الجمهوريــة الجزائريــة الديمقراطيــة الشعبيــة
République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

وزارة التعليم العالي و البحث العلمي

Université 20 août 1955 – Skikda-

جامعة 20 أوت 1955 سكيكدة

FACULTE DE TECHNOLOGIE

THESE

Présentée Pour l'obtention du

Diplôme de

DOCTORAT

Filière : Electronique

Option : Communication et traitement de l'information

Par

HERBADJI Djamel

THEME

Transmission sécurisée des images par cryptage basé sur les systèmes chaotiques 1-D, dans les systèmes de communication.

Soutenue le 11/10/2020 devant le Jury:

OUCHTATI Salim	Professeur	Université 20 août 1955 Skikda	Président
DEROUICHE Nadir	Professeur	Université 20 août 1955 Skikda	Rapporteur
BELMEGUENAI Aissa	Professeur	Université 20 août 1955 Skikda	Co-rapporteur
FERDI Youcef	Professeur	ENS. Biotechnologie Constantine	Examinateur
DOGHMANE Noureddine	Professeur	Univ. Badji Mokhtar Annaba	Examinateur

العنوان: **النقل الآمن للصور عن طريق التشفير القائم على أنظمة 1D-الفوضوية، في أنظمة الاتصالات.**

ملخص:

يتطلب نقل المعلومات الرقمية سرية كبيرة خاصة بالنسبة للصور الرقمية. إن استخدام خوارزميات التشفير التقليدية مثل ، Rivest - Shamir-Adlemanومعيار التشفير المتقدم ، ومعيار تشفير البيانات لم تعد صالحة بما يكفي لحماية جميع أنواع الصور الرقمية. و لهذا السبب، أصبح تشفير الصور تحديا ملحا وقلق بالغ وهو اجتذبت العديد من الباحثين في السنوات الأخيرة. لقد تم تطوير العديد من مخططات تشفير الصور باستخدام تقنيات متنوعة مثل نظرية الكم، ترميز الحمض النووي، و نظرية الفوضى. ومع ذلك ، فإن معظم المخططات المقترحة لها عدة إخفاقات على مستوى الأمان. في تشفير الصور على أساس الخرائط الفوضوي ة، يعتمد أمن خوارزمية التشفير على خصائص الخرائط الفوضوية وهيكل الخوارزمية، وبالتالي هناك حاجة لتوزيع خريطة فوضوي أفضل. ومع ذلك ، تعاني الأنظمة الفوضوية الكلاسيكية ، مثل الخرائط اللوجيستية والتربيعية، من العديد من نقاط الضعف ، وليس أقلها هو النطاق المحدود لسلوكياتها الفوضوية ، بالإضافة إلى توزيع البيانات غير المنتظم للتسلسلات الفوضوية المتولدة. يمكن للخرائط الفوضوية ذات السلوكيات الفوضوية الضعيفة أن تجعل الأنظمة المشفرة عرضة للهجمات ويمكن كسرها بسهولة. في هذا العمل، اقترحنا تقنيتين مختلفتين للتغلب على عيوب هذه الخرائط الفوضوية عن طريق تحسين خصائص التوزيع الفوضوي من أجل تحسين أداء وفعالية تشفير الصور وأيضا للتغلب على قيود خوارزميات تشفير الصور الموجودة.

كلمات مفتاحيه : أنضمه التشفير القائمة على الفوضى , مولدات الأرقام الزائفة والفوضوية , تشفير الصور

Title : Secure transmission of images by encryption based on chaotic 1-D systems, in communication systems.

Abstract :

The transmission of digital information requires great secrecy especially for digital images. The use of conventional encryption algorithms such as Rivest- Shamir-Adleman, advanced encryption standard, and data encryption standard have not become valid enough to protect all types of digital images.For this reason, image encryption has become an urgent challenge and high concern which has attracted many researchers in recent years. Several image encryption schemes have been developed using diverse techniques such as quantum theory , DNA coding et chaos theory. However, most of the proposed schemes have several failures at the security level. In the image encryption based on chaotic maps, the encryption algorithm security depends on the characteristic of the chaotic maps and the structure of the algorithm, thus a better chaotic map distribution is required. However, classical chaotic systems such as logistic and quadratic maps suffer many weaknesses, not the least is the limited range of their chaotic conducts, in addition to non-uniform data distribution of the generated chaotic sequences. Chaotic maps with weak chaotic behaviors can make the cryptosystems vulnerable to attacks and can be easily broke. in this work, we have suggested two different techniques to overcome the disadvantages of these chaotic maps by improving the properties of chaotic distribution for better performance and effectiveness of image encryption and also to overcome the limitations of the existing image encryption algorithms.

Keywords: images security, Chaos-based cryptosystems, pseudo-chaotic number generators, Image encryption

Titre : Transmission sécurisée des images par cryptage basé sur les systèmes chaotiques 1-D, dans les systèmes de communication.

Résumé :

La transmission d'informations numériques nécessite un grand secret, en particulier pour les images numériques. L'utilisation d'algorithmes de chiffrement conventionnels tels que Rivest - Shamir - Adleman, la norme de chiffrement avancée et la norme de chiffrement des données ne sont pas devenues suffisamment valides pour protéger tous les types d'images numériques. Pour cette raison, le chiffrement des images est devenu un défi urgent et une préoccupation à attirer de nombreux chercheurs ces dernières années. Plusieurs schémas de cryptage d'images ont été développés en utilisant diverses techniques telles que la théorie quantique, le codage de l'ADN, la théorie du chaos. Cependant, la plupart des schémas proposés présentent plusieurs échecs au niveau de la sécurité. Dans le cryptage d'image basé sur des cartes chaotiques, la sécurité de l'algorithme de cryptage dépend de la caractéristique des cartes chaotiques et la structure de l'algorithme, donc une meilleure distribution des cartes chaotiques est nécessaire. Cependant, les systèmes chaotiques classiques tels que les cartes logistiques et quadratiques souffrent de nombreuses faiblesses, notamment la gamme limitée de leurs conduites chaotiques, en plus de la distribution non uniforme des données des séquences chaotiques générées. Les cartes chaotiques avec des comportements chaotiques faibles peuvent rendre les cryptosystèmes vulnérables aux attaques et peuvent être facilement brisées. Dans ce travail, nous avons suggéré trois techniques différentes pour surmonter les inconvénients de ces cartes chaotiques en améliorant les propriétés de la distribution chaotique pour de meilleures performances et efficacité de cryptage d'image et aussi pour surmonter les limites des algorithmes de cryptage d'image existants.

Mots clés : sécurisation des images, Cryptosystèmes basés sur le chaos, générateurs de nombres pseudo-chaotiques, Chiffrement d'image

Liste des publications

- ✓ **Colour image encryption scheme based on enhanced quadratic chaotic map** **Djamel Herbadji**, Aissa Belmeguenai, Nadir Derouiche, Hongjung Liu Date de publication 2019/10/24 Revue IET Image Processing Volume 14 Numéro 1 Pages 40-52
  https ://digital-library.theiet.org/content/journals/10.1049/iet-ipr.2019.0123
- ✓ **A Tweakable Image Encryption Algorithm Using an Improved Logistic Chaotic Map** Auteurs **Djamel Herbadji**, Nadir Derouiche, Aissa Belmeguenai, Abderrahmane Herbadji, Selma Boumerdassi Date de publication 2019/10/1 Revue Traitement du Signal Volume 36 Numéro 5 Pages 407-417
  http ://www.iieta.org/journals/ts/paper/10.18280/ts.360505
- ✓ **Personal authentication based on wrist and palm vein images** Auteurs Abderrahmane Herbadji, Noubeil Guermat, Lahcene Ziet, Mohamed Cheniti, **Djamel Herbadji** Date de publication 2019 Revue International Journal of Biometrics Volume 11 Numéro 4 Pages 309-327
  https ://www.inderscienceonline.com/doi/abs/10.1504/IJBM.2019.102860
- ✓ **A New Colour Image Encryption Approach using a Combination of Two 1D Chaotic Map** by **Djamel Herbadji**, Nadir Derouiche, Aissa Belmeguenai, Nedal Tahat, Selma Boumerdassi , International Journal of Electronic Security and Digital Forensics Vol. 12, No. 4, 2020
  https ://www.inderscience.com/info/inarticle.php ?artid=110649
- ✓ **Information Gathering and Controlling over the Internet by Internet of Things (IoT)** by Abderrahmane Herbadji, **Djamel Herbadji**, Abdelhadi Labiad , Review of Computer Engineering Studies Vol. 7, No. 3, September, 2020, pp. 49-54
  http ://www.iieta.org/journals/rces/paper/10.18280/rces.070301

Liste des communications

- ✓ **A new image encryption scheme using an enhanced logistic map**
  Djamel Herbadji, Nadir Derouiche, Aissa Belmeguenai, Toufik Bekkouche, Abdelhadi Labiad, Mohamed Lashab, Abderrahmene Herbadji 2018 International Conference on Applied Smart Systems (ICASS)
  https ://ieeexplore.ieee.org/abstract/document/8652065/
- ✓ **A new Algorithm for Image encryption and decryption**, **Djamel Herbadji**, Nadir Derouiche, Belmeguenai Aissa, Toufik Bekkouche, Abderrahmene Herbadji and Selma Boumerdassi. The Third International Symposium on Informatics and its Applications November 6-7, 2018 in M'sila .

✓ **A Novel Color Image Encryption Scheme Using Logistic Map and Quadratic Map Systems** Auteurs **Djamel Herbadji**, Aissa Belmeguenai, Nadir Derouiche, Youcef Zennir, Salim Ouchtati Date de publication 2018/6/18 Conférence International Conference on Mobile, Secure, and Programmable Networking
$https://link.springer.com/chapter/10.1007/978-3-030-03101-5_2$

Remerciement

"Soyons reconnaissants aux personnes qui nous donnent du bonheur : Elles sont les charmants jardiniers par qui nos âmes sont fleuries"

Après avoir rendu grâce à Allah le tout puissant et le miséricordieux, nous tenons à remercier vivement tous ceux qui, de près ou de loin ont participé à la préparation de cette thèse. Il s'agit plus particulièrement de : mon directeur et mon co-directeur de thèse, Messieurs Nadir Derouiche et Aissa Belmeguenai, pour leurs conseils avisés qui ont été prépondérants pour la bonne réussite de cette thèse.

Pour la confiance qu'ils m'ont témoigné en acceptant la direction scientifique de mes travaux, je leur suis reconnaissant de m'avoir fait bénéficier tout au long de ce travail de leur grande compétence, de leur rigueur intellectuelle, de leur dynamisme, de leur efficacité certaine que je n'oublierai jamais. Soyez assuré de mon attachement et de ma profonde gratitude.

Je suis très honoré de remercier pour leur présence à mon jury de thèse :Messieurs OUCHTATI Salim Professeur à l'université de Université 20 août 1955 -Skikda- , FERDI Youcef Professeur à ENS. Biotechnologie Constantine et DOGHMANE Noureddine Professeur à l'université de Badji Mokhtar Annaba , pour le temps consacré à la lecture de cette thèse, et pour les suggestions et le remarques judicieuses qu'ils ne manquerons pas de m'indiquer.

Ce travail de thèse s'est déroulé au sein du Laboratoire de Recherche en Electronique de Skikda, à l'Université 20 Aout 1955 de Skikda. Je remercie tous les membres du laboratoire et tous les membres du doctorat " Communication et traitement de l'information" sous la direction de Monsieur OUCHTATI Salim Professeur à l'Université 20 août 1955 de Skikda.

Je voudrais aussi adresser toute ma reconnaissance à mes professeurs des universités de Setif, Djelfa et Skikda, qui m'ont fourni les outils nécessaires à la réussite de mes études universitaires. À titre plus personnel, j'aimerais exprimer ma gratitude à mes parents et mes frères pour leur grande patience, leurs encouragements et leur amour, ainsi que pour leur soutien qui m'a été bien utile durant ma thèse. À tous mes amis et mes collègues qui m'ont apporté leur soutien moral et intellectuel tout au long de ma démarche.

Merci

Djamel Herbadji

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations

✓ **PLCM** :Piecewise Linear Chaotic Map

✓ **DES** : Data Encryption Standard

✓ **RSA** : Ronald Rivest, Adi Shamir et Leonard Adleman

✓ **AES** : Advanced Encryption Standard

✓ **PRNG** : Pseudo Random Number Generator

✓ **DCT** : Discrete Cosine Transform

✓ **NPCR** : Number of Pixels Change Rate

✓ **UACI** : Unified Average Changing Intensity

✓ **NIST** : National Institute of Standards Technologie

✓ **COA** : Attaque sur texte chiffré seul

✓ **CPA** : Attaque texte clair choisi

✓ **PWLCM** : Piecewise Linear Chaotic Map

✓ **RVB** : Rouge Vert Bleu

✓ **RGB** : Red Green Blue

✓ **XOR** : OU exclusif

✓ **EQM** :Enhanced quadratic map

Introduction Générale

Les développements modernes dans les technologies des télécommunications ont conduit à de grands défis dans tous les domaines. Le plus important de ces développements est la transmission d'informations via des réseaux et des ordinateurs qui ont facilité la vie des gens en réduisant le volume de travail et en développant des méthodes de production, de stockage et de distribution d'informations. Cependant, ces développements ont posé de nouveaux défis afin de sécuriser le transfert et le stockage de ces informations contre les utilisateurs non autorisés. De nombreux chercheurs ont relevé ces défis en concevant différents algorithmes de cryptage, tels que le cryptage d'images numériques. Ce dernier devient un problème important attirant les chercheurs, car les propriétés des images numériques sont distinctes des informations textuelles, telles que la quantité de données élevées et la corrélation étroite entre le pixel adjacent [2]. Le cryptage d'images peut être défini comme la méthode de transformation de l'image entière en une image non reconnue. Il vise également par le chiffrement d'images à protéger l'image numérique transmise sur les réseaux publics et partagés contre les accès non autorisés. Le chiffrement d'images est nécessaire pour exécuter des applications du monde réel qui utilisent des images numériques, car il protège principalement contre diverses attaques. De nombreuses applications du monde réel qui utilisent des images numériques, en particulier pour les bases de données d'images militaires, l'imagerie médicale et les albums de photos personnels en ligne, nécessitent un système de sécurité robuste et rapide pour transmettre et stocker des images numériques. Le système de cryptage d'image est très important pour des applications telles que le courrier électronique, un dossier médical, un document juridique car il les protège de toute attaque, en plus d'autres exigences telles que la précision et la préservation du format d'image. Pour toutes ces contraintes, on peut voir que les algorithmes de chiffrement conventionnels tels que Rivest-Shamir Adleman (RSA), la norme de chiffrement avancée (AES) et la norme de chiffrement des données (DES) ne conviennent pas au chiffrement d'images numériques. Cela est dû à la difficulté d'appliquer ces algorithmes à l'image, en plus du traitement lent [3]. Des travaux de recherche récents suggèrent des algorithmes efficaces, la plupart d'entre eux sont basés sur des systèmes chaotiques. Les systèmes chaotiques présentent de nombreux avantages tels qu'une sensibilité élevée aux variations des valeurs initiales et la capacité de générer facilement une valeur pseudo-aléatoire de la même manière que le bruit [4][5]. Habituellement, lorsque de petites variations sont introduites dans les valeurs initiales ou dans les paramètres de contrôle, nous obtenons une séquence aléatoire différente. Ces fonctionnalités rendent les schémas de chiffrement basés sur des systèmes chaotiques extrêmement efficaces en termes de sécurité et de rapidité. Ces systèmes sont utilisés en permutation (mélange de pixels) et en diffusion (modification des valeurs des pixels) [6].

Mais, ces cartes chaotiques ont également quelques faiblesses [7] telles que la gamme limitée de comportements chaotiques et la distribution des données des séquences chaotiques de sortie sont non uniformes. Par conséquent, il est très important d'améliorer un système chaotique qui offre de meilleures performances chaotiques. La motivation de ce travail est d'analyser les lacunes de certaines cartes chaotiques en termes de séquences pseudo-aléatoires et également d'améliorer leur comportement chaotique afin de convenir au chiffrement d'images. Sur la base de cette carte chaotique améliorée, trois algorithmes de cryptage d'images efficaces ont proposé dans ce travail, afin d'assurer les exigences de sécurité de la transmission d'images numériques
Les travaux de la thèse sont organisés comme suit :

- ✓ Dans le premier chapitre, nous abordons l'état de l'art et les outils de mesure des performances et, nous rappelons les principales notions relatives au chiffrement, incluant une description sommaire des différentes techniques de chiffrement.

- ✓ Dans le deuxième chapitre, nous abordons l'évaluation des performances de certaines cartes chaotiques comme bases de chiffrement utilisant le chaos proposé.

- ✓ Dans le troisième chapitre, nous proposons un schéma de cryptage des images couleur basé sur une carte chaotique quadratique améliorée.

- ✓ Le quatrième chapitre, présente une nouvelle technique de chiffrement d'image modifiable utilisant une carte chaotique logistique améliorée.

- ✓ Le cinquième chapitre, présente une nouvelle approche de cryptage d'image couleur utilisant une combinaison de deux cartes chaotiques 1D

Enfin, nous terminons par une conclusion générale qui donne une synthèse des travaux réalisés et des résultats obtenus.

Chapitre 1

Etat de l'art et outils de mesure des performances

Sommaire

1.1 Introduction

La transmission d'informations par les médias sur divers réseaux de communication est devenue très courante de nos jours en raison du développement rapide d'Internet et des technologies de réseau. Par conséquent, il est important de garantir la sécurité de la transmission des informations contre la copie et la distribution illégale sen particulier les images fréquemment utilisées sur les réseaux de communication. Les stratégies de protection des informations couramment utilisées incluent le cryptage des données, la signature numérique, la stratégie de routage de confiance, etc. Parmi eux, le cryptage des informations en tant que solution commune et efficace de protection des informations[8]. Nous dédions ce chapitre pour expliquer tout d'abord les concepts fondamentaux des primitives de cryptographie et cryptage d'images, nous décrivons brièvement les objectifs de la cryptographie ainsi que les principes de la cryptanalyse et les deux principales familles de cryptosystèmes à savoir les algorithmes symétriques et asymétriques. Nous discutons en détail chiffrement par bloc et chiffrement par flux et aussi les techniques d'évaluation des algorithmes de cryptage.

1.2 Concepts de base

Presque depuis le début du langage écrit, il était nécessaire de trouver des moyens de cacher des informations précieuses [9]. La cryptographie est la science qui concerne la transformation de l'information de sorte qu'il ne soit pas possible pour d'autres personnes différentes de la source et de la destination légitimes d'accéder à cette information[10]. La sécurité des images est basée sur la cryptographie, où nous utilisons des concepts de base de la cryptographie comme blocs de construction pour des applications en sécurité image[11]. Pour mieux clarifier les questions relatives à la sécurité de l'image, nous présentons une revue générale de la science de la cryptographie.

1.2.1 Cryptologie

La cryptologie est une science basée sur les mathématiques, qui peut être subdivisée en deux branches : la cryptographie et la cryptanalyse.

1.2.1.1 Cryptographie

La cryptographie est une science concernée par la construction de systèmes répondant aux préoccupations de sécurité. Cryptographie signifie exactement les techniques de masquage du contenu des messages, mais également la science de la protection des informations sensibles, car elle permet de transformer les données lisibles en données de charabia et en données qui semblent aléatoires et insensées. Les informations que nous devons masquer sont appelées

texte en clair. Elles peuvent prendre la forme de caractères, de données numériques, d'images numériques ou de tout autre type d'informations. Les informations qui seront transmises sont appelées ciphertext, ce qui signifie des données dénuées de sens ou un texte peu clair que personne ne peut comprendre, à l'exception des destinataires.

1.2.1.2 Cryptanalyse

La cryptanalyse est la science qui consiste à étudier les cryptosystèmes en trouvant leurs faiblesses afin de trouver des messages clairs correspondant à des messages cryptés sans disposer de la clé de cryptage, et l'opération permettant de comprendre un message crypté s'appelle une attaque. Lorsque des messages en clair sont obtenus à partir de messages cryptés sans utiliser la clé de cryptage, il est dit que le cryptosystème utilisé a été brisé. Plus un cryptosystème est difficile à détruire, plus il est sécurisé. La personne qui pratique la cryptanalyse s'appelle :cryptanalyste.

1.2.2 Objectifs de la cryptographie

La cryptographie est un mécanisme mathématique utilisé pour atteindre plusieurs objectifs afin d'assurer la sécurité de la communication. Ces objectifs peuvent tous être atteints simultanément, dans une seule application, ou seulement certains d'entre eux. Ces objectifs sont[12] :

1.2.2.1 Confidentialité

La confidentialité est une opération qui garantit que personne ne peut comprendre et accéder aux données reçues, à l'exception des utilisateurs autorisés. Un bon exemple de confidentialité peut être obtenu en utilisant des algorithmes de chiffrement car le processus de déchiffrement n'est activé que par les utilisateurs autorisés.

1.2.2.2 Authentification

L'authentification consiste simplement à confirmer et à tester l'identité d'un utilisateur et à s'assurer qu'il s'agit bien de la personne qu'il prétend être, et à garantir à chacun des correspondants que son partenaire est celui qu'il croit être[13].

1.2.2.3 Intégrité

Un mécanisme garantit au destinataire que les données n'ont pas été modifiées lors de la transmission. Un troisième utilisateur ne doit pas pouvoir substituer un message légitime (provenant de l'expéditeur) par un message frauduleux ou accidentel.[14].

1.2.2.4 Non-répudiation

La non-répudiation est un mécanisme permettant de prouver qu'un message transféré a été réellement envoyé par l'expéditeur et reçu par le destinataire. Par conséquent, l'expéditeur ne peut pas prétendre qu'il n'a pas réellement envoyé le message et que le destinataire ne peut pas refuser la réception du message.

1.3 Terminologies de cryptage d'image

1.3.1 Algorithme de cryptage

Le chiffrement est un mécanisme permettant de convertir une image lisible et compréhensible en une image qui semble aléatoire et du bruit (image peu claire)[8] en utilisant une clé secrète pour rendre la compréhension d'une image impossible pour toute personne ne possédant pas cette clé secrète.

1.3.2 L'algorithme de décryptage

L'algorithme de décryptage est le mécanisme qui permet le retour dans l'image originale (image claire) à partir de l'image cryptée (image non claire) et est l'inverse de l'algorithme de cryptage. Le succès du processus de déchiffrement est impossible pour ceux qui ne possèdent pas la clé secrète.

1.3.3 Image simple

L'image brute est l'image originale ou claire qui sera cryptée afin de la protéger.

1.3.4 Image cryptée

Une image cryptée (chiffrée) est le résultat du cryptage de l'image en clair et de sa transformation en une image incompréhensible.

1.3.5 Image décryptée

L'image décryptée est le résultat du décryptage de l'image cryptée, c'est-à-dire, la transformez en une image compréhensible.

1.3.6 Clés secrètes

La clé secrète est un ensemble de données utilisé dans les algorithmes de cryptage d'image, où sa taille est mesurée en bits. Plus la taille de la clé est grande, plus la sécurité augmente. Les clés doivent être échangées dans un canal sécurisé et seul le propriétaire est en mesure de les atteindre et de les utiliser.

1.4 Principe d'un système cryptographique

Le cryptosystème affiché à la Fig. 1.1 décrit une méthode de cryptage / décryptage d'image. Où, chez l'émetteur, le message "image plaintext noté par P" est transformé au moyen d'une "fonction de cryptage notée par E" en un message crypté "image ciphertext noté par C ". E transforme P en C en utilisant la formule suivante :

$$C = E_{ke}(P) \tag{1.1}$$

Où ke est la clé de cryptage. Et chez le destinataire, le message chiffré est transformé à l'aide d'une "fonction de déchiffrement notée par D" en un message clair "image claire". D transforme C en P en utilisant la formule suivante :

$$P = D_{kd}(C) \tag{1.2}$$

Où kd est la clé de décryptage.

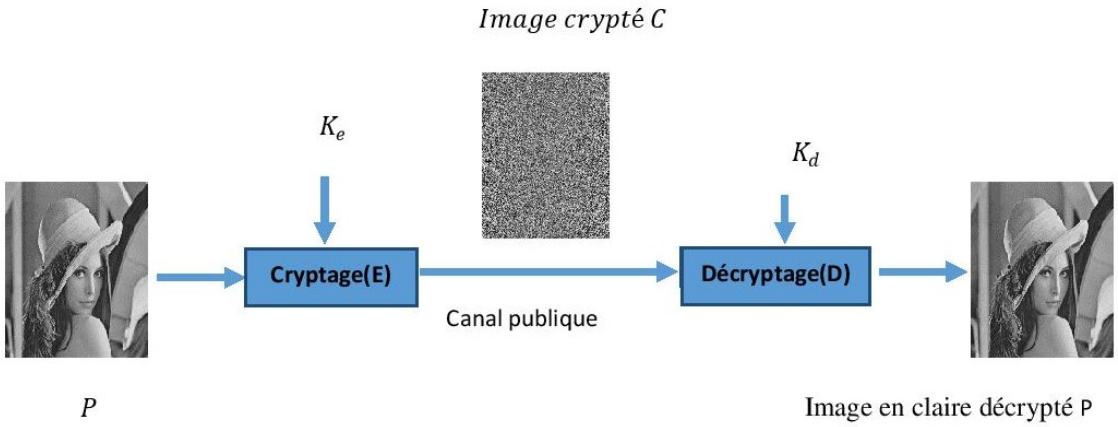


Fig. 1.1 *Cryptage et décryptage d'une image*

1.5 Principe de Kerckhoffs

Le principe de Kerckhoffs, formalisé par Auguste Kerckhoffs est à l'origine de systèmes de cryptographie modernes. Kerckhoffs expose une liste de six exigences, connues aujourd'hui sous le nom de principes de Kerckhoffs, que doit satisfaire un système de chiffrement pour protéger pendant un temps illimité les correspondances entre les membres d'une organisation comme l'armée[15] :

- ✓ Le système doit être matériellement, sinon mathématiquement indéchiffrable.
- ✓ Il faut qu'il n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi
- ✓ La clef doit pouvoir en être communiquée et retenue sans le secours de notes écrites, et être changée ou modifiée au gré des correspondants
- ✓ Il faut qu'il soit applicable à la correspondance télégraphique .
- ✓ Il faut qu'il soit portatif, et que son maniement ou son fonctionnement n'exige pas le concours de plusieurs personnes.
- ✓ Enfin, il est nécessaire, vu les circonstances qui en commandent l'application, que le système soit d'un usage facile, ne demandant ni tension d'esprit, ni la connaissance d'une longue série de règles à observer.

1.6 Classification en fonction de la clé de cryptage

Dans cette section, nous présentons les types d'algorithme de chiffrement en fonction des clés, où il existe deux types de chiffrement en fonction de la relation entre les clés de chiffrement et de déchiffrement.Il peut être classé sous différents aspects : selon les clés, selon les structures des algorithmes, [11].

1.6.1 Cryptage symétrique

Le chiffrement symétrique est une forme de cryptosystème dans lequel les opérations de chiffrement et de déchiffrement sont effectuées à l'aide de la même clé, comme illustré à la Fig.1.2 La norme de chiffrement de données (Data Encryption Standard, ou DES) est un exemple de cryptosystème symétrique largement utilisé par le gouvernement américain. Pour les cryptages à clé symétrique, la clé de cryptage / décryptage doit être transmise de l'expéditeur au destinataire via un canal sécurisé séparé. [14].

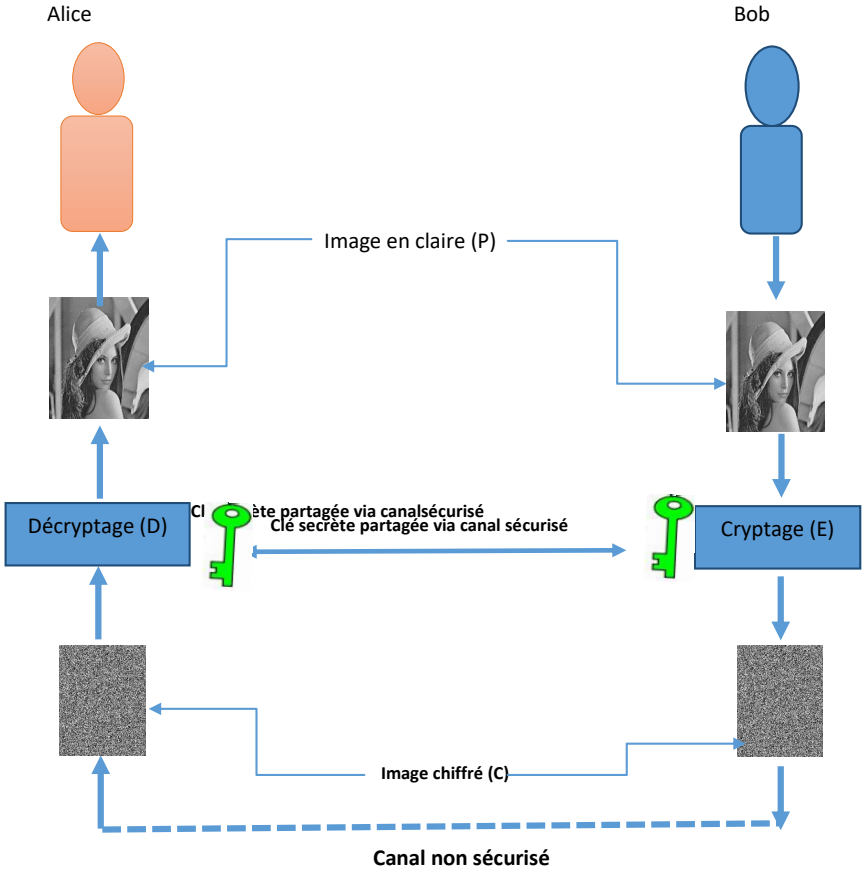


Fig. 1.2 *Cryptage symétrique*

1.6.2 Cryptage asymétrique

Lorsque l'expéditeur et le destinataire utilisent des clés différentes ($Ke \neq Kd$), le cryptosystème est appelé cryptage à clé publique ou cryptosystème asymétrique, comme illustré à la Fig.1.3 Pour les chiffrements à clé publique, la clé de chiffrement Ke est connue de tous les utilisateurs, tandis que la clé de déchiffrement Kd est maintenue privée, pour laquelle aucun canal secret supplémentaire n'est nécessaire pour le transfert de clé[11]. Au lieu de cela, chaque utilisateur applique l'algorithme de génération de clé pour produire une paire de clés [16].

1.6.3 Classification en fonction de la structure du cryptage

Les algorithmes de chiffrement peuvent être classés en fonction de la structure de chiffrement en chiffrements par blocs et chiffrements de flux [11].

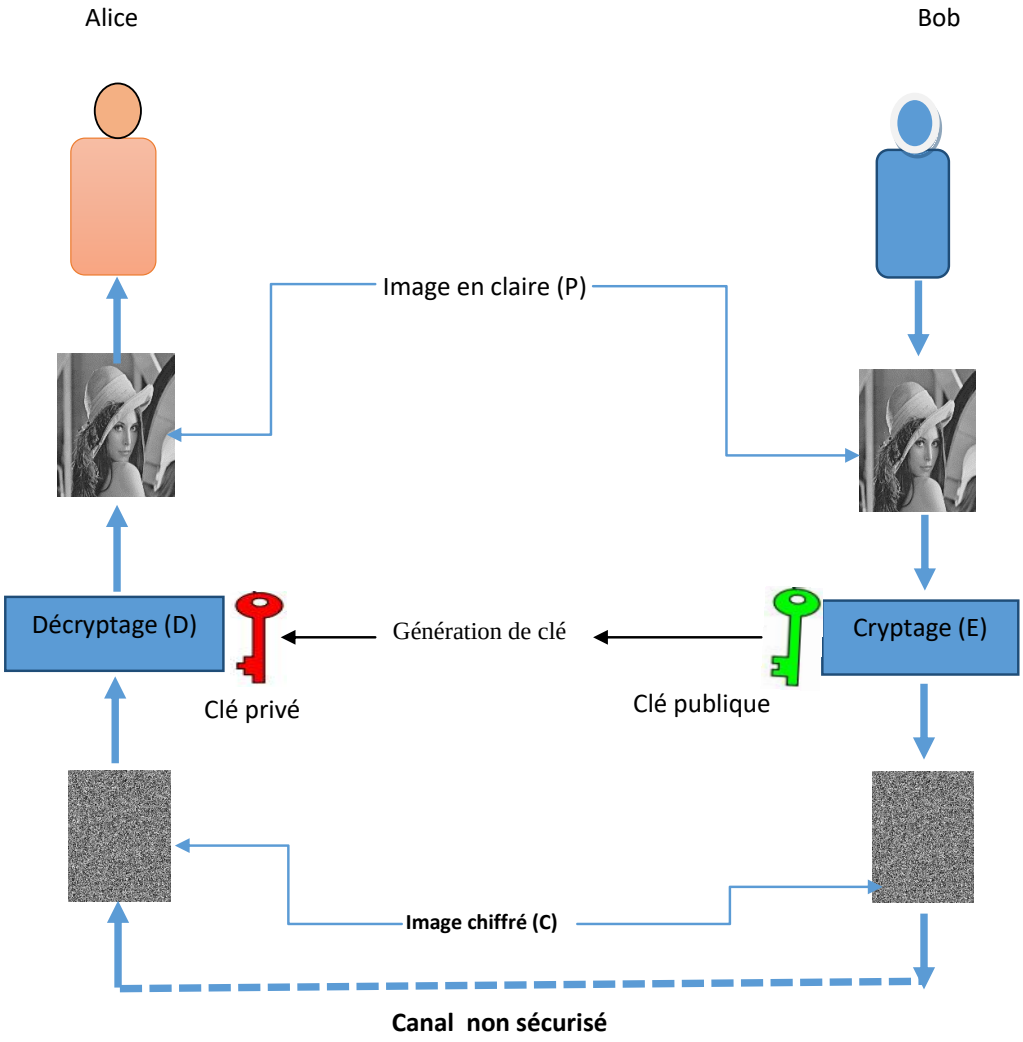


Fig. 1.3 *Cryptage asymétrique*

1.6.3.1 Chiffrement par blocs

Un chiffrement par bloc est un schéma de chiffrement qui décompose les messages en clair à transmettre en chaînes (appelées blocs) de longueur fixe t au lieu de bits individuels. Les techniques de chiffrement à clé symétrique les plus connues sont les chiffrements par blocs[17]. L'avantage des chiffrements de blocs réside dans le fait que la sortie de chiffrement des chiffreurs de blocs dépend étroitement du contenu des données en clair, la sortie change presque complètement lorsqu'un seul bit de l'entrée est modifié[18].

1.6.3.2 Chiffrement par flot

Le principe de ce système consiste à générer une clé aléatoire de même taille de message à l'aide d'un générateur de nombres pseudo-aléatoires (PRNG) et à chiffrer en clair un octet ou un bit à la fois en le combinant à la clé aléatoire par l'opération XOR dans laquelle la sortie est produite bit par bit ou octet par octet à partir d'un flux d'entrée en texte clair, l'opération XOR au niveau du bit est choisie généralement pour sa simplicité et son efficacité dans le processus de cryptage. De plus, le récepteur applique le même mécanisme [19].

