REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE DES MATHEMATIQUES ET DE L'INFORMATIQUE

DEPARTEMENT D'INFORMATIQUE

N° :……..

DOMAINE : MATHEMATIQUE ET INFORMATIQUE

FILIERE : INFORMATIQUE

OPTION : Réseaux et Technologie de l'Information et de Communication

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par : HADJI Faiçal

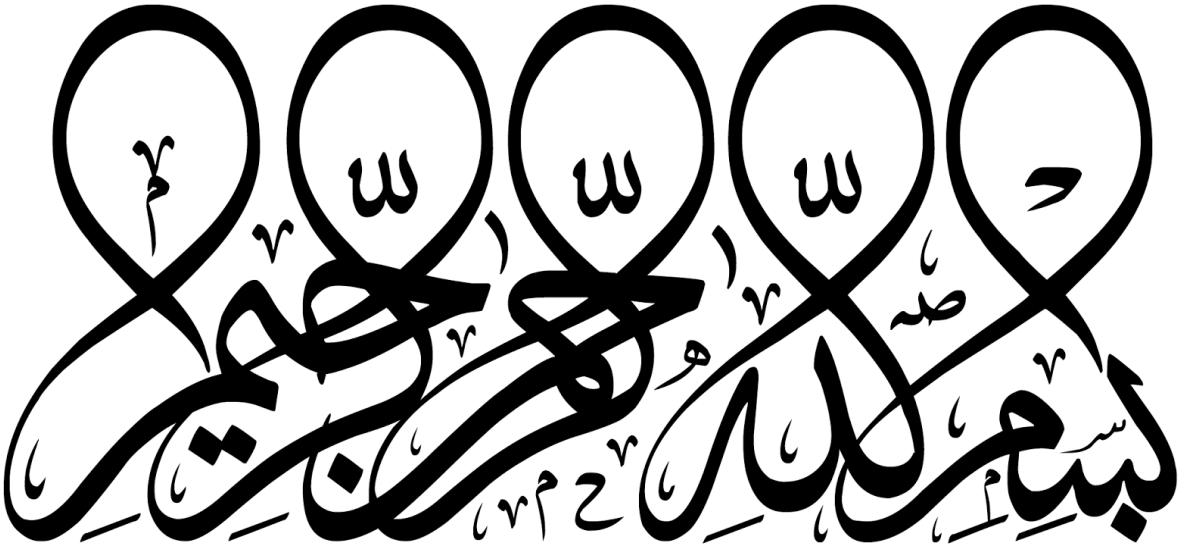
Intitulé

Conception et réalisation d'un système de cryptage pour les images médicales

Soutenu devant le jury composé de :

Mme. SAOUDI Lalia	Université de M'sila	Président
Dr. LAMICHE Chaabane	Université de M'sila	Rapporteur
Dr. MOUSSAOUI Adel	Université de M'sila	Examinateur

Année universitaire : 2017 /2018


بسم الله الرحمن الرحيم

Dédicace

Je dédie ce modeste travail :

À tous les membres de ma famille pour leur soutien continu et je leurs souhaite bonne santé et long vie

À tous mes amis, surtout à mon ami Saadone Billal qui a été mon compagnon pour les dernières années

À tous mes enseignants qui ont fait leurs possibles pour nous donner le maximum d'informations concernant notre étude

Enfin, à toutes celles et tous ceux qui ont contribué de près ou de loin à l'accomplissement de ce travail.

HADJI Faiçal

Remerciements

Avant tout on tient notre remerciement à notre Dieu tout puissant de nous avoir donné la foi, la force et le courage pour achever ce modeste travail

Je remercie mon encadreur Dr Lamiche Chaabane, de sa méthodologie et l'exactitude de ses précieux conseils

Je souhaite remercier tous les personnes qui m'ont aidé d'une façon directe ou indirecte à la réalisation de ce mémoire.

Merci infiniment

HADJI Faiçal

Table des matières

Liste des figures

Liste des Tableaux

INTRODUCTION GÉNÉRALE

Actuellement, avec la grande accélération dans le développement des technologies d'Internet et de la communication, la communication des images en général et les images médicales en particulier est une sorte qui s'applique fortement et joue un rôle très important dans la transmission de l'information. Cependant, la sécurité de l'information est un sujet sensible pour la recherche, la discussion et le développement, et le cryptage est l'une des meilleures alternatives qui s'est avérée efficace tout au long de l'histoire pour assurer la confidentialité et la sécurité de l'information.

Problématique et objectif

Maintenant, il est devenu clair que nous ne pouvons pas utiliser les méthodes de chiffrement classiques standard comme RSA, DES, AES, pour le chiffrement d'image numériques, par ce que ils ne sont pas atteindre fin requis pour ce type de données et l'accélération en multimédias. Ainsi les images numériques sont caractérisées par la redondance élevée, la forte corrélation et la taille volumineuse. Le problème posé est comment peut-on concevoir un système de cryptage pour assurer la sécurité de ce type de données ? Dans ce mémoire nous allons répondu à cette problématique, on va développer et implémenter un système hybride qui gère à la fois des images médicales et les images normales. Ce système est basé sur la Suite de Fibonacci modifiée, les cartes chaotiques sine, logistique et standard aussi les techniques de permutation en exploitant les avantages apportés par chacune d'elles.

Organisation du mémoire

Nous avons structuré notre mémoire en trois chapitres. Le premier chapitre met le point sur les notions de base d'image numérique et un résumé sur les images médicales. Dans le deuxième chapitre on a une brève présentation sur les techniques de cryptographie et ses classifications et les techniques de cryptage d'image. En troisième chapitre consiste à présenter notre méthode proposée. Puis on va terminer par une conclusion générale et quelques perspectives pouvant aider dans l'amélioration du système dans le futur.

CHAPITRE 1

INITIATION AUX IMAGES NUMÉRIQUES

1. Introduction

En raison de l'importance des images numériques et de la valeur des informations qu'elles contiennent, dans ce chapitre, nous allons référer aux concepts de base de l'imagerie à travers les types des images numériques. Ensuite, nous allons parler des méthodes de codage des couleurs dans les images. Puis nous allons parler des formats les plus importantes et les plus célèbres. Enfin nous allons parler des formats les plus importantes de l'imagerie médicale en particulier.

2. Notions de base sur l'imagerie

2.1. L'image numérique

Une image numérique est une mosaïque de points unicolores (pixels) [1], et peut être définie comme une fonction bidimensionnelle, $f(x, y)$, où x et y sont des coordonnées spatiales (plan) pour chaque pixel [2], Ces pixels seront affectés de nombres binaires permettant de définir des teintes de gris ou des couleurs [3].

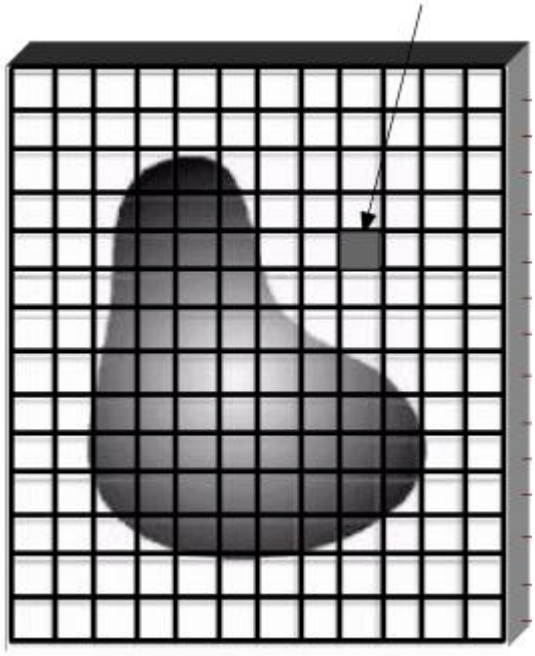


Figure 1.1 : Image numérique [4].

2.2. Pixel

Les composants élémentaires d'image sont des points appelés pixels (abréviation de **PIC**ture **El**ement) pour former une image. Le pixel représente ainsi le plus petit élément constitutif d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image [5], et chaque pixel à sa propre couleur (valeur).

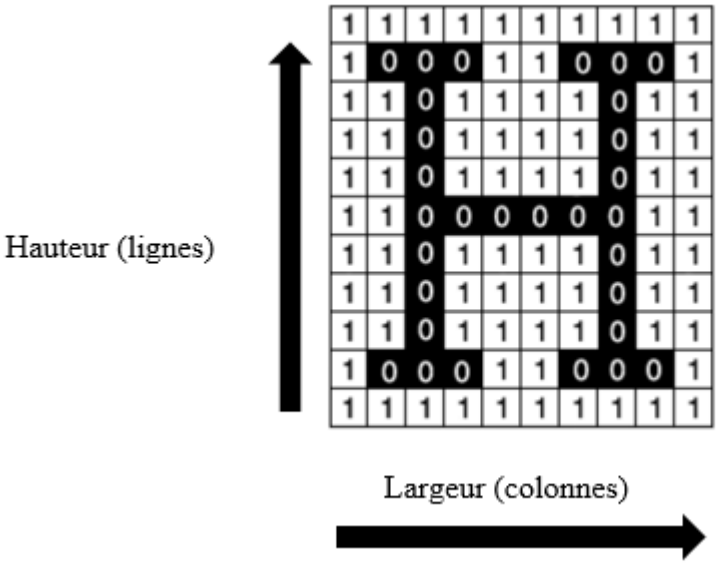


Figure 1.2 : Distribution des pixels par lignes et colonnes [6].

2.3. Définition

La définition est le nombre de pixels constituants l'image [3].

2.4. La taille

La taille de l'image est la place qu'elle occupe dans le codage binaire. Son unité est «l'octet» [3].

Taille = nombre d'octets pour chaque pixel × définition

2.5. Résolution

La résolution d'une image est définie par le nombre de pixels par unité de longueur **dpi** (**d**ot **p**er **i**nch = point d'encre par pouce) pour une imprimante ou (**ppp** = **p**ixels **p**ar **p**ouce pour un fichier image). Cette résolution dépendra de la qualité de la numérisation.

$$\text{Résolution} = {}^{définition}/_{longueur}$$

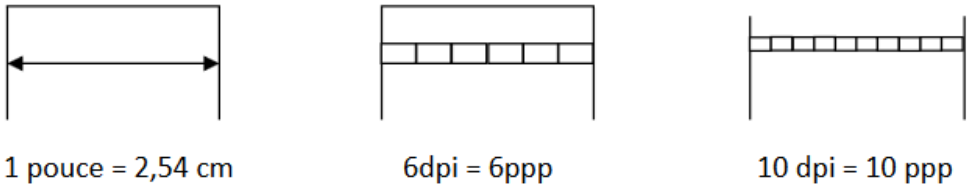


Figure 1.3 : Explication de résolution d'une image [3].

3. Les différents types d'image

Il existe deux types d'images numériques :

3.1. Matricielle (bitmap)

Formée d'une grille composée de pixels. Plus on zoom, plus les pixels deviennent apparents [6]. Les formats d'images bitmap : BMP, PCX, GIF, JPEG, TIFF.

Les photos numériques et les images scannées sont de ce type [7].

3.2. Vectorielle

Formée de lignes calculées de manière géométrique. Lors d'un zoom avant ou arrière, la forme est recalculée en fonction de notre position sans perdre de qualité [6].

Le processeur est chargé de "traduire" ces formes en informations interprétables par la carte graphique (images Word, Publisher, CorelDraw - format WMF, CGM, etc.)

Les avantages d'une image vectorielle : les fichiers qui la composent sont petits, les redimensionnements sont faciles sans perte de qualité.

Les inconvénients : une image vectorielle ne permet de représenter que des formes simples. Elle n'est pas donc utilisable pour la photographie notamment pour obtenir des photos réalistes [7].

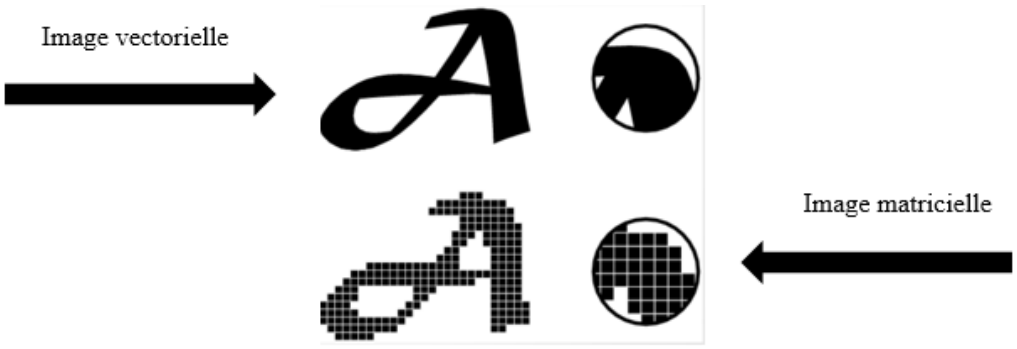


Figure 1.4 : Différence entre image vectorielle et image matricielle [6].

4. Les différents modes de couleurs des images

4.1. Mode binaire

Appelé aussi Mode bitmap (noir et blanc) : Avec ce mode, il est possible d'afficher uniquement des images en deux couleurs pour chaque pixel : noir et blanc. Il utilise une seule couche [5].

- Codage en 1 bit par pixel (bpp) : 2^1= 2 possibilités : [0,1].

1	1	1	1	1	1	1	1	1	1
1	1	1	0	0	0	0	1	1	1
1	1	0	1	1	1	1	0	1	1
1	0	1	1	1	1	1	1	0	1
1	0	1	0	1	1	0	1	0	1
1	0	1	1	1	1	1	1	0	1
1	0	1	0	1	1	0	1	0	1
1	0	1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0	1	1
1	1	1	0	0	0	0	1	1	1
1	1	1	1	1	1	1	1	1	1

Figure 1.5 : Codage binaire (0,1) [5].

Figure 1.6 : Image codée en binaire**.**

4.2. Mode niveau de gris

A chaque pixel codé en n bits est affecté un nombre binaire variant de «0» (pour le noir) à «2^n-1» (pour le blanc), avec n le nombre de bits pour chaque pixel.

Il y aura alors «2^n» niveaux de gris.

Si le codage se fait en 8 bits par pixel, il y aura : $2^n=2^8=256$ niveaux de gris allant du blanc au noir.

Si le codage se fait en 16 bits par pixel, il y aura : $2^n=2^{16}=65536$ niveaux de gris allant du blanc au noir [3].

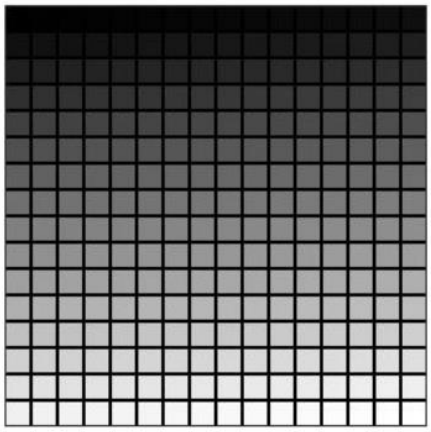


Figure 1.7 : Nuance de 256 gris [5].

Figure 1.8 : Image codée en niveau de gris**.**

4.3. Mode couleurs indexées

Permet d'obtenir jusque 256 couleurs fixes, définies à l'avance dans une palette. Il n'utilise qu'une seule couche [5].

Codage en 8 bits par pixel (bpp) : 2^8=256 possibilités.

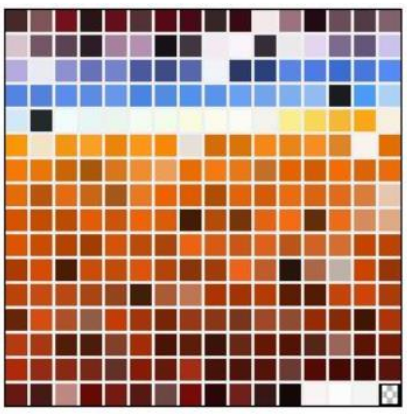


Figure 1.9 : Palette de 256 couleurs utilisées [5].

Figure 1.10 : Image codée en couleurs indexées [5].

4.4. Les modes colorimétriques RVB / CMJN

Afin de créer des images encore plus riches en couleurs (et donc disposer de plus qu'une palette limitée à 256 couleurs), l'idée de mélanger des couleurs primaires en « couches » est arrivée [5].

Il existe deux systèmes de représentation des couleurs par mélange, selon qu'on les reproduit sur un écran d'ordinateur ou sur support papier via une imprimante :

Figure 1.11 : Les deux modes colorimétriques [5].

4.4.1. Mode couleur RVB (lumière éteinte)

Grâce au mélange des 3 couches de couleur (Rouge, Vert, Bleu), il est possible de reproduire un plus grand nombre de nuances qu'avec une palette en mode couleurs indexées [5].

Avec un codage en RVB 8 bits par couche :

Chaque couche utilise 8bits (1 octet), soit 256 nuances possibles : 8 bits pour le Rouge, 8 bit pour le Vert et 8 bits pour le Bleu.

Donc utilisation de 3 x 8 bits = 24 bits utilisées au total.

=> 256 x 256 x 256 = 2^{24} = 16,7milions possibles !

Avec un codage en RVB 16 bits par couche :

Chaque couche utilise le double, soit 16 bits ! (65535 nuances). 3 x 16 = 48bits utilisées au total.

=> 65535 x 65535 x 65535 = 2^{48} = 4 milliards possibles !

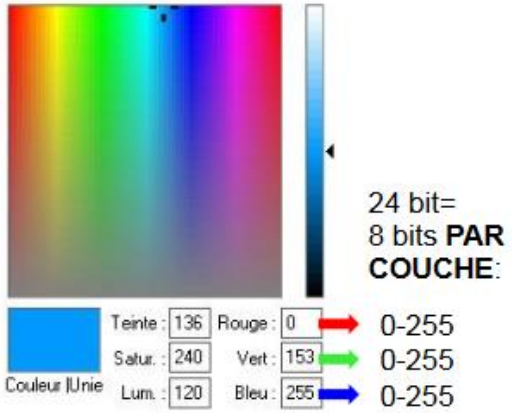


Figure 1.12 : Le mode RVB [5].

Figure 1.13 : Image codée en couleurs [5].

4.4.2. Mode couleur CMJN (support papier)

Comme les écrans d'ordinateur ne peuvent afficher que du RVB, Photoshop sépare les images CMJN en 4 couches (Cyan, Magenta, Jaune et Noir ou chaque couleur est exprimée en pourcentage) et converti le tout en RVB pour être affiché sur l'écran. Cependant pour L'utilisateur, le fichier possède bien 4 couches distinctes sur lesquels il est possible de travailler [5].

Avec un codage en CMJN 8 bits par couche :

Chaque couche utilise 8 bits (soit 256 nuances possibles) : 8 bits pour le Cyan, 8 bits pour le Magenta, 8 bits pour le Jaune et 8bits pour le Noir.

Donc utilisation de 4 x 8 bits = 32 bits utilisées au total.

=> 256 x 256 x 256 x 256 = 2^{32}= 4 milliards possibles !

Avec un codage en CMJN 16 bits par couche :

Chaque couche utilise le double, soit 16 bits ! (65535 nuances). 4 x 16 = 64 bits utilisées au total.

=> 65535 x 65535 x 65535 x 65535 = 2^{64} possibilités !

5. Format d'enregistrement d'une image

Les formats des images ont une relation avec le type d'image lui-même

5.1. Les formats matriciels

Nom du format	Points forts	Points faibles	Note
JPEG **JPEG 2000** **J**oint **P**hotographic **E**xperts **G**roup	Compression Excellente	Compression destructrice	Spécialement conçu pour les photographies, il est cependant à utiliser avec délicatesse tant sa compression peut brouiller l'image. Le format JPEG2000, évolution du format original, peut être réglé pour compresser sans pertes**.**
GIF (**G**raphical **I**nterchange **F**ormat)	Possibilité d'animation et de transparence compression efficace	Limité à 256 couleurs	Très répandu sur le Web malgré ses faiblesses et un problème de droit sur son format de compression. À déconseiller pour les photos
PNG (**P**ortable **N**etwork **G**raphics)	Compression Excellente sans perte. Possibilité de transparence. Standard donc pérenne.	Pas très efficace pour les larges photographies	Format destiné à remplacer le format GIF et ses limitations, mais ayant encore du mail à s'implanter dans les habitudes des développeurs. Peut remplacer les JPEG comme les GIF (sauf en ce qui concerne l'animation)**.**
TIFF (**T**agged **I**mage **F**ile **F**ormat)	Compression sans perte efficace. Couche de transparence	Lourdeur des fichiers non compressés. Format propriétaire.	Format de stockage très utilisé, à éviter pour le Web
BMP (**Bi**t**ma**p**)**	Format par défaut de Windows	Disponible uniquement sur	Généralement non compressé et de ce fait des fichiers très « lourds »

		la plateforme de Microsoft	

Table 1.1 : Les formats matriciels [7].

5.2. Les formats vectoriels

Nom du format	Points forts	Points faibles	Note
AI (**A**dobe **I**llustrator)	Reconnu par tous les logiciels graphiques.	Format propriétaire.	Format standard d'Adobe Illustrator, l'un des plus utilisés du fait de la popularité du logiciel.
PS/EPS (**P**ost**s**cript / **E**ncapsulated **P**ost**s**cript)	Très bien reconnu sur tous les systèmes.	N'a d'intérêt que dans le cadre d'une impression. Fichier très lourd.	Format hybride bitmap/vectoriel, réservé à l'impression. EPS est un fichier PS qui comporte quelques restrictions supplémentaires.
SVG (**S**calable **V**ector **G**raphics)	Format XML donc extensible. Très compressible car format texte. Standard donc pérenne. Permet les animations et la transparence. Peut afficher des images bitmap.	Encore très peu reconnu, car peu d'outils disponibles et manque d'implémentation au sein de navigateurs (besoin d'un plugin).	Promis à un grand avenir malgré un démarrage lent, ce format est souvent cité comme capable de rivaliser avec les premières versions de Flash.
FLA/SWF (Flash)	Très polyvalent, peut utiliser des mp3, des JPEG, des vidéos... Très répandu sur le Web.	Format propriétaire et fermé.	C'est le standard de fait des animations vectorielles sur le Web.
PDF (**P**ortable **D**ocument **F**ormat)	Affiche les documents	Taille prohibitive. Ne peut se lire qu'avec le logiciel Acrobat ou logiciel équivalent.	Version simplifiée de PostScript, il a été conçu pour afficher les documents de la même manière quel que soit le système.
PICT (**Pict**ure)	Format par défaut de Mac OS, donc encore utilisé.	Disponible uniquement sur la plateforme d'Apple	N'a plus grand intérêt face aux autres formats existants.

Table 1.2 : Les formats vectoriels [7].

5.3. Formats des images médicales

Le standard DICOM (Digital **I**maging and **C**ommunication in **M**edicine**),** Créé en 1985 par :

- l' ACR (American College of Radiology)
- la NEMA (National Electric Manufacturers Association)

Les caractéristiques les importants :

- Standardiser les données transmises entre les différents appareils de radiologie.
- Format de fichier + protocole de transmission des données (basé sur TCP/IP).
- Faciliter les transferts d'images entre les machines de différents constructeurs.

