MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE FERHAT ABBAS –SETIF 1-

UFAS (ALGERIE)

THÈSE

Présentée à la faculté de technologie

Département d'Electronique

Pour l'obtention du diplôme de

Doctorat 3ème cycle (LMD)

Domaine : Sciences et technologie

Option : Traitement du signal

Par :

AZOUG SEIF EDDINE

Thème

Développement et implémentation des techniques de cryptage des signaux image et vidéo

Soutenue le **26/05/2016** devant le jury composé de :

Pr A. KHELLAF	Prof à l'université de Sétif 1	Président
Pr S. BOUGUEZEL	Prof à l'université de Sétif 1	Rapporteur
Dr N. AMARDJIA	M.C.A à l'université de Sétif 1	Examinateur
Pr D. CHIKOUCHE	Prof à l'université de M'Sila	Examinateur
Pr R. BENZID	Prof à l'université de Batna	Examinateur

A mes très chers parents, qu'Allah vous garde

A mes frères et sœurs

A tous mes amis

Remerciements

Je remercie mon directeur de thèse, Professeur Saad BOUGUEZEL pour son soutien, ses remarques pertinentes, et ses conseils durant la réalisation de ce travail.

Je remercie également le Professeur Dietmar SAUPE et le Professeur Chengqing Li d'avoir accepté de m'accueillir au sein de leur groupe de travail « Multimedia Signal Processing » dans le cadre d'un stage de courte durée.

Je remercie Monsieur Abdelhafid KHELLAF, Professeur à l'université de Sétif, d'avoir accepté d'être président du jury.

Je remercie Monsieur Nourredine AMARDJIA, Maître de Conférences à l'université de Sétif, d'avoir accepté d'être membre du jury.

Je remercie Monsieur Djamel CHIKOUCHE, Professeur à l'université de M'sila, d'avoir accepté d'être membre du jury.

Je remercie Monsieur Redha BENZID, Professeur à l'université de Batna, d'avoir accepté d'être membre du jury.

Enfin, je tiens à exprimer toute ma gratitude à mes parents qui ont été toujours là pour moi, et à mes frères, et à mes sœurs qui m'ont toujours soutenu tout au long de ces années.

Liste des figures

Liste des tableaux

Liste des acronymes

2D	Deux dimensions / bidimensionnelle.
CCD	Charge Coupled Device. Capteur photographique.
CIF	Common Intermediate Format. Format d'image standard.
CPU	Central processing unit. Unité centrale de traitement.
CRPM	Chaotic Random Phase Mask. Masque chaotique de phases aléatoires.
DRPE	Double Random Phase Encoding.
EQM	Erreur Quadratique Moyenne.
PLCM	Piecewise Linear Chaotic Map. Suite chaotique linéaire par morceaux.
PSNR	Peak Signal-to-Noise Ratio. Rapport signal/bruit de crête.
ROP	Réciproque-Orthogonale Paramétrique
SLM	Spatial Light Modulator. Modulateur spatial de lumière.
TF	Transformée de Fourier
TFR	Transformée de Fourier fractionnaire
TFRD	Transformée de Fourier fractionnaire discrète
TFRDR	Transformée de Fourier fractionnaire discrète réelle
TWH	Transformée de Walsh-Hadamard
XOR	Fonction logique OU exclusif

Table des matières

Chapitre 3: Proposition d'un cryptage numérique collectif d'images basée sur la transformée ROP

Chapitre 4 : Proposition d'un nouveau prétraitement non linéaire pour le cryptage d'images

Chapitre 5 : Proposition d'une nouvelle méthode de cryptage des séquences d'images vidéo

Introduction générale

Introduction générale

Nous vivons aujourd'hui l'ère du tout numérique où l'échange des données multimédias est en perpétuelle évolution grâce à la vulgarisation des systèmes de communication comme Internet. En parallèle, cela a engendré un besoin réel de protéger la confidentialité des données. Ce besoin est dû essentiellement au fait que l'homme a toujours cherché à préserver la confidentialité de ses secrets et de ses correspondances en utilisant des techniques de cryptage ou de chiffrement qui permettent de transformer un message compréhensible en un message incompréhensible, appelé message crypté, en utilisant une clé secrète. Seulement celui qui détient cette clé pourra déchiffrer correctement le message crypté.

Le développement des techniques de cryptage est une discipline à part entière qui appartient au domaine de la cryptographie qui est l'une des branches de la cryptologie ou la science du secret [1]. En effet, l'utilisation de la cryptographie n'est pas récente et remonte assez loin dans le temps. L'empereur Jules César utilisait pour ses communications secrètes un cryptage primitif mais efficace à cette époque-là [1]. Au fil du temps, l'utilisation de la cryptographie s'est développée et a eu un poids stratégique majeur sur la balance de certains conflits ou guerres comme la machine Enigma utilisée par les Allemands avec leurs communications secrètes pendant la deuxième guerre mondiale [2]. L'utilisation de la cryptographie s'est généralisée et n'est plus du ressort exclusif des organes gouvernementaux ou militaires. Aujourd'hui, les techniques de cryptage sont utilisées dans le domaine commercial, mais aussi dans le domaine de la protection de la vie privée [2].

Les techniques traditionnelles [3], [4] du cryptage numérique sont conçues seulement pour crypter des données binaires bit par bit peu importe le type des données. Leurs utilisation directe sur des données multimédias telles que les signaux images ou vidéo s'avère parfois difficile et lente [5], [6], car une image possède une grande redondance ainsi qu'une forte corrélation entre ses pixels. De plus, une image ou une séquence vidéo a une taille considérablement volumineuse. Par conséquent, il est très souhaitable d'explorer de nouvelles méthodes de cryptage appropriées aux signaux images et vidéos.

Vu les nombreuses applications des transformées mathématiques et leurs importance dans le domaine du traitement d'image, Javidi et al. ont développé la fameuse méthode de cryptage d'images DRPE [7], ou « Double Random Phase Encoding » en anglais, en utilisant la transformée de Fourier. Cette méthode peut être implémentée optiquement ou

numériquement. Elle consiste à multiplier une image par un masque de phases aléatoires dans le domaine spatial et par un autre masque de phases aléatoires dans le domaine de la transformée de Fourier. Ce dernier masque est considéré comme une clé secrète de cryptage et de décryptage.

Du fait de son efficacité et de sa robustesse, la méthode DRPE n'a pas tardée à susciter l'intérêt des chercheurs, car elle a été largement exploitée dans le domaine des transformées paramétriques [8]-[49]. Ces transformées sont des généralisations des transformées standards obtenues par l'introduction d'un ou plusieurs paramètres indépendants dans leurs noyaux (kernels). Cette paramétrisation permet d'avoir plusieurs représentations du signal dans le domaine fréquentiel en fonction des paramètres de la transformée. Ainsi, ce degré de liberté a été exploité par la méthode DRPE pour renforcer la sensibilité en considérant les paramètres indépendants de la transformée comme des clés secrètes additionnelles. Pour cet objectif, nous proposons dans ce travail une nouvelle technique de cryptage simultané de deux images [50] basée sur la transformée réciproque-orthogonale paramétrique (ROP) présentée dans [51] en exploitant ses paramètres qui sont indépendants comme une clé secrète supplémentaire.

Afin d'améliorer davantage la sécurité de la méthode DRPE, d'autres fonctions de cryptage telles que les fonctions de permutations linéaires ont été introduites [33]-[43], [45]-[48]. Ces fonctions sont généralement basées sur l'utilisation du chaos qui est connu pour sa grande sensibilité à ses paramètres [52]. Cependant, la méthode DRPE est considérée vulnérable à certaines attaques complexes [53]-[63]. Cela est dû essentiellement au fait que le cryptage DRPE, les transformées et les fonctions de permutations sont des opérations linéaires [63], [64]. Pour remédier au problème de linéarité présent dans les méthodes DRPE, nous proposons dans cette thèse une nouvelle technique de cryptage d'images basée sur un nouveau prétraitement non linéaire [65], [66].

La thèse est organisée comme suit :

- Dans le chapitre 1, nous présentons quelques définitions et principes de bases essentiels en cryptographie. Les caractéristiques cryptographiques du chaos sont également exposées.

- Dans le chapitre 2, nous élaborons la théorie des transformées paramétriques continues et discrètes les plus utilisées en cryptage, ensuite nous donnons un état de l'art détaillé

sur les méthodes existantes de cryptage des signaux image et vidéo basées sur les transformées paramétriques. Des techniques de mesure et d'évaluation de la sécurité de ce type de cryptage sont décrites à la fin de ce chapitre.

- Dans le chapitre 3, nous proposons une nouvelle méthode de cryptage numérique collectif des images en utilisant la transformée paramétrique ROP. Ce chapitre est divisé en deux parties, la première traite le cas du cryptage d'une seule image et la deuxième considère la méthode proposée dans le cas de deux images.

- Dans le chapitre 4, nous proposons une nouvelle méthode de cryptage d'images qui est une solution efficace pour le problème de linéarité des méthodes de cryptages DRPE basées sur les transformées paramétriques en introduisant un processus de prétraitement non-linéaire.

- Nous proposons dans le chapitre 5 une nouvelle méthode efficace de cryptage vidéo basée sur l'utilisation d'une transformée paramétrique réelle.

Nous clorons ce travail de recherche par une conclusion générale et quelques perspectives pour la poursuite des recherches effectuées.