1.7 Types d'attaque sur un chiffrement

Selon le principe de Kerckhoffs, lors de l'analyse d'un algorithme de chiffrement, une hypothèse est que le cryptanalyste connaisse exactement la conception et le fonctionnement du système cryptographique, à l'exception des clés secrètes. À savoir, l'attaquant connaît tous les mécanismes de fonctionnement du cryptosystème mais ne connaît pas les clés secrètes[8]. Un cryptanalyste essaye de casser le chiffrement sans connaître la clé secrète, et ceci avec plusieurs niveaux de difficultés basés sur quatre types d'attaques classiques :

1.7.1 Attaque sur texte chiffré seul(COA)

L'attaque texte chiffré seul est une modèle d'attaque pour cryptanalyse où l'attaquant ne peut accéder qu'à certains textes cryptés. Une attaque COA réussie quand on peut déterminer le texte en clair à partir du texte chiffré Il y a de nombreuses façons d'exécuter une attaque de texte chiffré uniquement : attaque de force brute, attaque d'analyse statistique. Cette attaque peut être illustrée dans la figure 1.4

1.7.2 Attaque texte clair connu(KPA)

L'attaque en texte clair connu est un modèle d'attaque pour cryptanalyse où l'attaquant dispose de quelques échantillons du texte en clair et du texte chiffré correspondant et s'en sert

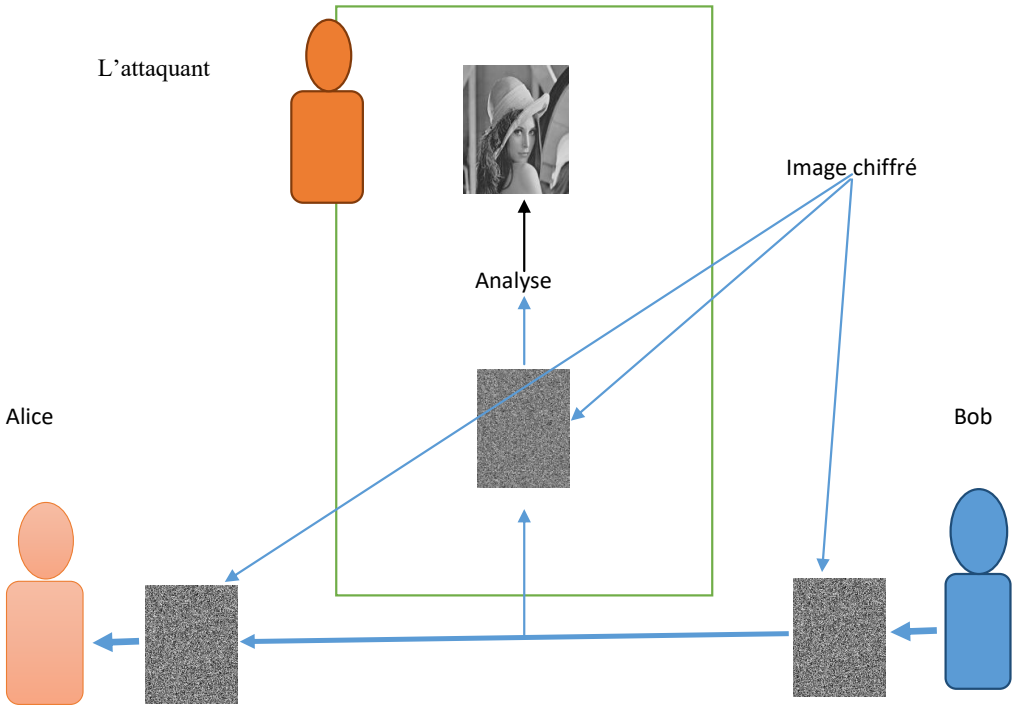


Fig. 1.4 *Attaque texte chiffré seul.*

pour révéler des informations secrètes, telles que des clés secrètes et / ou ses équivalents[20]. Cette attaque peut être illustrée dans la figure 1.5

1.7.3 Attaque texte clair choisi(CPA)

Attaque texte clair choisi(CPA) qui définit un attaquant capable d'accéder temporairement à la machine de chiffrement et essayant de déchiffrer une image chiffrée sans clé secrète. Pour cette attaque, l'attaquant choisit de manière adaptative des images ordinaires, construit les images de chiffrement correspondantes en accédant à la machine de chiffrement, puis compare les images de chiffrement acquises afin d'extraire des termes inconnus concernant le système chiffré[21]. Cette attaque peut être illustrée dans la figure 1.6

1.7.4 Attaque texte chiffré choisi(CCA)

L'opposant a obtenu un accès temporaire à la machine de déchiffrement. Par conséquent, il peut choisir n'importe quel texte chiffré et obtenir le texte en clair correspondant. L'adversaire tente alors d'obtenir la clé secrète ou une partie de la clé en analysant les paires de texte chiffré / texte accumulé[8]. Cette attaque peut être illustrée dans la figure 1.7

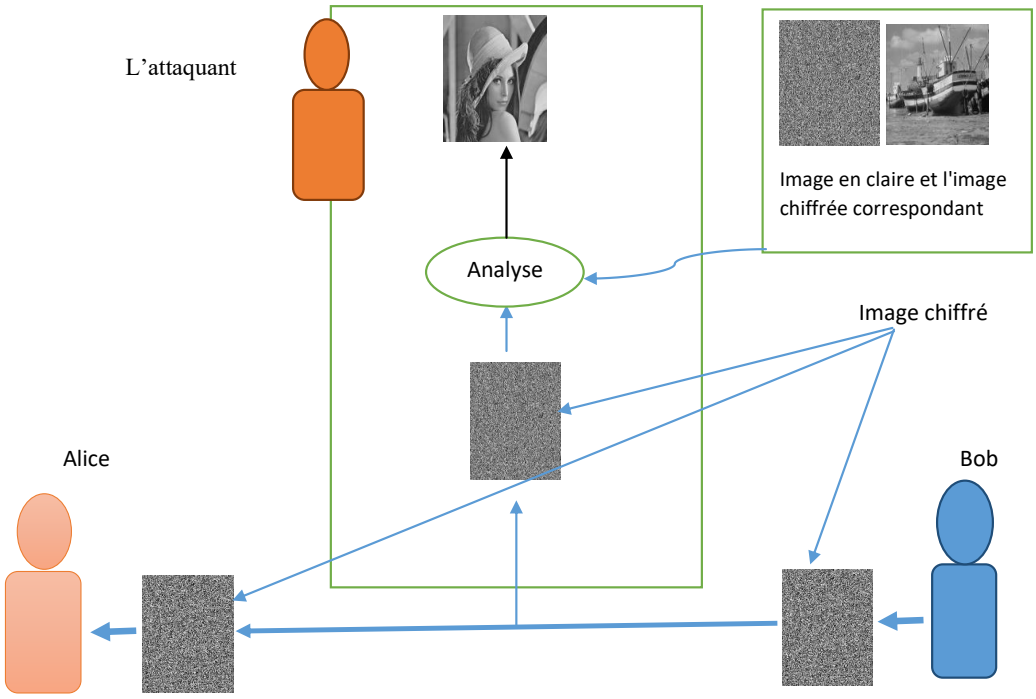


Fig. 1.5 *Attaque texte clair connu.*

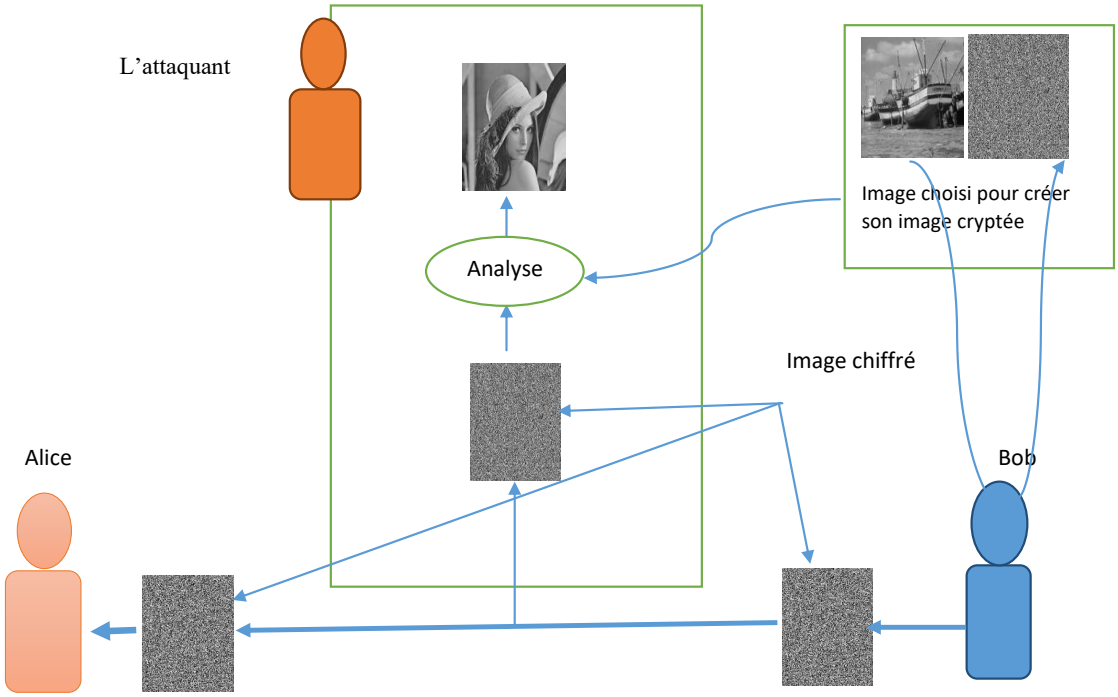


Fig. 1.6 *Attaque texte clair choisi(CPA).*

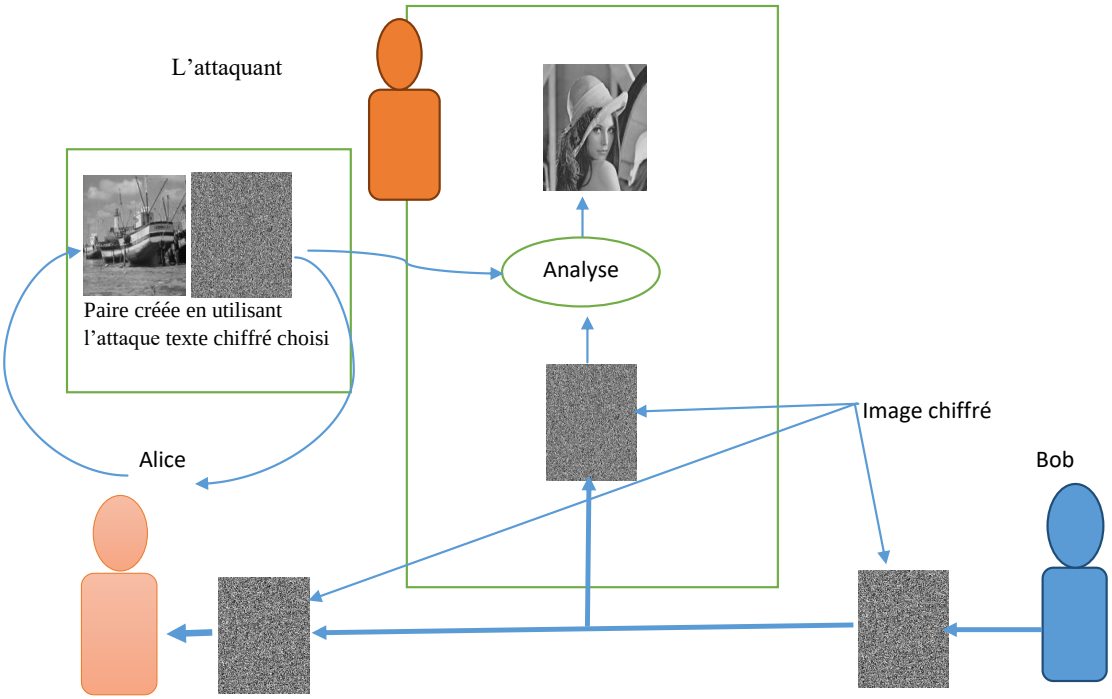


Fig. 1.7 *Attaque texte chiffré choisi(CCA).*

1.8 Outils et standard d'évaluation de sécurité communs des algorithmes de cryptage d'images

Dans cette section, nous donnons la définition des principales attaques connues, puis nous décrivons en détail les mesures de sécurité statistiques bien connues et les outils d'évaluation des algorithmes de chiffrement d'images. Nous utiliserons ces mesures et outils dans les chapitres 3,4 et 5.

1.8.1 La correlation

Dans les images, tout pixel est extrêmement corrélé avec les pixels voisins, soit horizontalement, soit verticalement, même en diagonale. Un schéma de chiffrement sécurisé devrait supprimer cette corrélation entre les pixels voisins et rendre les valeurs de corrélation plus proches de zéro, en maintenant le système résistant à toute attaque statistique. Pour tester la sécurité de tout nouvel algorithme, concernant ce type d'attaques, N paires aléatoires de pixels adjacents dans les directions verticale, horizontale et diagonale sont sélectionnées à partir de l'image standard et de sa version chiffrée, puis la corrélation est analysée. Le coefficient de

corrélation est donné par[22] :

$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sqrt{\mathrm{D(x)D(y)}}}, \tag{1.3}$$

$$\mathrm{cov}(x, y) = \mathrm{E([x\text{-}E(x)][y\text{-}E(y)])}, \tag{1.4}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i), \tag{1.5}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}[x_i - E(x)]^2, \tag{1.6}$$

Où x et y sont les valeurs d'échelle de gris des deux pixels adjacents de l'image, E (x) est l'attente de x, D (x) est la variance et N est le nombre total d'échantillons.

1.8.2 L'entropie

Selon la théorie de Shannon , l'entropie d'une information est la quantité d'information englobée ou libérée par une source d'information. Also , elle est l'une des principales mesures de l'aléatoire de l'information[13]. Les valeurs des pixels de l'image vont de 0 à 255. Dans un algorithme de chiffrement robuste, la probabilité d'occurrence d'un pixel doit être la même (ou presque la même). Le comportement aléatoire du message chiffré peut être évalué à l'aide des informations d'entropie définies par :[23]

$$H(C) = \sum_{i=0}^{i=255} Pro(c_i) \times \log_2 \frac{1}{Pro(c_i)}. \tag{1.7}$$

Où H (C) est l'entropie de l'image chiffrée C, Pro (c_i) est le numéro d'occurrence de chaque niveau (i = 0,1,2 ... 255).

En cas d'égalité de probabilités (Pro (c_i) = 2^{-8}), l'entropie d'information est maximale, H (C) = $\sum_{i=0}^{i=255} 2^{-8} \times \log_2 256 = 8$, selon 1.7.

1.8.3 Analyse de l'espace de la clé de cryptage

La taille de l'espace de clé est le nombre total de clés différentes pouvant être utilisées dans un algorithme de cryptage. Pour un bon modèle de chiffrement, l'espace clé doit être suffisamment grand pour résister à l'attaque par force brute. La taille de la clé doit être supérieure à 2^{100} pour assurer un niveau de sécurité élevé [2].

1.8.4 L'histogramme

L'analyse par histogramme est une méthode graphique d'analyse de la distribution de fréquence des valeurs d'intensité de pixel en niveaux de gris dans une image[24]. L'axe horizontal représente toutes les valeurs du niveau du gris de 0 à 255 et l'axe vertical indique le nombre de pixels ayant le niveau du gris correspondant[14]. L'histogramme de l'image joue un rôle important dans l'analyse de l'image. Un schéma de chiffrement d'image idéal devrait être robuste contre toute attaque statistique[25]. La figure 1.8 montre les histogrammes de l'image originale de Lena et l'image cryptée correspondante de Lena cryptée par l'algorithme proposé au chapitre 3. Les figures montrent clairement que les histogrammes de l'image cryptée sont très uniformes et très différents des histogrammes de l'image simple, ce qui rend les attaques statistiques difficiles, il ne peut donc pas être utile pour le déchiffrement.

1.8.5 Sensibilité à la clé de cryptage

Les caractéristiques les plus importantes d'un bon système cryptographique sont sa grande sensibilité aux modifications des clés utilisées dans le processus de cryptage et de décryptage. Par exemple, l'analyse de sensibilité de clé dans le processus de cryptage est effectuée en modifiant la différence d'un bit par rapport à la clé secrète d'origine utilisée dans le processus de cryptage et en maintenant les clés secrètes restantes inchangées. Ensuite, l'algorithme de décryptage d'image proposé est appliqué à l'image cryptée par en utilisant la clé secrète modifiée, cela conduit à l'échec complet du déchiffrement .

1.8.6 Analyse de l'attaque différentielle

Les attaques différentielles sont généralement considérées comme une méthode permettant de casser un système cryptographique, qui tente de modifier légèrement une image standard et d'obtenir l'image de chiffrement correspondante. Toute personne tentant de lancer une telle attaque modifierait d'abord légèrement une image standard avant de la chiffrer et d'obtenir l'image de chiffrement modifiée. Pendant ce temps, l'image d'origine est cryptée avec le même cryptosystème. Ensuite, deux images de cryptage sont comparées pour obtenir la relation entre les images simples et les images crypté[26]. Les NPCR et les UACI définis ci-après sont couramment utilisés pour tester l'influence d'un changement d'un bit dans l'image chiffrée et pour mesurer les performances de l'image chiffrée afin de résister aux attaques différentielles. Ils sont définis par :

$$NPCR = \frac{1}{n \times m} \sum_{i,j} D(i,j) \times 100\%, \tag{1.8}$$

$$UACI = \frac{1}{n \times m} \sum_{i,j} \frac{C_1(i,j) - C_2(i,j)}{255} \times 100\%, \tag{1.9}$$

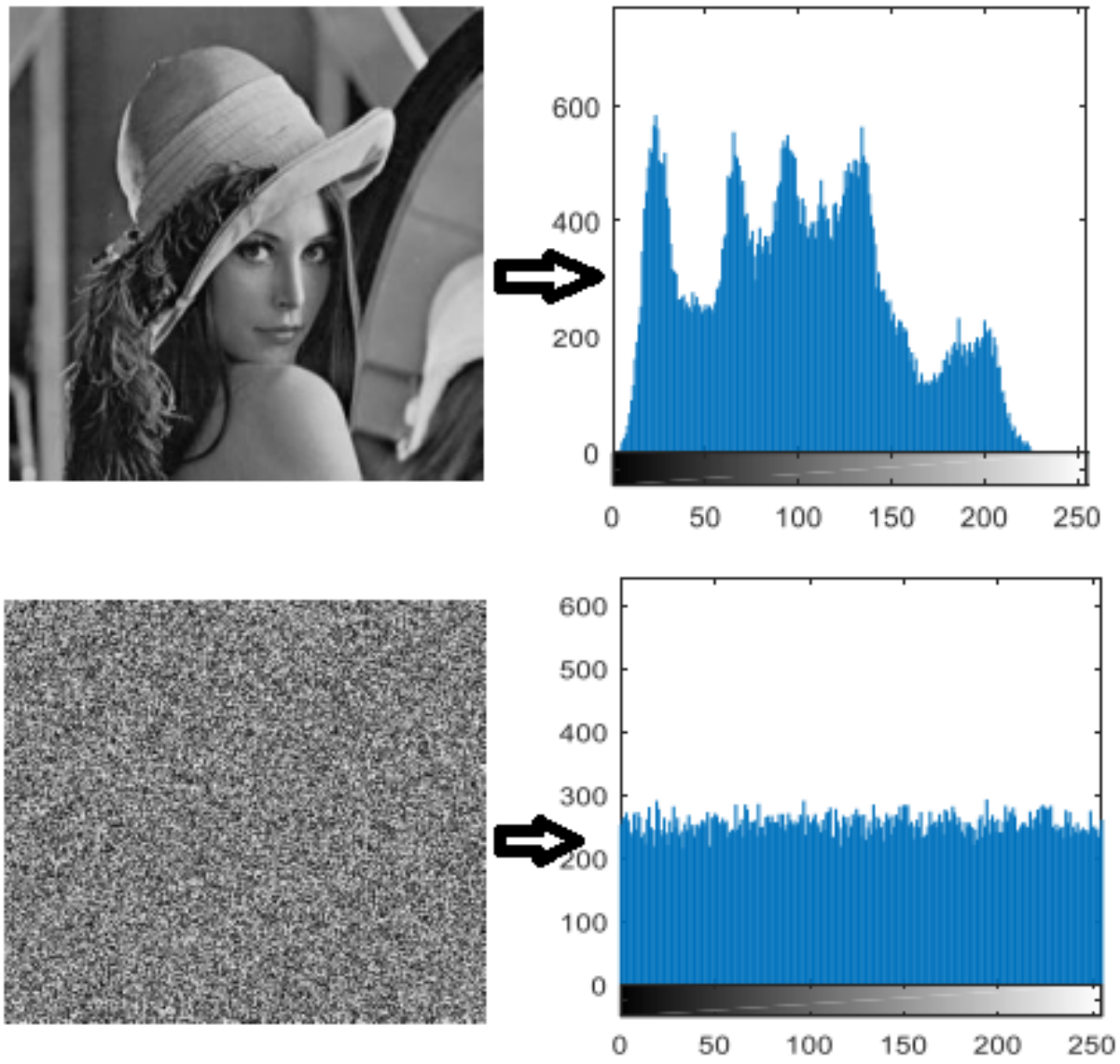


Fig. 1.8 *Les histogrammes de l'image originale de Lena et l'image cryptée correspondante de Lena cryptée .*

Où C_1 et C_2 désignent les deux images chiffrées dont le changement d'un bit correspond à la même image. D(i, j)=0 si $C_1(i, j) = C_2(i, j)$, sinon,D (i, j) = 1. Où m et nsont la hauteur et la largeur des images. La valeur idéale pour UACI est d'environ $33,4635\%$, alors que la valeur idéale pour le NPCR est d'environ $99,6094\%$ [13].

1.8.7 Test statistique de NIST

Un bon système de cryptage devrait être capable de transformer des images simples en images aléatoires. Par conséquent, il est nécessaire de tester le caractère aléatoire des images chiffrées obtenues par tout algorithme de chiffrement d'image. La suite de tests statistiques NIST SP800-22 (National Institute of Standards and Technology) comprend 15 statistiques[27].

Chaque test calcule une P_Value et le compare à un niveau de signification donné pour déterminer si la séquence est aléatoire. Lors de l'application de la suite de tests NIST, un niveau de signification α = 0,01 est choisi pour les tests. Si toute la valeur $P\_Value > \alpha$, la séquence est considérée comme aléatoire[8]. Pour chaque test, si P_Value est égal à 1, cela indique que la séquence testée est parfaitement aléatoire, alors qu'une $P\_Value = 0$ indique un caractère non aléatoire[27].

On présente par la suite les 15 tests du NIST[14, 13, 28, 29, 30]

1.8.7.1 Test de fréquence

Ce test s'intéresse à la proportion de « zéros » et de « uns » dans la séquence entière. Le but de ce test est de déterminer si le nombre de « zéros » et de « uns » dans la séquence est approximativement égal à celui prévu pour une vraie séquence aléatoire. Le test évalue la déviation de la proportion des « uns » par rapport à 1/2. Les étapes du test sont comme suit :
1) Supposons une séquence q = 11001010110011
2)Remplacer 1 par (+1) et 0 par (-1) et faire la somme S bit à bit de la séquence.
S = 1+1-1-1+1-1+1-1+1+1-1-1+1+1 = 2
3) Si S est très grand (trop de 1) ou très petit (trop de 0) alors la séquence est considérée comme non aléatoire.

1.8.7.2 Test de fréquence par bloc (Block frequency)

Ce test reprend le même principe du test de fréquence, en étudiant la proportion de « uns » et de « zéros » des sous-séquences de bits issues da la suite binaire à tester. A ce propos, la suite binaire sera divisée en blocs de bits, puis on vérifie si la fréquence des uns est approximativement 1/2 dans chaque bloc. Il est recommandé que chaque sous-séquence ait une longueur minimale égale à 100 bits. D ?où nous avons considéré des blocs de 128 bits pour le test du générateur de nombres pseudo-aléatoires proposé.

1.8.7.3 Test de la somme cumulée (Cumulative sums)

Le but de ce test est de déterminer si la somme cumulée des bits consécutifs de la séquence analysée ajustée à (-1, +1) est très grande ou très petite, afin de détecter la présence de nombre important de zéros ou de uns. En effet, la plus grande somme partielle (en valeur absolue) ne doit pas être trop grande, car elle indiquerait une mauvaise répartition des zéros et des uns. Mais elle ne doit pas non plus être trop faible, car au contraire cela indiquerait que les zéros et les uns sont trop bien mélangés. Deux approches sont possibles pour appliquer ce test. La première notée mode 0 consiste à parcourir la séquence à partir du premier bit. Tandis que la seconde notée mode 1 consiste à parcourir la séquence en sens inverse en partant du dernier bit.

1.8.7.4 Test des suites homogènes (Runs)

Le focus de ce test est le nombre total de runs dans la séquence ; où un run est une suite ininterrompue de bits identiques. Un run de longueur k consiste exactement à k bits identiques et est bornée avant et après avec un bit opposée. Le but du test des runs consiste à déterminer si le nombre de runs de uns et de zéros de différentes longueurs est aussi comme on l'a prévu pour une suite aléatoire. En particulier, ce test détermine si l'oscillation entre les zéros et les uns est trop rapide ou trop lent.

1.8.7.5 Test de longues séries de 1(Long Runs of Ones Test)

Le focus du test est la plus longue série de uns au sein des blocs de M bits. Le but de ce test est de déterminer si la longueur de la plus longue série de uns dans la séquence testée est compatible avec la longueur de la plus longue série de uns attendu dans une suite aléatoire. Notez que l ?irrégularité dans la longueur prévue de la plus longue série de uns implique qu ?il ya aussi une irrégularité dans la longueur prévue de la plus longue série de zéros. Par conséquent, seulement un test pour les uns est nécessaire.

1.8.7.6 Test de rang de la matrice binaire (Binary Matrix Rank)

L'objectif du test est de calculer le rang des sous-matrices disjointes formant la séquence toute entière. Les $sous - matrices$ sont de taille MQ avec M et Q égaux à 32 bits. Le but de ce test est de contrôler la dépendance linéaire entres des $sous - suites$ de longueur k de la séquence originale.

1.8.7.7 Test sur la transformée de Fourier discrète (Discrete Fourier Transform (Spectral) Test (DFT))

Ce test tient compte des hauteurs des pics de la transformée de Fourier de la séquence pour détecter une périodicité. L'intention est de détecter si le nombre de pics dépassant le seuil de 95% est largement différent de5%

1.8.7.8 Recherche d'un motif apériodique (Aperiodic Templates Test)

Ce test porte sur le nombre d'occurrence d'une sous-suite donnée à l'intérieur de la séquence. Le but de ce test est de rejeter les séquences présentant un trop grand nombre d'occurrences d'un motif apériodique. La recherche se fait en utilisant une fenêtre de m bits glissant sur la séquence à la recherche d'un motif de taille m. Lorsque ce motif a été trouvé, la fenêtre de recherche est replacée sur le premier bit suivant le motif découvert.

1.8.7.9 Recherche d'un motif périodique (Overlapping template Matching)

Le principe de ce test est identique au test de recherche d'un motif apériodique, qui s'agit de compter le nombre d'occurrence d'un motif particulier dans la séquence étudiée. Cependant, lorsque ce motif est trouvé dans la suite, la fenêtre de recherche ne sera pas déplacée à la fin de celui-ci, mais continue à traverser la suite normalement bit par bit. Ainsi, Le test rejettera les séquences qui ont un très grand nombre d'occurrence d'un motif.

1.8.7.10 Test universel de Maurer (Universal)

Le but de ce test est de déterminer si la séquence est compressible ou non sans perte d'information. Une séquence nettement compressible est considérée comme non aléatoire.

1.8.7.11 Test de la complexité linéaire (Linear Complexity Test)

Ce test détermine la longueur d'un registre à décalage (type LFSR) produisant la séquence analysée. Il permet de déterminer si la séquence est suffisamment complexe pour être considérée comme aléatoire. Si le registre est trop court, la suite n'est pas aléatoire.

1.8.7.12 Test de l'entropie approximative (Approximate entropy)

Le principe général de ce test est similaire au test série. Il s'agit de comparer les fréquences d'occurrence de toutes les sous-séquences possibles dans deux blocs superposés, de longueur consécutive M et M+1 , avec celles rencontrées dans une suite aléatoire.

1.8.7.13 Test d'excursions aléatoires (Random Excursion Test)

Ce test s'intéresse au nombre de cycles visités exactement K fois lors d'une marche aléatoire. La marche aléatoire est trouvée en effectuant la somme cumulée des séquences de (0, 1) ramenés à (-1, +1). Un cycle (ou excursion aléatoire) consiste en une séquence de n pas, de longueur unité pris au hasard, commençant et finissant à l'origine. Le but de ce test est de déterminer si le nombre de visites d'un état lors d'un parcours aléatoire dépasse celui attendu pour une séquence aléatoire.

1.8.7.14 Variante du test d'excursions aléatoires (Random Excursion Variant Test)

Ce test porte sur le nombre de fois où un même état est rencontré lors d'une marche aléatoire. Le but est de détecter les écarts par rapport au nombre d'occurrence normal des différents états lors d'une marche aléatoire.

1.8.7.15 Test série (Serial)

Ce test concerne la fréquence d'occurrence de chaque sous-séquence de M bits tout au long de la séquence entière. L'objectif de ce test est de déterminer si toutes les sous-séquences de M bits formant la suite à tester ont la même chance d'apparence, comme c'est le cas pour une vraie séquence aléatoire.

En résumé, la séquence qui passe le test avec succès doit être uniforme, de sorte que les nombres d'occurrence de tous les modèles de M bits soient identiques. De ce fait, pour M=1 , le test de série est équivalent au test de fréquence.

1.8.8 Le temps de traitement

L'évaluation de la vitesse d'un cryptosystème est un facteur important qui influe sur le coût de l'algorithme de chiffrement d'image proposé. Le temps de traitement est le temps nécessaire pour chiffrer et déchiffrer une image. Plus le temps de traitement est faible, meilleure est l'efficacité du cryptage[11].

1.9 Conclusion

Dans ce chapitre, nous avons discuté des différentes techniques de la cryptographie et nous nous sommes concentrés également sur les deux principales familles de cryptosystèmes à savoir les algorithmes symétriques et asymétriques.

Chapitre 2

Évaluation des performances de certaines cartes chaotiques comme bases de chiffrement basés sur le chaos proposé

Sommaire

2.1 Introduction

Les applications de sécurité de l'information nécessitent des séquences de nombres aléatoires pour la génération de clés secrètes dynamiques en cryptage, des algorithmes de stéganographie, etc. Parmi ces générateurs : les générateurs de nombres pseudo-aléatoires. Un PRNG est un algorithme déterministe qui produit des nombres dont la distribution est uniforme, en entrant une valeur initiale. Les PRNG sont importants dans les applications réelles pour leur rapidité dans la génération de nombreset la reproductibilité des séquences pseudo-aléatoires. L'introduction du chaos dans les systèmes cryptographiques est principalement due aux bonnes fonctionnalités qu'il offre pour la sécurité et pour les communications numériques, telles que : sa sensibilité extrêmement élevée aux conditions initiales et aux paramètres de contrôle, la non-linéarité, l'ergodicité et les comportements aléatoires. Un choix approprié de ces cartes chaotiques est fait en utilisant le diagramme de bifurcation, le test de Nist et l'exposant de Lyapunov. Dans ce chapitre, nous passons en revue les performances de sécurité de certaines cartes chaotiques célèbres, notamment : les cartes Logistic, Skew Tent et PWLCM, comme base des générateurs proposés basés sur le chaos. Nous passons aussi en revue certains travaux connexes dans la littérature qui sont associé à notre thèse.

2.2 Outils d'évaluation des performances de sécurité communes et standard des cartes chaotiques

Afin de quantifier les propriétés cryptographiques des séquences pseudo-chaotiques générées, plusieurs mesures de sécurité statistique et outils d'évaluation doivent être effectués. Ces tests de sécurité vérifient le degré de caractère aléatoire des séquences produites. Ces tests de sécurité incluent l'espace de phase, l'histogramme, l'auto-corrélation croisée et le test NIST [10]. Par ailleurs, les tests de l'exposant de Lyapunov et du diagramme de bifurcation, permettent de fixer les valeurs optimales des paramètres pour atteindre les régions chaotiques[1].

2.2.1 Analyse d'histogramme

L'histogramme est une représentation graphique du test de distribution des données numériques. Où une distribution uniforme de séquences aléatoires ou de variables pseudo-aléatoires indique que la séquence présente un caractère aléatoire robuste et des performances de sécurité favorables. En revanche, si la distribution des séquences aléatoires est inégale et que le nombre d'occurrences de valeurs différentes montre des différences significatives, l'attaque statistique peut alors avoir un bon effet d'attaque. Une distribution inégale peut également indiquer que le caractère aléatoire de la séquence est faible et que les performances de sécurité sont médiocres[31].

2.2.2 Auto et corrélation croisée

Une des bonnes propriétés desgénérateurs de nombres pseudo-chaotiques (PCNG) est que les séquences générées doivent être non corrélées. Ainsi, la corrélation croisée de deux séquences x et y (générées avec des clés légèrement différentes) doit être proche de zéro[12].

2.2.3 Diagramme de bifurcation

Dans le système dynamique, le diagramme de bifurcation met en évidence l'évolution qualitative du système en tant que fonction mathématique en termes de valeurs de paramètres de contrôle[32]. Le diagramme de bifurcation montre généralement les zones de bifurcation, la convergence et le chaos en termes de valeurs des paramètres de contrôle[33].

2.2.4 Exposant de Lyapunov

L'exposant de Lyapunov représente une valeur pour l'évaluation quantitative de la performance chaotique. En d'autres termes, si l'exposant de Lyapunov a une valeur positive, cela signifie que la carte non linéaire présente de meilleurs comportements chaotiques avec de meilleurs résultats dans la caractéristique de caractère aléatoire, elle est sensible aux conditions initiales[34], et en plus l'exposant de Lyapunov valorise les meilleures performances chaotiques [35]. L'équation de l'exposant de Lyapunov est mathématiquement définie comme suit :

$$Ly = \lim_{n \to +\infty} \frac{1}{n} \sum_{i=1}^{N} \ln |(f(x_i)')|, \tag{2.1}$$

Dans cette équation, f' est la fonction dérivée de la fonctionmathématique f, où f est une carte chaotique.

2.2.5 Test NIST

Il est nécessaire de tester le générateur de nombres pseudo-aléatoires chaotiques (CPRNG) dans le cas où il convient aux cryptosystèmes. Par conséquent, le test statistique de l'Institut National des Normes et de la Technologie a été effectué (National Institute of Standards and Technology NIST). La suite de tests NIST SP800-22 fournit 15 tests statistiques, chaque test calcule une valeur $P \in [0-1]$, si $P > 0,01$, cela prouve que la séquence testée est assez aléatoire et les PCNG sont très adapté aux cryptosystèmes[34].

2.3 Evaluation des performances de certaines cartes chaotiques

Les cartes (suites) chaotiques sont des systèmes dynamiques définis en temps réel par des relations de récurrence(voir figure 2.1) comme suit[10] :

$$x_i(n) = f(x_1(n-1), x_2(n-1), x_3(n-1), ..., x_m(n-1)), i = 1....m. \tag{2.2}$$

Où $x \in S$, $f : S \mapsto$ S est une fonction à m variables, $S \in [0, 1] ou [-1, 1]$. Dans cette section, nous discutons des performances de certaines cartes chaotiques unidimensionnelles telles que la carte logistique, les cartes chaotiques linéaires par morceaux (PWLCM) et la tente oblique, et qui ont été étudiées et améliorées dans la littérature et largement utilisées pour la conception des générateurs de nombres aléatoires dans les systèmes cryptographiques basés sur le chaos.

