstruire la séquence initiale à partir de tous les coefficients d'ondelettes et de la dernière approximation passe-bas.



(a) Banc de filtres d'analyse (b) Banc de filtres de synthèse.

Figure 1.2 : Décomposition et reconstruction par la transformée en ondelettes (un seul niveau).

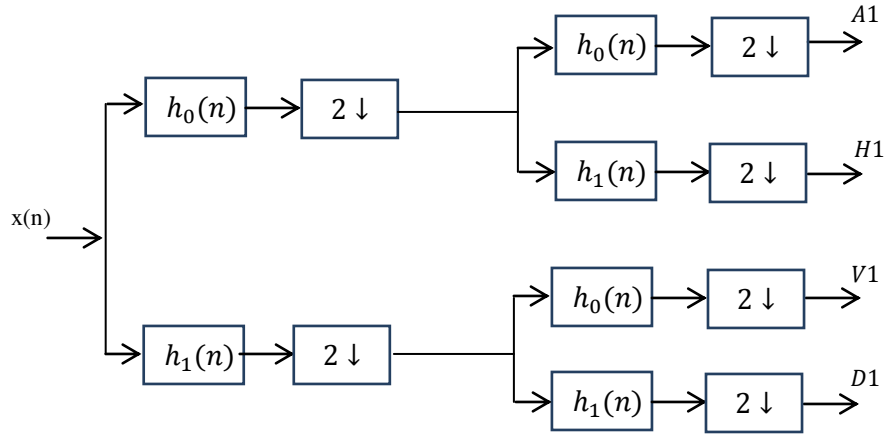


Figure 1.3 : Décomposition en ondelettes en un niveau.

Nous présentons sur la figure 1.4, un exemple d'analyse de l'image Lina pour 3 niveaux de résolution à partir d'un banc de filtres. La reconstruction d'une image à partir de ses coefficients en ondelettes prend une signification intuitive évidente : l'image, à sa résolution la plus grande, est égale à la somme d'une approximation, et des détails apparaissant à des échelles différentes, c'est à dire à des résolutions différentes.

Figure 1.4 : Décomposition par la transformée en ondelettes de l'image Lena.

Exemples d'ondelettes

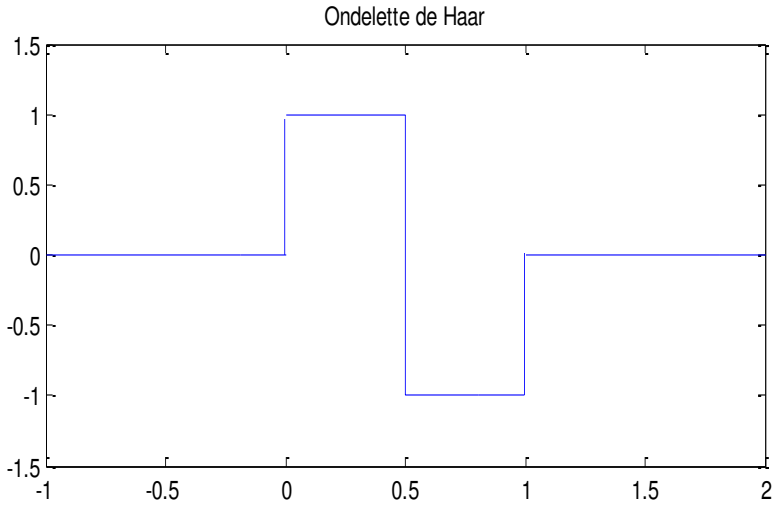
Il existe de nombreuses formes d'ondelettes, le choix de l'ondelette optimale dépend de l'application envisagée. Il convient de bien cerner le problème à étudier et d'identifier le type de transformée à utiliser. Nous avons choisi de présenter deux types d'ondelettes qui nous semblent être les plus utilisées dans le traitement du signal : les ondelettes de Haar et les ondelettes de Daubechies.

L'ondelette de Haar

Définition

Elle est définie par :

$$\psi(x) = \begin{cases} 1 \;\; si \; 0 \leq x \leq \frac{1}{2} \\ -1 \;\; si \; \frac{1}{2} \leq x \leq 1 \\ 0 \qquad\quad sinon \end{cases} \tag{1.8}$$



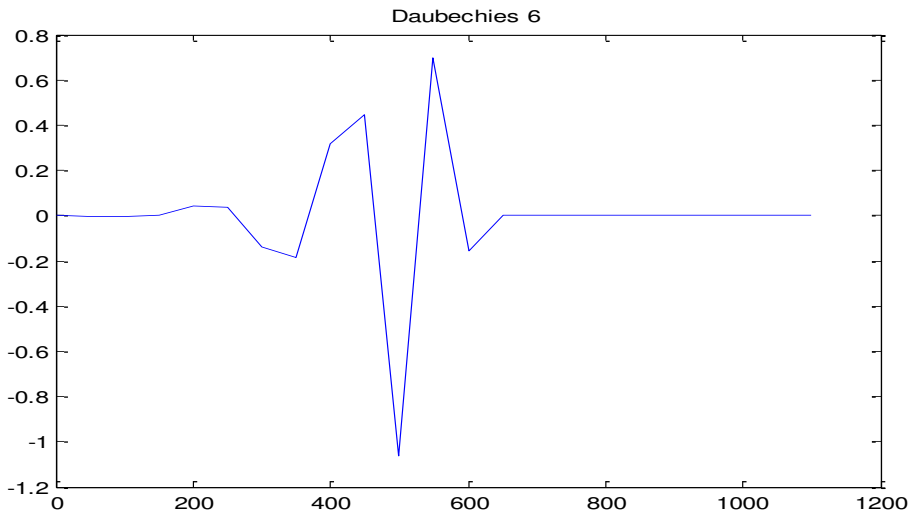
Propriétés et Intérêt

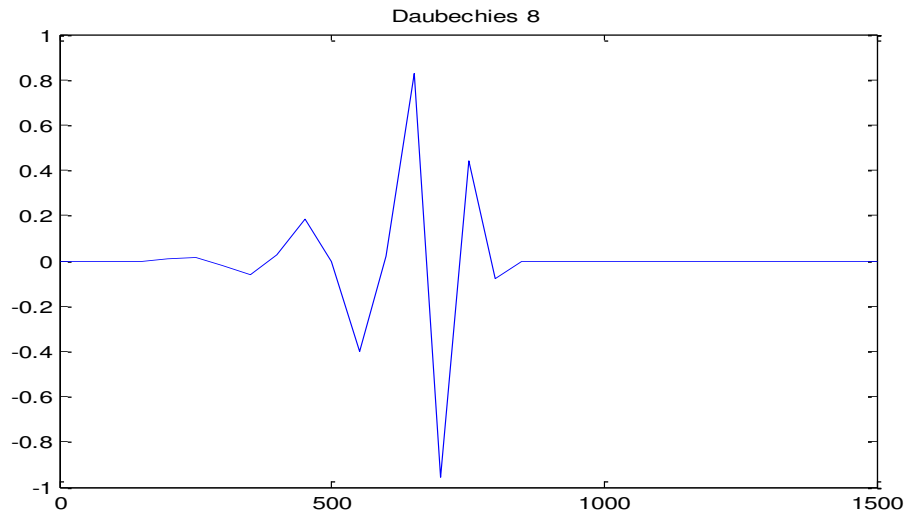
C'est une ondelette orthonormée à support compact et symétrique. Elle permet d'obtenir une reconstruction exacte du signal. Elle est utilisée à la fois pour les transformées continues et discrètes. Cette ondelette est très simple et facile à implémenter.

L'ondelette de Daubechies

Définition

L'ondelette de Daubechies est la famille la plus connue des ondelettes orthonormées. Ses ondelettes sont généralement dénommées par le nombre de coefficients a_k non nuls, on parlera donc d'ondelettes Daubechies 4, Daubechies 6, etc. (voir ci-dessous).





Propriétés

Quand l'ordre augmente, les supports grandissent ainsi que la régularité des ondelettes.

Intérêt

La mathématicienne Ingrid Daubechies a cherché dans ses travaux à concilier deux contraintes respectives : l'orthogonalité de la base d'ondelettes et la compacité du support de l'ondelette-mère. Ceci implique que toute ondelette de la base est à support compact et donc le calcul de la transformée en ondelettes est exacte. De plus, elle a imposé à ses ondelettes une troisième condition : avoir *n* moments nuls.

Inconvénients

Les ondelettes à support compact ne sont pas symétriques ce qui peut être un problème dans certains cas comme la détection de frontières.

1.7 La stéganalyse

Contrairement à la stéganographie, la stéganalyse est l'art et la science de détecter si un medium donné cache un message secret, et si possible, de récupérer ce message caché. Ceci est analogue à la cryptanalyse appliquée à la cryptographie.

La Stéganalyse est en effet une tâche très difficile, en raison de la grande diversité des mediums, la grande variation des données, les différents algorithmes d'insertion et généralement la faible distorsion due à l'intégration du message. Malgré cela, la stéganalyse est encore possible, car l'insertion du message perturbe les statistiques du medium original [Farid, 2002]. En d'autres termes, la présence d'un message intégré rend un medium original et sa version stégo correspondante différents à certains aspects, bien que cette présence est souvent imperceptible à l'œil humain.

Selon la littérature, les méthodes de stéganalyse peuvent généralement être classées en deux catégories : spécifiques et universelles. Alors que la première vise à casser un algorithme stéganographique spécifique, la deuxième tente à briser tous les algorithmes de stéganographie. En général, les approches spécifiques ont une précision de détection plus élevée par rapport à celles universelles, car elles ont une connaissance préalable de la méthode d'insertion. Néanmoins, la stéganalyse universelle est plus attrayante dans l'utilisation pratique, car elle fonctionne indépendamment de la technique d'insertion et donc elle s'applique à des algorithmes d'insertion inconnus.

La stéganalyse universelle peut être considérée comme un processus de reconnaissance de formes. Elle doit au final déterminer à quelle classe : médium original, ou médium stégo, appartient un médium sous test. Un principe général d'un algorithme de stéganalyse universel est d'identifier et d'extraire des caractéristiques qui sont particulièrement sensibles à l'intégration de message. Ces caractéristiques doivent être en mesure de capter les variations résultantes de cette intégration. Cela signifie que les caractéristiques extraites d'un médium original devraient être différentes de celles médium stégo correspondant [shi et al., 2005]. En général, plus la différence est grande, meilleur est le choix des caractéristiques extraites. Après l'extraction de caractéristiques, un classificateur est généralement utilisé pour distinguer le deux médiums propre et stégo à partir de leurs vecteurs caractéristiques. Globalement, la performance d'un système de stéganalyse dépend notamment de choix du vecteur de caractéristiques à N dimensions, extrait à partir du médium, et de l'efficacité du classificateur utilisé. Naïvement, on pourrait s'attendre à ce que N doit être aussi grand que possible pour que le vecteur de caractéristiques soit plus efficace. Or, des études récentes [Wang et al., 2007], [Qin et al., 2009] montrent que d'une part, N ne doit pas être nécessairement très grande et d'autre part, un N très grand, généralement entraîne des coûts informatiques élevés et un impact négatif sur la précision de la détection.

Les transformées en ondelettes sont très utilisées pour extraire les vecteurs des caractéristiques des mediums originaux et stégos [Wang et al., 2007], [Farid, 2002].

Pour la classification (ou stéganalyse), les vecteurs de caractéristiques extraites à partir d'un ensemble de médiums avec ou sans une information cachée dans chacun d'eux, sont introduits dans un algorithme d'apprentissage pour apprendre le classifieur avant de l'utiliser pour séparer les deux types de médiums : original et stégo. A ce titre, l'analyse discriminante linéaire de Fisher (FLD) [Farid, 2002], et l'analyse discriminante non linéaire, tels que les

Machines à vecteurs de support (SVM) [Pevny et Fridrich, 2008], sont largement utilisées avec succès dans des techniques de stéganalyse antérieures.

Nous rappelons ci-dessous l'analyse discriminante linéaire de Fisher utilisée dans notre étude, mais aussi les Machines à vecteurs de support (SVM) (que nous n'avons pas pu utiliser, faute de temps)

1.7.1 Classification FLD (Fisher Linear Discriminant)

La discrimination linéaire de Fisher permet l'identification d'une direction de l'espace sur laquelle la projection des caractéristiques est particulièrement bien séparée (Figure 1.5).

L'objectif de la FLD est de trouver la direction optimale *w* telle que la projection de deux classes originales et stégos selon cette direction permet de séparer efficacement les deux classes.

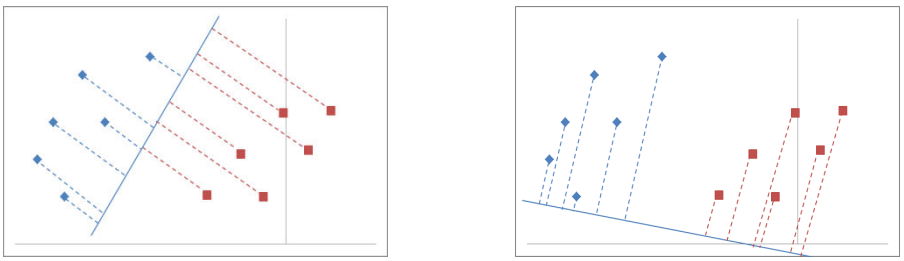


Figure 1.5 : Projection des points en deux directions

Soit $\{x_1, x_2, \ldots, x_n\}$ un ensemble d'échantillons parmi lesquels n_1 appartiennent à la classe c_1 *et* n_2 appartiennent à la classe c_2. Leurs projections sur une droite d'orientation w, sont notées y_i $(i=1,\ldots,n)$.
On souhaite que les points y_i correspondants à c_1 et ceux correspondants à c_2 soient les mieux séparés possible. Afin d'atteindre cet objectif, il faut :

- maximiser **la variance inter-classes** (entre les centres de gravité de deux classes)
- minimiser **les variances intra-classes** (entre les points et le centre de gravité correspondant de chaque classe).

Variance inter-classes

Les centres de gravité (ou moyen) de chaque classe ($i = 1, 2$) sont donnés par :

$$m_i = \frac{1}{n_i}\sum_{x \in c_i} x \tag{1.9}$$

Les centres de gravité (ou moyen) de leurs projections sont donnés par :

$$m'_i = \frac{1}{n_i}\sum_{y \in c_i} y = \frac{1}{n_i}\sum_{x \in c_i} W^t x = W^t m_i \tag{1.10}$$

Avec W^t est un vecteur d'orientation.

La variance entre les moyens de deux classes (originale et stégo) est :

$$(m'_1 - m'_2)^2 = (W^t m_1 - W^t m_2)^2 = W^t(m_1 - m_2)(m_1 - m_2)^t W = W^t S_B W \quad (1.11)$$

Où :
$S_B = (m_1 - m_2)(m_1 - m_2)^t$ est la matrice de dispersion inter-classes.

La distance intra-classes

La dispersion de chaque classe (pour i=1,2) est définie par :

$$S_{ip} = \sum_{y \in w_i}(y - m'_i)^2 = \sum_{x \in w_i}(W^t x - W^t m_i)^2 = \sum_{x \in w_i} W^t(x - m_i)(x - m_i)^t W = W^t S_i W \quad (1.12)$$

Cette dispersion représente la distribution des éléments de chaque classe autour de sa moyenne.

La dispersion intra-classes totale est donc :

$$S_W = S_1 + S_2 = (x - m_1)(x - m_1)^t + (x - m_2)(x - m_2)^t \quad (1.13)$$

Discrimination linéaire de Fisher

Le discriminant de Fisher désigné qui vise à trouver l'hyper-plan qui maximise la variance inter-classes et minimise la variance intra-classes des données, se traduit finalement par le critère de Rayleigh suivant :

$$J(W) = \frac{(m\prime_1 - m\prime_2)^2}{S_{1p} + S_{2p}} = \frac{W^t S_B W}{W^t S_w W} \quad (1.14)$$

et

$$W_{opt} = S_W^{-1}(m_1 - m_2) \quad (1.15)$$

1.7.2 Classification SVM (Machines à Vecteurs de Support) :

Principe

Les machines à vecteurs de support forment une classe d'algorithmes d'apprentissage supervisé. Soit une fonction notée f qui à toute entrée Z fait correspondre une sortie $y = f(Z)$, son apprentissage se réalise à partir d'un ensemble de couple (Z_i, y_i).
Nous nous intéressons ici à construire une fonction *f* qui à chaque valeur d'entrée dans un ensemble $\mathbb{R}^d$ va faire correspondre une valeur de sortie $y \in \{-1, 1\}$:

$$f : \mathbb{R}^d \to \{-1, 1\}, \quad f(Z) = y \quad (1.16)$$

Dans le cas linéaire, une fonction discriminante *h* est obtenue par combinaison linéaire d'un vecteur d'entrée $Z = \{Z_1, \ldots\ldots, Z_d\}$ et s'écrit :

$$h(Z) = w.Z + b \tag{1.17}$$

La classe est donnée par le signe de $h(Z)$: $sign\big(h(Z)\big)$. $Si\ h(Z) \geq 0$ alors Z appartient à la classe 1 sinon Z est de classe -1. La séparatrice est alors un hyperplan affine d'équation : $w.Z + b = 0$. Si le couple (Z_i, y_i) est l'un des p éléments de la base d'apprentissage noté A_p, on veut trouver le classifieur h tel que :

$$y_i(w.Z_i + b) \geq 0,\ \ i \in [1, p] \tag{1.18}$$

Dans le cas simple linéairement séparable, il existe de nombreux hyperplans séparateurs comme nous pouvons le voir sur la figure 1.6. Selon la théorie de Vapnick [Cortes et Vapnik, 1995, Vapnik, 1995] l'hyperplan optimal (optimum de la distance inter-classes) est celui qui maximise la marge. Cette dernière étant définie comme la distance entre un hyperplan et les points les plus proches. Ces points particuliers sont appelés vecteurs de support. La distance entre un point Z quelconque et l'hyperplan est donnée par l'équation (1.19).

$$d(Z) = \frac{|w.Z+b|}{\|w\|} \tag{1.19}$$

Nous allons voir au paragraphe suivant que maximiser la marge va revenir à minimiser$\|w\|$.

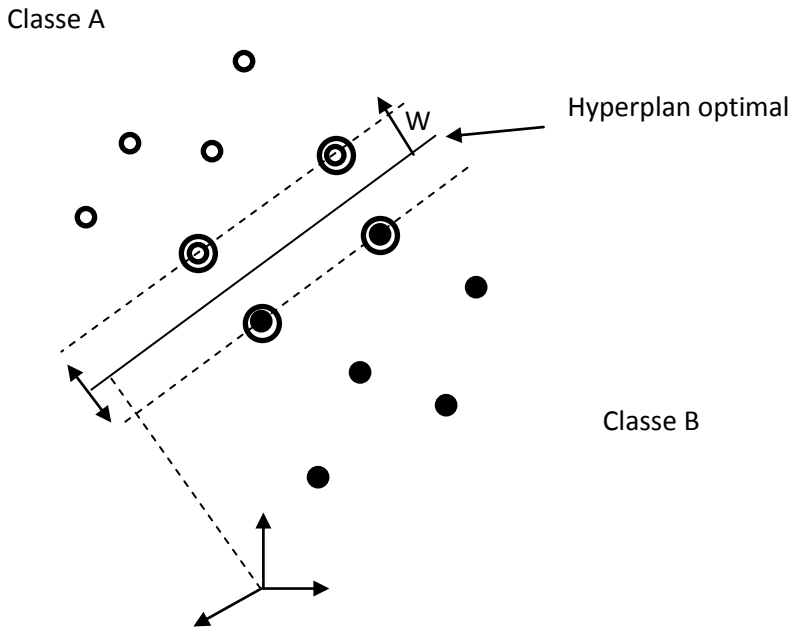


Figure 1.6 : Hyperplan optimal de séparation avec marge souple dans un cas non linéairement séparable.

Forme primale

Les paramètres w et b étant définis à un coefficient multiplicatif près, on choisit de les normaliser pour que les échantillons les plus proches (Z_s) vérifient l'égalité suivante :

$$y_s(w.Z_s + b) = 1 \tag{1.20}$$

Donc quel que soit l'échantillon Z_i on obtient :

$$y_i(w.Z_i + b) \geq 1 \tag{1.21}$$

La distance entre l'hyperplan et un point support est donc définie par $\frac{1}{\|w\|}$. La marge géométrique entre deux classes est égale à $\frac{2}{\|w\|}$. La forme primale (qui dépend seulement de w et b) des SVM est donc un problème de minimisation sous contrainte qui s'écrit :

$$\begin{cases} \min\left(\frac{1}{2}\|w\|^2\right) \\ \forall(Z_i, y_i) \in A_p, y_i(w.Z_i + b) \geq 1 \end{cases} \tag{1.22}$$

Forme duale

La formulation primale peut être transformée en formulation duale en utilisant les multiplicateurs de Lagrange. L'équation (1.22) s'écrit alors sous la forme suivante :

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{p} \alpha_i(y_i(w.Z_i + b) - 1) \tag{1.23}$$

La formulation de Lagrange permet de trouver les extremums en annulant les dérivées partielles de la fonction $L(w, b, \alpha)$. Le lagrangien *L* doit être minimisé par rapport à *w* et *b* et maximisé par rapport à α. On résout ce problème en calculant les dérivées partielles :

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{p} \alpha_i y_i Z_i = 0 \tag{1.24}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{p} \alpha_i y_i = 0 \tag{1.25}$$

En réinjectant les deux premières dérivées partielles (1.24) et (1.25) dans l'équation (1.23) nous obtenons :

$$L(\alpha) = \frac{1}{2}\sum_{i=1}^{p} \alpha_i y_i \sum_{j=1}^{p} \alpha_j y_j Z_i Z_j - \sum_{i=1}^{p} \alpha_i y_i \sum_{j=1}^{p} \alpha_j y_j Z_i Z_j - \sum_{i=1}^{p} \alpha_i y_i b + \sum_{i=1}^{p} \alpha_i \tag{1.26}$$

puis on en extrait la formulation duale (dépendante des α_i) suivante :

$$L(\alpha) = \sum_{i=1}^{p} \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j Z_i.Z_j \tag{1.27}$$

On cherche donc à maximiser $L(\alpha)$ sous les contraintes $\alpha_i \geq 0$ et $\sum_i \alpha_i y_i = 0$. A l'optimal, α^*, les conditions de Karush Kuhn Tucker (conditions KTT) sont satisfaites et permettent d'écrire l'égalité suivante :

$$\alpha_i[y_i(w.Z_i - 1) - 1] = 0, \forall\, i \in [1, p]. \tag{1.28}$$

Cela nous donne $\alpha_i = 0$ ou $(w.Z_i + b) - 1 = 0$. Ces deux possibilités impliquent que seuls les α_i associés à des exemples situés sur la marge peuvent être non nuls. Autrement dit, ces exemples sur la marge constituent les vecteurs supports, qui seuls contribuent à définir l'hyperplan optimal. Cette maximisation est un problème de programmation quadratique de dimension égale au nombre d'exemples. L'équation (1.24) nous donne la valeur optimale pour *w* noté w^* : $w^* = \sum_{i=1}^{p} \alpha_i^* y_i Z_i$ avec α_i^* les coefficients de Lagrange optimaux. En utilisant l'équation de l'hyperplan (1.17) nous obtenons l'hyperplan de marge maximale :

$$h(Z) = \sum_{i=1}^{p} \alpha_i^* y_i Z.Z_i + b \tag{1.29}$$

Non linéarité (cas non séparable/ marge molle)

On part du problème primal linéaire et on introduit des variables ≪ ressort ≫ pour assouplir les contraintes.

$$\begin{cases} \min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \\ \forall i,\ y_i(w.Z_i + b) \geq 1 - \xi_i \end{cases} \tag{1.30}$$

On pénalise par le dépassement de la contrainte.
On en déduit le problème dual qui a la même forme que dans le cas séparable :

$$\begin{cases} \max \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j Z_i.Z_j \\ \forall i,\ 0 \leq \alpha_i \leq C \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases} \tag{1.31}$$

La seule différence est la borne supérieure C sur les α.

Astuce du noyau

Le cas linéairement séparable est peu intéressant, car les problèmes de classification sont souvent non linéaires. Pour résoudre ce point la méthode classique est de projeter les données dans un espace de dimension supérieur appelé espace de redescription. L'idée étant qu'en augmentant la dimensionnalité du problème on se retrouve dans le cas linéaire vu précédemment. Nous allons donc appliquer une transformation non linéaire $\Phi(.)$ aux vecteurs d'entrée Z_i tel que $Z_i \in \mathbb{R}$ et $\Phi(Z_i) \in \mathbb{R}$, (e > d). Ce changement va conduire à passer d'un produit scalaire dans l'espace d'origine $Z_i.Z_j$ à un produit scalaire $\Phi(Z_i).\Phi(Z_j)$ dans l'espace de redescription. L'astuce est d'utiliser une fonction noyau notée K qui évite le calcul explicite du produit scalaire dans l'espace de redescription. Les fonctions noyaux doivent satisfaire le théorème de Mercer. Nous avons alors l'égalité suivante :

$$K(Z_i, Z_j) = \Phi(Z_i).\Phi(Z_j) \tag{1.32}$$

Il existe de nombreuses fonctions noyau prédéfinies, les deux les plus usuelles sont le noyau gaussien (équation 1.33) et le noyau polynomial (équation 1.34) :

$$K_\gamma(Z_i, Z_j) = e^{-\gamma\|Z_i - Z_j\|^2} \quad (1.33)$$

$$k_{\gamma,d,r}(Z_i, Z_j) = (\gamma Z_i . Z_j + r)^d \quad (1.34)$$

Les noyaux gaussien sont des noyaux dits de type radial (fonction à base radial abrégé RBF [Bottou et Chih-Jen, 2007], indiquant qu'ils dépendent de la distance entre deux exemples. L'hyperplan séparateur se réécrit alors avec la fonction noyau sous la forme suivante :

$$h(Z) = \sum_{i=1}^{p} \alpha_i^* y_i K_\gamma(Z, Z_i) + b \quad (1.35)$$

1.8 Théorie chaotique

Rappel Historique

La notion de fonctions chaotiques apparaît au début du XXème siècle dans les travaux d'Henri Poincaré sur la physique des corps. Cependant, il faut attendre les années 60, avec l'apparition de l'ordinateur, pour que cette notion soit approfondie. En effet, il fallait réaliser un nombre immense d'opérations de calcul, ce qui n'était pas possible avant les années 60.

En 1963, le météorologue Edward Lorenz prouve le caractère chaotique des conditions météorologiques, un infime changement de l'état initial pouvant entraîner une évolution totalement différente (ce qui lui inspira le fameux postulat du battement d'aile de papillon). Avec cette découverte les travaux d'Henri Poincaré connurent un regain d'intérêt et en 1975 le mathématicien James Yorke emploie pour la première fois le terme de « chaos »

Plusieurs domaines d'applications variés utilisent les principes de la théorie du chaos pour étendre et mieux comprendre les phénomènes liés à ses applications. Nous citons : la psychologie, la sociologie, la biologie, la physique, l'économie, la sécurité de l'information, etc.

1.8.1 Caractéristiques de Chaos

Sous certaines conditions, les systèmes dynamiques non linéaires génèrent des signaux chaotiques. Ces signaux sont de forme d'ondes apériodiques et possèdent des propriétés quasi identiques au bruit que ce soit dans le domaine temporel ou dans le domaine fréquentiel, ce qui les rend difficiles à prévoir et à intercepter.
Le chaos est extrêmement sensible à n'importe quel changement (même très faible) dans les conditions initiales ou dans les paramètres de contrôle (l'ensemble forme la clé secrète) d'un système donné. En effet, si deux systèmes chaotiques identiques ont des états initiaux ou des paramètres qui diffèrent de très peu, les orbites chaotiques de ces systèmes seront très différentes. Cette caractéristique de l'hyper sensibilité à la clé secrète, rend l'utilisation des signaux chaotiques très intéressante pour la sécurité de l'information. D'autant plus, que le comportement de tels systèmes est imprédictible, bien qu'ils soient gouvernés par des lois simples, connues et déterministes.
De nombreux travaux de recherche scientifique, montrant l'apport des signaux chaotiques dans la sécurité des systèmes de communications [Barenghi, 2010].

1.8.2 Utilisation du chaos

Le comportement dynamique très riche des signaux chaotiques les rend attractifs pour différentes applications. Ils ont été notamment utilisés pour la modélisation de systèmes de l'environnement naturel [Letellier, 2006], ou pour l'estimation de l'état d'un système. Ils ont aussi été employés pour la modélisation de certains phénomènes quantiques : on parle alors de chaos quantique [Naschie, 1996]. Une majorité de ces travaux se sont portés sur des systèmes à temps continu, tels que des systèmes d'analyse de battements cardiaques [Brandt et Chen, 2000], des systèmes électroniques [Ogorzalek, 2000] ou des systèmes mécaniques [Kapitaniak et al., 2000]. D'autres études s'intéressent à des systèmes spatio-temporels [Fangi et Ali, 2000]. Récemment, le chaos fut largement utilisé dans des applications touchant à la sécurité de l'information en stéganographie [Mooney et al., 2008], en génération des nombres pseudo-chaotique [El Assad et al., 2008], [Lozi, 2012], [El Assad et Noura, 2011], et en cryptographie basée chaos [Stavroulakis, 2006], [Zang et al., 2013], [Bakhache et al., 2013], [El Assad et al., 2014].

1.8.3 Cartes chaotiques

Parmi les nombreuses cartes chaotiques de la littérature, nous présentons très brièvement ci-dessous seulement les équations de trois cartes chaotiques très utilisées en pratique qui sont : la carte Logistique, la carte PWLCM (Piece Wise Linear Chaotic Map), et la carte Skewtent. Ces cartes possèdent plusieurs bonnes propriétés : réalisation simple, et généralement assez bonne propriété cryptographique.

1.8.3.1 La carte d'ARNOLD

La carte chaotique appelée la carte d'Arnold en reconnaissance de mathématicien russe Vladimir I. Arnold, qui l'a découverte en utilisant une image d'un chat. C'est une démonstration et une illustration simple et élégante de certains des principes de chaos, une évolution apparemment aléatoire d'un système.

Si nous considérons $X = \binom{x}{y}$, une matrice de taille n x n, la transformation d'Arnold T est :

$$\binom{x'}{y'} = T\binom{x}{y} = \binom{x+y}{x+2y} \; mod\; n$$

x'et y' est nouvelle position du pixel, x et y est la position originale de ce pixel.

1.8.3.2 Logistique map

Une **suite logistique** est une suite simple, dont la récurrence n'est pas linéaire et donnée par la relation suivante.

$$x_{n+1} = r x_n (1 - x_n) \tag{1.36}$$

x est la variable dynamique prenant des valeurs entre 0 et 1 non inclus et *r* est le paramètre du système. Selon la valeur de *r*, la suite peut être un point fixe, une suite périodique de période 2, 4, 8,..., et 64 pour *r* = 3,569692, ou une suite *chaotique* pour *r* compris entre 3,56996 et 4.

1.8.3.3 Carte PWLCM (Piece Wise Linear Chaotic Map)

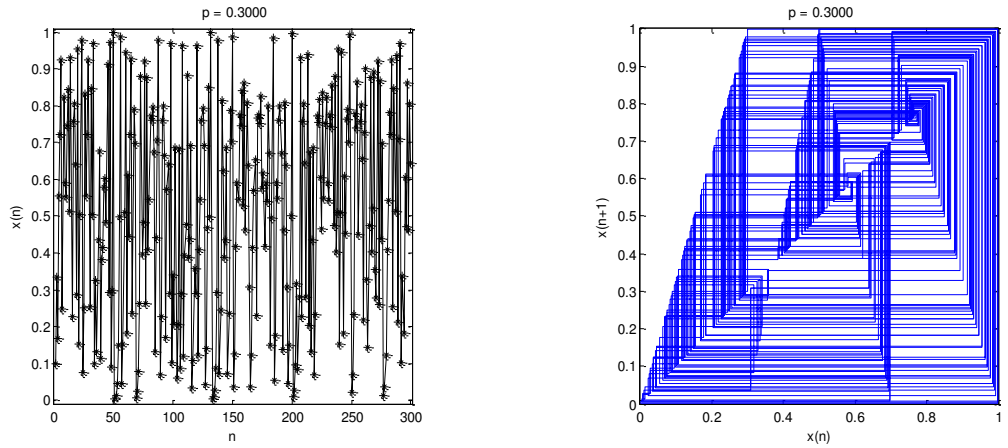
La carte chaotique Piece Wise Linear Chaotic Map (PWLCM) est composée de plusieurs segments linéaires par morceaux dont l'équation est donnée par :

$$x(n) = f[x(n-1), p] \tag{1.37}$$

$$x(n) = \begin{cases} x(n-1) \times \frac{1}{p} & si\ 0 \leq x(n-1) < p \\ [x(n-1) - p] \times \frac{1}{0.5-p} & si\ p \leq x(n-1) < 0.5 \\ F[1 - x(n-1)] & si\ 0.5 \leq x(n-1) < 1 \end{cases} \tag{1.38}$$

$p \in [0, 0.5]$ est le paramètre de contrôle et $x\ (0) \in [0, 1[$ est la valeur initiale.

La figure 1.7 (a) ci-dessous représente la forme temporelle de la fonction PWLCM pour 300 itérations, utilisant une valeur initiale $x(0)$ égale à 0.6, et une valeur de paramètre p égale à 0.3. La figure 1.7 (b), représente l'attracteur, courbe $[x(n), x(n+1)]$ de la carte PWLCM (tracé pour 1000 itérations)



(a) (b)

Figure 1.7 : Carte PWLCM : (a) Séquence *x*(n); (b) Attracteur

La carte PWLCM est caractérisée par :

1. une densité invariante et uniforme;
2. une réalisation simple du point de vue matériel et logiciel.

1.8.3.4 Carte Skew tent

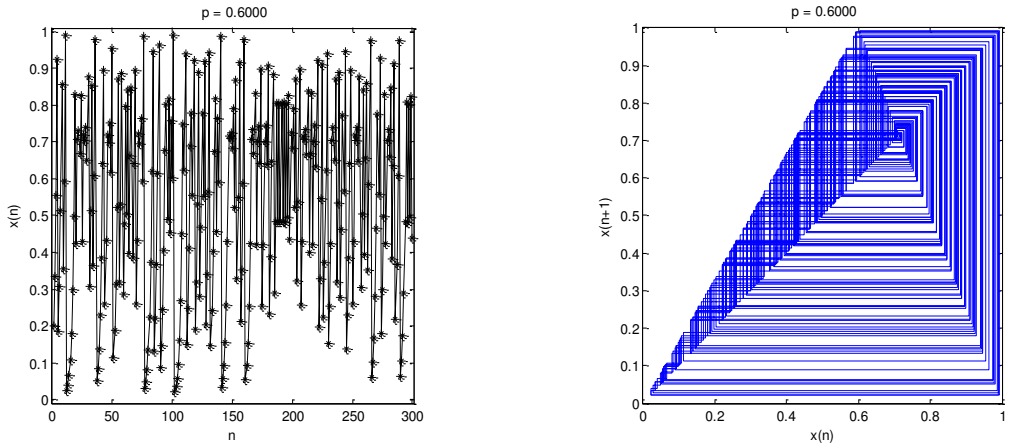
La carte Skew tent est une carte linéaire par morceaux, décrite par l'équation suivane:

$$x(n) = f(x(n-1), p) = \begin{cases} \frac{x(n-1)}{p} & si\ 0 \le x(n-1) \le p \\ \frac{1-x(n-1)}{1-p} & si\ p \le x(n-1) \le 1 \end{cases} \quad (1.39)$$

où $x(n) \in [0,1[$, et p le paramètre de contrôle qui varie dans l'intervalle suivant : $0 < p < 1$

L'histogramme de cette carte est pratiquement uniforme comparé à celle de la carte Logistique [Billings et Bollt, 2001].

La figure 1.8 (a) représente la séquence temporelle *x*(n) générée par la carte Skewtent avec *p* égale à 0.6, et la figure 1.8 (b), représente son attracteur.



(a) (b)

Figure 1.8 : Carte Skewtent : (a) Séquence *x*(n), (b) Attracteur

1.8.4 Intérêt et description de la technique de perturbation de l'orbite chaotique

En précision finie, pour un système de N bits de quantification, le nombre maximum de niveaux chaotiques différents est 2^N. Cette limitation de l'espace des valeurs (supposée infini pour le chaos analogique) entraîne des cycles périodiques des différentes orbites chaotiques, ayant chacune une longueur maximale forcement largement inférieure à 2^N (propriétés quasi chaotique), donc la dynamique des signaux chaotiques est dégradée [El Assad et Noura, 2011].

Par ailleurs, à chaque condition initiale, il existe une orbite chaotique formée généralement de deux parties : une branche transitoire de longueur l et un cycle de période c. Notons aussi que le cas l=0 ou c=1 est possible. Lorsque l=0, l'orbite est un simple cycle de longueur c, et lorsque c=1, l'orbite chaotique converge vers un point fixe (voir figures 1.9 et 1.10).

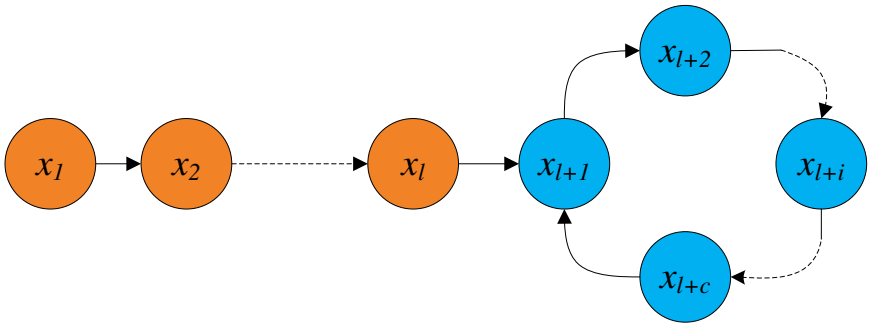


Figure 1.9 : Orbite chaotique de longueur *l+c*

La figure 1.10, montre un exemple explicatif, dans le cas N=4, de deux orbites chaotiques obtenues pour deux conditions initiales différentes.

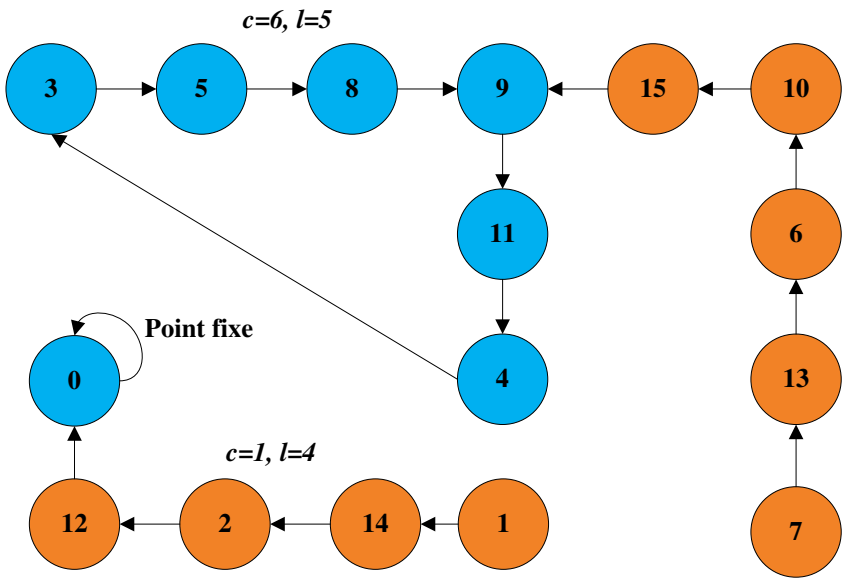


Figure 1.10 : Deux orbites chaotiques différentes pour deux conditions initiales différentes, avec *N=4*

Afin de contourner l'effet de la précision finie sur la dynamique chaotique, deux techniques sont utilisées : la technique de cascade et la technique de perturbation de l'orbite chaotique. La première technique permet effectivement une expansion de la longueur des cycles mais sans aucun contrôle. La seconde technique permet non seulement d'avoir un cycle de longueur très grande, mais aussi d'imposer une longueur minimale de cycle, dépendant directement du signal perturbateur.

La méthode de perturbation trouve son fondement sur le fait qu'aucun cycle stable n'existe pas, c.-à-d. si le système chaotique décrit, à un moment donné, un cycle donné, il peut, par application d'une perturbation, quitter ce cycle immédiatement pour aller vers un autre cycle. Le choix de la séquence perturbatrice est effectué selon les règles suivantes : elle devrait avoir une longue longueur de cycle contrôlable et une distribution uniforme; elle ne devrait

pas dégrader les bonnes propriétés statistiques de la dynamique chaotique, donc l'amplitude du signal perturbateur doit être nettement plus petite que celle du signal chaotique, de sorte que le rapport R entre les deux amplitudes maximales, soit supérieur ou égal à 40 dB :

$$R = 20 \times \log\left[\frac{Amplitude\ maximale\ du\ signal\ chaotique}{Amplitude\ maximale\ du\ signal\ perturbateur}\right] \geq 40\ dB \qquad (1.40)$$

Un bon candidat pour la génération de séquences perturbatrices est le registre à décalage à réaction à longueur maximale. En effet, ce dernier est caractérisé par : une bonne fonction d'auto-corrélation, par une distribution presque uniforme, par un cycle de longueur maximale égale à $2^k - 1$ (k est le degré du polynôme primitif utilisé) et une implémentation logicielle ou matérielle facile.

Partons de l'équation du générateur de base :

$$X(n) = F[X(n-1)] \in 2^N - 1 \qquad n = 1,2 \ldots\ldots \qquad (1.41)$$

où chaque valeur $X(n)$ est représentée par *N* bits :

$$X(n) = x_{N-1}(n)x_{N-2}(n) \ldots x_i(n) \ldots . x_0(n)\ \ x_i(n) \in A_b = [0,1] \qquad (1.42)$$

$$i = 0,1,2, \ldots, N-1$$

Soit Δ le cycle minimal du générateur de séquences chaotiques sans perturbation. La perturbation est appliquée si $n = m \times \Delta \quad m = 0, 1, 2, \ldots$ (c.a.d. pour *n=0* et toutes les Δ itérations, donc Δ est l'horloge du registre RDR). La séquence perturbée s'écrit selon l'équation suivante :

$$x_i(n) = \begin{cases} F[x_i(n-1)] & k \leq i \leq N-1 \\ F[x_i(n-1)] \oplus Q_i(n) & 0 \leq i \leq k-1 \end{cases} \qquad (1.43)$$

où $F[x_i(n-1)]$ représente le ième bit de $F[X(n-1)]$ et $Q_i(n)$ représente la séquence de perturbation (sortie du RDR) telle que :

$$Q_{k-1}^{+}(n) = Q_k(n) = g_0 Q_0(n) \oplus g_1 Q_1(n) \oplus \ldots \oplus g_{k-1} Q_{k-1}(n) \qquad (1.44)$$

$$n = 0,1,2 \ldots.$$

avec :

$[g_0, g_1, \ldots, g_{k-1}]$ sont les coefficients du polynôme primitif du registre et $[Q_0, Q_1, \ldots, Q_{k-1}]$ représente la valeur initiale non nulle du registre. Notons que la séquence perturbatrice est appliquée sur les *k* bits de poids faibles de $F[X(n-1)]$.