Chapitre 1

Principes généraux de la cryptographie

1.1 Introduction

La cryptographie est l'une des branches de la cryptologie ou la science du secret [1]. Le mot cryptographie est composé du mot grec « *kryptos* » qui signifie caché et du mot « graphein » qui signifie écrire. Littéralement, cela signifie « écriture cachée » ou « secrète » [2]. La cryptographie protège la confidentialité de l'information en transformant un message compréhensible en un message incompréhensible. Cette opération s'appelle cryptage. Le recours au cryptage pour protéger le secret n'est pas quelque chose de nouveau. L'empereur romain Jules César utilisait une technique de cryptage par décalage alphabétique sur les correspondances destinées aux chefs de son armée [2]. Avec l'arrivée des ordinateurs, la cryptographie est devenue une science en tant que telle où se rencontrent les mathématiques, l'informatique et la théorie de l'information [1]-[6].

Dans ce chapitre nous allons revoir quelques définitions et principes de bases essentiels en cryptographie. Nous présenterons également la relation qui existe entre le chaos et la cryptographie.

1.2 Objectifs de la cryptographie

L'objectif principal de la cryptographie est de garantir la confidentialité d'un message lors d'un échange entre deux personnes à travers un canal peu sécurisé, de ce fait, si un tiers est présent dans le même réseau de communication, il ne pourra pas déchiffrer ce message [1]-[6]. Pour accomplir cet objectif, le message que l'on appelle souvent texte en clair [1] est crypté en un message incompréhensible appelé souvent texte crypté [1]. Par conséquent, seulement celui qui connait la clé secrète du cryptage peut le décrypter. Ce texte crypté peut être un texte, un enregistrement audio, des images ou une séquence d'images vidéo, ou autres types de données numériques [1]-[6].

En effet, la cryptographie ne dispose pas que d'un seul objectif. Parmi les autres objectifs de la cryptographie :

- **L'intégrité de l'information :** garantir la légitimité d'un message reçu, c'est à dire que le message n'a pas subi une manipulation malveillante durant son acheminement. Pour cela,

des algorithmes de hachages SHA (Secure Hash Algorithm) sont utilisés pour produire une empreinte unique à chaque message crypté [1].

- **L'authentification des communicants :** offrir au récepteur d'un message la possibilité de vérifier l'identité de l'émetteur afin de garantir qu'aucune usurpation d'identité n'a eu lieu. Pour cela, il existe des codes d'authentification [1].

- **La non-répudiation de l'information :** garantir que les participants dans un échange de messages ne pourront plus nier ce dernier à l'avenir. Pour cela, des signatures électroniques (numériques) sont utilisées [1].

1.3 Principe du cryptage et du décryptage

Un système cryptographique est un système qui peut être caractérisé par [3]-[5] :

- Un ensemble possible de textes en clairs M.
- Un ensemble possible de textes cryptés C.
- Un ensemble possible de clés de cryptage K_e et de clés de décryptage K_d.
- Un ensemble possible d'algorithmes cryptographiques

où l'algorithme cryptographique est composé d'une fonction de cryptage E et d'une fonction de décryptage D. La fonction de cryptage E permet d'obtenir un texte crypté C à partir d'un texte en clair M en utilisant une clé de cryptage K_e, quant à la fonction de décryptage D, elle permet de décrypter le texte crypté C afin d'obtenir le texte en clair M original en utilisant une clé de décryptage K_d. Cela illustre le principe fondamental du cryptage et du décryptage qui peut être formalisé comme suit [3]-[5]:

$$\begin{cases} E_{K_e}(M) = C \\ D_{K_d}(C) = M \end{cases} \tag{1.1}$$

Ce principe est illustré également dans la figure 1.1.

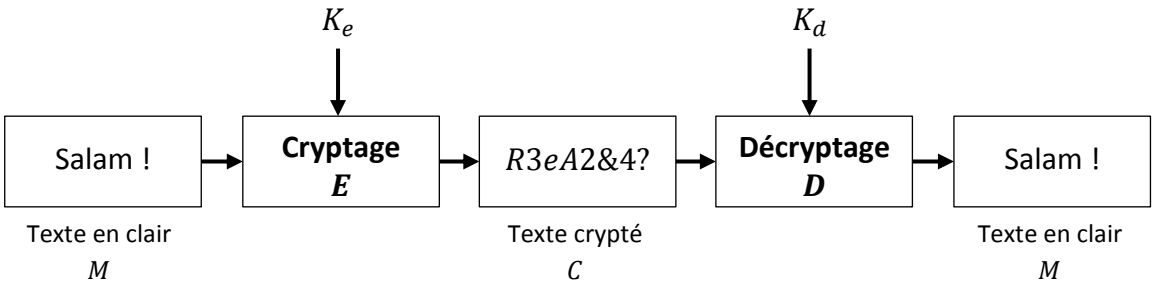


Figure 1.1 Principe fondamental du cryptage et du décryptage.

1.4 Classification des algorithmes de cryptage

En fonction de la relation existante entre la clé de cryptage et la clé de décryptage, un algorithme de cryptage peut être classé en deux classes d'algorithmes [1]-[5]. La première classe, comprenant les algorithmes à clé secrète ou à cryptage symétrique, et la seconde classe celle des algorithmes à clé publique ou à cryptage asymétrique.

1.4.1 Cryptage symétrique

Le principe du cryptage symétrique est illustré dans la figure 1.2.

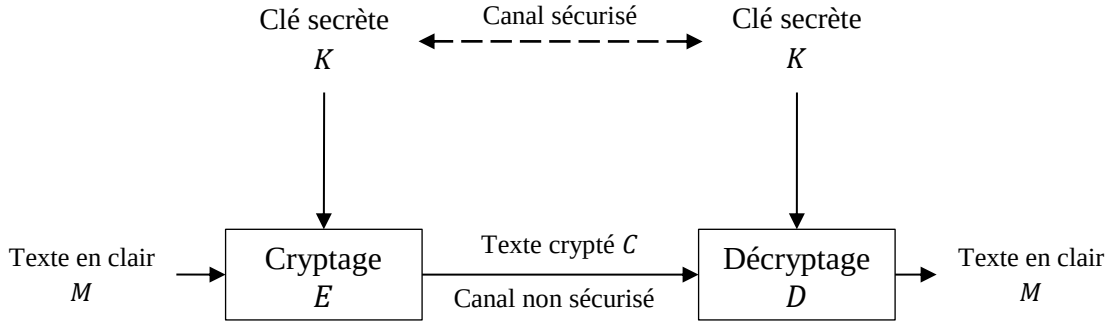


Figure 1.2 Principe du cryptage symétrique.

En cryptage symétrique, la clé de cryptage K_e est équivalente à la clé de décryptage K_d [3], ainsi l'équation précédente (1.1) devient :

$$\begin{cases} E_K(M) = C \\ D_K(C) = M \end{cases} \tag{1.2}$$

où la clé secrète K est utilisée à la fois pour le cryptage et le décryptage. En effet, la clé K doit absolument rester secrète, de plus, elle doit être échangée au préalable entre l'émetteur et le récepteur à travers un canal sécurisé [3].

Les algorithmes de cryptage symétrique peuvent être subdivisés en deux catégories:

- **Cryptage par flot** : « stream ciphers » en anglais où un message est crypté bit par bit par un générateur de flux de nombres pseudo-aléatoires [1], [3].
- **Cryptage par bloc :** « block ciphers » en anglais où un message est crypté par bloc de bits [3].

Le fameux standard de cryptage AES ou « Advanced Encryption Standard » en anglais [67] est un cryptage symétrique utilisé en pratique avec une clé secrète variable entre 128 bits et 256 bits. L'avantage du cryptage symétrique est la rapidité de cryptage/décryptage [1], cependant, il a l'inconvénient de nécessiter un canal sécurisé pour l'échange des clés, et cette situation se complique davantage dans un environnement multi-utilisateur.

1.4.2 Cryptage asymétrique

Dans un cryptage asymétrique, un message est crypté avec une clé K_e appelée clé publique, et décrypté avec une clé K_d différente appelée clé privée. Ce principe de cryptage asymétrique est illustré dans la Figure 1.3.Son principe repose sur le fait que le récepteur d'un message produit une paire de clés, une clé publique et une clé privée, cette dernière doit absolument rester secrète. En effet, la clé publique K_e sert uniquement à crypter les messages destinés au propriétaire de la clé, et la clé privée K_d sert à décrypter ces messages [1]-[4]. En conséquence, le propriétaire de la clé publique est l'unique personne pouvant les décryptés, car il est le seul à posséder l'autre partie de la clé qui est la clé privée.

L'avantage du cryptage asymétrique est de ne plus avoir besoin d'un canal sécurisé pour l'échange des clés entre l'émetteur et le récepteur [5], cependant, il est connu pour être lent par rapport au cryptage symétrique [2], étant donné qu'il nécessite des opérations arithmétiques complexes pour la génération des clés [2]. Le cryptage RSA [1], [3], [4] est parmi les algorithmes de cryptage asymétrique les plus connus.

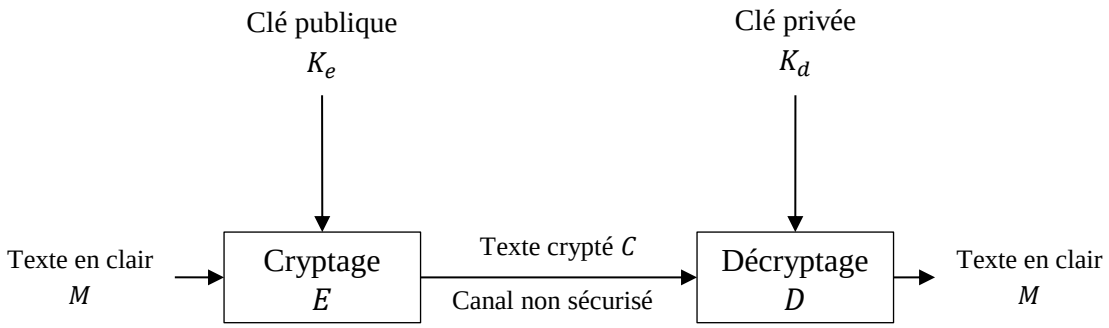


Figure 1.3 Principe du cryptage asymétrique.