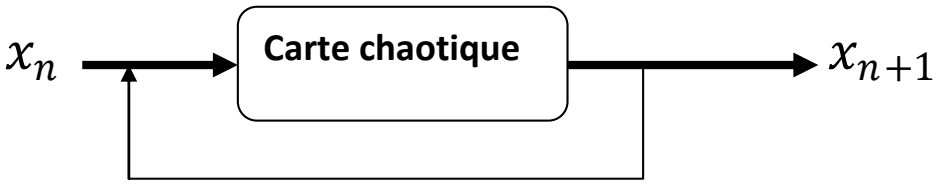


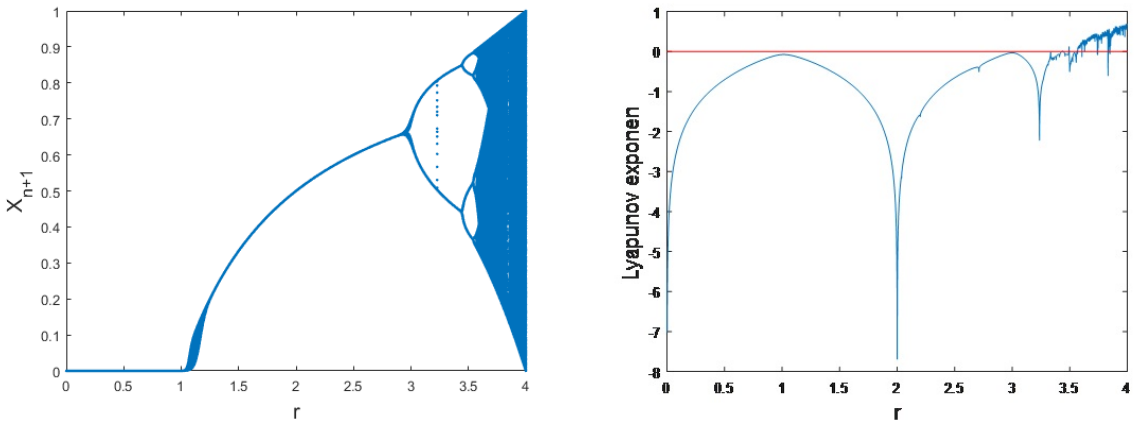
Fig. 2.1 *Schéma de génération d'une séquence pseudo-aléatoire par une carte chaotique.*

2.3.1 Évaluation des performances de la carte logistique

La carte logistique est l'une des célèbres cartes chaotiques 1D. Il s'agit d'une simple équation de récurrence dynamique avec un comportement chaotique complexe. Elle a été créée dans un premier temps par Pierre François Verhulst[36], c'est l'une des cartes les plus utilisées comme générateur pseudo aléatoire dans les applications cryptographiques. La définition mathématique peut être exprimée dans l'équation suivante[37] :

$$X_{n+1} = L(r, X_n) = rX_n(1 - X_n) \tag{2.3}$$

Où (r) est le paramètre de contrôle, X_0 est la valeur initial et $r \in [0, 4]$. Pour montrer son comportement chaotique, son diagramme de bifurcation et l'exposant de Lyapunov sont présentés sur les figures2.2. (a) et 2.2.(b). Comme le montre la figure 2.2.(a), sa plage chaotique n'est limitée que dans [3.57, 4] et le paramètre de contrôle r au-delà de la plage ne peut pas avoir de



(a) **(b)**

Fig. 2.2 *Évaluation des performances de la carte logistique (a) Diagramme de bifurcation (b) Exposant de Lyapunov.*

comportements chaotiques. Cela peut être vérifié dans le diagramme de lexposant de Lyapunov sur la Fig.2.2.(b), où le paramètre de contrôle$r < 3,57$ pour la carte logistique, le diagramme les exposant de Lyapunov de cette carte sont négatifs, cela implique qu'ils n'ont pas de comportements chaotiques. En ce qui concerne l'histogramme de la séquence générée X_L (n) celle montrée sur la figure2.3. Visuellement, la séquence générée X_L (n) n'est pas uniforme sur toutes les valeurs. La corrélation entre deux séquences $X_L et X_{L0}$ générées à l'aide de clés légèrement différentes a été calculée. La valeur obtenue est égale à $-0,0019$. De plus, l'auto-corrélation entre les deux séquences $X_L et X_{L0}$, est montrée sur la figure. Les résultats obtenus montrent que les différentes séquences générées $X_L et X_{L0}$ ont de bonnes propriétés d'auto et de corrélation croisée. Cela affirme le pseudo-aléatoire des séquences générées.

2.3.1.1 Analyse de test NIST

Le tableau 2.1 présente les résultats obtenus du test NIST. Les résultats obtenus montrent que les séquences ne passent pas tous les tests NIST. Cela montre que la carte logistique n'a pas de bonnes propriétés statistiques cryptographiques pour toutes les valeurs.

2.3.2 Évaluation des performances de la carte PWLCM et la carte Skew Tent

Le système de carte chaotique linéaire par morceaux (PWLCM) est défini par :

$$x_{(i+1)} = F_p(x_i) = \begin{cases} \frac{x_i}{p} & 0 \leq x_i < P \\ \frac{x_i - p}{0.5 - p} & P \leq x_i < 0.5 \\ F_p(1 - x_i) & 0.5 \leq x_i < 1 \end{cases} \tag{2.4}$$

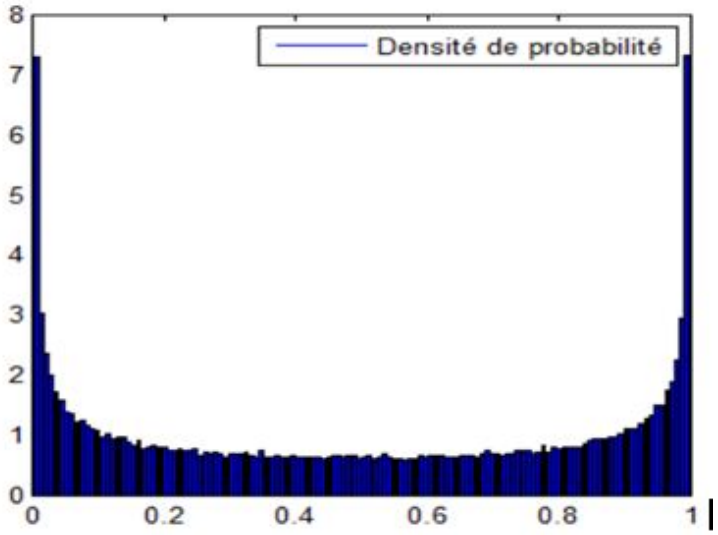


Fig. 2.3 *Histogramme de la séquence X_L générée par la carte logistique.*

Où $x_i \in (0,1)$ et le paramètre de commande $p \in (0,0,5)$ [38]. Le système PWLCM a une distribution invariante uniforme et une excellente ergodicité, confusion et détermination, de sorte qu'il peut générer une séquence aléatoire, qui convient au cryptosystème. Eq. (2.4) peut générer une séquence chaotique avec une valeur initiale donnée x_0 et un paramètre de contrôle p.

La carte de la tente oblique (la carte Skew Tent) est une simple équation dynamique non linéaire avec un comportement chaotique complexe, est l'une des célèbres cartes chaotiques, exprimée par l'équation suivante [12] :

$$x_{(i+1)} = F_s(x_i) = \begin{cases} \frac{x_i}{p} \; 0 \leq x_i < P \\ \frac{1-x_i}{1-p} \; P \leq x_i < 1 \end{cases} \tag{2.5}$$

Où $x_{(i+1)} \in [0,1]$ est l'état du système chaotique, $P \in [0,0,5] \cup [0,5,1]$ est le paramètre de contrôle et $x_i \in [0,1]$ avec n est le nombre d'itérations utilisé pour générer les valeurs itératives. Il ressort clairement de la fig.2.5 que l'histogramme d'une séquence X générée par STM a présente une densité de distribution uniforme. qui est visuellement plus uniforme que l'histogramme d'une séquenceX_L (n) générée par la carte logistique.

2.3.2.1 Analyse de corrélation

La figure 2.4 montre que la carte de la tente Skew a de bonnes propriétés auto/ inter-corrélation et nous confirmons ce résultat par la valeur du coefficient de corrélation de deux séquences X_S et X_{S0} généré avec une initiale proche, qui est très faible. Pour montrer le comportement chaotique du PWLCM et du Skew-Tent, leurs diagrammes de bifurcation et

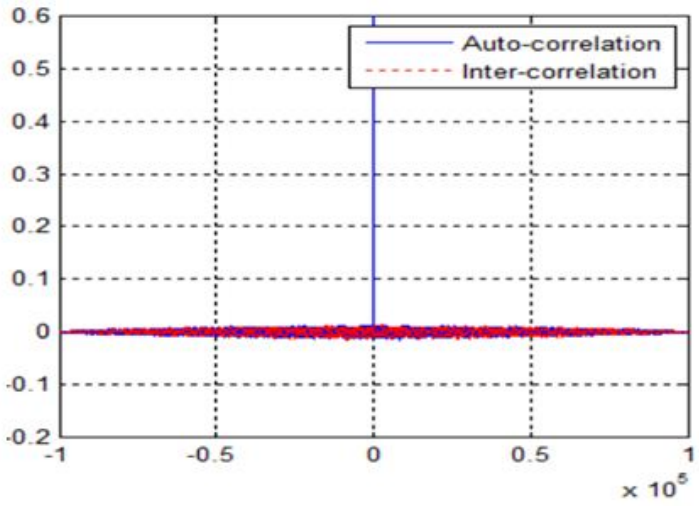


Fig. 2.4 *Auto/ inter-corrélation de la séquence X_L générée par la carte $Skew - Tent$*

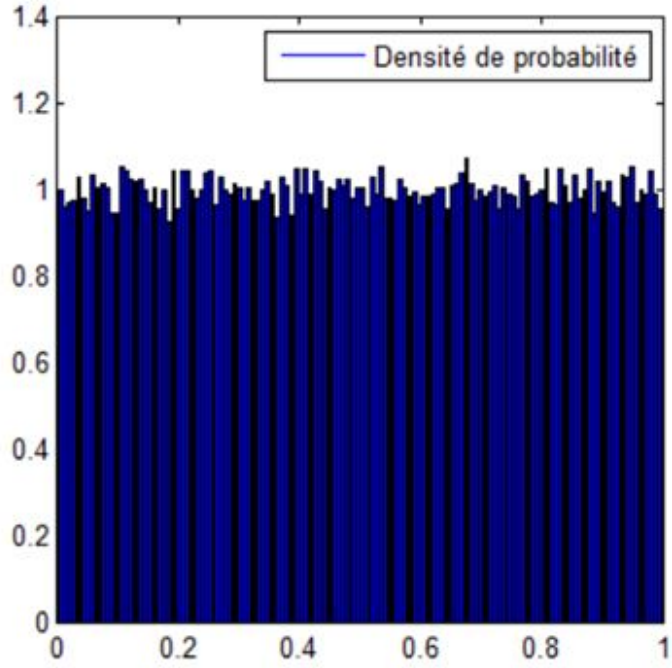


Fig. 2.5 *Histogramme de la séquence X_S générée par la carte discrète Skew Tent.*

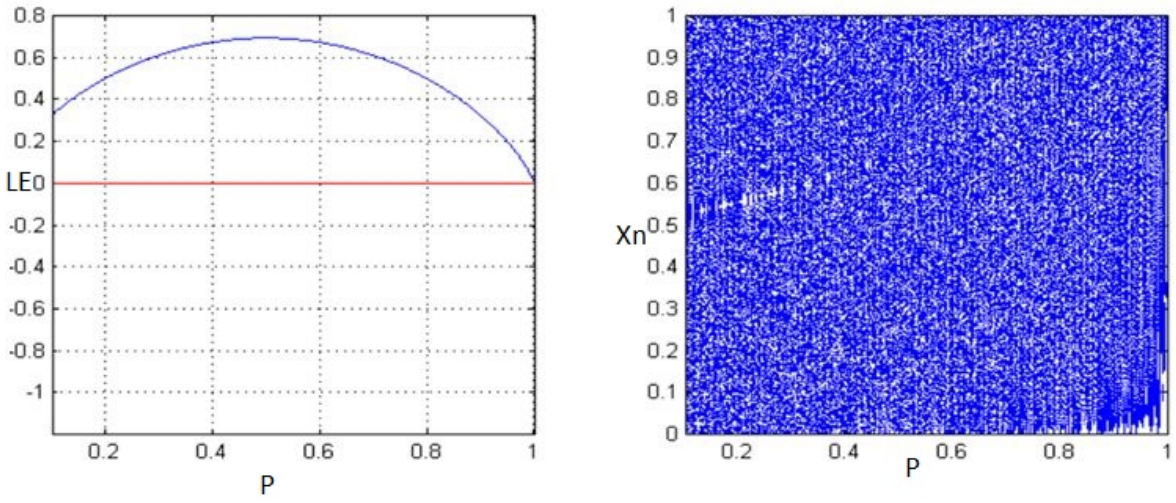


Fig. 2.6 *Diagrame de Lyapunov et diagramme de bifurcation de la récurrence Skew-Tent.*

l'exposant de Lyapunov sont présentés sur les figures 2.6.(a) et 2.6.(b). Comme le montre la figure 2.6.(a), la plage chaotique de Skew-Tent dans [0,1] et PWLCM avec [0, 0,5], pour Skew-Tent et PWLCM, l'exposant Lyapunov reste positif pour toutes les valeurs de leurs paramètres. Cela signifie que le comportement des deux systèmes est toujours chaotique,et ont ainsi été utilisés dans le chiffrement des données en raison de leur bonnes caractéristiques d'aléatoire contrôlable par les paramètres[39].

2.3.2.2 Analyse de test NIST

Le tableau 2.1 présente les résultats des tests NIST obtenus d'une séquence X_P et X générée respectivement par la carte PWLCM et Skew-Tent. Certains sous-tests n'ont pas réussi mais ils sont proches des valeurs acceptées. En outre, nous remarquons que la carte PWLCM a de meilleurs randomens que les deux autres cartes chaotiques étudiées Skew Tent et Logistic map.

2.4 Principe du crypto-système basé sur le chaos

La majorité des cryptosystèmes basés sur le chaos qui utilisaient des nombres pseudo-aléatoires générés par des fonctions chaotiques (ou générateurs) ont été conçus en utilisant la structure de confusion et de diffusion. La confusion signifie dans quelle mesure un changement de la clé secrète affecte le message chiffré et la diffusion signifie comment un changement d'un seul bit du texte en clair affecte les bits chiffrés. Une couche générale, une couche de confusion suivie d'une couche de diffusion qui fonctionnent séparément. Le processus de confusion est appliqué rc fois, puis le processus de diffusion est appliqué rd fois sur la sortie du processus de confusion, et enfin, les deux processus sont répétés r fois. Les deux couches ont nécessité (pour rc = rd = r = 1). comme le montre la figure 2.7. La séquence de nombres pseudo-aléatoires générée

TABLE 2.1 *Résultats des P-value du test NIST pour la carte PWLCM, Skew Tent et carte logistique[1]*

NIST tests	carte logistique	Skew Tent	PWLCM
Frequency	0.000	0.262	0.994
Block Frequency	0.000	0.000	0.456
The Run Test	0.000	0.198	0.856
Longest Run of Ones in a block	0.000	0.081	0.924
Binary Matrix Rank	0.000	0.575	0.720
DFT Spectral	0.419	0.000	0.616
Non-Overlapping Template Matching	0.000	0.000	0.040
Overlapping Template Matching	0.036	0.531	0.482
Maurer's Universal Statistical Test	0.000	0.760	0.964
Linear Complexity	0.000	0.000	0.868
Serial Test	0.000	0.575	0.868
Approximate Entropy	0.002	0.468	0.266
Cumulative Sums	0.400	0.369	0.308
Random Excursions	0.000	0.402	0.269
Random Excursions Variant	0.081	0.103	0.046

est utilisée par l'algorithme de cryptage pour crypter le message en clair chez l'expéditeur. Au niveau du récepteur, les mêmes cartes chaotiques avec les mêmes clés secrètes sont utilisées pour générer la même séquence de nombres pseudo-aléatoires. Cette séquence sera utilisée par un algorithme de décryptage afin de récupérer le message clair comme le montre la figure 2.8

2.5 Cryptosystèmes de chiffrement basés sur le chaos

Dans les systèmes informatiques d'aujourd'hui, il est très important de protéger les données échangées dont le contenu d'image est une partie écrasante. Pour cette raison, le cryptage d'images est devenu un défi urgent et une grande préoccupation qui a attiré de nombreux chercheurs ces dernières années. Plusieurs schémas de cryptage d'images ont été développés en utilisant diverses techniques telles que les cryptosystèmes basés sur le chaos.

2.5.1 Un générateur de nombres pseudo-aléatoires basé sur une nouvelle carte chaotique 3D avec une application pour le cryptage d'images couleur

Mohamed Lamine Sahari et Ibtissem Boukemara[40], ont proposé une nouvelle carte chaotique 3D obtenue en couplant les cartes par morceaux et les cartes logistiques ont d'excellentes propriétés, comme un fort caractère aléatoire, une grande complexité et une très longue période.

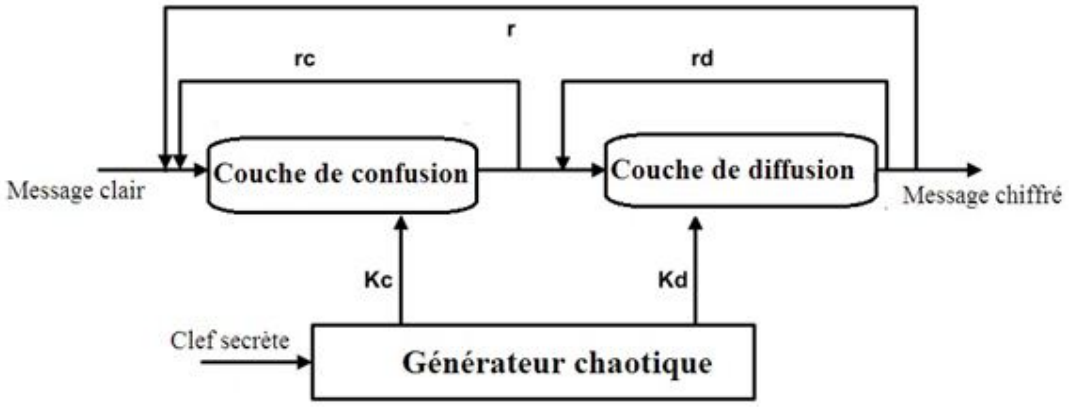


Fig. 2.7 *Structure générale de l'algorithme de chiffrement basé sur le chaos.*

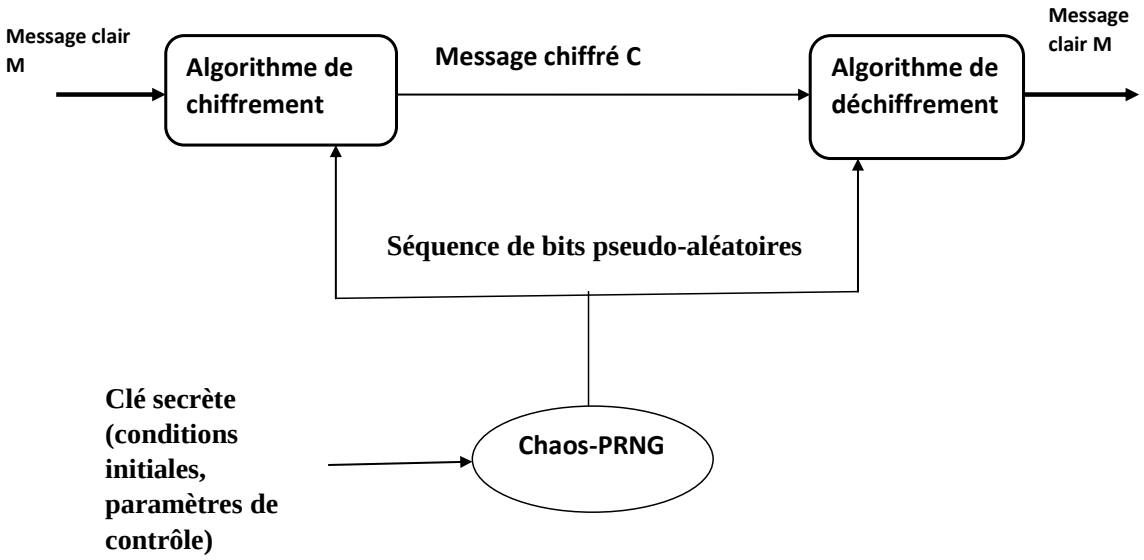


Fig. 2.8 *Schéma de principe d'un crypto-système basé chaos*

Cette carte leur a permis de mettre en œuvre et d'étudier un nouveau générateur de nombres pseudo-aléatoires chaotiques (CPRNG). Les nombres pseudo-aléatoires produits présentent une distribution uniforme et réussissent la suite de tests de hasard NIST SP 800-22. De plus, ils ont proposé une application dans le domaine du chiffrement d'images couleur proposée où la clé de chiffrement est fortement corrélée à l'image simple et est ensuite utilisée pour effectuer les étapes de confusion et de diffusion. De plus, la possibilité d'étendre la taille de la carte a un impact sur la complexité du système et augmente la taille de l'espace clé, rendant leurs systèmes de cryptage plus efficaces et plus sûrs.

2.5.2 Un nouveau cryptage d'image couleur utilisant la combinaison de la carte chaotique 1D

Chanil Pak et Lilian Huang[41] ont introduit une méthode pour créer un système chaotique simple et efficace en utilisant une différence des séquences de sortie de deux mêmes cartes chaotiques à une dimension (1D) existantes. Les simulations et les évaluations des performances montrent que le système proposé est capable de produire un système chaotique à une dimension (1D) avec de meilleures performances chaotiques et de plus grandes plages chaotiques par rapport aux cartes chaotiques précédentes. Pour étudier ses applications dans le chiffrement d'images, un nouveau système de chiffrement de structure linéaire-non-linéaire-linéaire basé sur le brassage total est proposé. L'expérience a démontré la précision de l'algorithme de chiffrement. Les expériences et l'analyse de sécurité prouvent que l'algorithme a une excellente performance dans le cryptage d'image et diverses attaques.

2.5.3 Schéma de cryptage d'image couleur rapide basé sur le chaos avec de vraies clés à nombres aléatoires provenant du bruit de l'environnement

Dans [42]Hongjun Liu et al ont proposé un schéma de cryptage d'image couleur basé sur le chaos, le point culminant est que le signal de bruit échantillonné au hasard est appliqué pour servir de valeurs initiales d'un système chaotique. La valeur de hachage de 256 bits du bruit est transformée en valeurs initiales uniques du système Liu. Un OU exclusif, la seule opération, est appliqué pour diffuser les pixels, et certaines mesures ont été prises pour accélérer le processus de cryptage. Ils ont effectué des tests statistiques pour évaluer la fiabilité et l'efficacité du cryptosystème proposé en termes de complexité temporelle et de sécurité. Les résultats expérimentaux démontrent l'efficacité du schéma proposé.

2.5.4 Carte de modulation logistique sinusoïdale 2D pour le cryptage des images

Zhongyun Hua et YicongZhou[43] ont introduit une nouvelle carte de modulation logistique sinusoïdale bidimensionnelle (2D-SLMM) dérivée des cartes logistique et sinusoïdale. Comparé aux cartes chaotiques existantes, il a une gamme chaotique plus large, une meilleure ergodicité, une propriété hyperchaotique et un coût de mise en œuvre relativement faible. Pour étudier les applications du 2D-SLMM, ils ont proposé une transformation magique chaotique (CMT) pour changer efficacement les positions des pixels de l'image. Combinant 2D-SLMM avec CMT, ils ont en outre introduit un nouvel algorithme de cryptage d'image. Les résultats de la simulation et l'analyse de sécurité démontrent que l'algorithme proposé est capable de protéger des images avec une faible complexité temporelle et un niveau de sécurité élevé ainsi que de résister à diverses attaques.

2.6 Conclucion

Dans ce chapitre, nous avons d'abord présenté des outils communs et standard pour mesurer les performances des générateurs chaotiques, afin de quantifier et de comparer les propriétés des séquences chaotiques générées des algorithmes de chiffrement basés sur le chaos. Deuxièmement, un aperçu du générateur de chaos existant célèbre et classique a été présenté. Enfin, un aperçu de certains générateurs basés sur le chaos et des schémas de chiffrement existants dans la littérature récente a été présenté.

Chapitre 3

Schéma de cryptage des images couleur basé sur une carte chaotique quadratique améliorée

Sommaire

3.1 Introduction

Dans le chiffrement d'images chaotique, la sécurité des schémas de cryptage dépend des caractéristiques des cartes chaotiques et de la structure du schéma, il est donc nécessaire d'avoir une meilleure distribution des cartes chaotiques. Cependant, les systèmes chaotiques traditionnels tels que les cartes quadratiques et logistiques présentent certaines lacunes telles que la distribution de données non uniforme et une gamme limitée de conduites chaotiques de flux pseudo-aléatoires générés [44][45], les cartes chaotiques qui ont une faible comportement chaotique peuvent rendre les cryptosystèmes faciles à craquer. Récemment, plusieurs chercheurs ont suggéré des cartes chaotiques améliorées pour surmonter leurs lacunes et ainsi obtenir des caractéristiques améliorées de la distribution chaotique [46, 31, 40, 47]. L'objectif premier de cette recherche est d'analyser les lacunes de la carte chaotique quadratique, puis de proposer une carte quadratique améliorée (EQM) pour éviter ces lacunes. Sur la base de cette carte chaotique améliorée, un algorithme de cryptage d'image efficace est proposé dans cette étude, en essayant de satisfaire la protection des exigences de transmission d'images numériques. Ce travail propose une amélioration du pseudo-aléatoire de la carte quadratique qui a été nommée EQM ; une modification de la carte quadratique classique en appliquant l'arithmétique modulaire. Pour démontrer l'avantage des applications d'algorithmes proposées, un nouveau schéma de cryptage d'image a été proposé ayant de bonnes caractéristiques ainsi que d'excellentes propriétés de confusion et de diffusion pour surmonter plusieurs attaques. L'algorithme est simple et très facile à mettre en œuvre pour le chiffrement et le déchiffrement d'images. Ce travail est structuré comme suit : nous commençons par présenter les cartes quadratiques classiques, ensuite les détails du système / carte chaotique amélioré (EQM) et trois exemples sont donnés. Par la suite nous présentons les détails du nouvel algorithme d'image proposé, et les résultats expérimentaux.

3.2 Carte quadratique classique

La carte quadratique est une célèbre carte chaotique avec un comportement dynamique complexe [32]. Cette carte a été largement utilisée dans les applications cryptographiques. Elle peut être exprimée comme une équation quadratique classique, elle est donnée comme suit :

$$X_{n+1} = Q(r, X_n) = r - (X_n)^2. \tag{3.1}$$

Ici, r est un paramètre de contrôle avec une plage de [0,2], n est le nombre d'itérations et X_n $\in$[0,1] est la séquence chaotique produite. Dans les prochaines sous-sections, le diagramme de bifurcation et l'exposant de Lyapunov seront étudiés pour le test de performance de la carte chaotique quadratique.

3.2.1 Diagramme de bifurcation

La figure 3. 1.a montre le diagramme de bifurcation de la carte quadratique. D'après la Fig. 3.1.a, nous pouvons voir que la carte quadratique chaotique n'a une conduite chaotique que pour r ∈[1.4,2].

3.2.2 Exposant de Lyapunov

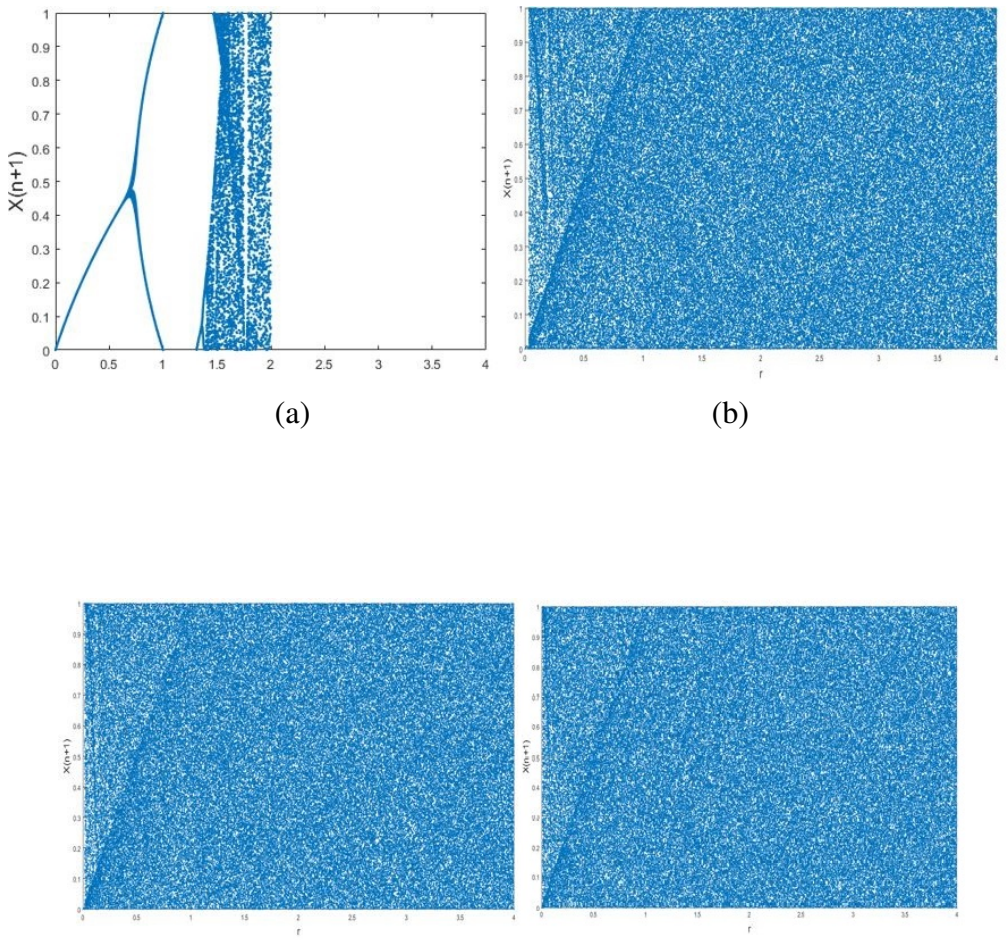
A partir de la figure 2a, on peut observer que l'exposant de Lyapunov de la carte quadratique est> 0 lorsque $r > 1,4$, et une valeur négative lorsque $r < 1,4$. Cela montre qu'ils n'ont pas de comportement chaotique dans cette gamme. Par conséquent, la valeur maximale d'exposant de Lyapunov (MLE) de la carte quadratique est de 0,9696. Les figures 3.1.a et fig.3.2.a illustrent deux problèmes dans la carte quadratique régulière. Premièrement, sa plage chaotique n'est confinée que dans [1.4, 2] qui est très petit pour être utilisé, comme l'exigence des clés d'espace dans les applications cryptographiques. Deuxièmement, comme le montre la figure 3.1a, la plage de données des séquences chaotiques a une distribution non uniforme à l'intérieur de [0, 1]. Ce problème a produit des séquences chaotiques qui pourraient affecter les performances du crypto-système principalement parce qu'un cryptosystème utilise ces séquences dans la permutation et la diffusion des données. De plus, il est facile d'observer que la plage comporte des zones vides et des valeurs des exposants de Lyapunov inférieures à zéro. Cela conduit à la carte quadratique pour produire une séquence non chaotique qui ne peut pas résister aux attaques statistiques.

3.3 Carte quadratique améliorée (EQM)

Dans cette section, une version améliorée de la carte quadratique (EQM) est proposée. La carte du chaos proposée résout les défauts mentionnés ci-dessus, ce qui la rend appropriée aux propriétés cryptographiques. Enfin, trois exemples sont présentés pour évaluer les performances de la carte du chaos proposée. La formule mathématique de l'EQM est donnée comme suit :

$$\begin{aligned} X_{n+1} = EQM(r, 2^k(X_n)) &= Q(r, 2^k(Q(r, X_n))mod\ 1 \\ &= (r - 2^k(r - X_n^2)^2)mod\ 1. \end{aligned} \tag{3.2}$$

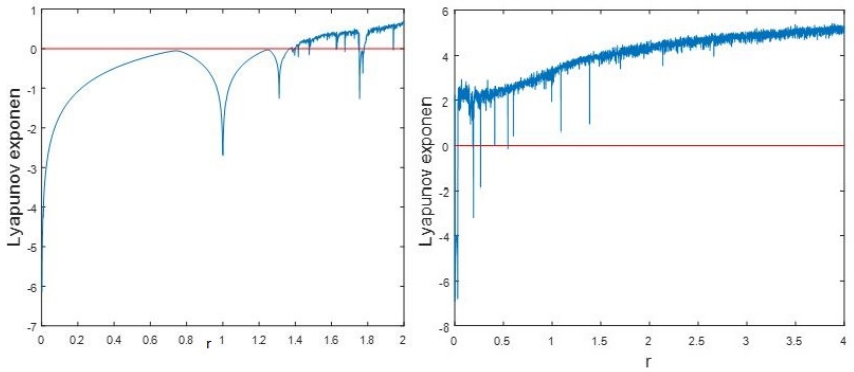
Dans cette équation, k ∈ [0,8]. Dans les sous-sections suivantes, la nouvelle version améliorée de la carte quadratique (EQM) est évaluée par l'exposant de Lyapunov et le diagramme de bifurcation, à trois valeurs différentes de k = 5, 6, 7.



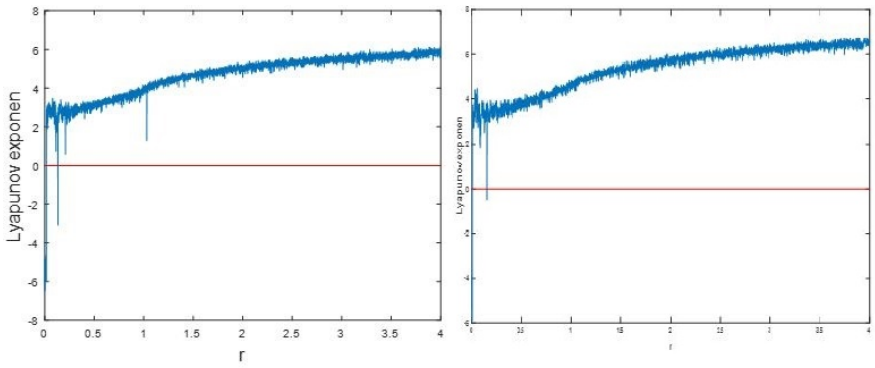
(a) (b)

(c) (d)

Fig. 3.1 *Diagramme de bifurcation de (a) Quadratique, (b) EQM1, (c) EQM2, (d) EQM3.*



(a) (b)



(c) (d)

Fig. 3.2 *Exposant de Lyapunov de (a) Quadratique, (b) EQM1, (c) EQM2, (d) EQM3*

3.3.1 Carte quadratique améliorée 1 (EQM1)

Le premier exemple de la nouvelle version améliorée de la carte quadratique à k = 5 est donné comme suit :

$$X_{n+1} = (r - 2^5(r - X_n^2)^2) mod 1. \tag{3.3}$$

Les figures 3.1.b et 3.2.b montrent l'analyse de cet exemple lié à l'EQM1. A partir de l'observation fournie par la figure 3.1.b, la carte quadratique suggérée 1 (EQM1) montre sa conduite chaotique à l'une ou l'autre valeur de r excluant la très petite plage [0, 0,05], en outre, les séquences chaotiques générées par ce système ont une distribution-uniforme dans [0 1]. De plus, la figure 3.2.b montre que l'exposant de Lyapunov de EQM1 a une valeur supérieure à zéro à toutes les valeurs pour le paramètre r excluant une petite plage [0, 0,05]. Le calcul numérique du plus grand exposant de Lyapunov pour EQM1 et la carte quadratique classique est de 5,2713 et 0,68894, respectivement. L'exposant de Lyapunov d'EQM1 est plus grand que celui des cartes quadratiques classiques. En conséquence, on peut conclure que les caractéristiques chaotiques du premier exemple de la carte quadratique sont considérablement améliorées.

3.3.2 Carte quadratique améliorée 2 (EQM2)

Le deuxième exemple des versions améliorées proposées de La carte quadratique à k = 6 s'écrit sous la forme suivante :

$$X_{n+1} = (r - 2^6(r - X_n^2)^2) mod 1. \tag{3.4}$$

Les figures 3.1.c et 3.2.c montrent l'analyse du deuxième exemple de l'EQM2. D'après l'observation donnée par la figure 3.1.c, EQM2 montre un comportement chaotique à toutes les valeurs de r excluant une petite plage dans [0, 0,03]. Les séquences chaotiques générées présentent une distribution uniforme dans [0 1]. En outre, sur la base de la courbe de la figure 3.2.c, le résultat

de l'exposant de Lyapunov a une valeur positive à toutes les valeurs de r à l'exception d'une petite plage [0, 0,025]. En conséquence, il est clair que l'exposant de Lyapunov du deuxième exemple d'EQM2 est plus grand que la carte quadratique conventionnelle et l'EQM1. En résumé, les exposants maximaux de Lyapunov (MLE) de EQM2, EQM1 et la carte quadratique classique sont respectivement 5,5807, 5,2713 et 0,68894. Par conséquent, on peut voir que les caractéristiques chaotiques du deuxième exemple de la carte quadratique sont considérablement améliorées par rapport à la carte quadratique classique.