Si $n \neq m \times \Delta \quad m = 0, 1, 2, \ldots$, la sortie du générateur de séquences chaotiques n'est pas perturbée, donc :

$$X(n) = F[X(n-1)]$$

La période du système chaotique perturbé est donnée par :

$$L = \sigma \times \Delta \times (2^K - 1) \qquad (1.45)$$

où σ est un entier positif. La période minimale est alors :

$$L_{min} = \Delta \times (2^K - 1) \quad (1.46)$$

1.9 Conclusion

Dans ce chapitre, nous avons défini les différents termes du domaine de la sécurité de l'information, telques le tatouage, la cryptographie, et la stéganographie,….etc.
La stéganographie etant notre travail dans ce qui suit, nous avons présenté ses différents types, ses supports et ses domaines spatial et fréquentiel.
Nous avons aussi présenté, la stéganalyse, la théorie chaotique et ses cartes qui sont utilisées dans notre étude.
Dans le chapitre suivant, nous allons voir la stéganographie spatiale, ses différents systèmes existants et les améliorations proposées par le chaos, la stéganographie fréquentielle fera l'objet du chapitre 3, et finalement la stéganalyse des systèmes stéganographiques celui du chapitre 4.

1.10 Références

[Bakhache et al., 2014] Bakhache, B., Ghazal, J. et El Assad, S. (2014). Improvement of the Security of ZigBee by a New Chaotic ALGORITHM, Systems Journal, IEEE, Vol. 8, NO. 4, December 2014, pages 1021-1030. ISSN: 1932-8184, DOI: 10.1109/JSYST.2013.2246011.

[Barenghi, 2010] Barenghi, C.F. (2010). Introduction to chaos: theoretical and numerical methods

[Benabdellah et al., 2003] Benabdellah, M., Rerbal, S., Habibes, N., Meziane Tani, A. et Nemmiche, A. (2003). Traitement numérique du signal physiologique :Application au débruitage et à l'analyse de l'ECG par Ondelettes, *CISTEMA*.

[Bergmair, 2007] Bergmair, R. (2007). A comprehensive bibliography of linguistic steganography.In Proceedings of the SPIE Conference on Security, Steganography, and Watermarking of Multimedia Contents, volume 6505.

[Billings et Bollt, 2001] Billings, L. et Bollt, E.M. (2001). Probability density functions of some skew tent maps, Chaos Solitons Fractals, Volume 12, pages 365–376.

[Bottou et Chih-Jen, 2007] BOTTOU, L. et CHIH-JEN, L. (2007). Support Vector Machine Solvers, in Large Scale Kernel Machines. MIT Press.

[Brandt et Chen, 2000] Brandt, M. E. et Chen, G. (2000). Controlling chaos and bifurcations in engineering systems, chapitre Delay feedback control of cardiac activity models, pages 325–345. CRC Press, Taylord and Francis.

[Brun, 2003] Brun, L. (2003). Traitement d'images couleurs.

[Chappelier, 2005] Chappelier, V. (2005). Codage progressif d'images par ondelettes orientées, *thèse de doctorant,* Université de Rennes 1.

[Cheddad et al., 2010] Cheddad, A., Condell. J., Curran K et McKevitt ,P. (2010). Digital image steganography: Survey and analysis of current methods, Signal Processing 90, pages 727–752.

[Cortes et Vapnik, 1995] CORTES, C. et VAPNIK, V. (1995). Support vector networks. Machine Learning, 20:273–297.

[Daubechies, 1992] Daubechies, I. (1992).Ten Lectures on Wavelet, *Capital city press,* États-Unis.

[Douzi et al., 2001] Douzi, H., Mammass, D. et Nouboud, F. (2001). Faber-Schauder Wavelet Transform, Application to Edge Detection and Image Characterization, *Journal of Mathematical Imaging and Vision*, Volume 14(2), pages 91-101.

[El Assad et al., 2008] El Assad, S., Noura, H. et Taralova, I. (2008). Design and analyses of efficient chaotic generators for crypto-systems, Advances in Electrical and Electronics

Engineering- IAENG Special Edition of the World Congress on Engineering and Computer Science 2008, volume I, pages 3-12, ISBN: 978-0-7695-3555-5.

[El Assad et Noura, 2011] El Assad, S., et Noura, H. (2011). Generator of chaotic sequences and corresponding generating system, Extensions internationales Brevets France n° FR20100059361 et FR201052288. Dépôt 28/03/2011. WO2011121218 (A1) 6/10/2011 et EP2553567 (A1) 06/02/2013. Publications : CN103124955(A) 29/05/2013 ; JP2013524271(A) 17/06/2013 ; US2013170641(A1) 4/07/2013.

[El Assad et al., 2008] El Assad, S., Farajallah, M. et Vladeanu, C. (2008). Chaos-based Block Ciphers: An Overview", IEEE, 10th International Conference on Communications, COMM-2014, Bucharest, Romania, May 2014, pages 23-26. Invited talk

[El Assad et al., 2014] S. El Assad, M. Farajallah, C. Vladeanu, "Chaos-based Block Ciphers: An Overview", IEEE, 10th International Conference on Communications, COMM-2014, Bucharest, Romania, May 2014, pp. 23-26. Invited talk

[Elbasi et Eskicioglu, 2006] Elbasi, E. et Eskicioglu, A. M. (2006). A DWT-Based Robust Semi-Blind Image Watermarking Algorithm Using Two Bands.

[Fangi et Ali, 2000] Fangi, J. et Ali, M. (2000). Controlling chaos and bifurcations in engineering systems, chapitre Control and synchronization of spatiotemporal chaos, pages 107–130. CRC Press, Taylord and Francis.

[Farid, 2002] Farid, H. (2002). Detecting hidden messages using higher-order statistical models, InProceedingsIEEE ICIP, volume 2, pages 905-908.

[Goel et al., 2013] Goel, S., Rana, A., Kaur, M., (2013). Comparison of Image Steganography Technique, International Journal of Computers and Distributed Systems , Numero 3, Issue I.

[Kaewkamnerd et Rao, 2000] Kaewkamnerd, N., et Rao, K. R. (2000). Wavelet based image adaptive watermarking scheme, IEE Electronic Letters, volume 36, pages 312-313.

[Kapitaniak et al., 2000] Kapitaniak, T., Brindley, J. et Czolczynski, K. (2000). Controlling chaos and bifurcations in engineering systems, chapitre Chaos in mechanical systems and its control, pages 71–88. CRC Press, Taylord and Francis.

[Kouider, 2013] Kouider, S. (2013). Insertion adaptative en stéganographie application aux images numériques dans le domaine spatial, Université de Montpellier.

[Letellier, 2006] Letellier, C. (2006). Le chaos dans la nature. Vuibert.

[Lozi, 2012] R. Lozi, Emergence of randomness from chaos, International Journal of Bifurcation and Chaos, vol. 22, n°. 2, 2012, 15 pages.

[Mallat, 1997] Mallat, S.G. (1997). A wavelet tour of signal processing, *Academie Press.*

[Mallat, 1998] Mallat, S. (1998). A Wavelet Tour of Signal Processing (Second Edition), Academic Press

[Mazumde et Hemachandran, 2013] Mazumde, J.A. et Hemachandran, K. (2013). Study of Image steganography using LSB,DFT and DWT, International Journal of Computers & Technology.

[Meyer, 1990] Meyer, Y. (1990). OndelettesetOperateurs, Hermann, Paris

[Mooney et al., 2008] Mooney, A., Keating, J. G. et Pitas, I. (2008). A comparative study of chaotic and white noise signals in digital watermarking. Chaos, Solitons & Fractals. Volume 35, numéro 5, pages 913–921.

[Naschie, 1996] Naschie, M. E. (1996). Introduction to chaos, information and diffusion in quantum physics. Chaos, Solitons & Fractals. Volume 7, numéro 5, pages vii – x.

[Ogorzalek, 2000] Ogorzalek, M. (2000). Controlling chaos and bifurcations in engineering systems, chapitre Design and implementation of chaos control systems, pages 45–69. CRC Press, Taylord and Francis.

[Pevny et Fridrich, 2008]Pevny, T. et Fridrich, J. (2008).Multiclass detector of current steganographic methods for JPEG format, IEEE Trans. on Information Forensics and Security.

[Qin et al., 2009] Qin, J., Sun, X., Xiang, X. et Niu, C. (2009). Principal feature selection and fusion method for image steganalysis, Journal of Electronic Imaging,

[Raja et al., 2004] Raja, K.B., Venugopal, K.R. et Patnaik, L.M. (2004). A Secure Stegonographic Algorithm using LSB, DCT and Image Compression on Raw Images, Technical Report, Department of Computer Science and Engineering, University Visvesvaraya College of Engineering,Bangalore University.

[Saejung et al., 2013] Saejung, S., Boondee, A., Preechasuk, J. et Chantrapornchai, C. (2013). On the comparison of digital image steganography algorithm based on DCT and wavelet, in *Computer Science and Engineering Conference (ICSEC)*, pages 328–333.

[Shi et al., 2005] Shi, Y.Q., Xuan, G., Zou, D., Gao, J., Yang, C., Zhang, Z., Chai, P., Chen, W. et Chen, C. (2005). Image steganalysis based on moments of characteristic functionsusing wavelet decomposition, prediction-error image, and neural network. IEEE,ICME.

[Stavroulakis, 2006] Stavroulakis, P., éditeur (2006). Chaos applications in telecom-munications. CRC Press, Taylord and Francis.

[Wang et al., 2007] Wang, Y., et Moulin, P. (2007). Optimized feature extraction for learning-based image steganalysis, IEEE Trans. on Information Forensics and Security.

[Vapnik, 1995] VAPNIK, V. (1995). The Nature of Statistical Learning Theory, Springer-Verlag.

[Zang et al., 2013] W. Zang, K. Wong, H. Yu, Z. Zhu, " An image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion", CNSNS, 18, 2013, pp. 2066-2080

[Zhang et al., 2009] Zhang, J., Hu, Y. et Yuan, Z. (2009). Detection of LSB Matching Steganography using the Envelope of Histogram, JOURNAL OF COMPUTERS, volume 4, numero 7, pages 645-653

Chapitre 2

STEGANOGRAPHIE SPATIALE

2. STEGANOGRAPHIE SPATIALE

2.1 Introduction

L'apparition de la stéganographie, ou l'art de cacher l'information, remonte au V^{ème} s. av J-C, et est racontée dans les Histoires d' Hérodote. Les grecs rasaient les cheveux d'un esclave, puis tatouaient sur son crâne un message. Une fois les cheveux repoussés, l'esclave pouvait traverser les territoires ennemis sans éveiller les soupçons. Une fois à destination, il suffisait de raser à nouveau le crâne pour récupérer le message.

Les Grecs utilisaient également les tablettes de cire, sur lesquelles ils gravaient leurs écritures. Demeratus, grec à la cour des Perses, qui voulait avertir Sparte d'une invasion imminente de Xerxès, roi de Perse, eut l'idée suivante, il enleva totalement la cire d'une tablette, grava directement son message sur le bois, puis la revêtit à nouveau de cire : elle paraissait neuve!

Une autre forme de stéganographie nous est familière. Il s'agit en effet de l'encre invisible, technique qui a déjà fait ses preuves. L'encre est fabriquée à base de jus d'oignons et de chlorure d'ammoniac. L'écriture est ensuite rendue visible grâce à une source de chaleur (comme une flamme par exemple). Ces encres furent utilisées durant la guerre de sécession (1775-1783) pour transmettre des messages entre George Washington et des agents tels que Benjamin Tallmadge.

Durant la première et la deuxième guerre mondiale et la guerre froide, de nouvelles techniques de stéganographie sont apparues telles que les marques micro-pointes et les microfilms cachés sous des timbres de postes ou sur des couvertures de magazines. Les microfilms sont de petites photographies (de la taille d'un caractère), mais qui peuvent contenir de l'information équivalente à une page d'un livre. Cette technique était très appréciée des Allemands.

De nos jours, avec le développement et la généralisation des moyens des communications numériques et d'Internet, la stéganographie moderne a pris de l'ampleur et de l'importance. En effet, les supports numériques, tels que les fichiers audio, les images ou les vidéos transportant un volume énorme de données échangées, représentent des supports privilégiés pour la stéganographie (transmission d'un message secret, caché dans un média numérique).

La question de la transmission d'une information secrète à travers les canaux de communications publiques intéresse, en plus les gouvernements, les mondes industriels et hospitaliers, voire les particuliers. Notons aussi que la sécurité des données échangées peut aussi être associée à un système de protection par cryptographie. Dans ce cas, toutes les données sensibles sont chiffrées avant leur transmission.
Le principe de la stéganographie moderne est de cacher un message secret *m*, de taille relativement importante, dans un média numérique (audio, image ou vidéo), appelé média cover (ou original), de sorte que le média résultant, appelé média stégo, reste sensiblement, au moins à l'œil nu, identique au média cover. Cela signifie que l'existence du message secret dans le média stégo est imperceptible et pratiquement indétectable.

Dans un média cover tel qu'une image ou une vidéo, le message secret peut être du texte brut, du texte chiffré ou une image. Ces médias transitent couramment via l'Internet, et sont un excellent support pour cacher une information secrète.
Nous donnons dans la figure 2.1, le principe de la stéganographie, dans le cas où le média cover est une image numérique (image de Lena) et le message secret lui aussi est une image numérique (image Bateau).

Le processus d'insertion dépend d'une clé secrète qui est une information secrète supplémentaire comme un mot de passe.
Le système est dit être sécurisé si l'on ne peut distinguer la différence entre une image originale et une image stégo.

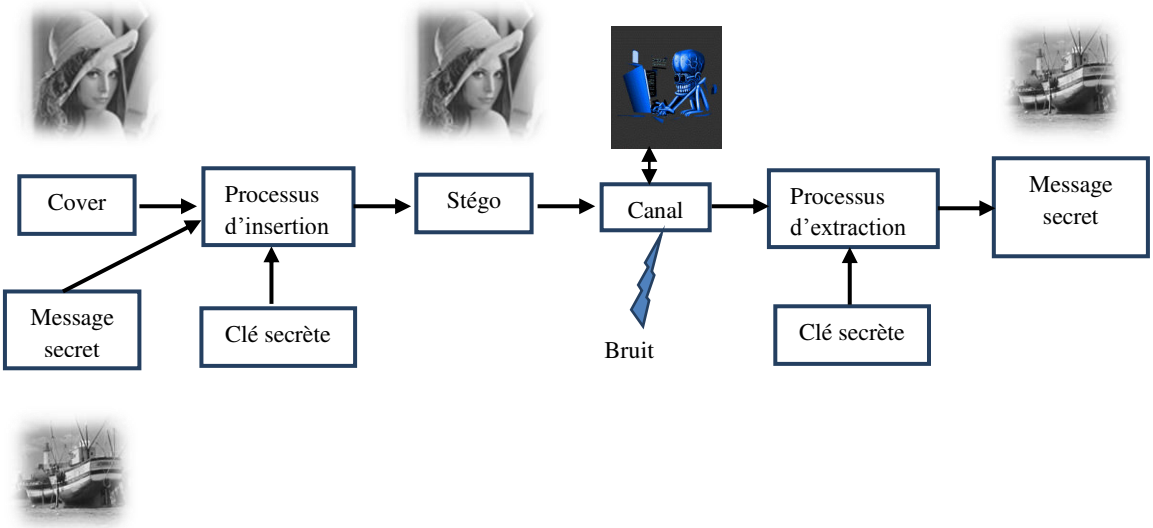


Figure 2.1 : Principe de la stéganographie

Les signaux chaotiques, contrairement aux signaux aléatoires, obéissent à des lois déterministes, parfois assez simples dans leur représentation mathématique. Ces signaux ou séquences chaotiques se caractérisent par des propriétés génériques fondamentales : larges bandes en fréquences, non-linéarité, sensibilité aux conditions initiales et aux paramètres du système, et très grand nombre de codes ou séquences ayant de très faibles inter-corrélations.
Afin de protéger la confidentialité du message secret, dans le cas de sa détection par un « adversaire », il faut le chiffrer ou l'insérer de façon aléatoire. Dans ce chapitre, nous protégeons la confidentialité du message secret par un procédé d'insertion chaotique des positions spatiales du message secret.

Le chapitre est organisé comme suit.
Dans la section 2.2, nous rappelons le principe et la description mathématique de la stéganographie, et dans la section 2.3, nous donnons quelques exemples de ses applications. Dans la section 2.4, nous présentons l'état de l'art de la stéganographie spatiale.
Dans la section 2.5, nous introduisons l'amélioration de la sécurité, par l'ajout d'un système chaotique, de deux méthodes de stéganographie LSB adaptatives les plus efficaces.
Dans la section 2.5.1, nous décrivons en détails la méthode EAELSB, qui repose sur une amélioration de l'intégration adaptative des données secrètes dans les régions bords par

substitution des LSB. Le système chaotique proposé pour sécuriser les positions des bits du message secret est décrit dans la section 2.5.1.1. Les processus d'insertion et d'extraction adaptés sont présentés dans les sections 2.5.1.2 et 2.5.1.3. Les performances de la méthode (et celle de référence correspondante) en termes de qualité visuelle et de qualité évaluée par les paramètres PSNR, IF et SSIM, sont données dans la section 2.5.1.4.

Dans la section 2.5.2, nous présentons le système de la méthode EEALSBMR, qui repose elle sur une : amélioration de la stéganographie adaptative dans les régions bords par correspondance de LSB revisitée. Les sections 2.5.2.1 et 2.5.2.2 décrivent les processus d'insertion et d'extraction adaptés. La section 2.5.2.3 présente les performances de la méthode (et celle de base correspondante) en termes de qualité visuelle et de qualité évaluée par les paramètres PSNR, IF et SSIM, avant de conclure dans la section 2.6.

2.2 Principe et description mathématique de la stéganographie

De manière intuitive, la stéganographie peut se résumer à l'histoire suivante. Alice et Bob sont en prison, enfermés dans deux cellules séparées l'une de l'autre, et souhaiteraient planifier une évasion. Ils sont autorisés à communiquer par le biais de messages surveillés, afin qu'ils ne discutent pas d'un plan pour s'échapper. La personne qui surveille les messages est Eve, la gardienne. Si Eve détecte le moindre signe de conspiration, elle transférera Alice et Bob dans une prison de haute sécurité, où personne n'a jamais pu s'évader. Alice et Bob, avant d'être enfermés, ont partagé un secret commun qu'ils vont utiliser afin de cacher leur plan d'évasion dans des messages innocents de sorte qu'Eve ne puisse soupçonner l'existence d'un secret caché.

Ci-dessous nous nous intéressons à la formulation mathématique de la stéganographie.
Soit K est l'ensemble des clés possibles, M est l'ensemble des messages secrets possibles à insérer, et C est l'ensemble des supports possibles.

Un schéma sténographique est défini par deux fonctions :

- La fonction d'insertion (Embedding : Emb), qui prend en paramètre : un support, un message et une clé privée, et retourne un élément de C

$$Emb : C \times M \times K \rightarrow C \tag{2.1}$$

Le résultat de la fonction *Emb* est habituellement appelée stégo-média, ou stégo, et le support c avant insertion est appelé média, ou cover.

- La fonction d'extraction (Ext) est définie par :

$$Ext : C \times K \rightarrow M \tag{2.2}$$

La fonction d'extraction permet de retrouver le message secret inséré dans un support. Plus précisément, ceci se traduit mathématiquement par :

$$Ext(Emb(c, m, k)) = m, \ \forall (c, m, k) \in C \times M \times K \tag{2.3}$$

Les algorithmes étudiés dans ce chapitre sont destinés à insérer des messages (texte ou

image) dans des images. On assimile l'ensemble des messages M à l'ensemble$\mathbb{F}_2^l = \{0,1\}^l$, l étant la taille du message en bits, et $\mathbb{F}_2$ étant le corps de Galois à deux éléments. Ainsi, chaque caractère d'un texte ou un pixel d'un plan d'une image est représenté par 8 éléments binaires, donc, par une valeur entière comprise entre 0 et 255. De même, on assimile l'ensemble des supports C à $\mathbb{F}_2^n = \{0,1\}^n$, n étant la taille du support en bits, à l'aide d'une certaine fonction, par exemple en prenant la valeur du LSB (Least Significant Bit : bit de poids faible, c'est à dire le bit se trouvant le plus à droite dans la représentation d'un nombre entier en binaire) de chaque pixel de l'image.

Certains algorithmes utilisent en plus une technique de cryptographie, en insérant alors non pas le message en lui-même, mais le message chiffré par un algorithme de chiffrement donné. L'objectif étant d'ajouter une sécurité supplémentaire au schéma, en empêchant de retrouver le message en clair dans le cas de sa détection par une attaque.

2.3 Applications de la stéganographie

Après la définition de la stéganographie, des questions apparaissent de façon automatique : à quoi peut bien servir la stéganographie? Pourquoi dissimuler un message si l'on n'a rien à se reprocher? Dans le passé, cette science a toujours été utilisée à des fins d'espionnage, pourtant, de nos jours, la stéganographie n'est pas toujours synonyme d'insécurité et peut, au contraire, servir à protéger le droit. Ci-dessous, nous donnons quelques exemples d'utilisation de la stéganographie.

1. Utilisations malveillantes :

Internet est une source inépuisable de ressources, la quantité innombrable d'images qui circulent sur le web ainsi que les nombreux fichiers audio qui s'échangent via les communications point à point (P2P) rendent difficile la détection de messages cachés dans ses médias. Un attaquant peut alors utiliser la stéganographie pour cacher un code malveillant fragmenté sur plusieurs images stégo, et procéder ensuite au réassemblage du code malveillant directement sur l'ordinateur de la victime.

La stéganographie peut être aussi utilisée pour dissimuler des messages interdits sur des images ou photos anodines et les échanger via l'internet sans provoquer le moindre signe de soupçon. Par exemple, des réseaux terroristes ou des pédophiles qui s'échangent des messages secrets à travers le web. De plus, la stéganographie est aussi utilisée pour l'espionnage industriel. En effet, cette technique semble bien adaptée au transfert de l'information confidentielle volée ou obtenue par corruption.

2. Utilisations légitimes :

Dans certains pays non démocratiques où la liberté d'expression est totalement interdite, la stéganographie apparaît comme un moyen de communiquer plus librement. Dans ces pays, l'utilisation de la stéganographie est illicite mais son usage dans ce cas est légitime. Aussi, lors d'une guerre entre deux nations, la stéganographie est utilisée, de part et d'autre, pour transmettre des messages secrets qui ne doivent pas tomber dans des mains ennemies et si la communication est interceptée, le message caché ne doit être dévoilé (message chiffré ou caché de façon aléatoire ou chaotique).

2.4 Etat de l'art de la stéganographie spatiale

Les techniques de stéganographie sont classées en : stéganographie par substitution de LSB, stéganographie par correspondance de LSB, stéganographie par Différence des valeurs des pixels (PVD : Pixel-Value differencing) et stéganographie Adaptative.

[Chang et al., 2002] et [Thien et Lin, 2003] proposent chacun une méthode de stéganographie par substitution de LSB. Cette technique est couramment utilisée et consiste à remplacer directement les bits de poids faibles des pixels de l'image cover (image originale) par les bits du message secret pour obtenir l'image stégo. Etant donné que seul le bit le moins significatif des pixels est modifié, il est en pratique très difficile de remarquer visuellement un quelconque changement de l'image stégo par rapport à l'image cover. La capacité de l'algorithme est de 1 bit par pixel.

[Wang et al., 2000 et 2001] ont travaillé à améliorer la qualité visuelle de l'image stégo en utilisant une technique de stéganographie basée sur un algorithme génétique. [Chan et cheng, 2001 et 2004], démontrent que l'utilisation d'un procédé génétique n'est pas nécessaire et proposent un processus optimal d'ajustement de pixels (OPAP : Optimal Pixel Adjustment Process) pour améliorer l'efficacité et la qualité visuelle de l'image stego générée par simple substitution de LSB.

La stéganographie par correspondance de LSB a été introduite par [Ker, 2004]. Elle modifie également les LSB de l'image originale pour cacher les bits du message secret, mais si le bit secret ne correspond pas au bit de poids faible de l'image cover, un 1 sera ajouté ou soustrait de la valeur de pixel de façon aléatoire.

Le procédé PVD a été introduit par [Wu et Tsai, 2003]. Il utilise la sensibilité HVS (Human Vision Sensitivity) aux variations d'intensité des régions lisses par rapport à celles des bords afin d'améliorer la qualité de l'image stégo.

[Wu et al., 2005] ont proposé un schéma de dissimulation de données avec une meilleure qualité d'image stégo, en combinant les méthodes d'insertion LSB et PVD. Dans leur approche, si la valeur de la différence entre deux pixels consécutifs est faible (zone lisse) ils utilisent la méthode de substitution de LSB, autrement dit le procédé PVD (zone de bord).

[Yang et al., 2008], ont proposé une méthode de stéganographie adaptative utilisant le procédé PVD et la substitution de LSB. Dans cette approche, la valeur de la différence de deux pixels consécutifs (PVD) est utilisée pour estimer le nombre k de bits du message à cacher dans chacun de deux pixels par substitution de LSB. La valeur k est adaptative selon la gamme de valeurs de différence à laquelle elle appartient. Plus précisément, la méthode utilise trois gammes de valeurs de la différence PVD :

$$k = 3, si\ PVD\ \in [0, 15];\ \ k = 4, si\ PVD\ \in [16, 31];\ \ k = 5, si\ PVD\ \in [32, 255]$$

Si après la procédure d'insertion de $2k$ bits de message, la gamme de valeurs de différence PVD a changé, alors une phase de rajustement est utilisée pour ramener à la même gamme de

valeurs de différence avant l'insertion. L'approche proposée fournit une capacité d'insertion importante tout en préservant une qualité d'image stégo élevée.

[Luo et al., 2010] ont proposé une méthode par correspondance de LSB revisitée adaptative. La méthode proposée est très intéressante car, contrairement à la plupart des méthodes de la littérature, elle tient compte de la relation entre le contenu de l'image elle-même et la taille du message secret. En effet, ce schéma sélectionne les régions d'insertion selon la taille de message secret et la valeur de la différence entre deux pixels consécutifs (PVD) dans l'image originale. Pour une taille de message secret faible, seules les régions de bord les plus aiguës sont utilisées en laissant les autres régions plus lisses inchangées. Pour une taille de message secret assez grande, plus de régions de bord peuvent être sélectionnées de manière adaptative selon la taille du message à insérer, en ajustant la valeur d'un paramètre seuil égale à la valeur absolue du PVD. Par ailleurs, la méthode inclut une procédure d'ajustement des pixels concernés par l'insertion si leur nouvelle valeur n'appartient pas à l'intervalle [0, 255], ou si leur PDV absolu est inférieur au seuil choisi. De ce fait, ce schéma de stéganographie est hautement adaptatif et donc très efficace.

Par la suite, nous étudions et réalisons les deux méthodes de stéganographie LSB adaptatives citées ci- dessus en leur rajoutant un système chaotique pour la sécurité du contenu du message dans le cas de sa détection par un adversaire.

2.5 Contribution : Méthodes de stéganographie LSB adaptatives sécurisées par séquences chaotiques.

A notre connaissance, parmi les différentes méthodes de dissimulation de données existantes dans la littérature actuelle, les méthodes de stéganographie LSB adaptatives sont les plus efficaces et plus particulièrement la méthode proposée par [Yang et al., 2008] et celle proposé par [Luo et al., 2010]. Cependant, l'aspect sécurité du contenu du message, dans le cas de sa détection par un adversaire, n'est pas assurée par la méthode de Yang et al., et est très faible dans la méthode de Luo et al. Nous nous proposons d'améliorer la sécurité du contenu de message de ces deux méthodes par l'ajout d'un système chaotique produisant des séquences chaotiques robustes.

2.5.1 Amélioration de l'Intégration adaptative des données secrètes dans les régions bords par substitution des LSB (Enhanced Adaptive data hiding in Edge areas of images with spatial LSB domain systems : EAELSB)

L'approche de l'intégration adaptative des données secrètes par substitution LSB est basée sur le fait que les zones de bord peuvent supporter un plus grand nombre de changements que les zones lisses. La valeur de la différence de deux pixels consécutifs (PVD) est utilisée pour distinguer les zones de bord et les zones lisses, et ainsi déterminer le nombre k de bits de message à insérer dans chaque pixel concerné. Dans cette méthode, l'intervalle [0, 255] de valeurs de différence PVD est divisé en trois gammes : gamme base, gamme moyenne et gamme haute. La table 2.1, donne la correspondance entre gammes (intervalles) et nombre k de bits à insérer.

Pour conserver pratiquement la même qualité visuelle de l'image stégo par rapport à l'image cover, les valeurs de différence avant et après l'intégration des données secrètes doivent appartenir à la même gamme (même niveau). Si, après intégration des données secrètes, la valeur de différence se transforme en un autre niveau, une phase de réajustement des valeurs des pixels impliqués est utilisée afin de ramener la nouvelle différence des valeurs à la même gamme avant insertion [Chan et Cheng, 2004].
La spécificité de cette méthode est sa grande capacité d'insertion tout en gardant une qualité haute.

	R	K
Bas niveau	R_1= [0 ; 15]	k_1=3
Niveau moyen	R_2= [16 ; 31]	k_2=4
Haut niveau	R_3= [32 ; 255]	k_3=5

Table 2.1 : Correspondance entre gammes (intervalles) et nombre de bits à insérer

Remarque : il faut que $k_i \leq k_{i+1}$ et $k_i \leq \log_2|R_i|$ avec $|R_i|$= largeur de l'intervalle R_i.

Dans cette méthode, l'image est divisée en un ensemble de blocs de deux pixels chacun. De même, on utilise la décomposition des intervalles et la différence entre les 2 pixels voisins de valeurs p_i et p_{i+1} pour identifier les zones contours el les zones lisses.
Le schéma de stéganographie sécurisé proposé, utilise pratiquement (après adaptation) les mêmes procédures d'insertion et d'extraction des bits du message secret de [Yang et al., 2008] mais il inclut un système chaotique permettant de sécuriser les positions des pixels concernés par l'insertion (Figure 2.2). En effet, contrairement à la méthode de [Yang et al., 2008] où l'intégration de message secret se fait séquentiellement de haut en bas, et de gauche à droite, le système chaotique choisit de façon quasi chaotique les positions des pixels concernés par l'insertion. De cette manière, dans le cas de sa détection le message est sécurisé.

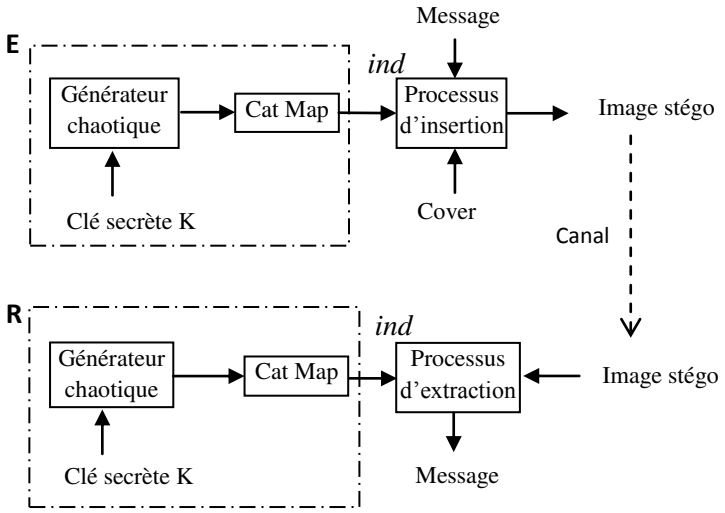


Figure 2.2 : Schéma de stéganographie sécurisé proposé

Ci-dessous, nous décrivons les éléments de la figure 2.2, à savoir le système chaotique proposé, le procédé d'insertion, le procédé d'ajustement et le procédé d'extraction [Tataru et al., 2012], [Battikh et al., 2013], [Yang et al., 2008].

2.5.1.1 Description du système chaotique proposé

Générateur de séquence chaotique

Le générateur de séquences discrètes chaotiques proposé présente des orbites ayant des très grandes longueurs. Il est basé sur deux filtres numériques non linéaires perturbés IIR (cellules) avec deux retards, connectés en parallèles. Les deux fonctions non-linéaires utilisées sont les cartes chaotiques PWLCM et SKEW TENT. La technique de perturbation appliquée à chaque cellule est réalisée par un registre à décalage à rétroaction linéaire m-LFSR (Figure 2.3). L'association de la cascade et de la technique de perturbation permet l'extension et le contrôle des orbites générées [El Assad et Noura, 2010]. En effet, si Δ_1 et Δ_2 sont respectivement la longueur des orbites des cellules de sorties s_1 et s_2 sans perturbation, la longueur minimale de l'orbite de la sortie s du générateur est donnée par :

$$o_{min} = ppcm\left[\Delta_1 \times \left(2^{k_1} - 1\right), \Delta_2 \times \left(2^{k_2} - 1\right)\right] \tag{2.4}$$

Où $ppcm$ est le plus petit commun multiple, et k_1, k_2 sont les degrés des polynômes primitifs de des LFSRs. Les différentes équations mise en œuvre du générateur chaotique sont :

$$s_i(n) = NLF_i\{u_i(n-1), p_i\}, i = 1,2 \tag{2.5}$$

$$u_i(n-1) = mod\left\{s_i(n-1) \times c_{i,1} + s_i(n-2) \times c_{i,2}, 2^N\right\}, i = 1,2 \tag{2.6}$$

$$s(n) = s_1(n) + s_2(n) \tag{2.7}$$

Les deux cartes chaotiques discrètes PWLCM et SKEW TENT sont définies par les relations suivantes :

$$s_1(n) = NLF_1[u_1(n-1), p_1]$$

$$= \begin{cases} \left\lfloor 2^N \times \frac{u_1(n-1)}{p_1} \right\rfloor & si\ 0 \le u_1(n-1) < p_1 \\ \left\lfloor 2^N \times \frac{2^N - u_1(n-1)}{2^N - p_1} \right\rfloor & si\ p_1 \le u_1(n-1) < 2^{N-1} \\ NLF_1[2^N - u_1(n-1)] & ailleurs \end{cases} \tag{2.8}$$

$$s_2(n) = NLF_2[u_2(n-1), p_2]$$

$$= \begin{cases} \left\lceil 2^N \times \frac{u_2(n-1)}{p_2} \right\rceil & si\ 0 \le u_2(n-1) < p_2 \\ \left\lfloor 2^N \times \frac{2^N - u_2(n-1)}{2^N - p_2} \right\rfloor + 1 & si\ p_2 \le u_2(n-1) < 2^N \end{cases} \tag{2.9}$$

Le paramètre de contrôle p_1 utilisé par la carte PWLCM discrétisée doit avoir une valeur comprise entre 1 et $2^{N-1} - 1$, et la valeur du paramètre de contrôle $p_2$utilisé par la carte SkewTent est comprise entre 1 et $2^N - 1$, avec *N*=32 est le nombre de bits utilisé pour la

quantification. Les paramètres $c_{1,1}, c_{1,2}, c_{2,1}, c_{2,2}$ font partie de la clé secrète et peuvent prendre des valeurs comprises entre 1 et $2^N - 1$.
La taille $|K|$de la clé secrète est formée par tous les paramètres du système et toutes les conditions initiales.

$$|K| = 6N + k_1 + k_2 + (N-1) + N + 4N$$

Si, nous prenons $k_1 = 23$ et $k_2 = 21$, alors la taille de la clé secrète est de 427 bits. Nous savons qu'une taille de la clé secrète de 128 bits est largement suffisante pour toute application industrielle sécurisée.

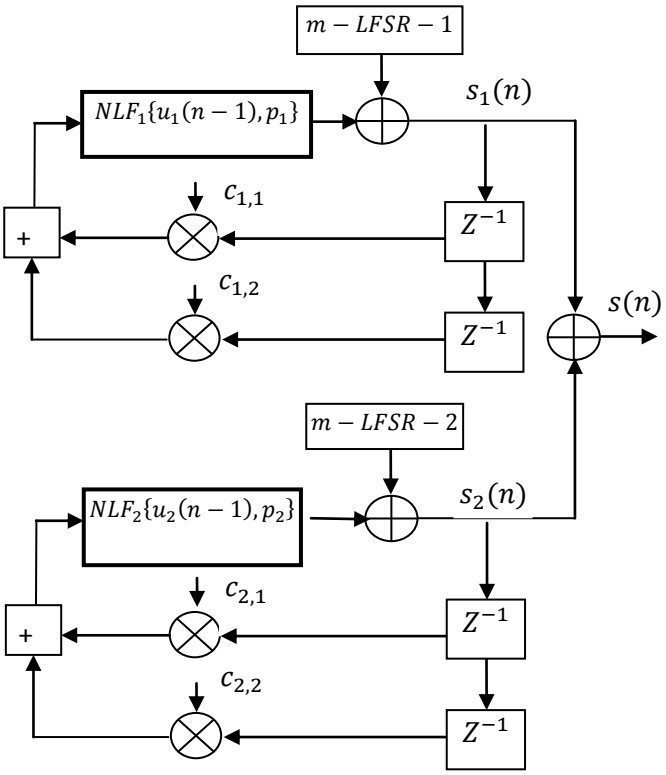


Figure 2.3 : Générateur chaotique

Carte chaotique 2-D Cat utilisée

La sortie *s* du générateur de séquences chaotiques fournit les clés dynamiques, paramètres $\{u, v, r_l, r_c\}$ de la carte chaotique 2-D Cat.
Le rôle de la carte Cat est de calculer les positions quasi-chaotiques des pixels concernés par le processus d'insertion des bits du message. L'équation standard de permutation de la carte chaotique Cat est définie par :

$$\begin{bmatrix} i_n \\ j_n \end{bmatrix} = mod\left\{ \begin{pmatrix} 1 & u \\ v & 1+uv \end{pmatrix} \times \begin{bmatrix} i \\ j \end{bmatrix} + \begin{bmatrix} rl+rc \\ rc \end{bmatrix}, \begin{bmatrix} M \\ M \end{bmatrix} \right\} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (2.10)$$

Où (i_n, j_n) sont les nouvelles positions après permutation des pixels d'insertion des *2k* bits du message dans l'image stégo de dimension *M* x *M* et (*i, j*) correspond aux positions d'origines.

Dans notre travail, au lieu d'utiliser l'équation de permutation ci-dessus, nous avons utilisé l'équation modifiée ci-dessous qui est plus efficace en termes de calcul en code Matlab.

$$\begin{bmatrix} M_{ln} \\ M_{cn} \end{bmatrix} = mod\left\{ \begin{pmatrix} 1 & u \\ v & 1+uv \end{pmatrix} \times \begin{bmatrix} M_l \\ M_c \end{bmatrix} + \begin{bmatrix} r_l+r_c \\ r_c \end{bmatrix}, \begin{bmatrix} M \\ M \end{bmatrix} \right\} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (2.11)$$

où M_l, M_c et M_{ln}, M_{cn} sont les indices originaux et les indices permutés de position. M_l et M_c sont des matrices carrées respectivement ligne et colonne de la forme suivante :

$$M_l = \begin{pmatrix} 1\,1 & . & . & 1 \\ 2\,2 & & & 2 \\ . & . & & . \\ . & . & & . \\ M\,M & . & . & M \end{pmatrix}; M_c = \begin{pmatrix} 1\,2 & . & . & M \\ 1\,2 & & & M \\ . & . & & . \\ . & . & & . \\ 1\,2 & . & . & M \end{pmatrix}$$

La matrice indice de permutation *ind* est donnée par :

$$ind = (M_{ln} - 1) + M \times (M_{cn} - 1) + 1 \quad (2.12)$$

2.5.1.2 Processus d'insertion :

Les différentes étapes de la procédure d'insertion sont décrites ci-dessous :

1. Diviser l'image originale (cover) en blocks de deux pixels consécutifs non chevauchés.
2. Choisir la position du bloc (p_i, p_{i+1}) d'insertion des bits du message de façon quasi chaotique en utilisant la valeur *ind* du système chaotique. Rappelons que dans la méthode de [Yang et al., 2008] cette étape est réalisée en balayant séquentiellement les blocs du haut en bas et de gauche à droite dans la direction horizontale jusqu'à la fin de l'image cover.
3. Calculer la valeur absolue de la différence entre les deux pixels $d = |p_i - p_{i+1}|$ du bloc choisi.
4. Identifier la valeur *k* correspondant à cette *d* selon la table 2.1.
5. Remplacer les *k* derniers bits de chacun des pixels du bloc sous traitement par *k* bits du message à cacher (ceci permet de cacher par bloc *2k* bits du message secret).
6. Appliquer le processus d'ajustement optimal de pixels (modified LSB substitution) ci-dessous [Chan et Cheng, 2004]

   Le processus d'ajustement optimal de pixels (OPAP) est proposé pour améliorer la qualité de l'image stégo. En effet :

   Soient p_i , p_i' les valeurs des pixels correspondants aux *i-th* pixels de l'image originale, et de l'image stégo utilisant la méthode de substitution LSB simple. Soit δ_i la différence entre p_i et p_i' .

$$\delta_i = p_i' - p_i \quad (2.13)$$
$$-2^k < \delta_i < 2^k \quad (2.14)$$

Segmentons la valeur δ_i en trois intervalles :

$$\text{Cas 1} : 2^{k-1} < \delta_i < 2^k$$

$$\text{Cas 2} : -2^{k-1} \leq \delta_i \leq 2^{k-1} \qquad (2.15)$$

$$\text{Cas 3} : -2^k < \delta_i < -2^{k-1}$$

En se basant sur les trois intervalles, la méthode d'ajustement modifie le pixel p_i' en $p_i^{"}$, comme suit :

Cas 1 ($2^{k-1} < \delta_i < 2^k$) :

if $p_i' > 2^k$,

$p_i^{"} = p_i^{'} - 2^k$;

else

$p_i^{"} = p_i^{'}$;

end

Cas 2 ($-2^{k-1} \leq \delta_i \leq 2^{k-1}$) :

$p_i^{"} = p_i^{'}$;

Cas 3 ($-2^k < \delta_i < -2^{k-1}$) :

If $p_i' < 256 - 2^k$,

$p_i^{"} = p_i^{'} + 2^k$;

else

$p_i^{"} = p_i^{'}$;

end

7. Calculer la nouvelle valeur absolue de la différence entre les deux pixels du bloc résultant de l'étape 6 (notés ici (p_i', p_{i+1}')) $d' = |(p_i' - p_{i+1}')|$ et tester si elle appartient à la même gamme ou intervalle que d selon la table 2.1. Si oui, alors le bloc stégo (p_i', p_{i+1}') porte une partie du message à cacher et aller à l'étape 2. Sinon, aller à l'étape 8.

8. Processus d'ajustement final :

   Le processus d'ajustement optimal des pixels est proposé pour que la ressemblance entre l'image stégo obtenue et l'image cover soit maximale.

a. Si d appartient à un intervalle dont les valeurs sont plus petites que l'intervalle sur lequel appartient d' alors :

   i. Si $p'_i \geq p'_{i+1}$, prendre comme valeurs des nouveaux pixels (p''_i, p''_{i+1}) le meilleur choix entre $(p'_i, p'_{i+1} + 2^k)$ et $(p'_i - 2^k, p'_{i+1})$ (valeurs les plus proches de (p_i, p_{i+1}) au sens de l'erreur quadratique moyenne (EQM).
   ii. Sinon, prendre comme valeurs des nouveaux pixels (p''_i, p''_{i+1}) le meilleur choix entre $(p'_i, p'_{i+1} - 2^k)$ et $(p'_i + 2^k, p'_{i+1})$ (valeurs les plus proches de (p_i, p_{i+1}) au sens de l'EQM.

b. Si d appartient à un intervalle dont les valeurs sont plus grandes que l'intervalle sur lequel appartient d' alors :

   i. Si $p'_i \geq p'_{i+1}$, prendre comme valeurs des nouveaux pixels (p''_i, p''_{i+1}) le meilleur choix entre $(p'_i, p'_{i+1} - 2^k)$ et $(p'_i + 2^k, p'_{i+1})$ (valeurs les plus proches de (p_i, p_{i+1}) au sens de l'EQM.
   ii. Sinon, prendre comme valeurs des nouveaux pixels (p''_i, p''_{i+1}) le meilleur choix entre $(p'_i, p'_{i+1} + 2^k)$ et $(p'_i - 2^k, p'_{i+1})$ (valeurs les plus proches de (p_i, p_{i+1}) au sens de l'EQM.