Il faut noter qu'il existe des algorithmes de cryptage hybrides basés sur une combinaison d'un algorithme de cryptage symétrique et un algorithme de cryptage asymétrique, par exemple, le cryptage PGP (Pretty Good Privacy) ou le protocole SSL (Secure Sockets Layer) [1].

1.5 Principe de confusion et de diffusion

Claude Shannon a présenté deux propriétés importantes en cryptographie dans son fameux article sur la théorie de la communication des systèmes de sécurité de base [68]. Ce sont la propriété de confusion et la propriété de diffusion. Si ces propriétés sont prises en considération lors de la conception d'un algorithme de cryptage, elles garantiront la complexité de la relation entre le texte crypté et le texte en clair. Cela permet de rendre l'algorithme robuste contre les attaques. Pour réaliser cela, des techniques de substitution et des techniques de permutation sont utilisées [3].

1.5.1 Technique de substitution

Le principe d'une technique de substitution consiste à remplacer le caractère d'un texte en clair par un autre caractère différent en utilisant une fonction mathématique prédéterminée [3]. Cela permet de brouiller la relation existante entre le texte crypté et le texte en clair afin de satisfaire la propriété de la confusion qui a pour but de décourager une attaque statistique basée sur la redondance statistique des caractères [3]. Comme exemple, la figure 1.4 illustre une technique de substitution où chaque lettre d'un message secret est décalée de trois positions suivant l'ordre alphabétique [1].

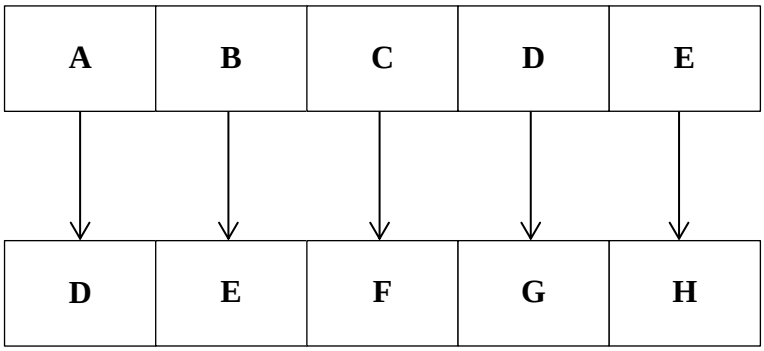


Figure 1.4 Exemple d'une technique de substitution.

1.5.2 Technique de permutation

Le principe d'une technique de permutation ou de transposition consiste à restructurer l'ordre des caractères d'un texte en clair en changeant leurs positions selon un arrangement prédéterminé [3]. Cela permet de satisfaire la propriété de diffusion par la dissipation de la redondance du texte sur toute sa longueur [3].

La figure 1.5 montre comme exemple une technique de permutation simple qui consiste à permuter deux caractères adjacents d'un message secret.

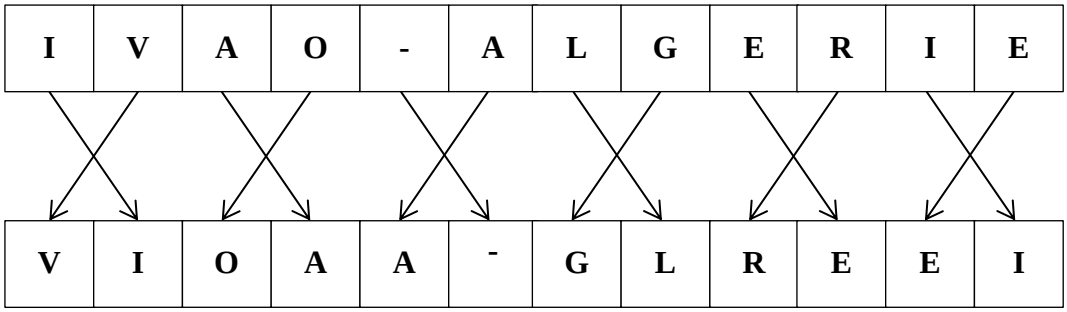


Figure 1.5 Exemple d'une technique de permutation.

1.6 Principes de Kerckhoffs

Par le passée, la sécurité d'un algorithme de cryptage reposait exclusivement sur le secret qui entoure la structure de l'algorithme [3]. Cela n'est plus recommandé aujourd'hui, car il est difficile de garder un algorithme secret [1]. Ainsi, en 1883, Auguste Kerckhoffs a présenté dans son article « La cryptographie militaire » six principes importants en cryptographie [69]. Nous nous limiterons seulement à trois principes encore valides aujourd'hui [1]:

- La sécurité d'un algorithme de cryptage doit reposer seulement sur la clé secrète.
- Le décryptage sans la clé secrète en un temps humainement raisonnable doit être impossible.
- Trouver la clé secrète à partir du texte en clair et du texte crypté doit être impossible en un temps humainement raisonnable.

Ces principes importants en cryptographie moderne doivent être pris en compte lors de la conception d'algorithmes de cryptage en supposons que l'attaquant connait déjà toute la structure de l'algorithme de cryptage et de décryptage sauf la clé secrète.

1.7 Chaos en cryptographie

Il est tout à fait possible de représenter un processus dynamique chaotique en utilisant des équations rigoureusement déterministes régies par un paramètre de contrôle et une condition initiale [52]. En effet, si ces équations sont utilisées itérativement, nous pouvons générer une suite mathématique qui caractérise le comportement de ce processus [52]. De plus, en régime chaotique, ces suites peuvent servir comme des générateurs de nombres aléatoires [70], car elles possèdent des propriétés intéressantes en cryptographie telles que [52], [70]:

- **Sensibilité à la condition initiale :** le moindre changement dans la condition initiale génère en sortie un régime pseudo-aléatoire complétement différent de l'état précédent.
- **Pseudo-aléatoires** : une suite chaotique gouvernée par une équation déterministe permet de générer un régime chaotique pseudo-aléatoire.
- **Ergodique :** un processus chaotique est ergodique, car il possède la même distribution en sortie quel que soit la distribution de la variable présente à l'entrée.

En effet, une suite chaotique peut être unidimensionnelle ou multidimensionnelle. Parmi les suites chaotiques simples et efficaces les plus connues, on trouve la fameuse suite logistique et les suites chaotiques linéaires par morceaux ou « piecewise linear chaotic maps » PLCM en anglais [70].

1.7.1 Suite logistique

Une suite logistique est générée itérativement en utilisant l'équation quadratique suivante [52]:

$$x_{n+1} = \mu . x_n . (1 - x_n) \tag{1.3}$$

où $x_n \in (0,1)$ avec $n \in \mathbb{N}$ et x_0 comme condition initiale. $\mu \in (0,4)$ est le paramètre de contrôle. La suite logistique possède une grande sensibilité à sa condition initiale x_0, cependant, elle est chaotique seulement si $\mu \in (3.57,4)$. De plus, elle est entièrement chaotique seulement si son paramètre de contrôle $\mu \cong 4$. Cela peut être vu sur un diagramme appelé diagramme de bifurcation illustré dans la figure 1.6. Ce diagramme permet de visualiser l'évolution du processus dynamique à partir d'un régime régulier vers un régime chaotique en fonction du paramètre de contrôle de la suite. [52].

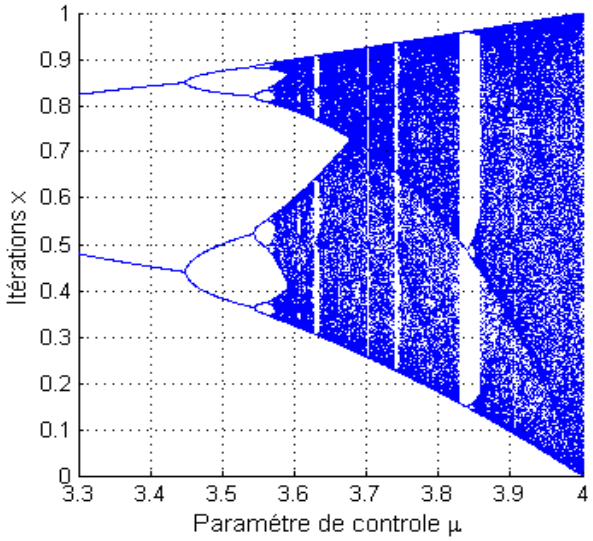


Figure 1.6 Diagramme de bifurcation de la suite logistique.

Nous remarquons d'après cette figure que la courbe est régulière au début, ensuite elle devient chaotique lorsque $\mu \in (3.57,4)$, cependant, il existe des intervalles réguliers en régime chaotiques appelées fenêtres où la suite n'est plus chaotique [52]. En effet, la suite logistique est purement chaotique seulement si $\mu \cong 4$ [71].