3.3.3 Carte quadratique améliorée 3 (EQM3)

Dans cette section, le troisième exemple des versions améliorées de la carte quadratique à k = 7 est présenté sous la forme suivante :

$$X_{n+1} = (r - 2^7(r - X_n^2)^2) mod 1. \tag{3.5}$$

Le diagramme de bifurcation et l'exposant de Lyapunov d'EQM3 sont affichés sur les Fig. 3.1.d et 3.2.d, respectivement. Comme dans les exemples précédents, EQM1 et EQM2, les performances d'EQM3 et de la plage chaotique sont meilleures que la carte quadratique conventionnelle. Le tableau 3.1 répertorie les performances des classiques et des EQM. Comme illustré dans les figures précédentes, les cartes chaotiques proposées ont une gamme chaotique plus large et une amélioration du MLE. Cela démontre que l'EQM a la capacité de répondre aux exigences de tout cryptosystème utilisant tous les cas de valeurs k.

TABLE 3.1 *Comparaison entre les cartes quadratiques proposées améliorées et la carte quadratique conventionnelle.*

Carte chaotique	Plage de paramètres chaotique	MLE
Carte quadratique conventionnelle	r ∈[1.4,2]	0.6894
EQM1	r ∈[0.05,4]	5.2713
EQM2	r ∈[0.025,4]	5.5807
EQM3	r ∈[0,4]	6.5177

3.3.4 tests de Nist

Tableau 3.2, où l'on peut voir que $P > 0,01$, et les séquences réussissent les 15 tests. Ainsi, les séquences aléatoires générées par le système EQM sont très adaptées aux cryptosystèmes.

TABLE 3.2 *Les résultats des tests NIST-800-22 d'EQM3.*

NIST tests	P-value	Results
Frequency	0,514401	Passed
Block Frequency	0,078642	Passed
The Run Test	0,35008	Passed
Longest Run of Ones in a block	0,800525	Passed
Binary Matrix Rank	0,909749	Passed
DFT Spectral	0,218821	Passed
Non-Overlapping Template Matching	0,048769	Passed
Overlapping Template Matching	0,112094	Passed
Maurer's Universal Statistical Test	0,626723	Passed
Linear Complexity	0,154303	Passed
Serial Test	0,955629	Passed
Approximate Entropy	0,068696	Passed
Cumulative Sums	0,810282	Passed
Random Excursions	0,757850	Passed
Random Excursions Variant	0,831406	Passed

3.4 Cryptosystème proposé

Dans cette section, les détails de l'algorithme du cryptosystème proposé sont présentés. En particulier, dans l'opération de diffusion, la relation entre le texte chiffré et le texte en clair devient très complexe, où chaque pixel chiffré est lié non seulement au pixel clair qui le produit, mais à tous les autres pixels. Ainsi, tout petit changement aléatoire dans un pixel d'image en clair produira une image de chiffrement complètement différente, et d'autre part, les valeurs initiales et les paramètres de contrôle (les clés secrètes) d'EQM sont associés à une image en clair à chiffrer. Cela garantit la résistance aux attaques différentielles et aux attaques connues / choisies.

3.4.1 Algorithme de cryptage d'image

Dans cette partie, les étapes de l'algorithme de cryptage d'image proposé sont présentées. Le schéma de principe de l'algorithme suggéré est illustré sur la figure 3.3. Comme le montre cette figure, l'algorithme proposé repose sur les séquences générées à partir d'EQM et sur deux cycles de processus de permutation et de diffusion.

Étape (1) : L'entrée est une image rouge - vert - bleu (RVB) de taille $3 \times N \times M$ notée P. Dans la première étape, les matrices RVB sont fusionnées en une matrice de taille $3 \times N \times M$. Ensuite, la matrice obtenue est transformée en vecteur unidimensionnel $O = (o1, o2, ...o_{3 \times N \times M})$, celui-ci

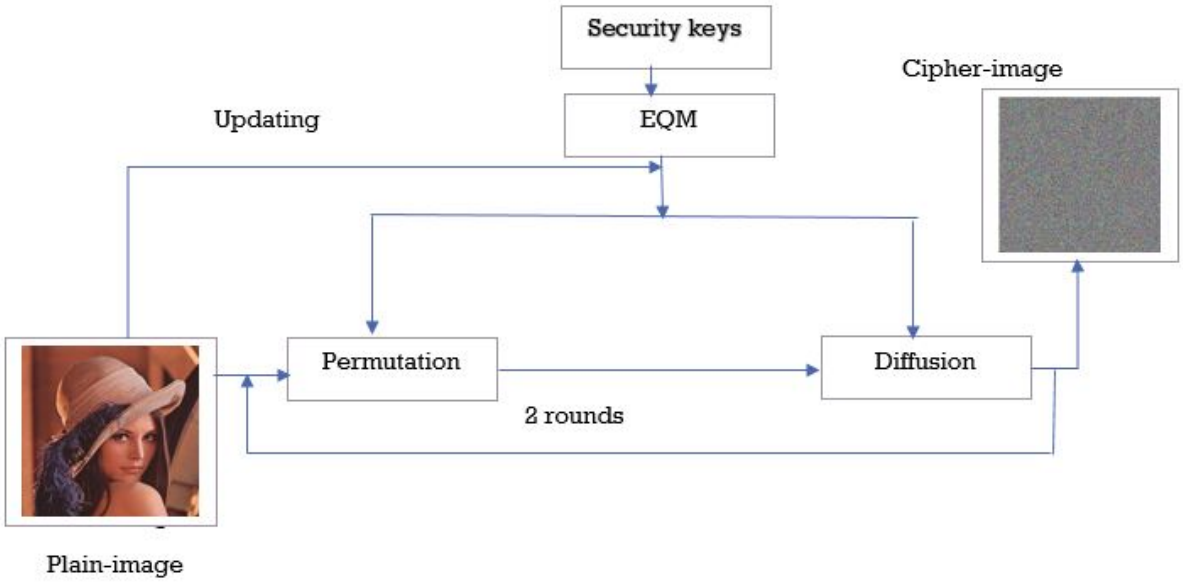


Fig. 3.3 *Le schéma fonctionnel des processus de chiffrement proposés.*

contient $3 \times N \times M$ pixels.

Étape (2) : Dans le schéma de chiffrement proposé, les paramètres (r_1, r_2, and r_3) et les valeurs initiales (x_{11}, x_{12}, and x_{13}) d'EQM3 sont considérés comme les clés secrètes. Afin de mettre à jour les paramètres et les valeurs initiales, trois valeurs liées à l'image ordinaire sont calculées par la formule suivante :

$$S_k = \sum_{i,j} I(i, j, k), k = 1, 2, 3. \tag{3.6}$$

De plus, pour produire des clés différentes à chaque fois, S_4 et S_5 sont considérés comme des nombres aléatoires dans [0, 1] et générés comme suit :

$$S_4 = rand(), S_5 = rand() \tag{3.7}$$

Les valeurs initiales et les paramètres de contrôle sont mis à jour comme suit :

$$S_k^0 = \frac{S_k}{M \times N \times 255}, k = 1, 2, 3. \tag{3.8}$$

$$\begin{cases} x'_{11} = (x_{11} + S_1^0 + S_4) mod\ 1 \\ x'_{12} = (x_{12} + S_2^0 - S_5) mod\ 1 \\ x'_{13} = (x_{13} + S_3^0 + S_5) mod\ 1 \end{cases} \tag{3.9}$$

$$\begin{cases} r'_1 = (r_1 + S_1^0 - S_4) mod\ 4 \\ r'_2 = (r_2 - S_2^0 + S_5) mod\ 4 \\ r'_3 = (r_3 + S_3^0 + S_5) mod\ 4 \end{cases} \tag{3.10}$$

Comme il est bien connu, les nouveaux paramètres r'_1, r'_2, r'_3 et les valeurs initiales $x'_{11}, x'_{12}, x'_{13}$, pour EQM sont très sensibles à tout petit changement dans les clés secrètes et l'image d'origine. Par conséquent, les différentes images en clair conduisent à différentes clés secrètes.

Étape (3) : trois séquences de chaos différentes sont générées par EQM3. En conséquence, trois valeurs initiales sont établies (x'_{11}, x'_{12}, et x'_{13}) avec trois paramètres (r'_1, r'_2, et r'_3) comme c'est le cas dans les clés secrètes de cryptage pour le cryptosystème proposé. L'algorithme 1 fait référence aux étapes de génération des clés pour le cryptosystème suggéré.

Étape (4) : basée sur les séquences générées à l'étape 3, les séquences suivantes sont extraites : X=($X_1, X_2, ..., X_{3\times N\times M}$), Y=($Y_1, Y_2, ..., Y_{N\times M}$)
et Z=($Z_1, Z_2, ..., Z_{3\times M\times N}$) D'EQM3. La séquence Z est triée par ordre croissant en fonction des indices des nombres de cette séquence dont l'index clé = ($ind_1, ind_2, ..., ind_{3\times M\times N}$) est créé en utilisant la fonction suivante :

$$[Z_s, index] = Sort(Z). \tag{3.11}$$

Où Z_s est la séquence arrangée de Z.

Étape (5) : Les positions des pixels de O sont décalées en utilisant l'indice de matrice de placement de permutation. La formule de permutation peut être exprimée comme suit :

$$Permuted(i) = O(index(i)). \tag{3.12}$$

Étape (6) : Ici, l'image en niveaux de gris P est convertie en trois composantes de couleur R, G et B avec une taille de $M \times N$. En outre, $R'_{M\times N}$, $G'_{M\times N}$ et $B'_{M\times N}$ converti en trois 1D les vecteurs $R'_{1\times L}$, $G'_{1\times L}$,$B'_{1\times L}$ où L =$M \times N$. R', G' et B' sont diffusés en utilisant (3.15) et (3.16) pour obtenir leur correspondant chiffré R", G", B", puis les composants obtenus sont remodelés pour obtenir des matrices R", G", B" de taille $R", G", B"$. trois composantes. Le processus proposé est illustré à la figure 3.5 et à l'algorithme 2 . Les séquences X et Y sont transformées en séquences entières à l'aide des fonctions (3.13) et (3.14)

$$key1(i) = floor(Y(i) \times 10^{15}) mod\ 256. \tag{3.13}$$

$$key2 = floor(X(i) \times 10^{15}) mod\ 256. \tag{3.14}$$

Où floor (X) arrondit les valeurs de X aux entiers les plus proches de X.

$$\begin{cases} R"_0(i) \leftarrow R'(1) \oplus G'(1). \\ G"_0(i) \leftarrow G'(1) \oplus B'(1). \\ B"_0(i) \leftarrow B'(1) \oplus key2(1). \end{cases} \tag{3.15}$$

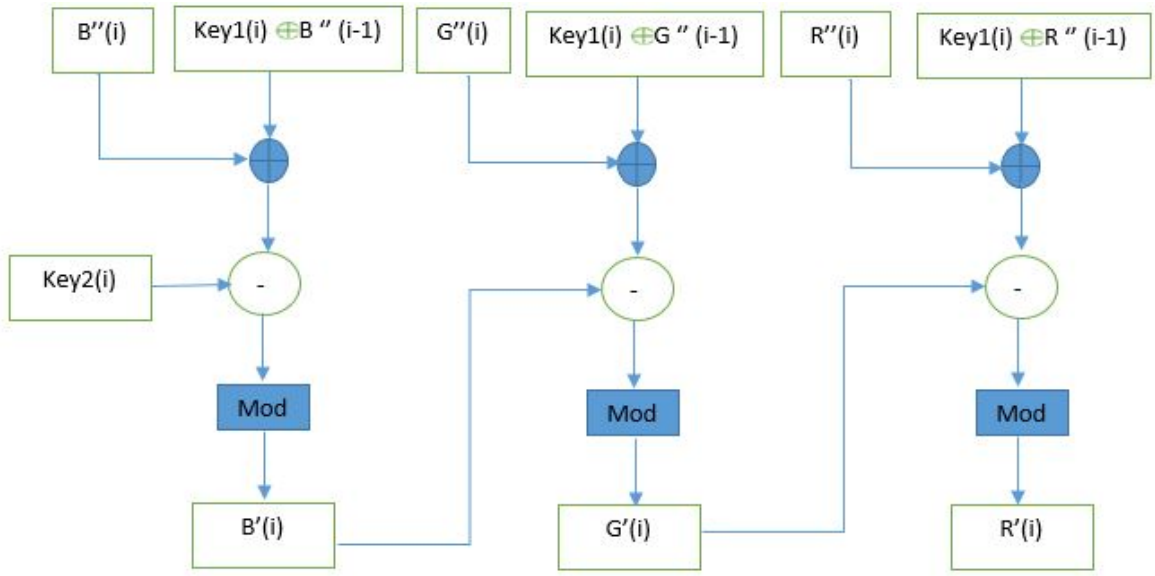


Fig. 3.4 *Diffusion process in Decryption.*

$$\begin{cases} R"(i) \leftarrow [(R'(i) + G'(i))mod256] \oplus [R"(i-1) \oplus key1(i)]. \\ G"(i) \leftarrow [(B'(i) + G'(i))mod256] \oplus [G"(i-1) \oplus key1(i)]. \\ B"(i) \leftarrow [(B'(i) + key2(i))mod256] \oplus [B"(i-1) \\ \oplus key1(i)]. \end{cases} \tag{3.16}$$

Dans ces équations,$\oplus$ fait référence à l'opérateur XOR au niveau du bit, et $R"(i-1)$,$G"(i-1)$ et $B"(i-1)$ sont les trois pixels cryptés précédents. De plus, si i = 1, $R"(i-1)$,$G"(i-1)$ et $B"(i-1)$ sont modifiés par trois valeurs $R"_0$, $G"_0$ and $B"_0$, respectivement

3.4.2 Algorithme de décryptage d'image

Dans cette partie, les étapes pour récupérer l'image originale à partir de l'image chiffrée sont présentées en utilisant les clés secrètes avec l'algorithme proposé dans cette section. Comme on le sait, le déchiffrement est la procédure inverse du chiffrement. Ainsi, toutes les étapes de cryptage sont inversibles, c'est juste pour que le lecteur le comprenne clairement, les étapes inverses de l'étape (5) et de l'étape (6) sont illustrées. Le processus est illustré sur la figure 3.3, pour le décryptage, les équations suivantes sont utilisées :

$$\begin{cases} B'(1) \leftarrow B"(1) \oplus key2(1). \\ G'(1) \leftarrow G"(1) \oplus B'(1). \\ R'(1) \leftarrow R"(1) \oplus G'(1). \end{cases} \tag{3.17}$$

Algorithm 1 La génération des clés par EQM3

Input : $(x_1, r_1), (x_2, r_2), (x_3, r_3), Plain\_image : P_{3\times M\times N}$
Output : $index, key1 key2$
$[M, N] \leftarrow size(P)$
for $k = 1 : 3$ **do**
  for $i = 1 : M$ **do**
    for $j = 1 : N$ **do**
      $S_k \leftarrow \sum P(i, j, k)$
    end for
  end for
end for
$S_4 \leftarrow rand()$
$S_5 \leftarrow rand()$
for $k = 1 : 3$ **do**
  $S_k^0 \leftarrow \frac{S_k}{M\times N\times 255}$
end for
$x_1^0 \leftarrow (x_1 + S_1^0 + S_4) mod\ 1$
$x_2^0 \leftarrow (x_2 + S_2^0 - S_5) mod\ 1$
$x_3^0 \leftarrow (x_3 + S_3^0 + S_5) mod\ 1$
$r_1' \leftarrow (r_1 + S_1^0 - S_4) mod\ 4$
$r_2' \leftarrow (r_2 + S_2^0 + S_5) mod\ 4$
$r_3' \leftarrow (r_3 + S_3^0 + S_5) mod\ 4$
$X_1 \leftarrow (r_3 + S_3$
$X_1 \leftarrow x_1^0$
$Z_1 \leftarrow x_3^0$
for $i = 2 : M \times N$ **do**
  $X_i \leftarrow (r_1' - 2^7(r_1' - X_i^2)^2) mod\ 1$
  $key1_i \leftarrow floor(X_i \times 10^{15}) mod\ 256$
  $Y_i \leftarrow (r_2' - 2^7(r_2' - X_i^2)^2) mod\ 1$
  $key2_i \leftarrow floor(Y_i \times 10^{15}) mod\ 256$
end for
for $i = 2 : 3 \times M \times N$ **do**
  $Z_i \leftarrow (r_3' - 2^7(r_3' - X_i^2)^2) mod 1$
end for
$key1 \leftarrow reshape(key1, M, N)$
$key2 \leftarrow reshape(key2, M, N)$
$[Z_s, index] \leftarrow Sort(Z)$

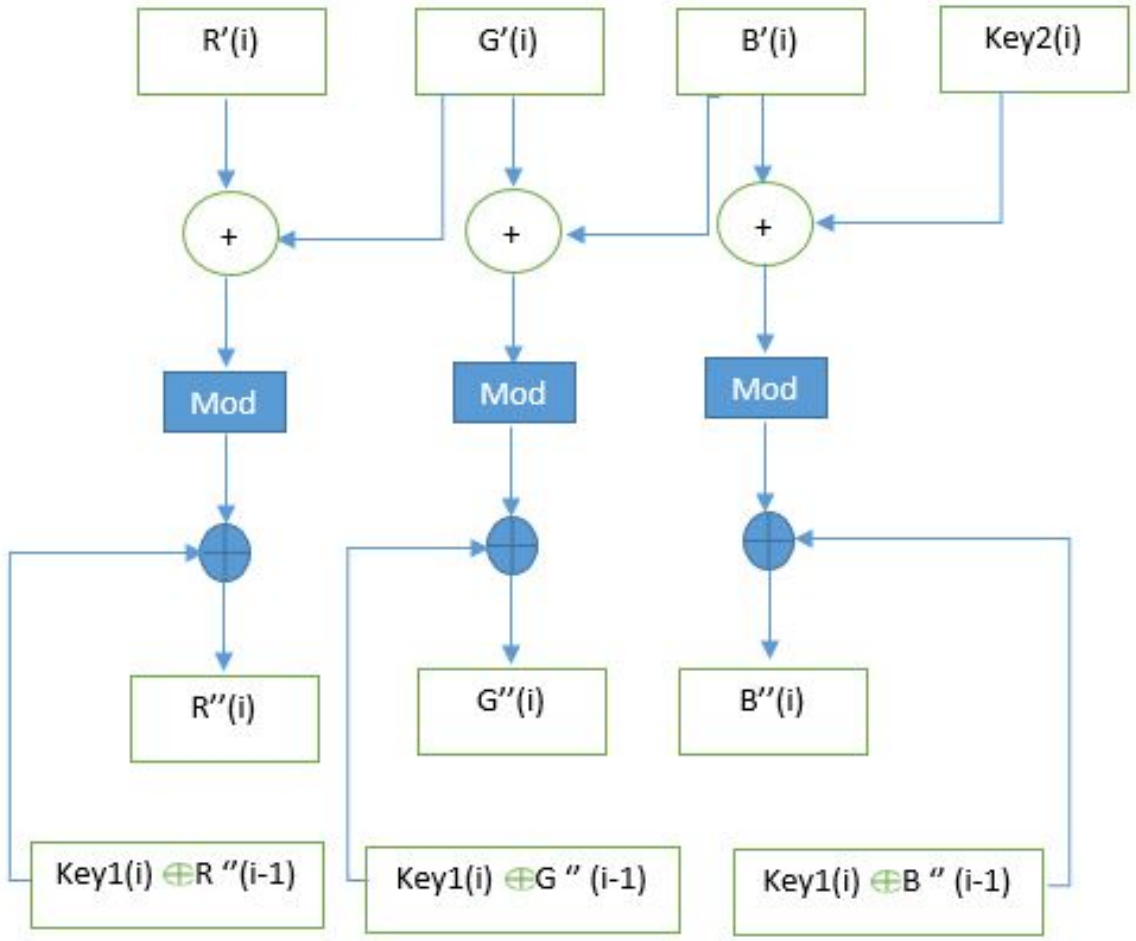


Fig. 3.5 *Processus de diffusion en cryptage .*

$$\begin{cases} B'(i) \leftarrow [(B"(i-1) \oplus key1(i)) \oplus B"(i)] - key2(i)] \\ mod\ 256. \\ G'(i) \leftarrow [(G"(i-1) \oplus key1(i)) \oplus G"(i)] - B(i)] mod\ 256. \\ R'(i) \leftarrow [(R"(i-1) \oplus key1(i)) \oplus R"(i)] - G'(i)] mod\ 256. \end{cases} \tag{3.18}$$

$$O(index(i)) = P(i). \tag{3.19}$$

3.4.3 Surligner

Le nouveau système chaotique amélioré, EQM, a de meilleures performances chaotiques que la carte quadratique classique. En générant des paramètres de contrôle et des valeurs initiales liées aux valeurs de pixels de l'image originale en tant que clé secrète, les flux de clés sont fortement liés à l'image à chiffrer, par conséquent, le schéma proposé peut résister aux attaques en texte clair choisies. La formule proposée dans les opérations de diffusion rend la relation

Algorithm 2 Algorithme de cryptage

Input : $key1, key2, index, Plain\_image : P_{3\times M\times N}$
Output : $index, key1key2$
Lire l'image originale P
Transformez la matrice d'image en matrice 1d avec une taille de $[M, 3\times N]$
transformer la matrice obtenue en vecteur unidimensionnel O de taille $[1, 3\times M\times N]$
$[M, N] \leftarrow size(P)$
for $i = 1 : 3\times M\times N$ **do**
  $Permuted_i \leftarrow O(index_i)$ %Permutation
end for
Transformez le vecteur permuté en matrice 1d avec une taille de $[M, 3\times N]$.
Transformez la matrice obtenue en 3 matrices R, G et B de taille $[M\times N]$.
$R"_1 \leftarrow R'_1 \oplus G'_1$
$G"_1 \leftarrow G'_1 \oplus B'_1$
$B"_1 \leftarrow B'_1 \oplus key2_1$
for $i = 2 : M\times N$ **do**
  $R"_i \leftarrow [(R'_i + G'_i) mod 256] \oplus R"_{i-1} \oplus key1_i$
  $G"_i \leftarrow [(B'_i + G'_i) mod 256] \oplus G"_{i-1} \oplus key1_i$
  $B"_i \leftarrow [(B'_i + key2_i) mod 256] \oplus B"_{i-1} \oplus key1_i$
end for
L'image cryptée finale C est composée en utilisant $R", G"and B"$. components.

entre le texte en clair et le texte chiffré plus complexe, tandis qu'un petit changement dans n'importe quelle valeur de pixel de l'image originale changera toutes les valeurs de pixel de l'image cryptée. Le schéma proposé chiffre les trois composantes de l'image en même temps, cela peut être considéré comme un avantage pour accélérer le processus de cryptage.

3.5 Résultats expérimentaux et discussion

Dans cette section, la performance de l'approche proposée sera démontrée et évaluée à travers les expériences de simulation réalisées sur la plateforme MATLAB.2016. À cette fin, Cinq données d'images couleur sont utilisées comme images d'entrée pour l'algorithme de chiffrement et de déchiffrement proposé, elles sont illustrées sur la figure 3.6 sous forme de vie réelle 1, de poivrons, de Babbon et de vie réelle 2. L'algorithme de chiffrement prend en compte la couleur fichier de données d'image en entrée, et effectue l'opération de cryptage sur le fichier pour produire l'image couleur cryptée représentée sur la figure3.7. D'autre part, l'algorithme de décryptage prend le fichier image couleur crypté en entrée et effectue une opération de décryptage sur le fichier pour produire l'image couleur d'origine. En comparant les images en couleur et leurs images cryptées sur les Fig. 3.6 et 3.7, il n'y a aucune information visuelle observée dans les images couleur cryptées, par conséquent, les images cryptées sont visuellement indiscernables même avec une grande différence trouvée dans les images originales. La robustesse de l'approche

proposée est testée sur Cinq images de test et les résultats sont calculés pour chaque image. Les résultats du schéma proposé obtenus à partir de la base de données d'images USC-SIPI sont présentés sur les Fig.3.6 et 3.7. Les tests de performance de l'algorithme proposé sont présentés dans l'analyse suivante.

3.5.1 Espace clé de sécurité

Pour un bon système de chiffrement, la taille de l'espace clé doit être suffisamment grande pour rendre impossible le succès des attaques par force brute, dans l'algorithme proposé, l'espace clé peut être résumé comme les paramètres k_i de EQM, où $k_i \in [0, 8]$, les valeurs et paramètres initiaux de l'EQM, à savoir x_{i0} et r_i, où i = 1, 2, 3. Pour une sécurité suffisante pour rendre l'attaque par force brute impossible, la taille de l'espace clé doit être $> 2^{100}$ [3, 48]. Supposons que la précision des clés soit de 10^{-15}, donc l'espace total des clés est de $10^{(15\times9)}$. Par conséquent, cette taille totale de l'espace clé est suffisamment importante pour s'opposer à l'attaque par force brute.

3.5.2 Analyse d'histogramme

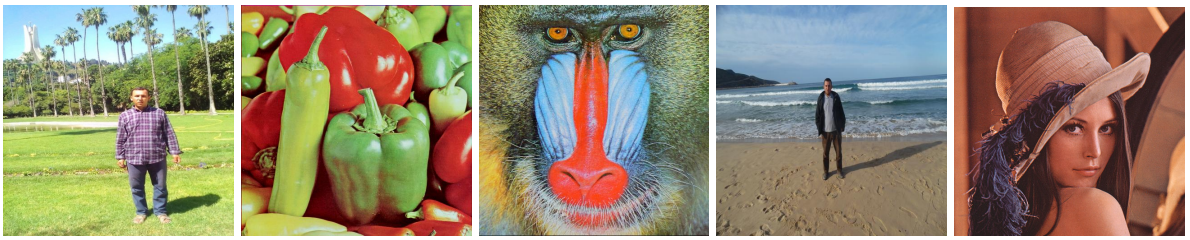
Le nombre de pixels correspondant à chaque intensité de couleur et la répartition des pixels dans une image sont représentés par un histogramme [49]. La figure3.8 montre les histogrammes de chaque composant d'une image Lena à côté de ses histogrammes d'image chiffrée. L'histogramme de l'image cryptée est présenté relativement uniforme et significativement différent de celui de l'image d'origine. Par conséquent, l'histogramme de l'image cryptée ne contient pas d'informations statistiques à utiliser dans les attaques statistiques.

3.5.3 Entropie de l'information

Le tableau 3.3 montre que l'entropie de l'information obtenue en appliquant le schéma proposé d'image cryptée est assez proche de la valeur théorique de 8 [50]. Par conséquent, la méthode de cryptage proposée est sécurisée lors de l'attaque d'entropie, en outre, est comparée à [42, 41, 46] les valeurs obtenues de notre schéma sont plus élevées, par conséquent, les images cryptées du schéma proposé ont de meilleures distributions aléatoires et également, ils sont comme une image aléatoire.

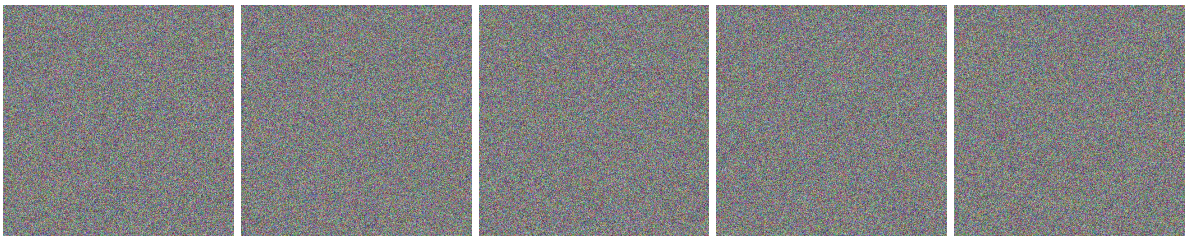
3.5.4 Coefficient de corrélation

Le tableau 3.4 montre que les degrés de corrélation obtenus de l'image chiffrée sont assez proches de la valeur de zéro dans toutes les directions, tandis que les coefficients de corrélation de l'image d'origine sont assez proches de la valeur de 1 dans toutes les directions, ainsi,



(a) *Vie réelle 1.* **(b)** *Peppers.* **(c)** *Babbon.* **(d)** *Vie réelle 2.* **(e)** *Lena.*

Fig. 3.6 *Cinq images clairs.*



(a) **(b)** . **(c)** . **(d)** **(e)**

Fig. 3.7 *Cinq images chiffrés.*
(a)Vie réelle 1, (b)Peppers, (c)Babbon ,(d) Vie réelle 2 and (e) Lena.

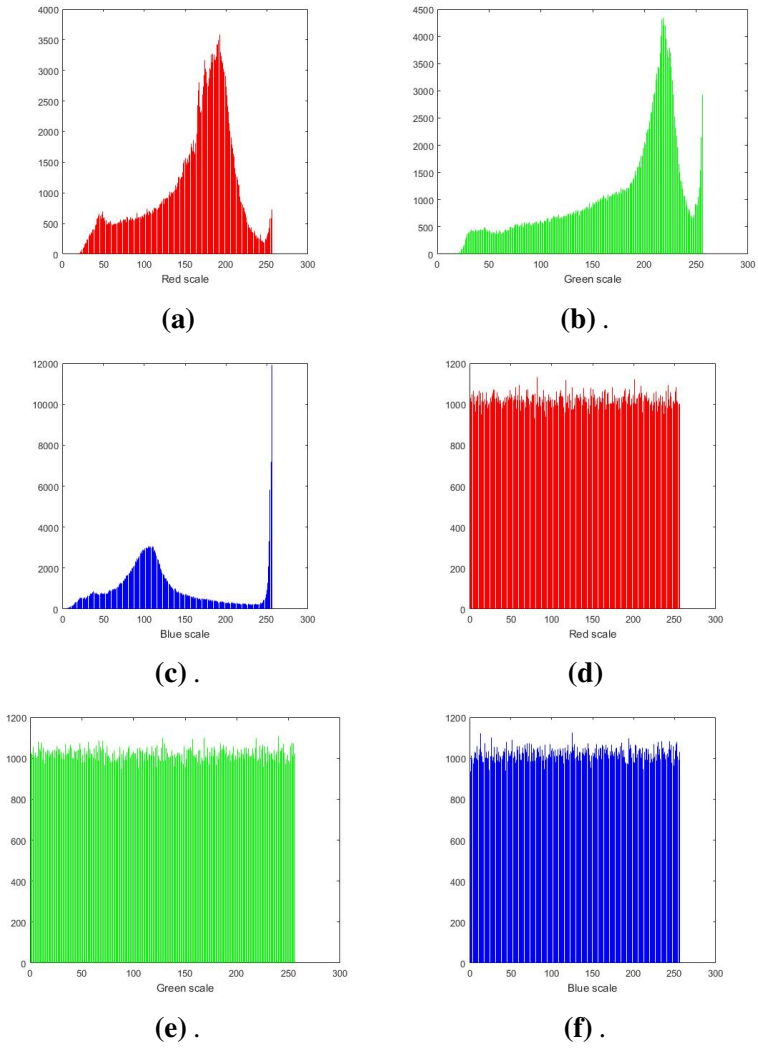
TABLE 3.3 *Analyse d'entropie d'informations de diverses images.*

Image	Images clair	Notre méthode	Ref[42]	Ref [41]	Ref [46]
Lena	7.4767	7.9998	7.9997	7.99971	7.9991
Baboon	7.7624	7.9998	7.9997	7.9996	7.9997
Peppers	7.6698	7.9998	7.99973	7.9996	7.9997
Fig.6.a	7.7509	9996	7.9991		

l'image cryptée est fortement indépendante de l'image d'origine, cela peut être démontré par la figure3.9 et comparé à certains algorithmes. Les résultats obtenus à partir du schéma proposé ont détruit la forte corrélation entre les pixels adjacents. On peut voir que le schéma de système proposé surpasse. Par conséquent, l'algorithme proposé sécurise les images contre les attaques statistiques.

3.5.5 Attaque différentielle

Le tableau 3.5.5 donne les résultats des tests de NPCR et UACI. On peut voir que les résultats obtenus sont plus proches des valeurs idéales. Ainsi, les résultats ont prouvé l'efficacité de l'algorithme contre les attaques différentielles, car les valeurs résultantes sont dans les valeurs trouvées dans [51].



(a) **(b)** .

(c) . **(d)**

(e) . **(f)** .

Fig. 3.8 *Cinq images cryptées.*
(a) - (c) Histogramme des composantes R, G, B de l'image ordinaire Lena, (d) - (f) Histogramme des composantes R, G, B de l'image chiffrée

3.5.6 la sensibilité à la clé

Un schéma de cryptage d'image sécurisé doit être extrêmement sensible à sa clé, la clé utilisée dans le processus de chiffrement et de déchiffrement, c'est-à-dire que les images déchiffrées doivent être totalement différentes même si la seule clé est légèrement modifiée. Dans notre algorithme, on peut considérer que les valeurs initiales et les paramètres de contrôle ($x_{0,1} = 0.235, x_{0,2} = 0.563, x_{0,3} = 0.6635, r_1 = 3.235, r_2 = 2.521, r_3 = 3.85, k_1 = 7, k_2 = 6$
$, k_3 = 5$) comme clés de sécurité. Afin de détecter la sensibilité, chaque clé est modifiée de 10^{-15} pour observer l'effet de l'image déchiffrée. La figure 3.10.a montre l'image déchiffrée avec toutes les clés secrètes correctes. L'image décryptée est présentée sur la figure 3.10.b avec le clés incorrectes $x_{0,1} + 10^{-15}$; de même, si f$x_{0,2}, x_{0,3}, r_1, r_2, r_3, k_1$ changent de 10^{-15}, respectivement, alors que toutes les autres clés sont correctes, les Fig. 3.10.c-f illustrent les images déchiffrées

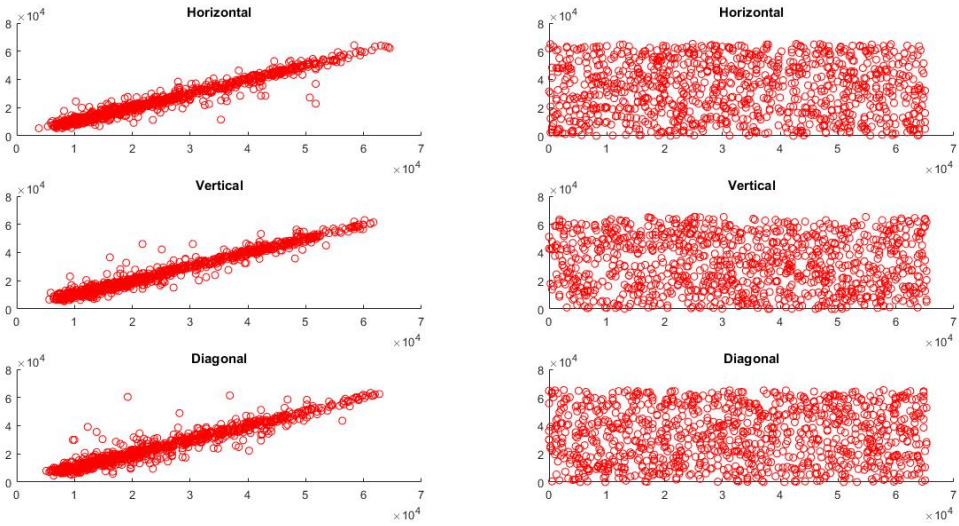


Fig. 3.9 *Distribution des pixels voisins dans différentes directions de Lena. La première colonne présente l'image clair, la deuxième colonne présente l'image chiffrée (*512 × ×512 *pixels) (a) Image originale, (b) Image cryptée.*

TABLE 3.4 *Coefficients de corrélation de deux pixels adjacents dans la Lena chiffrée et clair et comparer avec différents cryptosystèmes.*

Canal	Directions	Notre Algorithme	Ref [46]	Ref [42]	Ref [41]	Ref [5]	Ref [40]
Canal R	Horizontal	-0.0022	0.006	0.0021	0.0037	0.000538	0.0028
	Vertical	0.0022	-0.0009	-0.00042	-0.076	-0.007058	0.0085
	Diagonal	-0.00019	-0.0010	-0.0019	-0.0030	0.000573	0.0017
Canal G	Horizontal	-0.00017	-0.00044	-0.0030	-0.0033	0.001186	0.0020
	Vertical	-0.0004	0.00091	0.0031	-0.759	0.000177	0.00005
	Diagonal	-0.0005	-0.0031	-0.0029	0.0021	-0.001693	0.0035
Canal B	Horizontal	0.0006	0.00075	-0.0024	0.0035	-0.002372	0.00199
	Vertical	-0.0054	0.0021	0.0023	-0.0758	0.007818	0.00039
	Diagonal	-0.0032	0.0021	0.0017	-0.0031	-0.000927	-0.0014

correspondantes. On peut observer que les images déchiffrées par de mauvaises clés sont toutes méconnaissables. On peut conclure que les clés de sécurité présentent une sensibilité élevée.