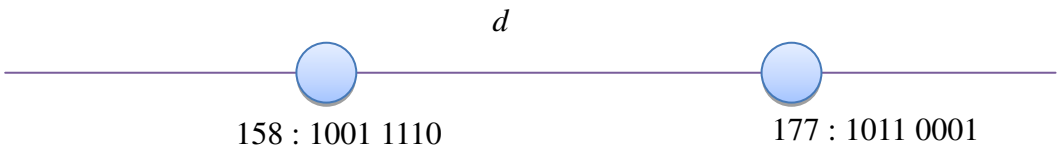
Le meilleur choix au sens de l'erreur quadratique moyenne est défini par :

$$EQM\{(p_i, p_{i+1}), (p''_i, p''_{i+1})\} = (p_i - p''_i)^2 + (p_{i+1} - p''_{i+1})^2$$

Exemple

1. Soit un bloc de deux pixels ayant les valeurs : 158, 177, et un message secret suivant :

   1001 1001

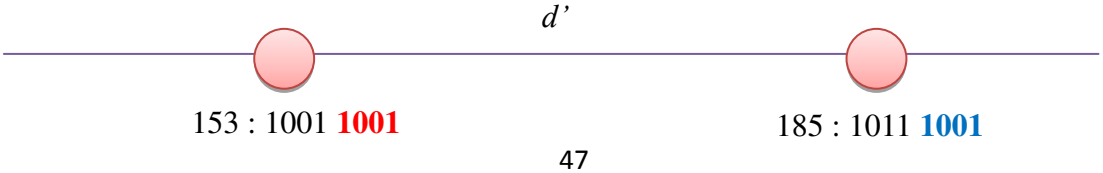


On calcule la distance :

$$d = |p_i - p_{i+1}| = |158 - 177| = 19$$

2. $d \in R_2 \Rightarrow k = 4$

3. Substitution de 4 bits LSB de deux pixels par les 8 bits MSB du message vecteur :



4. $\delta_i = p'_i - p_i$= 153-158 = -5

   δ_i appartient à l'intervalle 1 : $-\mathbf{8} \leq \boldsymbol{\delta_i} \leq 8$

   Alors $p''_i = p'_i = 153$

   $\delta_{i+1} = p'_{i+1} - p_{i+1} = 185 - 177 = 8$

   δ_i appartient à l'intervalle 1 : $-\mathbf{8} \leq \boldsymbol{\delta_{i+1}} \leq 8$

   Alors $p''_{i+1} = p'_{i+1} = 185$

5. $d' = |p''_i - p''_{i+1}| = |153 - 185| = 32$, $d' \in R_3 = R_3 \Rightarrow$ processus d'ajustement

d<d' et $p'_i \leq p'_{i+1}$, alors :

(p''_i, p''_{i+1}) est le meilleur choix entre :

$(p''_i, p''_{i+1}) = (p'_i, p'_{i+1} - 2^k) = (153,169)$

$(p''_i, p''_{i+1}) = (p'_i + 2^k, p'_{i+1}) = (169,185)$

L'EQM entre (158,177) et (153,169) = 9.434

L'EQM entre (158,177) et (169,185) = 13.6015

Donc le meilleur choix est 153,169

2.5.1.3 Procédure d'extraction

La procédure d'extraction du message secret à partir de l'image stégo est relativement plus simple que celle de l'insertion. Elle contient les étapes suivantes :

1. Diviser l'image stégo en blocs de deux pixels consécutifs non recouvrants.
2. Choisir la position du bloc stégo (p''_i, p''_{i+1}) contenant les bits du message secret en utilisant le même système chaotique de l'émission qui fournit la même valeur *ind*.

Ceci suppose bien entendu que le récepteur connaisse préalablement la clé secrète du système chaotique.

3. Calculer la valeur absolue de la différence entre les deux pixels $d^{"} = \left|p_i^{"} - p_{i+1}^{"}\right|$ du bloc choisi, et chercher la valeur de *k* correspondante.
4. Extraire les *k* LSB bits de chacun de deux pixels, les sauvegarder dans le vecteur message *M* et aller à l'étape 2. Les étapes 2 à 5 sont répétées jusqu'à exploration de tous les blocs pour extraire le contenu total du message secret.

Exemple

1. $\left(p_i^{"}, p_{i+1}^{"}\right)$ = (153, 169)

2. $d^{"} = \left|p_i^{"} - p_{i+1}^{"}\right| = |153 - 169| = 16$
   $d^{"} = 16 \in R_2 \Rightarrow k = 4$

3. 153 : 10011001
   169 : 10101001
   Extraire le 4 bits LSB de chaque pixel :
   Le message secret est : 1001 1001

2.5.1.4 Résultats expérimentaux obtenus par les méthodes EAELSB et AELSB

Pour les deux méthodes EAELSB et AELSB, nous donnons tout d'abord ci-dessous un exemple sur l'impact du processus d'insertion d'un message secret dans une image cover. Ensuite nous quantifions les performances de deux méthodes en utilisant les paramètres usuels de mesure, à savoir les **PSNR**, **IF** et **SSIM :**

PSNR (Peak Signal to Noise Ratio), IF (Image Fidelity) et SSIM (Structural similarity index), couramment cités dans la littérature spécialisée. Le PSNR mesure le niveau de distorsion d'une image stéganographiée par rapport à l'image originale. Le **PSNR** est considéré comme une mesure très indicative. IF est une mesure de la précision de la répartition de la luminosité. Le **SSIM** mesure la similarité de structure entre les images cover et stégo, il est un meilleur indicateur de qualité que le **PSNR** [Wang et al., 2004].

Ces paramètres sont donnés dans les équations (2.16), (2.17) et (2.18) respectivement :

$$PSNR = 10 \times \log_{10}\left(\frac{Max\ p_c^2(i,j)}{\frac{1}{M\times N}\left(\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[p_c(i,j)-p_s(i,j)]^2\right)}\right) \quad (2.16)$$

$$IF = 1 - \frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[p_c(i,j)]^2}{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[p_c(i,j)-p_s(i,j)]^2} \quad (2.17)$$

$$SSIM = \frac{(2\mu_c\mu_s+c_1)(2cov_{cs}+c_2)}{(\mu_c^2+\mu_s^2+c_1)(\sigma_c^2+\sigma_s^2+c_2)} \quad (2.18)$$

Où $p_c(i,j)$) et $p_s(i,j)$) sont les valeurs du pixel de la i ème ligne et de la j ième colonne des images cover et stégo, et M et N sont les tailles de des images considérées.

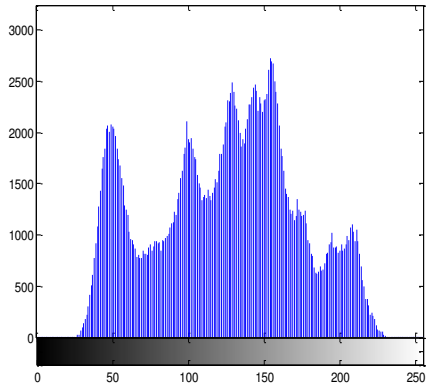
μ_c est la moyenne de image originale (cover); μ_s est la moyenne de l'image stégo; σ_c^2 est la variance de l'image originale; σ_s^2 est la variance de l'image stégo; cov_{cs} est la covariance des images originale et stégo; $c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$ sont deux variables destinées à stabiliser la division quand le dénominateur est très faible; L est La gamme dynamique des valeurs des pixels, soit 255 et k_1, k_2 sont deux constantes plus petites que 1, $k_1 \ll 1, k_2 \ll 1$, dans notre expérimentation nous avons pris, $k_1 = 0.05, k_2 = 0.05$
Les facteurs PSNR, IF et SSIM indiquent des bons résultats en termes de qualité visuelle de l'image stégo pour un PSNR > 40 dB, un IF et un SSIM proche de 1.

Impact visuel du processus d'insertion

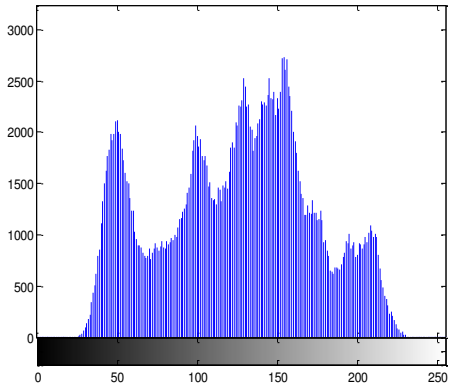
Nous présentons dans les figures 2.4 et 2.5, l'impact visuel du processus d'insertion d'un message secret (bateau) de taille 128 x 128, dans l'image originale « Lena » de taille 512 x 512 pour les méthodes EAELSB et AELSB respectivement.
Nous montrons dans la figure 2.4 (méthode EAELSB) : a) l'image cover, b) l'image stégo, c) l'histogramme de l'image cover, d) l'histogramme de l'image stégo, e) le message secret, et f) la différence amplifiée de l'image stégo et cover. Visuellement, nous remarquons que les images cover et stégo sont similaires ainsi que leur histogrammes. Nous observons aussi dans la figure 2.4 f) que la différence amplifiée de deux images cover et stégo qui reflète la présence du message est répartie de façon quasi chaotique sur toute l'image différence. Ceci assure la sécurité du contenu du message dans le cas de sa détection par un adversaire.

(a) image originale

(b) image stégo

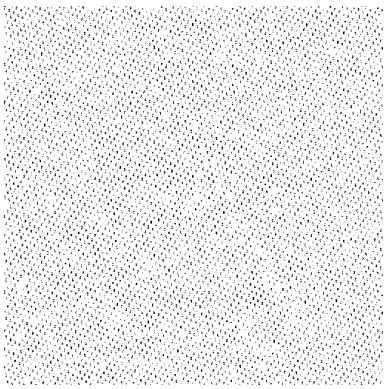


(c) histogramme de l'image originale



(d) histogramme de l'image stégo

(e) message secret



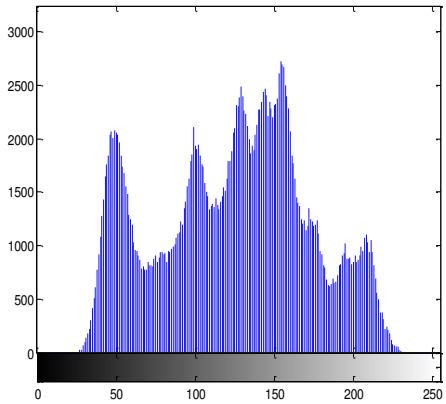
(f) image différence amplifiée

Figure 2.4 : Impact visuel du processus d'insertion par la méthode EAELSB : (a) image originale, (b) image stégo (c) histogramme de l'image originale, (d) histogramme de l'image stégo, (e) message secret inséré, (f) différence entre les images originale et stégo.

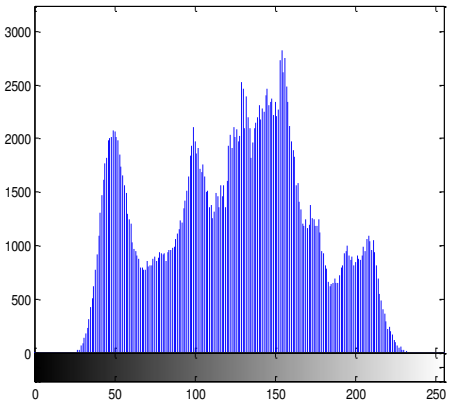
Nous montrons dans la figure 2.5 (méthode AELSB) : a) l'image cover, b) l'image stégo, c) l'histogramme de l'image cover, d) l'histogramme de l'image stégo, e) le message secret, et f) la différence amplifiée de l'image stégo et cover. Visuellement, nous remarquons que les images cover et stégo sont similaires ainsi que leur histogramme. Nous observons aussi dans la figure 2.5 f) que la différence amplifiée de deux images cover et stégo, qui reflète la présence du message, est localisée dans la partie supérieure de l'image différence. Ceci représente une faiblesse de la méthode sur le plan de la sécurité du contenu du message dans le cas de sa détection par un adversaire.

(a) image originale

(b) image stégo

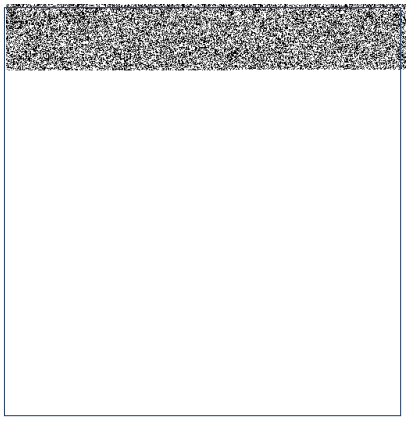


(c) histogramme de l'image originale



(d) histogramme de l'image stégo

(e) message secret



(f) image différence amplifiée

Figure 2.5 : Impact visuel du processus d'insertion par la méthode AELSB : (a) image originale, (b) image stégo (c) histogramme de l'image originale, (d) histogramme de l'image stégo, (e) message secret inséré, (f) différence entre les images originale et stégo.

Evaluation de la performance des méthodes d'insertion

Dans la table 2.2, nous donnons les résultats obtenus par les deux algorithmes EAELSB et AELSB, dans le cas de trois images cover de nature différente à savoir «Lena», «Baboon » et «Peppers » de taille 512 x 512 chacune et pour trois tailles du message secret 32x32, 128x128 et 256x256.

Les résultats obtenus par les deux algorithmes montrent que, même pour une grande taille du message secret (256x256) relativement à la taille de l'image cover, à part pour l'image Baboon, les trois paramètres de mesure indiquent une bonne qualité de l'image stégo. Ces résultats corroborent celles obtenus visuellement dans les figures 2.4 et 2.5.

Cover	Taille du message secret	PSNR - EAELSB	PSNR - AELSB	IF-EAELSB	IF-AELSB	SSIM-EAELSB	SSIM-AELSB
Lena	32*32	60.1062	62.7430	0.9998	0.9999	0.9996	0.9997
	128*128	48.2215	50.2557	0.9965	0.9978	0.9940	0.9896
	256*256	42.4492	42.8823	0.9866	0.9879	0.9772	0.9740
Baboon	32*32	57.2386	54.7168	0.9995	0.9992	0.9999	1.0000
	128*128	45.1956	43.6481	0.9924	0.9892	0.9979	0.9986
	256*256	39.2869	38.8772	0.9704	0.9675	0.9915	0.9925
Peppers	32*32	60.8387	61.1090	0.9998	0.9998	0.9997	1.0000
	128*128	48.0239	48.1923	0.9966	0.9967	0.9947	0.9946
	256*256	42.4604	42.1924	0.9876	0.9868	0.9803	0.9808

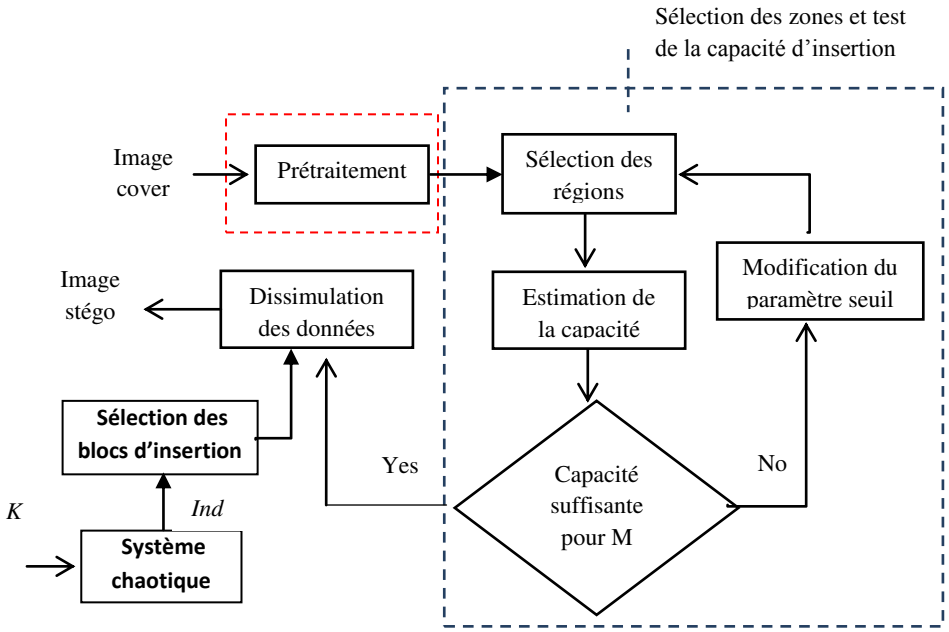
Table 2.2 : Résultats obtenus des PSNR, IF et SSIM

2.5.2 Amélioration de la Stéganographie adaptative dans les régions bords par correspondance de LSB revisitée (Enhanced Edge Adaptive Image Steganography Based on LSB Matching Revisited : EEALSBMR)

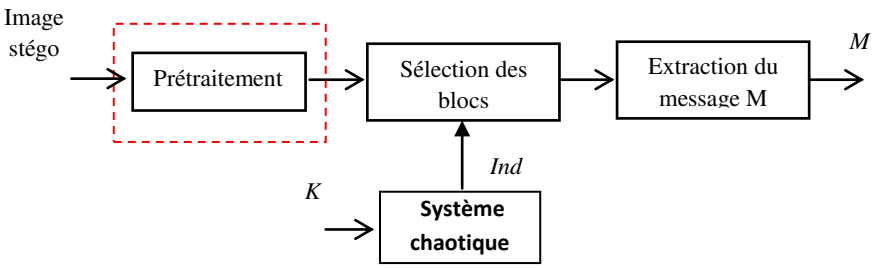
Cette méthode est plus efficace que les autres méthodes quelque soit la nature de l'image cover. Elle a un degré d'adaptabilité très grand du fait qu'elle prenne en compte la nature de l'image elle-même (son contenu) et de la taille du message secret à insérer. Le système d'insertion, basé sur un paramètre qui est un seuil adaptatif dépendant de la différence entre deux pixels consécutifs, permet de sélectionner les régions d'insertion nécessaires à l'intégration du message secret de taille donnée.
La sécurité du contenu du message dans le cas de sa détection est assurée par le même système chaotique utilisé précédemment dans le paragraphe 2.5.1.1. Sans ce système, la

sécurité de la méthode de base est faible, car elle s'appuie sur une clé secrète formée par le paramètre de rotation K1 de taille 2 bits et du paramètre de seuil codé sur 5 bits.

Dans la figure 2.6 a) et b) nous le schéma du système d'insertion et d'extraction respectivement de la méthode EEALSBMR.



(a) Système d'insertion adaptatif



(b) Système d'extraction

Figure 2.6 : Schéma adaptatif de la méthode EEALSBMR, (a) insertion, (b) extraction

2.5.2.1 Processus d'insertion

Les différentes étapes de la procédure d'insertion sont décrites ci-dessous :

1- **Prétraitement** :

Balayer séquentiellement l'image originale (cover) et la diviser en blocs de deux pixels consécutifs (p_i, p_{i+1}) non recouvrants et réarranger le résultat des blocs dans un vecteur V monodimensionnel.

Remarque : dans la méthode d'origine, cette étape consiste à diviser l'image cover en blocs de taille Bz x Bz, avec Bz= 4, 8, ou 12. Puis, à tourner chaque bloc par un angle de rotation choisi de façon pseudo aléatoire parmi l'ensemble {0°, 90°, 180°, 270°} selon une clé secrète de taille 2 bits.

2- Sélection des zones d'insertion et test de la capacité d'insertion

Pour un message secret de taille $|M|$, la sélection des zones d'insertion nécessaires se fait par la détermination de façon adaptative du seuil *T* comme suit :

Pour chaque paramètre seuil $t \in \{31, 30, \dots, 2, 1\}$, on calcule l'ensemble des paires de pixels EU(t) dont la valeur absolue de leur différence est plus grande ou égale au paramètre *t* :

$$\mathrm{EU(t)} = \{(p_i, p_{i+1}) / \ |p_i - p_{i+1}| \geq t, \forall\ (p_i, p_{i+1}) \in V\} \tag{2.19}$$

Le seuil *T* est alors donné par :

$$T = \mathrm{argmax}_t\{2 \times\ |EU(t)| \geq |M|\} \tag{2.20}$$

où $|EU(t)|$ désigne le nombre des éléments dans l'ensemble $EU(t)$

3- Insertion du message secret :

L'insertion proprement dite du message secret se fait comme suit :

3.1- Calculer l'ensemble des blocs (paires de pixels) EU(t) tel que :

$$EU(t) = \{(p_i, p_{i+1}) / \ |p_i - p_{i+1}| \geq T, \forall\ (p_i, p_{i+1}) \in V\}$$

3.2- Sélectionner de façon quasi-chaotique un bloc parmi l'ensemble $EU(t)$ précédent et insérer deux bits du message secret m_i et m_{i+1} selon un des quatre cas suivants : **(**voir annexe A) [Mielikainen, 2006])

$$\begin{aligned}
&\text{Cas 1 : } LSB(p_i) = m_i \ \&\ f(p_i, p_{i+1}) = m_{i+1} \implies (p'_i, p'_{i+1}) = (p_i, p_{i+1}) \\
&\text{Cas 2 : } LSB(p_i) = m_i \ \&\ f(p_i, p_{i+1}) \neq m_{i+1} \implies (p'_i, p'_{i+1}) = (p_i, p_{i+1} + r) \\
&\text{Cas 3 : } LSB(p_i) \neq m_i \ \&\ f(p_i - 1, p_{i+1}) = m_{i+1} \implies (p'_i, p'_{i+1}) = (p_i - 1, p_{i+1}) \\
&\text{Cas 4 : } LSB(p_i) \neq m_i \ \&\ f(p_i - 1, p_{i+1}) \neq m_{i+1} \implies (p'_i, p'_{i+1}) = (p_i + 1, p_{i+1})
\end{aligned}$$

La fonction *f* est définie par l'équation suivante :

$$f(a, b) = \mathrm{LSB}\left(\left\lfloor \frac{a}{2} \right\rfloor + b\right) \tag{2.21}$$

p'_i, p'_{i+1} sont les valeurs des pixels après l'insertion, *r* est une valeur pseudo-aléatoire de valeur {-1, 1}.

3.3- Test de validité de l'insertion des bits du message

Si $(p'_{\mathrm{i}}, p'_{\mathrm{i+1}}) \notin [0,255]$ ou $|p'_{\mathrm{i}} - p'_{\mathrm{i+1}}| < T \Rightarrow aller$ à l'étape réajustement

Sinon aller à l'étape 3.2.

Et ceci jusqu'à insertion de tous les bits du message secret.

4- Processus de réajustement

Ce processus consiste à chercher la valeur la plus proche de la valeur initiale de chacun de deux pixels, en minimisant la différence entre les nouvelles valeurs et les valeurs initiales des pixels. Les nouvelles valeurs des pixels sont données par :

$$(p_i^{"}, p_{i+1}^{"}) = argmin_{(e_1,e_2)} = \{|e_1 - p_i| + |e_2 - p_{i+1}|\} \tag{2.22}$$

avec

$$\begin{cases} e_1 = p'_i + 4k_1 \\ e_2 = p'_{i+1} + 2k_2 \end{cases} \qquad k_1, k_2 \in Z \tag{2.23}$$

$$|e_1 - e_2| \geq T, \quad 0 \leq e_1, e_2 \leq 255$$

L'interprétation de la relation (2.23) est donnée en annexe B

Apres le processus d'ajustement, le processus d'extraction reste valide :

$$LSB(p_i^{"}) = m_i \quad \& \quad f(p_i^{"}, p_{i+1}^{"}) = m_{i+1} \tag{2.24}$$

$$\text{Avec} \quad 0 \leq p_i^{"}, p_{i+1}^{"} \leq 255 \text{ , } |p_i^{"}, p_{i+1}^{"}| \geq T$$

Le paramètres T est inséré dans une région de l'image non utilisée par l'algorithme d'insertion du message secret, ce paramètre est de taille 5 bits. Il est inséré dans les 5 premiers pixels de la première ligne de l'image (excluant la valeur 255, qui est valeur destructive), et par précaution, la première ligne n'est pas utilisée par le processus d'insertion.

Exemple :

Procédure d'insertion :

Soit :

$$(p_i, p_{i+1}) = (62, 81)$$
$$(m_i, m_{i+1)}) = (1, 0)$$
$$T=19$$

Nous vérifions si le bloc en question peut être porteur de l'information à cacher :

$$|81 - 62| = 19 \geq T, \text{ oui}$$

$$\begin{cases} LSB(62) = 0 \neq m_i \\ LSB\left(\left\lfloor\frac{62-1}{2}\right\rfloor + 81\right) = 1 \neq m_{i+1} \end{cases} \Rightarrow \text{Cas 4 de la procédure d'insertion}$$

D'où :

$$(p'_i, p'_{i+1}) = (p_i + 1, p_{i+1}) = (63, 81)$$

$$d' = |81 - 63| = 18 < T \quad \Rightarrow \text{Processus d'ajustement}$$

Processus d'ajustement :

Pour

$$k_1 = 0,\ k_2 = 1, \text{ les relations ci-dessous sont vérifiées}$$

$$(p''_i, p''_{i+1}) = argmin_{(e_1,e_2)} = \{|e_1 - p_i| + |e_2 - p_{i+1}|\}$$

$$\begin{cases} e_1 = p' + 4k_1 = 63 + 4 \times 0 = 63 \\ e_2 = p' + 2k_2 = 81 + 2 \times 1 = 83 \end{cases}$$

On a :

$$\left|p''_i - p''_{i+1}\right| = |e_1 - e_2| = |63 - 83| = 20 > T$$

Et

$$\begin{cases} LSB(63) = 1 = m_i \\ LSB\left(\left\lfloor\frac{63-1}{2}\right\rfloor + 83\right) = 0 = m_{i+1} \end{cases} \text{ cas 1 de l'insertion}$$

2.5.2.2 Processus d'extraction:

Les différentes étapes du processus d'extraction sont décrites ci-dessous :

1. Extraire le paramètre T.
2. Réarranger l'image stégo en un vecteur ligne Vs et la diviser en blocs de deux pixels consécutifs (p_i, p_{i+1}).
3. Calculer l'ensemble des blocs (paires de pixels) $EU(t)$ tel que
$$EU(t) = \left\{(p''_i, p''_{i+1}) / \left|p''_i - p''_{i+1}\right| \geq T, \forall\ (p''_i, p''_{i+1}) \in Vs\right\}$$
4. Sélectionner de façon quasi-chaotique un bloc parmi l'ensemble $EU(t)$ et extraire les deux bits du message secret m_i et m_{i+1} comme suit :
$$m_i = LSB(p''_i) \quad \& \quad m_{i+1} = LSB\left(\left\lfloor\frac{p''_i}{2}\right\rfloor + p''_{i+1}\right) \tag{2.25}$$

Exemple d'extraction

D'après l'exemple d'insertion précédent nous avions
$(m_i, m_{i+1)}) = (1, 0),\ T = 19,$ et $(p''_i, p''_{i+1}) = (63, 83)$ après insertion de deux bits du message.
D'où l'extraction exacte de deux bits $(m_i, m_{i+1)})$ du message par :

$$m_i = LSB(63) = 1 \quad \& \quad m_{i+1} = LSB\left(\left\lfloor\frac{63}{2}\right\rfloor + 83\right) = 0$$

2.5.2.3 Résultats expérimentaux obtenus par les méthodes EEALSBMR et EALSBMR

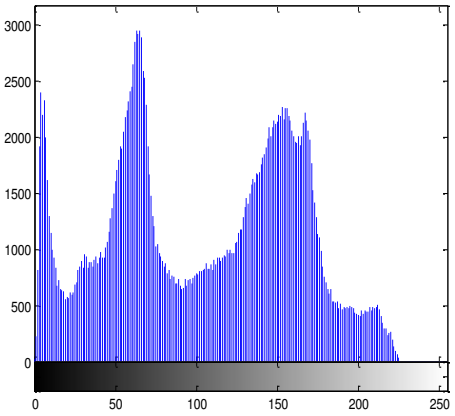
Comme pour les méthodes EAELSB et AELSB, nous donnons pour les méthodes EEALSBMR et sa version de base EALSBMR d'abord, un exemple sur l'impact du processus d'insertion d'un message secret dans une image cover. Ensuite nous présentons les performances obtenus de deux méthodes en utilisant les paramètres **PSNR**, **IF** et **SSIM.**

Impact visuel du processus d'insertion

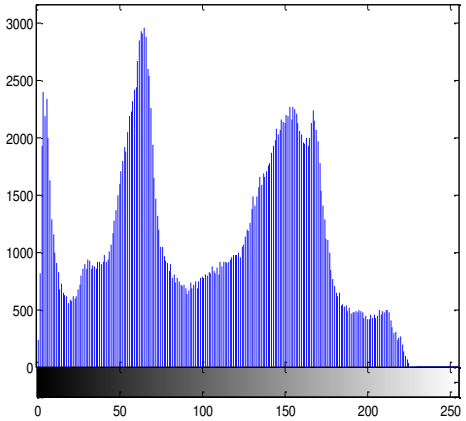
Nous présentons dans les figures 2.7 et 2.8, l'impact visuel du processus d'insertion d'un message secret (woman) de taille 64 x 64, dans l'image originale « Peppers » de taille 512 x 512 pour les méthodes EEALSBMR et EALSBMR respectivement. Nous montrons dans la figure 2.7 (méthode EEALSBMR) : a) l'image cover, b) l'image stégo, c) l'histogramme de l'image cover, d) l'histogramme de l'image stégo, e) le message secret, et f) la différence amplifiée de l'image stégo et cover. Visuellement, nous remarquons que les images cover et stégo sont similaires ainsi que leur histogrammes. Dans la figure 2.7 f) la différence amplifiée de deux images cover et stégo qui reflète la présence du message est répartie de façon quasi chaotique essentiellement sur les régions de bord. Ceci assure une meilleure qualité visuelle de l'image stégo et aussi la sécurité du contenu du message dans le cas de sa détection par un adversaire.
L'extraction du message est parfaitement réalisée.

(a) image originale

(b) image stégo

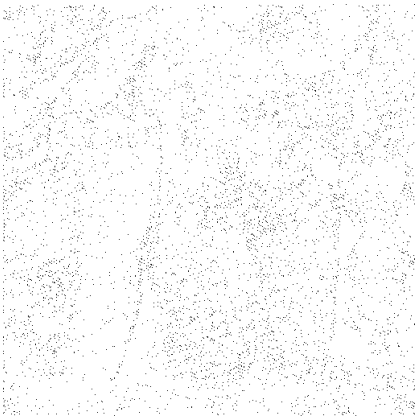


(c) histogramme de l'image originale



(d) histogramme de l'image stégo

(e) message secret



(f) image différence amplifiée

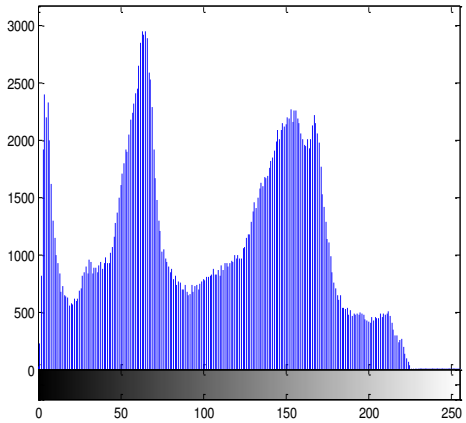
Figure 2.7 : Impact visuel du processus d'insertion par la méthode EEALSBMR : (a) image originale, (b) image stégo (c) histogramme de l'image originale, (d) histogramme de l'image stégo, (e) message secret inséré, (f) différence entre les images originale et stégo.

Nous montrons dans la figure 2.8 (méthode de base EALSBMR) : a) l'image cover, b) l'image stégo, c) l'histogramme de l'image cover, d) l'histogramme de l'image stégo, e) le message secret, et f) la différence amplifiée de l'image stégo et cover. Visuellement, nous remarquons aussi que les images cover et stégo sont similaires ainsi que leur histogrammes. Nous observons par ailleurs, sur la figure 2.8 f) que la différence amplifiée de deux images cover et stégo qui reflète la présence du message est répartie (de façon séquentielle) essentiellement sur les zones de bords.

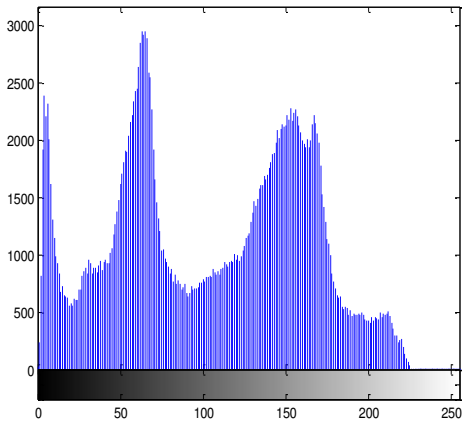
La sécurité de la méthode est assez faible car elle dépend seulement de la clé de rotation de 2 bits et du paramètre *T* de taille 5 bits.

(a) image originale

(b) image stégo

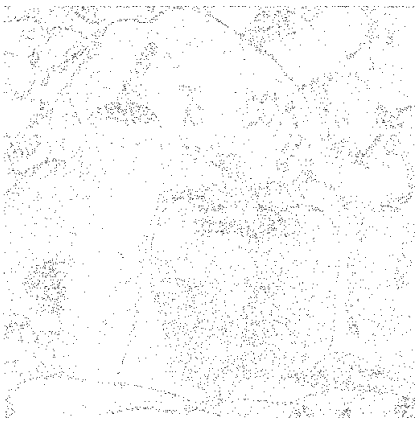


(c) histogramme de l'image originale



(d) histogramme de l'image stégo

(e) message secret



(f) image différence amplifiée

Figure 2.8 : Impact visuel du processus d'insertion par la méthode EALSBMR : (a) image originale, (b) image stégo (c) histogramme de l'image originale, (d) histogramme de l'image stégo, (e) message secret inséré, (f) différence entre les images originale et stégo.

Evaluation de la performance des méthodes d'insertion

Dans la table 2.3, nous donnons les valeurs obtenues pour les trois paramètres PSNR, IF, SSIM, pour comparer les performances des deux algorithmes EEALSBMR et EALSBMR. Nous avons ici le cas de trois images cover de natures différentes à savoir «Lena», «Baboon » et «Peppers » de taille 512 * 512 chacune, et pour trois tailles du message secret 32x32, 64x64, et 128x128.

Les résultats obtenus par les deux algorithmes montrent une très bonne qualité de l'image stégo et un gain de plus de 10 dB des PSNR comparés à ceux obtenus par les algorithmes EAELSB et AELSB (voir table 2.2).

Cover	Secret message size	PSNR-EEALSBMR	PSNR-EALSBMR	IF-EEALSBMR	IF-EALSBMR	SSIM-EEALSBMR	SSIM-EALSBMR
Lena	32*32	70.2456	70.2544	1.0000	1.0000	1.0000	1.0000
	64*64	64.3158	64.3562	0.9999	0.9999	0.9999	0.9999
	128*128	58.3985	58.4056	0.9997	0.9997	0.9982	0.9983
Baboon	32*32	70.5481	70.3754	1.0000	1.0000	1.0000	1.0000
	64*64	64.4066	64.4116	0.9999	0.9999	1.0000	1.0000
	128*128	58.4192	58.4230	0.9996	0.9996	0.9999	0.9999
Peppers	32*32	69.7208	69.6386	1.0000	1.0000	1.0000	1.0000
	64*64	63.6486	63.7568	0.9999	0.9999	0.9999	0.9999
	128*128	57.7893	57.7569	0.9996	0.9996	0.9987	0.9987

Table 2.3 : Résultats obtenus des PSNR, IF et SSIM

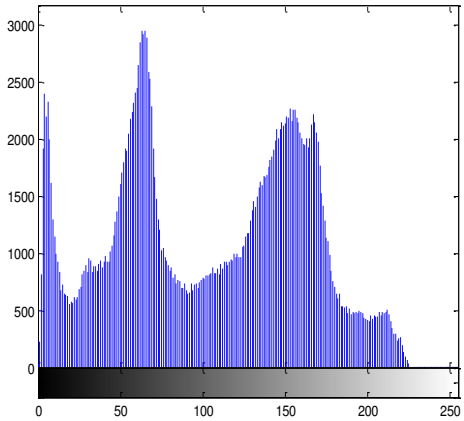
Comparaison des performances entre EEALSBMR et EAELSB

Impact visuel du processus d'insertion

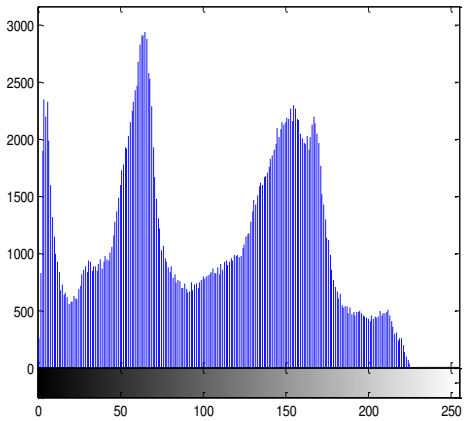
Les résultats de la figure 2.9, sont obtenus par la méthode EAELSB, en utilisant les mêmes conditions d'expérimentation que celles de la figure 2.7, à savoir : message secret (Woman) de taille 64 x 64, image originale « Peppers » de taille 512 x 512. Nous remarquons que visuellement la différence entre les deux résultats se trouve dans la figure 2.9 f) par rapport à la figure 2.7 f). Ceci montre que la méthode EEALSBMR préserve les contours, contrairement à la méthode EAELSB.

(a) image originale

(b) image stégo

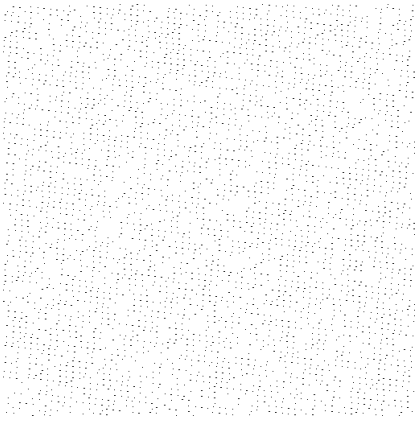


(c) histogramme de l'image originale



(d) histogramme de l'image stégo

(e) message secret



(f) image différence amplifiée

Figure 2.9 : Impact visuel du processus d'insertion par la méthode EAELSB : (a) image originale, (b) image stégo (c) histogramme de l'image originale, (d) histogramme de l'image stégo, (e) message secret inséré, (f) différence entre les images originale et stégo.

Test de performance

Mesure de PSNR, IF et SSIM

Dans la table 2.4, nous donnons les valeurs obtenues de PSNR, IF, SSIM, pour les deux algorithmes EEALSBMR et EAELSB. Les résultats obtenus montrent la supériorité de la méthode EEALSBMR par rapport à la méthode EAELSB.

Cover	Secret message size	PSNR-EEALSBMR	PSNR-EAELSB	IF-EEALSBMR	IF-EAELSB	SSIM-EEALSBMR	SSIM-EAELSB
Lena	32*32	70.2456	60.1062	1.0000	0.9998	1.0000	0.9996
	64*64	64.3158	54.9292	0.9999	0.9992	0.9999	0.9985
	128*128	58.3985	48.2215	0.9997	0.9965	0.9982	0.9940
Baboon	32*32	70.5481	57.2386	1.0000	0.9995	1.0000	0.9999
	64*64	64.4066	51.6680	0.9999	0.9983	1.0000	0.9996
	128*128	58.4192	45.1956	0.9996	0.9924	0.9999	0.9979
Peppers	32*32	69.7208	60.8387	1.0000	0.9998	1.0000	0.9997
	64*64	63.6486	54.6367	0.9999	0.9993	0.9999	0.9986
	128*128	57.7893	48.0239	0.9996	0.9966	0.9987	0.9947

Table 2.4 : Résultats obtenus des PSNR, IF et SSIM des méthodes EAEALSB et EEALSBMR

2.6 Conclusion

Dans ce chapitre nous avons étudié les méthodes de stéganographie basées LSB dans le domaine spatial pour transmettre une information sécurisée par obscurité. Les méthodes de stéganographie LSB adaptatives sont les plus répandues parmi les différentes méthodes de dissimulation spatiale de données de la littérature. En effet, ces méthodes adaptatives tentent d'insérer la totalité du message secret dans les zones de bord de l'image, qui sont plus adéquates pour cacher avec efficacité (sans dégrader la qualité visuelle de l'image résultante) que les zones lisses de l'image. Dans cette optique, nous avons étudié et analysé en détails les deux méthodes de stéganographie LSB adaptatives les plus efficaces, à savoir celle proposée par [Yang et al., 2008] et celle proposée par [Luo et al., 2010]. La première insère plus de données secrètes dans les zones de bords que les zones lisses avec une très grande capacité d'insertion, et ce tout en gardant une haute qualité de l'image stégo. La deuxième méthode tient compte de la nature de l'image cover et de la taille du message à insérer. Elle utilise en premier lieu les zones de bord les plus contrastées pour insérer les données secrètes. Si ces zones ne sont pas suffisantes, elle cherche d'autres zones de bords moins contrastées et ainsi de suite. De ce fait, elle est très adaptative et donc est la plus efficace. Cependant, l'aspect

sécurité du contenu du message, dans le cas de sa détection par un adversaire, n'est pas assurée par la première méthode (celle de Yang et al.,) et est très faible dans la deuxième méthode (celle de Luo et al.,).
Afin d'assurer et d'améliorer significativement la sécurité du contenu du message secret, nous avons adapté et implémenté les deux méthodes citées ci-dessus en ajoutant un système chaotique robuste permettant une insertion quasi-chaotique des bits du message secret.
Le système chaotique proposé consiste en un générateur de séquences chaotiques robustes fournissant les clés dynamiques d'une carte Cat 2-D chaotique.
Les performances obtenues par les deux techniques adaptatives améliorées EAELSB et EEALSBMR, visuellement et au travers des critères de qualité objective PSNR, IF et SSIM, montrent tout l'intérêt de ces techniques.

2.7 Références

[Battikh et al., 2013] Battikh, D., El Assad, S., Bakhache, B., et al. (2013). Enhancement of two spatial steganography algorithms by using a chaotic system : comparative analysis., IEEE, 8th International Conference for Internet Technology and Secured Transactions, ICITST-2013, London, UK.

[Chan et Cheng, 2001] Chan, C.-K., Cheng, L. M. (2001). Improved Hiding Data in Images by Optimal Moderately-Significant-Bit Replacement, IEE Eelectronics letters, volume 37, numéro 16, pages 1017-1018.

[Chan et Cheng, 2004] Chan, C.-K., Cheng, L. M. (2004). Hiding data in images by simple LSB substitution. Pattern Recognition, Volume 37, pages 469-474.

[Chang et al., 2002] Chang, C.-C., Lin, M.-H., Hu, Y.-C. (2002). A Fast And Secure Image Hiding Scheme Based on LSB Substitution, International Journal of Pattern Recognition and Artificial Intelligence, volume 16, numéro 4, pages 399-416.

[El Assad et Noura, 2010] El Assad, S. et Noura, H. (2010). French Patent : FR2958057 Generator of Discrete Chaotic Sequences with almost Infinite Orbits, Mars 2010, Extension PCT.

[Ker, 2004] Ker, A. (2004). Improved Detection of LSB Steganography in Grayscale Images, In Proc. 6th International Workshop, Toronto (Canada), Springer LNCS, volume 3200, pages 97–115.

[Luo et al., 2010] Luo, W., Huang, F. et Huang, J. (2010). Edge Adaptive Image Steganography Based on LSB Matching Revisited, IEEE transactions on information forensics and security, volume 5, numéro 2.

[Mielikainen, 2006] Mielikainen, J. (2006). LSB matching revisited, *IEEE Signal Process, Lett.*, volume 13, numéro 5, pages 285–287.

[Tataru et al., 2012] Tataru, R. L., Battikh, D., El Assad, S., Noura, H. et Deforges O. (2012) Enhanced Adaptive Data Hiding in Spatial LSB Domain by using Chaotic Sequences, IIH-MSP 2012, Pages 85-88.