1.7.2 Suites chaotiques linéaires par morceaux PLCM

Zhou et al. ont proposé une suite PLCM efficace définit comme suit [72], [73] :

$$z_{n+1} = F(z_n, \lambda) = \begin{cases} \frac{z_n}{\lambda}, & 0 \leq z_n < \lambda \\ \frac{z_n - \lambda}{0.5 - \lambda}, & \lambda \leq z_n < 0.5 \\ F(1 - z_n, \lambda), & 0.5 \leq z_n < 1 \end{cases} \quad (1.4)$$

où $z_n \in (0,1)$ avec $n \in \mathbb{N}$ et z_0 comme condition initiale. $\lambda \in (0,0.5)$ est considéré comme le paramètre de contrôle. La suite PLCM définit par Zhou possède une grande sensibilité à sa condition initiale z_0, et contrairement à la suite logistique, elle est chaotique sur tout l'intervalle de définition du paramètre de contrôle λ. Cela peut être clairement vu dans son diagramme de bifurcation illustré dans la figure 1.7.

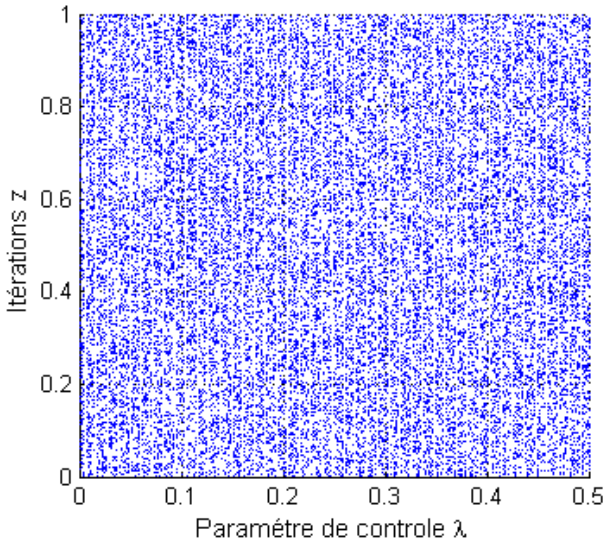


Figure 1.7 Diagramme de bifurcation de la suite PLCM définit par Zhou et al.

1.8 Attaques considérées dans la cryptanalyse

Contrairement à la cryptographie, en cryptanalyse, le cryptanalyste ou l'attaquant tente de mettre en place des attaques de différents niveaux afin de divulguer la clé secrète ou le texte en clair sans [1], [3], [5]. Ces attaques ne sont pas méthodiques, mais elles peuvent être classées en quatre attaques génériques [1], [5].

- **Attaque à texte crypté seulement :** dans ce cas, le cryptanalyste a en sa possession seulement des textes cryptés. Il va tenter de deviner la clé secrète ou les textes en clair originaux à partir des textes cryptés. L'attaque par force brute est parmi ce type d'attaques où l'attaquant tente de deviner la clé secrète de décryptage par l'essai de toutes les combinaisons réelles possibles que peut avoir une clé secrète.

- **Attaque à texte en clair connu :** dans ce cas, le cryptanalyste a en sa possession plusieurs paires de textes cryptés / textes en clairs. Son but consiste à récupérer la clé secrète par l'analyse de la relation existante entre ces paires de textes.

- **Attaque à texte en clair choisi :** Dans ce cas, le cryptanalyste possède beaucoup plus de ressources que dans l'attaque à texte en clair, car il est libre de choisir le type de texte en clair à crypter. Une analyse du texte crypté en sortie est effectuée dans le but de trouver une quelconque relation ou faille qui pourrait permettre de divulguer la clé secrète ou le texte en clair.

- **Attaque à texte crypté choisi :** dans ce cas, le cryptanalyste est libre de choisir le type de texte à décrypter. Cela a pour but aussi de divulguer la clé secrète.

1.9 Conclusion

Dans ce chapitre, nous avons revu quelques notions préliminaires en cryptographie qui sont nécessaires pour la compréhension des autres chapitres. Au début, nous avons vu le principe fondamental du cryptage et de décryptage ainsi que les deux principaux types d'algorithmes de cryptage qui sont le cryptage symétrique et le cryptage asymétrique. Nous avons par la suite vu le principe de confusion et de la diffusion ainsi que les principes de Kerckhoffs. Nous avons également montré l'intérêt du chaos en cryptographie, ainsi que les définitions de deux suites chaotiques connues. Enfin, les définitions des attaques de base en cryptanalyse ont été présentées.

Chapitre 2

Méthodes de cryptage d'images/vidéos basées sur des transformées paramétriques

2.1 Introduction

Les méthodes traditionnelles [3], [67] de cryptage sont efficaces pour le cryptage de données binaires bits par bits sans prendre en compte le contenu des données. De ce fait, l'utilisation directe de ces méthodes avec des données multimédia telle qu'une image ou une séquence d'images vidéo est lent et parfois inefficace [6]. Une image possède des caractéristiques particulières, telle que la présence d'une forte redondance et une corrélation élevée parmi ses pixels [5],[6], de plus, une image a une taille considérable et nécessite parfois des traitements en temps réel [6]. Par conséquent, de nouvelles méthodes de cryptage d'images et vidéo ont été proposées [5]-[49], parmi eux, la fameuse méthode de cryptage par deux masques de phases aléatoires ou « double random phase encoding » DRPE en anglais [7], [8]. La méthode DRPE était au début basée seulement sur deux masques de phases aléatoires et la transformée de Fourier [7], ensuite elle a été largement développée grâce à l'utilisation des transformées paramétriques [8]-[49].

Dans ce chapitre, nous présentons l'importance de l'utilisation des transformées paramétriques en cryptage. Pour y faire, nous avons élaboré un rappel théorique de deux transformées paramétriques très utilisées en cryptage qui sont la transformée fractionnaire de Fourier (TFR) [75] et la transformée réciproque-orthogonale paramétrique (ROP) [51]. Ensuite, nous donnons un état de l'art détaillé sur les méthodes de cryptage d'images/vidéo basées sur les transformées paramétriques. A la fin de ce chapitre, nous présentons les techniques de base utilisées pour l'évaluation de la sécurité de ce genre de méthodes de cryptage.

2.2 Importance des transformées paramétriques

Une transformée paramétrique peut être vue comme une généralisation ou une paramétrisation d'une transformée standard telle que la transformée de Fourier (TF) ou la transformée de Walsh-Hadamard (TWH). La paramétrisation ou la généralisation d'une transformée standard consiste à modifier son noyau (kernel) par l'introduction de paramètres indépendants. Cette généralisation permet d'avoir en sortie plusieurs interprétations du signal en fonction des paramètres de la transformée, une chose qui n'est pas possible avec une transformée fixe standard. En effet, les transformées paramétriques sont un outil mathématique important en traitement du signal avec des applications en filtrage, tatouage, cryptage et autres [74]. La transformée TFR [74] et la

transformée ROP [51] sont parmi les transformées paramétriques qui ont été utilisées avec succès en cryptage grâce à leurs nombreux paramètres indépendants qui peuvent être considérés comme des clés secrètes supplémentaires.

2.3 Transformée de Fourier fractionnaire TFR

La transformée TFR est une généralisation de la transformée TF. Parfois, elle est appelée la transformée TF rotationnelle ou la transformée TF angulaire, car la transformée TFR a un angle de rotation α variable et un paramètre $a = 2\alpha / \pi$ appelé ordre fractionnaire où $0 < \alpha < \frac{\pi}{2}$ [74]-[78].

2.3.1 Définition de base

Une transformée TFR d'ordre a d'un signal $x(t)$ est définie par [74]-[78]:

$$\mathcal{F}^a\big(x(t)\big) = \int_{-\infty}^{+\infty} x(t) K_a(u,t)\, dt \tag{2.1}$$

où $K_a(u,t) = \sqrt{1 - j\, ctg(\alpha)}\, . e^{j\pi(t^2 cot\alpha - 2t\, u\, csc\alpha + u^2 cot\alpha)} = \sum_{n=0}^{\infty} \exp(-j\, n\, \alpha)\, . \mathrm{H}_n(t)\, \mathrm{H}_n(u)$,

$\alpha = a\, \frac{\pi}{2}$ et $\mathrm{H}_n(t)$ indique la fonction normalisée d'Hermite-Gauss d'ordre n définie par :

$$\mathrm{H}_n(t) = \frac{2^{1/4}}{\sqrt{2^n\, n!}}\, h_n(t)\, \left(\sqrt{2\pi}\right) e^{-\pi t^2} \tag{2.2}$$

où $h_n(t)$ est le n-ième polynôme d'Hermite ayant k zéros réels.

2.3.2 Propriétés

La transformée TFR possède plusieurs propriétés intéressantes telles que [74]-[78]:

- Commutativité: $\mathcal{F}^a\, .\, \mathcal{F}^b = \mathcal{F}^b\, .\, \mathcal{F}^a$
- Associativité: $\mathcal{F}^a\, .\, (\mathcal{F}^b . \mathcal{F}^c) = (\mathcal{F}^a\, . \mathcal{F}^b)\, .\, \mathcal{F}^c$
- Linéarité: $\mathcal{F}^a\big(\sum_k c_k\, x_k(t)\big) = \sum_k c_k\, .\, \mathcal{F}^a\big(x_k(t)\big)$
- Additive: $\mathcal{F}^a\, . \mathcal{F}^b = \mathcal{F}^{a+b}$

- Unitaire: $(\mathcal{F}^a)^{-1} = \mathcal{F}^{-a} = (\mathcal{F}^a)^*$ où $(.)^*$ est le conjugué Hermitien appelé aussi la transposée-conjuguée.
- L'inverse de la transformée TFR est équivalent à $(\mathcal{F}^a)^{-1} = \mathcal{F}^{-a}$
- La transformée TFR $\mathcal{F}^a$ est réduite à la transformée TF standard $\mathcal{F}$ si $a = 1 \Rightarrow \alpha = \frac{\pi}{2}$
- La transformée TFR d'ordre $a = 0$ revient au signal lui-même, exemple, $\mathcal{F}^{a=0}\big(x(t)\big) = x(t)$, c'est le cas aussi pour les angles α multiples de 2π.