3.5.7 Perte de données et attaques de bruit

Les images numériques peuvent être sujettes à des phénomènes anormaux tels que la perte de données et les attaques de bruit pendant la transmission via le réseau et pendant le stockage. Un bon schéma de cryptage d'image devrait résister à ces phénomènes anormaux. L'attaque de bruit et la perte de données sur une image cryptée sont utilisées pour vérifier

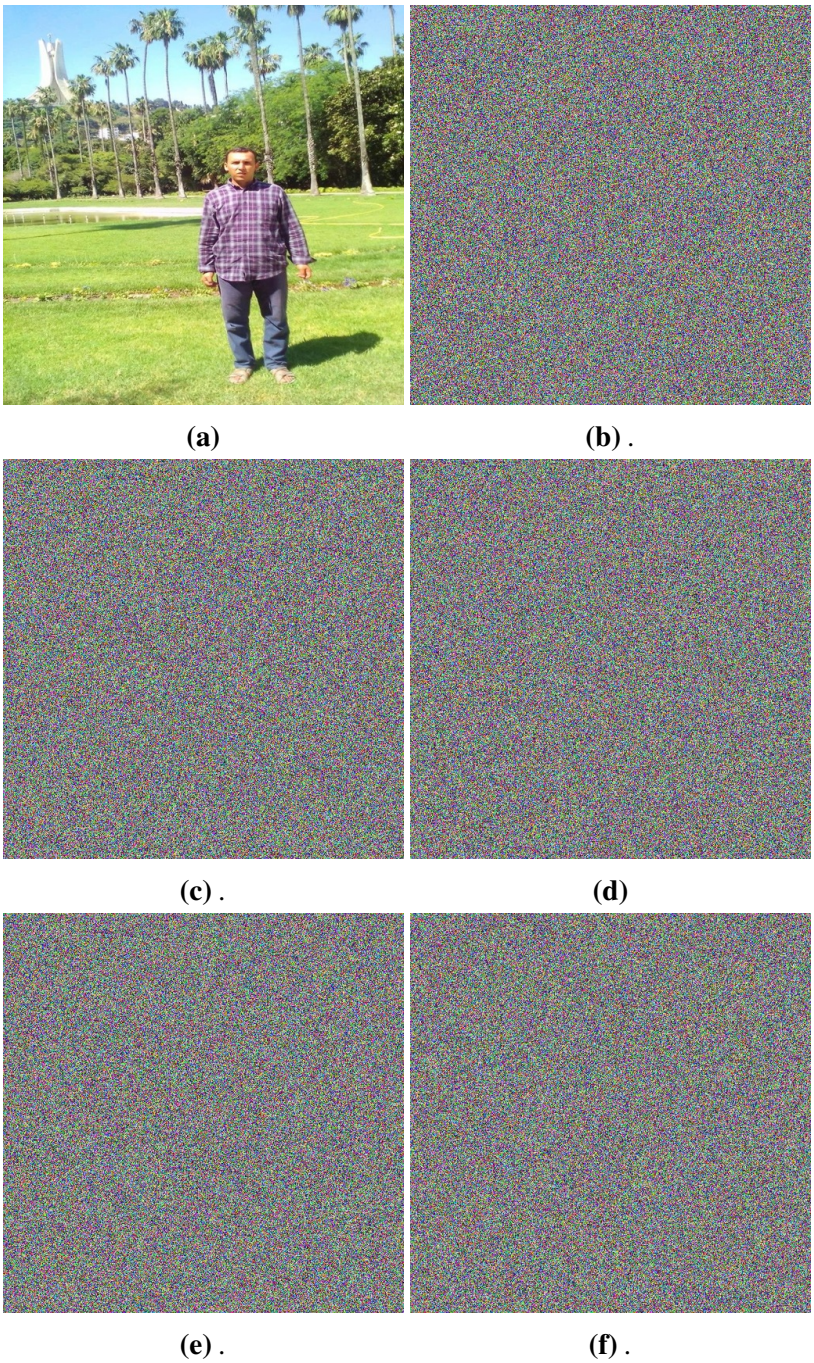
TABLE 3.5 *NPCR et UACI de diverses images cryptées.*

	NPCR(%)			UACI(%)		
Image	R	G	B	R	G	B
Lena	99.59	99.60	99.61	33.46	33.44	33.47
Baboon	99.59	99.60	99.61	33.47	33.48	33.46
Peppers	99.59	99.60	99.61	33.45	33.43	33.48
Fig.6.a	99.58	99.60	99.58	32.89	33.66	33.32
Lena in[46]	99.6425	99.6271	99.6387	33.3827	33.3775	33.3329
Peppers in [46]	99.65	99.62	99.58	33.45	33.38	33.59
Lena in [41]	99.65	99.64	99.64	33.36	33.59	33.45
Fig.6.a in [40]	99.66	99.64	99.64	33.08	32.95	33.63

la capacité de résistance aux attaques de l'algorithme proposé. L'image Lena est cryptée par l'algorithme proposé, puis sur cette image cryptée est appliquée une découpe de données d'une taille de 64×64 (Fig. 3.12.a), 128×128 (Fig.3.12.b) et 128×256 (Fig.3.12.c), individuellement, ces images sont ensuite déchiffrées. Comme on le voit sur (Fig.3.12.d), 128×128 (Fig.3.12.e), 128×512 (Fig.3.12.12f), respectivement. En outre, l'image cryptée est attaquée avec $1\%'salt\&pepper'$, $5\%'salt\&pepper'$, $10\%'salt\&pepper'$ de bruit de sel et de poivre individuellement. Leurs images déchiffrées sont représentées sur les Fig. 3.11.e et 3.11.f. On peut observer que l'image déchiffrée contient la majorité des informations visuelles originales même si les pertes de données et le bruit sont limités. Par conséquent, le schéma proposé est robuste contre les attaques de perte de données et la contamination par le bruit.

3.5.8 Attaque connue / choisie

Un cryptosystème efficace devrait résister aux attaques de texte en clair choisi et de texte en clair connu. Selon l'algorithme proposé, les paramètres de contrôle et les conditions initiales sont associés à l'image claire. Ainsi, diverses images ont des clés diverses, en plus des nombres aléatoires utilisés, où l'algorithme proposé génère diverses itérations d'images chiffrées lorsque le schéma de chiffrement est utilisé dans la même image. Cela peut être prouvé par les résultats montrés sur la figure 3.13. Les mêmes clés de sécurité sont utilisées, l'algorithme proposé a été appliqué à l'image en clair deux fois sur la figure 3.13a. On peut voir deux images cryptées C1 (Fig.3.13b) et C2 (Fig. 3.13c) dans la première et la deuxième itération de cryptage, respectivement, la différence pixel par pixel est $C1 - C2$ comme le montre la Fig.3.13d, qui



(a) **(b)** .

(c) . **(d)**

(e) . **(f)** .

Fig. 3.10 *Image de déchiffrement avec un petit changement de clés (a) L'image chiffrée avec toutes les clés secrètes est correcte, (b) Image déchiffrée avec* $x_{0,1} + 10^{-15}$*, (c) Image déchiffrée avec* $x_{0,2} + 10^{-15}$ *; (d) Image déchiffrée avec* $x_{0,3} + 10^{-15}$ *; (e) Image déchiffrée avec* $r_1 + 10^{-15}$ *; (f) Image déchiffrée avec* $k_1 + 10^{-15}$

montre que les deux images cryptées sont extrêmement différentes. Ainsi, l'algorithme proposé est capable de contrer l'attaque en texte clair choisi, par conséquent, il peut résister à d'autres types d'attaques connues / choisies en raison du fait que l'attaque en texte brut choisie est la plus robuste [52].

Fig. 3.11 *Images déchiffrées avec bruit de sel et de poivre*

TABLE 3.6 *Temps de cryptage sur l'image Lena.*

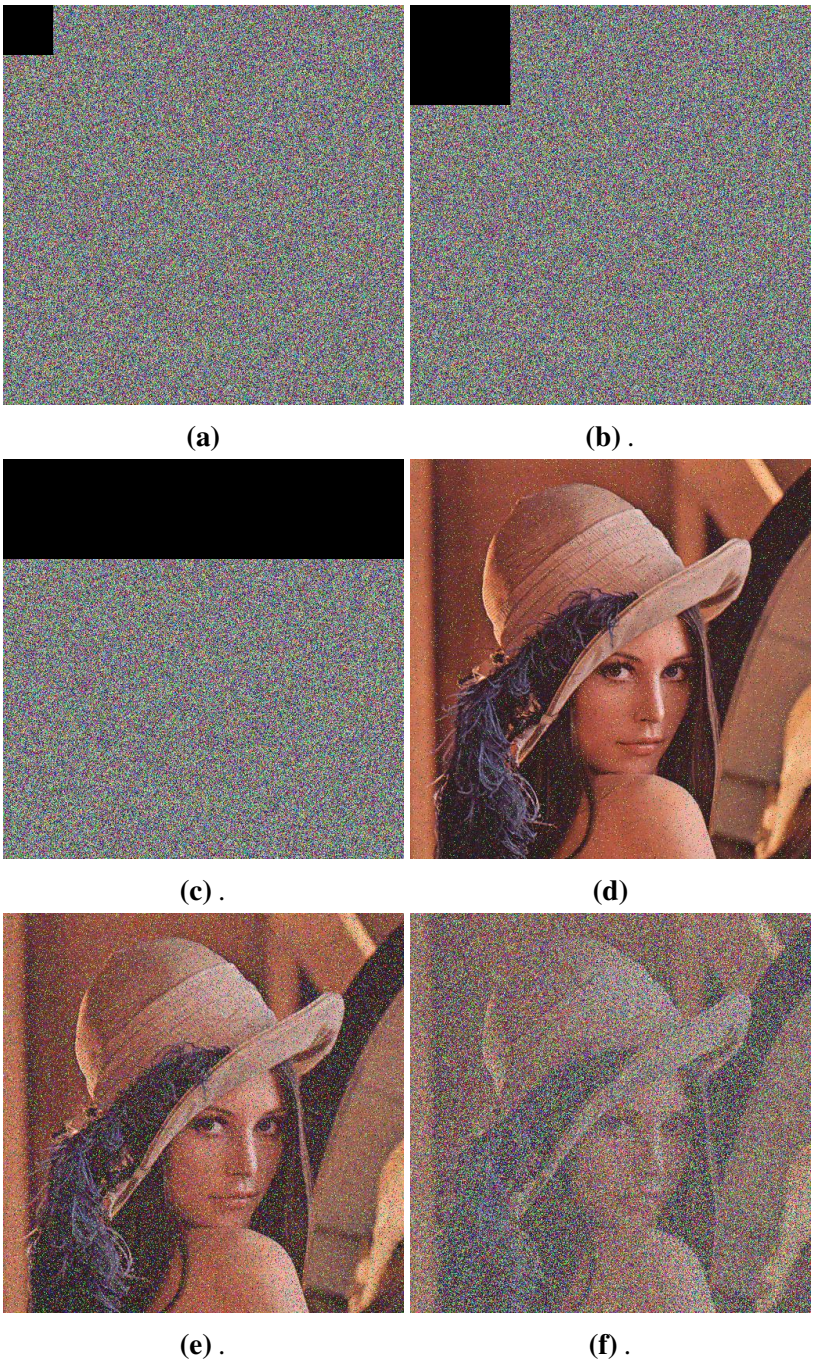
Image	Schéma proposé	Ref. [41]	Ref. [42]	Ref. [5]
Lena 512×512	3.42	3.64	22	4.02

3.5.9 Analyse de vitesse

Dans les applications réelles, le schéma de chiffrement doit avoir de bonnes performances et une vitesse d'exécution rapide pour être utilisé dans les applications en temps réel. Notre algorithme fonctionne sous MATLAB 16 (R2016b) sur PC avec le système d'exploitation Windows 7, le processeur Intel (R) Core (TM) i3-5005 à 2,00 GHz et 4 Go de RAM. Une image couleur Lena de taille 512×512 a été utilisée comme exemple pour tester le cryptage, le taux de décryptage de notre schéma de cryptage d'image, qui consiste à lire l'image originale, à diffuser, à confondre et à générer des séquences chaotiques par EQM. Le tableau 3.6 montre le temps pris dans le processus de cryptage en plus de la comparaison avec d'autres schémas. A partir du tableau 3.6, il peut être prouvé que le schéma proposé a de bonnes performances et une vitesse de fonctionnement rapide, par conséquent, il convient aux applications réelles.

3.5.10 Tests du NIST

Afin de tester le caractère aléatoire des images cryptées par le schéma suggéré dans NIST SP 800-22 (présenté dans la section 4.4). Les détails des résultats obtenus sont illustrés dans le tableau 3.7 Les tests ont prouvé que toutes les images chiffrées ont réussi à remplacer tous les tests NIST. Par conséquent, on peut voir que le schéma proposé crypte les images par de bonnes caractéristiques de caractère aléatoire.



(a) **(b)** .

(c) . **(d)**

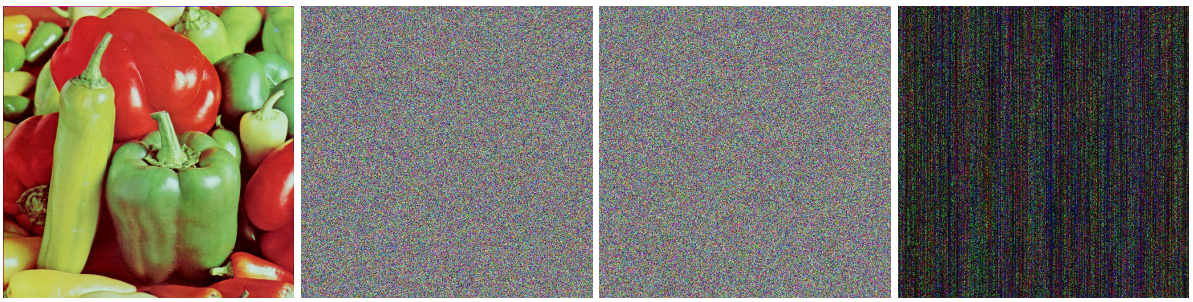
(e) . **(f)** .

Fig. 3.12 *14 Perte de données et attaques de bruit (a) Image cryptée avec une perte de* 64×64 *données, (b) Image cryptée avec une perte* 128×128 *données, (c) Image cryptée avec une perte de* 128×512 *Image décryptée de (a), (e) Image décryptée de (b), (f) Image décryptée de (c)*

.

3.6 Conclusion

Ce travail a deux objectifs. Tout d'abord, une amélioration de la carte quadratique (EQM) a été réalisée en utilisant la carte quadratique classique modifiée et en appliquant le modulaire arithmétique. Le système EQM présente d'excellentes performances telles qu'un meilleur ex-



(a) **(b)** **(c)** **(d)**

Fig. 3.13 *L'algorithme suggéré crypte l'image deux fois en utilisant le même jeu de clés de sécurité (a) Image originale, (b) Première image cryptée, (c) Deuxième image cryptée, (d) Différence pixel à pixel .*

TABLE 3.7 *Résultats des tests NIST-800-22 des images cryptées*

NIST tests	Peppers	Fig.7.a	Results
Frequency	0.3821	0.0386	Passed
Block Frequency	0.4806	0.5095	Passed
The Run Test	0.4370	0.1650	Passed
Longest Run of Ones in a block	0.4135	0.8398	Passed
Binary Matrix Rank	0.2514	0.3315	Passed
DFT Spectral	0.4572	0.0895	Passed
Non-Overlapping Template Matching	0.6298	0.0124	Passed
Overlapping Template Matching	0.2704	0.6681	Passed
Maurer's Universal Statistical Test	0.1231	0.3070	Passed
Linear Complexity	0.6327	0.7861	Passed
Serial Test	0.5451	0.2196	Passed
Approximate Entropy	0.3605	0.0873	Passed
Cumulative Sums	0.0276	0.0549	Passed
Random Excursions	0.9052	0.8512	Passed
Random Excursions Variant	0.5734	0.7112	Passed

posant de Lyapunov et de plus grandes plages chaotiques par rapport à la carte quadratique classique. Le deuxième objectif est l'introduction de la nouvelle approche de chiffrement des images couleur basée sur EQM, le schéma proposé est simple et hautement sécurisé et facile à mettre en œuvre pour le chiffrement et le déchiffrement des images couleur. L'évaluation de la sécurité du schéma est prouvée en le comparant à certains travaux connexes, il a été constaté que le travail proposé peut fournir un grand espace de clés, une sensibilité élevée aux clés et une corrélation plus faible. De plus, ce travail présente un histogramme uniforme et sécurise le cryptage des images contre les attaques statistiques, les attaques différentielles et les attaques de bruit (attaque connue / choisie). Les résultats expérimentaux montrent que le schéma proposé présente des performances supérieures par rapport aux travaux précédents. Actuellement, la reconnaissance de la biométrie a été de plus en plus adoptée dans divers domaines tels que la vérification des aéroports, les secteurs commerciaux, l'authentification des utilisateurs de smartphones personnels ainsi que la criminalistique. Cependant, la fraude d'identité est un problème de sécurité dans les contrôles du système d'accès sécurisé. Dans ce but, les recherches se multiplient pour concevoir des schémas de cryptage d'image non conventionnels tels que le cryptage chaotique pour protéger des informations telles que le modèle de visage, le modèle d'empreinte digitale, etc.

Chapitre 4

Un algorithme de chiffrement d'image modifiable utilisant une carte chaotique logistique améliorée

Sommaire

4.1 Introduction

Dans le cryptage d'image basé sur des cartes chaotiques, la sécurité de l'algorithme de cryptage dépend de la caractéristique des cartes chaotiques et de la structure de l'algorithme [53, 54], donc une meilleure distribution des cartes chaotiques est requise. Cependant, les systèmes chaotiques classiques tels que les cartes logistiques et quadratiques souffrent de nombreuses faiblesses, notamment la gamme limitée de leurs conduites chaotiques, en plus de la distribution non uniforme des données des séquences chaotiques générées [44]. Les cartes chaotiques avec des comportements chaotiques faibles peuvent rendre les cryptosystèmes vulnérables aux attaques et peuvent être facilement brisées. Ces dernières années, plusieurs travaux disponibles dans la littérature ont suggéré de surmonter les inconvénients de ces cartes chaotiques en améliorant les propriétés de la distribution chaotique pour de meilleures performances et une meilleure efficacité des algorithmes de chiffrement d'images [55, 56]. Hua et al. [57] ont suggéré un modèle chaine sinus (SCM) pour améliorer la complexité du chaos de la gamme chaotique existante des cartes chaotiques (1-D). Hua et al. [58] ont proposé un système chaotique basé sur la transformation sinusoïdale (STBCS) pour produire une carte chaotique unidimensionnelle efficace, où ils ont effectué une transformation sinusoïdale à la combinaison des sorties de deux cartes chaotiques existantes. Afin de dissoudre les inconvénients précités, ce travail propose une carte logistique améliorée et évalue ses performances[59]. Les résultats d'analyse du diagramme de bifurcation de ce système chaotique amélioré et l'exposant de Lyapunov montrent qu'il a de bonnes performances chaotiques. De plus, les performances chaotiques sont analysées par un nouvel algorithme de cryptage d'images proposé. Pour tester les applications de la carte logistique améliorée dans le chiffrement d'images, un nouvel algorithme de chiffrement d'images ajustable consistant en une architecture confusion-diffusion a été proposé. Le schéma comprend deux cycles du processus de diffusion et de confusion. Le concept de chiffrement tweak est utilisé pour garantir la variabilité de la technique proposée. Ainsi, avec l'utilisation du tweak, une image originale sera cryptée en différentes images cryptées en utilisant le même flux de clé secrète, car modifié le tweak, il est moins coûteux et plus rapide que de changer la clé du schéma suggéré [60]. Par conséquent, la méthode suggérée peut résister avec succès à l'ttaque texte clair choisi. Ainsi, l'objectif du tweak est de garantir la variabilité. De plus, les tweaks prennent des paramètres supplémentaires en entrée en plus de l'image simple et de la clé secrète, où ces paramètres permettent de contrôler la valeur de l'image cryptée émise sans affecter les clés secrètes [60]. Ainsi, l'algorithme suggéré rompt la limitation de l'algorithme basé sur des clés à usage unique. De plus, le schéma proposé peut crypter de nombreuses images en toute sécurité et rapidement en utilisant la même clé. Les résultats expérimentaux démontrent que l'algorithme est simple, efficace et a une bonne exécution dans le cryptage d'image et a également la capacité de résister à plusieurs attaques. Le chapitre est organisé de la façon suivante : La deuxième section présente brièvement l'efficacité de la carte logistique existante. La troisième section passe en

revue un système chaotique logistique amélioré en utilisant la carte chaotique logistique classique susmentionnée et explique sa précision. La quatrième section propose un nouveau schéma de chiffrement des images modifiables. La cinquième section affiche les résultats de la simulation et l'analyse. Enfin, la section 6 présente la conclusion.

4.2 Analyse de la carte logistique

La carte logistique est l'une des cartes chaotiques 1D les plus célèbres, qui a une équation dynamique non linéaire simple et classique avec des comportements chaotiques complexes, elle peut être décrite par l'équation suivante [61].

$$X_{n+1} = L(r, X_n) = rX_n(1 - X_n). \tag{4.1}$$

Où r est le paramètre de contrôle avec une plage de $r \in [0, 4]$ et X_n est la séquence chaotique de sortie.

4.2.1 Diagramme de bifurcation

Le schéma de bifurcation est l'étude du système chaotique en fonction mathématique des valeurs des paramètres de contrôle [32]. D'après son diagramme de bifurcation présenté dans la figure 4.1(a), deux défauts peuvent être remarqués dans cette carte chaotique : 1) elle a une portée limitée de chaos et, 2) comme démontré dans la figure 4.1(a), la figure 4.2(a), la portée chaotique n'existe que dans [3.57, 4]. Lorsque le paramètre r n'appartient pas à cette plage, il ne peut pas être considéré comme ayant un comportement chaotique et une distribution non uniforme des séquences chaotiques de sortie qui ont affecté les distributions des données d'image cryptées et les performances du système de cryptage.

4.2.2 Exposant de Lyapunov

D'après la figure 4.2 (a), on peut voir que la carte logistique n'est chaotique que lorsque $r \in [3, 57, 4]$ et l'exposant de Lyapunov maximum de la carte logistique est 0,6720.

4.3 La carte logistique améliorée proposée

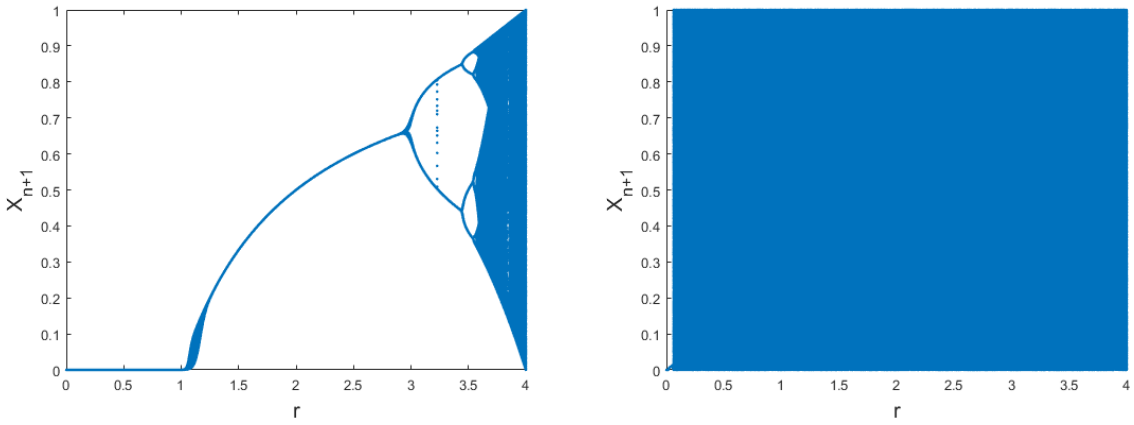
Dans cette section, la carte logistique est améliorée pour résoudre les défauts susmentionnés. La formule de la carte logistique améliorée est décrite comme suit :

$$X_{n+1} = r(2^k \times X_n)(1 - (2^k \times X_n))mod\ 1 \tag{4.2}$$

Où, X_n dans l'équ.4.2 est remplacé par le terme $(2^k \times X_n)$, l'opération "mod" est également appliquée pour garantir que les flux chaotiques générés se trouvent dans la plage de [0, 1] et que la carte présente de bonnes performances chaotiques, lorsque k est dans le [2, 10], cette plage a été prouvée dans l'expérience. La carte chaotique améliorée est examinée par le diagramme de bifurcation et par l'exposant de Lyapunov à la valeur de k = 8.

4.3.1 Analyse de la carte logistique améliorée

Les figures 4.1(b) et 4.2(b) montrent le diagramme de bifurcation et l'exposant de Lyapunov de la carte chaotique améliorée. Les comportements chaotiques de ce système amélioré existent dans l'ensemble des paramètres de contrôle et il est bien plus grand que sa carte de départ. Ainsi, la carte améliorée a de bonnes performances chaotiques et est très appropriée dans les schémas de chiffrement.

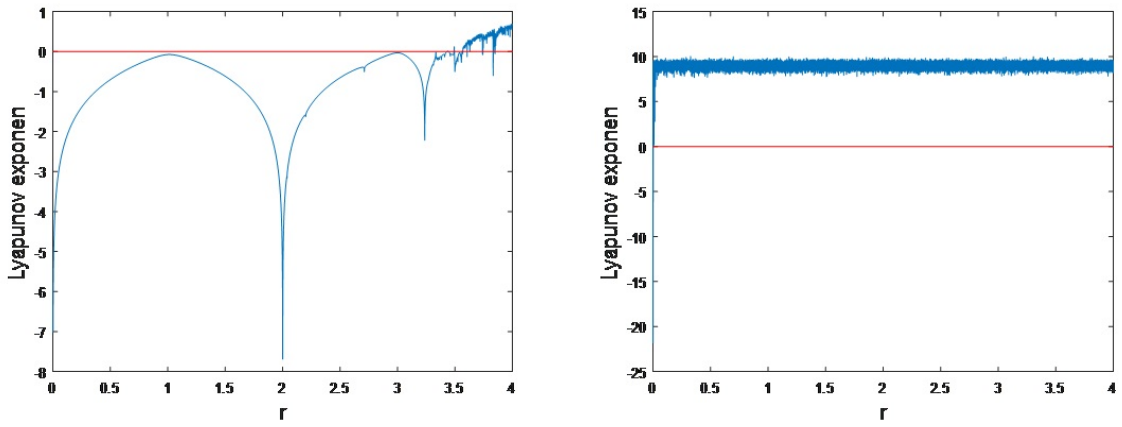


(a) **(b)**

Fig. 4.1 *Les diagrammes de bifurcation de la (a) carte logistique classique ; b) La carte logistique améliorée*

4.3.2 Randomness

Dans les schémas de cryptage d'images basés sur le chaos, il est important d'évaluer la qualité du caractère aléatoire des séquences générées par la carte chaotique améliorée pour s'assurer qu'elles sont appropriées pour les cryptages, pour cette raison, nous avons effectué le test NIST. Les résultats du tableau 4.1 montrent que la carte logistique améliorée a réussi tous les tests NIST. Ainsi, les nombres aléatoires générés par cette carte chaotique améliorée sont prêts pour les cryptages.



(a) **(b)**

Fig. 4.2 *Exposant de Lyapunov de (a) la carte logistique classique ; b) La carte logistique améliorée*

4.4 Algorithme de chiffrement d'images suggéré

Dans cette section, nous suggérons un nouvel algorithme de cryptage d'image ajustable, l'algorithme de cryptage proposé utilise 16 paramètres, donnés comme suit : ($x_{0,1}, x_{0,2}, x_{0,3}, x_{0,4}, x_{0,5}, x_{0,6}$, $r_1, r_2, r_3, r_4, r_5, r_6$, T1, T2, T3, T4) considérée comme clé de sécurité. Le schéma de l'algorithme de chiffrement proposé est illustré à la figure 4.3. Il comporte deux tours d'opération de chiffrement, dans cet algorithme, chaque pixel chiffré n'est pas seulement lié au pixel d'origine qui le génère, mais à tous les autres pixels. Par conséquent, un petit changement dans n'importe quel pixel d'image d'origine conduit à une image cryptée totalement différente. En outre, le concept de chiffrement par blocs modifiables a été utilisé pour garantir que la caractéristique qui modifiera le tweak soit moins coûteuse et plus rapide que la modification de la clé du schéma proposé [60]. Dans notre schéma proposé, les tweaks (T1, T2) sont du premier tour et (T3, T4) du deuxième tour ils sont liés à l'image en clair à chiffrer. Différentes images ont différents tweaks, (T1, T2) du premier tour et (T3, T4) du deuxième tour, donc même si l'attaquant obtient le tweaks (T1, T2, T3 et T4) et la clé de cryptage de certaines images spéciales choisies en texte brut, ces ajustements ne peuvent pas être appliqués pour décrypter l'image de texte chiffré que l'attaquant souhaite obtenir. Ainsi, l'objectif du tweak est d'accorder un changement sans changer la clé de cryptage.

4.4.1 Algorithme de chiffrement

Entrée : image clair I avec une taille de $W \times H$.

Sortie : l'image chiffrée C.

Étape 1 : Deux séquences chaotiques différentes sont générées $S = (S_1, S_2 \ldots . S_{W \times H})$, $Z = (Z_1, Z_2, \ldots . Z_{W \times H})$ de taille $W \times H$, en utilisant l'équation 4.2 avec les valeurs initiales $(x_{0,5}, r_5)$,

TABLE 4.1 *Nist SP800-22 randomness results for the improved logistic map*

NIST tests	P_values	Results
Frequency	0.0367978346837154	Passed
Block Frequency	0.379761079097109	Passed
The Run Test	0.25610508084267	Passed
Longest Run of Ones in a block	0.892791595823537	Passed
Binary Matrix Rank	0.382355232911561	Passed
DFT Spectral	0.0128873729555428	Passed
Non-Overlapping Template Matching	0.451951905074747	Passed
Overlapping Template Matching	0.854994085337196	Passed
Maurer's Universal Statistical Test	0.0598171684079449	Passed
Linear Complexity	0.3071407630330160	Passed
Serial Test	0.433025725806859	Passed
Approximate Entropy	0.501974855003042	Passed
Cumulative Sums	0,887616419296518	Passed
Random Excursions	0,887616419296518	Passed
Random Excursions Variant	0.987809436580346	Passed

$(x_{0,6}, r_6)$, respectivement.

Étape 2 : Afin de briser la corrélation intense entre les pixels voisins, nous proposons un nouvel algorithme de permutation avec la possibilité de changer simultanément la ligne et la colonne d'une image dans le même processus. Deux tours d'une permutation peuvent obtenir un excellent résultat de confusion en théorie. Par conséquent, notre approche proposée utilise deux tours de chiffrement pour obtenir une haute sécurité. La procédure de l'algorithme de permutation proposé peut être décrite comme suit :

- ✓ 1- Deux séquences aléatoires, $X = (X_1, X_2 ..... X_H)$ de longueur H et $Y = (Y_1, Y_2 ..... Y_W)$ de longueur W sont générées en utilisant les équations 4.2, avec les valeurs initiales $(x_{0,1}, r_1)$, $(x_{0,2}, r_2)$ respectivement.
- ✓ 2- X et Y sont triés respectivement pour obtenir deux séquences d'index, I et J.
- ✓ 3- Deux matrices aléatoires, A de longueur $H \times 2$ et B de longueur $W \times 2$ sont générées en utilisant Eq.4.2, avec les valeurs initiales $(x_{0,3}, r_3)$, $(x_{0,4}, r_4)$, respectivement. Ceux-ci seront utilisés pour contrôler la direction de balayage et de permutation. L'algorithme 3 affiche le code du processus de permutation proposé, et pour mieux expliquer le processus de la permutation proposée, un exemple numérique avec une image de taille 8×8 est donné à la figure4.5 et à la figure 4.6. L'image obtenue est remodelée en un vecteur, $P''' = (p_1, p_1 ..... p_{W \times H \times 1})$.

Étape 3 : Le processus de diffusion du schéma proposé est détaillé dans l'algorithme 1 et la figure

4.4, qui prend trois entrées : les clés secrètes, en sortie une image chiffrée $C = (C_1, C_2, ...C_{1\times W\times H})$ est produite et deux tweak T1 et T2.

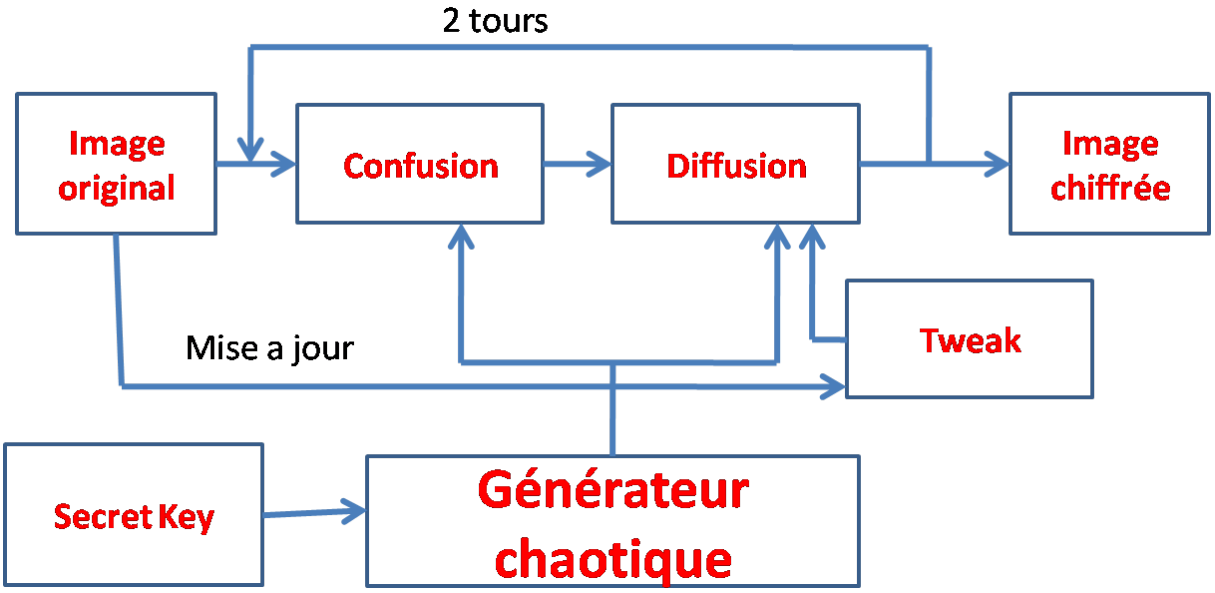


Fig. 4.3 *Schéma fonctionnel du schéma proposé*

4.4.2 Algorithme de déchiffrement

Le processus de décryptage est le processus inverse du cryptage. Le processus inverse de diffusion a été expliqué dans l'algorithme 2.