[Thien et Lin, 2003] Thien, C. C., Lin, J. C. (2003). A Simple and High-Hiding Capacity Method for Hiding Digit-By-Digit Data in Images Based On Modulus Function, Pattern Recognition, volume 36, pages 2875-2881.

[Wang et al., 2000] Wang, R.-Z., Lin, C.-F., Lin, J.-C. (2000). Hiding Data in Images by Optimal Moderately Significant Bit Replacement, IET Electronics Letters, volume 36, numéro 25,pages 2069-2070.

[Wang et al., 2001] Wang, R.-Z., Lin, C.-F., Lin, J.-C. (2001). Image Hiding by Optimal LSB Substitution And Genetic Algorithm, Pattern Recognition, volume 34, pages 671-683.

[Wang et al., 2004] Wang, Z., Bovik, A. C., Sheikh, H. R. et Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity, IEEE transactions on Image Processing, volume 13, numéro 4, pp. 600-612.

[Wu et Tsai, 2003] Wu, D.C., Tsai, W. H. (2003). A Steganographic Method for Images by Pixel-Value Differencing, Pattern Recognition Letter, volume 24, numéro 9- 10, pages 1613–1626.

[Wu et al., 2005] Wu, H.C., Wu, N.I., Tsai, C.-S., Hwang, M.S (2005). Image Steganographic Scheme Based on Pixel-Value Differencing and LSB Replacement Methods, IEE Proceedings-Vision, Image and Signal Processing, volume 152, numéro 5, pages 611-615.

[Yang et al., 2008] Yang, C.-H., Weng, C.-Y. et Wang, S.-J. (2008). Adaptive data hiding in edge areas of images with spatial LSB domain systems, IEEE Trans. on information forensics and security, volume 3, numéro 3.

Chapitre 3

STEGANOGRAPHIE FREQUENTIELLE

3. STEGANOGRAPHIE FREQUENTIELLE

3.1 Introduction

Nous avons précédemment montré que l'enfouissement de données dans le domaine spatial présente des avantages mais aussi certains inconvénients. En effet, la complexité des calculs reste faible par rapport aux autres méthodes qui elles requièrent un changement de domaine, et la quantité de données dissimulables est conséquente. Cependant, la dissimulation dans le domaine spatiale résiste mal à la plupart des attaques d'image, ce qui limite clairement sa robustesse, donc son intérêt.

Les représentations d'une image dans le domaine fréquentiel s'appuient sur plusieurs types de transformation comme la transformée de Fourier discrète (Discret Fourier Transform DFT), la transformée en cosinus discrète (Discret Cosine Transform DCT) et les transformées en ondelettes discrètes (Discret Wavelet Transform DWT). Ces transformées sont couramment utilisées dans des systèmes de codage d'images, où les données sont alors compressées dans le domaine fréquentiel. Comme nous allons le montrer dans la suite de ce chapitre, la réalisation de méthodes d'enfouissement des données dans le domaine fréquentiels permet d'obtenir des solutions plus robustes, en comparaison avec le domaine spatial.

Dans ce chapitre, nous allons ainsi présenter les algorithmes stéganographiques fréquentielles (basées sur les différentes transformées) les plus utilisés, en expliquant leur processus d'insertion et d'extraction, et en présentant en final leurs avantages / inconvénients.. Dans la deuxième partie du chapitre, nous proposons des améliorations aux algorithmes présentés. Nos propositions cherchent essentiellement à hausser le niveau de sécurité et la capacité des méthodes stéganographiques considérées. En final, nous montrons des tests comparatifs entre nos solutions et les algorithmes originaux afin de démontrer l'efficacité des modifications suggérées. Une partie des améliorations proposées est basée sur le chaos, que ce soit pour dissimuler les informations d'une manière aléatoire, ou pour chiffrer ces informations avant de les cacher.

3.2 Etat de l'art

La nécessité d'améliorer les méthodes stéganographiques, a conduit à l'élaboration de nouveaux algorithmes opérant dans le domaine fréquentiel, dans le but de renforcer la robustesse du marquage. Les méthodes spatiales, basées par exemple simplement sur la technique LSB, présentent une faible résistance contre les attaques. A l'image des méthodes de watermarking, la recherche de techniques d'enfouissement de données plus robustes s'est portée sur des solutions opérant dans le domaine fréquentiel, avec notamment l'utilisation de transformées usuelles telles que les DFT, DCT, et DWT.

Une étude approfondie pour la comparaison des différentes techniques existantes de stéganographie, y compris celles basées DCT, DWT et DFT, est introduite dans [Paulson, 2006]. Pour ce faire, les auteurs ont développé une application avancée appelée « technologie de réseau neuronal artificiel pour la stéganographie » (ANNTS : Artificial Neural Network Technology for Steganography. Il a pu être constaté que la transformée de Fourier discrète inverse (IDFT) conduit à une erreur d'arrondi. De ce fait, la DFT est inappropriée pour les applications de stéganographie, et les techniques associées qui ont pu être développées ne seront pas présentées dans le reste du document.

Un schéma de stéganographie sans perte et réversible a été introduit dans [Chang et al., 2007]. Il utilise les blocs quantifiés des coefficients de la transformation en cosinus discrète (DCT) des images JPEG pour incorporer des données secrètes.

Dans [Danti et Preethi, 2010], on propose une nouvelle méthode stéganographique, appelée "LSB-DCT avec seuillage", basée sur l'insertion du message dans les coefficients DCT. Elle consiste à calculer d'abord la transformée DCT de l'image porteuse, ensuite l'image stégo est construite en cachant l'image secrète bit par bit, dans le bit de poids faible LSB des coefficients DCT qui sont inférieurs à un certain seuil.

Une autre méthode d'insertion, appelée "ajustement DC", basée sur la transformation DCT est proposée dans [Saejung et al., 2013]. Elle consiste à faire ajuster le coefficient DC en fonction du bit à insérer du message secret.

La Transformée en ondelettes (WT), qui convertit l'image du domaine spatial en domaine de fréquence, est proposée pour construire des algorithmes stéganographiques en raison d'un certain nombre d'avantages et surtout que cette transformée divise et sépare d'une manière claire la haute fréquence et la basse fréquence des informations. Une méthode réversible de dissimulation de données basée sur la transformation en ondelettes de Haar discrète (HDWT) a été proposé en 2009 [Chan et al., 2009]. Dans cette méthode, l'image porteuse est transformée du domaine spatial en domaine fréquentiel sur la base HDWT et ensuite les coefficients de haute fréquence sont utilisés pour cacher les données secrètes. Cette méthode fournit une bonne capacité et une bonne qualité de l'image stégo.

De même, une autre méthode basée sur la transformation DWT a été proposée dans [Saejung et al., 2013]. Cette méthode, appelée "DWT-LSB avec seuillage", consiste à faire dissimuler les bits du message secret dans les coefficients DWT qui sont inférieurs à un seuil donné.

Dans [Boora et Gambhir, 2013], on propose la méthode appelée "DWT Alpha-fusion" qui fusionne les bandes de fréquences de la décomposition en ondelettes DWT des images originales et secrètes.

A côté des méthodes basées DCT et DWT, on a proposé [Marvel et al., 1999] une technique stéganographique basée sur l'étalement de spectre, qui module le message secret en élargissant sa bande passante avant de le dissimuler dans l'image porteuse. Dans [Bhattacharya et al., 2012] on a développé une technique mixte basée DWT et étalement de spectre, où deux messages secrets sont insérés dans deux bandes V1 et D1 de la

décomposition en ondelettes niveau 1, en utilisant un générateur de nombres pseudo aléatoires. Le problème majeur des méthodes basées sur l'étalement de spectre est sa faible capacité.

3.3 Algorithmes stéganographiques existants

Les techniques fréquentielles DCT et DWT sont protégées contre certaines attaques, surtout lorsque le message caché est petit. Ceci peut être expliqué par la façon dont elles changent les coefficients dans le domaine de transformation, en réduisant la distorsion de l'image au minimum. La stéganographie basée DCT est beaucoup plus puissante et moins vulnérable aux attaques statistiques que les algorithmes opérant dans le domaine spatial [Khan]. La stéganographie basée DWT possède une bonne robustesse et une meilleure capacité d'enfouissement que celle basée DCT et celle opérant dans le domaine spatial. D'un autre côté, les méthodes basées sur l'étalement de spectre permettent de cacher le message secret d'une manière moins perceptible, sous forme d'un bruit blanc gaussien dont la bande passante est élargie. La principale limitation de ces dernières est leur faible capacité d'enfouissement.

3.3.1 Algorithmes à base de DCT

Il existe dans la littérature beaucoup d'approches stéganographiques basées sur la transformée DCT. Nous présentons parmi ces algorithmes les deux méthodes les plus courantes

1- Ajustement DC

Cet algorithme [Saejung et al., 2013] traite seulement les images de format JPEG où l'image RGB est convertie en YCbCr. La dissimulation des données n'est réalisée que dans le plan Y qui contient l'essentiel de l'information. Cet algorithme possède comme entrée le message secret et l'image porteuse, et comme sortie l'image dite stégo. Les étapes du traitement sont les suivantes :

1. Convertir l'image secrète en binaire.
2. Convertir l'image porteuse depuis l'espace couleur RGB en celui YCbCr, le plan Y étant le seul utilisé par la suite pour cacher l'image secrète.
3. Diviser l'image porteuse en blocs 8×8 pixels.
4. Appliquer la transformée DCT sur chaque bloc.
5. Lire le message secret bit à bit, chaque bit étant caché de la manière suivante :
    - La position *p(1,1)* de chaque bloc est utilisée pour calculer la valeur *dc* donnée par *dc=(round(p(1,1)/16))*.
    - Si la valeur de *dc* est paire et le bit secret à cacher est *0,* ou si la valeur de *dc* est impaire et le bit secret est *1*, alors on ne fait aucun changement sur le bloc considéré et on passe au bloc suivant.

Si la valeur de *dc* est paire et le bit secret est *1,* ou si la valeur de *dc* est impaire et le bit secret est *0*, on calcule la valeur de *dc1* donnée par *dc1=(p(1,1)/16),* et on compare *dc et dc1*. Si *dc>dc1*, on calcule la différence *df=(dc-dc1)* et le nombre N_p défini par $N_p=(0.5-df)$. La Table 3.1 donne le nombre de pixels qui doivent être modifiés dans l'image porteuse selon la valeur de N_p et selon l'intervalle dans lequel il se trouve. Ces pixels sont modifiés de telle sorte qu'on retranche un de chaque pixel, c.à.d. on modifie le bit LSB de chacun de ces pixels.

Np	Quantité
0.000-0.063	8
0.064-0.125	16
0.126-0.188	24
0.189-0.250	32
0.251-0.313	40
0.314-0.375	48
0.376-0.437	56
0.438-0.500	64

Table 3.1 : Table donnant le nombre de pixels à modifier

6. Répéter les étapes 4 et 5 jusqu'à ce que la valeur de *dc* soit paire et le bit secret valle 0, ou que la valeur de dc soit impaire et le bit secret valle 1.
7. Appliquer la transformation Inverse DCT (IDCT) pour obtenir l'image stégo.
8. Sauvegarder le nouveau plan Y de l'image contenant le message secret puis reconvertir l'image en RGB.

Cette méthode possède deux avantages : sa simplicité et sa robustesse. Par contre, sa capacité d'enfouissement est médiocre puisqu'un seul bit peut être intégré dans chaque bloc de 64 pixels. C'est ce point faible que la méthode suivante cherche à améliorer.

2- LSB-DCT avec seuillage

Cette méthode appelée "LSB-DCT avec seuillage" est l'une des techniques la plus utilisée en stéganographie. Elle consiste à cacher une image secrète dans une image porteuse en se basant sur les valeurs des coefficients DCT. Pour ne pas avoir une distorsion visuelle, l'intégration des informations secrètes est évitée pour les coefficients DCT de valeur égale à 0 [Danti et Preethi, 2010]. La dissimulation des bits secrets se fait selon les étapes suivantes :

1. Sélectionner une image porteuse dont la taille doit être au moins le double de celle de l'image secrète.

2. Diviser l'image porteuse en des blocs 8×8 pixels et appliquer la transformation DCT sur chacun de ces blocs.

3. Déterminer les coefficients DCT dont la valeur est inférieure à un certain seuil *t* (pour de meilleures illustrations, le seuil $t = 0$ est souvent utilisé, et dans les coefficients négatifs qu'on dissimule le message secret). C'est dans ces coefficients transformés que l'information sera cachée.

4. Construire un vecteur contenant les positions de ces coefficients potentiels, qui sera utilisé comme clé par le récepteur afin de retrouver les bits cachés.

5. Convertir l'image secrète en vecteur binaire 1D.

6. Remplacer un par un, les bits du vecteur binaire 1D avec les LSB des coefficients DCT successifs des pixels potentiels.

7. Appliquer la transformée DCT inverse (IDCT) pour obtenir l'image stégo.

3.3.2 Algorithmes à base de DWT

L'utilisation des ondelettes dans la stéganographie est récente. Le plus souvent, les techniques proposées conservent le principe du stockage dans les bits les moins significatifs (LSB) mais appliqué aux coefficients de la transformée DWT. Rappelons que l'ondelette divise l'image en des régions de basse (A), de moyenne (H et V), et de haute fréquence (D), ceci pouvant être itéré (plusieurs niveaux). La figure 3.1 montre l'espace de représentation fréquentiel pour une transformée d'ondelette DWT dyadique appliquée sur une image de niveau 1, 2, et 3. La question principale qui survient lorsque l'on souhaite développer une méthode de stéganographie basée DWT est de déterminer quelles bandes fréquentielles (A, H, V, ou D) et quels niveaux (1, 2, ou 3) seront les mieux adaptés. Les différentes techniques existantes vont ainsi principalement se distinguer sur ce point en fonction des applications et objectifs visés : certaines vont cacher les bits secrets dans les basses fréquences, d'autres dans les moyennes, d'autres dans les hautes. Selon [Pratibha et Shanti, 2013], une méthode stéganographique cachant l'image secrète dans la bande à basse fréquence (A) possèdera une bonne robustesse puisque la plus grande partie de l'énergie de l'image est stockée dans cette bande. Malgré tout, il existe un risque de dégradation de l'image, si la dissimulation des données n'est pas bien géré.
Il existe dans la littérature plusieurs algorithmes stéganographiques qui se basent sur DWT. Parmi ces algorithmes, nous présentons les deux le plus utilisés baptisés " DWT-LSB avec seuillage", et "DWT Alpha-Fusion".

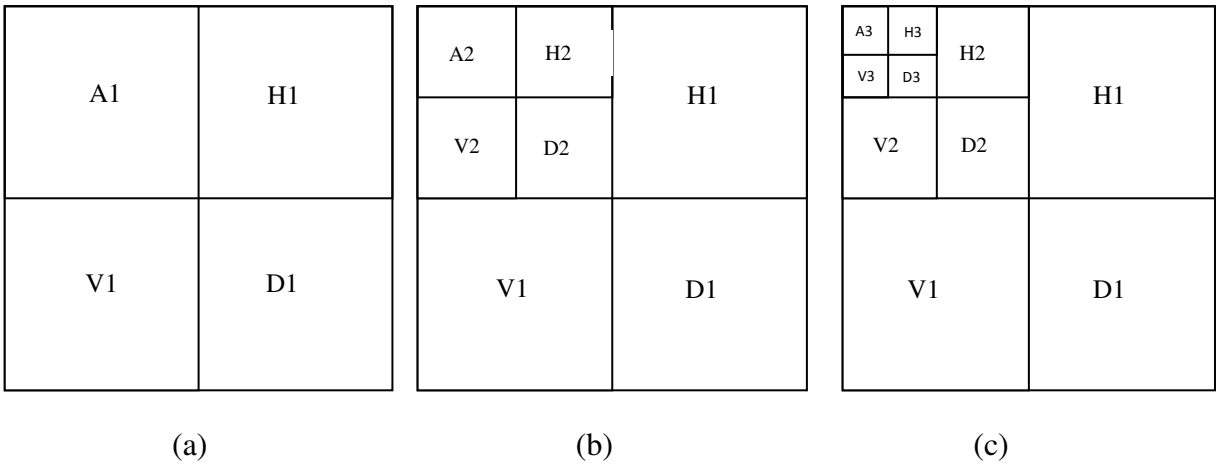


Figure 3.1 : Décomposition fréquentielle d'une image selon la transformée DWT (a) niveau 1 (b) niveau 2 (c) niveau 3

1- DWT-LSB avec seuillage

Cet algorithme introduit par [Saejung et al., 2013] possède de manière naturelle en entrée l'image porteuse et l'image secrète, et en sortie l'image stégo. Il cache un seul bit du message secret dans le LSB d'un coefficient DWT en se basant sur un certain seuil. Ses étapes sont les suivantes :

1. Convertir le message secret ou l'image secrète en vecteur binaire.
2. Appliquer la transformation en ondelette Haar-DWT et trouver la valeur moyenne des valeurs absolues des coefficients DWT de l'image transformée (appellée *moyenne)*.
3. Fixer le seuil *T* de l'image tel que : $T = 0.9 \times moyenne$.
4. Trouver les positions de dissimulation ayant les valeurs absolues des coefficients inférieures au seuil *T*, et insérer un bit du message secret dans chacun de ces coefficients
5. Appliquer la transformée inverse (IDWT) pour obtenir l'image stégo.

Le problème majeur de cet algorithme est sa faible capacité d'insertion. Cette limitation est en partie levée dans la méthode « DWT Alpha- Fusion » présentée ci-dessous.

2- DWT Alpha- Fusion

Cet algorithme fut introduit dans [Boora et Gambhir, 2013]. Le principe est de crypter l'image secrète via la carte d'Arnold, puis d'effectuer une transformée DWT à la fois sur l'image cryptée et sur l'image porteuse. Les deux transformées sont ensuite fusionnées (selon le modèle donné par l'eq. 3.1) . En final, on applique une transformée inverse (IDWT) pour obtenir l'image stégo. Dans cet algorithme, Les deux images, secrète et porteuse sont de même taille. Le cryptage et le fusionnement donnent à ce type de stéganographie un bon niveau de sécurité.

La fusion de l'image secrète avec l'image originale se fait par simple somme pondérée des deux bandes de basse fréquences A de la décomposition DWT de niveau 2 de ces deux images. Ceci est donné par l'équation suivante :

$$X' = \alpha X + \beta M \quad (3.1)$$
$$\alpha + \beta = 1 \ ; \ \alpha \gg \beta$$

X' est la matrice de la bande de fréquence considérée de la transformée DWT de l'image stégo.
X est la matrice de la bande de fréquence considérée de la transformée DWT de l'image initiale.
M est la matrice de la bande fréquentielle considérée de la transformée DWT de l'image secrète cryptée.

α et β sont les coefficients d'intégration. Ce sont deux réels compris dans l'intervalle [0, 1], et choisis de telle sorte que β soit très petit devant α pour limiter la distorsion introduite dans l'image.

Les différentes étapes de l'insertion sont (voir figure 3.2) :

1. Effectuer la transformation en ondelettes discrète DWT au niveau 2 de l'image support.
2. Appliquer la transformation Arnold sur l'image secrète basée sur une certaine clé (ou une condition initiale), dans le but de la crypter.
3. Effectuer la transformation DWT au niveau 2 de l'image secrète cryptée.
4. Extraire les bandes à basse fréquence du deuxième niveau (A2) de l'image originale et de l'image cryptée.
5. Appliquer le processus de fusion de deux bandes extraits.
6. Effectuer la transformation inverse IDWT (inverse de la transformation en ondelettes discrète au deuxième niveau) pour former l'image stégo.

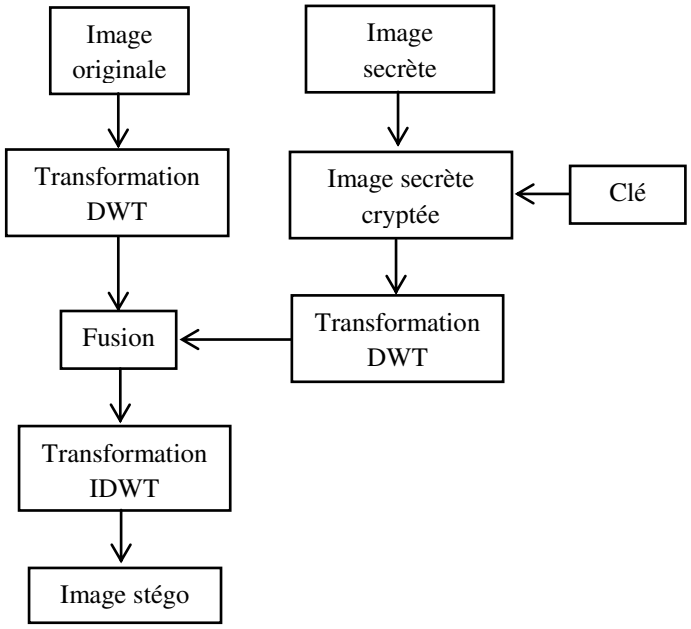


Figure 3.2 : processus d'insertion par DWT Alpha-Fusion

Il est vrai que l'image secrète a la même taille (MxN) que l'image originale, mais en fait la fusion se fait pour les deux bandes de basses fréquences A de la décomposition DWT au

niveau 2 de l'image originale et de l'image secrète. Ceci signifie qu'on ne cache pas vraiment toute l'image secrète mais seulement une partie significative de taille M/4 x N/4 pixels.

3.3.3 Algorithme stéganographique par étalement de spectre SSIS

Les transmissions numériques à spectre étalé sont devenues très populaires ces dernières années. Pour les applications militaires, l'intérêt majeur est la propriété de grande discrétion de telles communications (détection difficile). Dans le domaine civil, celles-ci sont exploitées pour l'accès simultané de plusieurs émetteurs à la même bande de fréquence (Code Division Multiple Access - CDMA). De plus, les transmissions à spectre étalé présentent généralement une bonne robustesse sur des canaux même fortement perturbés (trajets multiples).
Le principe de l'étalement de spectre [Cox et al., 1995], [Meel, 1999] consiste à repartir l'énergie du signal à émettre sur une bande de fréquence plus large que celle réellement nécessaire à la transmission du signal utile. Les deux principales techniques de modulation par étalement de spectre sont la séquence directe [Karim et Sarraf, 2002] (Direct Sequence Spread Spectrum) et le saut de fréquence [Karim et Sarraf, 2002] (Frequency Hopping Spread Spectrum). Dans le cas de la séquence directe, l'énergie du signal est repartie sur toute la bande de fréquence disponible, alors que pour le saut de fréquence, la bande de fréquence disponible est divisée en un grand nombre de sous-canaux. La fréquence porteuse se déplace alors d'un sous-canal à l'autre par des sauts discrets pseudo-aléatoires. Dans les deux cas, l'objectif est le même : étaler la puissance du signal transmis sur une large bande de fréquence afin de noyer cette puissance dans le bruit ambiant ou dans les autres communications. C'est pourquoi, ces techniques ont été initialement développées dans le domaine militaire : elles permettent de cacher des transmissions dans le bruit.

L'étalement de la bande fréquentielle confère différent avantages :

- La densité spectrale du signal étant étalée, elle devient plus faible pour une fréquence donnée et perturbe donc moins les systèmes de communications à bande étroite.

- L'information est codée et répond donc à une règle de transmission. Tout signal ne répondant pas à cette règle ne perturbera pas le récepteur. Cela implique une certaine immunité aux interférences provoquées par d'autres signaux.

- Les effets des multi-trajets sont limités car les ondes retardées peuvent être assimilées à des interférences

- Grâce au codage de l'information, un utilisateur ne connaissant pas le code ne peut pas l'intercepter.

- L'attribution d'un code par utilisateur permet l'adressage et donc la constitution d'un réseau.

L'étalement de spectre a été utilisé récemment dans le domaine de la stéganographie et l'algorithme dit SSIS (spread spectrum image steganography) est le plus connu de ce genre. Il consiste à dissimuler un message binaire, en étalant son spectre par une modulation, puis la transformant en image de même taille que l'image porteuse avant sa dissimulation, dans une image porteuse. En effet, le message secret est crypté au début par un système de chiffrement

symétrique classique [Marvel et al., 1998], ensuite le message crypté est codé par un code correcteur d'erreur à faible débit, avant d'être modulé en le multipliant par une séquence pseudo-aléatoire. Le signal résultant est enfin entrelacé avant d'être dissimulé dans l'image cover. Le problème majeur de cette méthode SSIS est la faible capacité puisque l'étalement de spectre ne permet pas de moduler et d'étaler une grande quantité d'informations binaires.

A- Processus d'insertion

Les différentes étapes du processus de dissimulation sont les suivantes (voir figure 3.3) :

1. Chiffrer le message secret (ou l'image secrète), en se basant sur une clé K1.
2. Coder le message chiffré par un code correcteur d'erreurs (ECC), dont le but est d'aider à corriger les erreurs de transmission d'une information.
3. Moduler [Härtung et Girod, 1997] la séquence chiffrée en la multipliant par une séquence pseudo aléatoire générée en utilisant une clé secrète K2.
4. Transformation 1D-2D.
5. Fusionner le signal résultant avec l'image originale.

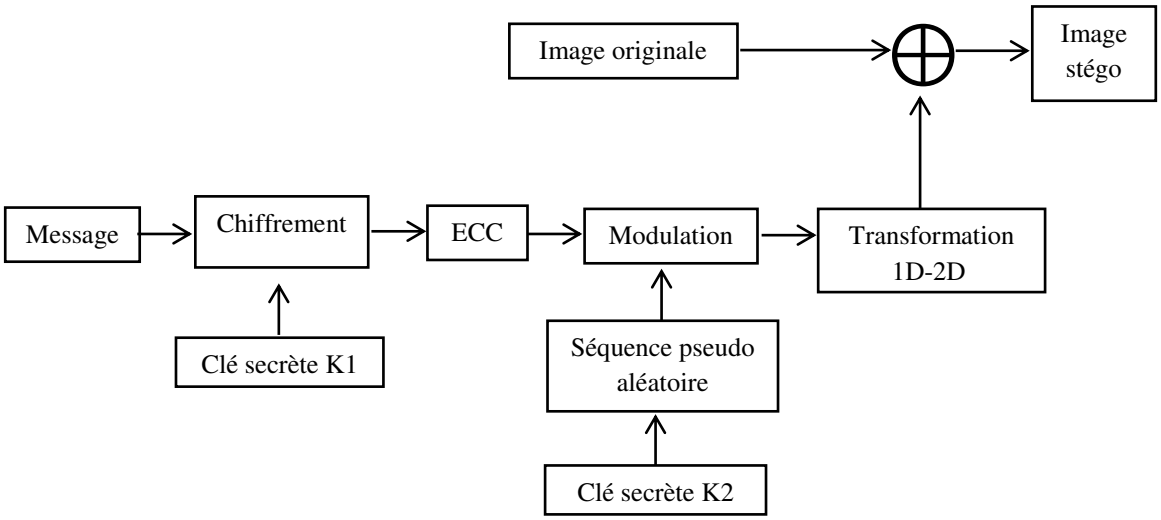


Figure 3.3 : processus de dissimulation par l'algorithme SSIS

B- Processus d'extraction

1. Extraire de l'image stégo, par fusion inverse, le message entrelacé
2. Effectuer le désentrelacement et démoduler le message en utilisant la séquence pseudo aléatoire générée en se basant sur la clé secrète K2.
3. Décoder le message (par le code ECC).
4. Déchiffrer le message en utilisant la clé secrète K1.

3.4 Proposition d'amélioration de l'algorithme LSB-DCT avec seuillage

Nous avons présenté deux algorithmes à base de la transformation DCT. Le premier était de faible capacité, et le deuxième LSB-DCT comptait sur certain seuil pour cacher les informations secrètes dans les coefficients DCT. C'est le deuxième algorithme que nous allons modifier et améliorer. Au lieu de choisir les coefficients DCT (dans lesquels nous insérons nos bits confidentiels) en fonction d'un certain seuil, nous proposons d'utiliser un générateur chaotique pour les sélectionner. Ceci améliore d'un côté, la capacité de l'algorithme puisque nous pouvons cacher dans la plupart des coefficients, sauf les coefficients de la fréquence zéro et ceux ayant des fréquences proches de zéro, d'une manière pseudo-aléatoire, et d'un autre côté, la sécurité de l'algorithme puisque le choix des lieux où on cache nos bits se fait pseudo-aléatoirement. Il reste à choisir un bon générateur chaotique. Plus ce dernier est aléatoire plus la méthode stéganographique est sécurisée.

3.4.1 Générateur chaotique utilisé

Dans le chapitre 1, nous avons présenté plusieurs cartes chaotiques : carte logistique, carte PWLCM et carte skewtent, et nous avons vu les nombreux avantages qui caractérisent la PWLCM. Ces avantages ont poussé les chercheurs à s'appuyer sur cette carte chaotique pour construire et proposer un nouveau générateur appelé RFCA (Robust and Fast Chaotic encryption Algorithm) [Bakhache et al., 2013] formé de deux cartes PWLCMs perturbées et combinées ensemble par un XOR. En effet, chacune de deux opérations (la perturbation d'un générateur chaotique et la combinaison de plusieurs cartes chaotiques) prolonge chacune la longueur du cycle de l'orbite chaotique, améliore le comportement chaotique de la carte et augmente la sécurité. Certaines analyses pratiques et théoriques effectuées, montrent que l'utilisation d'un couple de cartes chaotiques suffit et peut fournir une très bonne performance et l'utilisation de plus de deux cartes n'apporte pas trop d'avantages. En plus, le calcul parallèle dans un matériel rend l'exécution d'un couple de cartes chaotiques, implémentée pratiquement, très rapide.

La figure 3.4 représente le schéma du RFCA où les sorties des deux PWLCM *i* perturbés, *i* = 1 ou 2, (R1 et R2) sont combinées par une opération OU exclusif afin de produire un nouveau flux chaotique R avec une meilleure aléatoirité. Partageant les mêmes conditions initiales et les mêmes paramètres du système (la clé secrète), le récepteur peut générer les mêmes séquences aléatoires R1 et R2, et calculer R.

Afin de vérifier les performances du RFCA, des simulations (sur Matlab) ont été réalisées, et les figures 3.5, 3.6 et 3.7 montrent les résultats obtenus. La figure 3.5 présente l'attracteur de la carte chaotique considérée, où la dispersion des points montre une bonne couverture de l'espace géométrique, et par suite un haut degré d'aléatoirité. La figure 3.6 présente l'auto-corrélation et l'inter-corrélation et la figure 3.7 présente le spectre de la séquence chaotique R. Ces deux figures prouvent clairement que la série chaotique a la forme d'un bruit.

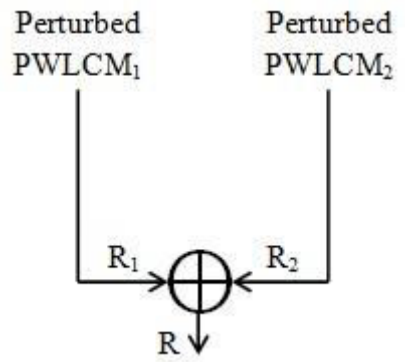


Figure 3.4 : Générateur chaotique RFCA utilisé

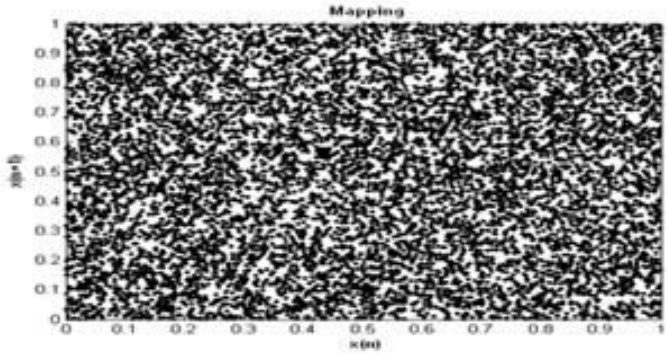


Figure 3.5 : Attracteur du générateur RFCA

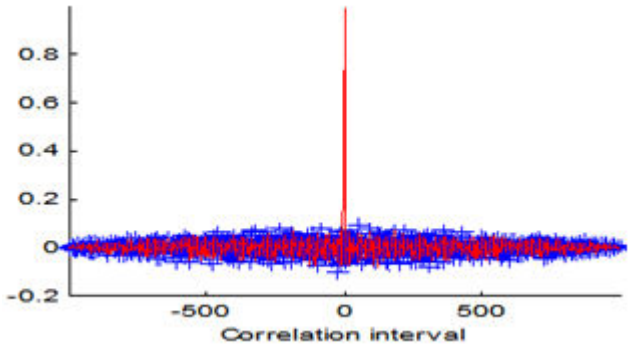


Figure 3.6 : Auto-corrélation et inter-corrélation du RFCA

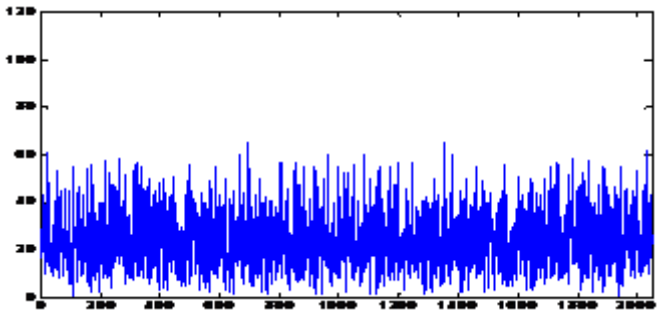


Figure 3.7 : Spectre du générateur chaotique proposé

3.4.2 Méthode proposée

Notre objectif est de modifier et d'améliorer la méthode "LSB-DCT avec seuillage". Cette dernière cache les bits secrets dans les coefficients DCT dont la valeur est inférieure à un certain seuil. Ceci se fait d'une manière systématique et successive coefficient après coefficient. Nous proposons d'utiliser le générateur chaotique RFCA pour sélectionner le coefficient dans lequel les bits secrets sont cachés. Ceci permet une insertion dans les coefficients de manière pseudo-aléatoire. Ainsi, le changement se fait toujours dans les LSB des coefficients DCT de l'image porteuse en introduisent de faibles modifications non décelées par l'œil humain.

L'utilisation du générateur chaotique RFCA se fait de la manière suivante : l'image secrète de dimension $p \times q$ est convertie en un vecteur binaire. Chaque pixel de cette image est représenté sur 8 bits. Supposons que s soit la longueur du vecteur binaire, alors $s = p \times q \times 8$. Chaque bit de ce vecteur devra remplacer le LSB d'un coefficient DCT transformé. Le nombre de positions nécessaires pour cacher toute l'image secrète est donc égal à la valeur s. Le coefficient DCT de l'image originale dans lequel nous cacherons un de nos bits secrets, est choisi pseudo-aléatoirement par un couple de coordonnées (X, Y) où X représente le numéro de la ligne et Y représente le numéro de la colonne de l'image cover. Par simplicité, nous considérons seulement les images porteuses carrées dont la hauteur et la largeur sont égales (souvent la dimension 512×512 pixels est considérée). Les positions X et Y sont représentés chacun sur k bits (de la séquence chaotique générée), alors la position d'un coefficient est définie par $2 \times k$ bits. Par conséquent il est nécessaire de générer du RFCA un flux binaire de longueur l bits, où $l = 2 \times k \times s$ pour intégrer toute l'image secrète.

A- Processus d'insertion

Pour pouvoir insérer nos informations secrètes selon notre algorithme proposé, il est nécessaire de posséder comme paramètres d'entrée l'image originale, l'image secrète et les conditions initiales du générateur chaotique RFCA. On aura en sortie l'image stégo. Les étapes du processus d'intégration sont illustrées par la figure 3.8, et sont les suivantes:

1. Lire l'image porteuse et l'image secrète.
2. Convertir l'image secrète en un vecteur binaire, où chaque pixel est représenté sur 8 bits.
3. Diviser l'image originale en blocs 8×8 pixels, et appliquer la transformation DCT sur chaque bloc.
4. Utiliser le générateur chaotique RFCA pour fournir une longue séquence de $l = 2 \times k \times s$ bits.
5. Extraire de la séquence chaotique générée les coordonnées (X, Y), qui représentent les positions des coefficients DCT transformés dans lesquelles nous insérons les bits du vecteur binaire formé à partir de l'image secrète. Les coordonnées (X, Y) sont pris de telle façon que k premiers bits représentent X, et les k bits qui suivent représentent Y et ainsi de suite.
6. Remplacer le LSB de ces coefficients choisis par les bits des données secrètes.
7. Appliquer la fonction inverse IDCT pour obtenir l'image stégo.

Cet algorithme a deux avantages majeurs : il offre une grande capacité et un haut niveau de sécurité. En effet, contrairement à la méthode "LSB-DCT avec seuillage" qui ne cache que

dans une partie des coefficients, la méthode proposée peut insérer dans la plupart des coefficients de l'image cover. Ce qui améliore la capacité, et comme le choix des pixels se fait pseudo-aléatoirement et non pas d'une manière systématique ceci hausse la sécurité de la méthode stéganographique. Le partage des conditions initiales du générateur chaotique et la taille de l'image secrète avec le récepteur permet à ce dernier de générer la même séquence aléatoire et par conséquent de récupérer le message secret en appliquant le processus d'extraction suivant.

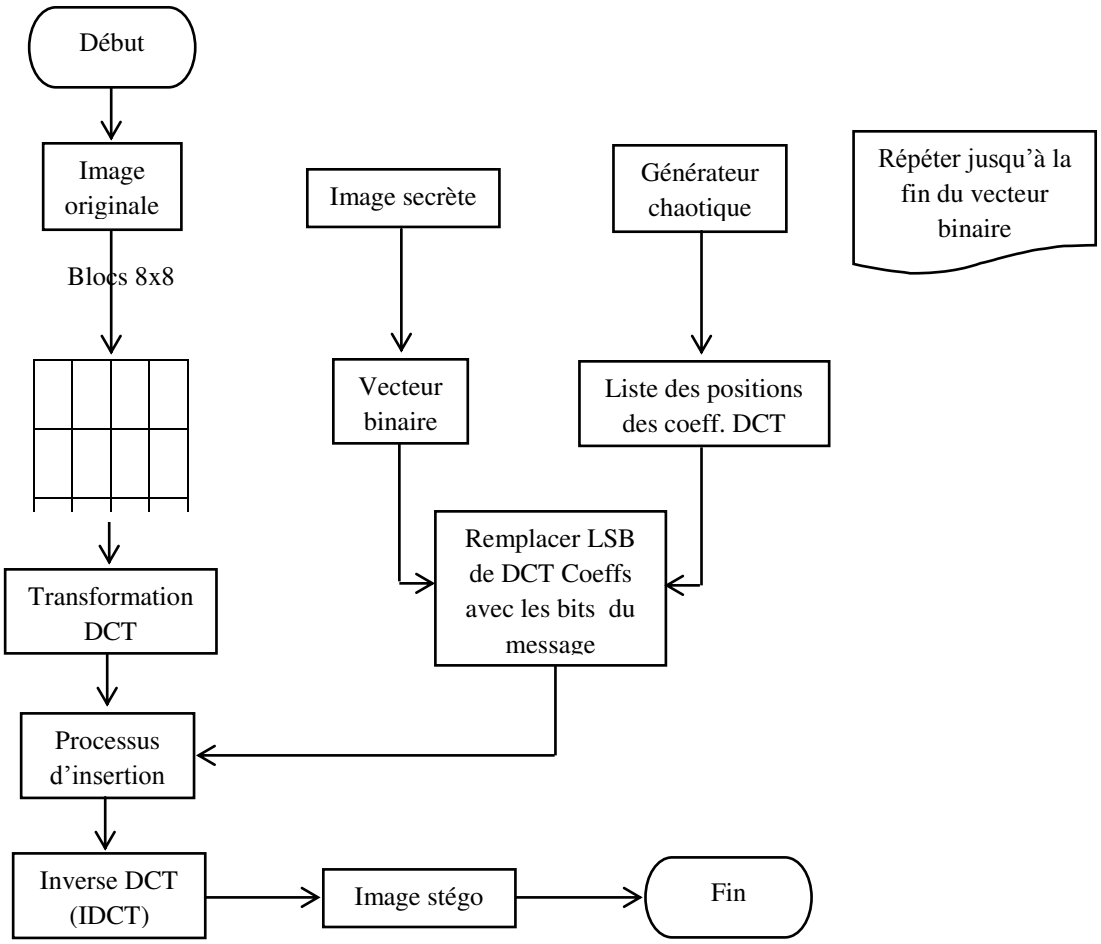


Figure 3.8 : Processus d'insertion Schéma bloc de la méthode proposée

B-Processus d'extraction

Ce processus prend comme entrée l'image stégo, la taille de l'image secrète et les paramètres initiales du générateur chaotique RFCA pour donner en sortie l'image secrète. Les étapes d'extraction sont les suivantes :

1. Lire l'image stégo.
2. Diviser l'image stégo en blocs 8×8 pixels, et appliquer la transformation DCT sur chaque bloc.

3. Générer la séquence aléatoire du RFCA en utilisant les mêmes conditions initiales que l'émetteur et extraire les positions des coefficients DCT qui contiennent les données secrètes.
4. Extraire le LSB des coefficients sélectionnés.
5. Construire l'image secrète.

3.4.3 Test et analyse de la méthode proposée

Pour tester la performance de l'algorithme proposé, les tests suivants sont réalisés :

A- Test d'imperceptibilité visuelle.

B- Test de qualité.

C- Test de capacité d'insertion.

Ces tests sont menés de manière comparative entre notre solution et l'algorithme initial "LSB-DCT avec seuillage".

A- Test d'imperceptibilité visuelle

L'insertion de l'image secrète doit respecter la contrainte forte d'imperceptibilité, qui signifie que la distorsion introduite dans l'image originale soit visuellement imperceptible. Cette contrainte exige que la qualité de l'image stégo soit le plus proche possible de celle de l'image d'origine.

Pour effectuer cette série de tests, nous allons considérer plusieurs images secrètes de tailles différentes à insérer dans plusieurs images porteuses, dans le but d'estimer la distorsion introduite. La diversité des images de test permet de pouvoir donner une conclusion plausible sur les résultats.

La figure 3.9 présente les images secrètes de tailles 40x40, 57x57 et 81x81 pixels, qui représentent respectivement 5%, 10%, et 20% de la taille de l'image porteuse de 512x512 pixels. La figure 3.10 montre les images originales porteuses (« Baboon », « Lena » et « Cameraman ») avec lesquelles nous avons effectué nos tests. Les trois images secrètes considérées sont insérées dans chacune des images porteuses, selon les deux algorithmes "LSB-DCT avec seuillage" et celui proposé. Les deux figures 3.11 et 3.12 montrent les images stégos obtenues en dissimulant les images secrètes dans les images porteuses considérées et avec les deux algorithmes étudiés.

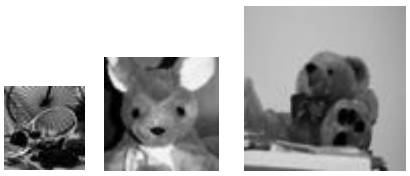


Figure 3.9 : Images secrètes

Figure 3.10 : Images originales

(a)

(b)

(c)

Figure 3.11 : Images stégos obtenues par « LSB-DCT avec seuillage » pour un taux d'insertion (a) de 5% (b) de 10% (c) de 20%

(a)

(b)

(c)

Figure 3.12 : Images stégos obtenues par la méthode proposée pour un taux d'insertion (a) de 5% (b) de 10% (c) de 20%

Les résultats obtenus montrent que les deux techniques assurent une bonne qualité d'imperceptibilité pour une taille moyenne d'image insérée

Test de qualité

La qualité de l'image stégo par rapport à l'image originale peut être évaluée à l'aide d'outils mathématiques tels que le PSNR (« Peak Signal to Noise Ratio ») ou SSIM (« Structural Similarity »).