2.3.3 Définition discrète

Pour l'utilisation de la transformée TFR avec des signaux discrets, il existe plusieurs méthodes de calcul de la transformée de Fourier fractionnaire discrète (TFRD) qui sont proposées [74]. La méthode de décomposition en éléments propres est parmi les méthodes les plus utilisées [74]-[77], car c'est celle qui se rapproche le plus de la définition de la transformée TFR sans perdre ses propriétés importantes.

Suivant la décomposition en éléments propres de la matrice **F** de la transformée TF discrète, Pei et al. [75] ont défini la transformée TFRD d'ordre a par une matrice notée $\mathbf{F}^a$ telle que :

$$\mathbf{F}^a = \mathbf{V}\,\Lambda^a\mathbf{V}^{\mathrm{T}} = \begin{cases} \sum_{n=0}^{N-1} {\lambda_n}^a\, v_n v_n^T, & \text{Si } N \text{ impair} \\ \sum_{n=0}^{N-2} {\lambda_n}^a\, v_n v_n^T + {\lambda_N}^a\, v_N v_N^T, & \text{Si } N \text{ pair} \end{cases} \tag{2.3}$$

où $(.)^{\mathrm{T}}$ indique l'opération de la transposée, et Λ^a est une matrice diagonale dont les coefficients non nuls sont les valeurs propres ${\lambda_n}^a$, où ${\lambda_n}^a = e^{-j\frac{\pi}{2}na}$, et $a = \frac{2\alpha}{\pi}$. De plus, si N est un nombre impair, la matrice $\mathbf{V} = [v_0|v_1|\cdots|v_{N-2}|v_{N-1}]$, sinon $\mathbf{V} = [v_0|v_1|\cdots|v_{N-2}|v_N]$. Les éléments v_n représentent le n-iéme vecteur propre d'une matrice presque tri-diagonale **S** de taille $N \times N$ et définie par [75] :

$$\mathbf{S} = \begin{bmatrix} 2 & 1 & 0 & 0 & \dots & 0 & 1 \\ 1 & 2\cos\omega & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 2\cos 2\omega & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \vdots & \vdots \\ 1 & 0 & 0 & 0 & \dots & 1 & 2\cos(N-1)\,\omega \end{bmatrix} \tag{2.4}$$

où $\omega = 2\pi/N$. Cette matrice est commutative avec la matrice de Fourier **F** si l'égalité $\mathbf{F}.\mathbf{S} = \mathbf{S}.\mathbf{F}$ est satisfaite. Notez que la matrice de Fourier **F** possède seulement quatre valeurs propres distinctes $\{1, j, -1, j\}$ [75].

La transformée TFRD préserve toutes les propriétés importantes de la version continue telles que [75]:

- L'additivité des ordres fractionnaires : $\mathbf{F}^a.\mathbf{F}^b = \mathbf{F}^{a+b}$
- L'inverse de la transformée TFRD est obtenu en prenant simplement $\mathbf{F}^{-a}$ de sorte que $\mathbf{F}^a.\mathbf{F}^{-a} = \mathbf{I}$, où **I** indique la matrice identité.

2.3.4 Définition discrète et réelle

La transformée TFRD est une transformée complexe transformant un signal réel en un signal complexe. Venturini et al. ont introduit dans [79] une technique générale pour la construction d'une matrice réelle $\mathbf{R}^a$ de taille N à partir d'une matrice fractionnaire complexe $\mathbf{M}^a$ d'ordre a et de taille $N/2$ en utilisant l'équation suivante :

$$\mathbf{R}^a = \mathbf{P}^{-1}\,\mathbf{W}_a\,\mathbf{P} \tag{2.5}$$

Où **P** est une matrice de permutation, et $\mathbf{W}_a$ est une matrice définie par :

$$\mathbf{W}_a = \begin{bmatrix} \mathrm{Re}(\mathbf{M}^a) & -\mathrm{Im}(\mathbf{M}^a) \\ \mathrm{Im}(\mathbf{M}^a) & \mathrm{Re}(\mathbf{M}^a) \end{bmatrix} \tag{2.6}$$

Ainsi, si on considère $\mathbf{M}^a$ comme étant une matrice $\mathbf{F}^a$ de la transformée discrète TFRD d'ordre a, on obtient la matrice $\mathbf{R}^a$ de la transformée TFRD réelle (TFRDR) qui est [79]. Cette nouvelle définition préserve la plupart des propriétés importantes de la transformée TFRD complexe [79] comme :

- L'orthogonalité, par exemple : $\mathbf{R}^a\,.\,(\mathbf{R}^a)^{\mathrm{T}} = \mathbf{I}$
- L'inverse de la transformée est équivalent à $(\mathbf{R}^a)^{-1} = \mathbf{R}^{-a}$

- L'additivité, par exemple : $\mathbf{R}^{a} \cdot \mathbf{R}^{b} = \mathbf{P^{-1}}\,\mathbf{W}_{a}\;\mathbf{P}\mathbf{P^{-1}}\,\mathbf{W}_{b}\;\mathbf{P} = \mathbf{P^{-1}}\,\mathbf{W}_{a}\,\mathbf{W}_{b}\;\mathbf{P} = \mathbf{P^{-1}}\,\mathbf{W}_{a+b}\,\mathbf{P} = \mathbf{R}^{a+b}$

2.3.5 Définition discrète à paramètres multiples

Pei et al. [24] ont proposé de généraliser davantage la transformée TFRD afin d'avoir plus d'ordres fractionnaires comme paramètres. Si on reprend l'équation (2.3), la matrice de la transformée TFRD à paramètres multiples que l'on note $\mathbf{F}^{\bar{a}}$ peut-être définie par [24]:

$$\mathbf{F}^{\bar{a}} = \mathbf{V}\,\Lambda^{\bar{a}}\mathbf{V}^{\mathrm{T}} \tag{2.7}$$

où

$$\Lambda^{\bar{a}} = \begin{cases} \text{diag}\,({\lambda_0}^{a_0}, {\lambda_1}^{a_1}, \cdots, {\lambda_{N-1}}^{a_{N-1}}), & \text{Si } N \text{ impair} \\ \text{diag}\,({\lambda_0}^{a_0}, {\lambda_1}^{a_1}, \cdots, {\lambda_{N-2}}^{a_{N-2}}, {\lambda_N}^{a_N}), & \text{Si } N \text{ pair} \end{cases} \tag{2.8}$$

et

$$\bar{a} = \begin{cases} (a_0, a_1, \cdots, a_{N-1}) & \text{Si } N \text{ impair} \\ (a_0, a_1, \cdots, a_{N-2}, a_N) & \text{Si } N \text{ pair} \end{cases} \tag{2.9}$$

où $\bar{a}$ est un vecteur paramétrique qui comprend N ordres fractionnaires indépendants, et les éléments propres ${\lambda_n}^{a_n} = e^{-j\frac{\pi}{2}n\,a_n}$ avec $a_n = 2\,\alpha_n\,/\pi$.

Pour la même complexité de calcul que la transformée TFRD, la transformée TFRD à paramètres multiples offre $N-1$ ordres fractionnaires supplémentaires qui peuvent être choisis aléatoirement de l'intervalle (0,2) [24], de plus, elle garde les mêmes propriétés de la transformée TFRD.

Parmi les propriétés importantes de la transformé TFRD à paramètres multiples on trouve [24] :

- L'inverse de la transformée TFRD à paramètres multiples est équivalent à $(\mathbf{F}^{\bar{a}})^{-1} = \mathbf{F}^{-\bar{a}}$ où $\mathbf{F}^{\bar{a}}.\mathbf{F}^{-\bar{a}} = \mathbf{F}^{\bar{a}-\bar{a}} = \mathbf{I}$.

- La matrice $\mathbf{F}^{\bar{a}}$ est périodique avec une période $4/n$ dans le paramètre a_n du vecteur paramétrique $\bar{a}$:
  - $\lambda_n{}^{a_n} = e^{-j\frac{\pi}{2}n\,.(a_n+\frac{4}{n})} = e^{-j\frac{\pi}{2}n\,a_n}$, $n \neq 0$.
  - $\lambda_0{}^{a_0} = 1$, $\forall\, a_0$.

- Si $\bar{a} = (a, a, \cdots, a)$ nous revenons à la définition de la transformée TFRD de l'équation (2.3), car la transformée TFRD à paramètres multiples est un cas particulier de la TFRD à paramètres multiples. C'est le cas aussi pour la transformée TF discrète si $\bar{a} = (1,1,\cdots,1)$.

Il faut noter que la TFRD à paramètres multiples a un vecteur paramétrique $\bar{a}$ de N ordres fractionnaires comme paramètres indépendants. Ces paramètres peuvent être considérés comme des clés supplémentaires en cryptage [24].

La transformée TFRD bidimensionnelle (2D) à paramètres multiples d'un signal 2D en forme de matrice **P** de taille $N \times M$ est définie par l'équation suivante :

$$\mathbf{F}^{(\bar{a},\bar{b})}[\mathbf{P}] = \mathbf{F}^{\bar{a}}.\mathbf{P}\,.\mathbf{F}^{\bar{b}} \tag{2.10}$$

où $\mathbf{F}^{\bar{a}}$ et $\mathbf{F}^{\bar{b}}$ sont les matrices de la transformée TFRD 2D à paramètres multiples construites en utilisant l'équation (2.7) avec un vecteur paramétrique $\bar{a}$ de taille $1 \times N$ et un autre vecteur paramétrique $\bar{b}$ de taille $1 \times M$, respectivement.