Eviter d'avoir pour chaque constructeur de matériel d'imagerie un format de données propriétaire (incompatibilités, coût, perte d'information).

Tout numérique possible :

- Pour éviter le tirage des clichés sur papier argentique
- Pour diminuer le coût d'une radiographie.
- Amélioration du suivi médical des patients (transfert d'un établissement de santé à un autre).

Les images au format **DICOM** accompagnant les dossiers médicaux sont lisibles sur tout matériel informatique compatible.

La sécurisation des échanges, via un service appelé "accord de stockage", et différents mécanismes de signature des documents, et la cohérence du rendu des images [8].

5.3.1. Les standards de compression

Les standards en compression de données peuvent être classés en deux catégories :

- Ceux qui ne font aucune hypothèse sur la nature des données.
- Ceux qui s'appuient sur une organisation spatio-temporelle particulière (image 2D ou suite d'images 2D).

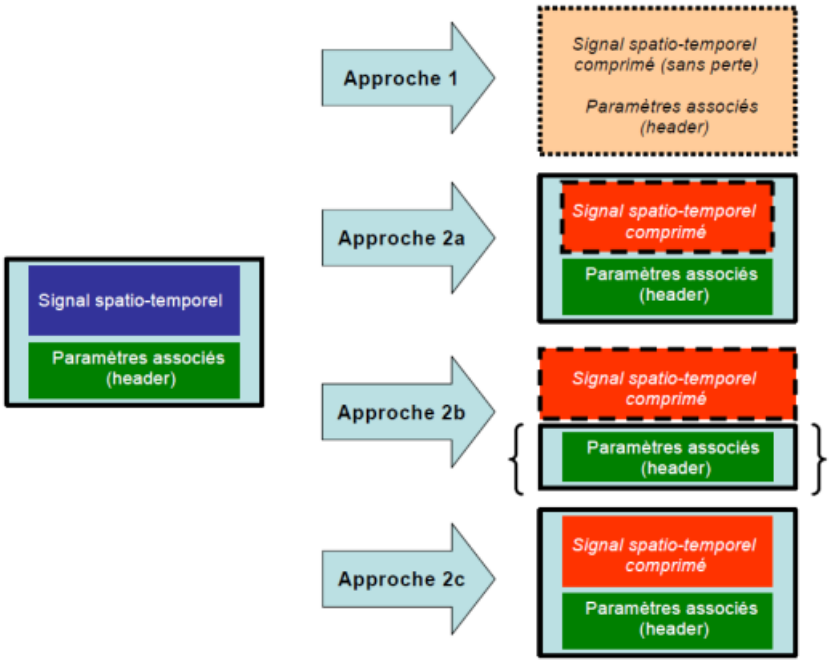


Figure 1.14 : Approches généralistes et spécifiques pour la compression d'image [8].

Les avantages et les inconvénients :

	Avantages	**Inconvénients**
Approche 1 "compression généraliste" (ex : gzip, compress)	Généricité Facilité de mise en œuvre Coût très faible	Performances faibles
Approche 2a "compression d'image généraliste" par encapsulation (ex : JPEG, MPEG)	Réutilisation d'implémentations existants pour la compression/ décompression et la virtualisation des images Performances très optimisées Prise en compte du contexte médical (*header* contenant le nom du patient, les paramètres d'acquisition, etc.)	Eventuellement inadaptée à des données très spécifique, ou performances sub-optimales
Approche 2b "compression d'image généraliste" (ex : JPEG, MPEG)	Facilite la diffusion la plus large (hors des services spécialisés, et vers le grand public), au moindre coût (navigateur web)	Pas de prise en compte du contexte médical (*header*)
Approche 2c "compression d'image spécifique"	Peut permettre d'obtenir des performances optimales, découlant d'une très bonne adéquation à la structure des données	Coût de développement inhérent au caractère spécifique

Table 1.3 : Avantages et inconvénients des standards généralistes et standards spécifiques [8].

6. Conclusion

Dans ce chapitre, nous avons parlé de l'importance des images numériques et de ses types, les méthodes de codage des pixels des images numériques, les formats connues les plus célèbres et leurs caractéristiques, avec un accent sur les images médical pour la sensibilité des informations contenues.

Dans le prochain chapitre, nous allons parler de la sécurité informatique, le cryptage en général, et le cryptage des images, en particulier avec une mention de quelques techniques dans ce domaine en détail.

CHAPITRE 2

CONCEPTS PRÉLIMINAIRES SUR LA CRYPTOGRAPHIE

1. Introduction

La sécurité informatique est devenue une préoccupation majeure pour tous ceux qui sont intéressés par l'informatique et à cette fin la plupart des développeurs se concentrent sur les techniques de cryptage pour fournir de bons résultats. Dans ce chapitre, nous allons parler sur les bases et les principes de sécurité informatique, la cryptographie et des domaines de convergence entre eux. Ensuit nous allons parler les classifications des algorithmes de cryptage en détail avec les types et la présentation des différences entre eux. Puis nous allons parler sur les types des méthodes pour crypter les images, et que nous allons parler sur des différents critères pour mesurer l'efficacité des algorithmes de cryptage d'image. Enfin nous allons parler des méthodes modernes pour crypter les images illustrant ces points forts.

2. Généralité sur la sécurité informatique & la cryptographie

2.1. Introduction à la sécurité informatique

2.1.1. La sécurité informatique

La sécurité informatique c'est l'ensemble des moyens mis en œuvre pour réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles [9].

2.1.2. Vulnérabilité

Faiblesse / faille : faute accidentelle ou intentionnelle introduite dans spécification, conception ou configuration du système [10].

2.1.3. Menace

Violation potentielle d'une propriété de sécurité [10].

2.1.4. Risque

La probabilité qu'une menace exploitera une vulnérabilité du système. Couple (menace, vulnérabilité) [10].

2.1.5. Attaque

Action malveillante qui tente d'exploiter une faiblesse dans le système et de violer un ou plusieurs besoins de sécurité [10].

2.2. Vocabulaire de base de la cryptographie

2.2.1. La cryptologie

Il s'agit d'une science mathématique comportant deux branches : la cryptographie et la cryptanalyse [11].

2.2.2. La cryptographie

L'étude des méthodes donnant la possibilité d'envoyer des données de manière confidentielle sur un support donné [11].

2.2.3. La cryptanalyse

Opposée à la cryptographie, elle a pour but de retrouver le texte clair à partir de textes chiffrés en déterminant les failles des algorithmes utilisés [11].

2.2.4. Crypto-système

Il est défini comme l'ensemble des clés possibles (espace de clés), des textes clairs et chiffrés possibles associés à un algorithme donné [11].

2.2.5. Texte en clair

C'est les données (message, texte,..) à protéger.

2.2.6. Le chiffrement

Processus de transformation d'un message M de telle manière à le rendre incompréhensible Basé sur une fonction de chiffrement E [10].

2.2.7. Texte chiffré

Appelé également *cryptogramme*, le texte chiffré est le résultat de l'application d'un chiffrement à un texte clair [11].

2.2.8. Le déchiffrement

Processus de reconstruction du message clair à partir du message chiffré [10].

2.2.9. Clef

Il s'agit du paramètre impliqué et autorisant des opérations de chiffrement et/ou déchiffrement [11].

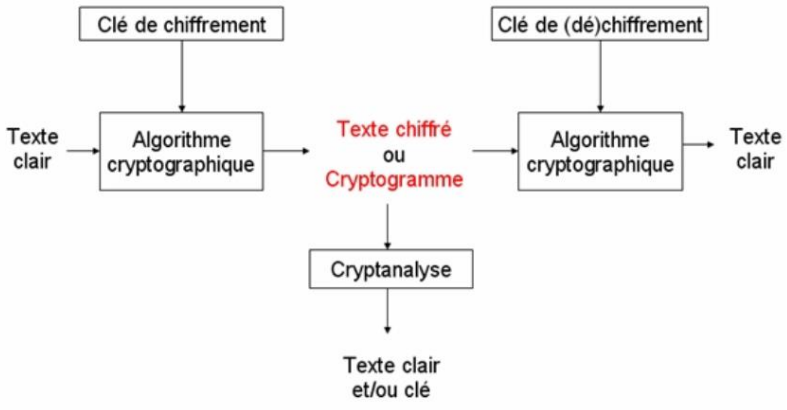


Figure 2.1 : Protocole de chiffrement **.**

2.2.10. Confusion

La confusion correspond à une volonté de rendre la relation entre la clé de chiffrement et le texte chiffré la plus complexe possible [12]**.**

2.2.11. Diffusion

La diffusion est une propriété où la redondance statistique dans un texte en clair est dissipée dans les statistiques du texte chiffré. En d'autres termes, un biais en entrée ne doit pas se retrouver en sortie et les statistiques de la sortie doivent donner le moins possible d'informations sur l'entrée [12].

2.2.12. Substitution

Les substitutions consistent à remplacer des symboles ou des groupes de symboles par d'autres symboles ou groupes de symboles dans le but de créer de la confusion [13].

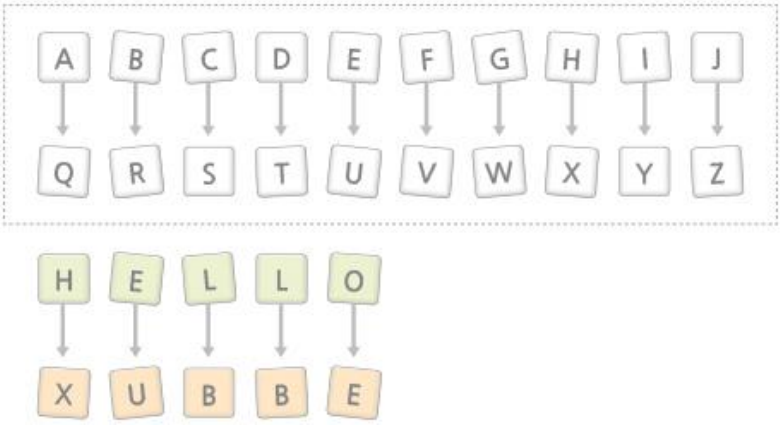


Figure 2.2 : Chiffrement par substitution [13]**.**

2.2.13. Permutation (transposition)

Chiffrement par permutation (Un chiffrement par transposition) est un chiffrement qui consiste à changer l'ordre des lettres, le chiffrement par transposition demande de découper le texte clair en blocs de taille identique. La même permutation est alors utilisée sur chacun des blocs [14].

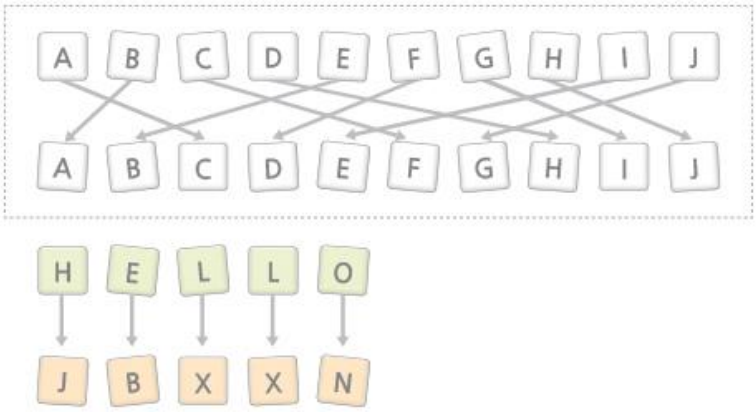


Figure 2.3 : Chiffrement par transposition [13]**.**

2.3. Les buts de la cryptographie (A quoi sert la cryptographie)

2.3.1. La confidentialité

Le texte chiffré ne doit être lisible que par les destinataires légitimes. Il ne doit pas pouvoir être lu par un intrus [15].

2.3.2. L'authentification

Le destinataire d'un message doit pouvoir s'assurer de son origine. Un intrus ne doit pas être capable de se faire passer pour quelqu'un d'autre [15].

2.3.3. L'intégrité

Le destinataire d'un message doit pouvoir vérifier que celui-ci n'a pas été modifié en chemin. Un intrus ne doit pas être capable de faire passer un faux message pour légitime [15].

2.3.4. Le non répudiation

Un expéditeur ne doit pas pouvoir, par la suite, nier à tort avoir envoyé un message [15].

3. Classification des crypto-systèmes

3.1. Crypto-système symétrique (à clé secrète)

Le principe est que la clé de chiffrement est la même que la clé de déchiffrement [10], L'avantage principal de ce mode de chiffrement est sa rapidité mais avec inconvénient d'échange d'un secret [11]. Les algorithmes les plus répandus sont : RC4 DES, AES, 3DES, etc.

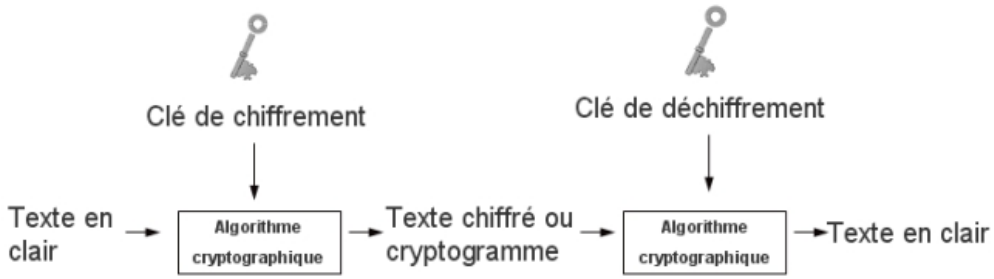


Figure 2.4 : Chiffrement symétrique [11].

Il existe deux grandes familles de chiffrement dans cette classe :

3.1.1. Chiffrement par blocs

Les messages sont découpés en blocs de taille à une relation avec la taille de clé (après le remplacement de chaque caractère par un code binaire), basé sur les deux catégories de chiffrement (par substitution et par transposition) et la combinaison entre eux, par exemple :

- DES : blocs de 64 bits, clés de 56 bits
- AES : blocs de 128 bits

3.1.2. Chiffrement par flots

Les données sont traitées en flux (traitement bit par bit), par exemple :

- RC4 : chiffrement octet par octet

3.2. Cryptage asymétrique (clé publique)

Le principe est que chaque personne (machine) a deux clés (une clé publique *PK* (symbolisée par la clé verticale) pour le chiffrement et une clé privée secrète *SK* (symbolisée par la clé horizontale) pour le déchiffrement) Propriété : La connaissance de *PK* ne permet pas de déduire *SK*, et : $D_{SK}(E_{PK}\ (M)) = M$, et l'algorithme de cryptographie asymétrique le plus connu est le RSA.

Le principe de ce genre d'algorithme est qu'il s'agit d'une fonction unidirectionnelle à trappe. Une telle fonction à la particularité d'être facile à calculer dans un sens, mais difficile voire impossible dans le sens inverse. La seule manière de pouvoir réaliser le calcul inverse est de connaître une trappe. Une trappe peut par exemple être une faille dans le générateur de clés. Cette faille peut être soit accidentelle ou intentionnelle de la part du concepteur [11].

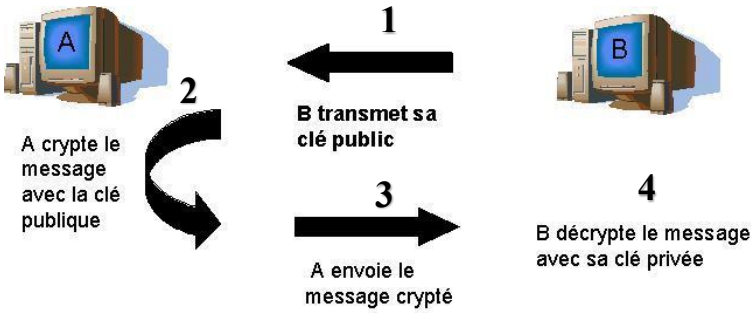


Figure 2.5 : Chiffrement asymétrique [10].

3.3. Cryptage hybride

Le chiffrement hybride (la combinaison entre le cryptage symétrique et asymétrique) d'un message M se déroule en deux étapes :

- Dans un premier temps, l'émetteur choisit une clé symétrique K aléatoire. Il utilise ensuite cette clé K pour chiffrer (symétriquement) le message M.
- Puis il chiffre (asymétriquement) la clé K avec la clé publique du destinataire. Il envoie à son destinataire les chiffrés de M et de K. Le destinataire déchiffre d'abord la clé K, puis l'utilise pour retrouver M.

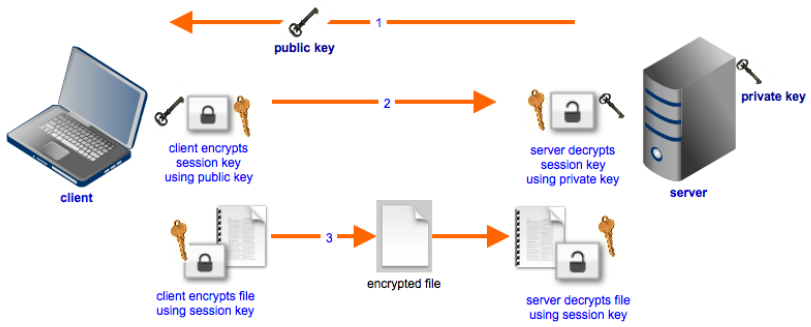


Figure 2.6 : Chiffrement hybride [10].

4. Méthodes du cryptage des images

Il existe deux grandes différences entre les données textuelles et les images numériques rendant les méthodes de cryptage de texte pour la plupart des cas inapplicable au cryptage des images :

- La différence principale réside dans la taille, en effet la quantité d'informations contenues dans l'image est beaucoup plus volumineuse que celles contenues dans les données textuelles.
- La deuxième différence concerne la perte de données, lorsqu'une technique de compression est appliquée.

Contrairement aux images, l'utilisation d'une méthode de compression avec perte est totalement interdite lors du chiffrement d'un texte, par conséquent, les chercheurs ont étudié plusieurs méthodes de chiffrement d'image avec/sans perte [16]. D'autre part, les algorithmes de chiffrement des images peuvent être classés selon le domaine d'application comme suit :

4.1. Méthodes dans le domaine spatial

Dans le domaine spatial, on applique le schéma de cryptage sur le plan d'image lui-même, et les approches de cette catégorie sont basées sur une manipulation directe des pixels d'une image. Dans ces algorithmes, le chiffrement détruit la corrélation entre les pixels et rend les images cryptées incompressibles. Les pixels de l'image peuvent être reconstruits (récupérés) complètement par un processus inverse sans aucune perte d'information.

Les algorithmes de cryptage d'image dans le domaine spatial existants peuvent être classés en deux catégories :

- Dans la première catégorie, un pixel est considéré comme le plus petit élément, et une image numérique est considérée comme un ensemble de pixels.
- Dans la deuxième catégorie, un pixel peut être en outre divisé en bits, sur lesquels des opérations au niveau de bits sont effectuées. Par exemple, un pixel dans une image en niveaux de gris est généralement constitué de 8 bits [16].

4.2. Méthodes dans le domaine fréquentiel

Les schémas de cryptage dans le domaine fréquentiel sont basés sur la modification de la fréquence de l'image en utilisant une transformation, ainsi, la reconstruction des pixels de l'image originale dans le processus de décryptage cause généralement une perte d'information [16].

5. Outils élémentaires d'analyse d'un algorithme du cryptage d'image (mesures de performance)

5.1. Espace de clés

La taille de l'espace de clé est le nombre de paires de clés de cryptage/décryptage qui sont disponibles dans le système de chiffrement [17]. Une condition nécessaire, mais pas suffisante à un schéma de cryptage pour qu'il soit sûr est que l'espace clés soit suffisamment grand pour assurer la sécurité contre l'attaque par force brute [16].

5.2. Analyse statistique

5.2.1. L'histogramme

L'histogramme d'une image désigne un histogramme des valeurs d'intensité des pixels. Cet histogramme est un graphique illustrant le nombre de pixels dans une image à chaque valeur d'intensité trouvée dans cette image. Pour une image grise il y a 256 intensités différentes possibles, ainsi, l'histogramme s'affiche graphiquement en utilisant 256 chiffres indiquant la distribution des pixels entre ces valeurs de niveaux de gris [18]. Dans un contexte de chiffrement d'image, l'histogramme de l'image chiffrée doit être uniforme pour qu'un adversaire ne puisse extraire aucune information à partir de cet histogramme [16].

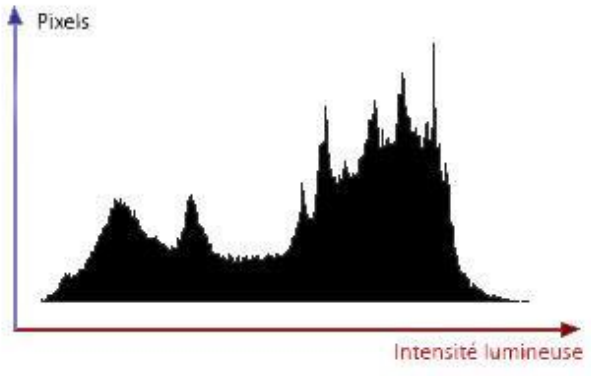


Figure 2.7 : Histogramme d'une image niveau de gris [19].

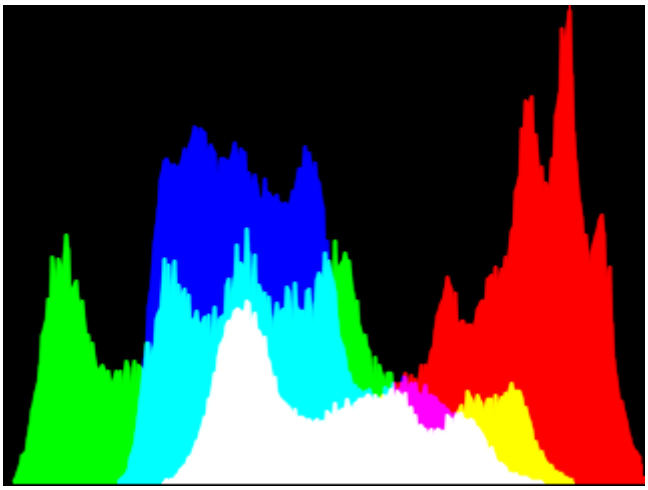


Figure 2.8 : Histogramme d'une image couleur [20].

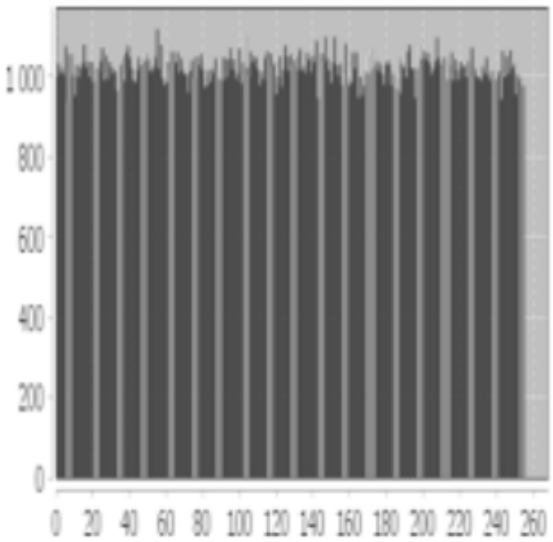


Figure 2.9 : Histogramme d'une image chiffrée [21].