La métrique classique la plus utilisée pour estimer objectivement la distorsion entre deux images est le PSNR qui est considérée comme une mesure indicative fiable. La table 3.2 montre la variation de PSNR en fonction de taux d'insertion (en testant les trois images secrètes avec différents taux) pour les différentes images standards, et pour les deux algorithmes "LSB-DCT avec seuillage" et celui proposé. Cette table montre que la valeur de PSNR, et ce pour les deux algorithmes considérés, diminue lorsque la taille de l'image secrète (ou le taux d'insertion) augmente, ce qui est parfaitement logique. Elle montre aussi que pour tous les taux d'insertion, la valeur de PSNR de la méthode proposée est supérieure à celle de la technique "LSB-DCT avec seuillage". Comme exemple, en prenant l'image « Baboon » comme porteuse et pour un taux d'insertion de 5%, la méthode proposée donne un PSNR égal à 70.6975 dB, tandis que le PSNR de "LSB-DCT avec seuillage" est égal à 65.3985 dB.

Taux d'insertion	Images originales	PSNR (db)	
		LSB-DCT avec seuillage	Méthode proposée
5%	Baboon	65.3985	70.6975
	Lena	65.0017	71.5038
	Cameraman	65.9383	73.8625
10%	Baboon	62.3859	64.6663
	Lena	62.1113	64.8786
	Cameraman	63.1865	65.9976
20%	Baboon	59.3306	59.6497
	Lena	59.0324	59.6906
	Cameraman	60.0079	61.1395

Table 3.2 : PSNR pour les taux d'insertion à 5,10 et 20%

On peut remarquer que l'écart de PSNR entre les deux méthodes est important lorsque le taux d'insertion est faible, et notre méthode proposée est bien meilleure que la méthode « LSB-DCT avec seuillage ».

L'autre métrique couramment utilisée est la SSIM qui tente de déterminer la similarité structurelle des images, avec pour but d'être plus proche de la perception visuelle que le PSNR. La SSIM possède des valeurs comprises entre 0 et 1 pour mesurer la corrélation avec l'image source, la valeur 1 indiquant une corrélation parfaite ou encore une similitude parfaite. La table 3.3 montre les valeurs de SSIM pour les deux algorithmes "LSB-DCT avec seuillage" et celui proposé, en considérant les différentes images porteuses et les différents taux d'insertion. D'après ces résultats, l'image Baboon présente le meilleur support parmi les trois images porteuses considérées puisqu'elle possède la plus grande valeur de SSIM (pour tous les taux d'insertion et pour les deux algorithmes). Pour les autres images support, de manière logique, la valeur de SSIM diminue lorsque la taille de l'image secrète augmente. On voit clairement aussi que pour ces images porteuses (« Lena » et « Cameraman »), la SSIM de l'algorithme proposé est plus élevée que celui de l'algorithme "LSB-DCT avec seuillage".

Taux d'insertion	Images originales	SSIM	
		LSB-DCT avec seuillage	Méthode proposée
5%	Baboon	0.9999	0.9999
	Lena	0.9994	0.9999

	Cameraman	0.9985	0.9999
10%	Baboon	0.9999	0.9999
	Lena	0.9992	0.9998
	Cameraman	0.9975	0.9976
20%	Baboon	0.9999	0.9999
	Lena	0.9987	0.9992
	Cameraman	0.9963	0.9974

Table 3.3 : SSIM pour taux d'insertion égal 5,10 et 20%

Les résultats ainsi obtenus pour le PSNR et la SSIM montrent clairement que la méthode proposée produit en final une meilleure qualité d'image.

B- Test de Capacité

La capacité désigne la quantité d'informations secrètes qu'il est possible de cacher. Ceci dépend bien sûr de l'image support. La technique "LSB-DCT avec seuillage" insère le message secret dans les LSB des coefficients DCT dont la valeur est inférieure à un seuil t . La capacité de cette technique est donc facilement mesurée (égale au nombre de coefficients DCT dont la valeur est inférieure à t). Notre méthode proposée peut insérer le message secret dans la plupart des coefficients DCT de l'image transformée. Ainsi, la capacité de l'algorithme proposé est plus grande que celle avec seuillage. La table 3.4 montre la capacité pour les différentes images porteuses (512×512 pixels) en utilisant la technique "LSB-DCT avec seuillage" et la méthode proposée. Il est très clair et très logique que la capacité de notre algorithme proposé est largement supérieure à celle du "LSB-DCT avec seuillage".

Images originales	**Capacité en bits**	
	LSB-DCT avec seuillage	***Méthode proposée***
Baboon	123.244	225.280
Lena	112.473	225.280
Cameraman	66.176	225.280

Table 3.4 : Capacité mesurée sur des images standards

3.5 Propositions d'amélioration de l'algorithme DWT Alpha-Fusion

Nous avons déjà présenté deux algorithmes basés sur la transformation en ondelette DWT. Le premier, "DWT-LSB seuillage", était de faible capacité. Le second, "DWT-LSB Alpha-Fusion", présentait de meilleures capacité et robustesse, en permettant de dissimuler une ou plusieurs images dont la taille totale ne doit pas dépasser M/4 x N/4. Dans cette partie, nous allons proposer trois améliorations pour ce deuxième algorithme : une première amélioration pour augmenter sa capacité, la deuxième et la troisième amélioration visant à hausser sa sécurité en utilisant un système chaotique.

Avant d'expliquer ces trois améliorations, nous présentons le système chaotique utilisé. Ce dernier [El Assad et al., 2008] [El Assad, 2012] aura pour rôle de crypter l'image simplement en permutant aléatoirement les pixels de l'image. Le système chaotique choisi (SC) pour faire cette tâche est présenté dans la figure 3.13. Il a comme entrée la matrice (M_l, et M_c) des positions initiales des pixels de l'image avant permutation, et comme sortie la matrice (M'_l, et M'_c) des nouvelles positions de ces pixels permutés. Ce système utilise la carte cat 2D et le générateur chaotique PWLCM perturbé, présentés tous les deux précédemment.

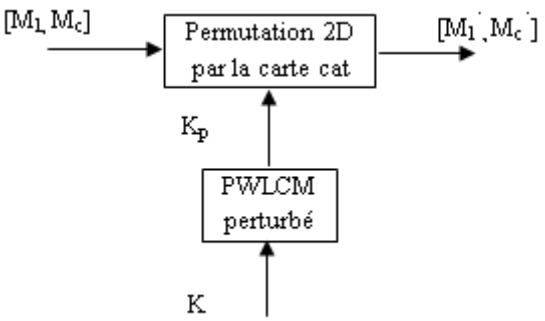


Figure 3.13 : Schéma du système chaotique

3.5.1 Première amélioration de " DWT Alpha-Fusion "

Au lieu de dissimuler nos informations secrètes dans la bande basses fréquences A de la transformée DWT au niveau 2 de l'image originale, nous proposons de l'insérer dans les bandes des fréquences H et V, au niveau 1 de la transformée DWT de l'image originale. Dans ce cas la capacité d'insertion est alors 2x (M/2 x N/2). Par ailleurs, la capacité du système peut être augmentée en comprimant le message secret avant son insertion. A ce sujet, dans cette méthode, on effectue la transformée DWT au niveau 1 de l'image secrète afin de n"utiliser que la bande basses fréquences A, qui contient la plus grande partie de l'énergie de l'image secrète. C'est donc cette partie uniquement de l'image secrète qui sera cachée et transmise.

Il s'agit maintenant de détailler le processus d'insertion et d'extraction dans le cas de la dissimulation de deux images secrètes au lieu d'une seule. Nous ne considérons pas ici le cryptage effectué par la transformation Arnold lors de l'algorithme standard, celui-ci sera en effet remplacé ultérieurement par le système chaotique présenté précédemment.

A- Processus d'insertion

Les étapes du processus d'insertion de deux images secrètes I1 et I2 de même taille dans l'image porteuse (MxN) sont les suivantes (voir figure 3.14) :

1- Appliquer la transformée DWT de Haar à l'image support.
2- Appliquer la transformée DWT de Haar aux images secrètes.
3- Fusionner la bande à basses fréquences As1 de la première image secrète I1 avec la bande à moyennes fréquences Hc de l'image originale, et la bande à basses fréquences As2 de la deuxième image secrète I2 avec la bande à moyennes fréquences Vc de l'image originale, ceci de la manière suivante :

$$H_s = \alpha_1 H_c + \beta_1 A_{s1}$$
$$V_s = \alpha_2 V_c + \beta_2 A_{s2} \qquad (3.2)$$

$$\alpha_1 \gg \beta_1 \; ; \alpha_2 \gg \beta_2$$
$$0 < \alpha < 1 \text{ et } \alpha + \beta = 1$$

4- Appliquer la transformée DWT inverse pour obtenir l'image stégo.

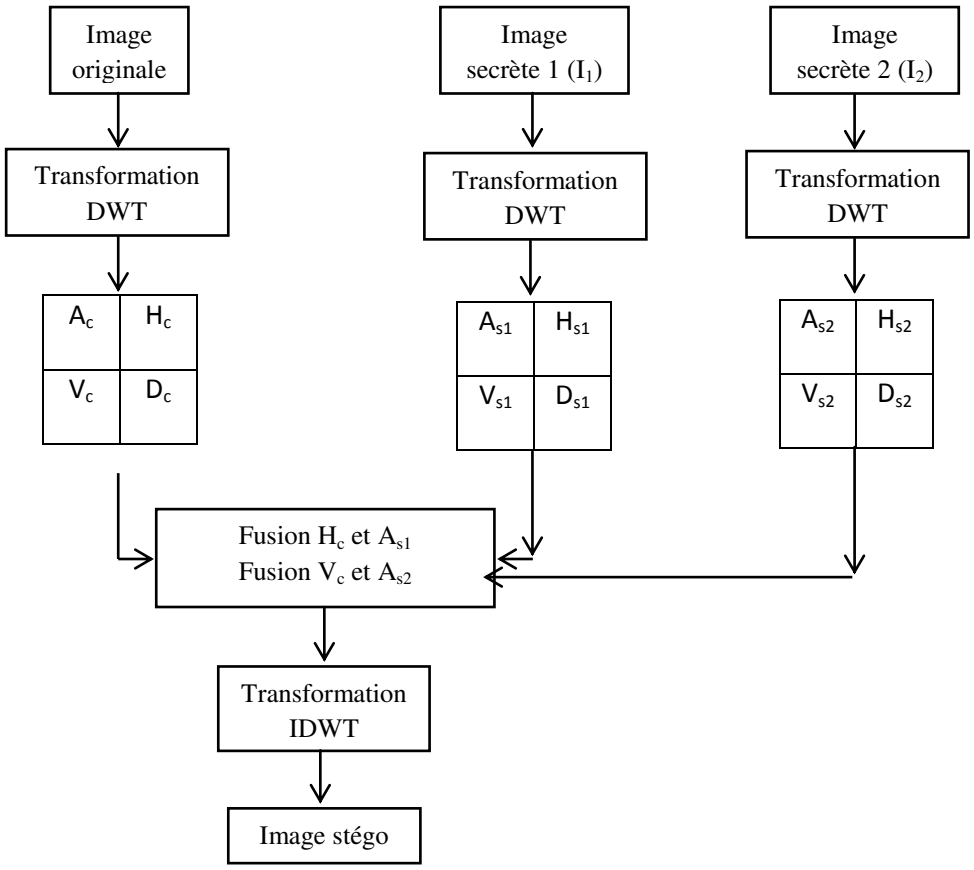


Figure 3.14 : Processus d'insertion de la première proposition

B- Algorithme d'extraction

L'extraction des images secrètes s'effectue de la manière suivante :

1- Appliquer la transformée en ondelettes DWT de Haar sur l'image stégo.
2- Extraire les images secrètes en appliquant la fusion inverse.
3- Appliquer la transformée inverse IDWT sur chacune des deux images secrètes.

C- Tests

Pour analyser notre première proposition, nous devons tester l'imperceptibilité de l'algorithme, et mesurer les deux métriques PSNR et SSIM, en les comparant avec ceux de l'algorithme standard « DWT-LSB Alpha-Fusion ».

Pour cela, nous avons dissimulé l'image « Boat » et l'image « Cameraman » dans l'image « Lena ». Elles sont toutes de même taille (512x512 pixels). La figure 3.15 (a) montre l'image originale, et les figures 15 (b) et 15 (c) montrent les deux images secrètes. L'image stégo obtenue en appliquant notre première proposition, est donnée par la figure 3.15 (d). Elle montre que l'algorithme possède une bonne qualité d'imperceptibilité.

Figure 3.15 : (a) image originale, (b) image 1 extraite, (c) image 2 extraite , (d) image stégo par la méthode proposée

Comme on ne transmet qu'une partie de l'image secrète, il faudrait donc estimer la qualité de l'image secrète reconstruite par rapport à l'image secrète d'origine. Pour cet objectif, nous calculons les paramètres PSNR et SSIM pour $\beta = 0.01$, et les résultats obtenus sont donnés dans la table suivante :

Image extraite	PSNR (dB)	SSIM
Boat 512x512	34.5897	0.9983
cameraman 512x512	38.6007	0.9912

Table 3.5 : les valeurs PSNR et SSIM de l'image récupérée par la méthode proposée ($\beta = 0.01$)

Les valeurs obtenues de PSNR et SSIM montrent que la qualité de l'image extraite est bonne. Il est logique que nous ne pouvons pas récupérer l'image secrète avec une excellente qualité puisque nous ne dissimulons que la bande basses fréquences A de la décomposition DWT au niveau 2 de l'image secrète (de taille 128x128).

Pour que l'on puisse effectuer une comparaison entre notre méthode et la méthode standard DWT-alpha fusion, en se basant sur les deux critères PSNR et SSIM, nous simplifions ici le processus en ne dissimulant qu'une seule image, « Boat », dans l'image « Lena ». L'image « Boat » considérée est supposée de taille égale à la capacité de la méthode standard, autrement dit égale à celle de l'image porteuse « Lena » (512x512 pixels). Plusieurs tests ont été effectués pour différentes valeurs de α et β, et la table 3.6 donne quelques-uns de ces résultats. Il est clair que la méthode proposée donne des meilleurs résultats que la méthode initiale DWT-alpha fusion. On remarque aussi que les valeurs des deux mesures, PSNR et SSIM, diminuent légèrement lorsque β augmente (légèrement aussi). Ceci s'explique par le fait que β règle l'influence de l'image secrète dans l'opération de fusion des deux images porteuse et secrète : plus la valeur de β est grande, plus le poids et l'influence de l'image dissimulée est grande, et plus les mesures PSNR et SSIM sont faibles.

Image originale	Image secrète	Méthode DWT-alpha fusion		Méthode proposée		β
		PSNR (dB)	SSIM	PSNR (dB)	SSIM	
Lena 512x512	Boat 512x512	33.9311	0.8392	50.7351	0.9999	0.008
Lena 512x512	Boat 512x512	33.9306	0.8374	47.8404	0.9999	0.01
Lena 512x512	Boat 512x512	33.9279	0.8364	46.4483	0.9999	0.011

Table 3.6 : Résultats obtenus des paramètres PSNR et SSIM pour différentes valeurs de β pour les méthodes « DWT Alpha-fusion » de référence et proposée

3.5.2 Deuxième amélioration de " DWT Alpha-Fusion "

La deuxième amélioration proposée consiste à augmenter le niveau de sécurité en cryptant la bande à basses fréquences A de l'image secrète (et toutes les bandes basses fréquences A_{s1} et A_{s2} des images secrètes s'il y en a plusieurs à cacher). Pour ceci, nous proposons d'utiliser le système chaotique SC (présenté dans la figure 3.13). Ce chiffrement chaotique remplace le

cryptage par transformation Arnold qui se fait dans l'algorithme de référence. Les différentes étapes de l'algorithme proposé et de ses deux processus d'insertion et d'extraction restent les mêmes comme dans l'algorithme précédent lors de la première amélioration. La figure 3.16 montre le schéma bloc qui résume notre deuxième proposition.

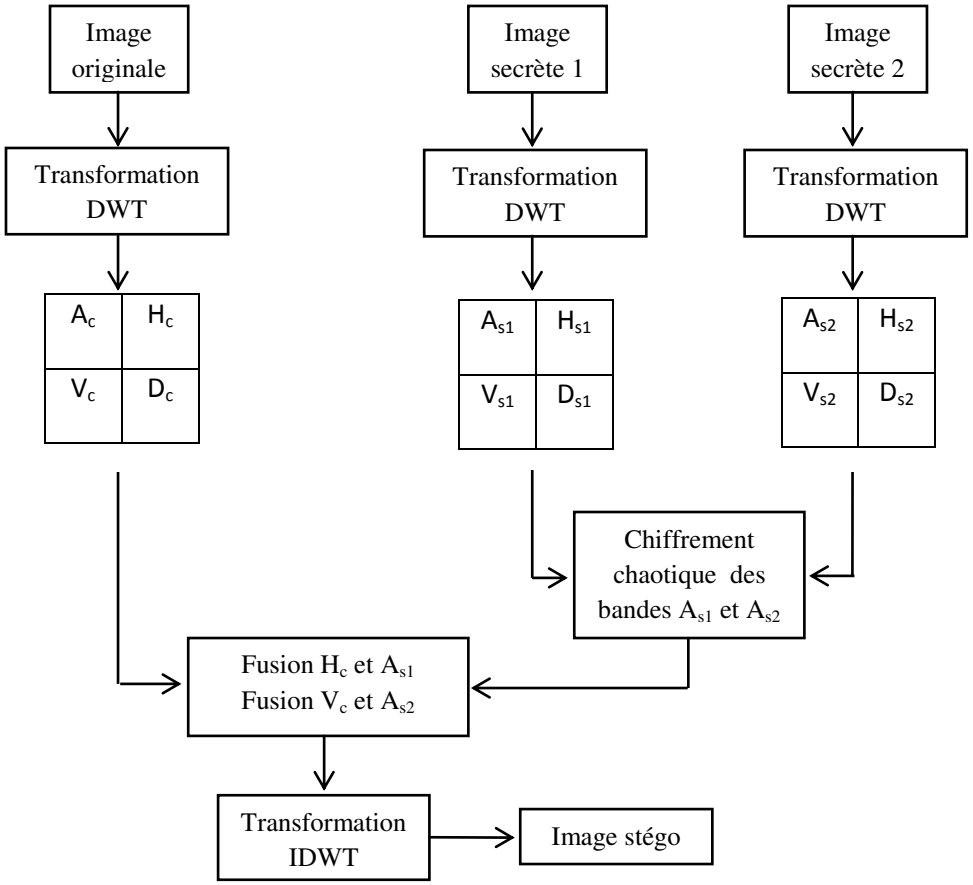
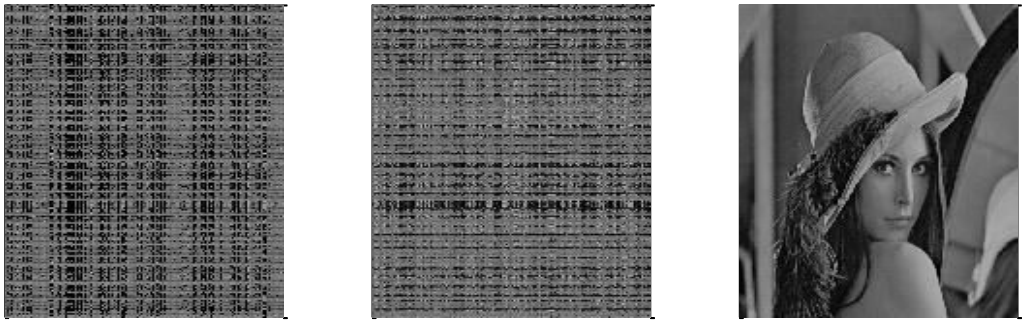


Figure 3.16 : Schéma bloc de la fusion du message chiffré pour la méthode « DWT Alpha-Fusion » proposée

Rappelons que le récepteur doit partager avec l'émetteur la même clé secrète du SC pour qu'il puisse décrypter les images secrètes ou plutôt leurs bandes à basses fréquences *A*.
Les figures 3.17 (a) et (b) montrent les bandes *A* cryptées des deux images secrètes « Boat » et « Cameraman ». La figure 3.17 (c) montre l'image stégo après le processus de fusion suivant notre deuxième proposition. Les figures 3.18 (a), et 3.18 (b) montrent respectivement les deux images secrètes reconstruites après l'application du processus d'extraction comprenant également le décryptage. Le résultat obtenu montre que l'algorithme proposé possède une bonne imperceptibilité.

Pour les deux mesures de PSNR et SSIM, pour notre solution et celle de référence, la table 3.7 donne les valeurs obtenues pour l'image secrète « Boat » de même taille (512x512) que l'image porteuse « Lena », et pour différentes valeurs de β. Notre algorithme donne des valeurs qui sont nettement meilleures que celles de l'algorithme de référence. Par exemple, pour β=0.01 le PSNR est égal à 47.83 db pour notre algorithme, alors qu'il n'est que de 33.93 db pour l'algorithme de référence. Dans les mêmes conditions de test, la SSIM est très proche de 1 pour notre algorithme, alors qu'elle n'est que de 0.84 pour l'algorithme de référence. Ceci tend à montrer la supériorité de notre solution.



(a) (b) (c)

Figure 3.17 : (a) message1 crypté, (b) message2 crypté, (c) image stégo (PSNR=47.8317, SSIM=0.9999, β=0,01)

(a) (b)

Figure 3.18 : (a) message1 récupéré, (b) message2 récupéré

Image originale	Image secrète	Méthode DWT-alpha fusion		Méthode proposée		β
		PSNR (dB)	SSIM	PSNR (dB)	SSIM	
Lena 512x512	Boat 512x512	33.9311	0.8392	50.7290	0.9999	0.008
Lena 512x512	Boat 512x512	33.9306	0.8374	47.8317	0.9999	0.01
Lena 512x512	Boat 512x512	33.9279	0.8364	46.4586	0.9999	0.011

Table 3.7 : Résultats obtenus pour les mesures PSNR et SSIM pour les méthodes « DWT Alpha-Fusion » de référence et proposée

Calculons les paramètres PSNR et SSIM (pour $\beta = 0.01$), afin d'estimer la qualité de l'image secrète reconstruite par rapport à l'image secrète d'origine. Les résultats obtenus sont donnés dans la table suivante :

Image extraite	PSNR (dB)	SSIM
Boat 512x512	34.5908	0.9983
cameraman 512x512	38.6006	0.9912

Table 3.8 : les valeurs PSNR et SSIM de l'image récupérée par la méthode proposée ($\beta = 0.01$))

Les valeurs obtenues montrent que l'image secrète reconstruite est de bonne qualité.

3.5.3 Troisième amélioration de " DWT Alpha-Fusion "

Dans ce schéma d'insertion, nous réalisons les mêmes étapes que pour l'algorithme précédent, mais nous ajoutons avant la phase de fusion, une permutation chaotique des bandes Hc et Vc selon le même système chaotique déjà utilisé pour le chiffrement des images secrètes, mais utilisant une autre clé secrète. Ceci a pour effet d'augmenter significativement le niveau de sécurité de la transformation proposée. La nouvelle structure se caractérise par une grande capacité et un très bon niveau de sécurité.
La figure 3.19 montre le schéma bloc de cette proposition qui inclut les trois améliorations proposées pour renforcer la méthode standard " DWT-LSB Alpha-Fusion ".

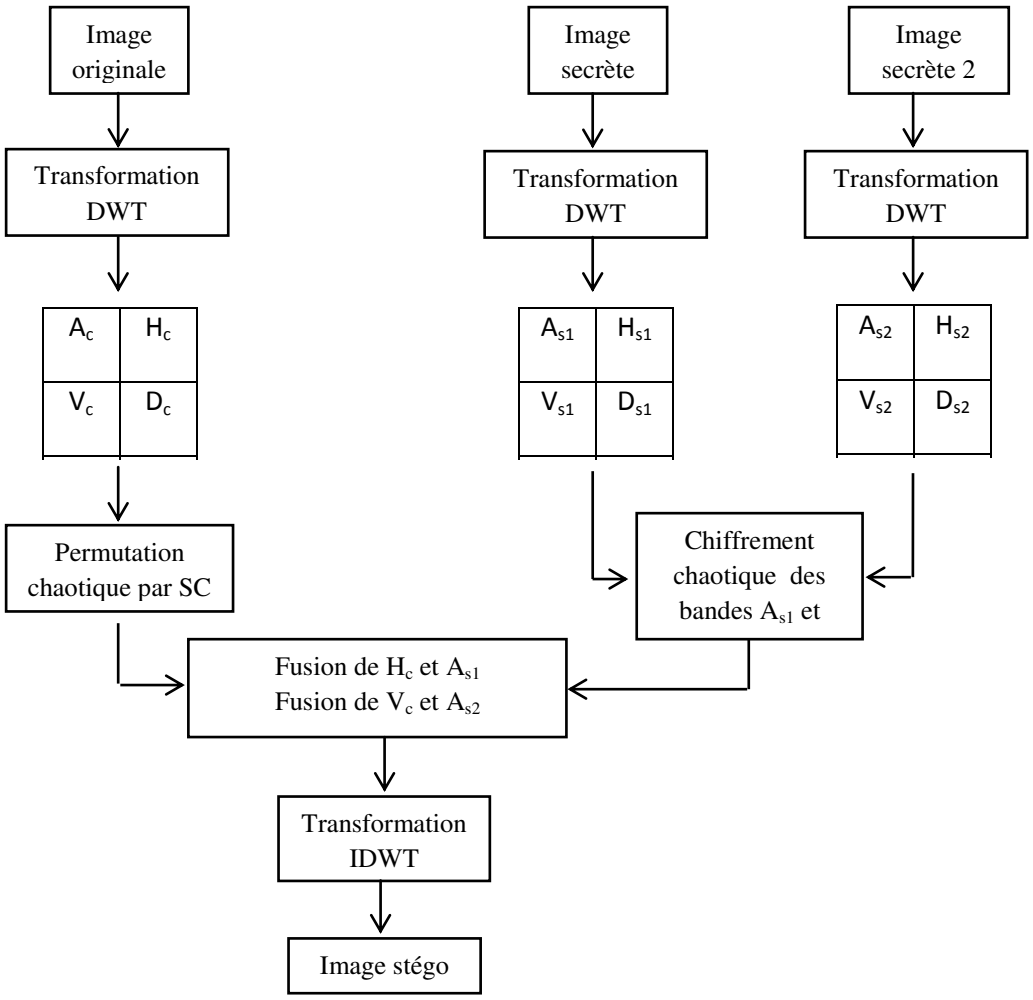


Figure 3.19 : Schéma bloc complet de la méthode proposée d'insertion incluant toutes les améliorations

Pour que le système puisse fonctionner, le récepteur doit partager avec l'émetteur, les tailles des images originales et secrètes (qui peuvent être de taille différente), ainsi que les deux clés secrètes du système chaotique SC.
Pour tester cette proposition, nous avons dissimulé les deux images secrètes « Boat » et « Cameraman » dans l'image « Lena ». Toutes ces images considérées ont la même taille

512x512 pixels. La figure 3.20 (a) montre l'image stégo, et les figures 3.20 (b) et (c) présentent les deux images secrètes reconstruites après l'application du processus d'extraction. L'image stégo montre que l'algorithme proposé possède une bonne imperceptibilité.
Afin de comparer notre algorithme avec l'algorithme standard, en se basant toujours sur les critères PSNR et SSIM, nous dissimulons l'image « Boat » dans l'image « Lena » de même taille 512x512 pixels. La table 3.9 donne les valeurs obtenues pour différentes valeurs de β montrant clairement la supériorité de notre algorithme.

La table 3.10 montre les valeurs des paramètres PSNR et SSIM, calculés pour l'image reconstruite.

(a) (b) (c)

Figure 3.20 : (a) image stégo (PSNR=47.8359, SSIM=0.9999, β=0,01) , (b) image 1 extraite, (c) image 2 extraite

Image originale	Image secrète	Méthode DWT-alpha fusion		Méthode proposée		β
		PSNR (dB)	SSIM	PSNR (dB)	SSIM	
Lena 512x512	Boat 512x512	33.9311	0.8392	50.7280	0.9999	0.008
Lena 512x512	Boat 512x512	33.9306	0.8374	47.8359	0.9999	0.01
Lena 512x512	Boat 512x512	33.9279	0.8364	46.4652	0.9999	0.011

Table 3.9 : Résultats des paramètres PSNR et SSIM pour l'algorithme « DWT Alpha-Fusion » de référence et proposé

Image extraite	PSNR (dB)	SSIM
Boat 512x512	34.5908	0.9983
cameraman 512x512	38.6002	0.9913

Table 3.10 : les valeurs PSNR et SSIM de l'image récupérée par la méthode proposée ($\beta = 0.01$)

3.6 Amélioration de la méthode SSIS

Dans ce paragraphe, nous proposons plusieurs améliorations pour la méthode stéganographique, basée sur l'étalement de spectre SSIS présentée précédemment. La

première consiste à utiliser le chaos pour chiffrer l'image secrète et pour générer la séquence pseudo-aléatoire qui sert à moduler l'image chiffrée. La seconde consiste à utiliser la transformation en ondelettes DWT pour transformer l'image originale afin d'améliorer la robustesse de l'algorithme, et dissimuler les informations par fusion dans les bandes à moyennes fréquences H et/ou V. Ces améliorations permettent d'augmenter significativement le niveau de sécurité des données cachées.
La figure 3.21 montre le schéma bloc du processus de l'intégration du message étalé dans la bande à basses fréquences A de la décomposition en ondelettes de l'image originale.

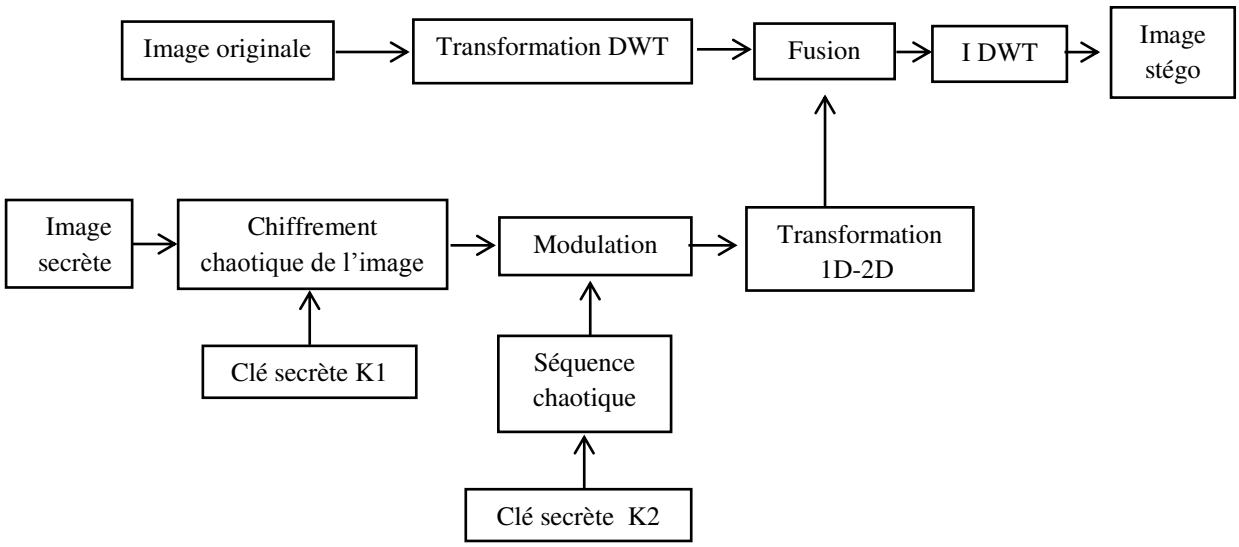


Figure 3.21 : Processus de dissimulation proposé pour améliorer l'algorithme SSIS

A- Processus d'insertion

Le processus de dissimulation de l'image secrète est réalisé par les étapes suivantes (voir figure 3.21) :

1. Appliquer la transformée de Haar-DWT à l'image originale.
2. Chiffrer l'image secrète par un système chaotique SC utilisant une clé secrète K1.
3. Moduler l'image chiffrée en utilisant une séquence chaotique générée à partir du générateur RFCA, décrit précédemment et utilisant une clé secrète K2.
4. Transformer le signal modulé en image.
5. Effectuer la fusion comme précédemment en utilisant la bande H ou V, ou les deux bandes H et V ensemble de l'image porteuse.
6. Appliquer la transformée DWT inverse pour obtenir l'image stégo.

B- Algorithme d'extraction

L'extraction de l'image cachée s'effectue suivant les étapes suivantes :

1. Appliquer la transformée DWT de l'image stégo.

2. Appliquer le processus inverse de la fusion pour séparer la bande H (ou V, ou H et V) de l'image originale de la partie insérée.
3. Désentrelacer, puis démoduler l'image chiffrée en utilisant la séquence pseudo-aléatoire générée par le RFCA (en utilisant la clé secrète K2).
4. Déchiffrer l'image résultante via le système SC utilisant la clé K1.

C- tests

Pour tester l'imperceptibilité de notre solution, nous cachons l'image « Toy » de taille 40x40 pixels (figure 3.22 (b)) dans l'image « Lena » de taille 512x512 pixels (figure 3.22 (a)).

L'image stégo obtenue, présentée dans la figure 3.22 (c), montre que notre algorithme assure une bonne imperceptibilité visuelle car l'image stégo n'étant pratiquement pas distordue. La figure 3.22 (d) montre l'image secrète extraite de l'image stégo en suivant les étapes décrites précédemment. Il faut noter que dans ce cas, l'image secrète est reconstruite sans perte d'information.

(a) (b)

(c) (d)

Figure 3.22 : (a) Image originale, (b) Image secrète, (c) image stégo obtenue par l'algorithme proposé (d) Image extraite

La table 3.11 donne les valeurs des PSNR et SSIM pour les deux méthodes SSIS et celle améliorée. La valeur de β choisie pour effectuer les mesures est égale à 0.01. Les résultats obtenus montrent que la solution proposée apporte une très bonne amélioration au niveau du paramètre PSNR qui augmente de 31 jusqu'à 76.6 db. En plus, nous pensons que l'amélioration majeure de notre proposition se situe au niveau de la sécurité et la robustesse. Ceci ne pourra être démontré qu'à travers l'étude de la stéganalyse des deux méthodes, ce qui doit être effectué dans un futur proche, constituant une perspective importante de notre travail.

Image originale	message	Méthode proposée		Méthode existante	
		PSNR (dB)	SSIM	PSNR (dB)	SSIM
Lena	Toy	76.5917	0.9999	31.0164	0.9988

Table 3.11 : résultats des PSNR et SSIM, des deux algorithmes SSIS et la version améliorée proposée

3.7 Conclusion

L'insertion dans le domaine fréquentiel est une façon plus complexe, mais qui cependant peut être qualifiée de plus robuste que les méthodes d'intégration qui opèrent dans le domaine temporel. Dans ce chapitre, nous avons présenté différentes techniques basées sur les transformations DCT, DWT et sur l'étalement de spectre.

Pour les algorithmes basés DCT, nous avons considéré le LSB-DCT avec seuillage, et nous avons proposé d'utiliser le système chaotique RFCA afin de choisir aléatoirement les coefficients DCT dans lesquels nous insérons nos bits secrets, au lieu d'utiliser les coefficients selon un seuil donné. Cette amélioration augmente la capacité ainsi que le niveau de la sécurité. Pour les algorithmes basés DWT, nous avons considéré le "DWT-LSB Alpha-Fusion", et nous avons proposé plusieurs améliorations : pour augmenter sa capacité en insérant les bits secrets dans les parties à moyenne fréquence H et V de la transformée DWT; et pour hausser sa sécurité en utilisant le système chaotique SC pour chiffrer les bandes à dissimuler et pour effectuer une fusion chaotique de l'image originale avec les images secrètes. Pour les algorithmes basés sur l'étalement de spectre, nous avons considéré le SSIS, et nous avons proposé d'utiliser le chaos pour chiffrer l'image secrète et pour ensuite la moduler et élargir son spectre.

Pour toutes les propositions, plusieurs tests ont été effectués sur l'imperceptibilité et sur la mesure des deux critères de qualité objective le PSNR et le SSIM, ainsi qu'une comparaison avec les méthodes originales. Les résultats obtenus ont montré l'efficacité de nos améliorations, mais il reste l'analyse de la robustesse des algorithmes proposés, en étudiant notamment la séganalyse afin d'estimer leur résistance contre les différents types d'attaques.

3.8 Références

[Bhattacharya et al., 2012] Bhattacharya, T., Dey, N. et Cha. udhuri, S.R.B. (2012). A novel session based dual steganographic technique using DWT and spread spectrum, International Journal of Modern Engineering Research, volume 1, numéro 1, pages 157-161.

[Boora et Gambhir, 2013] Boora, M. etGambhir, M. (2013). Arnold Transform Based Steganography, International Journal of Soft Computing and Engineering (IJSCE),Volume-3, Issue-4.

[Chan et al., 2009] Chan, Y.-K., Chen, W.-T., Yu, S.-S., Ho, Y.-A., Tsai, C.-S. et Chu, Y.-P. (2009). A HDWT-based reversible data hiding method, The Journal of Systems and Software 82, pages 411– 421.

[Chang et al., 2007] Chang, C.-C., Lin, C.-C., Tseng, C.-S. et Tai, W.-L. (2007). Reversible hiding in DCT-based compressed images, Information Sciences.

[Cox et al., 1995] Cox, I., Kilian, J., Leighton, T., et Shamoon, T. (1995). Secure spread spectrum watermarking for multimedia. Technical report, NEC Research Institute.

[Danti et Preethi, 2010] Danti, A. et Preethi, A. (2010). Randomized embedding scheme based on dct coefficients for image steganography, IJCA Special Issue on Recent Trends in Image Processing and Pattern Recognition.

[El Assad et al., 2008] El Assad, S., Noura, H. et Taralova, I. (2008).Design and Analyses of Efficient Chaotic Generators for Cryptosystems,wcecs, pages 3-12, Advances in Electrical and Electronics Engineering – IAENG Special Edition of the World Congress on Engineering and Computer Science.

[El Assad, 2012] El Assad,S. (2012). Chaos Based Information Hiding and Security, in 7th International Conference for Internet Technology and Secured Transactions, IEEE, London, United Kingdom, pages 67- 72.

[Härtung et Girod, 1997] Härtung, F., et Girod, B. (1997). Fast Public-Key Watermarking of Compressed Video, Proceedings of the IEEE International Conference on Image Processing, Santa Barbara, CA.

[Karim et Sarraf, 2002] Karim, M. R.etSarraf, M. (2002). W-CDMA and cdma2000 for 3G Mobile Networks, McGrauw-Hill Telecom Professional.

[Khan] Khan, I. An Efficient Neural Network based Algorithm of Steganography for image, International Journal of Computer Technology and Electronics Engineering (IJCTEE) Volume 1, Issue 2.

[Marvel et al., 1998] Marvel, L. M., C. G. Bonclet, and C. T. Retter, Reliable Blind Information Hiding for Images, in Proceedings of the Second International Workshop on Information Hiding, volume 1525 of Lecture Notes in Computer Science, Springer, pages 48-61

[Marvel et al., 1999] Marvel, L.M., Boncelet Jr., C.G. et Retter, C.T. (1999). Spread spectrum steganography, IEEE Trans. Image Processing volume 8, numéro 8, pages 1075-83.

[Meel, 1999] Meel, J. (1999). Spread Spectrum: Introduction,http ://www.sss-mag.com/ss.html#tutorial.

[Paulson, 2006] Paulson, L.D. (2006). New system fights steganography. News briefs, IEEE Computer Society, pages 25-27.

[Saejung et al., 2013] Saejung, S., Boondee, A., Preechasuk, J. , and Chantrapornchai, C. (2013). On the comparison of digital image steganography algorithm based on DCT and wavelet, in *Computer Science and Engineering Conference (ICSEC),* pages 328–333.

[Pratibha et Shanti, 2013] Pratibha S., Shanti s. (2013) Digital Image Watermarking Using 3 level Discrete Wavelet Transform", Conference on Advances in Communication and Control Systems 2013 (CAC2S 2013), pages 129- 133, 2013

Chapitre 4

STEGANALYSE

4. Stéganalyse

4.1 Introduction

La stéganographie laisse en général des traces qui peuvent être détectables dans l'image stégo. Cela peut permettre à un adversaire, en utilisant des techniques de stéganalyse, de révéler qu'une communication secrète se déroule. Parfois, un adversaire est aussi appelé un « gardien ». Il existe deux types d'adversaires : passifs et actifs. Un adversaire passif examine uniquement la communication pour savoir si la communication contient des messages cachés. Cet adversaire ne modifie pas le contenu de la communication et l'autorise si aucune preuve de message secret n'est trouvée, sinon, il l'a bloque. Un adversaire actif peut provoquer volontairementl'interruption, la distorsion, ou la destruction de la communication, bien qu'il n'y ait aucune preuve de communication secrète. La plupart des méthodes de stéganographie actuelles sont conçues pour le scénario d'un adversaire passif. En général, il existe deux types de stéganalyse : spécifique et universelle. La stéganalyse spécifique est conçue pour attaquer un algorithme de stéganographie particulier. Ce type de stéganalyse spécifique peut produire généralement des résultats plus précis, mais ne parvient pas à des résultats satisfaisants si l'algorithme d'insertion des messages secrets est modifié.
La stéganalyse universelle peut être considérée comme une technique universelle pour détecter différents types de stéganographie. Plus encore, elle peut être utilisée pour détecter de nouvelles techniques de stéganographie là où une stéganalyse spécifique n'existe pas encore. En d'autres termes, la stéganalyse universelle est un outil de détection irremplaçable si l'algorithme d'intégration est inconnu ou secret.

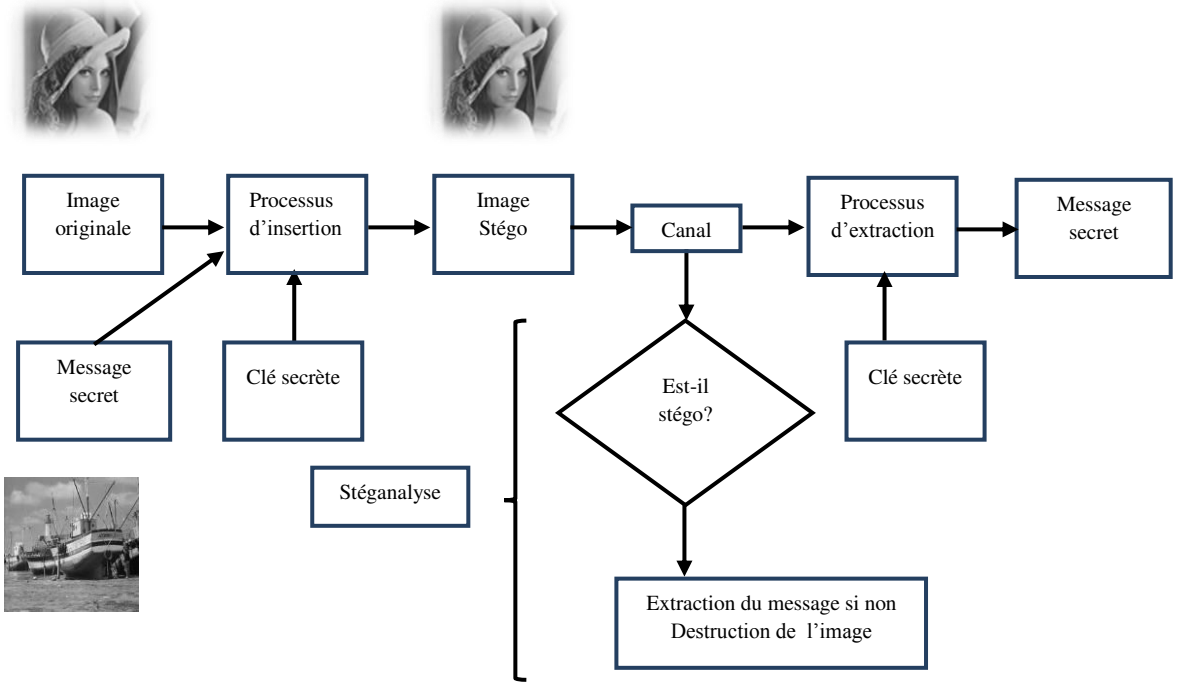


Figure 4.1: Schéma de principe de la stéganographie et de la stéganalyse

La figure 4.1 montre le schéma de principe de la stéganographie et de la séganalyse appliquée à l'image stégo lors de sa transmission à travers le canal de communication vers le récepteur.
L'image stégo peut faire l'objet d'opérations d'attaques malveillantes, par exemple l'ajout de bruit et de filtrage par un attaquant dont l'objectif est d'éliminer l'information intégrée (le message secret). L'attaque malveillante se peut se produire lorsque l'image stégo a « éveillé des soupçons », ou quand la présence des données cachées a été détectée par des techniques de stéganalyse.
Le chapitre est organisé comme suit :
Dans la section 4.2, nous présentons la stéganalyse visuelle, et dans la section 4.3, la stéganalyse universelle.
Dans la section 4.4, nous décrivons les étapes de la stéganalyse universelle utilisée. La transformée en ondelettes est décrite dans la section 4.4.1. L'analyse multi-résolution de l'image est présentée dans la section 4.4.1.1. Les méthodes d'extraction des caractéristiques des bandes H, V, D sont données dans les sections 4.4.2, 4.4.2.1, 4.4.2.2 et 4.4.2.3.
La classification est présentée dans la section 4.4.3, dans la section 4.4.3.1 nous décrivons les outils d'évaluation de performance de la classification, l'analyse discriminante de Fisher FLD est décrite dans la section 4.4.3.2, et la Quantification expérimentale des performances de la classification est donnée dans la section 4.4.3.3 avant de conclure dans la section 4.5.