2.4 Transformée réciproque-orthogonale paramétrique ROP

Bouguezel et al. ont proposé dans [51] une nouvelle transformée paramétrique discrète qu'on appelle la transformée réciproque-orthogonale paramétrique (ROP). Cette nouvelle transformée est basée sur une paramétrisation des matrices de la transformée TWH. En comparant avec la transformée TFR, la transformée ROP possède une structure plus simple, un algorithme rapide pour son calcul et un nombre de paramètres plus grand dans sa version récente [26], [27].

2.4.1 Définition de base et propriétés

Supposons $N = 2^r$ est la longueur d'une séquence quelconque avec $r \in \mathbb{N}^*$.

Soit $\mathbf{H}_N$ une matrice de Hadamard de taille $N \times N$ définie par :

$$\mathbf{H}_N = \mathbf{H}_2 \otimes \mathbf{H}_2 \otimes \ldots \otimes \mathbf{H}_2 \tag{2.11}$$

où $\mathbf{H}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ et $\otimes$ représente le produit de Kronecker.

Soit n, $0 \leq n \leq N-1$, un nombre entier décomposé en binaire :

$$n = n_{r-1}\, 2^{r-1} + n_{r-2}\, 2^{r-2} + \cdots + n_1 2 + n_0 \tag{2.12}$$

où n_i, $0 \leq i \leq r-1$, peut-être un 0 ou un 1.

Une ligne de la matrice $\mathbf{H}_N$ est considérée comme une ligne d'indice négatif si la décomposition binaire de son indice n dans la matrice satisfait l'égalité $(-1)^{\sum_{i=1}^{r-1} n_i} = -1$ [51].

De ce fait, une matrice $\mathbf{T}_N^{\mathbf{V}}$ d'ordre N de la transformée ROP est construite selon les étapes suivantes [51]:

1) Générer un vecteur symétrique $\mathbf{V}$ de longueur N que l'on appelle vecteur paramétrique et qui est défini par :

$$\mathbf{V} = \begin{bmatrix} 1 & a_1 & a_2 & \ldots & a_{\frac{N}{2}-1} & a_{\frac{N}{2}-1} & \ldots & a_2 & a_1 & 1 \end{bmatrix} \tag{2.13}$$

où a_i, $i = 1, 2, \ldots, N/2-1$ sont des paramètres indépendants qui peuvent être choisis aléatoirement du plan complexe.

2) Construire une matrice ROP $\mathbf{T}_N^{\mathbf{V}}$ d'ordre N en effectuant une multiplication élément par élément entre le vecteur paramétrique $\mathbf{V}$ et les lignes de la matrice de Hadamard $\mathbf{H}_N$ dont l'indice est désigné comme négatif.

On obtient ainsi une matrice ROP $\mathbf{T}_N^{\mathbf{V}}$ qui possède les propriétés suivantes :

- Réciproque et orthogonale : $\mathbf{T}_N^{\mathbf{V}} . \left(\mathbf{T}_N^{\mathbf{V}}\right)^{\mathrm{RT}} = N\mathbf{I}$ où $\mathbf{I}$ indique la matrice identité et $(.)^{\mathrm{RT}} = ((.)^{\mathrm{T}})^{\mathrm{R}} = ((.)^{\mathrm{R}})^{\mathrm{T}}$ l'opération de transposer réciproque de la matrice.

- L'inverse de la transformée ROP est équivalent à $\left(\mathbf{T}_N^{\mathbf{V}}\right)^{-1} = \frac{1}{N}\left(\mathbf{T}_N^{\mathbf{V}}\right)^{\mathrm{RT}}$.

- Les $N/2-1$ paramètres indépendants du vecteur paramétrique peuvent être choisis aléatoirement du plan complexe et servir comme des clés secrètes additionnelles en cryptage [25].

- Un algorithme rapide pour son calcul est disponible [51].

2.4.2 Nouvelle définition

La définition de base de la transformée ROP offre seulement $N/2-1$ paramètres indépendants pour une séquence d'entrée de taille N. Récemment, Bouguezel et al. ont proposé dans [26] une nouvelle définition de la transformée ROP avec une meilleure paramétrisation. Cette nouvelle définition est basée sur l'utilisation d'une méthode de construction récursive des matrices ROP et possède un algorithme rapide et efficace pour son calcul. En utilisant l'algorithme rapide de la nouvelle transformée ROP, une matrice ROP $\mathbf{T}_{2N}$ d'ordre $2N$ peut être définie par l'équation suivante pour n'importe quelle valeur $N=2^r, r>0$ [26]:

$$\mathbf{T}_{2N}=\left(\prod_{s=1}^{r}(\mathbf{I}_{2^{s-1}}\otimes\mathbf{H}_2\otimes\mathbf{I}_{2^{r+1-s}})\,\mathbf{E}^{(s)}\right)(\mathbf{I}_{2^r}\otimes\mathbf{H}_2) \tag{2.14}$$

$$\mathbf{E}^{(s)}=diag\left(\mathbf{E}_{(1)}^{(s)}\quad\mathbf{E}_{(2)}^{(s)}\quad\cdots\quad\mathbf{E}_{(2^{s-1})}^{(s)}\right), s=1,2,3,\dots,r \tag{2.15}$$

où

$$\mathbf{E}_{(n)}^{(s)}=\begin{bmatrix}\mathbf{I}_{2^{r+1-s}} & \mathbf{O}_{2^{r+1-s}}\\ \mathbf{O}_{2^{r+1-s}} & \mathbf{D}_{(n)}^{(s)}\end{bmatrix}, s=1,2,3,\dots,r\,;n=1,2\,,3\,,\dots,2^{s-1} \tag{2.16}$$

avec $\mathbf{O}$ une matrice nulle et $\mathbf{D}_{(n)}^{(s)}$ définie par :

$$\mathbf{D}_{(n)}^{(s)}=diag\left(\,1,d_{(n,1)}^{(s)},d_{(n,2)}^{(s)},\dots\,,d_{(n,2^{r+1-s}-1)}^{(s)}\right) \tag{2.17}$$

où $d_{(n,m)}^{(s)}$, pour $s=1,2,3,\dots,r\,,n=1,2\,,3\,,\dots,2^{s-1}$, et $m=1,2,3,\dots,2^{r+1-s}-1$, ce sont des paramètres indépendants. La nouvelle définition de la transformée ROP offre $N\,log_2\left(\frac{N}{2}\right)+1$ paramètres indépendants grâce à l'utilisation d'une approche récursive par la décomposition des matrices ROP en un produit (Kronecker) de matrices creuses.

Parmi les propriétés intéressantes de la nouvelle définition de la transformée ROP :

- Réciproque et orthogonale où $\mathbf{T}_{2N} \cdot \mathbf{T}_{2N}{}^{\mathrm{RT}} = 2N\,\mathbf{I}$
- L'inverse de la matrice $\mathbf{T}_{2N}$ est obtenu par $\mathbf{T}_{2N}^{-1} = \frac{1}{2N}\,\mathbf{T}_{2N}^{\mathrm{RT}}$ et il peut être défini par

$$\mathbf{T}_{2N}^{-1} = \frac{1}{2N}(\mathbf{I}_{2^r} \otimes \mathbf{H}_2) \quad \prod_{s=1}^{r}(\mathbf{E}^{r+1-s})^{-1}(\mathbf{I}_{2^{r-s}} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^s}) \tag{2.18}$$

- Pour une séquence de $2N$ points, on a $N\,log_2\left(\frac{N}{2}\right) + 1$ paramètres indépendants.

La transformée ROP 2D d'une matrice $\mathbf{P}$ de taille $2N \times 2N$ est définie en utilisant une structure ligne-colonne comme suit :

$$\mathbf{T}_{2N}^{(1,2)}[\mathbf{P}] = \mathbf{T}^1{}_{2N}\,\mathbf{P}\,\mathbf{T}^2{}_{2N} \tag{2.19}$$

où $\mathbf{T}_{2N}^1$et $\mathbf{T}_{2N}^2$ ce sont deux matrices ROP d'ordre $2N$ construites en utilisant l'équation (2.14).

2.5 Méthodes de cryptage basées sur des transformées paramétriques : état de l'art

Dans cette section, nous présentons un état de l'art détaillé sur les méthodes existantes de cryptage d'images/vidéo basées sur des transformées paramétriques.

2.5.1 Méthodes de cryptage par masques de phases aléatoires DRPE

Après la publication en 1995 de la fameuse méthode de cryptage DRPE par Refregier et Javidi [7], Unnikrishnan et Singh ont proposé dans [9] d'améliorer significativement sa sécurité en remplaçant la transformée TF dans la méthode DRPE par la transformée TFR où l'ordre fractionnaire de la transformée est exploité comme une clé supplémentaire. Ce principe de cryptage consiste à convertir un signal image 2D en un bruit blanc stationnaire d'amplitude complexe. Le cryptage DRPE dans le domaine de la transformée TFR a été repris par la suite de façon générique dans de nombreuses méthodes de cryptage basées sur les transformées paramétriques [10]-[49]. Par conséquent, il est très important de bien le comprendre.