5.2.2. La corrélation entre les pixels adjacents

La corrélation est une technique qui permet de comparer deux images pour estimer les déplacements des pixels d'une image par rapport à une autre image de référence. Les pixels adjacents d'une image standard ont une forte corrélation. Un bon schéma de cryptage d'image doit supprimer une telle corrélation afin d'assurer la sécurité contre l'analyse statistique [16], et les coefficients de corrélation de chaque paire ont été calculées en utilisant les formules suivantes :

$$r = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \qquad (2.1)$$

Où

$$cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}\big(x_i - E(x)\big)\big(y_i - E(y)\big) \qquad (2.2)$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad (2.3)$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}\big(x_i - E(x)\big)^2 \qquad (2.4)$$

Tel que :

r : la corrélation.

cov : la covariance.

E : l'espérance mathématique.

D : la variance.

x, y : les valeurs des pixels des images.

5.2.3. L'entropie

Selon la théorie de Shannon [22], l'entropie d'une information est la quantité d'information englobée ou libérée par une source d'information. En particulier, plus la source est redondante, moins elle contient d'information [23]. En absence de contraintes particulières, l'entropie est maximale pour une source dont tous les symboles sont équiprobables. Ainsi, elle est l'une des principales mesures de l'aléatoire de l'information. Les valeurs de l'entropie élevée manifestent un haut degré de caractère aléatoire, et pour tout message codé sur *M* bits, la limite supérieure de l'entropie est *M*. La formule utilisée pour calculer l'entropie d'une source *m* est comme suit [16] :

$$H(m) = -\sum_{i=0}^{2^n-1} p_i \, log_2(p_i) \qquad (2.5)$$

Où p_i définit la probabilité d'un pixel et n est le nombre de bits dans chaque pixel.

Donc pour un chiffrement d'images au niveau de gris, La valeur de l'entropie doit être très proche de 8.

5.3. Analyse de sensitivité

5.3.1. Attaques différentielles

Afin de détecter la relation entre l'image originale et l'image cryptée, un adversaire fait un petit changement sur l'image claire, ensuite utilise l'algorithme de cryptage pour crypter l'image avant et après le changement, dans le but de tester comment une petite modification

dans l'image originale affecte l'image cryptée. Ce genre d'attaque est appelé attaque différentiel.

Pour assurer la sécurité d'un schéma de cryptage d'image contre l'analyse différentielle, deux mesures quantitatives sont utilisés : le NPCR (Number of Pixels Change Rate) et l'UACI (Unified Average Changing Intensity).

Le NPCR représente le taux de pixels différents entre les deux images chiffrées, tandis que l'UACI représente la différence de l'intensité moyenne. La formule utilisée pour calculer ces deux pourcentages est définie comme suit :

$$NPCR = \frac{\sum_{i,j} f(i,j)}{W \times H} \times 100\% \qquad (2.6)$$

$$UACI = \frac{1}{W \times H}\left[\sum_{i,j} f(i,j)\frac{|C1(i,j) - C2(i,j)|}{255}\right] \times 100\% \qquad (2.7)$$

Où W et H représentent la largeur et la hauteur de l'image respectivement. $C1(i, j)$ est l'image cryptée et $C2(i, j)$ est l'image cryptée après avoir changé un pixel de l'image clair. Pour les pixels à la position (i, j), si $C1(i, j) \neq C2(i, j)$, alors $f(i, j) = 1$; sinon $f(i, j) = 0$.

Un NPCR > 99,6094% et un UACI > 33,4635% assure qu'un schéma de chiffrement d'image est sécurisé contre cette attaque [16].

5.3.2. Sensitivité de la clé

Un algorithme idéal de chiffrement d'image doit être sensible à la clé. C'est-à-dire le changement d'un seul bit dans la clé secrète devrait produire une image cryptée complètement différente. Pour tester la sensibilité de la clé de chiffrement, nous avons effectué les étapes suivantes [16] :

- Une image originale est chiffrée en utilisant la clé secrète.
- La même image originale est cryptée en faisant une légère modification dans la clé secrète.
- Ensuite, on compare les deux images chiffrées en utilisant les deux mesures NPCR et UACI.
- Ainsi, si les valeurs de l'NPCR et l'UACI obtenues sont supérieurs à 99,6094% et à 33,4635% respectivement : on dit que le schéma est sensible à la clé**.**

6. État de l'art sur les techniques de cryptage d'image

6.1. Méthode basé sur la théorie du Fibonacci

6.1.1. Fibonacci

Leonardo Fibonacci (v. 1175 à Pise - v. 1250) est un mathématicien italien. Il avait, à l'époque, pour nom d'usage « Leonardo Pisano » (il est encore actuellement connu en français sous l'équivalent « Léonard de Pise »), et se surnommait parfois lui-même « Leonardo Bigollo » (*bigollo* signifiant « voyageur » en italien).

S'il est connu pour la suite de Fibonacci, il joue surtout un rôle d'une importance considérable en faisant le lien entre le savoir mathématique des musulmans, notamment des chiffres indo-arabes, et l'Occident [24].

6.1.2. La suite du Fibonacci

La suite de Fibonacci est une suite d'entiers dans laquelle chaque terme est la somme des deux termes qui le précèdent. Elle commence généralement par les termes 0 et 1 (parfois 1 et 1) et ses premiers termes sont : 1, 1, 2, 3, 5, 8, 13, 21, etc. tel que :

$$U_0 = 1, U_1 = 1 \text{ Et } U_n = U_{n-1} + U_{n-2}, n \geq 2 \qquad (2.8)$$

Elle doit son nom à Leonardo Fibonacci qui, dans un problème récréatif posé dans l'ouvrage Liber abaci publié en 1202, décrit la croissance d'une population de lapins : « Un homme met un couple de lapins dans un lieu isolé de tous les côtés par un mur. Combien de couples obtient-on en un an si chaque couple engendre tous les mois un nouveau couple à compter du troisième mois de son existence ? » [25].

Cette suite est fortement liée au nombre d'or, φ (phi). Ce nombre intervient dans l'expression du terme général de la suite. Inversement, la suite de Fibonacci intervient dans l'écriture des réduites de l'expression de φ en fraction continue : les quotients de deux termes consécutifs de la suite de Fibonacci sont les meilleures approximations du nombre d'or [25].

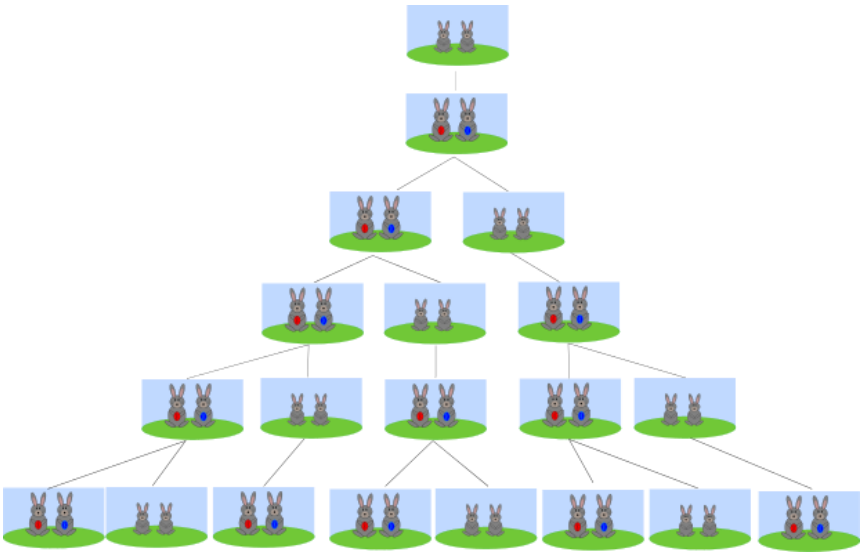


Figure 2.10 : Croissance d'une population de lapins selon la suite de Fibonacci, jusqu'au 6e mois [25].

6.1.3. Les travaux basés sur la théorie du Fibonacci

- Adda ALI-PACHA, Naima HADJ SAID [26], ont suggéré un nouveau schéma pour le chiffrement d'image basé sur la suite de Fibonacci modifiée. Leur réalisation de cette méthode permet la génération des nombres pseudo aléatoires à l'aide de Fibonacci modifiée qui basée sur la somme de deux graines modulo la valeur maximal désirée, puis faire l'addition entre les nombres pseudo aléatoires et les données de l'image en clair pour obtenir des données cryptée c'est-à-dire image cryptée [21].
- Yicong Zhou, Karen Panetta, Sos Agaian, C.L. Philip Chen [27], ont conçus un nouvel algorithme de cryptage d'image, qui basé sur le code P de Fibonacci pour la décomposition du plan bit-image et la transformée 2D P-Fibonacci pour cryptage d'image car ils dépendent des paramètres [21].
- Weijia Cao, Yicong Zhou, C.L. Philip Chen [28], ont proposé une nouvelle approche pour le chiffrement d'image, qui utilisé Truncated P-Fibonacci et Bit-planes [21].

6.2. Méthode basé sur la théorie du Chaos

6.2.1. Définition

Le chaos est défini par un comportement lié à l'instabilité et à la non-linéarité dans des systèmes dynamiques déterministes. La relation entre l'instabilité et la chaoticité est alors que le système manifeste une très haute sensibilité aux changements de conditions est ce qu'affirmait Henri Poincaré à la fin du 19ème siècle : «Une cause très petite, qui nous échappe, détermine un effet considérable que nous ne pouvons pas ne pas voir, et alors nous disons que cet effet est dû au hasard. (...). Il peut arriver que de petites différences dans les conditions initiales en engendrent de très grandes dans les phénomènes finaux. Une petite erreur sur les

premières produirait une erreur énorme sur les derniers. La prédiction devient impossible et nous avons le phénomène fortuit» [29].

Les cartes chaotiques peuvent être utilisées, dans les applications liées à la sécurité de l'information, pour la génération des clés secrètes dans les algorithmes de cryptage et de tatouage numérique [30].

6.2.2. La carte chaotique logistique (la récurrence logistique)

Une récurrence logistique est un exemple simple de suite dont la récurrence n'est pas linéaire. Souvent citée comme exemple de la complexité pouvant surgir de simple relation non linéaire, cette récurrence fut popularisée par le biologiste Robert May en 1976. Sa relation de récurrence est :

$$X_{n+1} = \mu X_n(1 - X_n) \qquad (2.9)$$

Elle conduit, suivant les valeurs de μ, à une suite convergente, une suite soumise à oscillations ou une suite chaotique [31], comportement selon μ :

Dans le modèle logistique, la variable notée ici X_n désigne l'effectif de la population d'une espèce. En faisant varier le paramètre μ, plusieurs comportements différents sont observés :

- Si $0 \leq \mu \leq 1$, l'espèce finira par mourir, quelle que soit la population de départ.
- Si $1 \leq \mu \leq 3$, la population se stabilisera sur la valeur $\frac{\mu-1}{\mu}$ quelle que soit la population initiale.
- Si $3 < \mu \leq 1 + \sqrt{6}$ (approximativement 3,45), la population oscillera entre deux valeurs. Ces deux valeurs sont indépendantes de la population initiale.
- Si $3{,}45 < \mu < 3{,}54$, la population oscillera entre quatre valeurs, là encore sont indépendantes de la population initiale.
- Si μ est légèrement plus grand que 3,54, la population oscillera entre 8 valeurs, puis 16, 32, etc.
- La plupart des valeurs au-delà de 3,57 présentent un caractère chaotique, mais il existe quelques valeurs isolées de μ avec un comportement qui ne l'est pas. Celles-ci s'appellent parfois les iles de la stabilité. Par exemple autour de la valeur 3,82, un petit intervalle de valeurs de μprésente une oscillation entre trois valeurs et pour μ légèrement plus grand, entre six valeurs, puis douze, etc. ces comportements sont encore indépendants de la valeur initiale.
- Au-delà de $\mu = 4$, la population quitte l'intervalle [0,1] et diverge presque pour toutes les valeurs initiales [31].

Un diagramme de bifurcation permet de résumer tout cela :

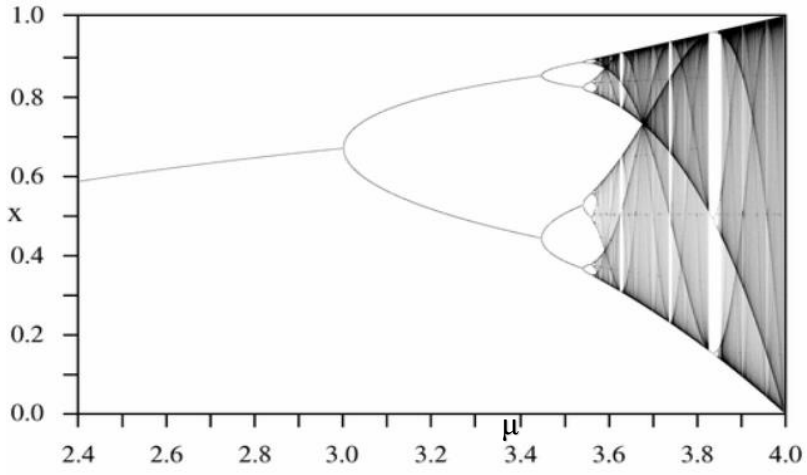


Figure 2.11 : Diagramme de bifurcation de la récurrence logistique [32].

6.2.3. La carte chaotique sine (la récurrence sine)

La récurrence sine d'une (01) dimension a pour représentation d'état :

$$X_{n+1} = \lambda \sin(\pi X_n) \qquad (2.10)$$

Avec λ = 1 le comportement chaotique est généré par une manière très similaire à la fonction logistique.

Comme la récurrence logistique, la carte sine est quadratique au voisinage de x = 0,5. Elles ont une distribution probabiliste et une évolution vers le chaos par doublement de période presque identique. Les fenêtres se produisent périodiquement dans le même ordre. Elle a le même nombre de Feigenbaum que la carte logistique. Malgré les similitudes, il existe quelques différences.

Les bifurcations par doublement de période survenir plus tôt, et les fenêtres périodiques sont plus larges par rapport à la carte logistique [31].

6.2.4. La carte chaotique standard (la récurrence standard)

La récurrence standard de deux (02) dimensions a pour représentation d'état :

$$\begin{cases} X_{n+1} = X_n + K\, sin(Y_n) \\ Y_{n+1} = Y_n + X_{n+1} \end{cases} \qquad (2.11)$$

Pour K = 0, la carte n'est pas linéaire et seules les orbites périodiques et quasi-périodiques existent. Lorsqu'elles sont tracées dans l'espace des phases, les orbites périodiques apparaissent comme des courbes fermées, et les orbites quasi-périodiques comme des petites courbes fermées dont leurs centres se situent dans une autre courbe fermée plus grande. Ces types

d'orbites sont observés suivant les conditions initiales utilisées. La non-linéarité de la carte est augmentée lorsque k augmente [31].

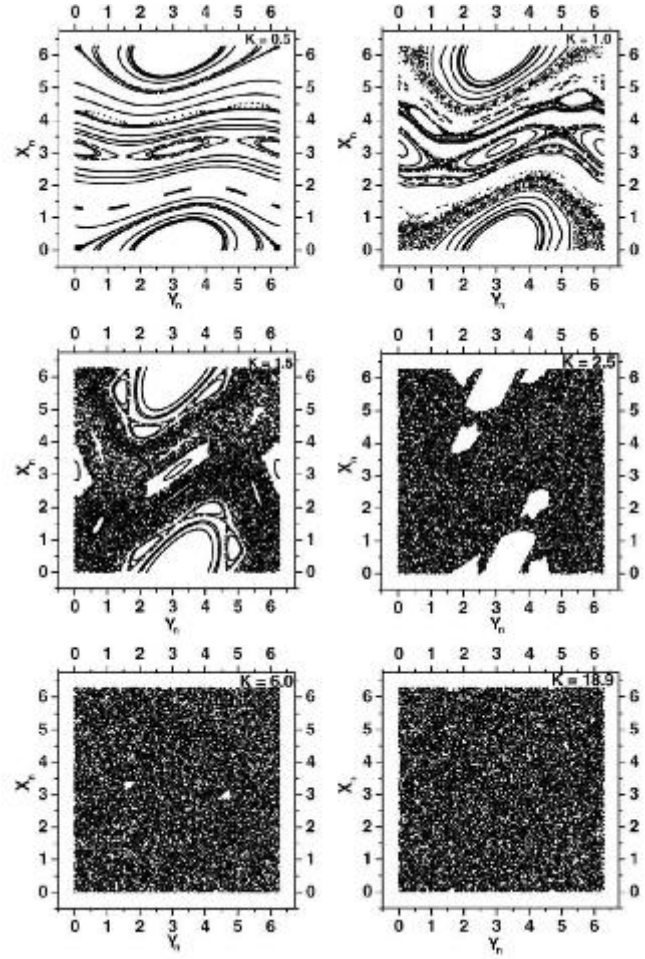


Figure 2.12 : L'espace de phase de la carte standard pour K = 0.5, 1.0, 1.5, 2.5, 6.0 et 18.9 [31].

6.2.5. Les travaux basés sur la théorie de chaos

- Tiegang Gao et Zengqiang Chen [33] ont suggéré un nouveau schéma de cryptage d'image. Le cryptage proposé ici se compose de deux processus, premièrement, ils mélangent l'image en fonction d'une matrice globale de brassage généré en utilisant la carte logistique, puis ils cryptent l'image mélangée en utilisant l'hyper-chaos [21].
- Baydda Flaeh AL-Saraji et Mustafa Dhiaa AL-Hassani [34] ont conçus un nouvel algorithme de cryptage d'image, Leurs recherche visent à améliorer le niveau de sécurité et le secret fourni par le chiffrement qui basé sur carte chaotique. Un générateur de flux de clés N-array est proposé dans ce travail, qui est basé sur des cartes de logistique multiple pour générer les clés de chiffrement et la matrice dynamique en utilisant LFSR pour augmenter le caractère aléatoire de l'image [21].

- G.A.Sathishkumar et Dr.K.Bhoopathy bagan et Dr.N.Sriraam [35] ont proposé une nouvelle approche pour le chiffrement d'image. Leur algorithme proposé décrit comme les suivants : Tout d'abord, une paire de sous-clés est donnée en utilisant des cartes logistiques chaotiques. Deuxièmement, l'image est cryptée à l'aide de la carte logistique sous-clé et dans sa transformation conduit à un processus de diffusion. Troisièmement, les clés secondaires sont générées par quatre différentes cartes chaotiques et les images sont traitées comme un tableau 1D en effectuant un balayage Raster et un balayage en Zigzag. Les tableaux numérisés sont divisés en divers sous blocs. Ensuite, pour chaque sous-bloc, la permutation de position et la transformation de valeur sont effectuées pour produire l'image cryptée [21].

6.3. Méthode basé sur la permutation

La permutation d'image basée sur le changement de place d'une partie de l'image, il existe trois techniques de ce mécanisme :

6.3.1. Permutation binaire (permutation des bits)

L'image peut être considérée comme un tableau de pixels, chacun avec huit bits pour 256 niveaux de gris. Dans cette technique de permutation les bits de chaque pixel pris de l'image sont permutés avec la clé choisie à partir de l'ensemble de touches à l'aide du générateur d'index Pseudo aléatoire. Toute la gamme de ces les pixels permutés forme l'image chiffrée. L'image chiffrée obtenue à partir du bit la technique de permutation est transmise au récepteur par le canal non sécurisé. Au récepteur l'image chiffrée est décryptée à l'aide du même jeu de clés. Comme le nombre de bits dans chaque pixel est de huit, nous prenons également la longueur de clé égale à huit. Le nombre de permutations obtenues avec huit éléments est 8 ! = 40320 permutation possible [36].

00 01 10 11	27
00 01 11 10	30
00 10 01 11	39
00 10 11 01	45
00 11 01 10	54
00 11 10 01	57
01 00 10 11	75
01 00 11 10	78
01 10 00 11	99
01 10 11 00	108
01 11 00 10	114
01 11 10 00	120
10 00 01 11	135
10 00 11 01	141
10 01 00 11	147
10 01 11 00	156
10 11 00 01	177
10 11 01 00	180
11 00 01 10	198
11 00 10 01	201
11 01 00 10	210
11 01 10 00	216
11 10 00 01	225
11 10 01 00	228

Figure 2.13 : Exemple sur la permutation des bits [37].

6.3.2. Permutation par pixel

Dans ce schéma, chaque groupe de pixels est extrait de l'image. Les pixels du groupe sont permutés en utilisant la touche sélectionnée à partir de l'ensemble de touches. La procédure de cryptage et de décryptage est la même que la technique de permutation des bits. La taille du groupe de pixels est identique à la longueur des clés, et toutes les clés sont de même longueur. Si la longueur des touches est supérieure à la taille du groupe de pixels, le l'information de perception diminue [36].

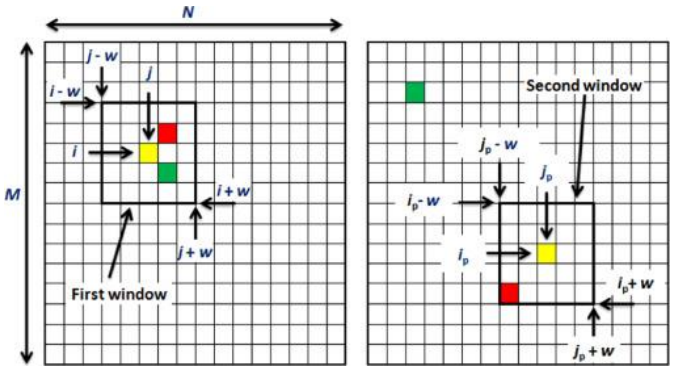


Figure 2.14 : Exemple sur la permutation pixel [38].

6.3.3. Permutation par bloc

Dans cette technique, l'image peut être décomposée en blocs. Un groupe de blocs est tiré de la l'image et ces blocs sont permutés mêmes que les permutations de bits et de pixel. Pour un meilleur cryptage la taille du bloc doit être inférieure. Si les blocs sont très petits, alors les objets et ses bords n'apparaissent clairement. Dans ce bloc de permutation les blocs sont permutés horizontalement dans l'image. La permutation des blocs le long du côté vertical est également semblable à la permutation horizontale de bloc latéral [36].

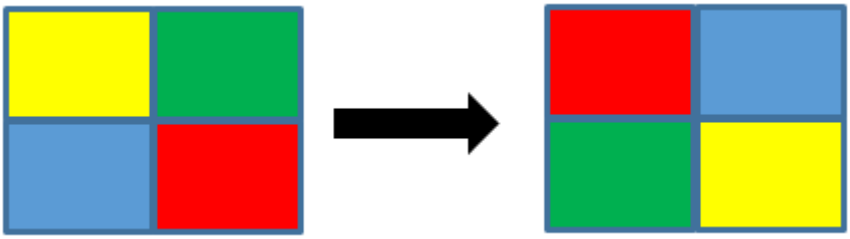


Figure 2.15 : Exemple sur la permutation par bloc.