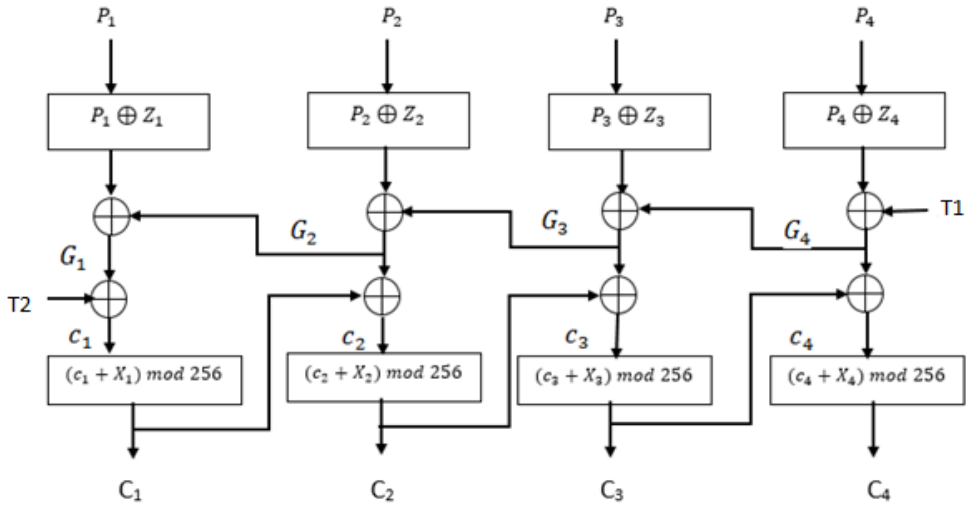


Fig. 4.4 *La méthode proposée de schéma de diffusion basé sur tweak utilisant 4 pixels*

Algorithm 2 invrse_diffusion

input: Encrypted_image C; Secret_keys: $x_{0,2}$,$u_{0,2}$, k_2, $x_{0,3}$,$u_{0,3}$, k_3,Two tweak T1,T2.

output: Permuted_image: P Les Secret_keys sont utilisés pour obtenir une séquence chaotique X et Z de taille H × W, Z et X sont convertis en une séquence de valeurs entières, en suivant l'équation Z(i)=floor(Z(i) $\times 10^{15}$)), X(i)=floor(X(i) $\times 10^{15}$).

n←H×W

$G_1 \leftarrow c_1 \oplus$T2

$c_1 \leftarrow (C_1\text{-}X_1)$ mod 256

for i←2 to n

$c_i \leftarrow (C_i\text{-}X_i)$ mod 256

$G_i \leftarrow c_i \oplus C_{i-1}$

End

$P_n \leftarrow G_n \oplus Z_n \oplus$T1

for i←n-1 to 1

$P_i \leftarrow G_i \oplus Z_i \oplus G_{i+1}$

End

Algorithm 1 diffusion

1 input: Permuted_image: P''' ; Secret_keys: $x_{0,2}$, $u_{0,2}$, k_2, $x_{0,3}$, $u_{0,3}$, k_3.

2 **output:** Encrypted_image: *C, Tweak T1,T2,T3 and T4*

3 Les clés secrètes sont utilisées pour obtenir une séquence chaotique X et Z de taille H × W, Z et X sont convertis en une séquence de valeurs entières, en suivant l'équation :

4 Z(i)=floor (Z(i) × 10^{15}), X(i)=floor (X(i) × 10^{15})

5 n←H×W, t1←rand(), where t1∈ [0,255]

6 $T1 \leftarrow P'''_n \oplus t1$

7 $G_{n+1} \leftarrow T1$

8 for i←n to 1

9 $G_i \leftarrow P'''_i \oplus Z_i \oplus G_{i+1}$

10 end

11 t2←rand (), where t2∈ [0,255]

12 $T2 \leftarrow P'''_1 \oplus t2$

13 $c_1 \leftarrow T2 \oplus G_1$

14 $C_1 \leftarrow (c_1 + X_1) \bmod 256$

15 for i←2 to n

16 $c_i \leftarrow G_i \oplus C_{i-1}$

17 $C_i \leftarrow (c_i + X_i) \bmod 256$

18 end

19 Le C chiffré est remodelé en matrice 2D avec la taille W×H

Algorithm 3 *Permutation*

1 **Input**: A, B, I , J, P.

2 **Output**: *Image en niveaux de gris permutée P'''*

3 k←H

4 **for** *i*← 1 **to** *W* **do**

5 **if** $A(I(i), 1) \geq A(I(i), 2)$ **then**

6 **for** $j \leftarrow 1$ **to** *H* **do**

7 *La ligne I (i) est permutée de gauche à droite en utilisant ce qui suit:*

8 $P'(I(i), j) \leftarrow P(i, j)$

9 **End**

10 **Else**

11 **for** $j \leftarrow 1$ **to** H **do**

12 *La ligne I (i) est permutée de droite à gauche en utilisant ce qui suit:*

13 $P'(I(i), k) \leftarrow P(i, k)$

14 $K \leftarrow k - 1$;

15 **End**

16 **End**

$K \leftarrow H$

18 **End**

19 *Le deuxième tour de permutation*

20 $P'' \leftarrow rot90(P')$

21 $k \leftarrow W$

22 **for** *i*← 1 **to** *H* **do**

23 **if** $B(J(i), 1) \geq B(J(i), 2)$ **then**

24 **for** $j \leftarrow 1$ **to** *W* **do**

25 *La ligne J (i) est permutée de gauche à droite en utilisant ce qui suit:*

26 $P'''(J(i), j) \leftarrow P''(i, j)$

27 **End**

28 **Else**

29 **for** $j \leftarrow 1$ **to** W **do**

30 *La ligne J (i) est permutée de droite à gauche en utilisant ce qui suit:*

31 $P'''(J(i), k) \leftarrow P''(i, k)$

32 $K \leftarrow k - 1$;

33 **End**

34 **End**

35 $K \leftarrow W$

36 **End**

37 $P \leftarrow P'''$

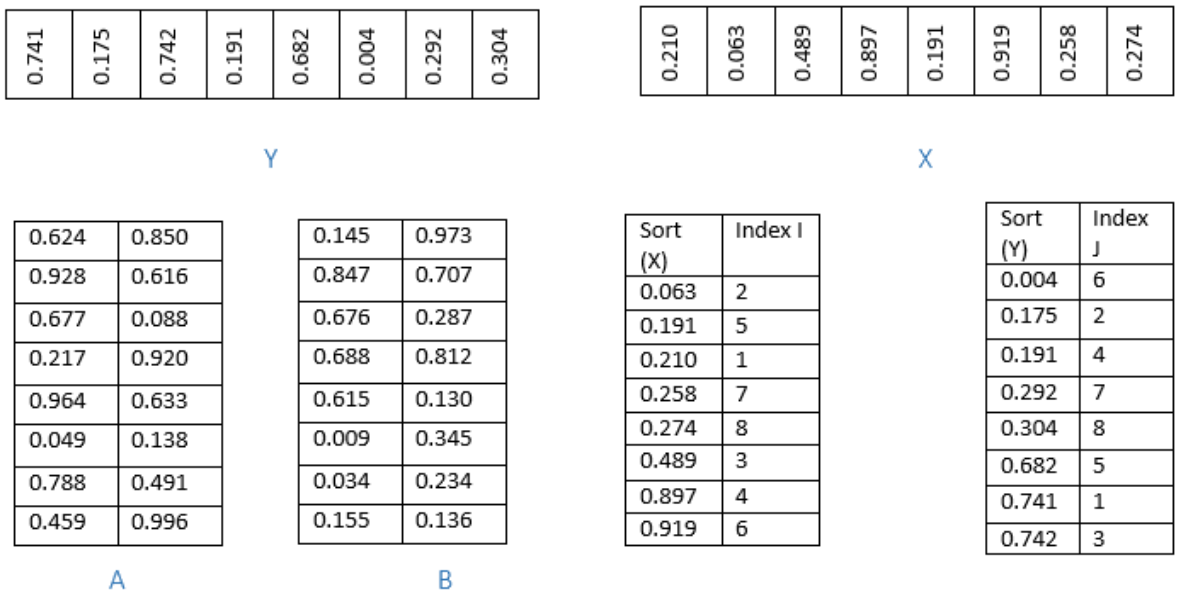


Fig. 4.5 *Exemple de génération de séquences chaotiques : (a) génération de deux matrices d'index 1d I et J ; (b) générer deux matrices aléatoires A et B*

4.5 Résultats expérimentaux

Dans cette section, les performances de l'algorithme proposé seront discutées à travers les résultats obtenus. De plus, certains types de tests seront utilisés pour montrer la supériorité de la méthode de cryptage proposée. Les quantités à mesurer sont : NPCR, UACI, l'analyse de corrélation, l'espace clé, l'évaluation de la sensibilité et de l'entropie des informations, le caractère aléatoire des images cryptées. Pour expliquer davantage l'efficacité de la méthode suggérée, elle a été comparée aux algorithmes avancés de cryptage d'image suivants : [62, 63, 43, 64, 62].

4.5.1 Analyse de l'espace clé

Afin de garantir que l'attaque par force brute est irréalisable, l'espace doit être supérieur à 2^{100} [27]. Les clés secrètes que nous avons utilisées dans notre schéma proposé sont résumées comme suit : 1- Les paramètres de contrôle r_1, r_2, r_3, r_4, r_5, r_6. 2- Les valeurs initiales $x_{0,1}$, $x_{0,2}, x_{0,3}, x_{0,4}, x_{0,5}, x_{0,6}$ et quatre tweak T1, T2, T3 et T4 $\in [0, 255]$, l'espace de chaque valeur initiale est 10^{15}, donc l'espace clé de notre schéma suggéré est $4 \times 256 \times 10^{12\times15}$. Le tableau 4.2 répertorie la comparaison de l'espace clé de notre algorithme proposé avec certains algorithmes avancés de chiffrement d'images[62, 32, 27, 63]. Par conséquent, il est suffisamment grand pour résister à l'attaque par force brute et l'algorithme de chiffrement suggéré est préférable de résister à une attaque par force brute.

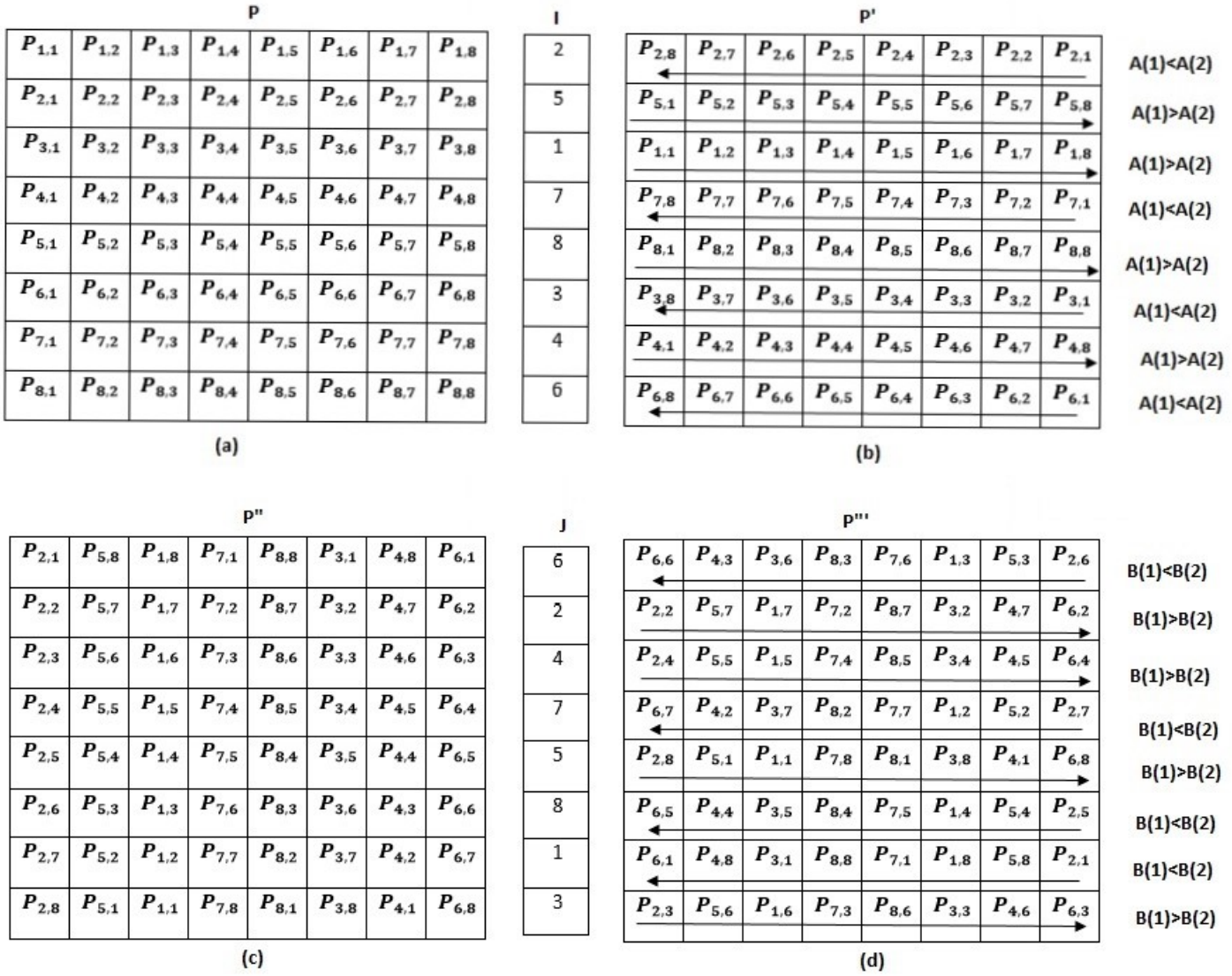


Fig. 4.6 *Un exemple de permutation proposée : (a) pixels dans l'image originale P ; (b) permutation vers P' en utilisant I (c) image tournée de 90 degrés dans le sens antihoraire dans le deuxième cycle de cryptage ; (d) permutation vers P" en utilisant j*

TABLE 4.2 *Comparaison avec certains algorithmes dans l'espace clé*

Algorithme.	Espace clé
Notre	$4 \times 256 \times 10^{12\times15} \simeq 2^{600}$
Ref.[62]	$\simeq 2^{256}$
Ref.[63]	$\simeq 2^{256}$
Ref.[64]	$\simeq 2^{256}$
Ref.[62]	$\simeq 2^{180}$
Ref[62].	$\simeq 2^{400}$

4.5.2 L'analyse de l'histogramme

L'histogramme de l'image illustre le nombre de pixels de chaque niveau de gris [43]. Pour résister aux attaques statistiques, l'histogramme doit être assez uniforme. Les figures 4.7 et 4.8 montrent les histogrammes des images claires et les histogrammes de leurs images chiffrées. D'après les figures 4.7 et 4.8, l'histogramme des images chiffrées est assez uniformément réparti et plat, de sorte qu'il suffit à rendre les attaques statistiques irréalisables.

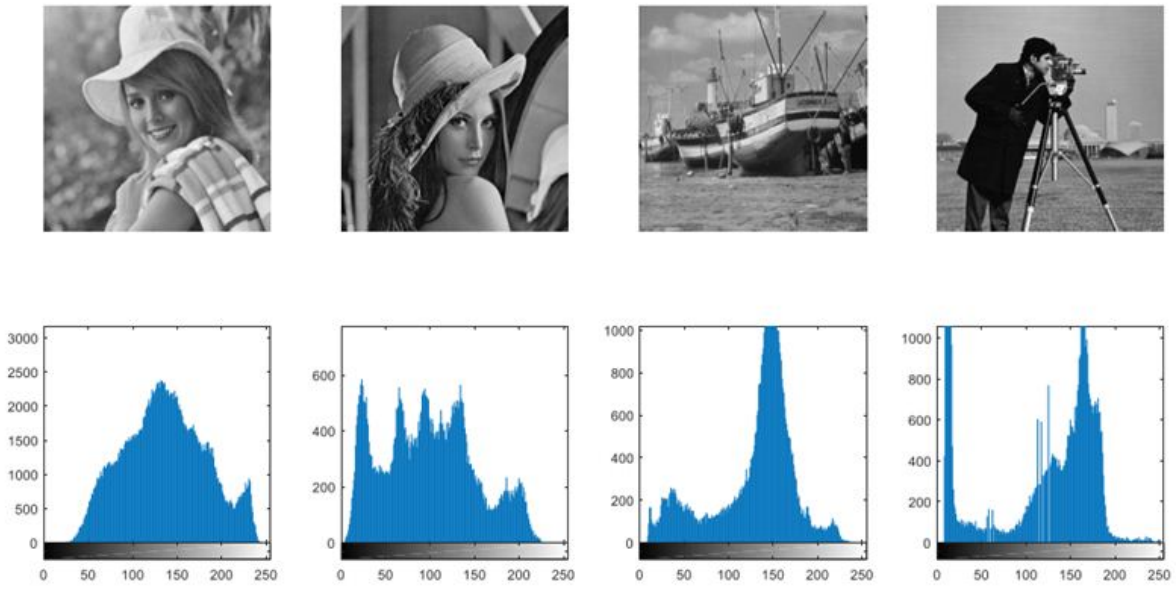


Fig. 4.7 *Images originales de Elaine, Lena, Boat, Cameraman et leurs histogrammes*

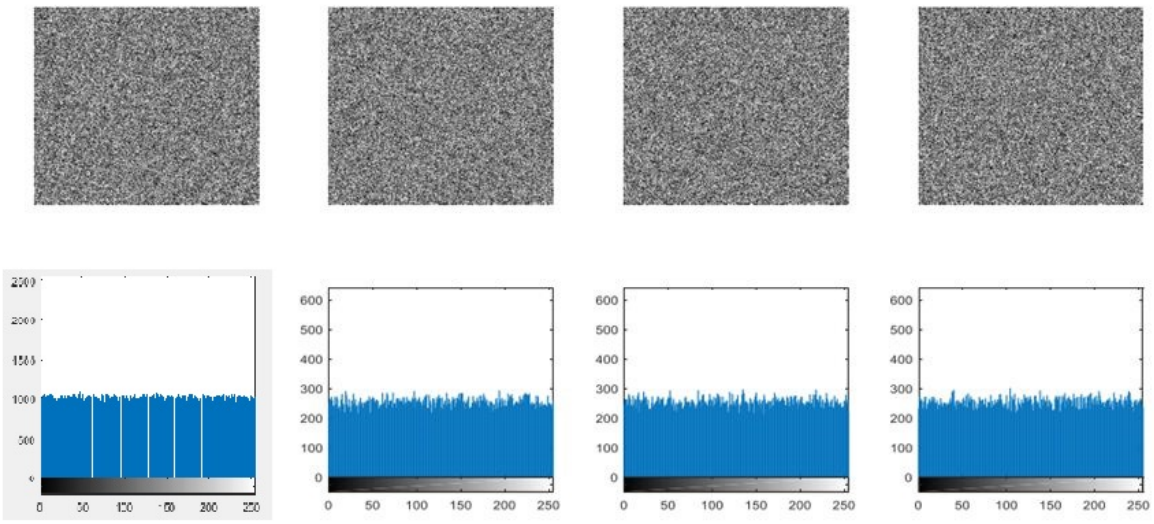


Fig. 4.8 *Images cryptées de Elaine, Lena, Boat, Cameraman et leurs histogrammes respectivement (de gauche à droite)*

4.5.3 Analyse d'entropie de l'information

Les valeurs d'entropie d'informations de différentes images cryptées en utilisant notre schéma sont répertoriées dans le tableau 4.3 et il se réfère que l'entropie des images cryptées proposées est proche de l'entropie idéalement 8 et qu'elles sont toutes supérieures aux valeurs d'entropie d'informations obtenues par Wang et al.[62], Ramadan et al. [32], et Wang et al.[27], on constate que notre schéma est meilleur que les schémas incluant ceux suggérés par Wang et al. [62], Ramadan et al. [32], et Wang et al. [27], la méthode de cryptage proposée est donc sécurisée contre l'attaque entropique.

TABLE 4.3 *Analyse d'entropie d'informations de diverses images.*

Image	Images clair	Notre méthode	Ref[43]	Ref [63]	Ref [62]
Lena	7.5691	7.9976	7.9975	7.9975	7.9970
Baot	7.1913	7.9971	7.9969	7.9974	7.9971
Cameraman	7.6879	7.9972	7.9971	7.9970	7.9972
Baboon	7.3579	7.9994	7.9974	7.9974	7.9969

4.5.4 Les coefficients de corrélation

Les corrélations entre les pixels voisins à la verticale, en diagonale ou à l'horizontale de l'image ordinaire et de son image cryptée correspondante sont illustrées à la figure 4.9. Les coefficients de corrélation selon toutes les directions de certaines images sont répertoriés dans le

tableau4.4. On peut voir que l'image cryptée est très proche de 0. De plus, nous avons comparé notre algorithme proposé à celui utilisé dans la réf.[62, 65, 32, 27, 63], la nôtre a les plus petites valeurs de corrélation dans toutes les directions donc, le cryptosystème proposé protège les images contre attaques statistiques.

TABLE 4.4 *Analyse de corrélation de coefficient.*

Image	Images clair	Image cryptée	Ref[63]	Ref [41]	Ref [62]	Ref[32]	Ref[64]
Diagonal	0.9346	-0.0009	0.0016	-0.0019	-0.0014	-0.0017	0.0012
Horizontal	0.9693	0.0003	-0.0022	0.0019	0.0020	0.0038	0.0024
Vertical	0.9179	-0.0022	-0.00041	0.0038	-0.0007	-0.0006	-0.0006

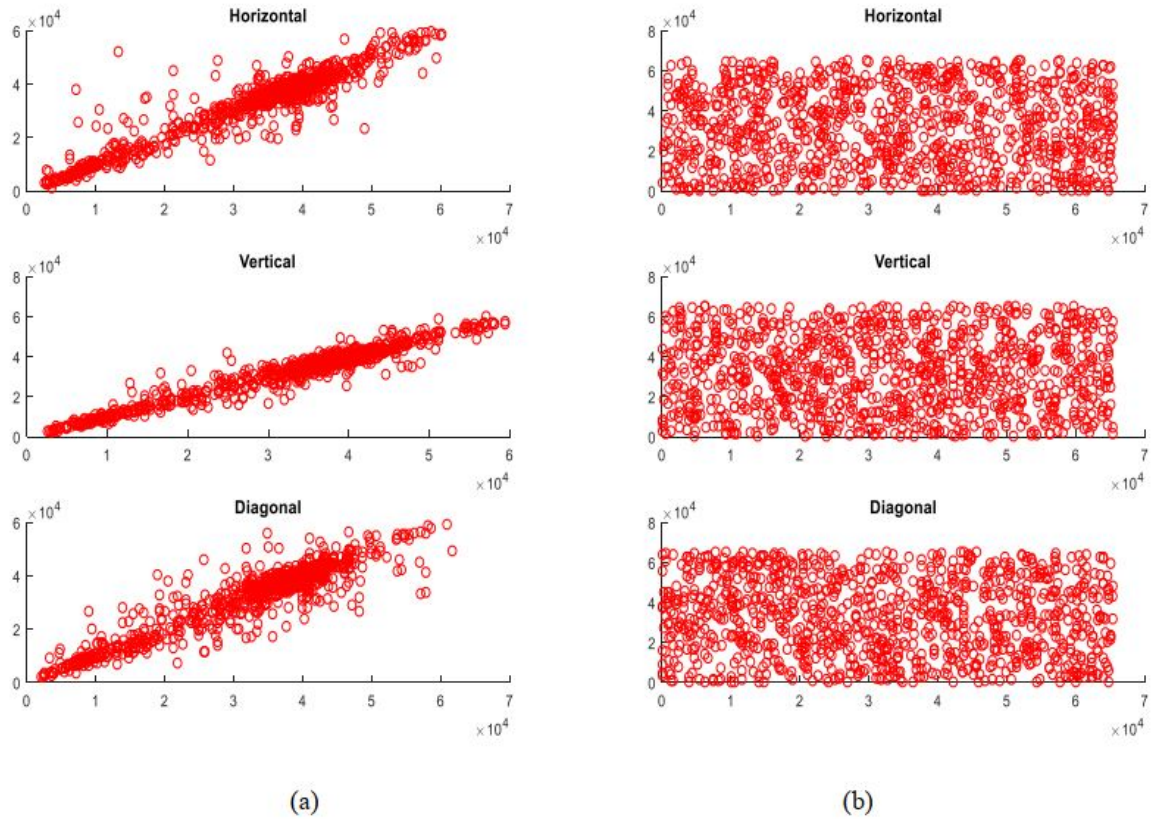


Fig. 4.9 *Corrélation des pixels adjacents dans différentes directions de Lena (256x256 pixels) (a) image originale, (b) image cryptée*

4.5.5 Analyse de la sensibilité aux clés

En plus du fait que le cryptosystème possède un espace clé adapté pour résister à l'attaque par force brute, le cryptosystème doit également être très sensible à leurs clés [66, 67], où même il n'y a qu'une variance ténue de 10^{-15}entre les clés de chiffrement et de déchiffrement entraîne

l'échec du déchiffrement. Pour évaluer la sensibilité de la clé de notre algorithme proposé, nous déchiffrons l'image de Lena avec une variance ténue dans l'un des paramètres de la clé secrète, les résultats sont présentés dans la figure 4.10. Nous notons que pour seulement un léger changement de 10^{-15}, l'image déchiffrée est complètement différente de l'image originale, ce qui prouve que le schéma suggéré est très sensible à ses clés.

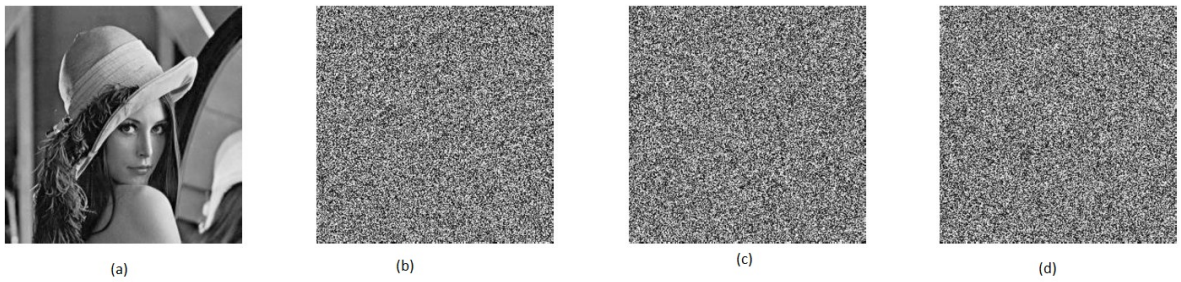


Fig. 4.10 *Résultats de la sensibilité de la clé de l'image décryptée : (a) avec la bonne clé ; (b) avec* $r_1 + 10^{-15}$ *; (c) avec mauvais* $x_1 + 10^{-15}$*, (d) avec mauvais* $x_2 + 10^{-15}$

4.5.6 La sensibilité des Tweaks

Pour vérifier la variabilité du schéma suggéré, la sensibilité des tweaks a été testée, où le dernier bit significatif du tweak est modifié dans le processus de décryptage avec la même clé secrète utilisée dans le processus de cryptage. La figure 4.11 montre les résultats de sensibilité au tweak du schéma proposé, où dans notre algorithme un petit changement dans les tweaks conduit à l'échec du décryptage. Par conséquent, le schéma proposé est si sensible à la minuscule modification du tweak.

4.5.7 Tests de NIST

Afin de confirmer le caractère aléatoire des images cryptées par le schéma suggéré, la version NIS Test des tests est utilisée. Le tableau 4.5 montre que toutes les images chiffrées ont réussi tous les tests NIST. Par conséquent, les images chiffrées qui ont été chiffrées par le schéma proposé ont de bonnes caractéristiques de caractère aléatoire.

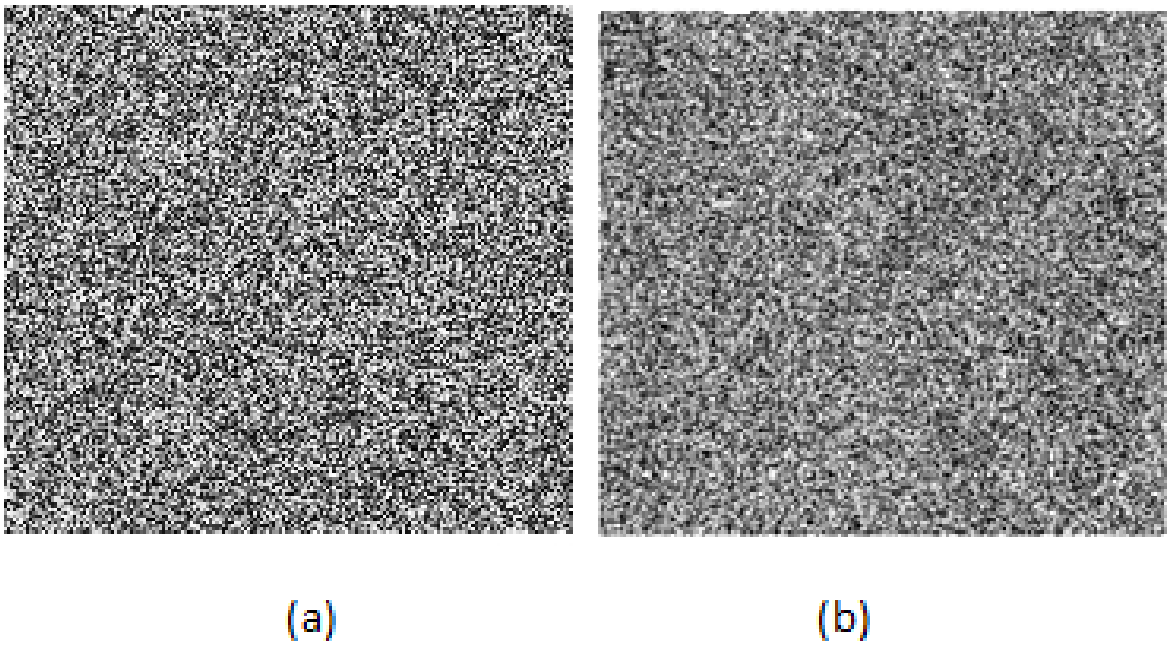


Fig. 4.11 *Résultats de la sensibilité Tweak. (a) image déchiffrée avec un mauvais T1, (b) Image déchiffrée avec un mauvais T2*

TABLE 4.5 *Tests de NIST.*

Test	Pepper	Lena	Boat	Baboon
FREQUENCY	0,424870	0,883921	0,939419	0,987234
Block frequency	0,228081	0,998861	0,419614	0,704799
RUNS	0,423341	0,505397	0,690625	0,866583
LONGEST RUNS	0,844824	0,844217	0,887254	0,048750
RANK TEST	0,784130	0,944003	0,212933	0,192916
FFT	0,287107	0,748072	0,600927	0,804313
Non-overlapping-templates	0,127988	0,967746	0,872085	0,490194
Overlapping-templates	0,184456	0,192844	0,938877	0,835474
Universal	0,651175	0,165172	0,064750	0,281020
Linear-complexity	0,719496	0,319321	0,137725	0,839870
Approximate entropy	0,189760	0,408787	0,230241	0,326120
CUMULATIVE SUMS	0,595743	0,981774	0,526378	0,991716
Random-excursions variant	0,599941	0,954384	0,746032	0,954384
Serial 1	0,743073	0,021625	0,267678	0,484260
Serial 2	0,795007	0,089326	0,088806	0,387108
RANDOM EXCURSIONS	0,927575	0,978737	0,611188	0,844386

TABLE 4.6 *Analyse de vitesse.*

Image	Notre méthode	Ref.[43]	Ref.[63]
Lena 256×256	0.34 s	0.36 s	1.21 s

4.5.8 Analyse de vitesse

Pour juger des performances de notre méthode, il existe un autre facteur qui influence la mesure de la vitesse de chiffrement qui est le facteur temps. L'environnement expérimental est MATLAB R2016a avec un processeur Intel (R) Core (TM) i3-5005 à 2,00 GHz et 4 Go de RAM sous Windows 10. Le temps de vitesse de chiffrement est illustré dans le tableau 4.6, où il montre que notre algorithme proposé a une performances de vitesse, il peut donc être utilisé dans une application réelle.

4.5.9 Attaque connue / choisie

Pour obtenir un schéma de chiffrement réussi, ce dernier doit résister aux types d'attaques classiques : le attaque texte clair connu, l'ttaque sur texte chiffre seul, Attaque texte clair choisi et les attaques par texte choisi. Parmi eux, l'attaque en texte clair choisie est l'attaque la plus puissante, généralement lorsqu'un cryptosystème résiste à cette attaque, il peut résister à trois autres attaques [34]. De nombreux schémas de cryptage d'images chiffrent une image en utilisant les mêmes clés secrètes, ce mécanisme rend le schéma moins protégé contre l'attaque en texte clair choisie . Le schéma de cryptage suggéré est très sensible aux modifications, de plus le processus de décryptage ne repose pas seulement sur les clés secrètes valides mais également sur les quatre Tweaks qui sont étroitement liés à l'image d'origine. Par conséquent, si vous chiffrez différentes images originales, leurs Tweaks sont également différents. Ainsi, le déchiffrement ne peut réussir que lorsque les attaquants ont les clés valides et les Tweaks appropriés pour chaque image ordinaire. De plus, les Tweaks dépendent de valeurs aléatoires pour chaque opération de cryptage, où si un attaquant tente de crypter une image deux fois, il obtiendra deux images chiffrées différentes comme le montre la figure 4.12. Cela prouve que le schéma suggéré peut résister au l'attaque en texte clair choisiet, il résistera ainsi aux autres attaques.

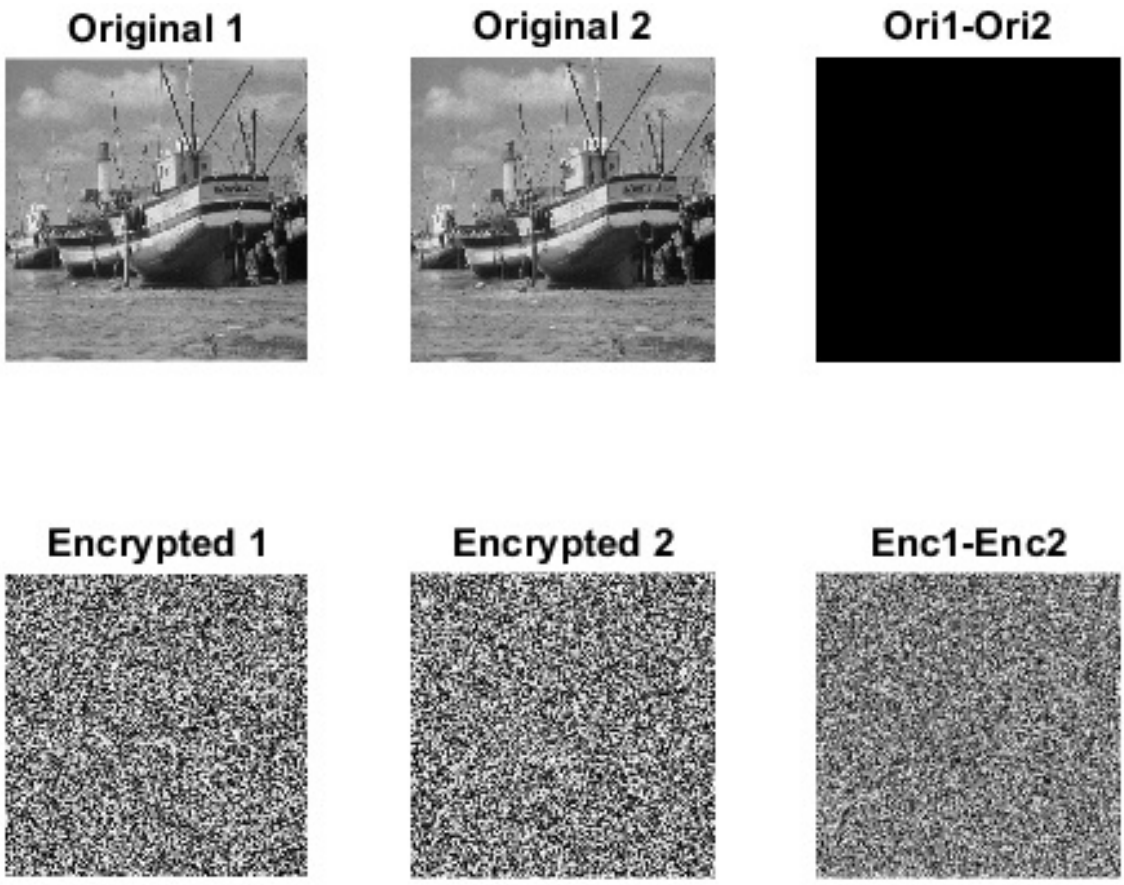


Fig. 4.12 *Le schéma proposé chiffre deux fois l'image du bateau utilisant le même ensemble de clés de sécurité*

4.6 Conclusion

Dans ce chapitre, une carte logistique classique améliorée est suggérée pour améliorer les caractéristiques chaotiques afin de protéger les images numériques lors de la transmission et du stockage. Les tests expérimentaux sur la conduite chaotique et la portée chaotique concernant la carte logistique améliorée en termes de test NIST, exposant de Lyapunov, bifurcation et comparaison avec la carte logistique classique, illustrent une meilleure performance chaotique. De plus, un nouvel algorithme de cryptage d'image ajustable basé sur une architecture de confusion-diffusion contenant deux tours de diffusion et de permutation est introduit. Les résultats des tests prouvent que l'algorithme proposé est simple, efficace et a une bonne exécution dans le cryptage d'image et la capacité de résister à plusieurs attaques.