Pour crypter un signal image $f(x,y)$ avec la méthode DRPE dans le domaine de la transformée TFR [9], le signal est multiplié par un masque de phases aléatoires $\mathbf{R_1}$, ensuite une transformée TFR 2D est appliquée sur le produit résultant en utilisant un ordre fractionnaire a et

un ordre fractionnaire b, respectivement. L'image ainsi obtenue est multipliée encore une fois par un autre masque de phases aléatoires $\mathbf{R_2}$. Une autre transformée TFR 2D est appliquée sur le produit résultant en utilisant un ordre fractionnaire c et un ordre fractionnaire d, respectivement. Au final, on obtient un signal complexe $\Psi(x, y)$. Ce processus de cryptage est illustré dans la figure 2.1(a).

Les étapes de décryptage sont illustrées dans la figure 2.1(b). Le conjugué complexe du signal complexe $\Psi(x, y)$ est calculé au début, ensuite les étapes restantes sont similaires aux étapes de cryptage à l'exception dans le domaine spatial où le signal image décrypté $f(x, y)$ est obtenu en prenant le module du signal décrypté, car le masque $\mathbf{R_2}$ du domaine fréquentiel et les ordres fractionnaires a, b, c, et d sont les seules clés secrètes. Noter que les masques de phases aléatoires sont statistiquement indépendants, et leurs phases sont distribuées de façon aléatoire et uniforme sur l'intervalle $[0,2\pi]$.

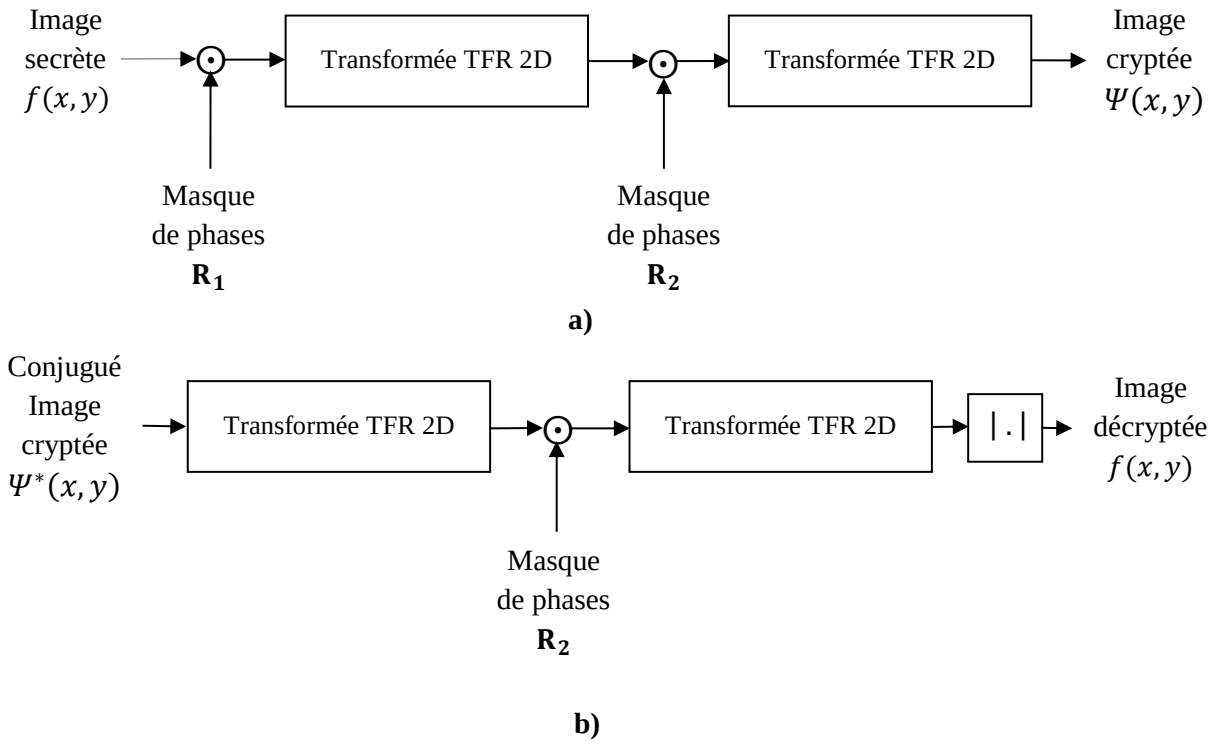


Figure 2.1 Méthode de cryptage DRPE dans le domaine de la transformée TFR,
a) algorithme de cryptage, b) algorithme de décryptage.

Pour améliorer la sensibilité de la clé secrète, certains auteurs proposent d'utiliser la méthode de cryptage DRPE itérativement ou en cascade [10]- [13]. Il faut noter qu'il existe également d'autres transformées équivalentes à la transformée TFR [17]-[19].

La méthode DRPE peut être implémentée optiquement [7], [9] et numériquement [24]. Dans les deux cas, la forme discrète de la transformée TFR doit être calculée numériquement pour le traitement des signaux discrets [74], [80]-[82]. Pei et al. ont proposé dans [24] d'utiliser la transformée TFRD à paramètres multiples afin d'améliorer significativement la sensibilité de la clé secrète. D'autres chercheurs [79] ont proposé d'autres versions de la transformée TFRD mais avec moins de paramètres [28],[83] que la transformée TFRD à paramètres multiples [24]. Comme la transformée TFRD est complexe, des versions réelles avec moins de paramètres ont été développées [39],[79].

Récemment, Bouguezel et al. ont proposé dans [25]-[27] de remplacer la transformée TFR à paramètres multiples dans la méthode DRPE par la transformée paramétrique discrète ROP comme illustrée dans la figure 2.2.

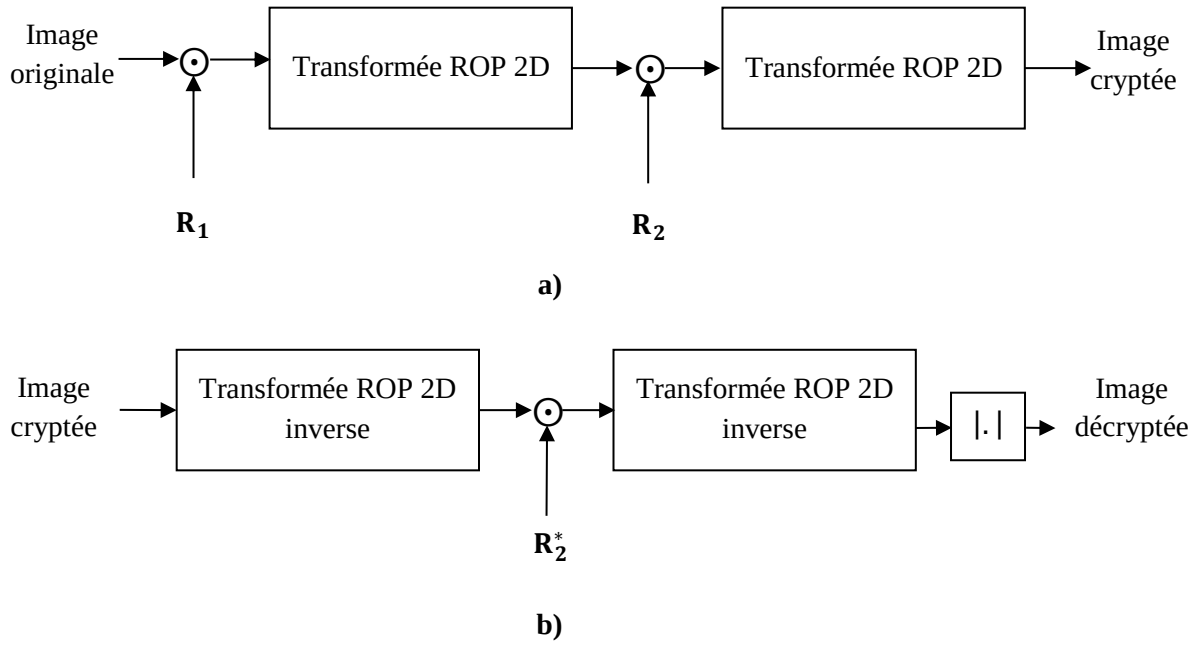


Figure 2.2 Méthode de cryptage DRPE dans le domaine de la transformée ROP,
a) algorithme de cryptage, b) algorithme de décryptage.

Cela améliore significativement la sécurité du cryptage, car la transformée ROP [26] offre un nombre de paramètres largement supérieur à celui de la transformée TFR à paramètres multiples [24]. De plus, elle possède une structure simple, et un algorithme rapide pour son calcul. D'autres transformées paramétriques discrètes ont été également proposées dans ce sens [28]- [32], [83]-[85] avec moins de paramètres indépendants que la transformée ROP.

2.5.2 Méthodes basées sur des permutations chaotiques

Comme la méthode DRPE est à l'origine une méthode de cryptage optique [7], le développement du domaine de l'holographie numérique [81] a facilité l'enregistrement, la transmission et le décryptage numérique des images et vidéos cryptées par la méthode DRPE. Ainsi, des méthodes de cryptage DRPE hybride opto-numérique ont été proposées [33]-[43]. La plupart de ces méthodes sont basées sur l'utilisation de fonctions numériques pour la permutation des pixels [33], [36]-[38], [40]-[43]. Hennelly et Sheridan ont proposé dans [33] une fonction de permutation pour remplacer le masque de phases aléatoires de la méthode DRPE. Cette fonction consiste à permuter des blocs de pixels de l'image dans le domaine de la transformée TFR. Cela a permis d'améliorer davantage la sensibilité de la clé secrète de la méthode DRPE.