6.3.4. Les travaux basés sur la permutation

- Avi Dixit, Pratik Dhruve and Dahale Bhagwan [36], présentent une méthode de permutation basée sur les trois technique (bit, pixel, bloc).
- Sesha Pallavi Indrakanti, P.S.Avadhani [39] ont suggéré un nouveau schéma de cryptage d'image basé sur la technique de permutation par bloc avec plusieurs formes (niveaux) Successives.
- Ravi Prakash Dewangan, Chandrashekhar Kamargaonkar [40] ont proposé une nouvelle méthode basée sur les trois techniques de permutation (bit, pixel, bloc) avec l'utilisation du maximum de combinaison entre eux.

6.4. Autres méthodes

Plusieurs algorithmes de chiffrement d'image existant ont été proposés à partir de différents technologie, tel que : le séquençage de l'ADN, l'automate cellulaire. Et de différents domaines tel que physique, biologie, …

7. Conclusion

Dans ce chapitre, nous avons parlé des méthodes plus importantes et la plus récentes pour crypter les images numériques.

Dans le prochain chapitre, nous allons parler sur notre plan et comment nous avons développé une méthode pour crypter les images numériques.

CHAPITRE 3

MÉTHODE PROPOSÉE

1. Introduction

Les chercheurs de cryptographie ont été proposés plusieurs techniques de chiffrement d'images numériques. Parmi eux il y a des algorithmes qui basés sur les théories comme la théorie de chaos, Fibonacci, la permutation, et aussi des algorithmes qui basés sur différentes technologies comme : le séquençage de l'ADN, l'optique, l'automate cellulaire et la transformation de Fourier, et beaucoup d'autres techniques

Dans ce chapitre nous allons présenter notre algorithme de cryptage que nous avons développé. Cet algorithme de chiffrement proposé est basé sur l'hybridation entre plusieurs techniques de cryptage comme les carte chaotiques sine, logistique et standard, la suite de Fibonacci Modifiée et les techniques de permutation, autrement dit c'est la combinant les propriétés et les avantages entre eux. Afin d'améliorer ses performances en terme de espaces de clés et empêche l'analyse par force brute. Les résultats de la simulation montrent l'efficacité et la sécurité de notre système proposé.

2. Méthode proposée

Dans le schéma proposé, il existe trois étapes dans le cas de chiffrement :

Étape 01 **:** nous avons utilisé trois algorithmes qu'utilisent les formules mathématiques du carte chaotique sine, suite de Fibonacci Modifiée et carte chaotique standard(pour la confusion) pour générer un flux de clés pseudo aléatoire avec même taille d'image, puis faire l'opération XOR élément par élément entre image en clair et le flux de clés pseudo aléatoire généré, afin d'obtenir une image sortie qu'est l'entrée de l'étape 02.

Étape 02 **:** nous avons utilisé des permutations qu'utilisent les permutations par pixel et par bloc de forme aléatoire en plusieurs niveaux pour obtenir une clé aléatoire de 24 caractères, afin d'obtenir une image sortie qu'est l'entrée de l'étape 03.

Étape 03 **:** nous avons utilisé trois algorithmes qu'utilisent les formules mathématiques du carte chaotique sine, carte chaotique logistique et carte chaotique standard (pour la confusion) pour générer un flux de clés pseudo aléatoire avec même taille d'image, puis faire l'opération XOR élément par élément entre image entrée et le flux de clés pseudo aléatoire généré, afin d'obtenir une image cryptée.

Le déchiffrement se compose également de trois étapes, mais dans l'ordre inverse.

Le but principal de ce chiffrement c'est faire un confusion-diffusion le plus possible pour garantir l'efficacité de l'algorithme.

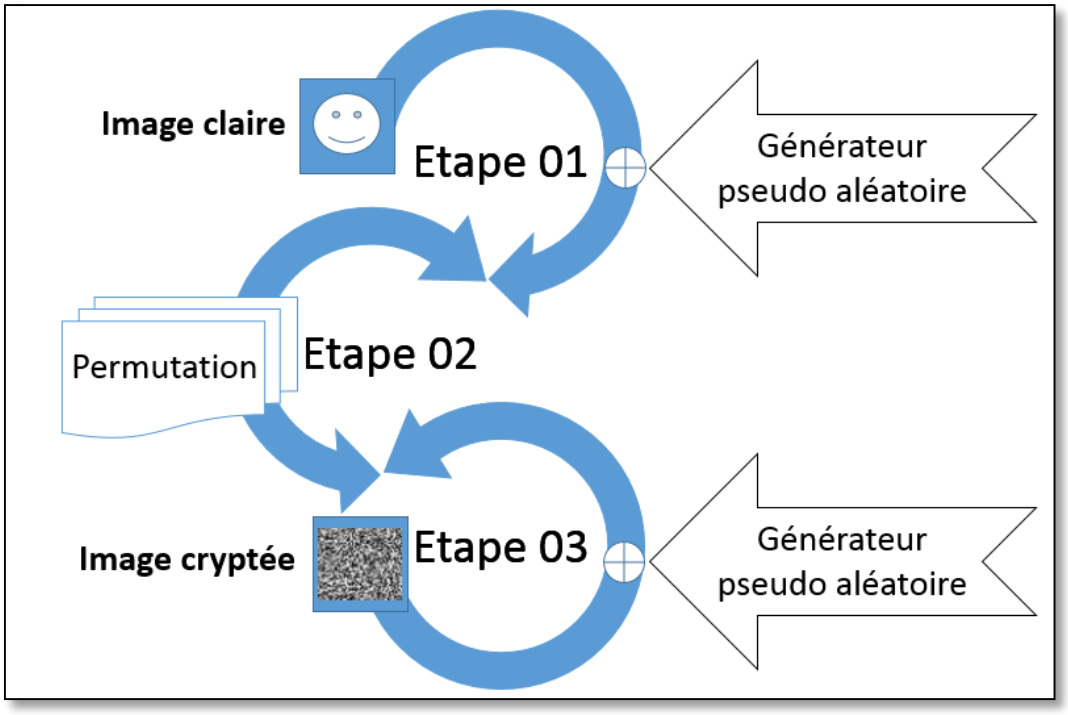


Figure 3.1 : Schéma de chiffrement proposé (symétrique).

2.1. Fonction de chiffrement

2.1.1. Génération un flux des clés pseudo-aléatoires

Les générateurs de clés pseudo-aléatoires (clé1, clé2) sont réalisés par :

Un générateur chaotique de la carte chaotique standard pour choisir entre deux générateurs (le premier générateur est le générateur de la carte chaotique sine et le deuxième est le générateur de la suite de Fibonacci modifiée) pour la clé1, et (le premier générateur est le générateur de la carte chaotique sine et le deuxième est le générateur de la carte chaotique standard) pour la clé2.

Le générateur pseudo-aléatoire clé1 :

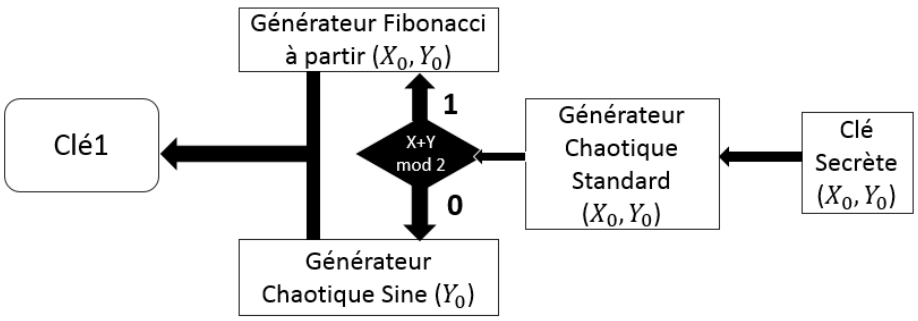


Figure 3.2 : Le générateur pseudo-aléatoire clé1.

Le générateur pseudo-aléatoire clé2 :

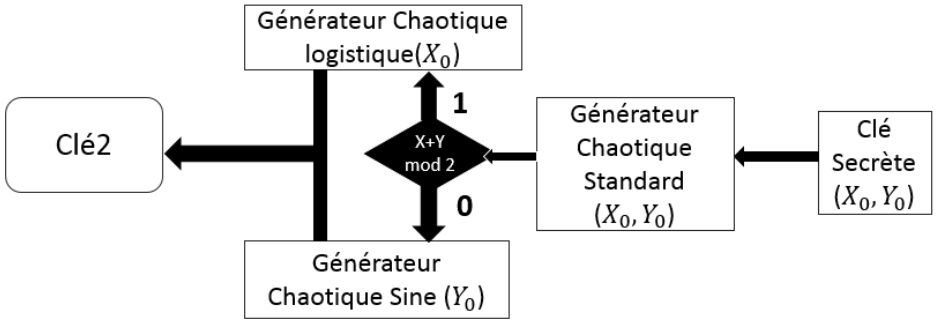


Figure 3.3 : Le générateur pseudo-aléatoire clé2.

Et ses formules mathématiques sont les suivants :

1) Le générateur pseudo-aléatoire de la carte chaotique sine qu'utilise l'équation (2.10) :

$$Y_{n+1} = \lambda \sin(\pi Y_n)$$

Et les paramètres initiales sont ($Y_0,\ \lambda$).

2) Le générateur pseudo-aléatoire de la carte chaotique logistique qu'utilise l'équation (2.9) :

$$X_{n+1} = \mu X_n(1 - X_n)$$

Et les paramètres initiales sont ($X_0,\ \mu$).

3) Le générateur pseudo-aléatoire de la carte chaotique standard qu'utilise l'équation (2.11) :

$$\begin{cases} X_{n+1} = X_n + K\ sin(Y_n) \\ Y_{n+1} = Y_n + X_{n+1} \end{cases}$$

Et les paramètres initiales sont (X_0, Y_0, K).

4) Le générateur pseudo-aléatoire de la suite de Fibonacci modifiée qu'utilise l'équation (2.8) :

$U_0 = (X_0 \times M) Mod\ M, U_1 = (Y_0 \times M) Mod\ M$ Et

$U_n = U_{n-1} + U_{n-2}, n \geq 2\ \ et\ M = 256$

Et les paramètres initiales sont (X_0, Y_0).

2.1.2. Étape 01 : le chiffrement à l'utilisation de clé1

1) Génération d'un flux pseudo aléatoire :
   a) Définition des paramètres initiaux ($X_0, Y_0, \mu,\ \lambda, K$).
   b) Génération de trois flux de nombre pseudo aléatoire à travers les formules mathématiques de Suite de Fibonacci Modifiée (K_1), de Chaotique Carte Logistique (K_2) et de Chaotique Carte sine (K_3), mais à condition que chaque flux généré doive être même taille d'image.
   c) Convertir de chaque valeur de K_2 et K_3 à valeur entier par $K_2 * 256$ et $K_3 * 256$.

d) Génération des clés clé1 et clé2 à partir des flux générés (K_1), (K_2) et (K_3).

e) Convertir les clés clé1 et clé2 sous forme des bits.

2) Convertir l'image en clair à un flux de données sous forme des bits m_i.

3) Faire la combinaison OU-exclusif (ou XOR) bit par bit entre le flux de données (l'image en clair) et flux de clé pseudo aléatoire clé1. Pour obtenir un flux de données chiffrés (image chiffrée C_i) $C_i = \oplus$ Clé1.

2.1.3. Étape 02 : le chiffrement en l'utilisation des techniques de permutation

1) Prendre une clé nulle (clé de permutation) à la fin de cette étape, cette clé sera composée de 24 caractères.

2) Permutation pixel : Choisir un caractère c aléatoirement et l'ajouter à la (clé de permutation) et prendre sa valeur *Vc* comme un paramètre pour appliquer une permutation pixel et notre algorithme de permutation pixel est comme suite :

```
pour i=0 à largeur
pour j=0 à hauteur
{
image_sortie[i][j]=image[(i+Vc)Mod(largeur)][(j+Vc*2) Mod(hauteur)];
}
```

3) Appliquer la permutation pixel sur l'image C_i.

4) Diviser l'image C_i sur 4 blocs.

5) Diviser chaque bloc sur 4 sous-blocs.

6) Applique la permutation pixel sur les 16 sous-blocs.

7) Diviser chaque sous-bloc sur autre 4 sous-blocs.

8) Appliquez une permutation par bloc pour chaque sous-bloc aléatoirement (16 cas) chaque forme de permutation a son propre code (caractère), et ajouter les 16 caractères à la clé de permutation.

9) Appliquez une permutation par bloc pour chaque bloc aléatoirement (4 cas) chaque forme de permutation a son propre code (caractère), et ajouter les 4 caractères à la clé de permutation.

10) Appliquez une permutation par bloc sur l'image C_i aléatoirement et ajouter le caractère à la clé de permutation.

11) décalage des lignes et des colonnes : Choisir deux caractères c_ligne et c_colonne aléatoirement et les ajouter à la (clé de permutation) et prendre leurs valeurs Vc et Vl

comme paramètres pour appliquer l'algorithme de décalage des lignes et des colonnes comme suit :

lignes:

Début

Cpt=0, lgn=0;
pour j=0 à hauteur, faire
{
si (lgn< hauteur) alors
{
pour i=0 à largeur, faire
{
image_sortie[i][j]=image[i][lgn];
}
lgn+=Vl;
}
sinon
{
Cpt++;
j--;
lgn=Cpt;
}}
fin

Colonnes:

Cpt=0, col=0;
pour i=0 à largeur, faire
{
si (col< largeur) alors
{
pour j=0 à hauteur, faire
{
image_sortie[i][j]=image[col][j];
}
col+=Vc;
}
sinon
{
Cpt++;
i--;
col=Cpt;
}
fin

12) enfin on a l'image C_j comme sortie de l'étape 02 et la clé de permutation qui contient 24 caractères.

2.1.4. Étape 03 : le chiffrement à l'utilisation de clé2

1) Convertir l'image d'entré C_j à un flux de donnée sous forme des bits.
2) Faire la combinaison OU-exclusif (ou XOR) bit par bit entre le flux de données (l'image C_j) et flux de clé pseudo aléatoire clé2, pour obtenir un flux de données chiffrés (image chiffrée C_f) $C_f = C_j \oplus$ Clé2.

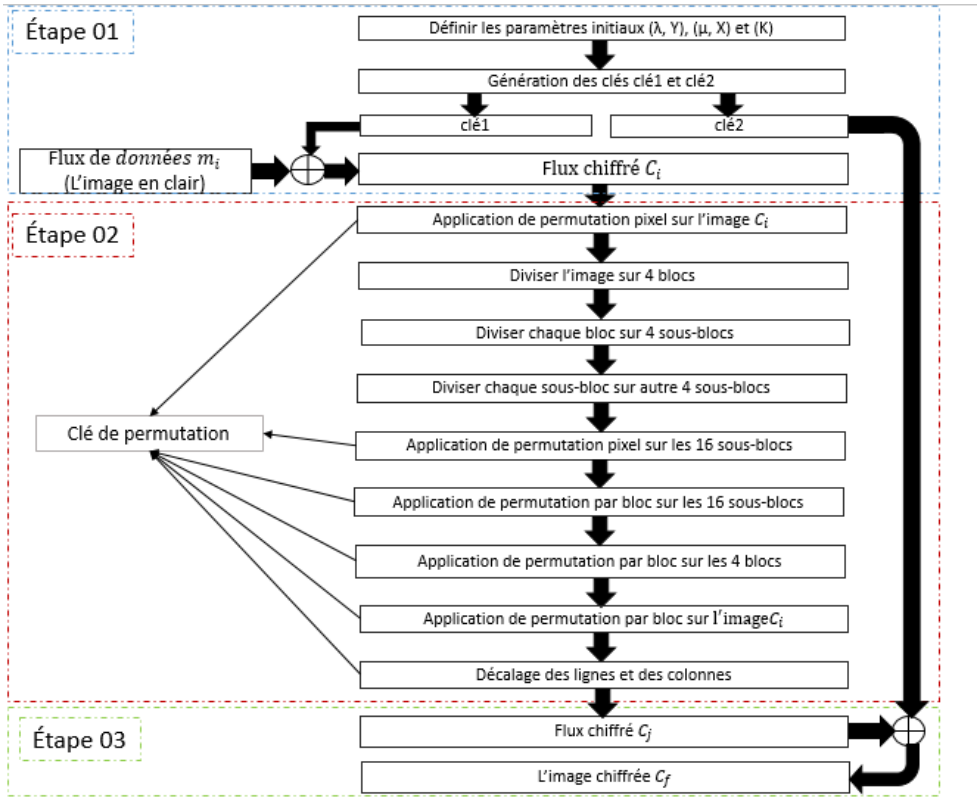


Figure 3.4 : Les étapes de chiffrement**.**

2.2.Fonction de déchiffrement

Le déchiffrement se compose également de trois étapes, mais dans l'ordre inverse, les paramètres initiaux (λ,Y_0,), (μ,X_0) et (K) avec la clé de permutation doit être le même lorsque utilisé dans la fonction de chiffrement, et la génération des clés est la même.

2.2.1. Génération un flux de clés pseudo-aléatoire

Comme le cas de chiffrement, et l'entré c'est l'image C_f.

2.2.2. Étape 01 : le déchiffrement à l'utilisation de clé2

L'image C_j est la sortie talque **:** $C_j = C_f \oplus$ Clé2**.**

2.2.3. Etape 02 : le déchiffrement à l'utilisation des techniques de permutation

1) Le décalage des lignes et des colonnes avec l'algorithme et l'ordre inverse.
2) Appliquer la permutation de bloc l'inverse sur l'image C_j.
3) Diviser l'image C_j sur 4 blocs.
4) Appliquer la permutation de bloc l'inverse sur les blocs.
5) Diviser chaque bloc sur 4 sous-blocs.
6) Diviser chaque sous-bloc sur autre 4 sous-blocs.
7) Appliquer la permutation de bloc l'inverse sur les sous-blocs.
8) Appliquer la permutation de pixel l'inverse sur les sous-blocs.

9) Appliquer la permutation de pixel l'inverse sur l'image C_j.

10) enfin on a l'image C comme sortie de l'étape 02.

2.2.4. Étape 03 : le déchiffrement à l'utilisation de clé1

La sortie de cette étape c'est l'image m_i et $m_i = C_i \oplus$ Clé1.

m_i : est l'image en clair.

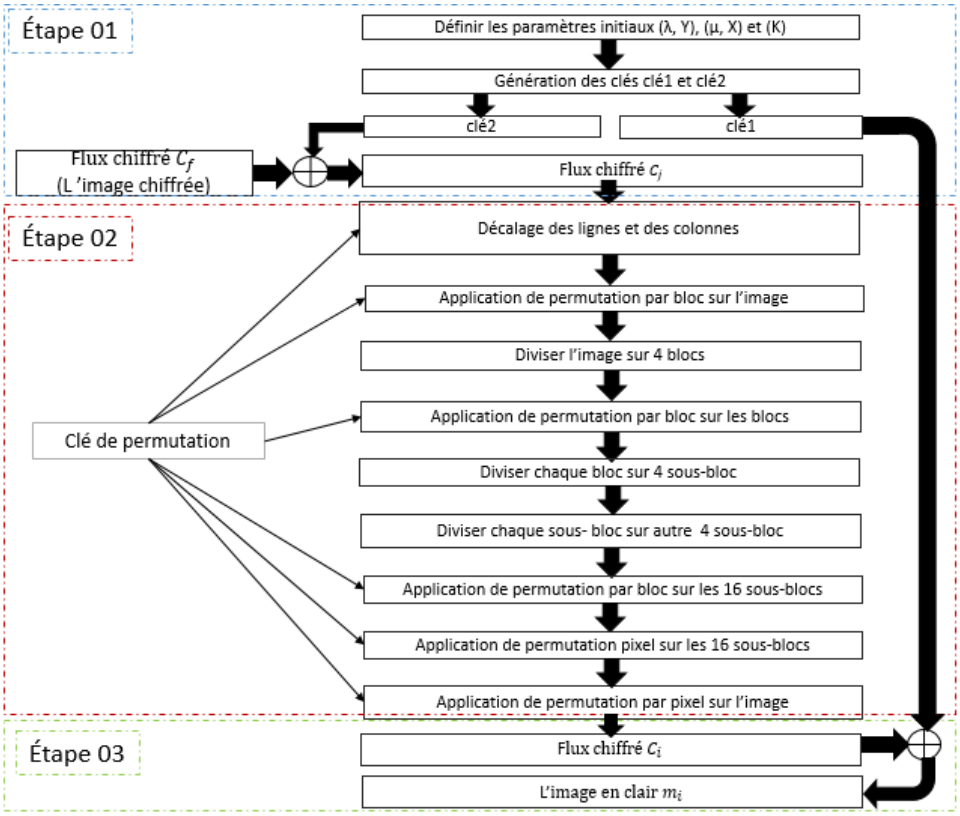


Figure 3.5 : Les étapes de déchiffrement**.**

3. Résultats expérimentaux

3.1. Environnement de développement

L'application a été créa depuis un PC TOSHIBA SATELLITE C660 :

- Mémoire : 8192 MB RAM.
- Processeur : Intel ® Core™ i5-2410M CPU @ 2.30 GHZ (4 CPUs).
- Système d'exploitation : Windows 8.1 Pro 64 bits.
- Carte Graphique : NVIDIA GeForce 315M.

3.2. Langage de programmation

Nous avons choisi le langage ***JAVA*** pour développer notre système. Ce choix de langage est motivé par les raisons suivantes :

- Java est organisée (orienté objet), et il contient des classes bien conçu et bien reparties.
- Java est connu et donc plus de chance de trouver des développeurs java, pour concevoir ou amélioré une application.
- Java est gratuite (open source).
- Java est portable (donc exécutable sur n'importe quel système, à condition d'avoir installé une JVM).

Nous avons exploité l'environnement de programmation ***Netbeans*** IDE. Et utilisé l'environnement ***SWING*** pour la réalisation de l'interface graphique.

Les bibliothèques utilisées :

- jfreechart-1.0.19.
- jai_imageio-1.1.
- jDeli.