Chapitre 5

Une nouvelle approche de cryptage des images couleur utilisant une combinaison de deux cartes chaotiques 1D

Sommaire

5.1 Introduction

Ce chapitre propose une carte chaotique améliorée en couplant deux cartes chaotiques existantes. Les tests de performance prouvent qu'il présente un comportement étroitement complexe et une gamme chaotique plus large que leurs cartes de graines. Un nouvel algorithme de chiffrement des images couleur utilisant la carte chaotique améliorée est proposé. Le schéma proposé est simple et très facile à mettre en œuvre pour le cryptage et le décryptage d'images.

La structure de ce chapitre se déroule comme suit. La section 2 décrit les systèmes chaotiques proposés. La section 3 présente le nouvel algorithme d'image proposé avec des détails. La section 4 présente les résultats expérimentaux. Enfin, la section 5 donne la conclusion.

5.2 Systèmes chaotiques

La suite logistique et la suite de la tente sont les suites chaotiques 1D les plus populaires, définies comme suit, qui ont une équation dynamique simple et classique avec des comportements chaotiques complexes, sont utilisées dans de nombreux domaines tels que les algorithmes de cryptage d'image[37] and [68] :

$$X_{n+1} = L(r, X_n) = rX_n(1 - X_n). \tag{5.1}$$

$$X_{n+1} = T(r, X_n) = \begin{cases} rX_n/2 & X_n < 0.5 \\ r(1 - X_n)/2 & X_n \geq 0.5 \end{cases} \tag{5.2}$$

Où (r) est le paramètre de contrôle et sa plage est r ∈ [0, 4]. Malheureusement, en s'appuyant sur le schéma de la bifurcation et de l'exposant de Lyapunov donné sur les figures 1 (a), 1 (b), 2 (a) et 2 (b), on peut voir que ces systèmes présentent des défauts tels que la plage limitée pour comportement chaotique et distribution inégale des séquences, où leurs plages chaotiques ne sont limitées que dans la plage [3.57, 4] pour la carte logistique et [2, 4] pour la carte des tentes. L'exposant de Lyapunov est défini comme une valeur pour l'évaluation quantitative de l'efficacité du chaos [45]. Si la valeur de l'exposant de Lyapunov est positive, la performance du chaos est bonne, cela implique une meilleure performance chaotique. Si le paramètre de contrôle r <3,57 pour la carte logistique et r <2 pour la carte tente, le diagramme des exposants de Lyapunov de cette carte est négatif, cela implique qu'ils n'ont pas de comportements chaotiques. Dans le schéma de cryptage, les opérations de diffusion et de permutation des pixels utilisent des séquences chaotiques générées. Ainsi, les séquences chaotiques de sortie non uniformes ont certains effets sur la distribution des pixels d'images cryptées et l'efficacité du schéma de cryptage. Comme on le sait, l'image cryptée a une forte corrélation avec sa clé de sécurité, par

conséquent, la nécessité d'utiliser un bon système de génération de clés de sécurité est requise. Pour résoudre ces problèmes, nous proposons un système chaotique amélioré en combinant ces cartes chaotiques comme suit :

- **The Tent-Tent system (TTS)**

La même carte de tente est combinée pour obtenir le TTS qui est présenté dans l'équation suivante :

$$X_{n+1} = (T(r, T(r, X_n)) \times a(k)) mod 1 = \begin{cases} ((r(rX_n/2)/2) \times a(k)) mod 1 & X_n < 0.5 \\ (r(1 - r(1 - X_n)/2)/2 \times a(k)) mod 1 & X_n \geq 0.5 \end{cases} \quad (5.3)$$

Où le paramètre r ∈[0,4] , $a(k) = 2^k$ et $8 \leq k \leq 20$

- **The Tent-Logistic system (TLS)**

La carte des tentes et les cartes logistiques sont combinées pour obtenir le TLS qui est présenté dans l'équation suivante :

$$X_{n+1} = (T(r, L(r, X_n)) \times a(k)) mod\ 1 = \begin{cases} ((r(rX_n(1 - X_n))/2) \times a(k)) mod\ 1 & X_n < 0.5 \\ (r(1 - rX_n(1 - X_n))/2 \times a(k)) mod\ 1 & X_n \geq 0.5 \end{cases} \quad (5.4)$$

Où le paramètre r ∈[0, 4], a (k) = 2^k et $8 \leq k \leq 20$ et a (k) considéré comme une fonction ajustable. Le paramètre k a une bonne conduite chaotique dans la plage de [8, 20]. La plage de valeurs de k a été prouvée au cours de l'expérience. Les séquences chaotiques générées sont assurées pour être dans la variété de [0, 1] à l'aide de l'opération «mod» Les figures 5.1.(e) , 5.1.(d) , 5.2.(e),et 5.2.(d) illustrent les diagrammes de bifurcation et Lyapunov exposant de TTS et TLS. Par rapport à leurs cartes de graines précédentes, TTS et TLS ont une gamme chaotique plus grande, leurs comportements chaotiques existent dans toute la gamme des paramètres de contrôle r et leurs séquences chaotiques ont une distribution uniforme dans [0, 4]. Par conséquent, le TTS et le TLS ont de meilleures performances chaotiques que leurs cartes de départ. Par conséquent, ils sont très adaptés à la cryptographie.

5.2.1 Propriétés aléatoire

Pour examiner le propriété aléatoire des nombres générés par le TLS et le TTS, le NIST SP 800-22 (2010) a été utilisé [61]. Le NIST SP 800-22 contient 15 tests statistiques. Un nombre

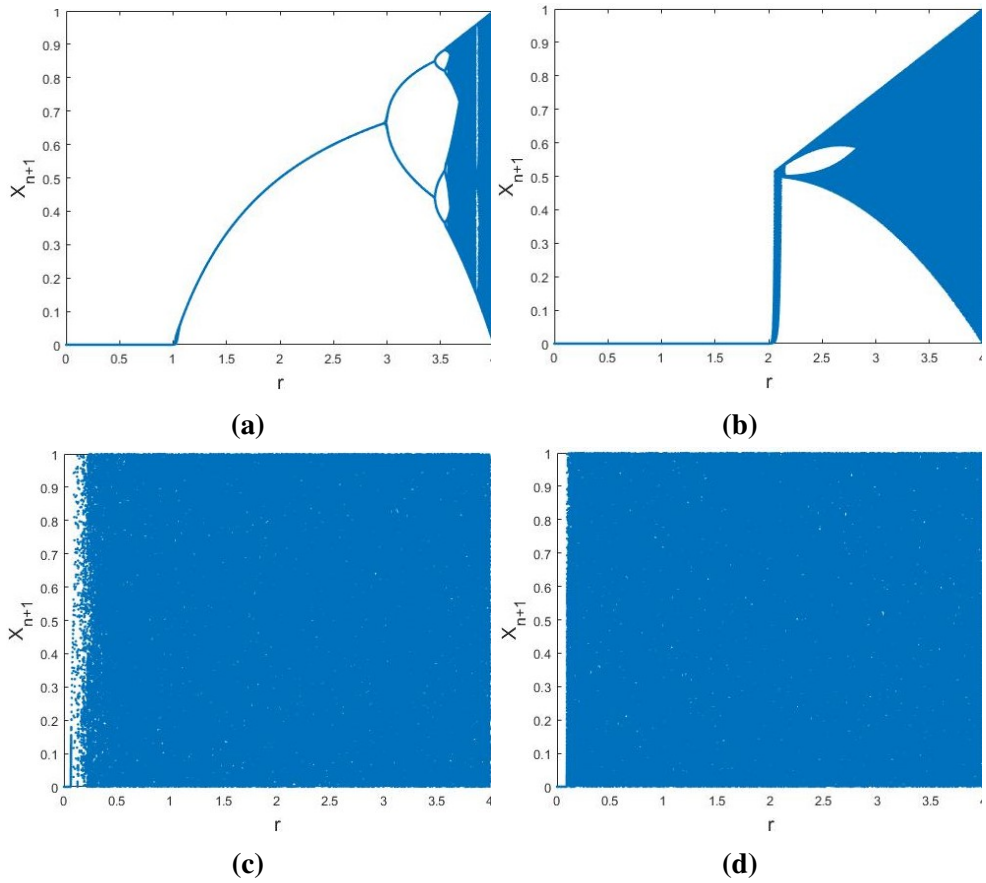


Fig. 5.1 *Diagramme de bifurcation de (a) logistique (b) tente (c) TLS avec k = 9 (d) TTS avec k = 9*

réel est produit par chaque test dans [0-1] qui est appelé P_{value}. Si P_{value} est supérieure à 0,01, les nombres générés sont aléatoires, sinon aucun aléatoire [8]. Le tableau 5.1 prouve que la valeur P est supérieure à 0,01 et que les séquences générées TTS et TLS réussissent tous les tests. Cela signifie que le système amélioré peut générer des séquences chaotiques avec un caractère aléatoire élevé.

5.3 Schéma de cryptage des images couleur proposé

Pour examiner les applications du système chaotique amélioré suggéré dans le cryptage d'images, une nouvelle approche pour le cryptage d'images couleur a été suggérée. La figure 5.3 illustre le schéma de principe du schéma proposé, où le schéma proposé a consisté en deux tours d'opérations de permutation et de diffusion. Les détails de l'algorithme sont décrits dans l'étape suivante :

- Étape 1 : L'image couleur originale $I_{N\times N\times 3}$ est lue, où (N, N) sont respectivement la hauteur et la largeur.

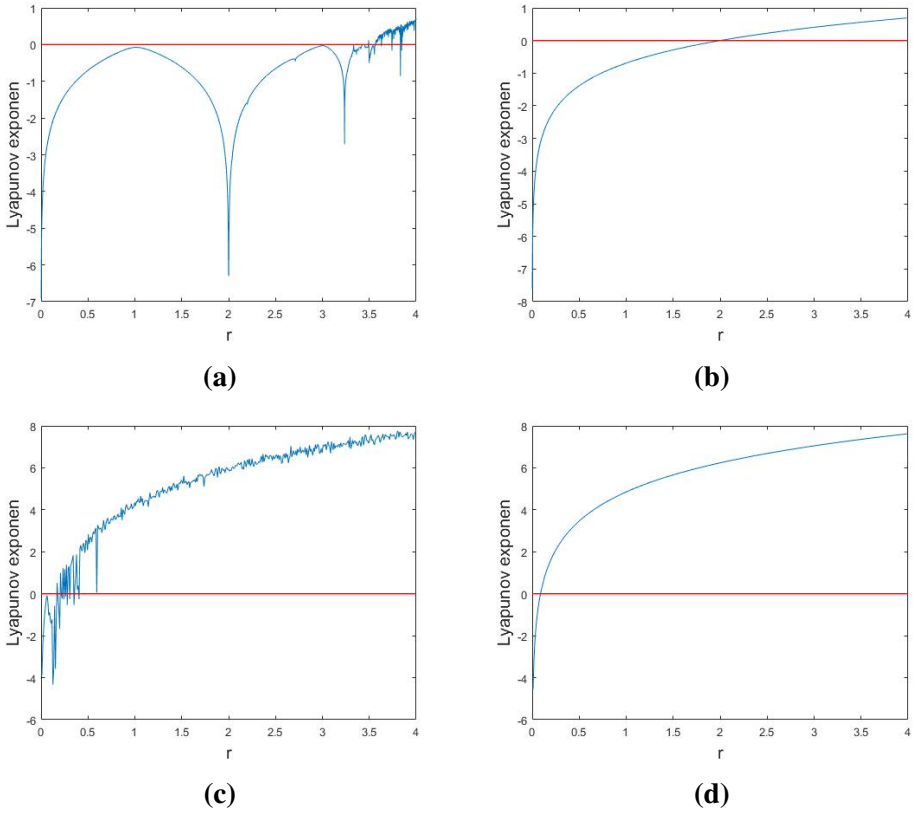


Fig. 5.2 *Diagramme exposant de Lyapunov, (a) logistique (b) tente (c) TLS avec k = 9 (d) TTS*

- Étape 2 : Nous avons divisé l'image couleur $I_{N\times N\times 3}$ en trois composantes couleurs R, G et B respectivement.
- Étape 3 : Nous avons choisi les valeurs initiales et les paramètres de contrôle des systèmes chaotiques qui seront utilisés en permutation et diffusion ($r_1 = 3.12$, $r_2 = 2.9$, $r3 = 1.6$, $r_4 = 3.88$,$r_5 = 3.18$, $x_{0,1} = 0.2$, $x_{0,2}0.58$, $x_{0,3} = 0.5$, $x_{0,4} = 0.8$, $x_{0,5} = 0.142$), qui sont considérées comme des clés secrètes.
  Les valeurs initiales et les paramètres de contrôle sont mis à jour par des relations : :

$$S_k = \sum_{i,j} O(i, j, k), k = 1, 2, 3. \tag{5.5}$$

$$S_k^0 = \frac{S_k}{N \times N \times 255}, k = 1, 2, 3. \tag{5.6}$$

  De plus, nous générons différentes clés à chaque itération, nous prenons S_4 et S_5 comme valeurs aléatoires dans [0, 1] en utilisant l'équation :

$$S_4 = rand(), S_5 = rand() \tag{5.7}$$

TABLE 5.1 *Les résultats des tests NIST-800-22 de LSS et TSS*

Test	P-value TTS	P-value TLS	Resultas
Frequency	0.2891	0.3231	Passed
Block Frequency	0.3479	0.4665	Passed
The Run Test	0.5386	0.6635	Passed
Longest Run of Ones in a block	0.7884	0,9575	Passed
Binary Matrix Rank	0.2609	0.8652	Passed
DFT Spectral	0.8905	0.3306	Passed
Non-Overlapping Template Matching	0.2088	0,5266	Passed
Overlapping Template Matching	0.1315	0.4490	Passed
Maurer's Universal Statistical Test	0,7578	0,0773	Passed
Linear Complexity	0.5123	0.4717	Passed
Serial Test	0.636339, 0.29141	0.6607 0.8724	Passed
Approximate Entropy	0.3698	0.2533	Passed
Cumulative Sums	0.2734	0.6029	Passed
Random Excursions(x=3)	0.6174	0.4597	Passed
Random Excursions Variant (x=-4)	0.9190	0.9187	Passed

$$\begin{cases} x'_{0,1} = (x_{0,1} + S^0_1 + S_4) mod\ 1 \\ x'_{0,2} = (x_{0,2} + S^0_2 - S_5) mod\ 1 \\ x'_{0,3} = (x_{0,3} + S^0_3 + S_4) mod\ 1 \\ x'_{0,4} = (x_{0,4} - S^0_3 - S_5) mod\ 1 \\ x'_{0,5} = (x_{0,5} - S^0_1 - S_4) mod\ 1 \end{cases} \tag{5.8}$$

$$\begin{cases} r'_1 = (r_1 + S^0_1 - S_4) mod\ 4 \\ r'_2 = (r_2 - S^0_2 + S_5) mod\ 4 \\ r'_3 = (r_3 + S^0_3 + S_4) mod\ 4 \\ r'_4 = (r_4 - S^0_3 - S_5) mod\ 4 \\ r'_5 = (r_5 - S^0_2 - S_4) mod\ 4 \end{cases} \tag{5.9}$$

Par conséquent, les valeurs initiales et les paramètres de contrôle mis à jour $(r'_1, r'_2, r'_3, r'_4, r'_5, x'_{0,1}, x'_{0,2}, x'_{0,3}, x'_{0,4}, x'_{0,5})$ sont liés au contenu de l'image originale $O_{N\times N\times 3}$ et ils sont également mis à jour à chaque itération.

- Étape 4 :Nous avons généré une séquence chaotique Z en utilisant les paramètres mis à jour et les valeurs initiales $(r'_5, x'_{0,5})$, où Z=(Z_1, $Z2$ ..... $Z_{N\times N\times 1}$) en utilisant l'équation (5.3), avec les valeurs initiales.

- Étape 5 : Z est convertie en séquences entières en utilisant l'équation (5.10).

$$Z(i) = floor(Z(i) \times 10^{15}) mod\ 256. \tag{5.10}$$

- Étape 5 : Afin de briser la corrélation intense entre les pixels voisins, nous avons proposé un nouvel algorithme de permutation qui peut mélanger simultanément les placements de pixels dans les orientations verticale et horizontale, contrairement à plusieurs algorithmes de mélange de pixels d'image qui exécutent le brouillage des pixels ligne par ligne et colonne par colonne. Par cette technique, l'algorithme proposé peut atteindre plusieurs avantages tels que : Il peut rapidement mélanger les emplacements de pixels dans l'image. Par conséquent, il peut détruire efficacement la corrélation robuste entre les pixels voisins. Nous supposons que l'image clair est chiffrée avec la taille $N \times N$: Les étapes de l'algorithme de permutation peuvent être expliquées comme suit :

  1. Deux séquences aléatoires, $V1$=(v_1, v_2....., $v_{n\times1}$) et $V2$=(v_1, v_2....., $v_{N\times1}$) de longueur $1 \times N$ sont générés en utilisant TTS, avec les valeurs initiales (r'_1, $x'_{0,1}$), (r'_2, $x'_{0,2}$) respectivement..
  2. V1 et V2 sont triés respectivement pour obtenir deux séquences d'indices, $V'1$ and $V'2$.
  3. Une autre matrice aléatoire, A de longueur $N \times 2$ est générée en utilisant TTS, avec les valeurs initiales (r'_3, $x'_{0,3}$).
  4. Chaque colonne de la matrice initiale est définie comme $V'1$ et obtient D comme indiqué sur la figure 5.3.
  5. Si A (i, 1) est supérieur à A (i, 2) pour i = 1 à H. la row_i de la matrice D est décalée vers la gauche de $V'1(i)$ positions, sinon row_i de la matrice D est décalé vers la droite de $V'1(i)$ positions. La figure 5.4 présente une explication numérique de la génération de $Q_{(4\times4)}$.
  6. Chaque indicateur de colonne j est initialisé par 1.
  7. Obtenez les pixels de l'image d'origine avec les emplacements $(1, Q_{1,j})$, $(2, Q_{2,j})$, $(N, Q_{N,j})$. Ces pixels sont connectés dans un anneau et les déplacent $Q_{N,j}$ positions dans le sens horaire si A (i ; 1) supérieur à A (i ; 2) dans l'image brouillée SH, sinon déplacez-les $Q_{1,j}$ positions dans le sens antihoraire.
  8. L'étape 7 est répétée jusqu'à l'étape 8 N - 1 fois avec j = 2 à N.

  Une explication numérique pour brouiller les pixels de l'image originale P de taille 4×4 est donnée à la figure 5.5, le processus détaillé de permutation peut être expliqué comme suit :

- Comme indiqué dans la colonne numéro un de Q est 4, 2, 1, 2, les pixels de l'image originale I dans le rouge avec les emplacements (1, 4), (2, 2), (3, 1), (4, 2) sont connectés en anneau, et nous les décalons$Q_{4,1} = 2$ positions dans le sens antihoraire. Ensuite, $SH_{1,4} = I_{3,1}$,$SH_{2,2} = I_{4,2}$, $SH_{3,1} = I_{1,4}$ et $SH_{4,2} = I_{2,2}$.
- Comme indiqué dans la colonne numéro deux de Q est 2, 1, 3, 1, les pixels de l'image originale P dans le bleu avec des emplacements (1, 2), (2, 1), (3, 3), (4, 1) sont connectés dans un anneau, et nous les décalons $Q_{4,2} = 1$ positions dans le sens antihoraire. Ensuite, $SH_{1,2} = I_{4,1}$,$SH_{2,1} = I_{1,2}$, $SH_{3,3} = I_{2,1}$ et $SH_{4,1} = I_{3,3}$.
- Comme indiqué dans la colonne numéro trois de Q est 1, 3, 4, 3, les pixels de l'image originale P dans le vert avec des emplacements (1, 1), (2, 3), (3, 4), (4, 3) sont connectés en anneau, et nous les décalons $Q_{4,3} = 3$ positions dans le sens horaire. Ensuite,$SH_{1,1} = I_{2,3}$,$SH_{2,3} = I_{3,4}$, $SH_{3,4} = I_{4,3}$ and $SH_{4,3} = I_{1,1}$.
- Comme indiqué dans la colonne numéro quatre de Q est 3, 4, 2, 4, les pixels de l'image originale P dans le jaune avec les emplacements (1, 3), (2, 4), (3, 2), (4, 4) sont connectés dans un anneau, et nous les décalons $Q_{4,4} = 4$ positions dans le sens antihoraire. Ensuite,$SH_{1,3} = I_{1,3}$,$SH_{2,4} = I_{2,4}$, $SH_{3,2} = I_{3,2}$ and $SH_{4,4} = I_{4,4}$.

- Étape 7 :L'image brouillée est convertie en 1D SH=(sh_1, sh_1 ..... $sh_{N\times 3N}$)

- Étape 8 : Afin d'obtenir un bon schéma de cryptage, il doit y avoir une excellente diffusion de pixels, ce qui signifie qu'un petit changement dans l'image clair entraîne une différence globale dans l'image cryptée, dans le schéma proposé, le pixel actuel est modifié en utilisant une valeur générée de manière aléatoire et le pixel diffusé précédemment. Le processus de diffusion 1D est conçu pour changer les valeurs des pixels par l'équation : (5.11) :

$$\begin{cases} C(i) = SH(1) \oplus Z(1). \; if \; i = 1; \\ C(i) = (SH(i) + Z(i)) mod \; 256 \oplus C(i-1) \; otherwise \end{cases} \tag{5.11}$$

  Où $\oplus$ est une opération XOR au niveau du bit ,$i = 1, 2, ..., W \times H$ and $C(i-1)$ le pixel diffusé précédent.

- Step(9) : Nous avons converti C en trois composants R, G, et B, puis nous avons composé l'image cryptée en couleur finale en utilisant ces trois composants. Les images en clair et leurs images cryptées sont représentées sur les figures 5.6 and figures.5.7

Lorsque nous voulons décrypter une image, nous appliquons le processus inverse de cryptage.

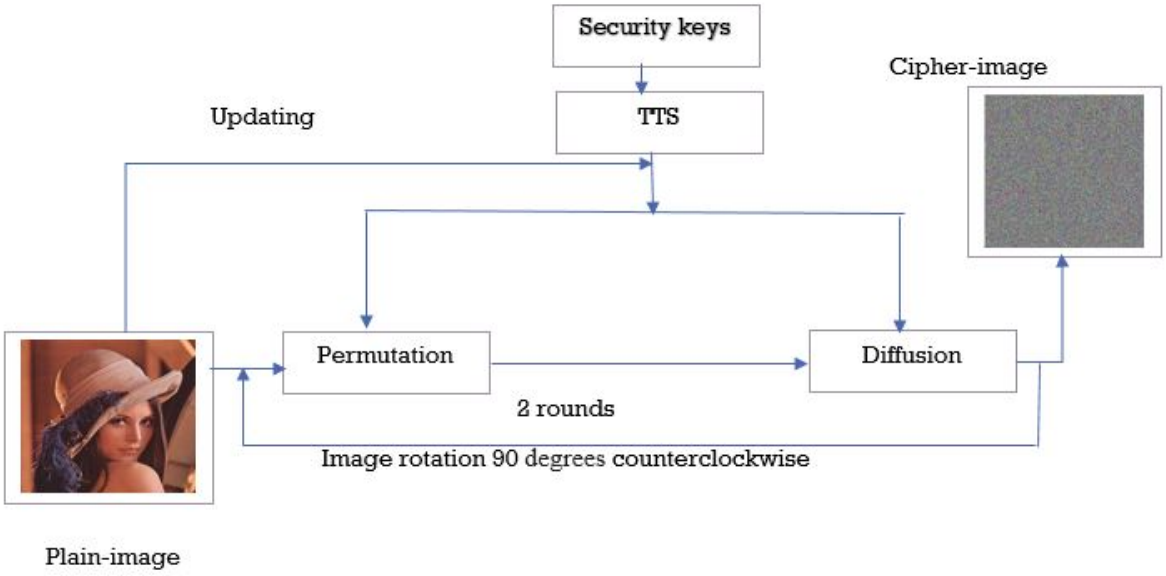


Fig. 5.3 *schéma de principe des processus de cryptage proposés.*

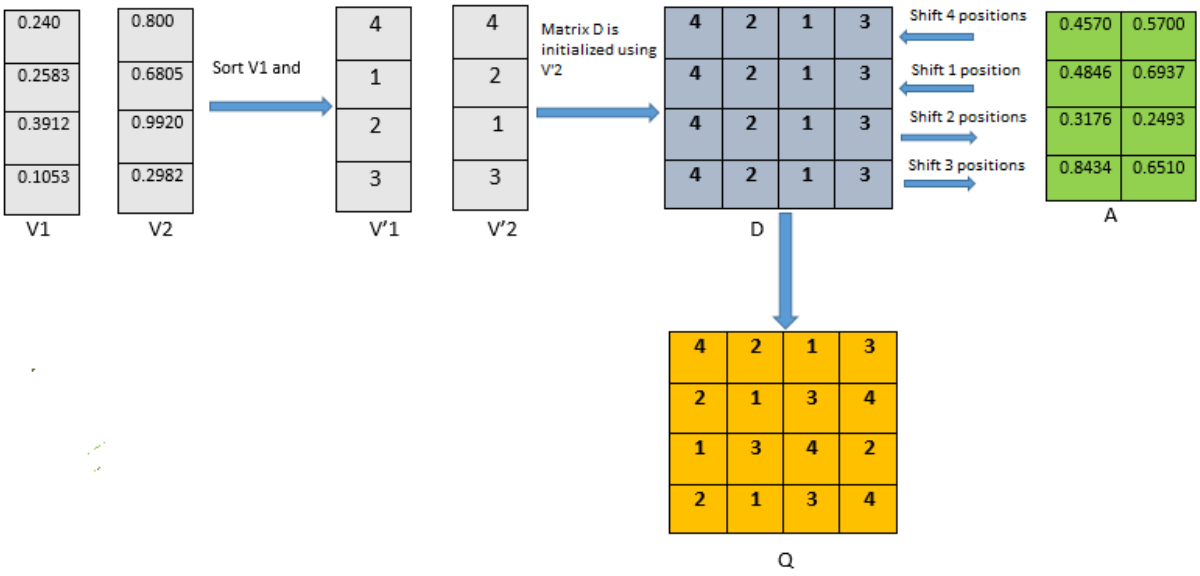


Fig. 5.4 *Un exemple d'une méthode de génération matrice Q*

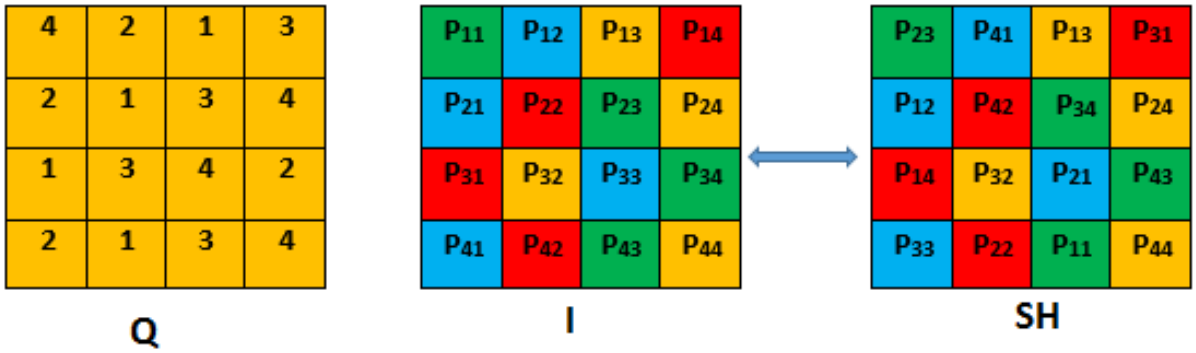


Fig. 5.5 *Un exemple numérique pour le permutation proposé.*

5.4 Résultats expérimentaux

Cette section discutera de l'efficacité et des performances de l'algorithme proposé. Un bon cryptosystème devrait avoir une résistance contre plusieurs attaques connues telles que les attaques statistiques et les attaques différentielles, qui sont destinées à obtenir le message d'origine ou la clé utilisée dans le processus de cryptage. Les résultats de la simulation sont présentés sur la figure 5.10. Pour découvrir la résistance et les performances de cet algorithme proposé contre certaines attaques, nous avons effectué des analyses telles que le nombre de taux de changement de pixels (NPCR), l'intensité de changement moyenne unifiée (UACI), l'analyse de corrélation parmi les pixels adjacents, et l'entropie

5.4.1 Analyse de l'entropie de l'information

A partir du tableau 5.2, nous pouvons voir que toutes les valeurs de l'entropie des images chiffrées sont assez proches de la valeur idéale. Par conséquent, la méthode de cryptage proposée est sécurisée sur l'entropie.

TABLE 5.2 *Analyse de l'entropiede l'information de diverses images*

Image	image originale	Encrypted Image cryptée	Ref[69]	Ref[70]
Lena	7.4767	7.9998	7.9997	7.9992
Baboon	7.7624	7.9997	7.9989	7.9991
Peppers	7.6698	7.9997	7.9984	7.9992

5.4.2 Analyse de l'espace clé

La sécurité des algorithmes de chiffrement dépend principalement de la clé. Un bon cryptosystème devrait avoir une longueur de clé suffisante, supérieure à 2^{100} pour contrer les attaques par force brute [71], [72]. Dans l'algorithme proposé, il y a huit clés pour les valeurs initiales et les paramètres de contrôle des fonctions chaotiques. Chaque séquence a besoin d'une valeur initiale et du paramètre de contrôle ($r_{0,1}$, $r_{0,2}$, $x_{0,1}$, $x_{0,2}$, $r_{0,3}$, $r_{0,4}$, , $x_{0,3}$, $x_{0,4}$), la précision des paramètres de contrôle et la valeur initiale est 10^{-15}. Ainsi, l'espace total de cette clé est de $10^{15\times8} = 10^{120}$, ce qui est supérieur à 2^{100}. Cela implique que cette clé est suffisante pour résister aux attaques par force brute.

5.4.3 Corrélation de deux pixels adjacents

Le tableau 5.3 montre les valeurs de corrélation des pixels adjacents verticalement, horizontalement et en diagonale des composantes de couleur des images originales et des images

cryptées, où la valeur de corrélation des composantes de couleur de l'image originale est très proche de la valeur de 1, tandis que la valeur de corrélation de l'image cryptée est très proche de zéro dans chaque direction. Cela signifie qu'il existe une faible corrélation entre les pixels adjacents dans les images chiffrées, la corrélation de l'image cryptée et de ses images originales est illustrée à la figure 5.9.

TABLE 5.3 *Coefficients de corrélation de deux pixels adjacents dans la Lena chiffrée et clair et comparaison avec différents cryptosystèmes*

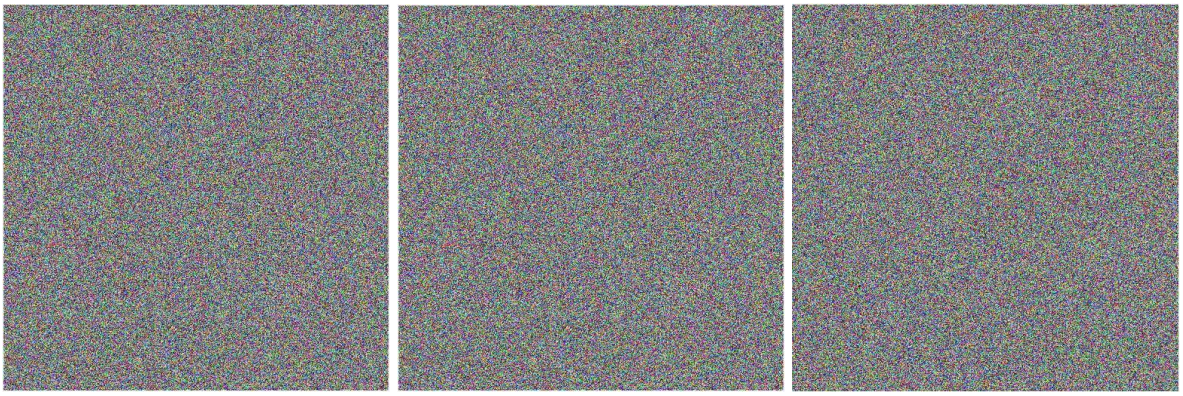
Channels	Directions	Original Image	Our Algorithm	Ref[73]	Ref[74]	Ref[75]
R channel	Horizontal	0.9790	-0.0019	0.0027	0.0019	0.0040
	Vertical	0.9782	-0.0021	-0.0013	0.0031	0.0003
	Diagonal	0.9593	-0.0018	0.0039	0.0007	0.0344
G channel	Horizontal	0.9476	-0.0066	0.0034	0.0054	0.0039
	Vertical	0.9515	-0.00174	-0.0034	0.001	0.0007
	Diagonal	0.9212	-0.0002	-0.0021	0.0017	-0.0043
B channel	Horizontal	0.9605	2.3978e-04	0.0046	0.0053	0.0038
	Vertical	0.9729	7.0294e-04	0.0038	0.0022	0.0003
	Diagonal	0.9497	-1.9311e-04	-0.0013	0.0007	-0.0438

(a) **(b)** **(c)**

Fig. 5.6 *Les images clair (a)Lena, (b) Baboon, (c)Pepper.*

5.4.4 L'analyse de l'histogramme

La figure 5.8 montre les histogrammes des composantes de couleur de l'image originale et cryptée, où il est observé que l'histogramme de l'image cryptée est uniforme et la distribution des valeurs de pixels sont égales, par opposition à l'image originale. Nous concluons que l'histogramme des composantes de couleur de l'image cryptée ne contient pas d'informations



(a) **(b)** **(c)**

Fig. 5.7 *Les images cryptées (a)Lena, (b) Baboon, (c) Pepper.*

statistiques sur la distribution des valeurs de pixels, ce qui rend l'algorithme proposé à l'abri des attaques statistiques.

5.4.5 Analyse de sensibilité clé

Lorsque le cryptosystème possède une clé suffisante pour résister aux attaques par force brute, le système de cryptage doit également être très sensible à la clé, où tout petit changement dans la clé conduit à un résultat différent des résultats qui ont été générés avant que la clé ne soit modifiée, ou le petit changement dans la clé conduit à l'impossibilité de récupérer l'image d'origine. Nous avons effectué une analyse pour garantir et démontrer la sensibilité du système de chiffrement pour un petit changement de valeur de clé. L'image Lena est chiffrée à l'aide de la clé($r_{0,1} = 3.12$, $r_{0,2} = 2.9$, $r_{0,3} = 1.6$, $r_{0,4} = 3.88$,$x_{0,1} = 0.2$, $x_{0,2}0.58$, $x_{0,3} = 0.5$, $x_{0,4} = 0.8$). La figure 5.11 (a) affiche l'image déchiffrée avec un petit changement dans le paramètre $x_{0,1} + 10^{-15}$ avec d'autres clés est prouvé. La figure 5.11 (b) montre l'image déchiffrée avec un petit changement dans le paramètre $x_{0,2} + 10^{-15}$ avec d'autres clés est prouvé. La figure 5.11 (c) montre l'image déchiffrée avec un petit changement dans le paramètre $x_{0,3} + 10^{-15}$ avec d'autres clés est prouvé. La figure 5.11 (d) montre l'image déchiffrée avec un petit changement dans le paramètre $r_{0,1} + 10^{-15}$ avec d'autres clés est prouvé. Les images résultantes indiquent que même avec le petit changement de 10^{-15}, l'image déchiffrée est complètement différente des images originales.

5.4.6 Attaque connue / choisie

Le cryptosystème peut être cryptanalysé par quatre types classiques d'attaques, attaque sur texte chiffré seul(COA), attaque texte clair choisi(CPA), attaque texte chiffré choisi(CCA), et attaque texte clair connu(KPA) [20], si un cryptosystème peut résister à un attaque texte clair choisi(CPA) et attaque texte clair connu(KPA) , donc il peut en résister à d'autres [65],[52].