4.2 Stéganalyse visuelle

L'attaque visuelle est considérée comme la forme la plus simple de stéganalyse. Comme son nom l'indique, une attaque visuelle implique l'examen de l'image stégo à l'œil nu ou avec l'assistance de l'ordinateur, pour détecter la présence des artefacts dus à l'insertion du message secret. Cependant, cela nécessite la présence de l'image originale pour réaliser la comparaison entre les deux images originale et stégo.
Comme discuté précédemment, la première règle de la stéganographie est que toutes les modifications apportées à l'image originale, du fait de l'insertion du message secret, ne doivent pas conduire à une dégradation de la qualité au moins visuelle de l'image stégo obtenue par rapport à l'image originale.

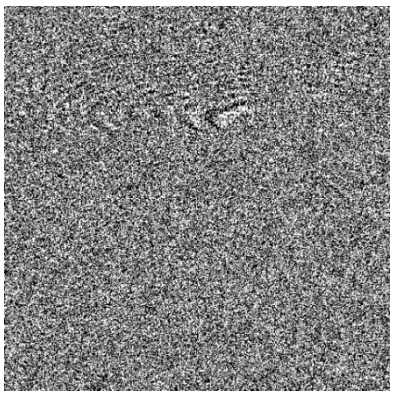
L'insertion d'un message dans le dernier plan de bit (plan LSB) peut se faire de façon séquentielle à partir du début de l'image, ou de façon aléatoire sur l'ensemble des pixels. L'attaque visuelle n'est pas efficace contre les méthodes d'insertion courantes utilisant essentiellement une insertion aléatoire, ou pour des images très texturées. Toutefois ce type d'attaque est une bonne introduction à la stéganalyse du schéma LSB séquentiel. L'insertion séquentielle du message perturbe le plan LSB proportionnellement à la taille du message secret. On observe sur ce plan une zone de bruit correspondant au message séquentiel.
Dans la figure 4.2, nous montrons une image naturelle originale de 512x512 pixels et dans les figures 4.3 et 4.4 les images stégo résultant respectivement de la stéganographie LSB adaptative (EEALSB) séquentielle et chaotique.
La figure 4.5 représente à gauche le plan LSB de l'image originale, et à droite le plan LSB de l'image stégo (EEALSB) séquentielle. Sur ce plan, nous observons des traces dans une zone déterminée en haut de l'image indiquant la présence d'un message secret inséré. Le gardien peut ainsi utiliser cette observation pour estimer le message secret inséré.

En revanche, lorsque le message est inséré de façon chaotique, il est difficile de remarquer la présence d'un message secret sur le plan LSB de l'image stego. En effet, dans ce cas et d'après la figure 4.6, les deux plans LSB avant et après l'insertion du message secret sont similaires. En conclusion, contrairement à l'insertion séquentielle, l'insertion chaotique (ou aléatoire) préserve la qualité visuelle de l'image stégo.

Figure 4.2 : Image originale

Figure 4.3 : Image stégo par AELSB séquentielle

Figure 4.4 : Image stégo par EAELSB chaotique



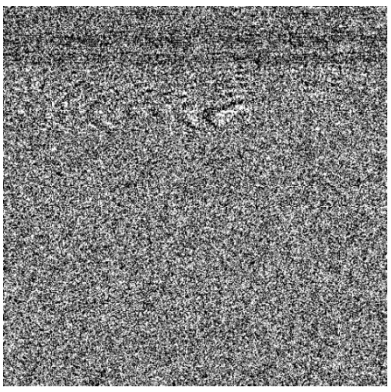


Figure 4.5 : Dernier plan de bits avant (à gauche) et après (à droite) insertion séquentielle du message

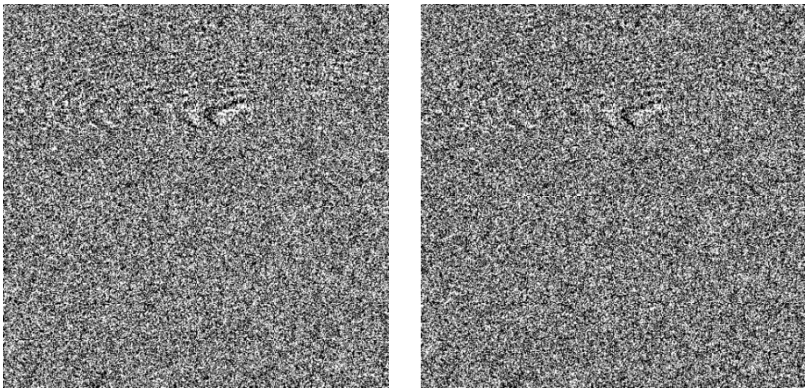


Figure 4.6 : Dernier plan de bits avant (à gauche) et après (à droite) insertion chaotique du message

4.3 Stéganalyse universelle

La Stéganalyse statistique universelle ne nécessite en principe aucune information a priori sur les méthodes de stéganographie utilisées pour la détection du message caché. A cet effet, elle s'appuie sur un processus d'apprentissage utilisant des images originales et des images stégos dont le résultat sera exploité par un processus de classification. Les réseaux de neurones ou des algorithmes de classification standards peuvent être utilisés pour construire le modèle de détection à partir des données expérimentales.
Farid propose une technique de séganalyse universelle pour les images en niveaux de gris et basée sur l'extraction des statistiques d'images d'ordre supérieur et l'analyse discriminante linéaire de Fisher comme classifieur [Farid, 2002]. Il crée un vecteur de caractéristiques de taille 24 (n - 1), où n est le degré de décomposition en ondelettes appliquée sur les images. Les composantes de ce vecteur sont : la moyenne, la variance, l'asymétrie et l'aplatissement des coefficients d'ondelettes et les statistiques d'erreurs calculées à partir d'un prédicteur linéaire optimal des magnitudes des coefficients.
Un meilleur classifieur non linéaire est proposé dans [Lyu et Farid, 2002]. Il est basé sur des « machines à vecteurs de support » (SVM : Support Vector Machine), qui sont une classe d'algorithmes d'apprentissage supervisé et destinés à résoudre des problèmes de discrimination et de régression. Lie et al., [Lie et Lin, 2005], analysent les propriétés statistiques (énergie de gradient et la variance statistique) des domaines spatiaux et la DCT (Discrete Cosines Transform) pour déterminer l'existence de messages cachés dans une image. Une méthode de séganalyse universelle, proposée par Shi et al. [Shi et al., 2005], utilise les moments statistiques de fonctions caractéristiques de l'image d'erreur de prédiction, l'image de test, leurs sous-bandes d'ondelettes, et un réseau neuronal comme classifieur. Wang et al., [Wang et Moulin, 2007], extraient les caractéristiques de coefficients d'ondelettes et utilisent les fonctions de densité de probabilité PDF (Probability Density Function) empiriques et les moments de la fonction caractéristique FC (Characteristic Function) d'images de sous-bandes d'ondelettes pour la stéganalyse universelle. Ces deux types de moments ont été largement utilisés en tant que caractéristiques dans les approches de stéganalyse.
Dans notre étude, nous avons utilisé l'analyse discriminante linéaire de Fisher comme classifieur des vecteurs caractéristiques de Farid, Shi et Wang. Ce choix se base sur la large variété de vecteurs caractéristiques testés qui fournissent une information sur les propriétés de l'image avant et après l'insertion du message. Ceci permettra par la suite d'aider à la conception d'algorithmes d'insertion robustes vis-à-vis de la stéganalyse.

Nous avons aussi, pour plus de clarté, fait un effort sur la formulation et les notations dans le problème traité dans cette étude. Par la suite, nous décrivons les étapes de la stéganalyse universelle.

4.4 Description des étapes de la stéganalyse universelle utilisée

La stéganalyse universelle se réalise en deux processus. Le premier est un processus d'apprentissage (Figure 4.7) appliqué aux vecteurs caractéristiques d'un ensemble d'images d'apprentissage avec et sans messages cachés, afin d'apprendre (optimiser) à un classifieur à distinguer entre les images originales et stégos. Le deuxième est un processus de test (Figure 4.8) similaire au processus d'apprentissage, mais qui utilise le classifieur du premier processus, donc déjà optimisé, pour séparer les vecteurs caractéristiques des images originales et stégos en deux classes distinctes.
Les diagrammes des processus d'apprentissage (Figure 4.7) et de test (Figure 4.8) montrent les différentes étapes pour réaliser la stéganalyse universelle :

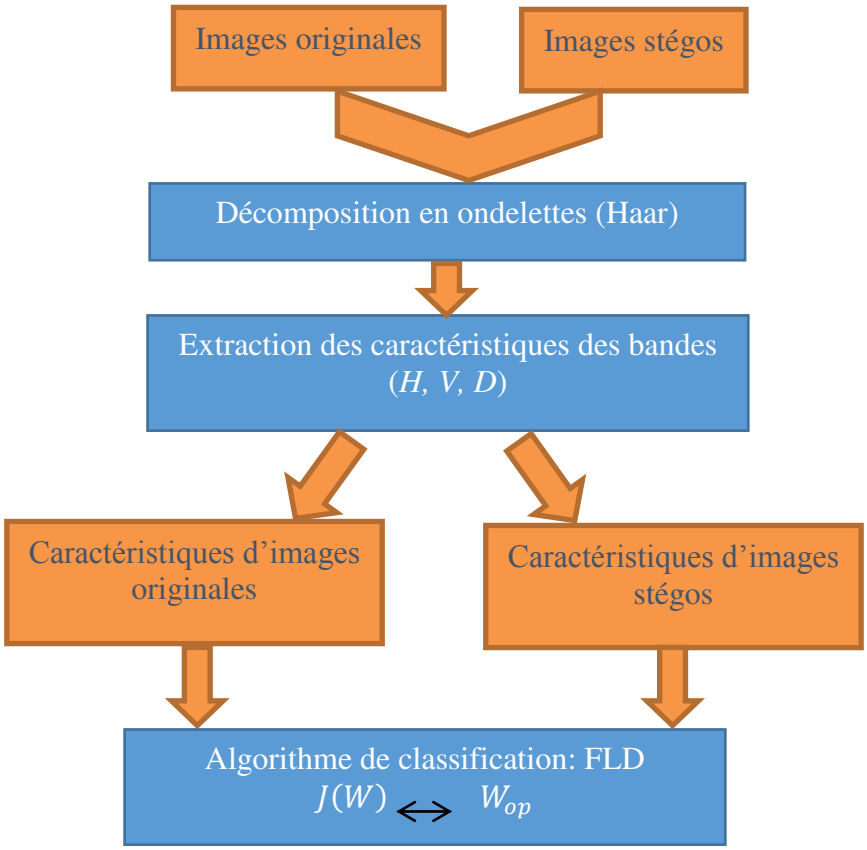


Figure 4.7 : Etapes du processus d'apprentissage

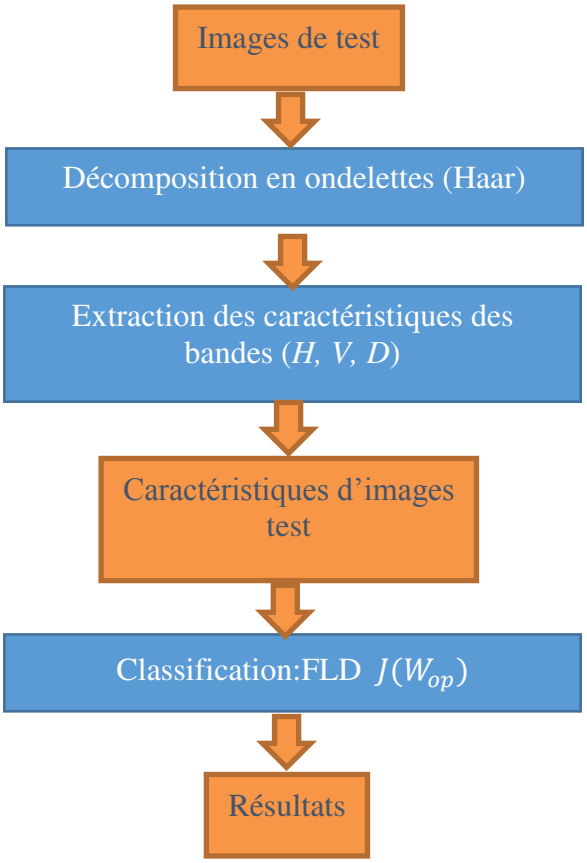


Figure 4.8 : Etapes du processus de test

Dans les paragraphes suivants, nous allons détailler les opérations faites dans chaque étape de deux processus de la stéganalyse universelle.

4.4.1 Transformée en ondelette

La transformée en ondelette WT (Wavelet Transform) aborde le problème de la résolution. Elle utilise une analyse multi-résolution permettant d'extraire des résolutions temporelles et fréquentielles différentes suivant les fréquences. L'analyse multi-résolution fournit une bonne résolution temporelle (et donc une mauvaise résolution fréquentielle) aux hautes fréquences, et une bonne résolution fréquentielle (donc une mauvaise résolution temporelle) aux basses fréquences. Cette transformée est particulièrement intéressante pour les signaux qui ont des composantes basses fréquences pendant un lapse de temps très court et des composantes hautes fréquences pendant des temps relativement longs. Dans le paragraphe suivant nous rappelons les équations de la famille d'ondelettes Haar utilisée dans nos simulations expérimentales.

4.4.1.1 Analyse multi-résolution de l'image :

Bases d'ondelettes orthonormées :

Une famille d'ondelettes ψ_n^m est définie à partir de l'ondelette mère ψ par

$$\psi_n^m(x) = a_0^{-m/2}\psi(a_0^{-m}x - nb_0) \quad (4.1)$$

Si ψ_n^m constitue un frame strict, l'orthonormalité est vérifiée par l'égalité de l'équation suivante :

$$\langle \psi_n^m, \psi_r^m \rangle = \delta_{n,r} \tag{4.2}$$

Où $\delta_{n,r}$ est le symbole de Kronecker défini par : $\begin{cases} \delta_{n,r}, si\ n = r \\ 0 \quad\quad sinon \end{cases}$

Avec : m l'échelle et n la translation.

L'orthonormalité est souvent démontrée à partir de l'ondelette de Haar (1910) où $a_0 = 2$ et $b_0 = 1$. Le choix de $a_0 = 2$ et $b_0 = 1$ est particulièrement adapté pour l'analyse multi-résolution en ondelettes. La famille d'ondelettes Haar est donc définie à partir des équations (4.3) et (4.4).

$$\psi_n^m(x) = 2^{-m/2}\psi(2^{-m}x - n) \tag{4.3}$$

$$\psi(x) = \begin{cases} 1 \ \ si\ 0 \leq x \leq \frac{1}{2} \\ -1 \ \ si\ \frac{1}{2} \leq x \leq 1 \\ 0 \quad\quad\quad sinon \end{cases} \tag{4.4}$$

Il est aisé pour cette ondelette de vérifier l'orthonormalité puisque à la même échelle m, il n'y a aucun recouvrement (overlapping) entre translations n et r. Rappelons que n, r et m appartiennent à $\mathbb{Z}$.

Nous donnons ci-après un exemple d'ondelettes Haar obtenues sous Matlab (Figure 4.9) par la fonction suivante : [wfun,xgrid] = wpfun('Haar',5); où 5 est le nombre des fonctions d'ondelettes de Haar, générées à partir de la fonction mère.

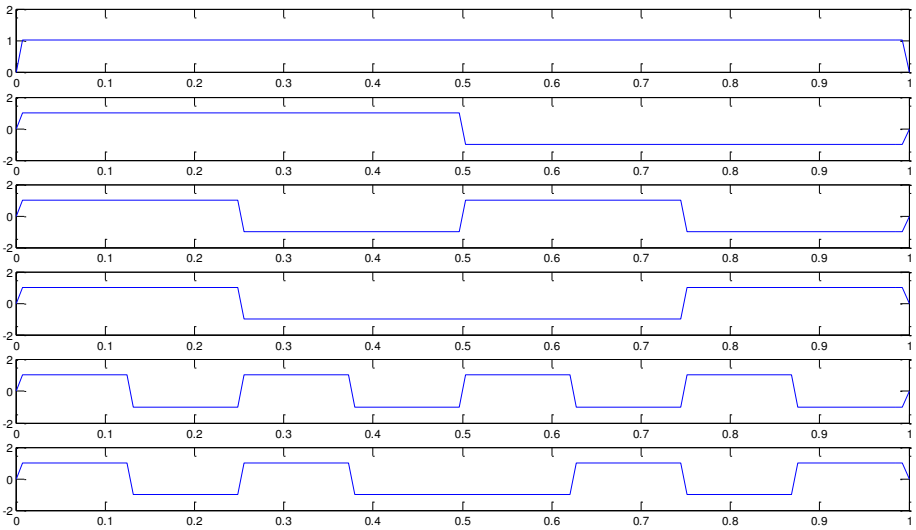


Figure 4.9 : Ondelettes de Haar

Ondelettes bidimensionnelles.

Nous abordons maintenant les ondelettes à deux dimensions utilisées dans le cadre de ce travail, sachant qu'il est tout à fait possible de définir des ondelettes de dimensions supérieures. Le traitement d'images constitue l'un des plus importants domaines de recherche

et développement de l'analyse en ondelettes à deux dimensions [Zettler et al., 1990], [Antonini et al., 1992], [Lewis et Knowles, 1992]. Par exemple, l'emploi de l'analyse en ondelettes bidimensionnelles a permis l'insertion de signature digitale [Inoue et al., 1998]. De la même manière que dans le cas monodimensionnel, les ondelettes à 2 dimensions sont définies par des produits tensoriels dans le repère cartésien, voir équation (4.5) :

$$\begin{aligned} \varphi(x,y) &= \varphi(x) \otimes \varphi(y) \\ \psi_H(x,y) &= \psi(x) \otimes \varphi(y) \\ \psi_V(x,y) &= \varphi(x) \otimes \psi(y) \\ \psi_D(x,y) &= \psi(x) \otimes \psi(y) \end{aligned} \tag{4.5}$$

Chaque direction est analysée de manière indépendante, et c'est le produit tensoriel $\otimes$ qui nous donne l'analyse en ondelettes bidimensionnelles. La fonction $\varphi(x,y)$ est une fonction d'échelle 2D, et les ondelettes bidimensionnelles 2D $\psi_H(x,y), \psi_V(x,y), \psi_D(x,y)$ représentent les détails dits horizontaux, verticaux et diagonaux. Cela se traduit, dans le plan d'analyse, par un découpage rectangulaire.

De ce fait, les coefficients en ondelettes dépendent du niveau *m* mais aussi de deux directions *p* et *q*, une dans la direction $\vec{p}$ et l'autre dans la direction $\vec{q}$. Les fonctions d'ondelettes bidimensionnelles données par l'équation (4.6), appliquées sur une image fournissent les coefficients d'ondelettes.

$$\begin{aligned} \varphi_{pq}^{m}(x,y) &= \varphi_{p}^{m}(x) \otimes \varphi_{q}^{m}(y) \\ \psi_{H\,pq}^{m}(x,y) &= \psi_{p}^{m}(x) \otimes \varphi_{q}^{m}(y) \\ \psi_{V\,pq}^{m}(x,y) &= \varphi_{p}^{m}(x) \otimes \psi_{q}^{m}(y) \\ \psi_{D\,pq}^{m}(x,y) &= \psi_{p}^{m}(x) \otimes \psi_{q}^{m}(y) \end{aligned} \tag{4.6}$$

Les fonctions ψ et φ sont associées à des filtres discrets miroirs en quadrature pour l'analyse de multi-résolution.

A chaque niveau *m*+1, les coefficients d'approximations (coefficients de la sous-bande A, voir ci-dessous), sont de nouveaux analysés par la relation (4.6) et la sous-bande A est recoupée en 4. La figure 4.10, montre un exemple de résultat de décomposition en ondelettes en 3 premiers niveaux de l'image Lena de taille 512 x 512.

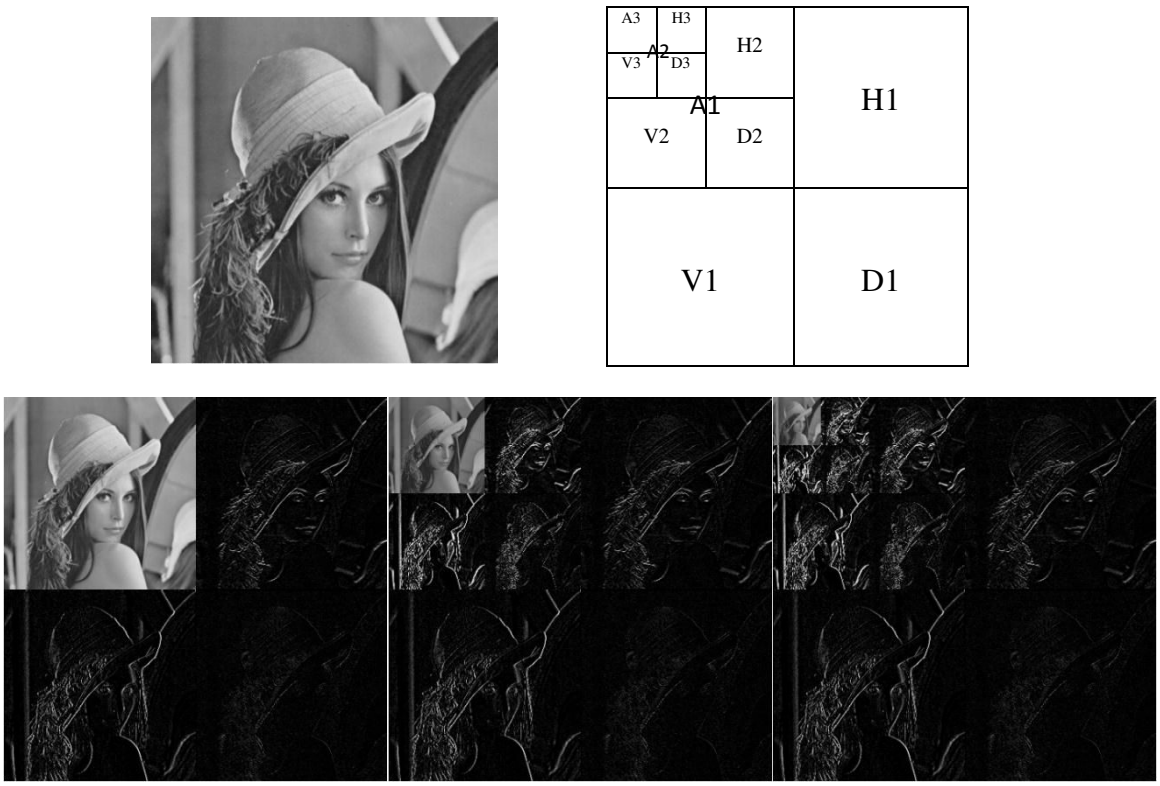


Figure 4.10 : Trois premiers niveaux d'une décomposition en ondelettes de l'image Lena

4.4.2 Extraction des caractéristiques des bandes H, V, D

L'extraction des caractéristiques est un processus de construction d'un ensemble de descripteurs statistiques discriminatoires ou attributs distinctifs statistiques à partir d'une image. Ces descripteurs ou attributs sont appelés caractéristiques. Alternativement, l'extraction des caractéristiques peut être considérée comme une forme de réduction de dimension. Il est souhaitable que les caractéristiques extraites soient sensibles aux artefacts résultants de l'insertion des messages secrets, par opposition au contenu de l'image. Dans le stade précoce de la recherche de stéganalyse universelle, les caractéristiques extraites comprenaient des mesures de qualité de l'image, la décomposition en ondelettes et des statistiques sur les moments des histogrammes. Récemment, les caractéristiques les plus utilisées sont constituées des moments statistiques des images dans les domaines spatiaux et fréquentiels [Farid, 2002], [Shi et al., 2005] et [Wang et Moulin, 2007] et de la matrice de co-occurrence, [Chen et al., 2006] et [Xuan et al., 2006].
Dans ce travail, nous avons étudié et appliqué 3 méthodes différentes de construction du vecteur caractéristique qui est utilisé par la suite dans la procédure de classification. Ces 3 méthodes sont détaillées ci-dessous :

4.4.2.1 Méthode 1 : Vecteur caractéristique construit à partir des moments empiriques basés PDF des coefficients multi-résolution et de leurs erreur de prédiction

Fonction Densité de Probabilité f(x) (PDF) :

Rappel

On dit que X est une variable aléatoire réelle continue s'il existe une fonction f de $\mathbb{R}$ (ensemble des éventualités possibles) dans $\mathbb{R}$, telle que :

1. pour tout x réel, $f(x) \geq 0$;
2. f est continue sur $\mathbb{R}$, sauf peut-être en un nombre fini de points où elle admet une limite à gauche et une limite à droite finies;
3. $\int_{-\infty}^{+\infty} f(x)dx$ existe et vaut 1;
4. pour tout *x* réel, la fonction de répartition *F(X)* de *X*, est définie par :

$$F(x) = \int_{-\infty}^{x} f(t)dt$$

On dit alors que f est une fonction densité de probabilité de X.

Ci-dessous, nous donnons un exemple de calcul de la fonction densité de probabilité *f(x)* d'une image (celle de Lena) à partir du calcul de son histogramme. Les figures 4.11 et 4.12 montrent les courbes correspondantes obtenues.

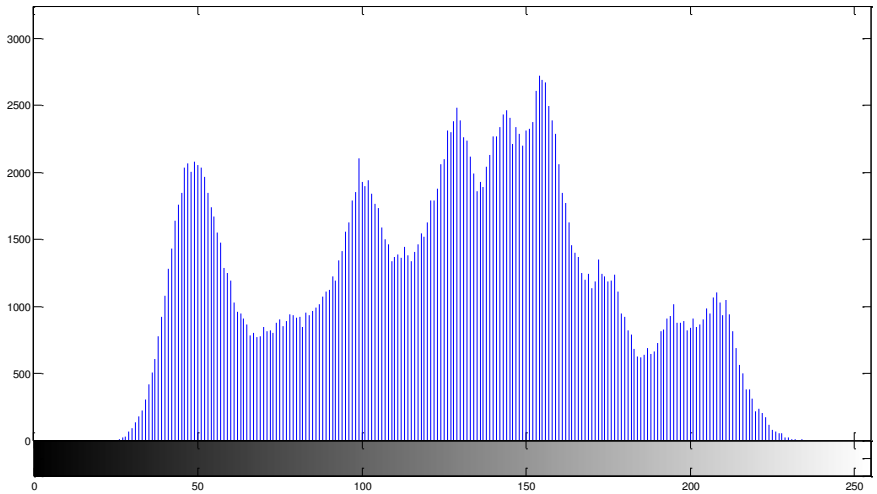


Figure 4.11 : Histogramme de l'image Lena

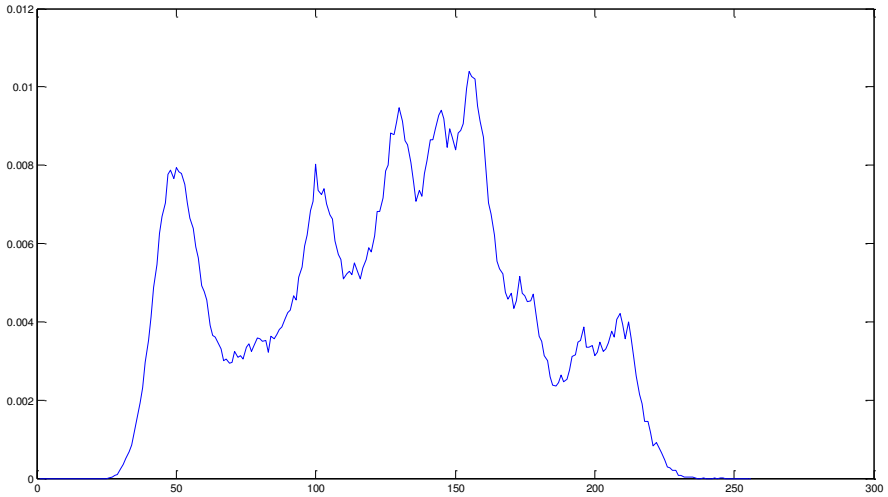


Figure 4.12 : Fonction densité de probabilité f(x) de l'image Lena

Moments empiriques basés fonction densité de probabilité

Les paramètres caractéristiques extraits de l'image doivent faire ressortir au mieux les propriétés du message inséré. Farid [Farid, 2002] réalise une décomposition multi-résolution de l'image (décomposition en ondelettes) par des filtres miroirs en quadrature qui décomposent l'image en sous-bandes d'orientations et de fréquences différentes : une sous-bande horizontale H, une sous-bande verticale V, une sous-bande diagonale D et une sous-bande basse fréquence A. En itérant le processus sur la sous-bande basse fréquence A, on obtient une décomposition multi-résolution de l'image (voir figure 4.10). Les différentes sous-bandes à l'échelle *m*= 1 . . . *n* sont notées *Hm, Vm et Dm*.
Dans notre travail, une première série de caractéristiques sont extraites à partir des statistiques sur les coefficients notés $S_m(x,y)$ de chaque sous-bande aux niveaux (échelles) *m* = 1 à *n* = 3. Ces caractéristiques sont : les moments centraux normalisés d'ordre 1 à 4, précisément la moyenne μ, la variance σ^2, l'asymétrie ξ et le kurtosis κ. Pour une sous-bande donnée, cela donne :

$$\begin{aligned} \mu &= \frac{1}{N_x N_y}\sum_{x,y} S_m(x,y) \\ \sigma^2 &= \frac{1}{N_x N_y}\sum_{x,y}(S_m(x,y)-\mu)^2 \\ \xi &= \frac{1}{N_x N_y \sigma^3}\sum_{x,y}(S_m(x,y)-\mu)^3 \\ \kappa &= \frac{1}{N_x N_y \sigma^4}\sum_{x,y}(S_m(x,y)-\mu)^4 - 3 \end{aligned} \qquad (4.7)$$

Où la somme s'effectue sur tous les coefficients S_m de la sous-bande en question de taille N_x x N_y.

$\boldsymbol{\mu}$ est la **moyenne** des coefficients $S_m(x,y)$.
$\boldsymbol{\sigma^2}$est la **variance** de $S_m(x,y)$.

$\boldsymbol{\xi}$ est le **coefficient de dissymétrie** (*skewness* en anglais) qui correspond à une mesure de l'asymétrie de $S_m(x,y)$. C'est le premier des paramètres de forme qui n'est pas un paramètre de position ni un paramètre d'échelle.
$\boldsymbol{\kappa}$ (**kurtosis)** est le **coefficient d'aplatissement** ou **coefficient d'aplatissement de Pearson**, qui correspond à une mesure de l'aplatissement de $S_m(x,y)$. C'est le deuxième des paramètres de forme.

La relation (4.7) nous permet de construire un premier vecteur caractéristique Z_s de taille : $Nm \times Nbd \times n$ éléments, soit $4 \times 3 \times 3 = 36$ composantes, où : Nm et Nbd représentent respectivement le nombre des moments et le nombre des bandes de détail utilisées.

La forme du vecteur caractéristique Z_s est donnée par :
$Z_s = [Z_1, Z_2, Z_3]$, où :
$Z_1 = [\mu_{H_1}, \mu_{V_1}, \mu_{D_1} \mid \sigma_{H_1}, \sigma_{V_1}, \sigma_{D_1} \mid \xi_{H_1}, \xi_{V_1}, \xi_{D_1} \mid \kappa_{H_1}, \kappa_{V_1}, \kappa_{D_1}]$.
$Z_2 = [\mu_{H_2}, \mu_{V_2}, \mu_{D_2} \mid \sigma_{H_2}, \sigma_{V_2}, \sigma_{D_2} \mid \xi_{H_2}, \xi_{V_2}, \xi_{D_2} \mid \kappa_{H_2}, \kappa_{V_2}, \kappa_{D_2}]$.
$Z_3 = [\mu_{H_3}, \mu_{V_3}, \mu_{D_3} \mid \sigma_{H_3}, \sigma_{V_3}, \sigma_{D_3} \mid \xi_{H_3}, \xi_{V_3}, \xi_{D_3} \mid \kappa_{H_3}, \kappa_{V_3}, \kappa_{D_3}]$.
Où Z_m (m = 1, 2, 3) représente les caractéristiques des sous-bandes de détail de chaque niveau m.

Prédiction linéaire

Une deuxième série de caractéristiques des coefficients $S_m(x,y)$ peut être calculé en tenant compte des propriétés intrinsèques des images. En effet, dans une image naturelle, comme les pixels ne varient pas aléatoirement, il est possible de prédire la valeur d'un pixel grâce à ceux voisins dans le domaine spatial ou dans le domaine multi-résolution. Cependant, lorsque l'on insère un message secret dans une image, la corrélation locale est perturbée. L'erreur de prédiction est donc discriminante et peut être utilisée afin de construire un deuxième vecteur caractéristique Z_p.
Plusieurs techniques de prédiction des coefficients $S^p_{H_m}(x,y)$, $S^p_{V_m}(x,y)$ et $S^p_{D_m}(x,y)$ (m = 1, 2, 3), peuvent être utilisées. Dans ce travail, nous avons utilisé un prédicteur linéaire, et plus précisément celui proposé par Farid dans [Farid, 2002] :

$$S^p_{H_m}(x,y) = w_1 S_{H_m}(x-1,y) + w_2 S_{H_m}(x+1,y) + w_3 S_{H_m}(x,y-1) w_4 S_{H_m}(x,y+1) + w_5 S_{H_{m+1}}\left(\frac{x}{2},\frac{y}{2}\right) + w_6 S_{D_m}(x,y) + w_7 S_{D_{m+1}}\left(\frac{x}{2},\frac{y}{2}\right) \quad (4.8)$$

$$S^p_{V_m}(x,y) = w_1 S_{V_m}(x-1,y) + w_2 S_{V_m}(x+1,y) + w_3 S_{V_m}(x,y-1) + w_4 S_{V_m}(x,y+1) + w_5 S_{V_{m+1}}\left(\frac{x}{2},\frac{y}{2}\right) + w_6 S_{D_m}(x,y) + w_7 S_{D_{m+1}}\left(\frac{x}{2},\frac{y}{2}\right) \quad (4.9)$$

$$S^p_{D_m}(x,y) = w_1 S_{D_m}(x-1,y) + w_2 S_{D_m}(x+1,y) + w_3 S_{D_m}(x,y-1) + w_4 S_{D_m}(x,y+1) + w_5 S_{D_{m+1}}\left(\frac{x}{2},\frac{y}{2}\right) + w_6 S_{H_m}(x,y) + w_7 S_{V_{m+1}}\left(\frac{x}{2},\frac{y}{2}\right) \quad (4.10)$$

La prédiction des différents coefficients font intervenir non seulement les coefficients voisins d'une sous-bande donnée (H, V ou D) pour un niveau m donné, mais aussi des coefficients venant des niveaux m+1 de la sous-bande courante et des autres sous-bandes.

Pour plus de clarté, nous donnons dans la figure 4.13, un schéma explicatif pour la prédiction du coefficient $S_{V_1}^p(x, y)$.

Les w_i sont les paramètres du prédicteur qu'il faut ajuster pour minimiser l'erreur de prédiction. Ces paramètres sont déterminés par la minimisation de la fonction d'erreur quadratique des coefficients de chaque sous-bande pour un niveau *m* donné :

$$E(w) = [S_m - Qw]^2 \tag{4.11}$$

Où S_m contient les coefficients des sous-bandes d'un niveau *m* donné, w contient les paramètres du prédicteur, et Q contient les amplitudes des coefficients voisins.

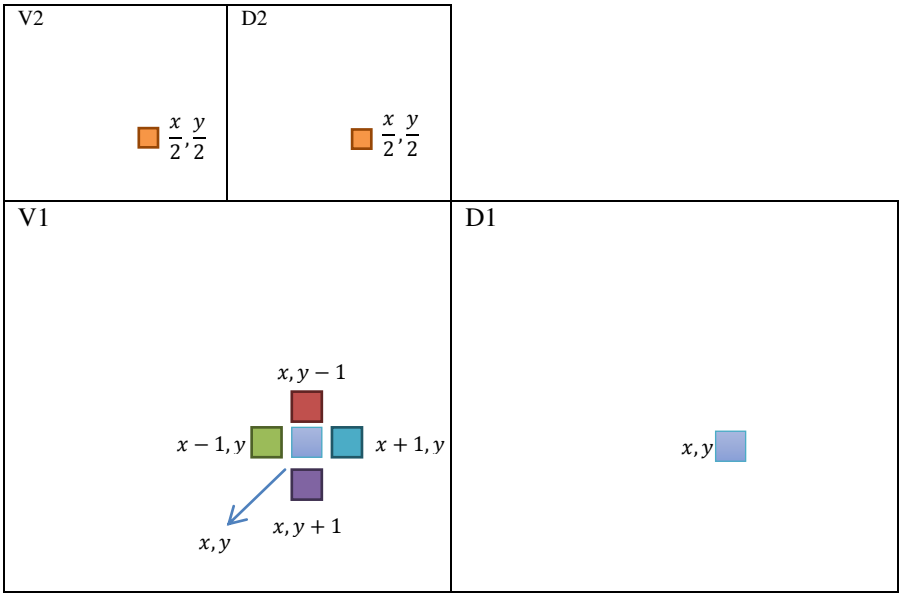


Figure 4.13 : Schéma de principe pour la prédiction du coefficient $S_{V_1}^p(x, y)$

La minimisation de l'équation (4.11) est donnée par :

$$\frac{dE(w)}{dw} = 2Q^T[S_m - Qw] = 0 \tag{4.12}$$

d'où :

$$w_{opt} = (Q^T Q)^{-1} Q^T S_m \tag{4.13}$$

Pour le prédicteur optimal, nous utilisons l'équation suivante de l'erreur de prédiction des coefficients de chaque sous-bande pour un niveau *m* donné :

$$\epsilon_m^p = \log_2 S_m - \log_2(|Qw_{opt}|) \tag{4.14}$$

L'équation (4.14) montre que w_{opt} (vecteur formé de 7 composantes) est appliqué à l'ensemble des coefficients concernés (voir équations (4.8) à (4.10)).

Nous calculons à partir de l'équation (4.14) pour chaque sous-bande d'un niveau *m* donné, les mêmes statistiques : *moyenne* μ*, variance* σ*, asymétrie* ζ *et kurtosis* κ, sur les coefficients erreur de prédiction (voir relation (4.7)).

La forme du vecteur caractéristique Z_ϵ^p est similaire à celui de Z_s et il est donné par :
$Z_\varepsilon^p = [Z_{1\varepsilon}^p, Z_{2\varepsilon}^p, Z_{3\varepsilon}^p]$, où

$$Z_{1\varepsilon}^p = [\mu_{\varepsilon_{H_1}}^p, \mu_{\varepsilon_{V_1}}^p, \mu_{\varepsilon_{D_1}}^p \mid \sigma_{\varepsilon_{H_1}}^p, \sigma_{\varepsilon_{V_1}}^p, \sigma_{\varepsilon_{D_1}}^p \mid \xi_{\xi_{H_1}}^p, \xi_{\xi_{V_1}}^p, \xi_{\xi_{D_1}}^p \mid \kappa_{\varepsilon_{H_1}}^p, \kappa_{\varepsilon_{V_1}}^p, \kappa_{\varepsilon_{D_1}}^p].$$
$$Z_{2\varepsilon}^p = [\mu_{\varepsilon_{H_2}}^p, \mu_{\varepsilon_{V_2}}^p, \mu_{\varepsilon_{D_2}}^p \mid \sigma_{\varepsilon_{H_2}}^p, \sigma_{\varepsilon_{V_2}}^p, \sigma_{\varepsilon_{D_2}}^p \mid \xi_{\varepsilon_{H_2}}^p, \xi_{\varepsilon_{V_2}}^p, \xi_{\varepsilon_{D_2}}^p \mid \kappa_{\varepsilon_{H_2}}^p, \kappa_{\varepsilon_{V_2}}^p, \kappa_{\varepsilon_{D_2}}^p].$$
$$Z_{3\varepsilon}^p = [\mu_{\varepsilon_{H_3}}^p, \mu_{\varepsilon_{V_3}}^p, \mu_{\varepsilon_{D_3}}^p \mid \sigma_{\varepsilon_{H_3}}^p, \sigma_{\varepsilon_{V_3}}^p, \sigma_{\varepsilon_{D_3}}^p \mid \xi_{\varepsilon_{H_3}}^p, \xi_{\varepsilon_{V_3}}^p, \xi_{\varepsilon_{D_3}}^p \mid \kappa_{\varepsilon_{H_3}}^p, \kappa_{\varepsilon_{V_3}}^p, \kappa_{\varepsilon_{D_3}}^p].$$

Finalement le vecteur caractéristique que sera utilisé pour la classification par apprentissage est donné par $Z = [Z_s \mid Z_\epsilon^p]$, comportant 72 composantes.

4.4.2.2 Méthode 2 : Vecteur caractéristique construit à partir des moments empiriques basés FC des coefficients multi-résolution

Fonction caractéristique (FC) : Rappel

Soit X une variable aléatoire réelle continue de fonction densité de probabilité *f*. La fonction caractéristique de X, notée par $\phi_X(t)$ est la transformée de Fourier de la fonction densité de probabilité *f*. Elle est définie de $\mathbb{R}$ dans $\mathbb{R}$ par :

$$\phi_X(t) = E\left(e^{itX}\right) = \int_{-\infty}^{+\infty} f_X(x)\, e^{itx}\, dx \qquad (4.15)$$

Dans la figure 4.14, nous donnons un exemple de calcul de la fonction caractéristique de l'histogramme de l'image de Lena.
La courbe obtenue est :

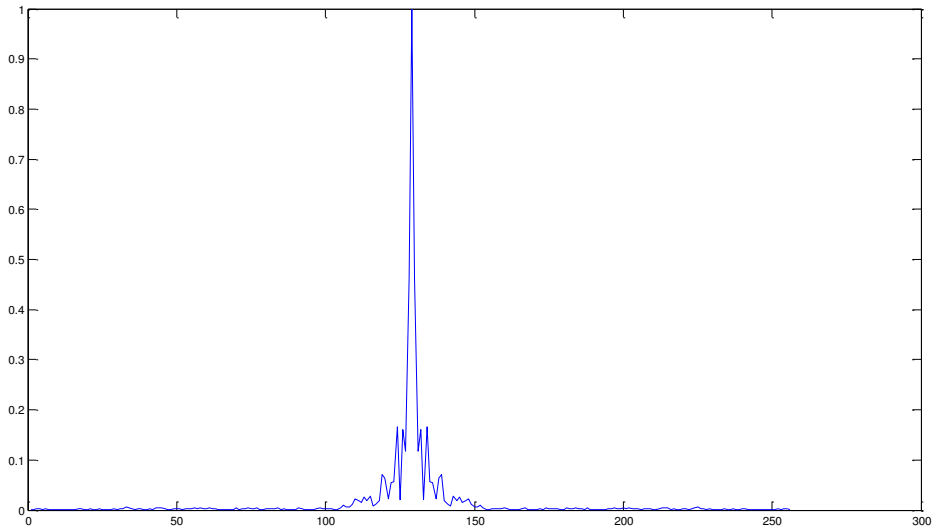


Figure 4.14 : Exemple de calcul de la fonction caractéristique de l'image Lena (considérée comme une variable aléatoire).