3.3. Les interfaces du logiciel développé

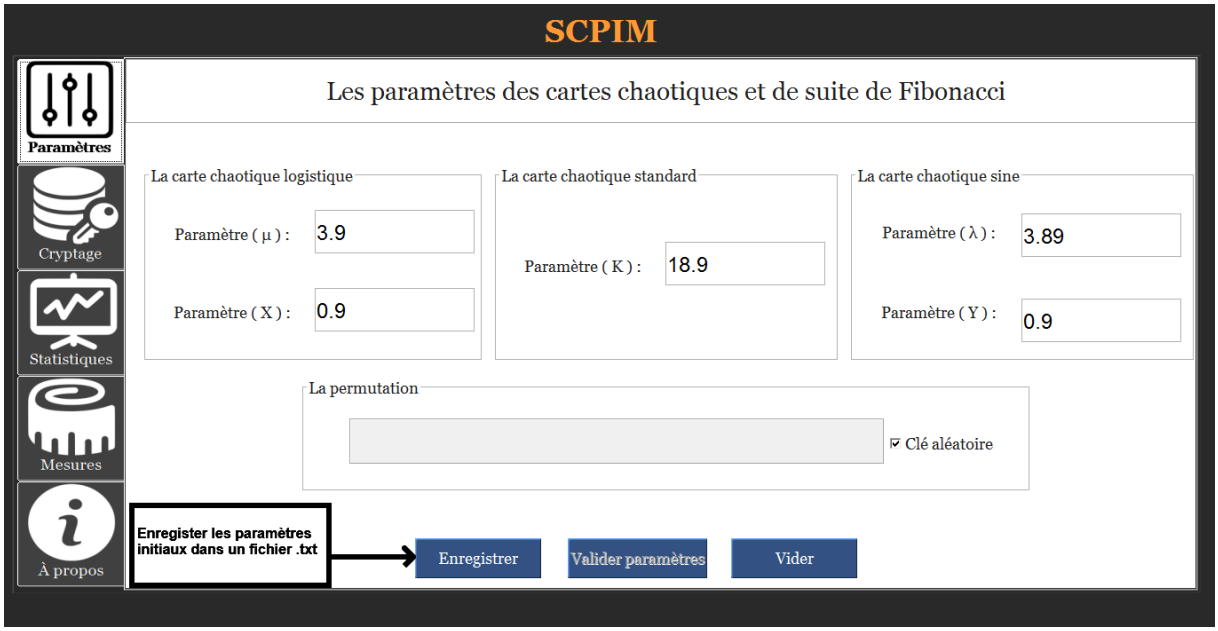


Figure 3.6 : Forme des paramètres**.**

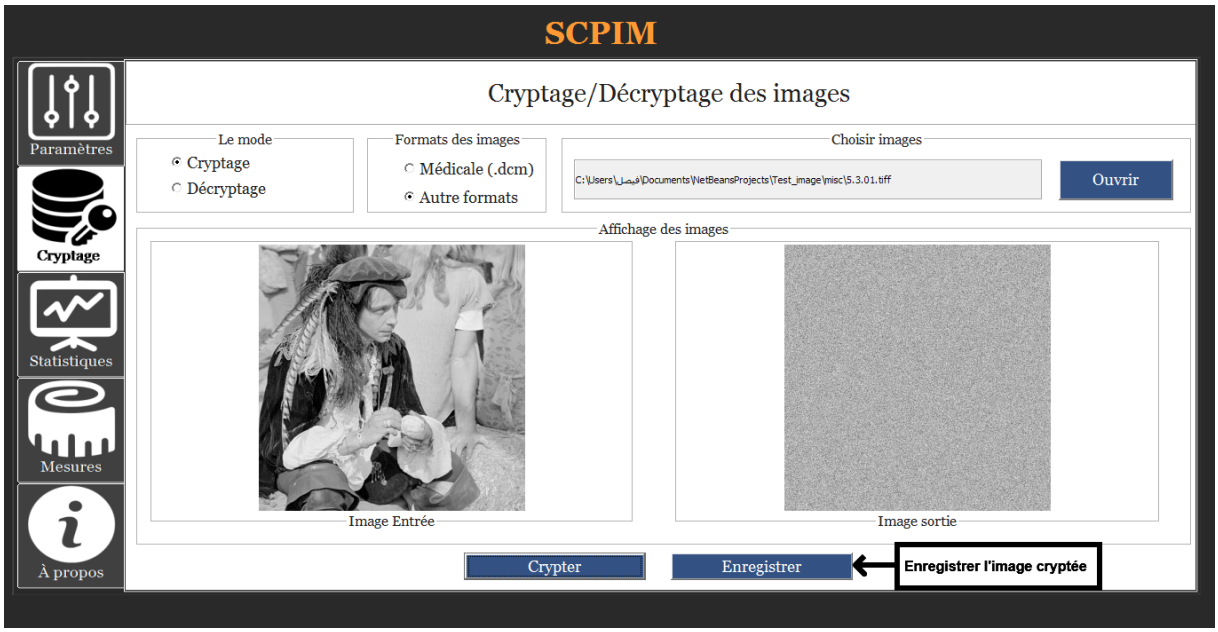


Figure 3.7 : Forme de Cryptage (Mode cryptage)**.**

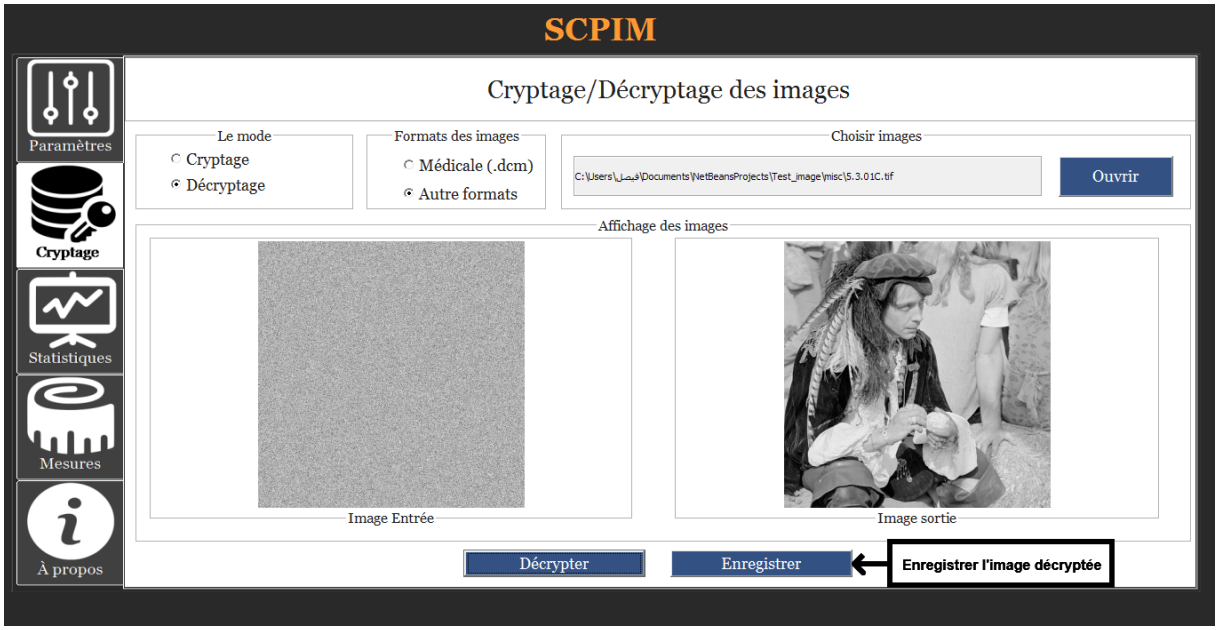


Figure 3.8 : Forme de Cryptage (Mode décryptage)**.**

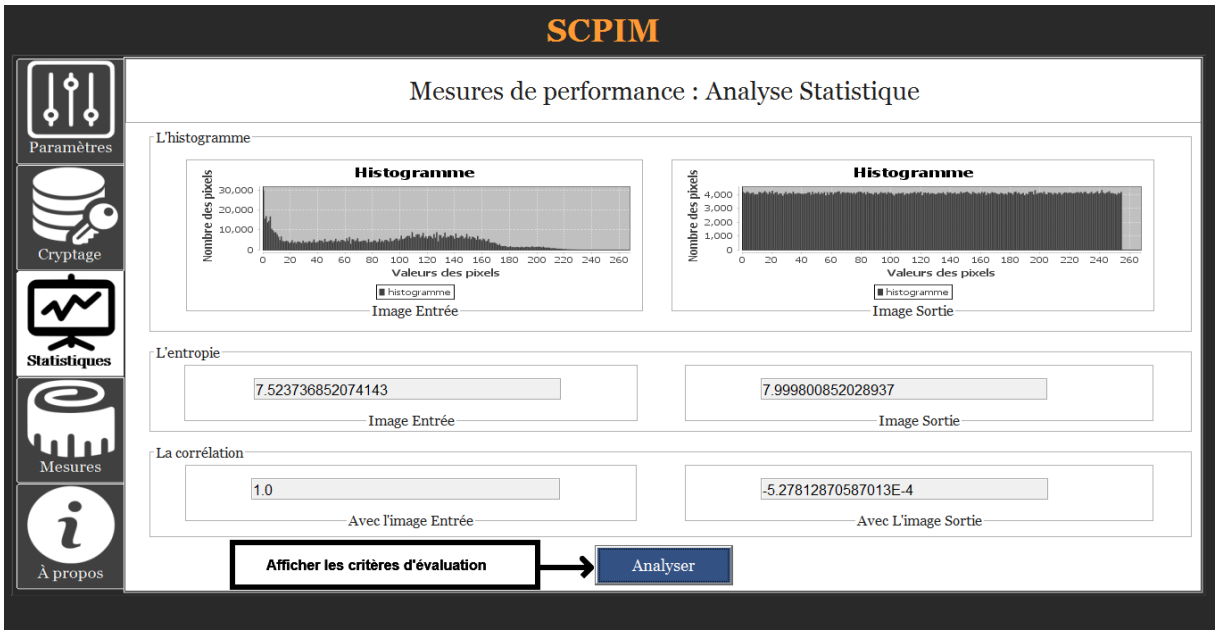


Figure 3.9 : Forme d'évaluation : Analyse statistique**.**

Figure 3.10 : Forme d'évaluation : Analyse de sensitivité**.**

3.4. Les données utilisées

Les données utilisées dans notre mémoire, est une base de données d'images, Ils sont disponibles gratuitement sur les sites Web suivantes : University of Southern California [41], Et University of Waterloo [42], Et le dernier University of Wisconsin-Madison [43]. Ces images sont conçues pour le traitement et d'analyse d'un cryptage d'images numériques.

3.5. Images niveau de gris

Des simulations numériques ont été faites pour confirmer les bonnes performances de notre schéma. Les figures au-dessous montrent plusieurs images au niveau de gris de différentes tailles sont cryptées en utilisant l'algorithme proposé.

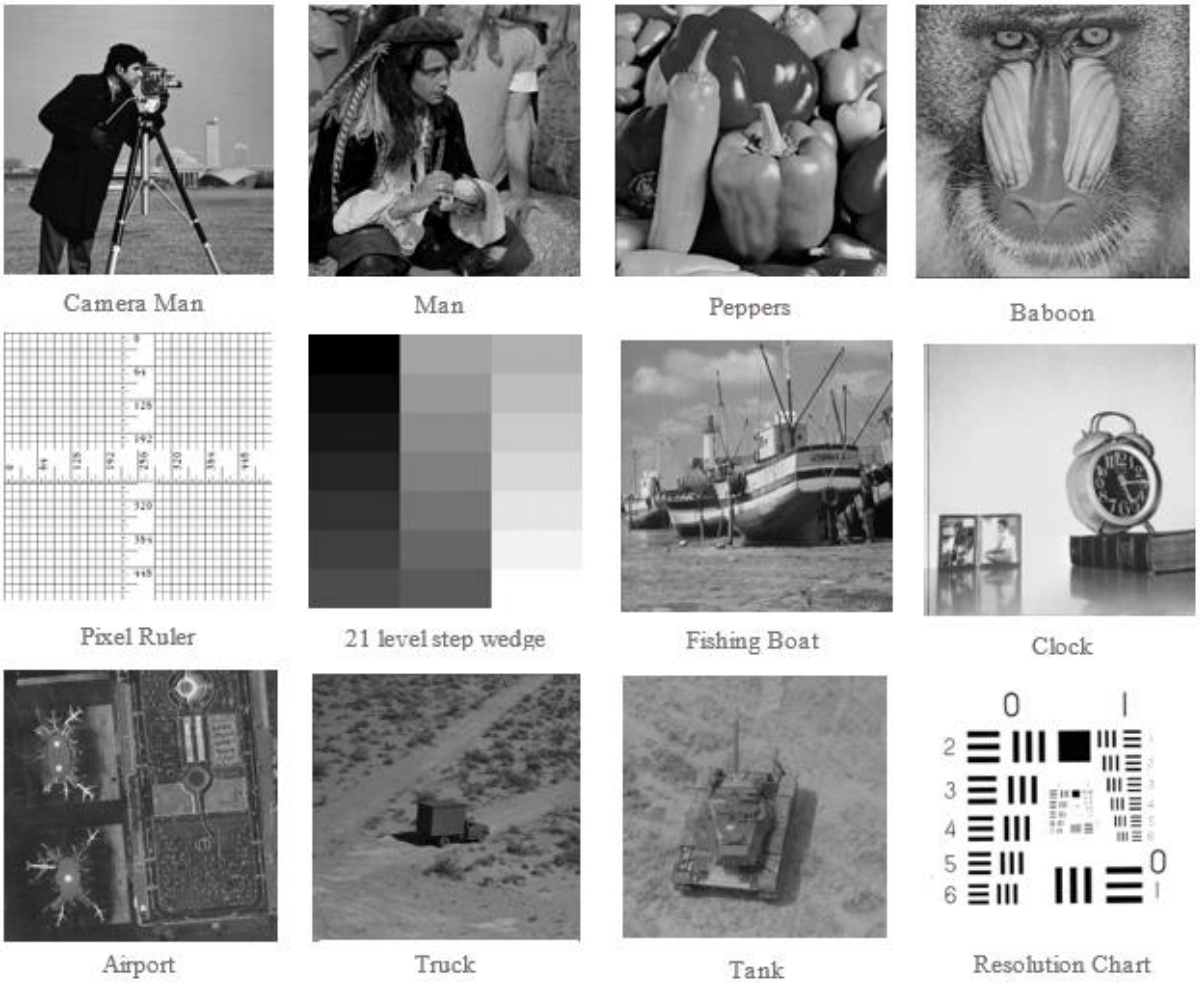


Figure 3.11 : Les images originales.

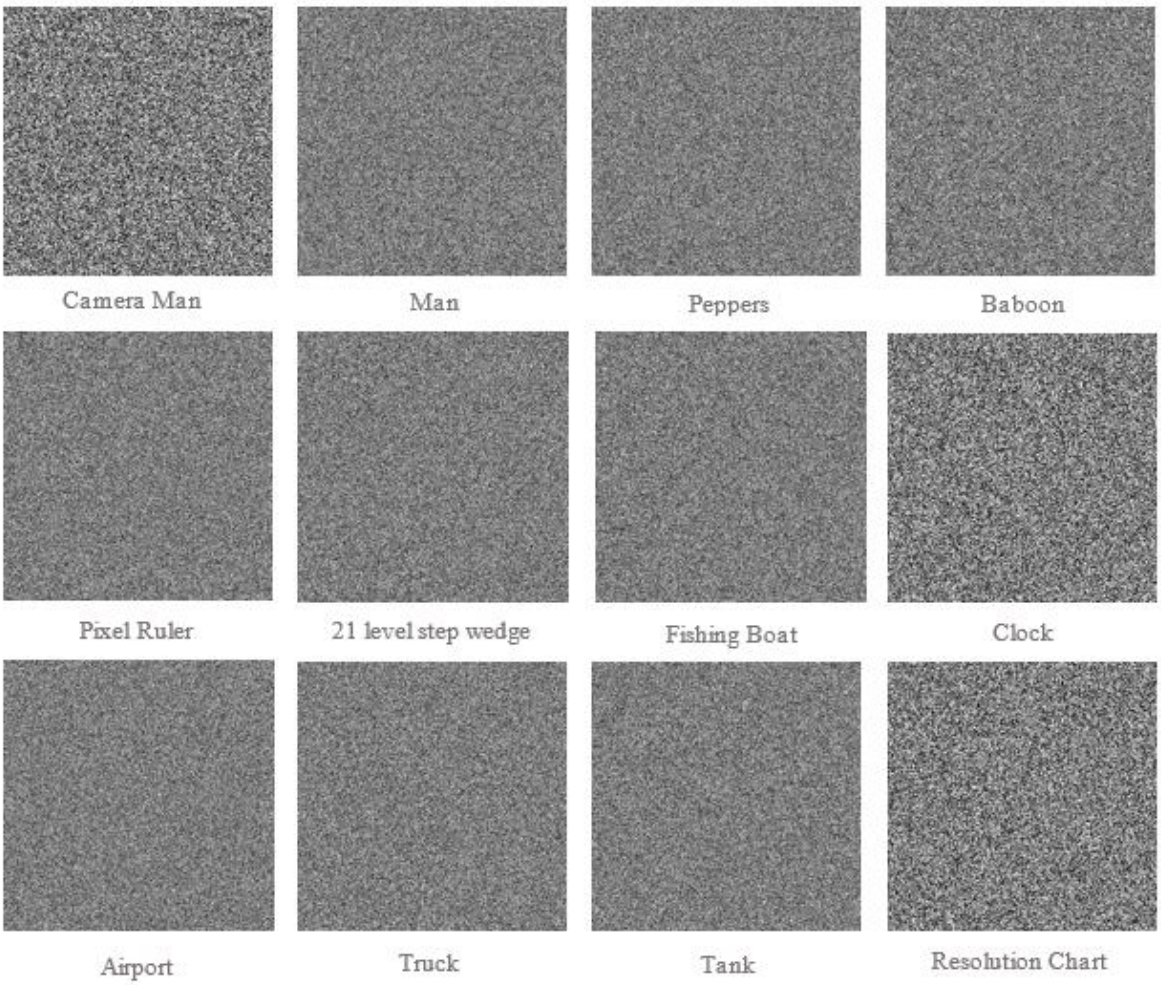


Figure 3.12 : Les images cryptées**.**

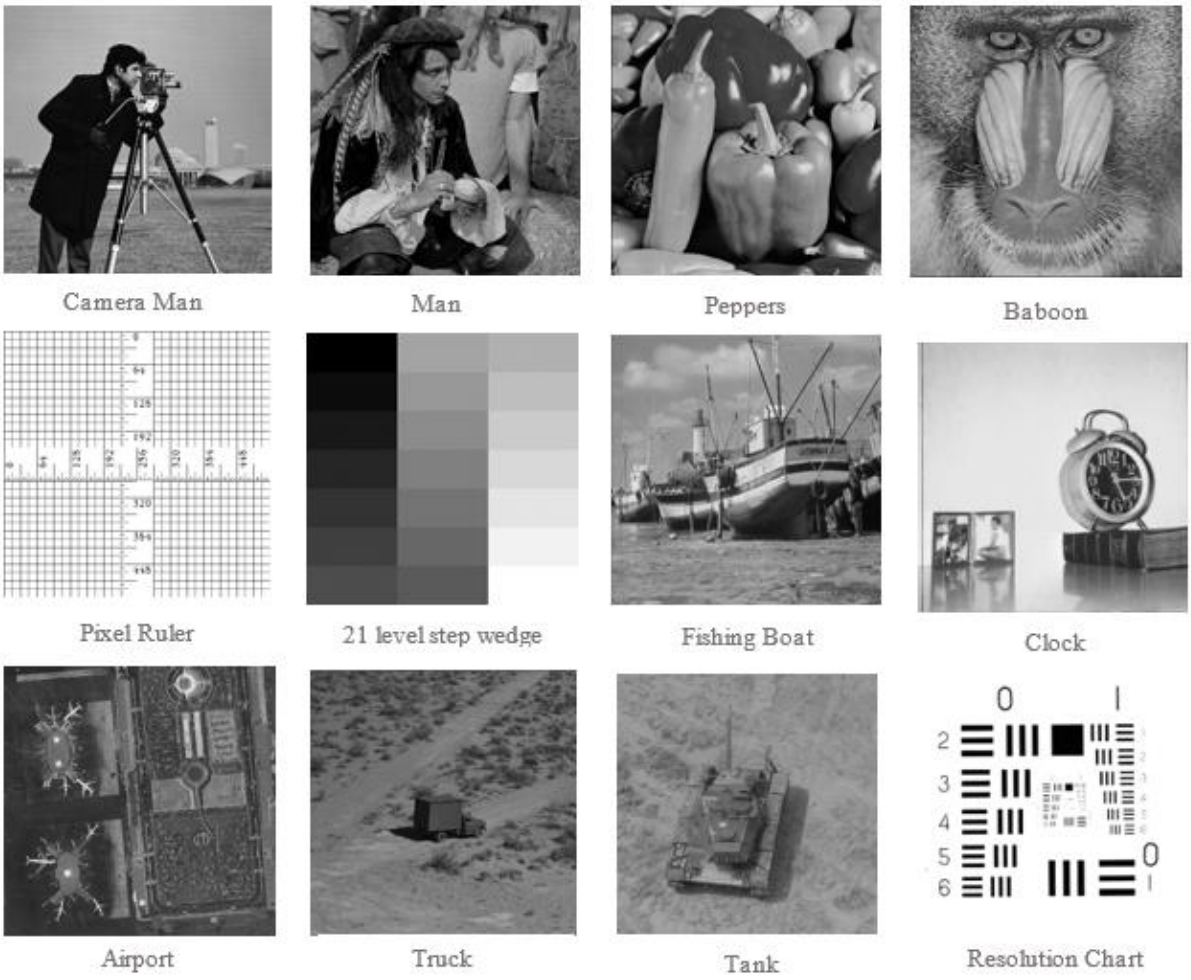


Figure 3.13 : Les images décryptées.

3.6. Images médicales

Après l'application sur les images normales et après avoir obtenu des bons résultats, nous avons appliqué notre algorithme sur les images médicales que sont notre objectif principal.

Récemment, les images médicales ont été envoyées largement sur les réseaux et l'Internet et pour cela nous avons développé notre système essentiellement orienté vers les images médicales du format DICOM, les images médicales enregistrées en tant qu'images normales. Les résultats ont été les suivants :

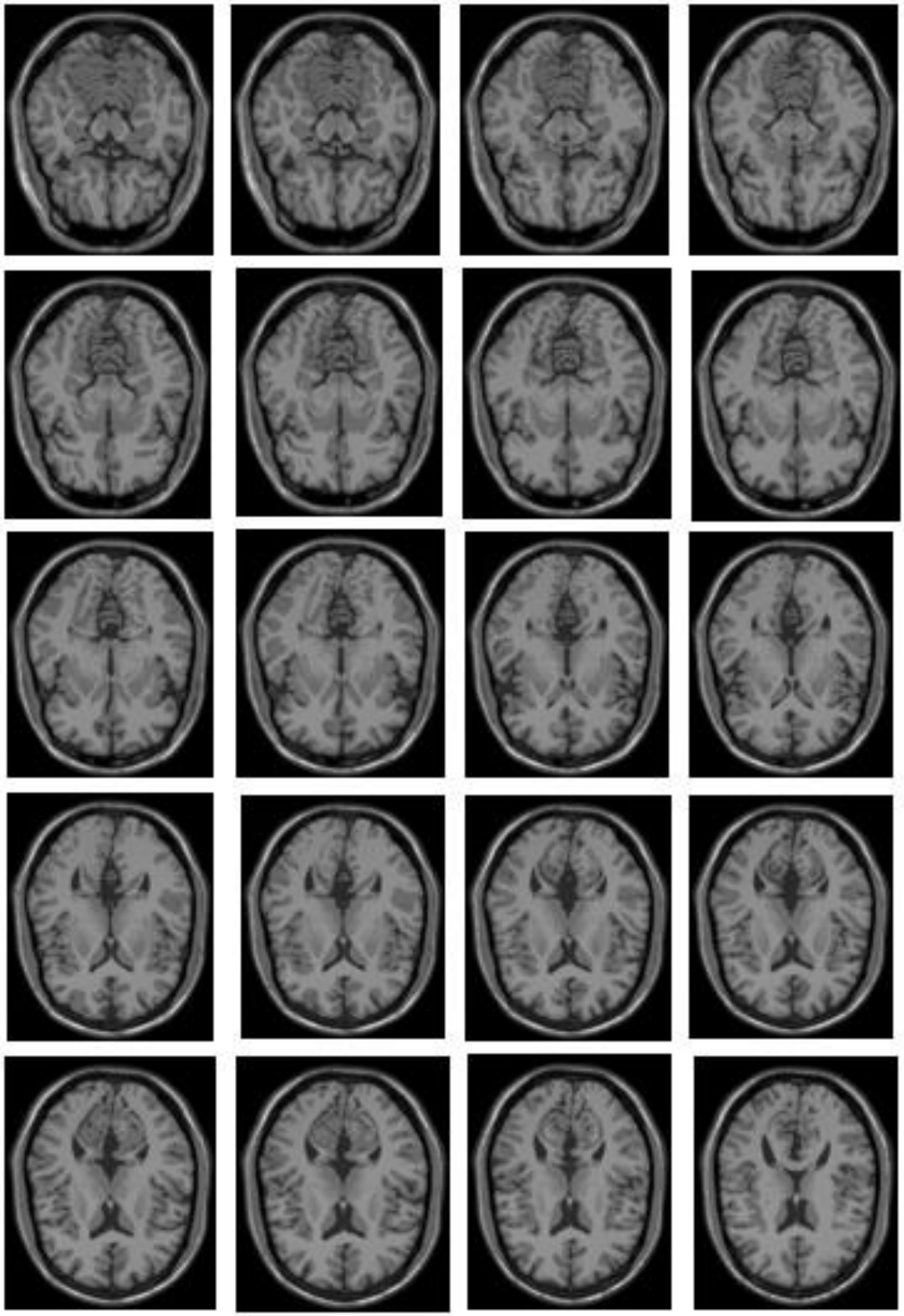


Figure 3.14 : Les images médicales originales.