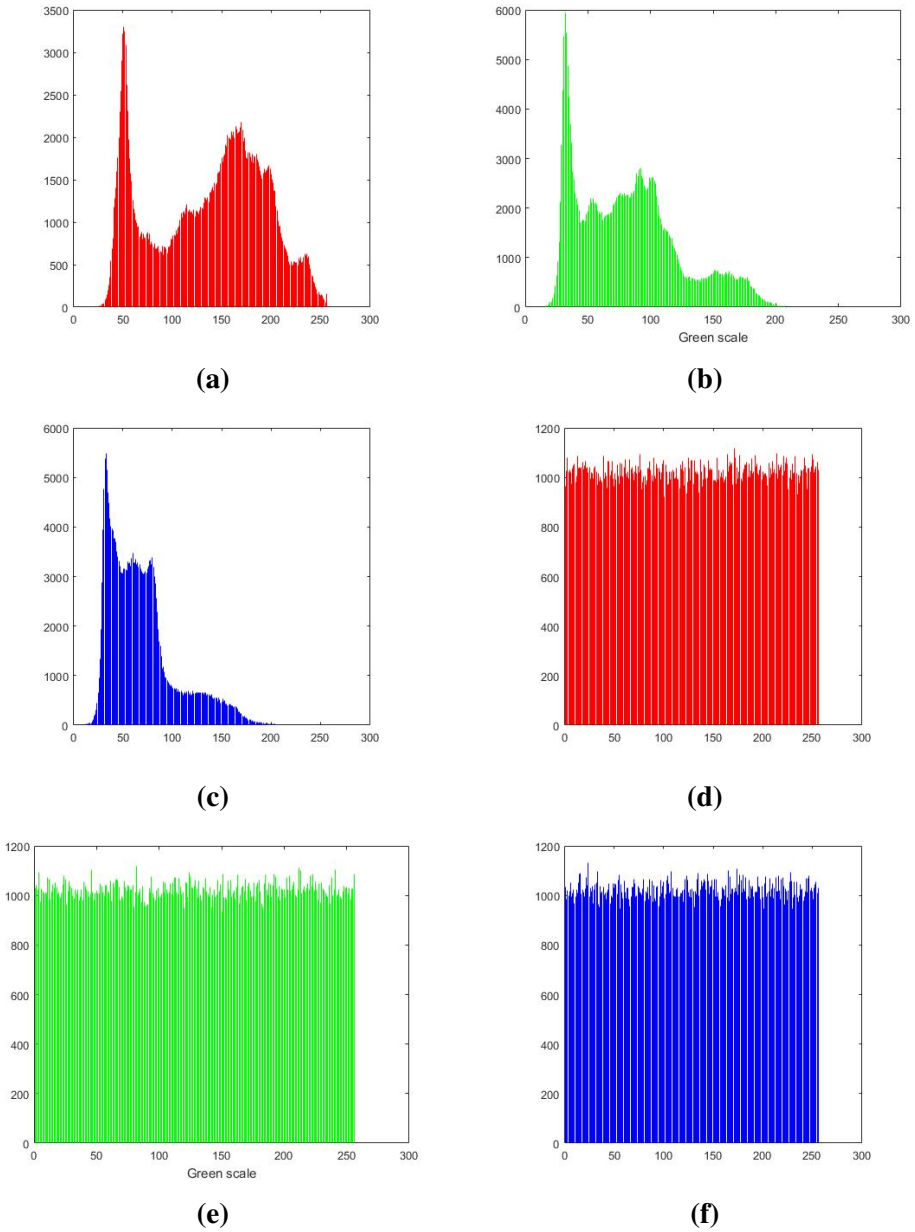
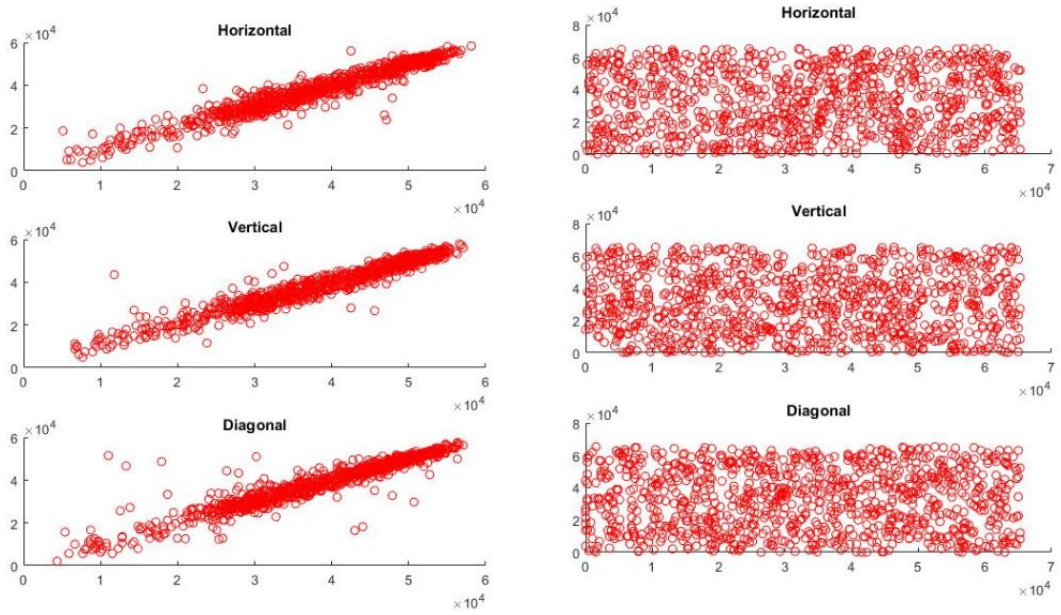
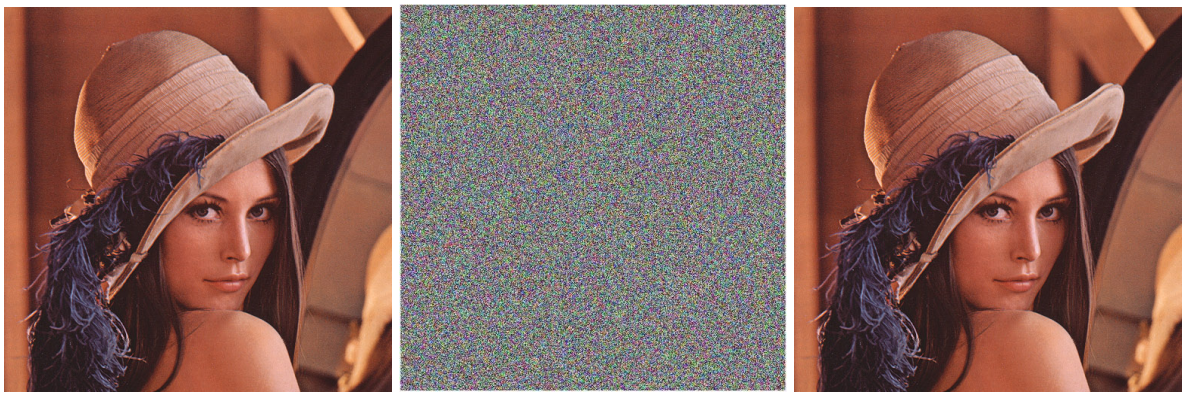


Fig. 5.8 *(a) - (c) Histogrammes des canaux R, G, B de l'image clair (d) - (f) Histogrammes des canaux R, G, B de l'image chiffrée*



(a) **(b)**

Fig. 5.9 *Analyse de corrélation de l'image Lena dans la composante R, (a) corrélation de la Lena originale dans toutes les directions (b) corrélation de la Lena cryptée dans toutes les directions.*



(a) **(b)** **(c)**

Fig. 5.10 *Résultats expérimentaux, (a) image originale (b) image cryptée (c) image décryptée*

Dans notre cryptosystème proposé, les paramètres de contrôle et les valeurs initiales du chaos sont mis à jour pour chaque image et à chaque fois, sur la base d'un nombre aléatoire qui dépend extrêmement de l'image ordinaire, donc différentes images conduisent à des clés différentes, donc si l'attaquant a un flux de clés avec certaines images en texte clair choisies, ces flux de clés ne peuvent pas être appliqués pour déchiffrer les images chiffrées cibles. Nous voyons une image cryptée complètement différente même si nous appliquons l'algorithme de cryptage deux fois à la même image avec une clé de cryptage fixe comme le montre la figure 5.12. Ainsi, notre cryptosystème proposé est capable de résister à l'attaque texte clair choisi(CPA) et attaque texte

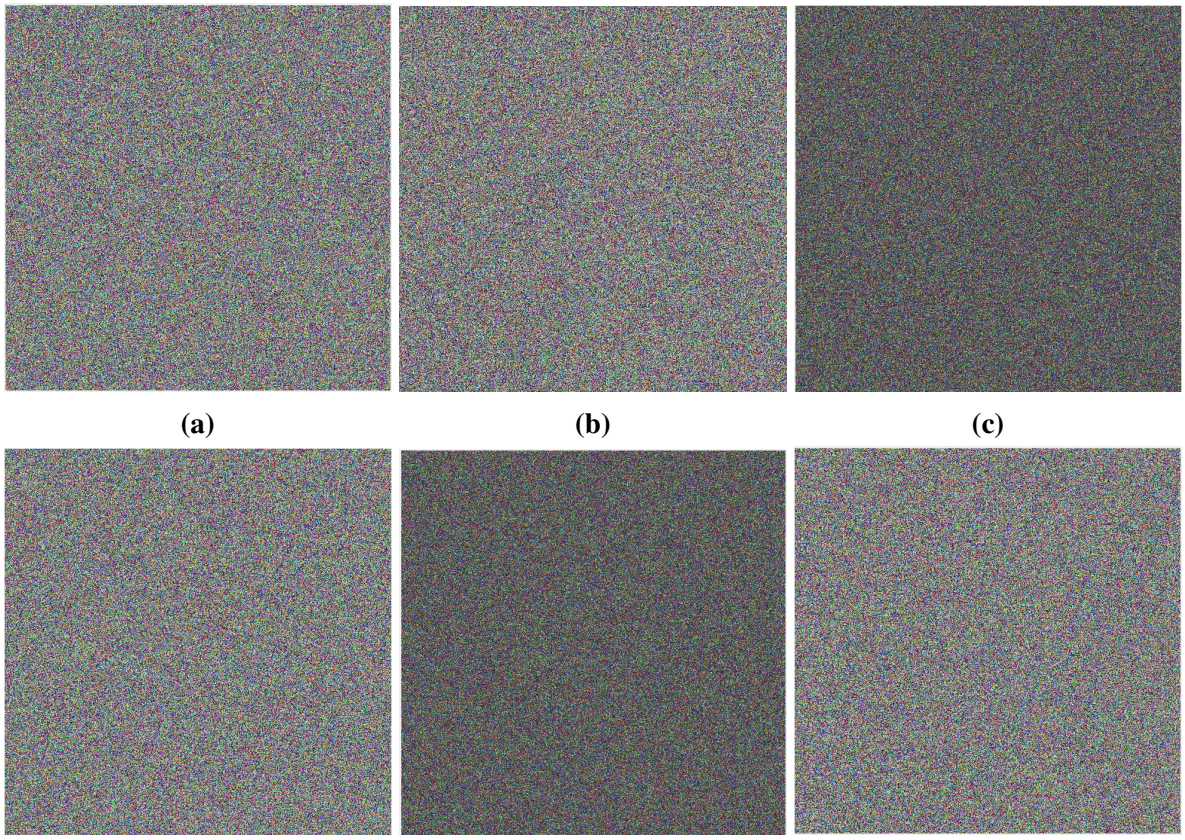


Fig. 5.11 *Images déchiffrées avec un petit changement de clé avec (a)* $x_{0,1}+10^{-15}$*,(b)* $x_{0,2}+10^{-15}$*, (c)* $x_{0,3}+10^{-15}$*, (d)* $r_{0,1}+10^{-15}$*, (e)* $r_{0,3}+10^{-15}$*, (f)* $r_{0,4}+10^{-15}$.

clair connu(KPA) [46].

5.4.7 Analyse d'attaque différentielle

Où les valeurs optimales du NPCR et de l'AUCI sont respectivement de 1 et 0,33 [51]. Nous avons chiffré une image claire, puis changé un pixel de l'image ordinaire à une position aléatoire. Les images chiffrées résultantes sont respectivement C et C'. Nous avons calculé le NPCR et l'AUCI entre les deux images cryptées. Les résultats montrés dans le tableau 5.4 sont dans les valeurs rapportées par Wu et al. [51, 76], l'algorithme proposé prouve donc son efficacité contre les attaques différentielles.

5.4.8 Analyse de vitesse

De plus, si le cryptosystème a une sécurité élevée contre toutes les attaques, il doit également atteindre une vitesse élevée dans le processus de cryptage. Dans cette section, nous avons comparé le temps pris pour le processus de cryptage de notre algorithme avec trois des algorithmes [70], [5] et [62]. Nous avons testé notre algorithme sous MATLAB 16 (R2016b) sur un PC avec

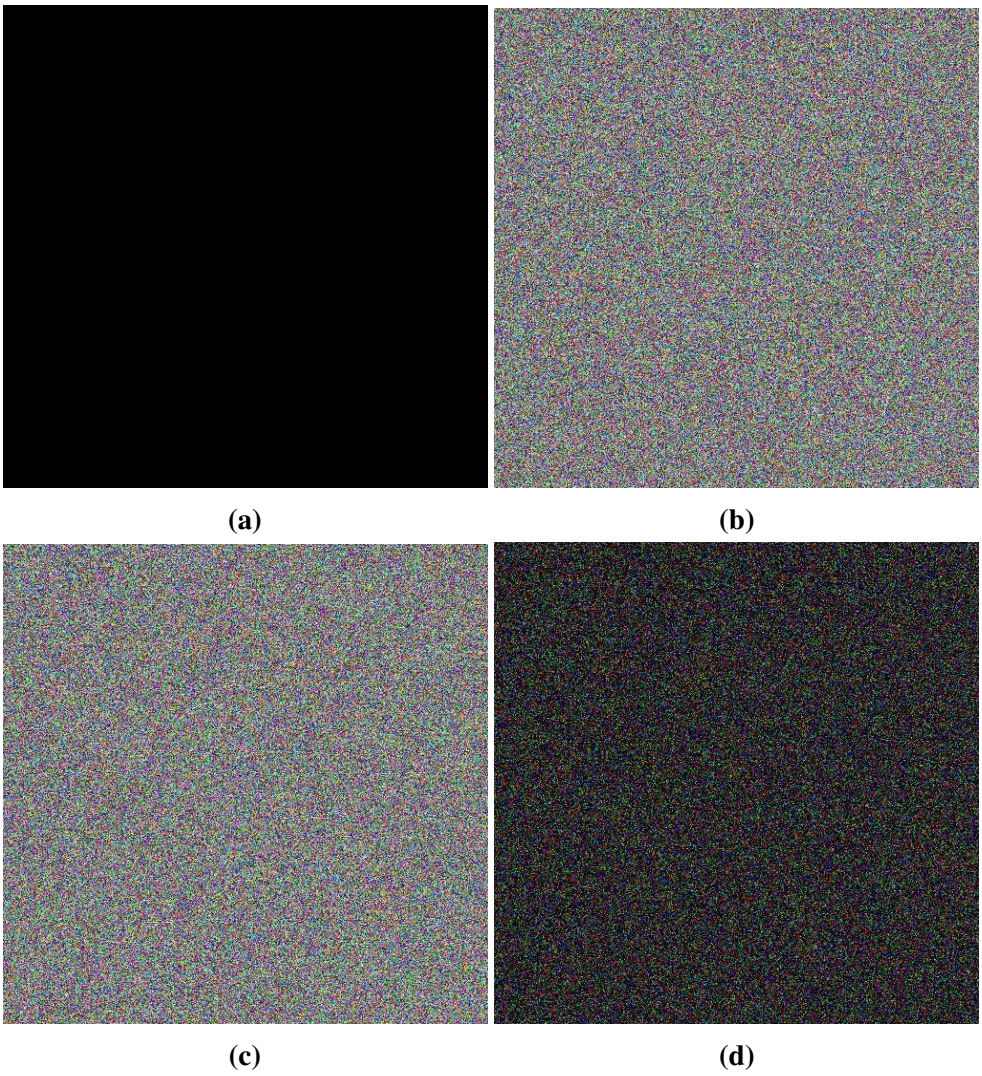


Fig. 5.12 *(a) L'image originale (b) La première image cryptée (c) La deuxième image cryptée (d) La différence de pixel à pixel*

TABLE 5.4 *NPCR et UACI de diverses images cryptées.*

	NPCR(%)			UACI(%)		
Image	R	G	B	R	G	B
Lena	99.63	99.59	99.62	33.51	33.47	33.48
Baboon	99.60	99.60	99.59	33.58	33.47	33.49
Peppers	99.61	99.61	99.61	33.40	33.57	33.40
Ref[70]	99.6425	99.6271	99.6387	33.3827	33.3775	33.3329
Ref[41]	99.65	99.62	99.58	33.45	33.38	33.59
Ref[42]	99.65	99.64	99.64	33.36	33.59	33.45

le système d'exploitation Windows 7, un processeur Intel (R) Core (TM) i3-5005 à 2,00 GHz et 4 Go de RAM. Le tableau 5.5 se réfère au temps pris dans le processus de cryptage pour l'image Lena avec une taille $256 \times 256 \times 3$. Nous concluons que notre algorithme peut convenir à des applications réelles.

TABLE 5.5 *Comparaison du temps de chiffrement de différents algorithmes*

Image name	Image size	Proposed algorithm (s)	Ref[70]	Ref[5]	Ref[62]
Lena	$256 \times 256 \times 3$	0.55145	0.6059	1.85	1.2612

5.5 Conclusion

Dans ce chapitre, nous avons introduit une carte chaotique améliorée en couplant deux cartes existantes. Les résultats des tests prouvent que ce dernier présente de meilleures performances de chaos que leurs cartes de graines. Ensuite, nous avons proposé une technique de cryptage des images couleur en utilisant la carte chaotique améliorée. La technique proposée définit un nouveau processus de permutation pour brouiller aléatoirement les pixels adjacents. L'efficacité de l'algorithme proposé est analysée en le comparant à certains travaux connexes et il est constaté que l'algorithme proposé peut résister à certaines attaques et peut fournir une sécurité satisfaisante.

Chapitre 6

Conclusion générale

Conclusion générale

Actuellement, les chercheurs accordent beaucoup d'attention à l'amélioration, au test et à l'application de la carte chaotique dans le domaine de la cryptographie pour protéger les données numériques, et en particulier les images, car elles ont plusieurs caractéristiques intrinsèques telles que la redondance en hauteur des données, la grande taille et la corrélation puissante entre pixels adjacents, qui sont différentes des données normales comme dans le texte. En plus, des caractéristiques temporelles nécessaires à la transmission d'images numériques en temps réel. Par conséquent, un cryptosystème à vitesse rapide et à haute sécurité devient nécessaire. L'utilisation d'algorithmes de chiffrement classiques tels que Rivest-Shamir-Adleman (RSA), la norme de chiffrement avancée (AES) et la norme de chiffrement des données (DES) sont devenues insuffisantes pour chiffrer des images numériques en temps réel.

Dans cette thèse, nous avons étudié, conçu et amélioré, des générateurs de séquences chaotiques, des crypto-systèmes image, et nous avons étudié et analysé leurs performances.

Nous avons présenté des outils communs et standard pour mesurer les performances des générateurs chaotiques, afin de quantifier et de comparer les propriétés des séquences chaotiques générées des algorithmes de chiffrement basés sur le chaos.

Nous avons ensuite proposé et étudié un crypto-système basé sur une amélioration de la carte quadratique (EQM), réalisée en utilisant la carte quadratique classique modifiée et en appliquant le modulaire arithmétique. Le système EQM présente d'excellentes performances telles qu'un meilleur exposant de Lyapunov et de plus grandes plages chaotiques par rapport à la carte quadratique classique et nous avons proposé une nouvelle approche de chiffrement des images couleur basée sur EQM, le schéma proposé est simple et hautement sécurisé et facile à mettre en œuvre pour le chiffrement et le déchiffrement des images couleur. L'évaluation de la sécurité du schéma est prouvée en le comparant à certains travaux connexes, il a été constaté que le travail proposé peut fournir un grand espace de clés, une sensibilité élevée aux clés et une corrélation plus faible. De plus, ce travail présente un histogramme uniforme et sécurise le cryptage des images contre les attaques statistiques, les attaques différentielles et les attaques de bruit (attaque connue / choisie).

Nous avons ensuite proposé et étudié une carte logistique classique améliorée, suggérée pour améliorer les caractéristiques chaotiques afin de protéger les images numériques lors de la transmission et du stockage. Les tests expérimentaux sur la conduite chaotique et la portée chaotique concernant la carte logistique améliorée en termes de test NIST, exposant de Lyapunov, bifurcation et comparaison avec la carte logistique classique, illustrent une meilleure performance chaotique. De plus, un nouvel algorithme de cryptage d'image ajustable basé sur une architecture de confusion-diffusion contenant deux tours de diffusion et de permutation a

été introduit. Les résultats des tests prouvent que l'algorithme proposé est simple, efficace et a une bonne exécution dans le cryptage d'image et la capacité de résister à plusieurs attaques.

Nous avons enfin proposé une carte chaotique améliorée en couplant deux systemes chaotique existantes. Les tests numériques prouvent que la technique proposée présente un comportement étroitement complexe et une gamme chaotique plus large que leurs cartes de semences, et une nouvelle approche de cryptage d'image couleur utilisant la carte chaotique améliorée a été suggérée. Le schéma proposé est basé sur la structure conventionnelle de diffusion de confusion qui contient un nouveau processus de permutation, il a conçu pour brouiller de manière aléatoire les pixels voisins. La performance et la mesure de la qualité du schéma proposé sont analysées en le comparant à certaines recherches existantes. Les résultats expérimentaux montrent que les schémas proposés présentent des performances supérieures à celles proposées dans beaucoup de travaux précédents.

Parmi les travaux futurs , nous pouvons citer :

- ✓ Implémenter les approches dans un système réel tel que les systèmes embarqués (FPGA).
- ✓ Nous étendrons nos travaux pour sécuriser le reste des données, pas seulement les images numériques.

Bibliographie

[1] H. Noura, *Conception et simulation des générateurs, crypto-systèmes et fonctions de hachage basés chaos performants*. PhD thesis, 2012.

[2] J. Wu, X. Liao, and B. Yang, "Color image encryption based on chaotic systems and elliptic curve elgamal scheme," *Signal Processing*, vol. 141, pp. 109–124, 2017.

[3] A. Jolfaei, X.-W. Wu, and V. Muthukkumarasamy, "On the security of permutation-only image encryption schemes," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 2, pp. 235–246, 2016.

[4] N. A. Hikal and M. M. Eid, "A new approach for palmprint image encryption based on hybrid chaotic maps," *Journal of King Saud University-Computer and Information Sciences*, 2018.

[5] X. Wu, B. Zhu, Y. Hu, and Y. Ran, "A novel color image encryption scheme using rectangular transform-enhanced chaotic tent maps," *IEEE Access*, vol. 5, pp. 6429–6436, 2017.

[6] M. Annaby, M. Rushdi, and E. Nehary, "Color image encryption using random transforms, phase retrieval, chaotic maps, and diffusion," *Optics and Lasers in Engineering*, vol. 103, pp. 9–23, 2018.

[7] C. Han, "An image encryption algorithm based on modified logistic chaotic map," *Optik*, 2018.

[8] C. Zhu and K. Sun, "Cryptanalyzing and improving a novel color image encryption algorithm using rt-enhanced chaotic tent maps," *IEEE Access*, vol. 6, pp. 18759–18770, 2018.

[9] S. Singh, "The science of secrecy from ancient egypt to quantum cryptography," 2000.

[10] O. Jallouli, *Chaos-based security under real-time and energy constraints for the Internet of Things*. PhD thesis, 2017.

[11] O. S. Faragallah, F. E. A. El-Samie, H. E. H. Ahmed, I. F. Elashry, M. H. Shahieen, E.-S. M. El-Rabaie, and S. A. Alshebeili, *Image encryption : a communication perspective*. CRC Press, 2013.

[12] M. AbuTaha, M. Farajallah, R. Tahboub, and M. Odeh, "Survey paper : cryptography is the science of information security," 2011.

[13] A. Beloucif, *Contribution à l'étude des mécanismes cryptographiques*. PhD thesis, Université de Batna 2, 2016.

[14] T. Bekkouche, *Développement et implémentation des techniques de cryptage des données basées sur les transformées discrètes*. PhD thesis, 2018.

[15] P. Guillot, "Auguste kerckhoffs et la cryptographie militaire," *Bibnum. Textes fondateurs de la science*, 2013.

[16] O. Goldreich *et al.*, "Foundations of cryptography–a primer," *Foundations and Trends® in Theoretical Computer Science*, vol. 1, no. 1, pp. 1–116, 2005.

[17] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*. CRC press, 2018.

[18] R. Verdult, *The (in) security of proprietary cryptography*. [Sl : sn], 2015.

[19] C. Merdjal, A. Merakchi, and I. Nini, "Cryptage d'image par un signal unidimensionnel quelconque," 2018.

[20] C. Li and K.-T. Lo, "Optimal quantitative cryptanalysis of permutation-only multimedia ciphers against plaintext attacks," *Signal processing*, vol. 91, no. 4, pp. 949–954, 2011.

[21] J. Wu, X. Liao, and B. Yang, "Image encryption using 2d hénon-sine map and dna approach," *Signal Processing*, vol. 153, pp. 11–23, 2018.

[22] D. Herbadji, N. Derouiche, A. Belmeguenai, N. Tahat, and S. Boumerdassi, "A new colour image encryption approach using a combination of two 1d chaotic map," *International journal of electronic security and digital forensics (IJESDF)*, 2019.

[23] M. Farajallah, *Chaos-based crypto and joint crypto-compression systems for images and videos*. PhD thesis, Universite de Nantes, 2015.

[24] K. A. K. Patro and B. Acharya, "Secure multi–level permutation operation based multiple colour image encryption," *Journal of information security and applications*, vol. 40, pp. 111–133, 2018.

[25] X. Wu, H. Kan, and J. Kurths, "A new color image encryption scheme based on dna sequences and multiple improved 1d chaotic maps," *Applied Soft Computing*, vol. 37, pp. 24–39, 2015.

[26] X. Lv, X. Liao, and B. Yang, "Bit-level plane image encryption based on coupled map lattice with time-varying delay," *Modern Physics Letters B*, vol. 32, no. 10, p. 1850124, 2018.

[27] X. Wang, S. Wang, Y. Zhang, and K. Guo, "A novel image encryption algorithm based on chaotic shuffling method," *Information Security Journal : A Global Perspective*, vol. 26, no. 1, pp. 7–16, 2017.

[28] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," tech. rep., Booz-allen and hamilton inc mclean va, 2001.

[29] M. Doulcier, *Test intégré de circuits cryptographiques*. PhD thesis, 2008.

[30] N. W. ABDERRAHIM, *Étude et conception d'un modèle chaotique dédié aux transmissions chiffrées*. PhD thesis.

[31] R. Li, Q. Liu, and L. Liu, "Novel image encryption algorithm based on improved logistic map," *IET Image Processing*, vol. 13, no. 1, pp. 125–134, 2018.

[32] N. Ramadan, H. E. H. Ahmed, S. E. Elkhamy, and F. E. A. El-Samie, "Chaos-based image encryption using an improved quadratic chaotic map," *American Journal of Signal Processing*, vol. 6, no. 1, pp. 1–13, 2016.

[33] Y. Wang, Z. Liu, J. Ma, and H. He, "A pseudorandom number generator based on piecewise logistic map," *Nonlinear Dynamics*, vol. 83, no. 4, pp. 2373–2391, 2016.

[34] D. Herbadji, A. Belmeguenai, N. Derouiche, and H. Liu, "Colour image encryption scheme based on enhanced quadratic chaotic map," *IET Image Processing*, 2019.

[35] I. Hussain, A. Anees, M. Aslam, R. Ahmed, and N. Siddiqui, "A noise resistant symmetric key cryptosystem based on s 8 s-boxes and chaotic maps," *The European Physical Journal Plus*, vol. 133, pp. 1–23, 2018.

[36] P. Verhulst, "La loi d'accroissement de la population," *Nouv. Mem. Acad. Roy. Soc. Belle-lettr. Bruxelles*, vol. 18, no. 1, 1845.

[37] Y. Zhou, L. Bao, and C. P. Chen, "A new 1d chaotic system for image encryption," *Signal processing*, vol. 97, pp. 172–182, 2014.

[38] X. Wang and D. Xu, "A novel image encryption scheme based on brownian motion and pwlcm chaotic system," *Nonlinear dynamics*, vol. 75, no. 1-2, pp. 345–353, 2014.

[39] S. Papadimitriou, T. Bountis, S. Mavroudi, and A. Bezerianos, "A probabilistic symmetric encryption scheme for very fast secure communication based on chaotic systems of difference equations," *International Journal of Bifurcation and Chaos*, vol. 11, no. 12, pp. 3107–3115, 2001.

[40] M. L. Sahari and I. Boukemara, "A pseudo-random numbers generator based on a novel 3d chaotic map with an application to color image encryption," *Nonlinear Dynamics*, vol. 94, no. 1, pp. 723–744, 2018.

[41] C. Pak and L. Huang, "A new color image encryption using combination of the 1d chaotic map," *Signal Processing*, vol. 138, pp. 129–137, 2017.

[42] H. Liu, A. Kadir, and X. Sun, "Chaos-based fast colour image encryption scheme with true random number keys from environmental noise," *IET Image Processing*, vol. 11, no. 5, pp. 324–332, 2017.

[43] Z. Hua, Y. Zhou, C.-M. Pun, and C. P. Chen, "2d sine logistic modulation map for image encryption," *Information Sciences*, vol. 297, pp. 80–94, 2015.

[44] D. Arroyo, J. Diaz, and F. Rodriguez, "Cryptanalysis of a one round chaos-based substitution permutation network," *Signal Processing*, vol. 93, no. 5, pp. 1358–1364, 2013.

[45] C. Pak, K. An, P. Jang, J. Kim, and S. Kim, "A novel bit-level color image encryption using improved 1d chaotic map," *Multimedia Tools and Applications*, pp. 1–16, 2018.

[46] R. Parvaz and M. Zarebnia, "A combination chaotic system and application in color image encryption," *Optics & Laser Technology*, vol. 101, pp. 30–41, 2018.

[47] D. Herbadji, N. Derouiche, A. Belmeguenai, T. Bekkouche, A. Labiad, M. Lashab, and A. Herbadji, "A new image encryption scheme using an enhanced logistic map," in *2018 International Conference on Applied Smart Systems (ICASS)*, pp. 1–6, IEEE, 2018.

[48] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International journal of bifurcation and chaos*, vol. 16, no. 08, pp. 2129–2151, 2006.

[49] C. Cao, K. Sun, and W. Liu, "A novel bit-level image encryption algorithm based on 2d-licm hyperchaotic map," *Signal Processing*, vol. 143, pp. 122–133, 2018.

[50] V. M. Silva-García, R. Flores-Carapia, C. Rentería-Márquez, B. Luna-Benoso, C. A. Jiménez-Vázquez, and M. D. González-Ramírez, "Cipher image damage and decisions in real time," *Journal of Electronic Imaging*, vol. 24, no. 1, p. 013012, 2015.

[51] Y. Wu, J. P. Noonan, and S. Agaian, "Npcr and uaci randomness tests for image encryption," *Cyber journals : multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications (JSAT)*, pp. 31–38, 2011.

[52] X. Wang, L. Teng, and X. Qin, "A novel colour image encryption algorithm based on chaos," *Signal Processing*, vol. 92, no. 4, pp. 1101–1108, 2012.

[53] M. Dridi, M. A. Hajjaji, B. Bouallegue, and A. Mtibaa, "Cryptography of medical images based on a combination between chaotic and neural network," *IET Image Processing*, vol. 10, no. 11, pp. 830–839, 2016.

[54] J. S. Khan and J. Ahmad, "Chaos based efficient selective image encryption," *Multidimensional Systems and Signal Processing*, vol. 30, no. 2, pp. 943–961, 2019.

[55] R. R. Kumar and M. B. Kumar, "A new chaotic image encryption using parametric switching based permutation and diffusion.," *ICTACT Journal on Image & Video Processing*, vol. 4, no. 4, 2014.

[56] H. Zhu, Y. Zhao, and Y. Song, "2d logistic-modulated-sine-coupling-logistic chaotic map for image encryption," *IEEE Access*, 2019.

[57] Z. Hua, B. Zhou, and Y. Zhou, "Sine chaotification model for enhancing chaos and its hardware implementation," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 2, pp. 1273–1284, 2018.

[58] Z. Hua, B. Zhou, and Y. Zhou, "Sine-transform-based chaotic system with fpga implementation," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 3, pp. 2557–2566, 2017.

[59] D. Herbadji, N. Derouiche, A. Belmeguenai, A. Herbadji, and S. Boumerdassi, "A tweakable image encryption algorithm using an improved logistic chaotic map," *Traitement du Signal*, vol. 36, no. 5, pp. 407–417, 2019.

[60] M. Liskov, R. L. Rivest, and D. Wagner, "Tweakable block ciphers," in *Annual International Cryptology Conference*, pp. 31–46, Springer, 2002.

[61] F. Özkaynak, "A novel method to improve the performance of chaos based evolutionary algorithms," *Optik-International Journal for Light and Electron Optics*, vol. 126, no. 24, pp. 5434–5438, 2015.

[62] X.-Y. Wang, Y.-Q. Zhang, and Y.-Y. Zhao, "A novel image encryption scheme based on 2-d logistic map and dna sequence operations," *Nonlinear Dynamics*, vol. 82, no. 3, pp. 1269–1280, 2015.

[63] Z. Hua and Y. Zhou, "Design of image cipher using block-based scrambling and image filtering," *Information Sciences*, vol. 396, pp. 97–113, 2017.

[64] M. Zhang and X. Tong, "A new algorithm of image compression and encryption based on spatiotemporal cross chaotic system," *Multimedia Tools and Applications*, vol. 74, no. 24, pp. 11255–11279, 2015.

[65] N. B. Slimane, N. Aouf, K. Bouallegue, and M. Machhout, "A novel chaotic image cryptosystem based on dna sequence operations and single neuron model," *Multimedia Tools and Applications*, pp. 1–27, 2018.

[66] S. F. Abbasi, J. Ahmad, J. S. Khan, M. A. Khan, and S. A. Sheikh, "Visual meaningful encryption scheme using intertwinning logistic map," in *Science and Information Conference*, pp. 764–773, Springer, 2018.

[67] J. S. Khan, J. Ahmad, S. F. Abbasi, S. K. Kayhan, *et al.*, "Dna sequence based medical image encryption scheme," in *2018 10th Computer Science and Electronic Engineering (CEEC)*, pp. 24–29, IEEE, 2018.

[68] Y.-R. Bai, D. Baleanu, and G.-C. Wu, "A novel shuffling technique based on fractional chaotic maps," *Optik*, vol. 168, pp. 553–562, 2018.

[69] B. Yang and X. Liao, "A new color image encryption scheme based on logistic map over the finite field z n," *Multimedia Tools and Applications*, Jan 2018.

[70] W. Liu, K. Sun, Y. He, and M. Yu, "Color image encryption using three-dimensional sine icmic modulation map and dna sequence operations," *International Journal of Bifurcation and Chaos*, vol. 27, no. 11, p. 1750171, 2017.

[71] S. M. Seyedzadeh and S. Mirzakuchaki, "A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map," *Signal processing*, vol. 92, no. 5, pp. 1202–1215, 2012.

[72] M. Zarebnia, H. Pakmanesh, and R. Parvaz, "A fast multiple-image encryption algorithm based on hybrid chaotic systems for gray scale images," *Optik*, vol. 179, pp. 761–773, 2019.

[73] S. Huang, Huiqing and, "Colour image encryption based on logistic mapping and double random-phase encoding," *IET Image Processing*, vol. 11, no. 4, pp. 211–216, 2016.

[74] A. Y. Niyat, M. H. Moattar, and M. N. Torshiz, "Color image encryption based on hybrid hyper-chaotic system and cellular automata," *Optics and Lasers in Engineering*, vol. 90, pp. 225–237, 2017.

[75] R. Wei, X. Li, and Q.-H. Wang, "Double color image encryption scheme based on off-axis holography and maximum length cellular automata," *Optik-International Journal for Light and Electron Optics*, vol. 145, pp. 407–417, 2017.

[76] S. Dhall, S. K. Pal, and K. Sharma, "A chaos-based probabilistic block cipher for image encryption," *Journal of King Saud University-Computer and Information Sciences*, 2018.