Moments empiriques basés fonction caractéristique :

[Shi et al., 2005] proposent une méthode de construction d'un premier vecteur caractéristique Z_s basée sur la fonction caractéristique et la décomposition en ondelettes. Les moments empiriques normalisés basés sur la fonction caractéristique $\phi(k)$d'ordre n = 1 à 3 sont donnés pour chaque sous-bande (*Am*, *Hm*, *Vm*, *Dm*) aux différents niveaux m = 1, 2 et 3 de la décomposition par la relation (4.16) :

$$M_{S_m}^n = \frac{\sum_{k=1}^{N/2} |\phi(k)| \times k^n}{\sum_{k=1}^{N/2} |\phi(k)|} \tag{4.16}$$

Où

$$\phi(k) = \sum_{i=1}^{N} h(i) exp\left\{\frac{j2\pi ik}{K}\right\} \quad 1 \leq k \leq K \tag{4.17}$$

est une composante de la fonction caractéristique à la fréquence k calculée à partir de l'histogramme de la sous-bande S_m, et N est le nombre total de points de l'histogramme.

La relation (4.16) nous permet de construire un premier vecteur caractéristique Z_s de moments de taille : 12* 3= 36 composantes, plus 3 moments de l'image initiale, soit un total de 39 composantes. La forme du vecteur caractéristique Z_s est alors :

$Z_s = [M_I^1, M_I^2, M_I^3 \,|M_{A_1}^1, M_{A_1}^2, M_{A_1}^3|M_{H_1}^1, M_{H_1}^2, M_{H_1}^3 \,|\, M_{V_1}^1, M_{V_1}^2, M_{V_1}^3|M_{D_1}^1, M_{D_1}^2, M_{D_1}^3|M_{A_2}^1, M_{A_2}^2, M_{A_2}^3|$
$M_{H_2}^1, M_{H_2}^2, M_{H_2}^3 \,|\, M_{V_2}^1, M_{V_2}^2, M_{V_2}^3|M_{D_2}^1, M_{D_2}^2, M_{D_2}^3 \,|\, M_{A_3}^1, M_{A_3}^2, M_{A_3}^3|\ M_{H_3}^1, M_{H_3}^2, M_{H_3}^3 \,|$
$M_{V_3}^1, M_{V_3}^2, M_{V_3}^3|M_{D_3}^1, M_{D_3}^2, M_{D_3}^3]$.

où, M_I^1, M_I^2, M_I^3 sont les moments de l'image initiale

La deuxième catégorie des caractéristiques est calculée à partir des moments de l'image erreur de prédiction et ses décomposition en ondelettes.
En stéganalyse, nous nous soucions seulement de la distorsion causée par la dissimulation de données. Il est connu que ce type de déformation peut être assez faible et, par conséquent couverte par d'autres types de bruits, notamment ceux dus à la caractéristique particulière de l'image elle-même. Pour rendre la stéganalyse plus efficace, il faut conserver le bruit dû à la dissimulation et éliminer au maximum les autres bruits. A cet effet, nous utilisons comme caractéristiques les moments d'ordre n = 1 à 3, basés fonction caractéristique de l'image d'erreur de prédiction (calculée à partir de l'image originale et l'image prédite) et les moments, d'ordre n = 1 à 3, basés fonction caractéristique des différentes sous- bandes de la décomposition en ondelettes aux différents niveaux m = 1, 2 et 3 (voir la relation (4.16)).

Le niveau de gris de chaque pixel de l'image prédite utilise le pixel correspondant de l'image originale et ses voisins.

Image erreur de prédiction :

Le niveau de gris de chaque pixel $\hat{x}$ de l'image prédite est donné par la relation suivante :

$$\hat{x} = \begin{cases} \max(a,b) & c \leq \min(a,b) \\ \min(a,b) & c \geq \max(a,b) \\ a+b-c & autrement \end{cases} \quad (4.18)$$

x	b
a	c

Où a, b et c sont les éléments voisins de x vers le bas, vers la droite et en diagonale vers le bas à droite.

La forme du vecteur caractéristique Z_ε^p est donné par :
$Z_\varepsilon^p = [M_{\epsilon_I}^{p1}, M_{\varepsilon_I}^{p2}, M_{\varepsilon_I}^{p3} | M_{A_1}^1, M_{A_1}^2, M_{A_1}^3 | M_{H_1}^1, M_{H_1}^2, M_{H_1}^3 | M_{V_1}^1, M_{V_1}^2, M_{V_1}^3 | M_{D_1}^1, M_{D_1}^2, M_{D_1}^3 |$
$M_{A_2}^1, M_{A_2}^2, M_{A_2}^3 | M_{H_2}^1, M_{H_2}^2, M_{H_2}^3 | M_{V_2}^1, M_{V_2}^2, M_{V_2}^3 | M_{D_2}^1, M_{D_2}^2, M_{D_2}^3 | M_{A_3}^1, M_{A_3}^2, M_{A_3}^3 | M_{H_3}^1, M_{H_3}^2, M_{H_3}^3 |$
$M_{V_3}^1, M_{V_3}^2, M_{V_3}^3 | M_{D_3}^1, M_{D_3}^2, M_{D_3}^3]$.

Où, par exemple $M_{A_1}^1, M_{A_1}^2, M_{A_1}^3$ sont les moments d'ordre 1, 2 et 3, basés fonction caractéristique de la sous-bande A_1 de la décomposition en niveau 1 de l'image erreur.

Finalement le vecteur caractéristique que sera utilisé pour la classification par apprentissage est donné par $Z = [Z_S | Z_\varepsilon^p]$, comportant 78 composantes.

4.4.2.3 Méthode 3 : Vecteur caractéristique construit à partir des moments empiriques basés FC et PDF de l'erreur de prédiction de l'image et ses différentes sous-bandes de la décomposition multi-résolution.

[Wang et Moulin, 2007] proposent une méthode de construction d'un premier vecteur caractéristique Z_s qui combinent les deux types des moments normalisés, les moments basés fonction densité de probabilité et les moments basés fonction caractéristique de différentes sous-bandes de la décomposition multi-résolution à trois niveaux de l'image de gris.

Moments empiriques normalisés basés FC

Nous utilisons l'expression suivante, proposée par [Wang et Moulin, 2007], pour calculer les moments d'ordre n = 1 à 6 de l'image initiale et ses sous-bandes (*Am*, *Hm*, *Vm*, *Dm*) aux différents niveaux m = 1, 2 et 3 de la décomposition en ondelettes :

$$M_{S_m}^n = \frac{\sum_{k=1}^{N/2} |\phi(k)| \times sin^n(\frac{\pi k}{K})}{\sum_{k=1}^{N/2} |\phi(k)|} \quad (4.19)$$

Avec :

$$\phi(k) = \sum_{i=1}^{N} h(i) exp\left\{\frac{j2\pi ik}{K}\right\} \quad (4.20)$$
$$1 \leq k \leq K$$

est une composante de la fonction caractéristique à la fréquence k estimée à partir de l'histogramme.
Ceci permet déjà d'avoir 6*1 + 6 *(4*3) = 78 composantes.

Aussi, et afin d'améliorer les performances du système d'apprentissage, on calcule les moments des sous-bandes A_2', H_2', V_2', D_2' obtenues par la décomposition de la sous-bande diagonale D_1, soit (6 *4) = 24 composantes. Donc, la taille totale du vecteur Z_s est 102 composantes.

$Z_S = [M_I^i \,|M_{A_1}^i|M_{H_1}^i \,|\, M_{V_1}^i|M_{D_1}^i|\, M_{A_2}^i|\, M_{H_2}^i \,|\, M_{V_2}^i|M_{D_2}^i|\, M_{A_3}^i|\;\; M_{H_3}^i|\, M_{V_3}^i|M_{D_3}^i|\, M_{A_2'}^i|M_{H_2'}^i \,|\, M_{V_2'}^i|M_{D_2'}^i],$
$i = 1, 2, \cdots, 6.$
Avec par exemple, $M_I^i = [M_I^1, M_I^2, M_I^3, M_I^4, M_I^5, M_I^6]$, sont les moments d'ordre 1 à 6 de l'image initiale.

La deuxième catégorie des caractéristiques est formée des moments d'ordre 1 à 6 de l'erreur de prédiction : $\epsilon_m^p = \log_2 S_m - \log_2(|Qw_{opt}|)$ des coefficients de chaque sous-bande pour un niveau m donné :

$$m_{\epsilon_m^p}^n = \frac{1}{N}\sum_{i=1}^{N}\left(\epsilon_m^p\right)^n \quad n = 1, 2, \cdots, 6 \tag{4.21}$$

Le vecteur de la deuxième catégorie est défini par Z_ε^p :

$Z_\epsilon^p = [m_{\varepsilon H_m}^i \,|\, m_{\varepsilon V_m}^i|m_{\varepsilon D_m}^i \,]$, pour chaque $m = \{1, 2, 3\}$; $i = 1, 2, \cdots, 6$
La taille de Z_ϵ^p est : 3*6 * 3 = 54 composantes
Finalement le vecteur caractéristique qui sera utilisé pour la classification par apprentissage est $Z = [Z_s|Z_\varepsilon^p]$, il comporte 156 composantes.

4.4.3 Classification

La dernière étape des processus d'apprentissage (Figure 4.7) et de test (Figure 4.8) de la stéganalyse universelle est la classification. Elle a pour objectif de grouper les images dans deux classes : classe des images originales et classe des images stégos, en fonction de leurs valeurs caractéristiques. Dans le processus d'apprentissage, la classification utilisée est supervisée. En apprentissage supervisé, un ensemble d'échantillons d'apprentissage (comprenant des fonctions d'entrée et des étiquettes de classe) est introduit pour apprendre le classifieur. Une fois le classifieur formé, il peut être utilisé dans le processus de test de la Figure 4.8 pour prédire l'étiquette de classe sur la base du vecteur des caractéristiques d'entrée. Dans la stéganalyse universelle, des classifieurs basés sur l'analyse discriminante linéaire de Fisher (FLD), du réseau de neurones et des machines à vecteurs supports (SVM) sont couramment utilisés.
Dans notre travail, nous utilisons l'analyse discriminante linéaire de Fisher (FLD). Dans les paragraphes suivants, nous rappelons son principe et précisons son utilisation dans le contexte de notre étude.

4.4.3.1 Outils d'évaluation de performance de la classification :

La quantification de la performance d'une classification peut être faite en utilisant la matrice de confusion et le coefficient de Kappa [Santos, 2013], http ://kappa.chez alice.fr/Kappa_2juges_Def.htm, http ://fr.wikipedia.org/wiki/Kappa_de_Cohen. La courbe ROC (Receiver Operating Characteristics), couramment utilisée pour évaluer la performance de la classification, est seulement efficace dans le cas d'un paramètre caractéristique et non d'un vecteur caractéristique.
Par la suite nous définissons et détaillons les paramètres matrice de confusion et coefficient de Kappa afin de bien les exploiter dans l'interprétation de nos classifications.

Matrice de confusion : Une matrice de confusion de dimension 2 x 2 [Kohavi et Provost, 1998], (Wikipédia) contient des informations sur les classes de références et estimées réalisées par un système de classification supervisé. La performance du système est généralement évaluée en utilisant les données de la matrice. Chaque colonne de la matrice représente le nombre d'occurrences d'une classe de référence, tandis que chaque ligne représente le nombre d'occurrences d'une classe estimée.
Un des intérêts de la matrice de confusion est qu'elle montre rapidement si le système parvient à classifier correctement.
La table 4.1, donne la matrice de confusion 2 x 2 résumant les différentes situations.

Dans cette matrice, nous avons :

Vrai positif (VP) : True positive (TP) : représente le nombre d'images stégos correctement classifiées parmi l'ensemble d'images stégos testées.
Faux négatif (FN) : False negative (FN) : représente le nombre d'images stégos non correctement classifiées parmi l'ensemble d'images stégos testées.
Faux positif (FP) : (False positive (FP) : représente le nombre d'images originales classifiées stégos parmi l'ensemble d'images originales testées.
Vrai négatif (VN) : True negative (TN) : représente le nombre d'images originales classifiées originales parmi l'ensemble d'images originales testées.
Sensibilité (Se) : Sensitivity : la sensibilité est la proportion d'images stégos qui sont correctement classifiées comme telles parmi toutes les images stégos testées. Elle est donnée par :

$$Se = \frac{VP}{VP+FN} \tag{4.22}$$

Une mesure de la sensibilité s'accompagne toujours d'une mesure de la spécificité.

Spécificité (Sp) : **Specificity** : la spécificité est la proportion d'images originales qui sont correctement classifiées comme telles parmi toutes les images originales testées. Elle est donnée par :

$$Sp = \frac{VN}{VN+FP} \tag{4.23}$$

		Images examinées (classes de références) Condition		
		Images Stégos : H0 (Condition positive)	Images Originales : H1 (Condition négative)	
Images Classifiées (Classe estimées) (Test outcome)	H0 (Test outcome positive)	**Vrai positif : VP (True positive : TP) (1 - α)**	**Faux positif : FP (False positive : FP)** (Type II error) **β (PFA)**	Précision (Pr) Pr = VP / (VP+FP) Precision = Σ True positive / Σ Test outcome positive
	H1 Test outcome negative	**Faux négatif : FN (False negative : FN)** (Type I error) **α (PND)**	**Vrai négatif : VN (True negative : TN) 1 - β**	Valeur Prédictive Négative (VPN) VPN = VN / (VN+FN) Negative predictive value = Σ True negative / Σ Test outcome negative
		Sensibilité (Se) Se = VP / (VP+FN) Sensitivity = Σ True positive / Σ Condition positive	Spécificité (Sp) Sp = VN / (VN+FP) Specificity = Σ True negative / Σ Condition negative	Exactitude (Ex) Ex = (VP + VN) / (VP+FN+FP+VN) Accuracy = Σ True positive + Σ True negative / Σ Total population

Table 4.1 : Matrice de confusion

Notons qu'en statistique, la sensibilité d'un test mesure sa capacité à donner un résultat positif lorsqu'une hypothèse est vérifiée. Elle s'oppose à la spécificité qui mesure la capacité d'un test à donner un résultat négatif lorsque l'hypothèse n'est pas vérifiée.
Ensemble, la sensibilité et la spécificité d'un test donnent une appréciation de sa validité intrinsèque. Prises séparément, elles ne veulent rien dire. Par exemple, un test avec une sensibilité 95% n'a aucune valeur si sa spécificité n'est que de 5%. Dans cet exemple, le test est simplement positif chez 95% des images examinées sans aucune corrélation avec la présence d'un message secret.

Précision (Pr) ou Valeur Prédictive Positive : Precision or Positive Predective Value : est la proportion d'images classifiées stégos lorsque le résultat du test de classification est positif (Hypothèse H0 retenue). Elle est donnée par :

$$Pr = \frac{VP}{VP+FP} \tag{4.24}$$

Valeur Prédictive Négative (VPN) : Negative predictive value : est la proportion d'images classifiées originales lorsque le résultat du test de classification est négatif (Hypothèse H1 retenue). Elle est donnée par :

$$VPN = \frac{VN}{VN+FN} \tag{4.25}$$

Exactitude (Ex) : Accuracy : est la proportion d'images stégos et originales qui sont correctement classifiées dans leurs classes respectives, parmi toutes les images examinées (originales et stégos).

$$Ex = \frac{VP+VN}{VP+FN+FP+VN} \tag{4.26}$$

Cette notion n'a que très peu d'intérêt en pratique contrairement à la notion de la sensibilité et de la spécificité.

Coefficient de correspondance Kappa : Le coefficient Kappa ou coefficient de correspondance, est un coefficient mesurant l'accord (concordance) entre 2 évaluations qualitatives liées, en tenant compte de la part de concordance due au hasard. Donc, sous l'hypothèse d'indépendance des juges, la concordance aléatoire P_a n'est pas nulle.

En faisant le lien avec la matrice de confusion, nous pouvons définir à partir de la table 4.2, ci-dessous le coefficient Kappa qui permet une mesure globale de la classification.

		Images examinées (classes de références)		
		Images Stégos : H0	Images Originales : H1	
Images classifiées (Classes estimées)	H0	**P_{11} (VP)**	**P_{12} (FP)**	**$P_{1.} = P_{11} + P_{12}$ (VP + FP)**
	H1	**P_{21} (FN)**	**P_{22} (VN)**	**$P_{2.} = P_{21} + P_{22}$ (FN + VN)**
		$P_{.1} = P_{11} + P_{21}$ (VP + FN)	**$P_{.2} = P_{12} + P_{22}$ (FP + VN)**	**1 (VP + FN + FP + VN)**

Table 4.2 : Lien entre la matrice de confusion et le coefficient Kappa

Le coefficient Kappa est défini par la relation suivante :

$$Kappa = \frac{P_0 - P_a}{1 - P_a} \tag{4.27}$$

Où :

P_0 est la proportion observée de paires concordantes (bonne classification sous chacune des hypothèses H0 et H1), donnée par :

$$P_0 = \sum_{i=1}^{2} P_{ii} = P_{11} + P_{22} \tag{4.28}$$

et P_a est la proportion de concordance aléatoire, donnée par

$$P_a = \sum_{i=1}^{2} P_{i.} \times P_{.i} = P_{1.} \times P_{.1} + P_{2.} \times P_{.2}$$
$$= (P_{11} + P_{12}) \times (P_{11} + P_{21}) + (P_{21} + P_{22}) \times (P_{12} + P_{22}) \tag{4.29}$$

Le coefficient Kappa varie entre $\left\{\frac{-P_a}{1-P_a},\ 1\right\}$.

L'accord entre les deux évaluations n'est intéressant que si $P_O > P_a$; la quantité $(1 - P_a)$, représente la concordance maximale sans prise en compte du hasard [Fabbro-Peray, 2006-2007] .

Intervalle de variation et interprétation du coefficient Kappa [Landis et Koch, 1977],

La table 4.3 suivante fournit la grille d'interprétation du coefficient Kappa.

Coefficient de Kappa	Accord
1 ($P_o = 1$)	Parfait
> 0.8	Très bon
0.6 -0.8	Bon
0.40-0.60	Assez bon
0.20-0.40	Passable
< 0.2	Mauvais
< 0	Exécrable

Table 4.3 : la grille d'interprétation du coefficient Kappa

Dans le paragraphe suivant nous introduisons l'analyse discriminante de Fisher FLD que nous utiliserons par la suite dans la procédure de classification.

4.4.3.2 Analyse discriminante de Fisher FLD

L'analyse discriminante de Fisher permet une réduction du nombre *nd* de dimensions, avec une faible augmentation de la probabilité d'erreur.

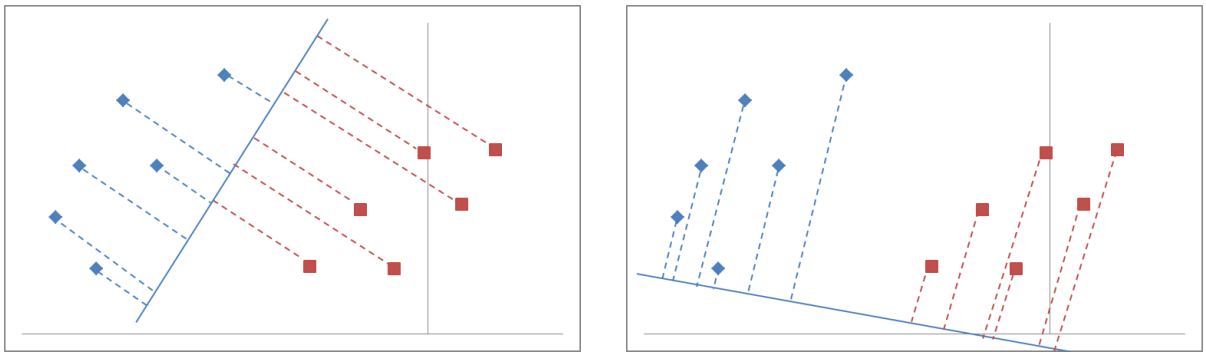


Figure 4.15 : Projection des points en deux directions

Pour un problème de discrimination à deux classes (ici, classe d'images originales et classe d'images stégo), l'opération de classification peut se voir comme la séparation d'un espace de grande dimension (l'espace des vecteurs caractéristiques *Z*) par un hyperplan, de sorte que tous les points d'un côté de l'hyperplan soient classés en tant qu'images originales (cover), et que les autres points soient classés en tant qu'images stégos.

Pour notre application, l'analyse discriminante de Fisher s'énonce comme suit :
Soit $\boldsymbol{Z} = \{Z_1, Z_2, \dots, Z_N\}$ un ensemble de vecteurs caractéristiques, ayant chacun *nd* dimensions. Parmi ces vecteurs, N_1 vecteurs sont des vecteurs caractéristiques $\boldsymbol{Z_c}$, d'étiquette 1, provenant d'images cover, et N_2vecteurs sont des vecteurs caractéristiques $\boldsymbol{Z_s}$, d'étiquette 2, provenant d'images stégos, avec $N = N_1+N_2$. Nous souhaitons former l'ensemble des valeurs de projection $\boldsymbol{Z_p} = \{Z_{p1}, Z_{p2}, \dots, Z_{pN}\}$, de dimension *N*, par combinaison linéaire des vecteurs caractéristique $\boldsymbol{Z}$ comme suit :

$$\boldsymbol{Z_p} = \boldsymbol{W^t}\, \boldsymbol{Z} \tag{4.30}$$

Où $\boldsymbol{W}$ est le vecteur d'orientation de dimension *nd*, pour effectuer la projection des donnés.

Dans notre étude, le vecteur de caractéristique Z est projeté dans un espace à deux classes. Cette projection tend à maximiser la distance entre les moyennes de ces deux classes (M_{cp}, M_{sp}), tout en minimisant la variance à l'intérieur de chacune des classes S_{cp}, S_{sp}, où : M_{cp} , M_{sp}, S_{cp} et S_{sp} sont respectivement les moyennes et les variances des classes d'images originales et d'images stégos.

Processus d'apprentissage
Le processus d'apprentissage consiste à optimiser (maximiser) l'expression suivante :

$$J(W) = \frac{|M_{cp} - M_{sp}|^2}{S_{cp} + S_{sp}} \tag{4.31}$$

Où :

$$M_{cp} = \frac{1}{N_1}\sum_{Z_p \in Z_{cp}} Z_p \ = \frac{1}{N_1}\sum_{Z \in Z_c} W^t Z = W^t M_c \quad (4.32)$$

Avec :

$$M_c = \frac{1}{N_1}\sum_{Z \in Z_c} Z \quad (4.33)$$

Le vecteur caractéristique moyen cover, de dimension *nd*

$$M_{sp} = \frac{1}{N_2}\sum_{Z_p \in Z_{sp}} Z_p \ = \frac{1}{N_2}\sum_{Z \in Z_s} W^t Z = W^t M_s \quad (4.34)$$

Avec :

$$M_s = \frac{1}{N_2}\sum_{Z \in Z_s} Z \quad (4.35)$$

Le vecteur caractéristique moyen stégo, de dimension *nd*

La matrice de variance des vecteurs caractéristiques des images cover après projection est :

$$S_{cp} = \sum_{Z_p \in Z_{cp}} \left(Z_p - M_{cp}\right)^2 = \sum_{Z \in Z_c} (W^t Z - W^t M_c)^2 = \sum_{Z \in Z_c} W^t\,(Z - M_c)(Z - M_c)^t\, W = W^t S_c W \quad (4.36)$$

Où :

$$S_c = (Z - M_c)(Z - M_c)^t \quad (4.37)$$

est la matrice de variance des vecteurs caractéristiques des images cover avant projection, de dimension *nd* x *nd*

La matrice de variance des vecteurs caractéristiques des images stégos après projection est :

$$S_{sp} = \sum_{Z_p \in Z_{sp}} (Z_p - M_{sp})^2 = \sum_{Z \in Z_s} (W^t Z - W^t M_s)^2 = \sum_{Z \in Z_s} W^t\,(Z - M_s)(Z - M_s)^t\, W = W^t S_s W \quad (4.38)$$

Où :

$$S_s = (Z - M_s)(Z - M_s)^t \quad (4.39)$$

est la matrice de variance des vecteurs caractéristiques des images stégos avant projection, de dimension *nd* x *nd*

La matrice de variance intra-classes après projection est alors :

$$S_{cp} + S_{sp} = W^t(S_c + S_s)W = W^t S_W W \quad (4.40)$$

Où :

$$S_W = (S_c + S_s) \quad (4.41)$$

est la matrice dispersion intra-classes avant projection, de dimension *nd* x *nd*

La distance entre les moyennes de deux classes est donnée par l'équation suivante :

$$\left(M_{cp} - M_{sp}\right)^2 = (W^t M_c - W^t M_s)^2 = W^t(M_c - M_s)(M_c - M_s)^t W = W^t S_B W \quad (4.42)$$

Où :

$$S_B = (M_c - M_s)(M_c - M_s)^t \tag{4.43}$$

est la matrice de dispersion inter –classes avant projection, de dimension *nd* x *nd*

Optimiser (maximiser) la relation (4.31), revient à optimiser la relation suivante :

$$J(W) = \frac{W^t S_B W}{W^t S_W W} \tag{4.44}$$

La solution de l'équation (4.44) est donnée par [Li et Wang, 2014] :

$$W_{opt} = S_W^{-1}(M_c - M_s) \tag{4.45}$$

Processus de test

Les différentes étapes du processus de test sont identiques à celles du processus d'apprentissage mise à part la dernière étape qui consiste à réaliser la classification.
L'étape de la classification se déroule comme suit :

Soit *Z*, la matrice de test, contenant des vecteurs caractéristiques covers et stégo :
La projection de *Z*, sur la droite d'orientation W_{opt}, fournit l'ensemble des valeurs projetées Z_p .

$$Z_p(j) = \sum_{i=1}^{9} W_{opt}(i) \times Z(i,j) + b \qquad j = 1, \dots, N \tag{4.46}$$

où *b* est un seuil de discrimination entre les deux classes, il peut être fixé à une valeur qui se trouve à mi-distance entre les deux moyennes projetées cover et stego

$$b = 0.5 \times (M_{cp} + M_{sp}) \tag{4.47}$$

Avec :

$$M_{cp} = W_{opt}^t \times M_c$$
$$M_{sp} = W_{opt}^t \times M_s$$

Où

W_{opt}^t est le transposé de W_{opt}

Le résultat $Z_p(j)$, $j = 1, \dots, N$ détermine le type de la classe de chaque image test (valeur projetée). En effet :
Si $Z_p(j) \geq 0$, l'image sous test est cover (originale), autrement elle est stégo.

Exemple explicatif

Afin de clarifier nos propos et pour plus de clarté, nous donnons ci-dessous un exemple explicatif. Dans cet exemple, nous considérons tout d'abord un ensemble de 6 vecteurs caractéristiques, extraient de 6 images originales, comprenant chacun 9 composantes

caractéristiques. Ces valeurs sont données par la matrice Z_c ci-dessous de dimension 9 x 6. Ensuite, des images stego correspondantes, nous avons extrait 6 vecteurs caractéristiques qui sont représentés par la matrice Z_s ci-dessous.

$$Z_c = \begin{pmatrix} 0.4658 & 0.5966 & 0.5128 & 0.5449 & 0.4928 & 0.6536 \\ 0.5785 & 0.5802 & 0.4651 & 0.5960 & 0.5632 & 0.4381 \\ 0.3956 & 0.5674 & 0.5226 & 0.5732 & 0.5266 & 0.3108 \\ 0.2252 & 0.1051 & 0.2286 & 0.3838 & 0.2422 & 0.1072 \\ 0.2082 & 0.2078 & 0.2025 & 0.3609 & 0.2909 & 0.1607 \\ 0.2106 & 0.1590 & 0.2255 & 0.3682 & 0.2725 & 0.0993 \\ 0.3683 & 0.5577 & 0.6036 & 0.4940 & 0.4364 & 0.4376 \\ 0.6980 & 0.6346 & 0.4638 & 0.5686 & 0.5917 & 0.4232 \\ 0.4397 & 0.4938 & 0.6957 & 0.4837 & 0.4264 & 0.5897 \end{pmatrix}$$

$$Z_s = \begin{pmatrix} 0.4660 & 0.5965 & 0.5130 & 0.5436 & 0.4931 & 0.6543 \\ 0.5820 & 0.5816 & 0.4678 & 0.6013 & 0.5659 & 0.4394 \\ 0.3961 & 0.5589 & 0.5295 & 0.5727 & 0.5180 & 0.3092 \\ 0.2256 & 0.1046 & 0.2294 & 0.3791 & 0.2426 & 0.1075 \\ 0.2075 & 0.2077 & 0.2030 & 0.3614 & 0.2919 & 0.1609 \\ 0.2117 & 0.1594 & 0.2265 & 0.3684 & 0.2718 & 0.0992 \\ 0.3686 & 0.5580 & 0.6029 & 0.4933 & 0.4360 & 0.4370 \\ 0.6984 & 0.6343 & 0.4644 & 0.5689 & 0.5920 & 0.4235 \\ 0.4387 & 0.4902 & 0.6980 & 0.4838 & 0.4297 & 0.5920 \end{pmatrix}$$

A partir de ces deux matrices, nous calculons toutes les composantes nécessaires pour déterminer la fonction d'optimisation donnée par la relation (4.44)
Le vecteur caractéristique moyen cover est donné par l'expression suivante :

$$M_c(i) = \frac{1}{6} \sum_{j=1}^{6} Z_c(i,j) \qquad (4.48)$$
$$i = 1,\ldots, 9$$

Le vecteur moyen cover obtenu est alors :

$$M_C = \begin{pmatrix} 0.5444 \\ 0.5368 \\ 0.4827 \\ 0.2154 \\ 0.2385 \\ 0.2225 \\ 0.4829 \\ 0.5633 \\ 0.5215 \end{pmatrix}$$

Le vecteur caractéristique moyen stégo est calculé comme suit :

$$M_s(i) = \frac{1}{6} \sum_{j=1}^{6} Z_s(i,j) \qquad (4.49)$$
$$i = 1,\ldots, 9$$

Le vecteur moyen de la matrice stégo est alors :

$$M_s = \begin{pmatrix} 0.5444 \\ 0.5397 \\ 0.4807 \\ 0.2148 \\ 0.2387 \\ 0.2228 \\ 0.4827 \\ 0.5636 \\ 0.5221 \end{pmatrix}$$

La matrice de variance des vecteurs caractéristiques de type cover est donnée par l'équation :

$$S_c(i,j) = \sum_{j=1}^{6} \sum_{i=1}^{9} [Z_c(i,j) - M_c(i)] \; [Z_c(i,j) - M_c(i)]^t \qquad (4.50)$$

Les valeurs du contenu de la matrice (de dimension 9 x 9) sont données ci-dessous :

$S_c \;=$

$$\begin{pmatrix} 0.0245 & -0.0109 & -0.0110 & -0.0201 & -0.0092 & -0.0184 & 0.0065 & -0.0205 & 0.0118 \\ -0.0109 & 0.0227 & 0.0207 & 0.0160 & 0.0163 & 0.0186 & -0.0063 & 0.0307 & -0.0286 \\ -0.0110 & 0.0207 & 0.0560 & 0.0253 & 0.0254 & 0.0323 & 0.0279 & 0.0161 & -0.0076 \\ -0.0201 & 0.0160 & 0.0253 & 0.0532 & 0.0330 & 0.0461 & -0.0023 & 0.0090 & -0.0117 \\ -0.0092 & 0.0163 & 0.0254 & 0.0330 & 0.0269 & 0.0322 & -0.0007 & 0.0104 & -0.0179 \\ -0.0184 & 0.0186 & 0.0323 & 0.0461 & 0.0322 & 0.0431 & 0.0018 & 0.0130 & -0.0154 \\ 0.0065 & -0.0063 & 0.0279 & -0.0023 & -0.0007 & 0.0018 & 0.0376 & -0.0170 & 0.0292 \\ -0.0205 & 0.0307 & 0.0161 & 0.0090 & 0.0104 & 0.0130 & -0.0170 & 0.0536 & -0.0428 \\ 0.0118 & -0.0286 & -0.0076 & -0.0117 & -0.0179 & -0.0154 & 0.0292 & -0.0428 & 0.0529 \end{pmatrix}$$

De la même manière, la matrice de variance des vecteurs caractéristiques de type stégo est donnée par l'équation suivante :

$$S_s(i,j) = \sum_{j=1}^{6} \sum_{i=1}^{7} [Z_s(i,j) - M_s(i)] \; [Z_s(i,j) - M_s(i)]^t \qquad (4.51)$$

La matrice obtenue est :

$S_s =$

$$\begin{pmatrix} 0.0246 & -0.0113 & -0.0117 & -0.0204 & -0.0094 & -0.0188 & 0.0064 & -0.0206 & 0.0118 \\ -0.0113 & 0.0232 & 0.0200 & 0.0164 & 0.0167 & 0.0193 & -0.0063 & 0.0309 & -0.0293 \\ -0.0117 & 0.0200 & 0.0549 & 0.0257 & 0.0251 & 0.0326 & 0.0285 & 0.0148 & -0.0058 \\ -0.0204 & 0.0164 & 0.0257 & 0.0518 & 0.0325 & 0.0455 & -0.0024 & 0.0089 & -0.0112 \\ -0.0094 & 0.0167 & 0.0251 & 0.0325 & 0.0271 & 0.0323 & -0.0007 & 0.0102 & -0.0177 \\ -0.0188 & 0.0193 & 0.0326 & 0.0455 & 0.0323 & 0.0430 & 0.0018 & 0.0131 & -0.0152 \\ 0.0064 & -0.0063 & 0.0285 & -0.0024 & -0.0007 & 0.0018 & 0.0375 & -0.0168 & 0.0290 \\ -0.0206 & 0.0309 & 0.0148 & 0.0089 & 0.0102 & 0.0131 & -0.0168 & 0.0535 & -0.0436 \\ 0.0118 & -0.0293 & -0.0058 & -0.0112 & -0.0177 & -0.0152 & 0.0290 & -0.0436 & 0.0538 \end{pmatrix}$$

Formons maintenant la matrice de dispersion intra-classes avant projection

$$S_w = (S_c + S_s) \tag{4.52}$$

alors :

$S_W =$

$$\begin{pmatrix} 0.0491 & -0.0222 & -0.0226 & -0.0405 & -0.0186 & -0.0372 & 0.0130 & -0.0411 & 0.0236 \\ -0.0222 & 0.0459 & 0.0407 & 0.0324 & 0.0330 & 0.0379 & -0.0126 & 0.0616 & -0.0579 \\ -0.0226 & 0.0407 & 0.1110 & 0.0511 & 0.0504 & 0.0649 & 0.0564 & 0.0310 & -0.0134 \\ -0.0405 & 0.0324 & 0.0511 & 0.1050 & 0.0656 & 0.0916 & -0.0047 & 0.0179 & -0.0229 \\ -0.0186 & 0.0330 & 0.0504 & 0.0656 & 0.0541 & 0.0645 & -0.0014 & 0.0206 & -0.0356 \\ -0.0372 & 0.0379 & 0.0649 & 0.0910 & 0.0645 & 0.08610 & 0.0037 & 0.0262 & -0.0306 \\ 0.0130 & -0.0126 & 0.0564 & -0.0047 & -0.0014 & 0.0037 & 0.0752 & -0.0339 & 0.0582 \\ -0.0411 & 0.0616 & 0.0310 & 0.0179 & 0.0206 & 0.0262 & -0.0339 & 0.1070 & -0.0864 \\ 0.0236 & -0.0579 & -0.0134 & -0.0229 & -0.0356 & -0.0306 & 0.0582 & -0.0864 & 0.1068 \end{pmatrix}$$

La solution W obtenue de la résolution de l'équation $W_{opt} = S_W^{-1}(M_c - M_s)$ donne le vecteur suivant :

$S_W^{-1} = 10^6 \times$

$$\begin{pmatrix} 0.2351 & -0.4687 & -0.0519 & -0.4344 & -0.0410 & 0.7882 & -0.0981 & 0.3414 & 0.1358 \\ -0.4687 & 1.0011 & 0.1064 & 0.8439 & 0.0088 & -1.5120 & 0.1812 & -0.7226 & -0.2724 \\ -0.0519 & 0.1064 & 0.0240 & 0.1144 & -0.0019 & -0.1955 & 0.0154 & -0.0824 & -0.0349 \\ -0.4344 & 0.8439 & 0.1144 & 0.8621 & 0.0565 & -1.5196 & 0.1925 & -0.6368 & -0.2833 \\ -0.0410 & 0.0088 & -0.0019 & 0.0565 & 0.1402 & -0.1811 & 0.0135 & 0.0077 & 0.0194 \\ 0.7882 & -1.5120 & -0.1955 & -1.5196 & -0.1811 & 2.7454 & -0.3342 & 1.1197 & 0.4689 \\ -0.0981 & 0.1812 & 0.0154 & 0.1925 & 0.0135 & -0.3342 & 0.0575 & -0.1397 & -0.0723 \\ 0.3414 & -0.7226 & -0.0824 & 0.6368 & 0.0077 & 1.1197 & -0.1397 & 0.5311 & 0.2147 \\ 0.1358 & -0.2724 & -0.0349 & -0.2833 & 0.0194 & 0.4689 & -0.0723 & 0.2147 & 0.1109 \end{pmatrix}$$

$$W_{opt} = 10^3 \times \begin{pmatrix} 0.5324 \\ -1.2499 \\ -0.0783 \\ -0.8030 \\ 0.0191 \\ 1.5142 \\ -0.1718 \\ 0.8477 \\ 0.2417 \end{pmatrix}$$

La matrice de test Z utilisée comprend 4 vecteurs caractéristiques cover, suivie de 4 vecteurs caractéristiques stégo :

$$Z = \begin{pmatrix} 0.4661 & 0.4880 & 0.5392 & 0.5285 & 0.4663 & 0.4879 & 0.5404 & 0.5285 \\ 0.2799 & 0.5572 & 0.3621 & 0.3840 & 0.2825 & 0.5600 & 0.3649 & 0.3855 \\ 0.6084 & 0.5799 & 0.5443 & 0.7225 & 0.6060 & 0.5877 & 0.5525 & 0.7328 \\ 0.0500 & 0.1115 & 0.1474 & 0.1389 & 0.0498 & 0.1103 & 0.1479 & 0.1400 \\ 0.1741 & 0.1507 & 0.2619 & 0.1998 & 0.1736 & 0.1512 & 0.2615 & 0.1996 \\ 0.0973 & 0.1169 & 0.2182 & 0.1184 & 0.0970 & 0.1170 & 0.2185 & 0.1173 \\ 0.6409 & 0.4450 & 0.6710 & 0.3786 & 0.6403 & 0.4456 & 0.6710 & 0.3783 \\ 0.2277 & 0.6148 & 0.3614 & 0.3899 & 0.2276 & 0.6145 & 0.3614 & 0.3901 \\ 0.6762 & 0.4418 & 0.4830 & 0.3178 & 0.6785 & 0.4390 & 0.4811 & 0.3190 \end{pmatrix}$$

Les différentes valeurs des composantes nécessaires pour le calcul de Z_p sont données ci-dessous :

$M_{cp} = 270.1863$

$M_{sp} = 268.1768$

$b = -269.1816$

Le vecteur Z_p résultant est :

$Z_p = (-61.5650 \ -109.4073 \ \ 47.5604 \ -110.5693 \ \ -64.3068 \ -113.3942 \ \ 43.6474 \ -115.0450)$

Donc le résultat de la classification est : [2 2 1 2 2 2 1 2]

Ci-dessous, avant d'entamer le processus de quantification expérimentale des performances de la classification, nous rappelons la structure de chaque vecteur caractéristique spécifique à chaque méthode de stéganalyse.

Pour la méthode de Farid :

$Z = [\ \mu_{H_1}, \mu_{V_1}, \mu_{D_1} \ | \ \sigma_{H_1}, \sigma_{V_1}, \sigma_{D_1} | \xi_{H_1}, \xi_{V_1}, \xi_{D_1} \ | \ \kappa_{H_1}, \kappa_{V_1}, \kappa_{D_1} | \ \mu_{H_2}, \mu_{V_2}, \mu_{D_2} \ | \ \sigma_{H_2}, \sigma_{V_2}, \sigma_{D_2} | \xi_{H_2}, \xi_{V_2}, \xi_{D_2} \ | \ \kappa_{H_2}, \kappa_{V_2}, \kappa_{D_2} | \ \mu_{H_3}, \mu_{V_3}, \mu_{D_3} \ | \ \sigma_{H_3}, \sigma_{V_3}, \sigma_{D_3} | \xi_{H_3}, \xi_{V_3}, \xi_{D_3} \ | \ \kappa_{H_3}, \kappa_{V_3}, \kappa_{D_3} | \ \mu^p_{\varepsilon_{H_1}}, \mu^p_{\varepsilon_{V_1}}, \mu^p_{\varepsilon_{D_1}} | \sigma^p_{\varepsilon_{H_1}}, \sigma^p_{\varepsilon_{V_1}}, \sigma^p_{\varepsilon_{D_1}} | \xi^p_{\xi_{H_1}}, \xi^p_{\xi_{V_1}}, \xi^p_{\xi_{D_1}} | \ \kappa^p_{\varepsilon_{H_1}}, \kappa^p_{\varepsilon_{V_1}}, \kappa^p_{\varepsilon_{D_1}} | \ \mu^p_{\varepsilon_{H_2}}, \mu^p_{\varepsilon_{V_2}}, \mu^p_{\varepsilon_{D_2}} | \ \sigma^p_{\varepsilon_{H_2}}, \sigma^p_{\varepsilon_{V_2}}, \sigma^p_{\varepsilon_{D_2}} | \kappa^p_{\varepsilon_{H_2}}, \kappa^p_{\varepsilon_{V_2}}, \kappa^p_{\varepsilon_{D_2}} | \ \mu^p_{\varepsilon_{H_3}}, \mu^p_{\varepsilon_{V_3}}, \mu^p_{\varepsilon_{D_3}} | \ \sigma^p_{\varepsilon_{H_3}}, \sigma^p_{\varepsilon_{V_3}}, \sigma^p_{\varepsilon_{D_3}} | \xi^p_{\varepsilon_{H_3}}, \xi^p_{\varepsilon_{V_3}}, \xi^p_{\varepsilon_{D_3}} | \ \kappa^p_{\varepsilon_{H_3}}, \kappa^p_{\varepsilon_{V_3}}, \kappa^p_{\varepsilon_{D_3}}].$

Pour la méthode de Shi :

$Z = [M_I^1, M_I^2, M_I^3 | M_{A_1}^1, M_{A_1}^2, M_{A_1}^3 | M_{H_1}^1, M_{H_1}^2, M_{H_1}^3 | M_{V_1}^1, M_{V_1}^2, M_{V_1}^3 | M_{D_1}^1, M_{D_1}^2, M_{D_1}^3$
$| M_{A_2}^1, M_{A_2}^2, M_{A_2}^3 | M_{H_2}^1, M_{H_2}^2, M_{H_2}^3 | M_{V_2}^1, M_{V_2}^2, M_{V_2}^3 | M_{D_2}^1, M_{D_2}^2, M_{D_2}^3 | M_{A_3}^1, M_{A_3}^2, M_{A_3}^3 | \quad M_{H_3}^1, M_{H_3}^2, M_{H_3}^3$
$| M_{V_3}^1, M_{V_3}^2, M_{V_3}^3 | M_{D_3}^1, M_{D_3}^2, M_{D_3}^3 | M_{\epsilon_I}^{p1}, M_{\varepsilon_I}^{p2}, M_{\varepsilon_I}^{p3} | M_{A_1}^1, M_{A_1}^2, M_{A_1}^3 | M_{H_1}^1, M_{H_1}^2, M_{H_1}^3 | M_{V_1}^1, M_{V_1}^2, M_{V_1}^3$
$| M_{D_1}^1, M_{D_1}^2, M_{D_1}^3 | M_{A_2}^1, M_{A_2}^2, M_{A_2}^3 | M_{H_2}^1, M_{H_2}^2, M_{H_2}^3 | M_{V_2}^1, M_{V_2}^2, M_{V_2}^3 | M_{D_2}^1, M_{D_2}^2, M_{D_2}^3 \quad | \quad M_{A_3}^1, M_{A_3}^2, M_{A_3}^3 |$
$M_{H_3}^1, M_{H_3}^2, M_{H_3}^3 | M_{V_3}^1, M_{V_3}^2, M_{V_3}^3 | M_{D_3}^1, M_{D_3}^2, M_{D_3}^3].$

Pour la méthode de Wang :

$Z = [M_I^i | M_{A_1}^i | M_{H_1}^i | M_{V_1}^i | M_{D_1}^i | M_{A_2}^i | M_{H_2}^i | M_{V_2}^i | M_{D_2}^i | M_{A_3}^i | M_{H_3}^i | M_{V_3}^i | M_{D_3}^i | M_{A_2'}^i | M_{H_2'}^i | M_{V_2'}^i | M_{D_2'}^i |$
$m_{\varepsilon H_m}^i | m_{\varepsilon V_m}^i | m_{\varepsilon D_m}^i$

4.4.3.3 Quantification expérimentale des performances de la classification :

Nous avons utilisé pour nos expérimentations la base de données UCID (Uncompressed Color Image Database : http ://vision.doc.ntu.ac.uk/) contenant 1338 images [Schaefer et Stich, 2004]. Sur ces images, nous avons d'abord réalisé la stéganographie (utilisant les deux méthodes du chapitre 2) avec les taux d'insertion de message suivants : 5%, 10%, 20%, 30%. L'extraction des paramètres se fait pour les deux catégories d' images (1338 images originales, 1338 images stégos).
Dans la table 4.4, nous donnons, pour plusieurs cas de figures, le nombre d'images utilisé dans le processus d'apprentissage et dans le processus de classification correspondant du classifieur FLD. Rappelons que le nombre total d'images est de 1338.