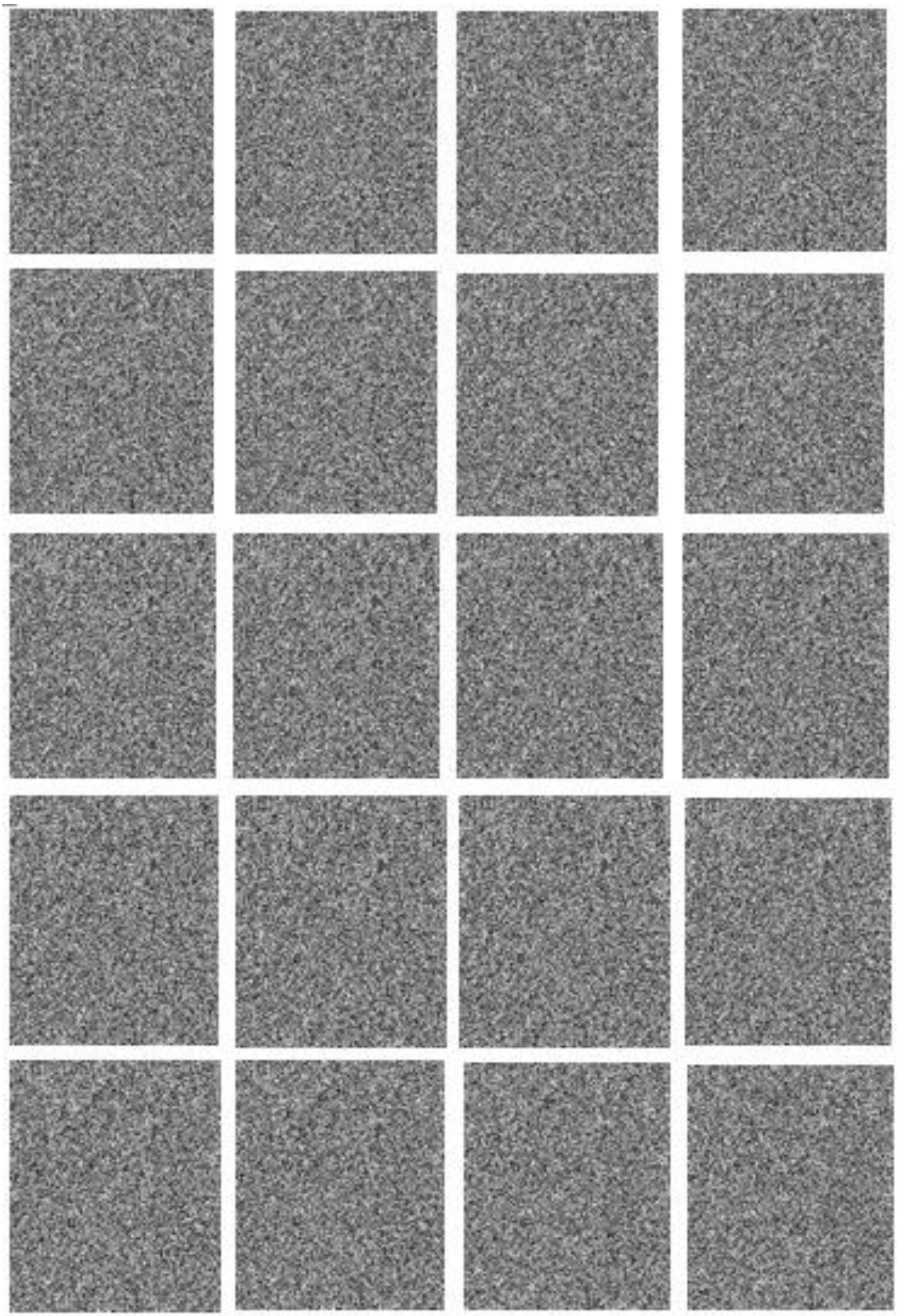


Figure 3.15 : Les images médicales cryptées.

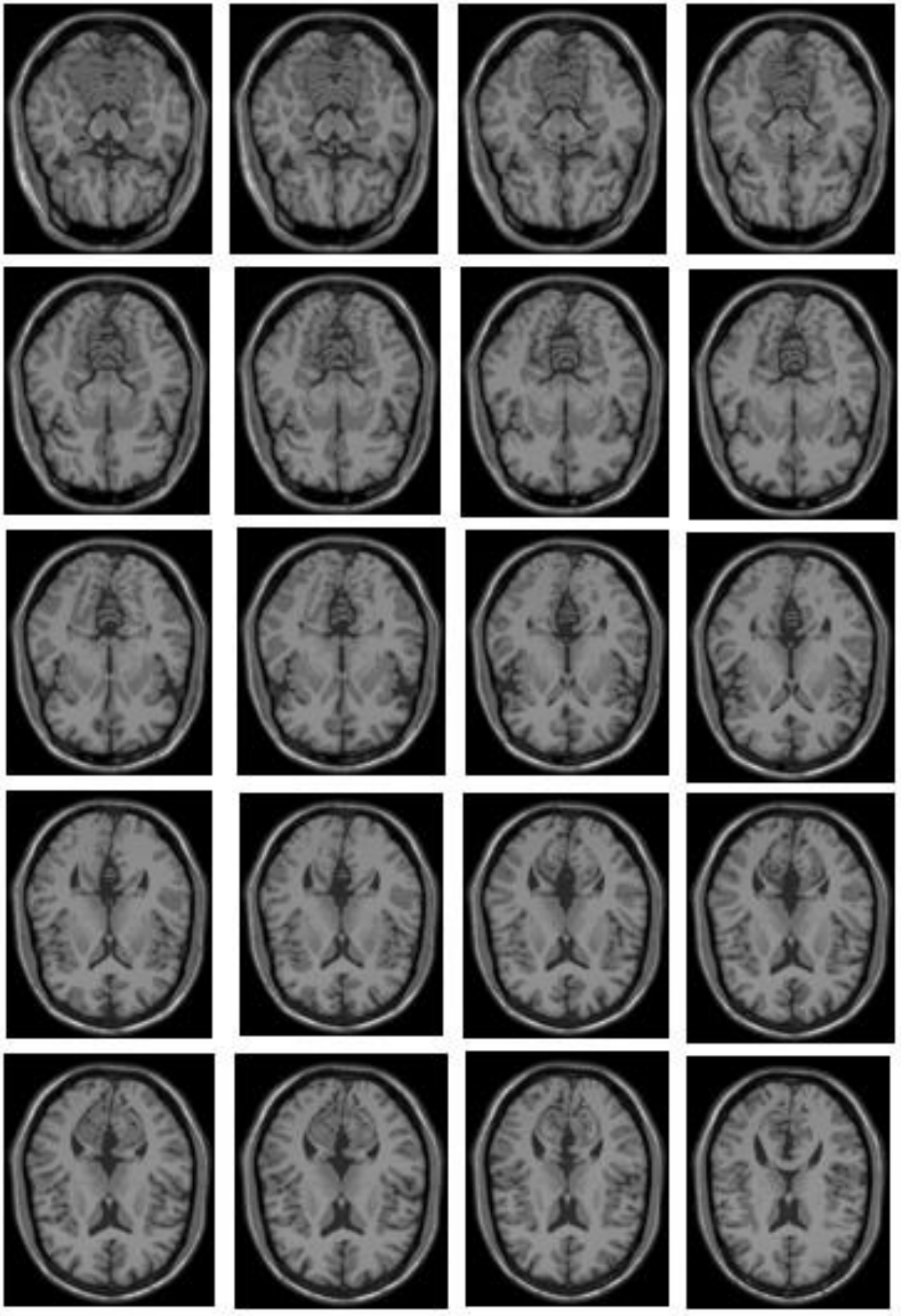


Figure 3.16 : Les images médicales décryptées.

4. Critères d'évaluation

Un bon système de cryptage doit être protégé contre toutes les attaques possibles de n'importe quelle sorte, il y a donc un ensemble de mesures qui servent cet objectif. Nous allons présenter les plus important comme : l'espace de clés, l'histogramme, l'entropie, la corrélation entre lespixels adjacents et la sensitivité de la clé.

4.1. Espace de clés

Un bon algorithme de chiffrement doit être sensible aux clés de chiffrement et l'espace clé doit être suffisamment grand pour rendre les attaques de force brute impossibles, et la taille de la clé peut être plus longue que la taille de l'image. Dans la technique de cryptage proposée, l'espace de clés est le nombre total de clés différentes utilisées dans la procédure composée de six parties (λ, Y, μ, X, K) et la clé de permutation (24 caractères), sans contraintes, la taille de l'espace clé est : $(2^{64})^5 \times (2^8)^{24} = 2^{512}$.

4.2. L'histogramme

Trois images de tests ont été utilisées dans l'analyse : Camera Man, Peppers et Baboon. Les tracés des histogrammes des images et les images chiffrées sont montrés dans les figures ci-dessous**.**

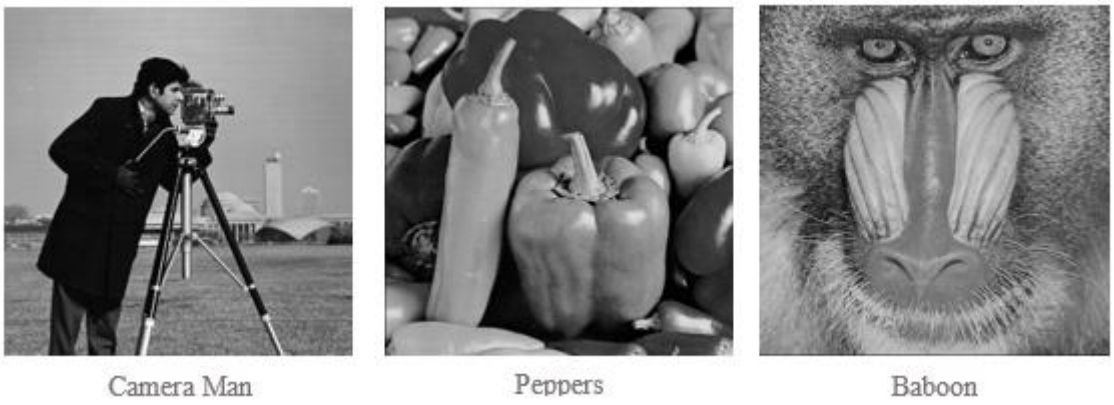


Figure 3.17 : Les trois images originales**.**

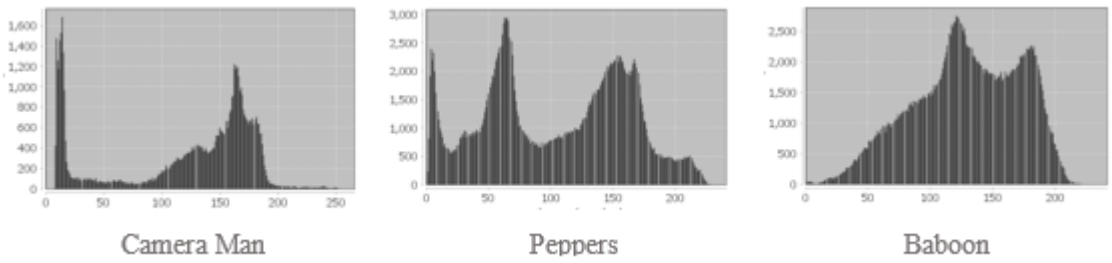


Figure 3.18 : Histogrammes sur les trois images originales.

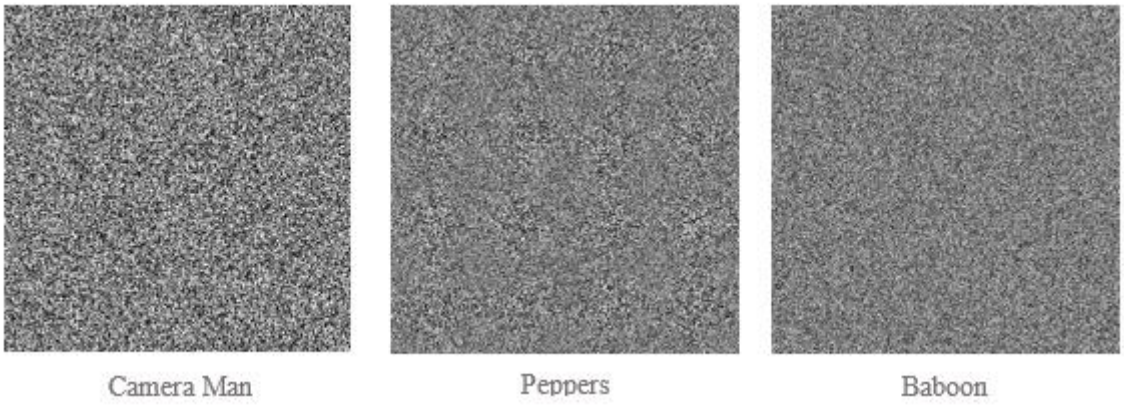


Figure 3.19 : Les trois images cryptées**.**

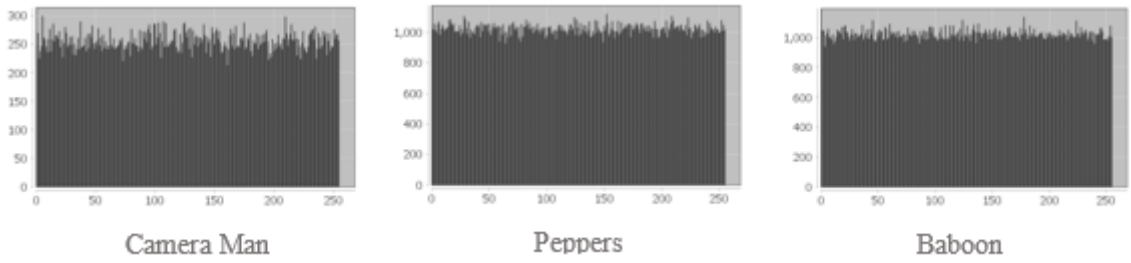


Figure 3.20 : Histogrammes sur les trois images cryptées.

Les résultats montrent que les histogrammes des images chiffrées sont uniformes après le cryptage. Par conséquent l'attaquant ne peut pas extraire information à partir de l'histogramme de l'image cryptée

4.3. L'entropie

Le tableau 3.1 montrant les valeurs de l'entropie des images claires et leurs chiffrées en utilisant le schéma proposé.

La valeur de l’entropie doit être très proche de 8, Parce que si l'entropie est inférieure à 8, il existe des degrés de prévisibilité, donc on ne peut pas assurer la sécurité contre l'analyse statistique**.**

Nom de l'image	Description de l'image	Taille	Type	L'entropie d'image	
				En claire	chiffrée
5.1.09.tiff	Moon surface	256×256	Niveau de gris	6.7093	7.9968
5.1.10.tiff	Aerial	256×256	Niveau de gris	7.3118	7.9969
5.1.11.tiff	Airplane	256×256	Niveau de gris	6.4522	7.9974
5.1.12.tiff	Clock	256×256	Niveau de gris	6.7056	7.9977
5.1.13.tiff	Resolution chart	256×256	Niveau de gris	1.5483	7.9971
5.1.14.tiff	Chemical plant	256×256	Niveau de gris	7.3424	7.9972
5.2.08.tiff	Couple	512×512	Niveau de gris	7.2010	7.9992
5.2.09.tiff	Aerial	512×512	Niveau de gris	6.9939	7.9992
5.2.10.tiff	Stream and bridge	512×512	Niveau de gris	5.7055	7.9992
5.3.01.tiff	Man	1024×1024	Niveau de gris	7.5237	7.9998
5.3.02.tiff	Airport	1024×1024	Niveau de gris	6.8303	7.9998

7.1.01.tiff	Truck	256×256	Niveau de gris	6.0274	7.9993
7.1.02.tiff	Airplane	1024×1024	Niveau de gris	4.0044	7.9993
7.1.03.tiff	Tank	512×512	Niveau de gris	5.4957	7.9993
7.1.04.tiff	Car and APCs	512×512	Niveau de gris	6.1074	7.9991
7.1.05.tiff	Truck and APCs	512×512	Niveau de gris	6.5631	7.9993
7.1.06.tiff	Truck and APCs	512×512	Niveau de gris	6.6952	7.9993
7.1.07.tiff	Tank	512×512	Niveau de gris	5.9915	7.9992
7.1.08.tiff	APC	512×512	Niveau de gris	5.0534	7.9992
7.1.09.tiff	Tank	512×512	Niveau de gris	6.1898	7.9993
7.1.10.tiff	Car and APCs	512×512	Niveau de gris	5.9087	7.9992
7.2.01.tiff	Airplane (U-2)	1024×1024	Niveau de gris	5.6414	7.9997
boat.512.tiff	Fishing Boat	512×512	Niveau de gris	7.1913	7.9993
elaine.512.tiff	Girl (Elaine)	512×512	Niveau de gris	7.5059	7.9993
gray21.512.tiff	21 level step wedge	512×512	Niveau de gris	4.3922	7.9993
numbers.512.tiff	256 level test pattern	512×512	Niveau de gris	7.7292	7.9993
ruler.512.tiff	Pixel ruler	512×512	Niveau de gris	0.5000	7.9990
testpat.1k.tiff	General test pattern	1024×1024	Niveau de gris	4.4077	7.9997
camera_man.tiff	Camera man	256×256	Niveau de gris	7.0097	7.9974
lena.tiff	Girl (lena)	512×512	Niveau de gris	7.4455	7.9992
peppers.tiff	Peppers	512×512	Niveau de gris	7.5714	7.9992
Baboon.tiff	Baboon	512×512	Niveau de gris	7.3579	7.9992
Barbara.tiff	Girl (Barbara)	512×512	Niveau de gris	7.4664	7.9993
Moyenne				**6.1387**	**7.9988**

Table 3.1 : Comparaison des entropies entre les images en claire et chiffrées.

Le résultat montre qu'après simuler de 33 images, la valeur moyenne de l'entropie des images chiffrées est **7,9988**, c'est-à-dire il est plus proche à la valeur 8. Cela montre qu'il est difficile d'avoir la prévisibilité.

4.4. La corrélation entre les pixels adjacents

Le tableau 3.1 montrant les corrélations des images claires et leurs chiffrées en utilisant le schéma proposé.

Si la valeur de corrélation proche 1, cela signifie que l'image-claire et de image-chiffrée sont très dépendantes. Et aussi si la valeur de corrélation proche $^{+}0$, cela signifie que l'Image-Chiffrée et l'Image-clair ne sont pas corrélés. Ainsi, plus faible est la valeur de corrélation, la qualité de cryptage est meilleure.

Nom de l'image	**Description de l'image**	**Taille**	**Type**	**La corrélation avec l'image**	
				En claire	**chiffrée**
5.1.09.tiff	Moon surface	256×256	Niveau de gris	1.0000	0.0020
5.1.10.tiff	Aerial	256×256	Niveau de gris	1.0000	-0.0006

5.1.11.tiff	Airplane	256×256	Niveau de gris	1.0000	0.0009
5.1.12.tiff	Clock	256×256	Niveau de gris	0.9999	0.0005
5.1.13.tiff	Resolution chart	256×256	Niveau de gris	1.0000	-0.0047
5.1.14.tiff	Chemical plant	256×256	Niveau de gris	1.0000	-0.0006
5.2.08.tiff	Couple	512×512	Niveau de gris	0.9999	0.0004
5.2.09.tiff	Aerial	512×512	Niveau de gris	1.0000	0.0003
5.2.10.tiff	Stream and bridge	512×512	Niveau de gris	1.0000	-0.0011
5.3.01.tiff	Man	1024×1024	Niveau de gris	1.0000	-0.0006
5.3.02.tiff	Airport	1024×1024	Niveau de gris	1.0000	0.0008
7.1.01.tiff	Truck	256×256	Niveau de gris	1.0000	-0.00002
7.1.02.tiff	Airplane	1024×1024	Niveau de gris	1.0000	0.00006
7.1.03.tiff	Tank	512×512	Niveau de gris	1.0000	0.0032
7.1.04.tiff	Car and APCs	512×512	Niveau de gris	1.0000	-0.0001
7.1.05.tiff	Truck and APCs	512×512	Niveau de gris	1.0000	0.0003
7.1.06.tiff	Truck and APCs	512×512	Niveau de gris	1.0000	0.0025
7.1.07.tiff	Tank	512×512	Niveau de gris	1.0000	-0.00003
7.1.08.tiff	APC	512×512	Niveau de gris	0.9999	0.0005
7.1.09.tiff	Tank	512×512	Niveau de gris	0.9999	-0.0031
7.1.10.tiff	Car and APCs	512×512	Niveau de gris	1.0000	0.0003
7.2.01.tiff	Airplane (U-2)	1024×1024	Niveau de gris	0.9999	0.0018
boat.512.tiff	Fishing Boat	512×512	Niveau de gris	1.0000	0.0003
elaine.512.tiff	Girl (Elaine)	512×512	Niveau de gris	0.9999	0.0006
gray21.512.tiff	21 level step wedge	512×512	Niveau de gris	1.0000	-0.0001
numbers.512.tiff	256 level test pattern	512×512	Niveau de gris	1.0000	-0.0017
ruler.512.tiff	Pixel ruler	512×512	Niveau de gris	1.0000	-0.0004
testpat.1k.tiff	General test pattern	1024×1024	Niveau de gris	1.0000	0.0003
camera_man.tiff	Camera man	256×256	Niveau de gris	1.0000	-0.0013
lena.tiff	Girl (lena)	512×512	Niveau de gris	0.9999	-0.0021
peppers.tiff	Peppers	512×512	Niveau de gris	1.0000	0.0016
Baboon.tiff	Baboon	512×512	Niveau de gris	0.9999	0.0008
Barbara.tiff	Girl (Barbara)	512×512	Niveau de gris	1.0000	0.0009
Moyenne				**0.9999**	**0.0010**

Table 3.2 : Comparaison des corrélations entre les images en claire et chiffrées**.**

Le résultat montre qu'après simuler de 33 images, la valeur moyenne des corrélations dès l'image chiffrée est **0,001**, c'est-à-dire il est plus proche à la valeur 0. Cela montre que les pixels adjacents après le cryptage n'ont pas de corrélation.

4.5. Sensitivité de la clé

Pour estimer la sensibilité de la clé secrète de l'algorithme proposé deux clés sont utilisées dans le test. La première clé est la clé d'origine tandis que nous avons fait une modification dans un seul paramètre de la clé et nous laissons les cinq autres paramètres sans modification. Puis, les deux clés légèrement différentes sont utilisées pour chiffrer l'image Peppers.

Les deux images chiffrées sont comparées. Les résultats sont récapitulés dans le tableau 3.3 montrant la sensibilité élevée de la clé du schéma proposé. En outre, si une petite modification est effectuée sur la clé, puis la clé modifiée est utilisée pour décrypter l'image chiffrée, le décryptage échoue totalement. La figure 3.21 illustre le résultat de ce test.

Les paramètres initiaux sont : ($X_{0=}$ 0.9, $Y_0 = 0.9, \mu = 3.9,\ \lambda = 3.89, K = 18.9$) et la clé de permutation = "qucnslmhaahjhrgoahagoHHn".

Les changements sont comme suit :

(a) : ($X_{0=}$ 0.9, $Y_0 = 0.9, \mu = 3.9,\ \lambda = 3.89, K = 18.9$) et clé de permutation = "qucnslmhaahjhrgoahagoHHm".

(b) : ($X_{0=}$ 0.9, $Y_0 = 0.9, \mu = 3.90001,\ \lambda = 3.89, K = 18.9$) et clé de permutation = "qucnslmhaahjhrgoahagoHHn".

(c) : ($X_{0=}$ 0.90001, $Y_0 = 0.9, \mu = 3.9,\ \lambda = 3.89, K = 18.9$) et clé de permutation = "qucnslmhaahjhrgoahagoHHn".

(d) : ($X_{0=}$ 0.9, $Y_0 = 0.9, \mu = 3.9,\ \lambda = 3.89, K = 18.91$) et clé de permutation = "qucnslmhaahjhrgoahagoHHn".

(e) : ($X_{0=}$ 0.9, $Y_0 = 0.9, \mu = 3.9,\ \lambda = 3.89001, K = 18.9$) et clé de permutation = "qucnslmhaahjhrgoahagoHHn".

(f) : ($X_{0=}$ 0.9, $Y_0 = 0.90001, \mu = 3.9,\ \lambda = 3.89, K = 18.9$) et clé de permutation = "qucnslmhaahjhrgoahagoHHn".

Paramètre Modifiée	NPCR	UACI
(a)	99.6101	33.4137
(b)	1.7723	0.5770
(c)	98.6175	33.1501
(d)	99.5346	33.4437
(e)	99.6280	33.4369
(f)	99.6391	33.4827

Table 3.3 : Sensibilité de la clé en utilisant les différents paramètres.

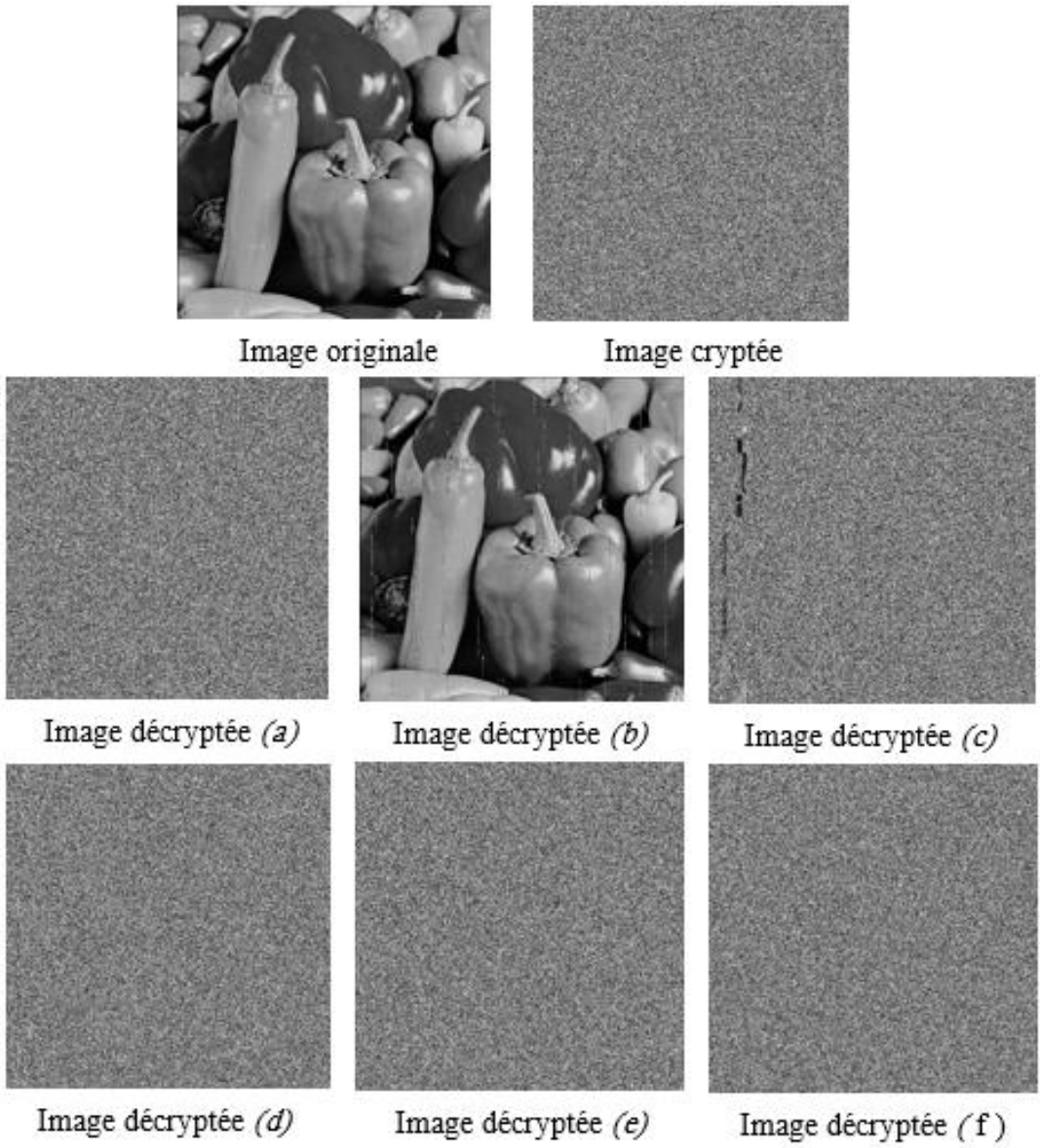


Figure 3.21 : Sensibilité de la clé en utilisant les différents paramètres en décryptage.

Nous avons trouvés que tous les clés ont une grande sensibilité au changement sauf pour la clé μ qu'a une petite sensibilité par rapport les autres.

5. Étude comparative

Dans cette étude, Nous avons comparé notre algorithme proposé en deux parties :

La première partie est une comparaison interne avec les trois étapes de l'algorithme proposé et la deuxième partie est une comparaison externe avec les autres techniques de cryptage d'image, qui été proposé par les chercheurs de cryptographie.

Les valeurs initiales et les paramètres utilisés dans notre algorithme proposé pour cette comparaison sont : ($X_{0=}$ $0.9, Y_0 = 0.9, \mu = 3.9,\ \lambda\ = 3.89, K = 18.9$) et la clé de permutation = "qucnslmhaahjhrgoahagoHHn".

5.1. Comparaison interne

Le tableau 3.4 montrant la comparaison entre l'algorithme proposé et les trois étapes et le critère utilisée dans cette comparaison c'est l'entropie, La figure 3.22 illustre le résultat de cette comparaison.

Nom de l'image	Description de l'image	Algorithme proposé	Étape 01	Étape 02	Étape 03
5.1.09.tiff	Moon surface	7.9968	7.9938	6.7093	7.9948
5.1.10.tiff	Aerial	7.9969	7.9966	7.3118	7.9966
5.1.11.tiff	Airplane	7.9974	7.9949	6.4522	7.9953
5.1.12.tiff	Clock	7.9977	7.9939	6.7056	7.9931
5.1.13.tiff	Resolution chart	7.9971	7.9417	1.5483	7.9485
5.1.14.tiff	Chemical plant	7.9972	7.9968	7.3424	7.9968
5.2.08.tiff	Couple	7.9992	7.9972	7.2010	7.9975
5.2.09.tiff	Aerial	7.9992	7.9974	6.9939	7.9974
5.2.10.tiff	Stream and bridge	7.9992	7.9982	5.7055	7.9983
5.3.01.tiff	Man	7.9998	7.9992	7.5237	7.9991
5.3.02.tiff	Airport	7.9998	7.9978	6.8303	7.9978
7.1.01.tiff	Truck	7.9993	7.9953	6.0274	7.9954
7.1.02.tiff	Airplane	7.9993	7.9884	4.0044	7.9888
7.1.03.tiff	Tank	7.9993	7.9942	5.4957	7.9945
7.1.04.tiff	Car and APCs	7.9991	7.9953	6.1074	7.9958
7.1.05.tiff	Truck and APCs	7.9993	7.9978	6.5631	7.9979
7.1.06.tiff	Truck and APCs	7.9993	7.9986	6.6952	7.9986
7.1.07.tiff	Tank	7.9992	7.9960	5.9915	7.9962
7.1.08.tiff	APC	7.9992	7.9894	5.0534	7.9896
7.1.09.tiff	Tank	7.9993	7.9969	6.1898	7.9970
7.1.10.tiff	Car and APCs	7.9992	7.9946	5.9087	7.9948
7.2.01.tiff	Airplane (U-2)	7.9997	7.9919	5.6414	7.9918
boat.512.tiff	Fishing Boat	7.9993	7.9979	7.1913	7.9981
elaine.512.tiff	Girl (Elaine)	7.9993	7.9986	7.5059	7.9987
gray21.512.tiff	21 level step wedge	7.9993	7.9942	4.3922	7.9940
numbers.512.tiff	256 level test pattern	7.9993	7.9991	7.7292	7.9991
ruler.512.tiff	Pixel ruler	7.9990	7.9343	0.5000	7.9348
testpat.1k.tiff	General test pattern	7.9997	7.9908	4.4077	7.9909
camera_man.tiff	Camera man	7.9974	7.9961	7.0097	7.9960
lena.tiff	Girl (lena)	7.9992	7.9985	7.4455	7.9986
peppers.tiff	Peppers	7.9992	7.9989	7.5714	7.9990
Baboon.tiff	Baboon	7.9992	7.9983	7.3579	7.9984
Barbara.tiff	Girl	7.9993	7.9986	7.4664	7.9986
Moyenne		**7.9988**	**7.9924**	**6.1387**	**7.9927**
Pourcentage		**100%**	**100%**	**100%**	**100%**

Table 3.4 : La comparaison interne : l'entropie**.**

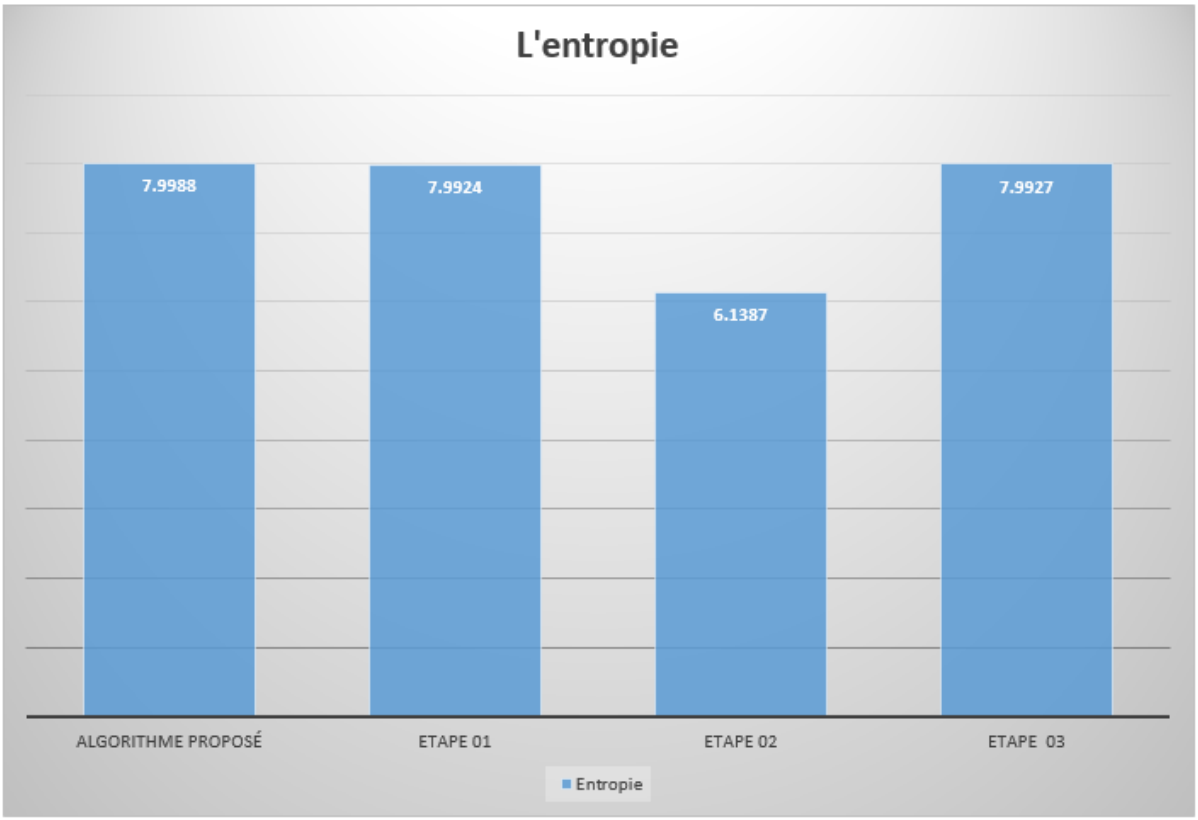


Figure 3.22 : Résultat de comparaison interne : entropie.

Le tableau 3.5 montrant la comparaison entre l'algorithme proposé et les trois étapes et le critère utilisée dans cette comparaison c'est la corrélation, La figure 3.23 illustre le résultat de cette comparaison.

Nom de l'image	Description de l'image	Algorithme proposé	Étape 01	Étape 02	Étape 03
5.1.09.tiff	Moon surface	0.0020	-0.0019	0.0078	-0.0002
5.1.10.tiff	Aerial	-0.0006	0.0014	0.0037	-0.0059
5.1.11.tiff	Airplane	0.0009	0.0033	-0.0098	-0.0049
5.1.12.tiff	Clock	0.0005	-0.0009	0.0945	-0.0072
5.1.13.tiff	Resolution chart	-0.0047	-0.0029	-0.0023	-0.0132
5.1.14.tiff	Chemical plant	-0.0006	-0.0009	-0.0126	-0.0082
5.2.08.tiff	Couple	0.0004	-0.0030	0.0191	-0.0036
5.2.09.tiff	Aerial	0.0003	-0.0008	-0.0015	-0.0019
5.2.10.tiff	Stream and bridge	-0.0011	-0.0029	-0.0332	-0.0042
5.3.01.tiff	Man	-0.0006	-0.0034	0.0081	-0.0041
5.3.02.tiff	Airport	0.0008	-0.0030	0.00003	-0.0036
7.1.01.tiff	Truck	-0.00002	-0.0014	0.0046	-0.0017
7.1.02.tiff	Airplane	0.00006	-0.0031	0.0114	-0.0035
7.1.03.tiff	Tank	0.0032	-0.0001	0.0323	0.0003
7.1.04.tiff	Car and APCs	-0.0001	-0.0031	-0.0046	-0.0024
7.1.05.tiff	Truck and APCs	0.0003	-0.0034	-0.0061	-0.0041
7.1.06.tiff	Truck and APCs	0.0025	-0.0047	0.0185	-0.0075

7.1.07.tiff	Tank	-0.00003	-0.0031	-0.0162	-0.0039
7.1.08.tiff	APC	0.0005	-0.0039	-0.0211	-0.0034
7.1.09.tiff	Tank	-0.0031	-0.0051	0.0226	-0.0053
7.1.10.tiff	Car and APCs	0.0003	-0.0024	-0.00004	-0.0026
7.2.01.tiff	Airplane (U-2)	0.0018	-0.0043	0.0176	-0.0048
boat.512.tiff	Fishing Boat	0.0003	-0.0024	0.0015	-0.0023
elaine.512.tiff	Girl (Elaine)	0.0006	-0.0030	0.0116	-0.0033
gray21.512.tiff	21 level step wedge	-0.0001	-0.0034	-0.0662	-0.0086
numbers.512.tiff	256 level test pattern	-0.0017	-0.0048	-0.0103	-0.0077
ruler.512.tiff	Pixel ruler	-0.0004	-0.0015	0.0145	-0.0078
testpat.1k.tiff	General test pattern	0.0003	-0.0011	0.0069	-0.0011
camera_man.tiff	Camera man	-0.0013	-0.0036	-0.0109	-0.0069
lena.tiff	Girl (lena)	-0.0021	-0.0038	-0.0164	-0.0047
peppers.tiff	Peppers	0.0016	-0.0040	0.0010	-0.0069
Baboon.tiff	Baboon	0.0008	-0.0048	-0.0106	-0.0059
Barbara.tiff	Girl	0.0009	-0.0027	-0.0373	-0.0053
Moyenne		**0.0010**	**0.0028**	**0.0162**	**0.0047**
Pourcentage			**90.90%**	**87.88%**	**93.93%**

Table 3.5 : La comparaison interne : la corrélation.

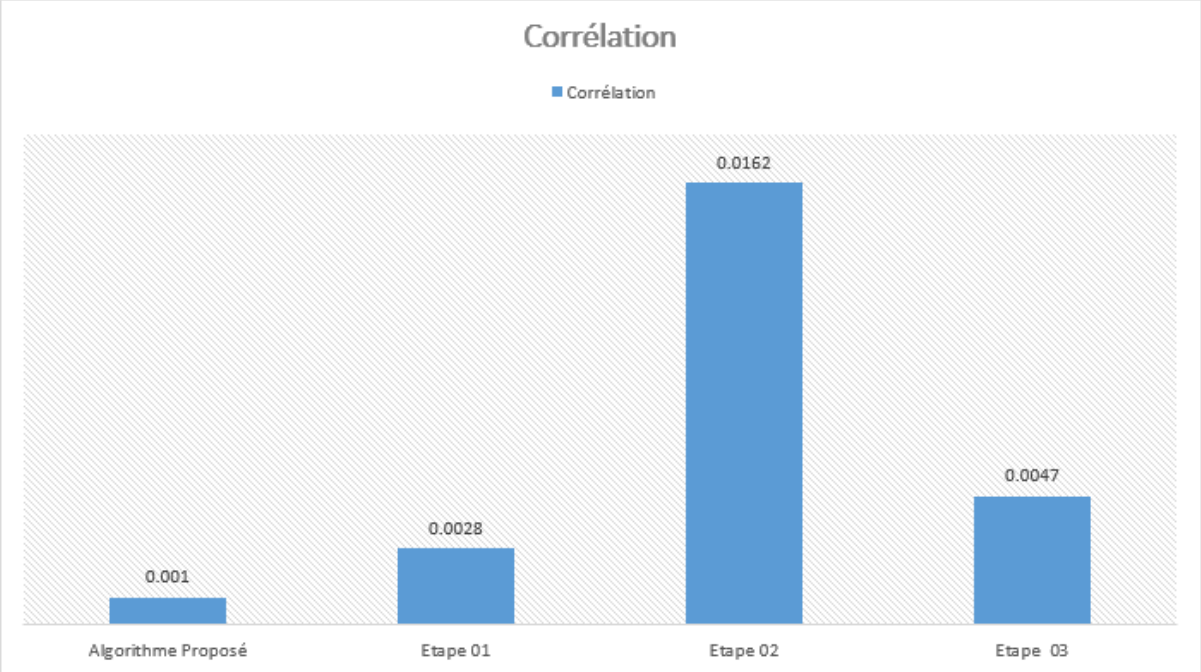


Figure 3.23 : Résultat de comparaison interne : corrélation.

A partir des résultats obtenus, on peut remarquer que la valeur moyenne de l'entropie de notre algorithme est supérieure que les algorithmes internes. De plus, cette valeur est plus proche à la valeur 8 par rapport ce qui indique son efficacité. La valeur moyenne de corrélation de notre algorithme proposé est inférieure que les algorithmes internes. Aussi, elle est plus proche à la

valeur 0 ce qui confirme la qualité de la sécurité des images utilisées.

Alors, Nous pouvons dire que notre hybridation est réussie.

5.2. Comparaison externe

On commence par la première comparaison externe, c'est l'algorithme proposé avec d'autres trois algorithmes de cryptage d'image. Le tableau 3.6 montrant : la comparaison entre l'algorithme proposé et les cinq autres algorithmes de cryptage, Et aussi la corrélation a été utilisée pour cette comparaison, La figure 3.24 illustre le résultat de cette comparaison.