Processus d'apprentissage		Processus de classification	
cover	stégo	cover	stégo
1238	1238	100	100
1188	1188	150	150
1138	1138	200	200
1088	1088	250	250
1038	1038	300	300
988	988	350	350
938	938	400	400
888	888	450	450
838	838	500	500
788	788	550	550

Table 4.4 : Données expérimentales

Quantification des performances de la classification et donc des méthodes de la stéganalyse

L'évaluation de la classification et donc, la stéganalyse (et indirectement l'efficacite d'insertion) est réalisée en calculant les paramètres sensibilité, spécificité, et précision de la matrice de confusion et le coefficient Kappa, résumés dans la table 4.5, ci-dessous, et ceci pour les différents taux d'insertion de chaque algorithme, de chaque méthode de séganalyse.

	H_0 : images stégos	H_1 : images originales		
H_0	VP : P_{11}	FP : P_{12}	Pr	$P_{1.}$
H_1	FN : P_{21}	VN : P_{22}	VPN	$P_{2.}$
	Se	Sp	Ex	
	$P_{.1}$	$P_{.2}$		
Kappa				

Table 4.5 : Paramètres utilisés dans l'évaluation de la classification et de la stéganalyse

Stéganalyse de Farid

Ci-dessous, nous donnons dans les tables 4.6, 4.7, 4.8 et 4.9, les résultats d'évaluation de la classification (stéganalyse) des données selon Farid [Farid, 2002] des différentes méthodes d'insertion EALSBMR, EEALSBMR, AELSB et EAELSB, et pour les différents taux d'insertion 5%, 10%, 20% et 30%, de chacune des méthodes.

Les résultats de la classification de la méthode EALSBMR, pour tous les taux d'insertion montrent que la classification n'est pas bonne. En effet, les valeurs Se, Sp et Pr varient autour de 50%, ces valeurs sont donc des valeurs non informatives et ne donnent aucune idée sur la nature des données. Par ailleurs, la valeur obtenue du coefficient Kappa (inférieur à 0.2) confirme ce résultat. La stéganalyse par Farid n'est donc pas efficace ici, de même que les méthodes EEALSBMR et AELSB (voir tables 4.7 et 4.8). Les méthodes d'insertion (stéganographie) citées sont alors efficaces, vis-à-vis de cette technique de stéganalyse.

Pour la méthode d'insertion EAELSB, dans le cas des taux d'insertion 20% et 30%, les valeurs Se, Sp et Pr deviennent assez bonnes (plus grande que 70%), de même pour le coefficient Kappa (plus grand que 0.4), et de même pour la stéganalyse. Par conséquent, cette méthode d'insertion n'est pas assez efficace pour ces taux d'insertion.

5%	H0 : images stégos	H1 : images originales		
H0	0.275	0.267	Pr=0.507	$P_{1.}$=0.542
H1	0.224	0.232	VPN=0.508	$P_{2.}$=0.456
	Se=0.550	Sp=0.534	Ex=0.508	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.01625				
10%	H0 : images stégos	H1 : images originales		
H0	0.256	0.253	Pr=0.503	$P_{1.}$=0.509
H1	0.243	0.246	VPN=0.503	$P_{2.}$=0.489
	Se=0.513	Sp=0.493	Ex=0.5037	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.007				
20%	H0 : images stégos	H1 : images originales		
H0	0.265375	0.236375	Pr= 0.5289	$P_{1.}$=0.5017
H1	0.234625	0.263625	VPN=0.5291	$P_{2.}$= 0.4983
	Se= 0.5308	Sp=0.5273	Ex=0.529	
	$P_{.1}$=0.5	$P_{.2}$=0.5		1
Kappa =0.058				
30%	H0 : images stégos	H1 : images originales		
H0	0.2928	0.2254	Pr= 0.5650	$P_{1.}$= 0.5181
H1	0.2073	0.2746	VPN= 0.5699	$P_{2.}$= 0.4819
	Se= 0.5855	Sp= 0.5493	Ex=0.5674	
	$P_{.1}$=0.5	$P_{.2}$=0.5		1
Kappa = 0.1347				

Table 4.6 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EALSBMR

5%	H0 : images stégos	H1 : images originales		
H0	0.2744	0.2714	Pr= 0.5027	$P_{1.}$= 0.5458
H1	0.2256	0.2286	VPN=0.5033	$P_{2.}$= 0.4542
	Se= 0.5487	Sp= 0.4572	Ex=0.5030	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0060				
10%	H0 : images stégos	H1 : images originales		
H0	0.2690	0.2645	Pr= 0.5042	$P_{1.}$= 0.5335
H1	0.2310	0.2355	VPN=0.5048	$P_{2.}$= 0.4665
	Se= 0.5380	Sp= 0.4710	Ex= 0.5045	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0090				
20%	H0 : images stégos	H1 : images originales		

H0	0.2745	0.2459	Pr=0.5275	$P_{1.}$= 0.5204
H1	0.2255	0.2541	VPN=0.5298	$P_{2.}$= 0.4796
	Se= 0.5490	Sp= 0.5082	Ex= 0.5286	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0572				
30%	H0 : images stégos	H1 : images originales		
H0	0.2944	0.2248	Pr= 0.5671	$P_{1.}$= 0.5191
H1	0.2056	0.2752	VPN=0.5724	$P_{2.}$= 0.4809
	Se= 0.5888	Sp= 0.5505	Ex=0.5696	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.1393				

Table 4.7 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EEALSBMR

5%	H0 : images stégos	H1 : images originales		
H0	0.2599	0.2460	Pr=0.5137	$P_{1.}$= 0.5059
H1	0.2401	0.2540	VPN=0.5140	$P_{2.}$= 0.4941
	Se= 0.5198	Sp= 0.5080	Ex=0.5139	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0278				
10%	H0 : images stégos	H1 : images originales		
H0	0.2815	0.2568	Pr= 0.5230	$P_{1.}$= 0.5383
H1	0.2185	0.2432	VPN=0.5268	$P_{2.}$= 0.4617
	Se= 0.5630	Sp= 0.4865	Ex=0.5247	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0495				
20%	H0 : images stégos	H1 : images originales		
H0	0.2853	0.2374	Pr= 0.5458	$P_{1.}$= 0.5226
H1	0.2148	0.2626	VPN=0.5501	$P_{2.}$= 0.4774
	Se= 0.5705	Sp=0.5252	Ex=0.5479	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0958				
30%	H0 : images stégos	H1 : images originales		
H0	0.2977	0.2274	Pr=0.5670	$P_{1.}$= 0.5251
H1	0.2023	0.2726	VPN=0.5741	$P_{2.}$= 0.4749
	Se=0.5955	Sp= 0.5453	Ex=0.5704	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.1407				

Table 4.8 : Résultats d'évaluation de la classification (stéganalyse) de la méthode AELSB

5%	H0 : images stégos	H1 : images originales		
H0	0.3041	0.2114	Pr=0.5900	$P_{1.}$= 0.5155
H1	0.1959	0.2886	VPN=0.5957	$P_{2.}$= 0.4845
	Se= 0.6083	Sp=0.5773	Ex=0.5928	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.1855				
10%	H0 : images stégos	H1 : images originales		
H0	0.3280	0.1698	Pr=0.6590	$P_{1.}$= 0.4977
H1	0.1720	0.3302	VPN=0.6575	$P_{2.}$= 0.5022
	Se=0.6560	Sp= 0.6605	Ex=0.6583	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.3165				
20%	H0 : images stégos	H1 : images originales		
H0	0.3750	0.1343	Pr= 0.7364	$P_{1.}$= 0.5092
H1	0.1250	0.3658	VPN=0.7453	$P_{2.}$= 0.4908
	Se= 0.7500	Sp=0.7315	Ex= 0.7408	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.4815				
30%	H0 : images stégos	H1 : images originales		
H0	0.3575	0.1364	Pr=0.7239	$P_{1.}$= 0.4939
H1	0.1425	0.3636	VPN=0.7184	$P_{2.}$= 0.5061
	Se= 0.7150	Sp=0.7272	Ex= 0.7211	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.4423				

Table 4.9 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EAELSB

Stéganalyse de Shi

Les résultats d'évaluation de la classification (stéganalyse) des données selon Shi [Shi et al., 2005] des méthodes d'insertion EALSBMR, EEALSBMR, et AELSB (voir respectivement tables 4.10, 4.11 et 4.12) sont assez similaires à ceux obtenus par Farid, pour les taux d'insertion 5%, 10%, et 20%. En effet, dans ces conditions, les valeurs *Se, Sp* et *Pr* varient autour de 50% (valeurs non informatives) et le coefficient Kappa est plus petit que 0.2. Ces résultats ne permettent pas de renseigner sur la présence d'un message secret caché. La stéganalyse n'est donc pas efficace, et par conséquent la méthode d'insertion est pertinente.
Pour le taux d'insertion de 30 % les valeurs de *Se*, *Sp*, *Pr* (>85%) et Kappa (>0.8) sont très bonnes et peuvent donc nous informer facilement sur la présence d'informations cachées.
Pour la méthode EAELSB, les résultats obtenus (voir table 4.13) montrent que pour le taux d'insertion de 5% seulement, la stéganalyse de Shi n'est pas efficace, autrement, elle est opérante pour les autres taux d'insertion 10%, 20% et 30%.

A la vue des résultats obtenus précédemment, nous pouvons déduire que les vecteurs caractéristiques basés CF (Characteristic functions) sont plus informatifs et plus sélectifs entre les images originales et les images stégos que les vecteurs caractéristiques basés PDF (Probability Density Function), surtout pour un taux d'insertion important.

5%	H0 : images stégos	H1 : images originales		
H0	0.2705	0.2467	Pr= 0.5230	$P_{1.}$= 0.5172
H1	0.2295	0.2533	VPN=0.5247	$P_{2.}$= 0.4828
	Se= 0.5410	Sp=0.5067	Ex=0.5238	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0477				
10%	H0 : images stégos	H1 : images originales		
H0	0.2545	0.2407	Pr=0.5140	$P_{1.}$= 0.4952
H1	0.2455	0.2593	VPN=0.5137	$P_{2.}$= 0.5048
	Se= 0.5090	Sp=0.5187	Ex=0.5138	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.0277				
20%	H0 : images stégos	H1 : images originales		
H0	0.3287	0.1880	Pr= 0.6361	$P_{1.}$= 0.5167
H1	0.1713	0.3120	VPN=0.6455	$P_{2.}$= 0.4833
	Se=0.6573	Sp= 0.6240	Ex=0.6407	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.2813				
30%	H0 : images stégos	H1 : images originales		
H0	0.4603	0.0592	Pr= 0.8861	$P_{1.}$= 0.5195
H1	0.0397	0.4408	VPN=0.9174	$P_{2.}$= 0.4805
	Se= 0.9207	Sp=0.8817	Ex=0.9012	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.8023				

Table 4.10 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EALSBMR

5%	H0 : images stégos	H1 : images originales		
H0	0.2612	0.2405	Pr=0.5207	$P_{1.}$= 0.5017
H1	0.2387	0.2595	VPN=0.5208	$P_{2.}$= 0.4983
	Se= 0.5225	Sp= 0.5190	Ex=0.5208	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0415				
10%	H0 : images stégos	H1 : images originales		
H0	0.2504	0.2448	Pr= 0.5057	$P_{1.}$= 0.4951
H1	0.2496	0.2552	VPN=0.5056	$P_{2.}$= 0.5049
	Se= 0.5008	Sp=0.5105	Ex=0.5056	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0112				
20%	H0 : images stégos	H1 : images originales		
H0	0.3191	0.1946	Pr= 0.6212	$P_{1.}$= 0.5138
H1	0.1809	0.3054	VPN=0.6280	$P_{2.}$= 0.4863
	Se= 0.6382	Sp=0.6108	Ex= 0.6245	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.2490				
30%	H0 : images stégos	H1 : images originales		
H0	0.4567	0.0585	Pr= 0.8865	$P_{1.}$= 0.5152
H1	0.0433	0.4415	VPN=0.9108	$P_{2.}$= 0.4848
	Se=0.9135	Sp= 0.8830	Ex=0.8982	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.7965				

Table 4.11 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EEALSBMR

5%	H0 : images stégos	H1 : images originales		
H0	0.2697	0.2200	Pr= 0.5508	$P_{1.}$= 0.4898
H1	0.2302	0.2800	VPN=0.5488	$P_{2.}$= 0.5102
	Se=0.5395	Sp=0.5600	Ex=0.5497	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0995				
10%	H0 : images stégos	H1 : images originales		
H0	0.2911	0.2035	Pr= 0.5886	$P_{1.}$= 0.4946
H1	0.2089	0.2965	VPN=0.5867	$P_{2.}$= 0.5054
	Se=0.5823	Sp= 0.5930	Ex=0.5876	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.1753				
20%	H0 : images stégos	H1 : images originales		

H0	0.3347	0.1620	Pr=0.6739	$P_{1.}$= 0.4967
H1	0.1652	0.3380	VPN=0.6716	$P_{2.}$= 0.5032
	Se= 0.6695	Sp= 0.6760	Ex= 0.6728	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.3455				
30%	H0 : images stégos	H1 : images originales		
H0	0.4649	0.0352	Pr= 0.9295	$P_{1.}$= 0.5001
H1	0.0351	0.4648	VPN=0.9297	$P_{2.}$= 0.4999
	Se= 0.9297	Sp= 0.9295	Ex= 0.9296	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.8592				

Table 4.12 : Résultats d'évaluation de la classification (stéganalyse) de la méthode AELSB

5%	H0 : images stégos	H1 : images originales		
H0	0.3469	0.1604	Pr= 0.6838	$P_{1.}$= 0.5072
H1	0.1531	0.3396	VPN=0.6892	$P_{2.}$= 0.4927
	Se= 0.6937	Sp= 0.6793	Ex=0.6865	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.3730				
10%	H0 : images stégos	H1 : images originales		
H0	0.3883	0.1314	Pr= 0.7472	$P_{1.}$= 0.5196
H1	0.1118	0.3686	VPN=0.7674	$P_{2.}$= 0.4804
	Se=0.7765	Sp=0.7372	Ex= 0.7569	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.5137				
20%	H0 : images stégos	H1 : images originales		
H0	0.4141	0.0940	Pr= 0.8150	$P_{1.}$= 0.5081
H1	0.0859	0.4060	VPN=0.8254	$P_{2.}$= 0.4919
	Se=0.8283	Sp=0.8120	Ex=0.8201	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.6402				
30%	H0 : images stégos	H1 : images originales		
H0	0.4717	0.0223	Pr= 0.9550	$P_{1.}$= 0.4940
H1	0.0283	0.4778	VPN=0.9442	$P_{2.}$= 0.5060
	Se= 0.9435	Sp= 0.9555	Ex= 0.9495	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.8990				

Table 4.13 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EAELSB

Stéganalyse de Wang

Les résultats d'évaluation de la classification (stéganalyse) des données selon Wang et al. [Wang et Moulin, 2007] des méthodes d'insertion EALSBMR, EEALSBMR, voir tables 4.14, et 4.15, sont assez similaires à ceux obtenus par Farid, pour les taux d'insertion 5%, 10%, et 20%. Pour la méthode AELSB, nous obtenons les mêmes résultats que Farid (voir table 4.16) pour les taux d'insertion de 5% et 10% seulement pour le taux d'insertion de 30 %, les valeurs de *Se, Sp, Pr* (>70%) et Kappa (>0.4) sont seulement assez bonnes, et l'efficacité de la détection d'informations cachées est moyenne.
Pour AE-LSB, la classification est bonne pour le taux d'insertion de 20%, et très bonne pour le taux d'insertion de 30% (Kappa > 0.8).
Pour EAELSB, les résultats obtenus par les différents paramètres (voir table 4.17) montrent que la classification (stéganalyse) est bonne pour les taux d'insertion de 5 et 10 %, et elle est très bonne pour les taux d'insertion de 20 et 30%.

5%	H0 : images stégos	H1 : images originales		
H0	0.2507	0.2463	Pr=0.5045	$P_{1.}$= 0.4970
H1	0.2493	0.2537	VPN=0.5045	$P_{2.}$= 0.5030
	Se= 0.5015	Sp=0.5075	Ex=0.5045	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0090				
10%	H0 : images stégos	H1 : images originales		
H0	0.2500	0.2306	Pr=0.5202	$P_{1.}$= 0.4806
H1	0.2500	0.2694	VPN=0.5187	$P_{2.}$= 0.5194
	Se= 0.5000	Sp= 0.5387	Ex=0.5194	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0387				
20%	H0 : images stégos	H1 : images originales		
H0	0.2945	0.1925	Pr= 0.6047	$P_{1.}$= 0.4870
H1	0.2055	0.3075	VPN=0.5994	$P_{2.}$= 0.5130
	Se=0.5890	Sp= 0.6150	Ex=0.6020	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.2040				
30%	H0 : images stégos	H1 : images originales		
H0	0.3655	0.1357	Pr= 0.7292	$P_{1.}$= 0.5012
H1	0.1345	0.3642	VPN=0.7303	$P_{2.}$= 0.4988
	Se= 0.7310	Sp= 0.7285	Ex=0.7297	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.4595				

Table 4.14 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EALSBMR

5%	H0 : images stégos	H1 : images originales		
H0	0.2489	0.2476	Pr=0.5013	$P_{1.}$= 0.4965
H1	0.2511	0.2524	VPN=0.5012	$P_{2.}$= 0.5035
	Se= 0.4977	Sp= 0.5048	Ex= 0.5012	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.0025				
10%	H0 : images stégos	H1 : images originales		
H0	0.2559	0.2299	Pr=0.5268	$P_{1.}$= 0.4858
H1	0.2441	0.2701	VPN=0.5253	$P_{2.}$= 0.5142
	Se= 0.5117	Sp=0.5403	Ex=0.5260	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.0520				
20%	H0 : images stégos	H1 : images originales		
H0	0.2990	0.1985	Pr=0.6010	$P_{1.}$= 0.4975
H1	0.2010	0.3015	VPN=0.6000	$P_{2.}$= 0.5025
	Se=0.5980	Sp= 0.6030	Ex=0.6005	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.2010				
30%	H0 : images stégos	H1 : images originales		
H0	0.3648	0.1314	Pr=0.7352	$P_{1.}$= 0.4961
H1	0.1353	0.3686	VPN=0.7316	$P_{2.}$= 0.5039
	Se=0.7295	Sp=0.7372	Ex=0.7400	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.4668				

Table 4.15 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EEALSBMR

5%	H0 : images stégos	H1 : images originales		
H0	0.2803	0.2070	Pr=0.5752	$P_{1.}$= 0.4872
H1	0.2198	0.2930	VPN=0.5714	$P_{2.}$= 0.5128
	Se=0.5605	Sp= 0.5860	Ex=0.5800	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.1465				
10%	H0 : images stégos	H1 : images originales		
H0	0.3200	0.1705	Pr=0.6524	$P_{1.}$= 0.4905
H1	0.1800	0.3295	VPN=0.6467	$P_{2.}$= 0.5095
	Se=0.6400	Sp=0.6590	Ex=0.6400	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.2990				

20%	H0 : images stégos	H1 : images originales		
H0	0.3504	0.1223	Pr=0.7413	$P_{1.}$= 0.4726
H1	0.1496	0.3777	VPN=0.7163	$P_{2.}$= 0.5274
	Se=0.7007	Sp=0.7555	Ex=0.7200	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.4562				
30%	H0 : images stégos	H1 : images originales		
H0	0.4415	0.0198	Pr=0.9572	$P_{1.}$= 0.4612
H1	0.0585	0.4803	VPN=0.8914	$P_{2.}$= 0.5387
	Se=0.8830	Sp=0.9605	Ex=0.9200	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.8435				

Table 4.16 : Résultats d'évaluation de la classification (stéganalyse) de la méthode AELSB

5%	H0 : images stégos	H1 : images originales		
H0	0.3659	0.1457	Pr=0.7151	$P_{1.}$= 0.5116
H1	0.1341	.3542	VPN=0.7254	$P_{2.}$= 0.4884
	Se=0.7318	Sp=0.7085	Ex=0.7200	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.4402				
10%	H0 : images stégos	H1 : images originales		
H0	0.4309	0.0779	Pr=0.8469	$P_{1.}$= 0.5088
H1	0.0691	0.4221	VPN=0.8593	$P_{2.}$= 0.4913
	Se=0.8618	Sp=0.8442	Ex= 0.8600	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.7060				
20%	H0 : images stégos	H1 : images originales		
H0	0.4630	0.0298	Pr=0.9396	$P_{1.}$= 0.4927
H1	0.0370	0.4703	VPN=0.9271	$P_{2.}$= 0.5072
	Se=0.9260	Sp=0.9405	Ex= 0.9400	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa =0.8665				
30%	H0 : images stégos	H1 : images originales		
H0	0.4865	0.0075	Pr=0.9848	$P_{1.}$= 0.4940
H1	0.0135	0.4925	VPN=0.9733	$P_{2.}$= 0.5060
	Se=0.9730	Sp=0.9850	Ex=0.9800	
	$P_{.1}$=0.5	$P_{.2}$=0.5		
Kappa = 0.9580				

Table 4.17 : Résultats d'évaluation de la classification (stéganalyse) de la méthode EAELSB

4.5 Conclusion

Ce chapitre présente la stéganalyse (classification) des méthodes spatiales présentées dans le chapitre 2.
Un principe général d'un algorithme de stéganalyse universel est d'identifier, cherchant à extraire, des caractéristiques qui sont particulièrement sensibles à l'intégration de données. Ces caractéristiques peuvent capter toute variation résultante de l'intégration.
Les paramètres utilisés pour l'évaluation des résultats de la classification sont la sensibilité, la spécificité, la précision et le coefficient Kappa. Les valeurs de la sensibilité, la spécificité et la précision sont non informatives si elles varient autour de 50%. Elles deviennent informatives si leurs valeurs sont supérieures à 85 %.
Nous avons appliqué trois méthodes de séganalyse courantes : Farid, Shi et Wang
Pour la méthode de Farid, la stéganalyse n'est pas pertinente pour les méthodes de stéganographie EALSBMR, EEALSBMR, AELSB quel que soit le taux d'insertion. Pour la méthode EAELSB, la stéganalyse est moyennement efficace pour les taux 20% et 30%.
Pour la méthode de Shi, la stéganalyse n'est pas efficace pour les méthodes EALSBMR, EEALSBMR, et AELSB avec les taux d'insertion 5%, 10% et 20%, autrement elle est efficace (taux insertion =30%).
Pour la méthode EAELSB, la stéganalyse de Shi n'est pas pertinente pour le taux d'insertion de 5%, mais le devient pour des taux supérieurs.

Pour la méthode de Wang, la stéganalyse n'est pas concluante pour les méthodes EALSBMR, EEALSBMR avec les taux d'insertion 5%, 10% et 20%, mais l'est pour des taux supérieurs.
Pour AELSB, la stéganalyse est pertinente pour le taux d'insertion de 20% et très pertinente pour un taux d'insertion de 30% (Kappa > 0.8).
Pour EAELSB, la stéganalyse est efficace pour les taux d'insertion de 5 et 10 %, et très efficace pour les taux d'insertion de 20 et 30%.
D'après ces résultats, nous pouvons conclure que la méthode de stéganalyse de Wang est la plus efficace parmi les trois méthodes étudiées, cependant la méthode de Shi est la plus affirmative.

Liste des publications:

Article Journal accepté, à paraître:

- **D. Battikh**, S. El Assad, B. Bakhache, O. Deforges, and M. Khalil, "Chaos-based spatial steganography system for images", International Journal of Chaotic Computing (IJCC).

Publications Internationales

- R. L. Tataru, **D. Battikh**, S. El Assad, H. Noura, O. Deforges, " Enhanced Adaptive Data Hiding in Spatial LSB Domain by using Chaotic Sequences". IIH-MSP 2012: 85-88

- **D**. **Battikh,** S. El Assad, B. Bakhache, O. Deforges, and M. Khalil: "Enhancement of two spatial steganography algorithms by using a chaotic system : comparative analysis", IEEE, 8th International Conference for Internet Technology and Secured Transactions, ICITST-2013, London, UK, December, 2013, 6 pages.

- **D. Battikh**, S. El Assad, B. Bakhache, O. Deforges, and M. Khalil,"Amélioration de la sécurité des méthodes stéganographiques par le chaos",19th LAAS International Science Conference, 5–6 Avril 2013, LAY Beirut, Lebanon

- **D. Battikh**, S. El Assad, B. Bakhache, O. Deforges, and M. Khalil, "Steganalysis of a chaos-based steganographic method", the 10th international conference on communications, may 29-31, 2014, Bucarest, Romania

- **D. Battikh**, S. El Assad, B. Bakhache, O. Deforges, and M. Khalil, "Statistical Steganalysis of chaos based spatial steganography", 20th LAAS International Science Conference, 27-29 March 2014, Beirut, Lebanon

- M. Habib, B. Bakhache, **D. Battikh**, S. El Assad, "Enhancement using chaos of a Steganography method in DCT domain"', IEEE - Fifth International Conference on Digital Information and Communication Technology and its Applications (DICTAP2015), April 29 – May 01, 2015 , Lebanese University, Beirut, Lebanon

4.6 Références

[Antonini et al., 1992] Antonini, M., Barlaud, M., Mathieu, P., et Daubechies, I. (1992) Image codingusingwavelettransforms. IEEE Transactions on Image Processing, volume 1, numéro 2, pages 205–220.

[Chen et al., 2006] Chen, X., Wang, Y., Tan, T. et Gei, L. (2006). Blind image steganalysis based on statistical analysis of empirical matrix. IEEE ICPR.

[Chen et al., 2006] Chen, C., Shi, Y.Q., Chen, W. et Xuan, G. (2006). Statistical Moments Based Universal Steganalysis using JPEG 2-D Array and 2-D Characteristic Function. *IEEE International Conference on Image Processing*, pages 105–108.

[Fabbro-Peray, 2006-2007] Fabbro-Peray, P. (2006-2007). MB6 – Biostatistique – Statistique inférentiele variable qualitatives (3) – Mesure de la concordance coefficient Kappa - Nîmes

[Farid, 2002] Farid, H. (2002). Detecting hidden messages using higher-order statistical models.Proceedings of IEEE ICIP, volume 2, pages 905-908.

[Inoue et al., 1998] Inoue, H., Miy, A., Yamamoto, Katsura, T. (1998). A digital watermark based on the wavelet transform and its robustness on image compression. Proc. IEEE Int. Conf. on Image Processing, Chicago, octobre.volume 2, pages 391-395.

[Kohavi et Provost, 1998] Kohavi, R., et Provost, F. (1998). On Applied Research in Machine Learning, In Editorial for the Special Issue on Applications of Machine Learning and the Knowledge Discovery Process, Columbia University, New York, volume 30.

[Landis et Koch, 1977] Landis, J.R., Koch, G.G. (1977). The measurement of observer agreement for categorical data, Biometrics, Volume 33, pages 159–174.

[Lewis et Knowles, 1992] Lewis, A., Knowles, G. (1992). Image compression using the 2-D wavelet transform. IEEE Trans. on Image Process, volume 1, numéro 2, pages 244-250.

[Li et Wang, 2014] Li, C., Wang, B. (2014). Fisher Linear Discriminant Analysis.

[Lie et Lin, 2005] Lie, W.-N., et Lin, G.-S. (2005). A feature based classification for blind image steganalysis. IEEE Transaction Multimedia, volume 7, numéro 6, pages 1007- 1020.

[Lyu et Farid, 2002] Lyu, S. et Farid, H. (2002). Detecting hidden messages using higher-order statistics and support vector machines.Proceedings of 5th International Workshop on Information Hiding.

[Santos, 2013] Santos, F. (2013). Le kappa de Cohen : un outil de mesure de l'accord inter-juges sur des caractères qualitatifs.

[Schaefer et Stich, 2004] Schaefer, G., Stich, M. (2004). UCID : An uncompressed color image database, Proc. SPIE Electronic Imaging, Storage and Retrieval Methods and Applications for Multimedia, volume 5307, pages 472–480.

[Shi et al., 2005] Shi, Y.Q., Xuan, G., Zou, D., Gao, J., Yang, C., Zhang, Z., Chai, P., Chen, W. et Chen, C. (2005). Image steganalysis based on moments of characteristic functions using

wavelet decomposition. prediction error image and neural network, IEEE ICME.

[Wang et Moulin, 2007] Wang, Y. et Moulin, P. (2007). Optimized feature extraction for learning-based image steganalysis. IEEE Trans. on Information Forensics and Security, 2(1) :31{45, 2007.

[Xuan et al., 2006] Xuan, G., Shi, Y. Q., Huang, C., Fu, D., Zhu, X., Chai, P. et Gao, J. (2006). Steganalysis Using High-Dimensional Features Derived from Co-occurrence Matrix and Class-Wise Non-Principal Components Analysis (CNPCA). *5th InternationalWorkshop on Digital Watermarking*, 4283:49–60.

[Zettler et al., 1990] Zettler, W., Huffman, J., Linden. (1990). Application of Compactly Supported Wavelets to Image Compression. Procedings of SPIE, volume 1244, pages 150–160.

Conclusion générale

Ces travaux de thèse ont tout d'abord permis de présenter les méthodes stéganographiques spatiales et fréquentielles les plus utilisées dans la littérature, et de montrer leurs limitations. Différentes modifications et améliorations ont été proposées et réalisées. Plus particulièrement, afin d'assurer la sécurité du contenu du message secret, nous avons développé un système chaotique et nous l'avons implémenté dans les deux méthodes de stéganographie LSB adaptatives proposées. Le système chaotique permet ainsi de sécuriser les positions des bits informatifs. La stéganalyse universelle des méthodes spatiales proposées a été effectuée dans le but de prouver leur robustesse et leur résistance contre des attaques.

Dans le premier chapitre, nous avons introduit les concepts de base et les outils mathématiques nécessaires pour la compréhension des recherches menées dans cette thèse, comme les transformées DCT et DWT, les signaux chaotiques, la classification FLD et la stéganalyse universelle.

Dans le deuxième chapitre, nous avons étudié les deux méthodes stéganographiques spatiales LSB les plus répandues, l'AELSB et l'EALSBMR, qui tentent d'insérer le message secret dans les zones de bord de l'image. La sécurité du contenu du message, lors de sa détection par un adversaire, n'est pas assurée par la méthode AELSB, et est très faible pour la deuxième méthode. Afin de pallier à ce problème, nous avons adapté et ajouté aux deux méthodes un système chaotique robuste permettant une dissimulation quasi-chaotique des bits du message secret. Le système chaotique proposé consiste en un générateur de séquences chaotiques robustes et une carte Cat 2-D chaotique modifiée. Les performances obtenues par les deux techniques adaptatives améliorées (que nous appelons EAELSB et EEALSBMR), montrent visuellement et au travers des critères de qualité objective PSNR, IF et SSIM, un grand intérêt. Au vu des résultats obtenus (critères visuels, PSNR, IF et SSIM), la méthode EALSBMR est plus intéressante que la méthode AELSB.

Dans le troisième chapitre, nous avons étudié quelques méthodes stéganographiques fréquentielles qui peuvent être qualifiées de plus complexes et plus robustes que celles d'intégration qui opèrent dans le domaine spatial. Nous avons étudié l'algorithme LSB-DCT avec seuillage, et proposé d'utiliser le système chaotique RFCA afin de choisir aléatoirement les coefficients DCT dans lesquels nous insérons les bits secrets. Ensuite, nous avons étudié l'algorithme "DWT-LSB Alpha-Fusion" et nous lui avons ajouté plusieurs améliorations. Pour les algorithmes basés sur l'étalement de spectre, l'algorithme SSIS a été considéré, et nous avons suggéré d'utiliser le chaos (système SC) pour chiffrer le message secret et ensuite le moduler. Pour toutes ces propositions, plusieurs tests ont été effectués sur l'imperceptibilité visuelle et objective par la mesure des deux paramètres PSNR et SSIM. Aussi la comparaison avec les algorithmes originaux est systématiquement réalisée. Les résultats obtenus ont montré l'efficacité des améliorations proposées en termes de capacité d'insertion, de sécurité et d'imperceptibilité. Concernant ce dernier critère, la méthode SSIS améliorée est la meilleure en comparaison avec l'ensemble des méthodes étudiées.

Dans le quatrième chapitre, nous avons étudié et implémenté trois méthodes de stéganalyse universelle différenciées par le type du vecteur caractéristique utilisé: Farid, Shi ou Wang. Nous les avons ensuite appliquées sur les images stégo produites par les méthodes de stéganographie spatiales déjà étudiées et proposées. Pour ce faire, nous avons d'abord, extrait les vecteurs caractéristiques des images cover et stego, où ces derniers peuvent contenir de l'information sur toutes variations résultantes de la dissimulation. Ensuite, nous avons classifié ces vecteurs en se basant sur les paramètres : sensibilité, spécificité, précision et coefficient Kappa. Les résultats obtenus, montrent que la robustesse d'une méthode de stéganographie dépend du taux d'insertion, et du type de vecteur caractéristique utilisé. Le vecteur caractéristique de Shi semble, d'après les résultats de stéganlayse obtenus, le plus efficace.

À la suite de cette synthèse des travaux présentés dans ce manuscrit, un certain nombre de pistes pourraient être explorées: étude de la stéganalyse des méthodes fréquentielles, étude et construction des vecteurs caractéristiques encore plus efficaces. Etude et application du classifieur SVM et enfin extension de certaines méthodes de stéganographie améliorées aux méthodes de tatouage numérique.

ANNEXES

ANNEXES

Annexe A

Stéganographie LSB par Correspondance Revisité (LSB Matching Revisited), [Mielikainen, 2006]

Soit p_i et p_{i+1} les niveaux de gris de deux pixels consécutifs de l'image originale. Après l'insertion du message, la valeur de ith bit de message m_i est égale au bit de poids faible de ith pixel stégo, la valeur de (i+1)th bit du message m_{i+1} est une fonction de p_i et p_{i+1}.
La méthode permet la sélection d'une addition/soustraction de p_i pour transporter l'information car cette sélection permet de mettre la fonction $f(p_i, p_{i+1})$ à la valeur désirée.
Lorsque la fonction binaire a la propriété suivante :

$$f(p_i - 1, p_{i+1}) \neq f(p_i + 1, p_{i+1}) \qquad \forall\, l, n \in Z \qquad \text{(A.1)}$$

Le contrôle de la valeur de p_i permet le réglage de $f(p_i, p_{i+1})$ à une valeur désirée.
Si la fonction binaire $f(p_i, p_{i+1})$ est telle que :

$$f(p_i, p_{i+1}) \neq f(p_i, p_{i+1} + 1) \qquad \text{(A.2)}$$

alors la diminution ou l'augmentation de p_{i+1} par 1 change la valeur de la fonction $f(p_i, p_{i+1})$.
En effet :
La fonction $f(p_i, p_{i+1}) = LSB(\left\lfloor \frac{p_i}{2} \right\rfloor + p_{i+1})$ a les propriétés (A.1) et (A.2).
$\left\lfloor \frac{p_i}{2} \right\rfloor$ est la fonction entière d'un nombre réel $\frac{p_i}{2}$
Si p_i est impair alors : $f(p_i, p_{i+1}) = f(p_i - 1, p_{i+1}) \neq f(p_i + 1, p_{i+1})$
Si p_i est pair alors : $f(p_i, p_{i+1}) = f(p_i + 1, p_{i+1}) \neq f(p_i - 1, p_{i+1})$
Donc le control de la valeur de p_i permet le réglage de $f(p_i, p_{i+1})$ à une valeur désirée.
De même on a : $f(p_i, p_{i+1}) \neq f(p_i, p_{i+1} \pm 1)$
Donc la diminution ou l'augmentation de p_{i+1} par 1 change la valeur de la fonction $f(p_i, p_{i+1})$.
Donc on peut utiliser l'algorithme suivant pour insérer les bits du message en utilisant les propriétés de la fonction binaire ci-dessus :

si $m_i = LSB(p_i)$
  si $m_{i+1} \neq f(p_i, p_{i+1})$
    $p'_{i+1} = p_{i+1} \pm 1$
  Si non
    $p'_{i+1} = p_{i+1}$
  Fin si
  $p'_i = p_i$
Si non
  si $m_{i+1} = f(p_i - 1, p_{i+1})$
    $p'_i = p_i - 1$
  Si non
    $p'_i = p_i + 1$
  Fin si
  $p'_{i+1} = p_{i+1}$
Fin si

Algorithme d'insertion d'une paire de pixels

Exemple :

p_i	p_{i+1}	m_i	m_{i+1}	relations	p_i ' =	p_{i+1} ' =
19	10	0	0	$LSB(p_i) \neq m_i$ $LSB(\lfloor \frac{p_i-1}{2} \rfloor + p_{i+1}) \neq m_{i+1}$	$p_i+1=20$	$p_{i+1}=10$
30	20	0	0	$LSB(p_i) = m_i$ $LSB(\lfloor \frac{p_i}{2} \rfloor + p_{i+1}) \neq m_{i+1}$	30	19 ou 21
15	23	0	0	$LSB(p_i) \neq m_i$ $LSB(\lfloor \frac{p_i-1}{2} \rfloor + p_{i+1}) = m_{i+1}$	16	23
10	5	0	0		10	5
5	20	0	1	$LSB(p_i) \neq m_i$ $LSB(\lfloor \frac{p_i-1}{2} \rfloor + p_{i+1}) \neq m_{i+1}$	6	20
12	6	0	1	$LSB(p_i) = m_i$ $LSB(\lfloor \frac{p_i}{2} \rfloor + p_{i+1}) \neq m_{i+1}$	12	5 ou 7
5	13	0	1	$LSB(p_i) \neq m_i$ $LSB(\lfloor \frac{p_i-1}{2} \rfloor + p_{i+1}) = m_{i+1}$	4	13
12	7	0	1		12	7
12	6	1	0	$LSB(p_i) \neq m_i$ $LSB(\lfloor \frac{p_i-1}{2} \rfloor + p_{i+1}) \neq m_{i+1}$	13	6
21	5	1	0	$LSB(p_i) = m_i$ $LSB(\lfloor \frac{p_i}{2} \rfloor + p_{i+1}) \neq m_{i+1}$	21	4 ou 6
34	10	1	0	$LSB(p_i) \neq m_i$ $LSB(\lfloor \frac{p_i-1}{2} \rfloor + p_{i+1}) = m_{i+1}$	33	10
71	5	1	0		71	5
4	5	1	1	$LSB(p_i) \neq m_i$ $LSB(\lfloor \frac{p_i-1}{2} \rfloor + p_{i+1}) \neq m_{i+1}$	5	5
11	13	1	1	$LSB(p_i) = m_i$ $LSB(\lfloor \frac{p_i}{2} \rfloor + p_{i+1}) \neq m_{i+1}$	11	12 ou 14
8	18	1	1	$LSB(p_i) \neq m_i$ $LSB(\lfloor \frac{p_i-1}{2} \rfloor + p_{i+1}) = m_{i+1}$	9	18
7	4	1	1		7	4

Annexe B

Nous avons vu que les propriétés importantes de la fonction binaire $f(p_i, p_{i+1}) = LSB(\lfloor \frac{p_i}{2} \rfloor + p_{i+1})$ sont :

$$f(p_i + 1, p_{i+1}) \neq f(p_i - 1, p_{i+1})$$

$$f(p_i, p_{i+1}) \neq f(p_i, p_{i+1} + 1)$$

Donc :

L'addition/soustraction d'un nombre pair n'affecte pas la valeur de la fonction f :

$$f(p_i, p_{i+1}) = f(p_i, p_{i+1} + 2k_2) = LSB\left(\left\lfloor \frac{p_i}{2} \right\rfloor + p_{i+1} + 2k_2\right) = LSB\left(\left\lfloor \frac{p_i}{2} \right\rfloor + p_{i+1}\right)$$

$$f(p_i, p_{i+1}) = f(p_i + 4k_1, p_{i+1}) = LSB\left(\left\lfloor \frac{p_i + 4k_1}{2} \right\rfloor + p_{i+1}\right) = LSB\left(\left\lfloor \frac{p_i}{2} + 2k_1 \right\rfloor + p_{i+1}\right)$$
$$= LSB\left(\left\lfloor \frac{p_i}{2} \right\rfloor + p_{i+1}\right)$$

$$\forall\, k_1, k_2 \in Z$$

INSA de RENNES
Service des Formations

AVIS DU JURY SUR LA REPRODUCTION DE LA THESE SOUTENUE

Titre de la thèse:
Sécurité de l'information par stéganographie basée sur les séquences chaotiques

Nom Prénom de l'auteur : BATTIKH DALIA

Membres du jury :
- Monsieur BAKHACHE Bassem
- Monsieur DIAB Chaouki
- Monsieur DEFORGES Olivier
- Madame TARALOVA Ina
- Monsieur PUECH William
- Monsieur KHALIL Mohamad
- Monsieur SHAHIN Ahmad
- Monsieur EL ASSAD Safwan

Président du jury : Ahmad SHAHIN

Date de la soutenance : 18 Mai 2015

Reproduction de la these soutenue

X Thèse pouvant être reproduite en l'état
Thèse pouvant être reproduite après corrections suggérées

Fait à Rennes, le 18 Mai 2015

Signature du président de jury

Ahmad SHAHIN

Le Directeur,

M'hamed DRISSI