Nom de l'image	Description de l'image	Algorithme proposé	Algo-1[44]	Algo-2[44]	Algo-3[44]
5.1.09.tiff	Moon surface	0.0020	0.0054	0.0043	0.0866
5.1.10.tiff	Aerial	-0.0006	0.0017	0.0057	0.1304
5.1.11.tiff	Airplane	0.0009	0.0035	0.0049	0.0965
5.1.12.tiff	Clock	0.0005	0.0037	0.0053	0.1552
5.1.13.tiff	Resolution chart	-0.0047	-0.0043	-0.0061	0.1738
5.1.14.tiff	Chemical plant	-0.0006	-0.0022	-0.0022	0.1215
5.2.08.tiff	Couple	0.0004	0.0008	0.0028	0.1166
5.2.09.tiff	Aerial	0.0003	-0.0052	0.0015	0.1086
5.2.10.tiff	Stream and bridge	-0.0011	0.0000	0.0009	0.1510
5.3.01.tiff	Man	-0.0006	0.0003	0.0055	0.1555
5.3.02.tiff	Airport	0.0008	0.0009	0.0003	0.0986
7.1.01.tiff	Truck	-0.00002	-0.0019	-0.0022	0.0776
7.1.02.tiff	Airplane	0.00006	-0.0019	0.0014	0.0602
7.1.03.tiff	Tank	0.0032	-0.0000	0.0003	0.0781
7.1.04.tiff	Car and APCs	-0.0001	-0.0007	0.0006	0.0998
7.1.05.tiff	Truck and APCs	0.0003	0.0007	-0.0013	0.0995
7.1.06.tiff	Truck and APCs	0.0025	-0.0042	-0.0065	0.0923
7.1.07.tiff	Tank	-0.00003	-0.0001	-0.0016	0.0730
7.1.08.tiff	APC	0.0005	0.0032	0.0017	0.0757
7.1.09.tiff	Tank	-0.0031	-0.0010	0.0012	0.1050
7.1.10.tiff	Car and APCs	0.0003	-0.0003	-0.0008	0.0784
7.2.01.tiff	Airplane (U-2)	0.0018	0.0001	-0.0094	0.0643
boat.512.tiff	Fishing Boat	0.0003	0.0009	-0.025	0.1314
elaine.512.tiff	Girl (Elaine)	0.0006	-0.0027	0.0006	0.1304
gray21.512.tiff	21 level step wedge	-0.0001	-0.0026	0.0030	0.2009
numbers.512.tiff	256 level test pattern	-0.0017	-0.0005	-0.0026	0.1689
ruler.512.tiff	Pixel ruler	-0.0004	0.0030	0.0006	0.1836
testpat.1k.tiff	General test pattern	0.0003	0.0005	0.0009	0.1952
Moyenne		**0.0009**	**0.0018**	**0.0035**	**0.1181**
Pourcentage			**75%**	**85.71%**	**100%**

Table 3.6 : Résultats de première comparaison externe**.**

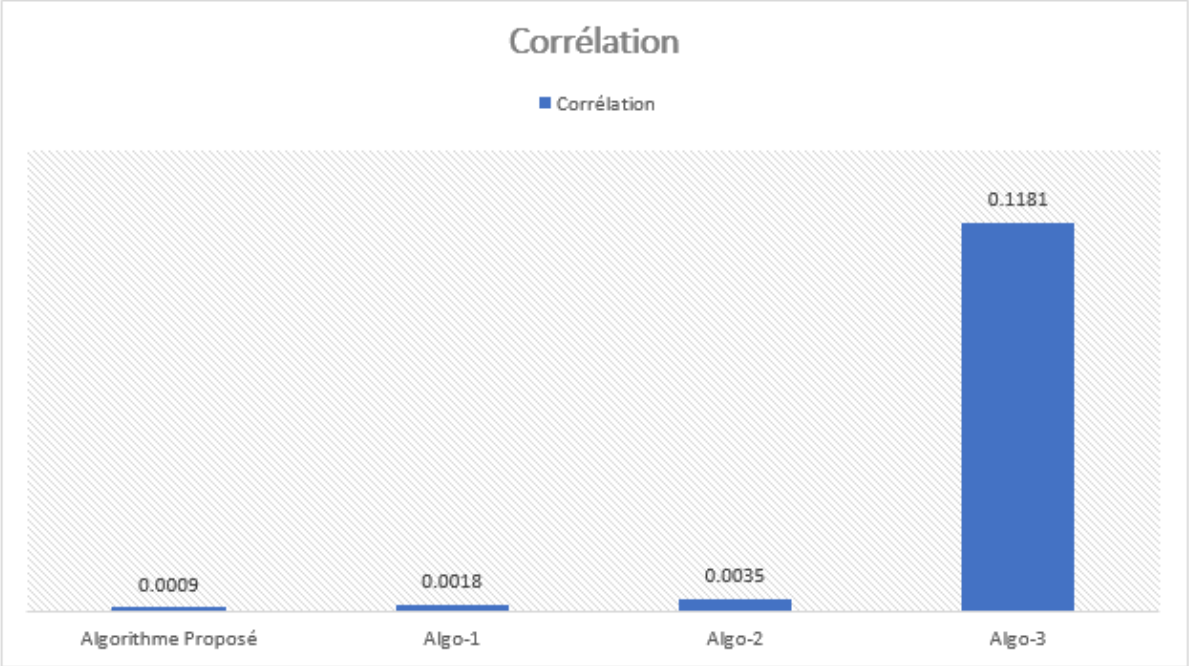


Figure 3.24 : Résultat de première comparaison externe.

Le résultat après le calcul de la valeur moyenne de corrélation de chaque algorithme, montre d'une part que la valeur moyenne obtenue par notre algorithme proposé est inférieure aux celles obtenues par les différents trois algorithmes cités. D'autres parts, cette valeur est plus proche à la valeur 0 ce qui signifié que les données sont sécurisées efficacement par notre système proposé.

Une deuxième comparaison a été établie avec un autre travail proposé dans [45]. Les résultats obtenus en utilisant l'entropie et la corrélation comme critères d'évaluation sont présentés dans le tableau 3.7 ci-dessous :

Nom de l'image	**Description de l'image**	**Algorithme proposé**		**Algo[45]**	
		Entropie	**Corrélation**	**Entropie**	**Corrélation**
lena.tiff	Girl (lena)	7.9992	-0.0021	7.9993	0.0013
peppers.tiff	Peppers	7.9992	0.0016	7.9992	0.0049
camera_man.tiff	Camera man	7.9974	-0.0013	7.9965	-0.0021
Baboon.tiff	Baboon	7.9992	0.0008	7.9993	-0.0045
elaine.512.tiff	Girl (Elaine)	7.9993	0.0006	7.9992	-0.0030
Moyenne		**7.9988**	**0.0012**	**7.9987**	**0.0031**
Pourcentage				**60%**	**80%**

Table 3.7 : Résultats de deuxième comparaison externe**.**

A partir du tableau 3.7, on peut remarquer que la valeur moyenne de l'entropie de notre

algorithme est supérieure de celle proposé [45]. De plus, cette valeur est plus proche à la valeur 8 par rapport ce qui indique son efficacité. La valeur moyenne de corrélation de notre algorithme proposé est inférieure de l'algorithme présenté dans [45]. Aussi, elle est plus proche à la valeur 0 ce qui confirme la qualité de la sécurité des images utilisées.

Pour mieux évaluer notre méthode de cryptage, une troisième comparaison a été établie avec des travaux récents. Les résultats obtenus en utilisant l'entropie critère d'évaluation sont présentés dans le tableau 3.8 ci-dessous, La figure 3.25 illustre le résultat de cette comparaison.

Nom de l'image	**Description de l'image**	**Algorithme proposé**	**Algo-1[46]**	**Algo-2[47]**
5.1.09.tiff	Moon surface	7.9968	7.9975	7.9987
5.1.11.tiff	Airplane	7.9974	7.9972	7.9974
5.1.12.tiff	Clock	7.9977	7.9975	7.9992
5.1.14.tiff	Chemical plant	7.9972	7.9971	7.9964
5.2.08.tiff	Couple	7.9992	7.9993	7.9976
5.3.01.tiff	Man	7.9998	7.9998	7.9975
5.3.02.tiff	Airport	7.9998	7.9998	7.9960
7.1.07.tiff	Tank	7.9992	7.9994	7.9969
boat.512.tiff	Fishing Boat	7.9993	7.9993	7.9980
elaine.512.tiff	Girl (Elaine)	7.9993	7.9992	7.9985
lena.tiff	Girl (lena)	7.9992	7.9991	7.9963
peppers.tiff	Peppers	7.9992	7.9992	7.9985
barbara.tiff	Girl (Barbara)	7.9993	7.9993	7.9978
Cameraman.tiff	Camera man	7.9974	7.9972	7.9985
Testpat.1k.tiff	General test pattern	7.9997	7.9998	7.9986
Moyenne		**7.9987**	**7.9987**	**7.9977**
Pourcentage			**73.33%**	**80%**

Table 3.8 : Résultats de troisième comparaison externe.

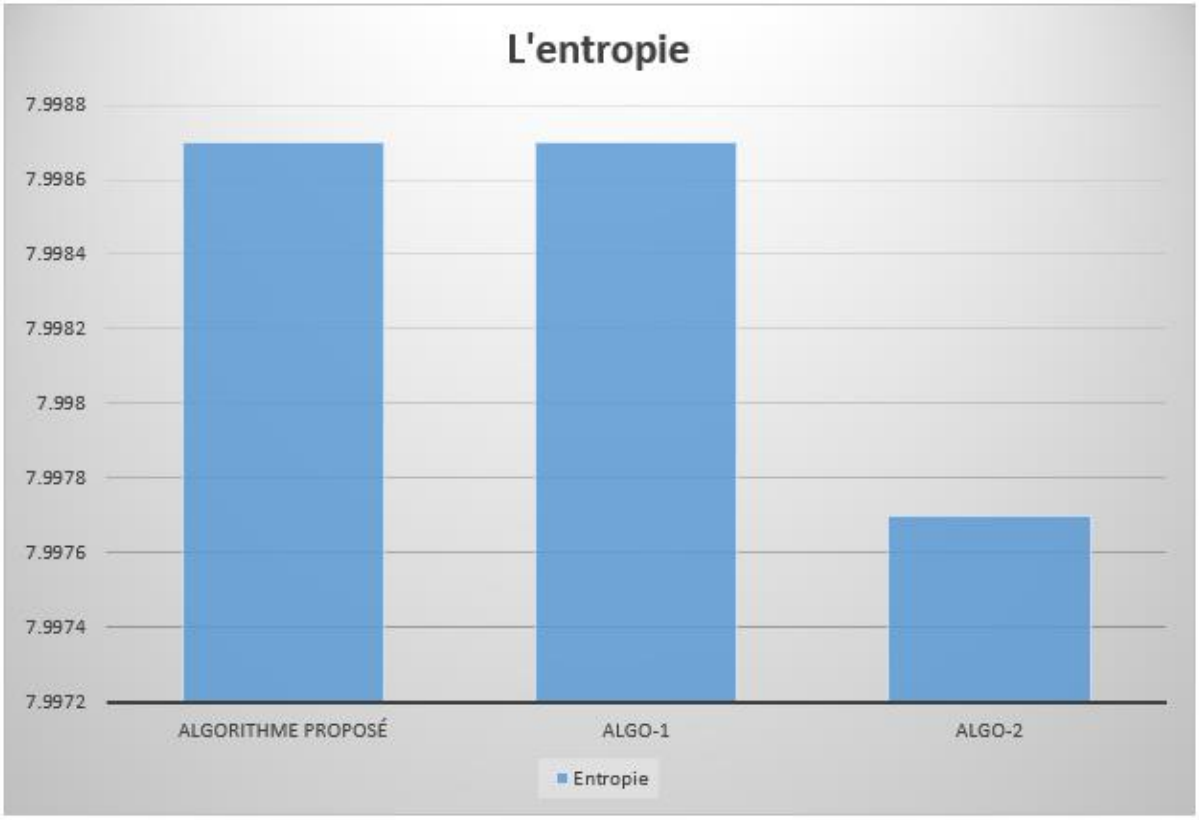


Figure 3.25 : Résultat de troisième comparaison externe.

A partir du tableau 3.8, on peut remarquer que la valeur moyenne de l'entropie de notre algorithme a des bons résultats contre les travaux les plus récents.

6. Conclusion

Dans ce chapitre, nous avons proposé un schéma de chiffrement d'image qui basé sur l'hybridation entre plusieurs techniques de cryptage comme les carte chaotiques sine, logistique et standard, la suite de Fibonacci Modifiée et les techniques de permutation. Les résultats expérimentaux ont montré que le système de cryptage d'image proposé possède un grand espace de clés et une sécurité de haut niveau. Ainsi l'analyse prouve la sécurité, l'efficacité et la sensibilité. De plus, les comparaisons avec les schémas de chiffrement d'image existants qui ont été réalisées, montrent que l'algorithme proposé offre des performances très favorables.

CONCLUSION GÉNÉRALE

Pour le moment, les images médicales sont de plus en plus utilisées, entraînant une transmission accrue de ce type de données entre les réseaux en général et l'Internet en particulier et la confidentialité de ce type de données est devenue indispensable.

Au cours de ce mémoire, nous avons proposé un schéma de chiffrement d'image basé sur l'hybridation entre plusieurs techniques de cryptage comme les carte chaotiques sine, logistique et standard, la suite de Fibonacci Modifiée et les techniques de permutation. Le but principal de ce chiffrement est la combinant les propriétés et les avantages entre eux pour obtenir un système de cryptage pour les images médicales de format DICOM.

Les résultats expérimentaux montrent clairement, que l'algorithme proposé dispose un niveau élevé de confusion. Et ainsi l'espace clé est suffisamment grand, ce qui rend une attaque force brute infaisable. Par conséquent l'histogramme d'image chiffrée est très uniforme après le cryptage, voire, l'attaquant il ne peut pas extraire l'information à partir de l'histogramme de l'image cryptée. Également l'algorithme proposé a été atteintes beaucoup amélioré sur l'entropie et la corrélation entre les pixels adjacents. De ce fait l'algorithme proposé montre l'efficacité et la sécurité de notre système proposé.

Enfin, les comparaisons avec les schémas de chiffrement d'image existants qui ont été réalisées et avec les algorithmes de base de notre méthode montrent que l'algorithme proposé offre des performances très favorables. Comme perspective à ce travail, nous allons améliorer notre approche sur tous les formats des images médicales en général et les images couleur entre eux en particulier.

BIBLIOGRAPHIE

[1] L. Grazide, L'image électronique, http://auch2.free.fr/Documents/Informatique/Image_electronique.pdf, consulté le 18-04-2018.

[2] Rafael C Gonzalez and Richard E Woods. Digital image processing 3rd edition, Pearson Prentice Hall, Upper Saddle River, 2007.

[3] Numeriksciences, http://numeriksciences.fr, consulté le 18-04-2018.

[4] GREYC IMAGE, Qu'est-ce qu'une image numérique ?, ENSICAEN & Université de Caen & CNRS, https://clouard.users.greyc.fr/fetedelascience/documents/image.pdf , consulté le 18-04-2018.

[5] R. Isdant, Traitement numérique de l'image, 2009, http://raphael.isdant.free.fr/traitement_numerique/2-traitement_numerique_de_l%27image.pdf, consulté le 18-04-2018.

[6] Léon Robichaud, L'image numérique Pixels et couleurs, support de cours, Département d'histoire, Université de Sherbrooke.

[7] Les formats d'images numériques, Serge WACKER – C2I niveau 1, http://serge.wacker.free.fr/technoprimaire/c2i/revisions/formats_image.pdf , consulté le 19-04-2018.

[8] W. Puech Archivage d'images médicales LIRMM, CNRS/ University of Montpellier, France.

[9] LESCOP Yves [V1.6], La sécurité informatique, Post BTS R2i, 2002, http://ylescop.free.fr/mrim/cours/securite.pdf , consulté le 18-04-2018.

[10] Mme L. SAOUDI, initiation à la cryptographie, support de cours du module Sécurité informatique, Département d'informatique, université de Msila, Année 2015/2016.

[11] R. Dumont, Cryptographie et Sécurité informatique, Notes de cours provisoires, Université de Liège, 2009 – 2010.

[12] Wikipédia, https://fr.wikipedia.org/wiki/Confusion_et_diffusion, consulté le 25-04-2018.

[13] Penta Security, https://www.pentasecurity.com/product/encryption/#tab-id-2, consulté le 25-04-2018.

[14] Wikipédia, https://fr.wikipedia.org/wiki/Chiffrement_par_transposition, consulté le 25-04-2018.

[15] principe de base de la cryptographie http://dspace.univ-tlemcen.dz/bitstream/112/1046/8/chapitre2.pdf.

[16] A. Beloucif, Contribution à l'étude des mécanismes cryptographiques, thése En vue de l'obtention du diplôme de Doctorat en Informatique, Université de Batna2, 2016.

[17] Alfred J Menezes, Paul C Van Oorschot, and Scott A Vanstone, Handbook of applied cryptography, CRC press, 1996.

[18] A. Walker, E. Wolfart, R. Fisher, S. Perkins, Image processing learning resources explore with java, http://homepages.inf.ed.ac.uk/rbf/HIPR2/hipr_top, Consulté le 25-04-2018.

[19] Hvinagre, http://hvinagre.doomby.com/pages/articles/lecture-de-l-histogramme.html, consulté le 25-04-2018.

[20] Open Class rooms, https://openclassrooms.com/courses/introduction-a-la-vision-par-ordinateur/etirement-et-egalisation-d-histogramme, consulté le 25-04-2018.

[21] A. Aimeur, Conception et implémentation d'un système hybride pour la sécurité de données : application aux images numériques, Mémoire présenté pour l'obtention Du diplôme de Master Académique, Université de Msila, 2017.

[22] Claude E Shannon, A mathematical theory of communication, ACM SIGMOBILE Mobile Computing and Communications Review, 5(1) :3–55, 2001.

[23] Wikipédia, https://fr.wikipedia.org/wiki/Entropie_de_Shannon, consulté le 25-04-2018.

[24] Wikipédia, https://fr.wikipedia.org/wiki/Leonardo_Fibonacci, consulté le 26-04-2018.

[25] Wikipédia, https://fr.wikipedia.org/wiki/Suite_de_Fibonacci, consulté le 26-04-2018.

[26] Adda ALI-PACHA, Naima HADJ SAID, Suite de Fibonacci, Généralisée appliquée à la confidentialité des Données. Actes de la Conférence Internationale sur le Traitement de l'Information Multimédia CITIM, 2015.

[27] Yicong Zhou, Karen Panetta, Sos Agaian, and CL Philip Chen. Image encryption using pfibonacci transform and decomposition. Optics Communications, 285(5): 594–608, 2012.

[28] Weijia Cao, Yicong Zhou, C.L. Philip Chen. A New Image Encryption Algorithm Using Truncated P-Fibonacci Bit-planes. IEEE International Conference on Systems, Man, and Cybernetics, 2012.

[29] T. Hamaizia, Systèmes Dynamiques et Chaos "Application à l'optimisation a l'aide d'algorithme chaotique", These pour obtenir le titre de Docteur en Sciences de l'Université de Constantine 1, 2013.

[30] S. BELKACEM, Chaos based image watermarking, These Présentée pour l'obtention du diplôme de DOCTORAT en Science en Electronique, université de Batna 2.

[31] D.E. Goumidi, Fonction logistique et standard chaotique pour le chiffrement des images

satellitaires, Mémoire Présenté pour l'obtention du diplôme de Magister, école doctorale en Electronique Spécialité, 2010.

[32] M.MADANI, Y.BENTOUTOU, Cryptage d'images médicales à la base des cartes chaotiques.

[33] Tiegang Gao, Zengqiang Chen, A new image encryption algorithm based on hyper-chaos, Physics Letters A, 372(4):394–400, 2008.

[34] Baydda Flaeh AL-Saraji, Mustafa Dhiaa AL-Hassani, Multi-Levels Image Encryption Technique based on Multiple Chaotic Maps and Dynamic Matrix, International Journal of Computer Applications: (0975 – 8887) Volume 151, 2016.

[35] G.A.Sathishkumar, Dr.K.Bhoopathy bagan, Dr.N.Sriraam. Image encryption based on diffusion and multiple chaotic maps. International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.2, 2011.

[36] Avi Dixit, Dahale Bhagwan, Pratik Dhruve, IMAGE ENCRYPTION USING PERMUTATION AND ROTATIONAL XOR TECHNIQUE, SIPM, FCST, ITCA, WSE, ACSIT, CS & IT 06, pp. 01–09, 2012.

[37] site web, http://villemin.gerard.free.fr/Wwwgvmm/Numerati/BINAIRE/Amusemen.htm, consulté le 27-04-2018.

[38] site web, https://www.sciencedirect.com/science/article/pii/S0143816616300689, consulté le 27-04-2018.

[39] Sesha Pallavi Indrakanti, P.S.Avadhani, Permutation based Image Encryption Technique, International Journal of Computer Applications (0975 – 8887),Volume 28– No.8, 2011.

[40] Ravi Prakash Dewangan, Chandrashekhar Kamargaonkar, Image Encryption using Random Permutation by Different Key Size, International Journal of Science, Engineering and Technology Research (IJSETR), Volume 4, Issue 10, 2015.

[41] University of Southern California, Base de données d'images, http://sipi.usc.edu/database/database.php?volume=misc, consulté le 3-05-2018.

[42] University of Waterloo, Base de données d'images, http://links.uwaterloo.ca/Repository.html, consulté le 3-05-2018

[43] University of Wisconsin-Madison, Base de données d'images, https://homepages.cae.wisc.edu/~ece533/images/, consulté le 3-05-2018

[44] F. K. Tabash, M.Q. Rafiq, M. Izharrudin. Image Encryption Algorithm based on Chaotic Map. International Journal of Computer Applications (0975 – 8887): Volume 64– No.13, 2013.

[45] G.S. Nandeesh, P.A. Vijaya, M.V, Sathyanarayana : AN IMAGE ENCRYPTION USING BIT LEVEL PERMUTATION AND DEPENDENT DIFFUSION : A Monthly Journal of

Computer Science and Information Technology , IJCSMC, Vol. 2, Issue. 5, 2013, pg.145 – 154.
[46] Rim Zahmoul, Ridha Ejbali and Mourad Zaied, Image encryption based on new Beta chaotic maps, Optics and Lasers in Engineering, 2017.
[47] Akram Belazi, Ahmed A Abd El-Latif, Safya Belghith, A novel image encryption scheme based on substitution-permutation network and chaos, Signal Process 2016;128:155–70. [ISSN 0165-1684].

ملخص

التسارع الكبير والتطورات الرائعة في معالجه الصور الرقمية وتكنولوجيات الاتصالات الشبكية جعلت الحياة سهلة وفي الوقت نفسه أضافت تعقيدا لعالم الأمن. فمن الضروري حماية معلومات الصورة المرسلة من الاستخدام غير المشروع. في هذا العمل المخصص لمذكرة نهاية الدراسة قد اقترحنا خوارزمية تشفير جديدة والتي يمكن تطبيقها على الصور الرمادية العادية والطبية والتي تعتمد على التهجين بين العديد من تقنيات التشفير مثل الخرائط الفوضوية اللوجستية، الجيبية والقياسية، بالإضافة إلى نظرية فيبوناكسي المعدلة وتقنيات التبديل. وقد أجريت مجموعة من المقارنات أظهرت نتائجها كفاءة العمل المطروح.

الكلمات المفتاحية: الصور الرقمية، الصور الطبية، الصورة السرية، التشفير، الخرائط الفوضوية، فيبوناكسي المعدلة، التبديل.

Abstract

The great acceleration and wonderful developments in digital image processing and network communication technologies have made life easy and at the same time added complexity to the security world. It is necessary to protect the image information sent from illegal use. In end of this memory, we have proposed a new encryption algorithm which can be applied to normal and medical greyscale images and that rely on hybridization among many encryption techniques such as the sine, logistic and standard of chaotic maps, the modified Fibonacci sequence and permutation techniques. A series of comparisons were made, the results demonstrated the efficacy of our method.

Key words: digital images, medical images, confidential image, encryption, chaotic maps, modified Fibonacci sequence, permutation.

Résumé

La grande accélération et les développements merveilleux dans le traitement d'image numérique et les technologies de communication de réseau ont rendu la vie facile et en même temps a ajouté la complexité au monde de sécurité. Il est nécessaire de protéger l'information d'image envoyée de l'utilisation illégale. À la fin de cette mémoire, nous avons proposé un nouvel algorithme de cryptage qui peut être appliqué aux images normales et médicales niveaux de gris et qui s'appuient sur l'hybridation de plusieurs techniques de chiffrement telles que les cartes chaotiques sine, logistique et standard, la suite de Fibonacci modifiée et techniques de permutation. Une série de comparaisons ont été faites, les résultats ont démontré l'efficacité de notre méthode.

Mots clés : Images numériques, images médicales, images confidentielles, chiffrement, cartes chaotiques, suite Fibonacci modifiée, permutation.