Singh et al. ont proposé dans [34],[35] d'utiliser le chaos dans la méthode DRPE pour générer numériquement les masques aléatoires en utilisant des suites chaotiques. Ces masques sont appelés masques chaotiques de phases aléatoires ou « chaotic random phases mask » CRPM en anglais [34]. Cela a pour avantage de réduire le masque de phases aléatoires utilisé comme clé en un seul paramètre sans compromettre la sécurité du cryptage. Ainsi, seule la condition initiale utilisée pour la génération des masques CRPM est échangée avec le récepteur ce qui nous épargne d'éventuelles problèmes de synchronisation des clés entre l'émetteur et le récepteur. [34]

Depuis l'introduction du chaos dans la méthode DRPE [34], de nombreuses méthodes de cryptage DRPE en utilisant les transformées paramétriques et différentes fonctions de permutation chaotiques ont été proposées [36]-[38], [40]-[43]. Ces méthodes permettent d'améliorer significativement la sécurité du cryptage, car les suites chaotiques sont connues pour être sensibles à leurs paramètres. Lang et al. ont proposé dans [36] de substituer le masque de phases aléatoire dans le domaine de la transformée TFR à paramètres multiples par des fonctions de permutations chaotiques. Cette permutation est basée sur une suite logistique. Liu et al. ont

proposé dans [40] de remplacer le masque de phases aléatoires de la méthode DRPE par une fonction de permutation chaotique basée sur la suite d'Arnold afin d'améliorer l'espace de la clé.

Enfin, l'utilisation d'une transformée paramétrique en combinaison avec une permutation chaotique est analogue au principe de confusion et de diffusion de Shannon en cryptographie (voir chapitre 1). En effet, l'objectif principal des méthodes de cryptage basées sur les transformées paramétriques et les fonctions de permutation chaotique est l'amélioration de la sécurité en utilisant une structure modifiée du cryptage DRPE.

2.5.3 Cryptage de deux images en même temps

Parfois, il est nécessaire d'envoyer deux images en même temps, et du fait que la méthode DRPE transforme une image réelle en entrée en une image d'amplitude complexe en sortie, certains auteurs proposent de crypter deux images réelles simultanément [44]-[48].

Pour cela, deux images réelles sont cryptées en une image complexe avant d'appliquer un cryptage par la méthode DRPE dans le domaine de la transformée TFR. Ainsi, la première image est considérée comme le module de l'image complexe et la seconde image comme sa phase. Des fonctions de permutations chaotiques sont également utilisées [44]-[48] afin d'améliorer la sécurité du cryptage. L'avantage de ce principe de cryptage est la possibilité de crypter deux images en même temps.

2.5.4 Cryptage des séquences d'images vidéo

Le principe du cryptage DRPE consiste à crypter dans le domaine d'une transformée paramétrique un signal 2D réel ou complexe. Ce signal peut être une image, mais il peut être aussi la trame d'une séquence d'images vidéo. Il existe peu de travaux sur l'utilisation de la méthode DRPE en cryptage vidéo. Jindal et al. ont proposé dans [49] de crypter plusieurs trames successives d'une séquence d'images vidéo en utilisant la méthode DRPE dans le domaine de la transformée paramétrique discrète TFRD, où les paramètres de la transformée TFRD et le masque de phases aléatoires ont été utilisées avec succès comme des clés secrètes additionnelles.

2.6 Techniques d'évaluation de la sécurité

Une méthode de cryptage nécessite toujours une évaluation préalable de sa sécurité avant sa mise en service en pratique, car cela permet de corriger d'éventuelles failles. Dans cette section, nous présentons les techniques de base utilisées pour l'évaluation de la sécurité d'une méthode de cryptage d'images basée sur les transformées paramétriques [5],[6], [39].

Une méthode de cryptage d'images est considérée sûre seulement si elle arrive à convertir une image en entrée en une image cryptée aléatoire. Ce critère important est appelé sécurité perceptuelle [6]. Il existe également d'autres critères d'évaluation, comme par exemple, la qualité de cryptage, l'analyse de l'espace de la clé secrète, l'analyse statistique par histogramme, ainsi que la résistance au bruit additif et aux erreurs de transmission [5], [39].

2.6.1 Sécurité perceptuelle

Pour vérifier la sécurité perceptuelle, une inspection visuelle de l'image cryptée doit être faite. La méthode de cryptage est considérée comme sûre visuellement si l'image cryptée est aléatoire et inintelligible, et qu'aucun détail visuel sur la texture ou la silhouette de l'image originale ne soit reconnaissable [6].

2.6.2 Qualité du cryptage

Le coefficient de corrélation est un critère de mesure de la qualité du cryptage [5]. Lorsque le coefficient de corrélation est proche du zéro, la qualité de cryptage est meilleure. Ainsi, le coefficient de corrélation que l'on note c_{xy} entre une image originale x et une image cryptée y est défini par :

$$c_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{2.20}$$

où $c_{xy} \in [-1,1]$, $cov(x,y)$ c'est la covariance, et $D(x)$ et $D(y)$ la variance de x et y, respectivement.

Ces variables peuvent être calculées numériquement comme suit [5] :

$$E(x) = \frac{1}{L}\sum_{l=1}^{L} x_l \tag{2.21}$$

$$D(x) = \frac{1}{L} \sum_{l=1}^{L} (x_l - E(x))^2 \quad (2.22)$$

$$cov(x, y) = \frac{1}{L} \sum_{l=1}^{L} (x_l - E(x)) \, (y_l - E(x)) \quad (2.23)$$

où L est le nombre de pixels, et E l'espérance de x.

2.6.3 Sensibilité de la clé secrète

Selon les principes de Kerckhoffs, la sécurité d'une méthode de cryptage dépend entièrement de la sécurité de la clé secrète (voir chapitre 1). Cela signifie que la clé doit être forte et impossible à deviner à court et moyen terme. En effet, une clé secrète est considérée forte si elle résiste à une attaque par force brute où un attaquant tente de deviner la clé secrète en essayant toutes ses combinaisons possibles [5].

Pour vérifier cela, nous considérons une erreur de déviation δ dans les paramètres de la clé secrète afin de déterminer sa sensibilité aux erreurs. Cela permet de faire une estimation de l'espace de la clé secrète qui représente le nombre réel de combinaisons de clés possibles. L'erreur quadratique moyenne (EQM) est également calculée entre une image décryptée $\mathbf{I}'$ de taille $N \times M$ et son image originale $\mathbf{I}$ correspondante. Ainsi, nous pouvons déterminer la sensibilité de la clé secrète aux erreurs de manière approfondie. L'EQM est donc définie par [39] :

$$EQM(i', i) = \frac{1}{N \times M} \sum_{n=1}^{N} \sum_{m=1}^{M} (i'(n, m) - i(n, m))^2 \quad (2.24)$$

En général, si l'espace de la clé est supérieur à 2^{100}, la méthode de cryptage est considérée comme robuste contre les attaques par force brute [70].

2.6.4 Analyse statistique par histogramme

L'histogramme d'une image permet d'avoir une représentation graphique de la distribution des pixels d'une image en fonction des valeurs possibles d'un pixel [5]. Une méthode de cryptage est considérée comme robuste contre l'analyse statistique si l'histogramme de l'image cryptée est [5] :

- Entièrement différent de l'histogramme de l'image secrète originale.
- Possède une distribution identique et aléatoire peu importe l'image originale.

2.6.5 Résistance au bruit additif et aux erreurs de transmission

Une image cryptée peut subir l'influence du bruit du canal durant son transit. Pour vérifier la résistance d'une méthode de cryptage contre le bruit additif, nous calculons l'EQM entre une image décryptée et son image originale après avoir ajouté à l'image cryptée C un bruit additif selon l'équation suivante [39] :

$$C' = C\ (1 + \sigma\ G) \tag{2.25}$$

où C' est l'image cryptée bruitée, G est un bruit blanc de distribution Gaussienne, et σ est son coefficient de puissance. Le PSNR sert aussi comme mesure de distorsion du bruit additif. Ainsi, pour une image décryptée i' et son image originale i correspondante, le PSNR est défini par [5] :

$$PSNR\ = 10\ log_{10}\left[\frac{255^2}{EQM(i',i)}\right] \tag{2.26}$$

La résistance d'une méthode de cryptage contre les erreurs de transmission qui peuvent avoir lieu dans un canal de communication [39] peut être vérifiée également. Pour cela, avant le décryptage d'une image cryptée, on suppose qu'une partie de ses pixels a été compromise ou perdue au cours du transit.

2.7 Conclusion

Dans ce chapitre, nous avons élaboré la théorie des transformées paramétriques les plus utilisées en cryptage d'images et vidéo. Par la suite, nous avons donné un état de l'art détaillé sur les méthodes existantes de cryptage des signaux image et vidéo basées sur les transformées paramétriques. Nous avons également vu les différentes techniques utilisées pour l'évaluation de leurs sécurités et leurs performances.

Chapitre 3

Proposition d'un cryptage numérique collectif d'images basée sur la transformée ROP

3.1 Introduction

Les méthodes de cryptage DRPE consistent en général à convertir une image réelle en une image cryptée complexe. Par conséquent, de nombreuses méthodes de cryptage ont été proposées pour le cryptage de deux images en même temps [44]-[48]. Ces méthodes sont basées sur l'utilisation d'une méthode de cryptage DRPE basée sur la transformée TFR et des fonctions de permutations chaotiques.

Récemment, la transformée ROP [25]-[27] a été utilisée avec la méthode de cryptage DRPE dans sa structure de base pour le cryptage d'une seule image. Ces paramètres indépendants ont été utilisés avec succès comme une clé secrète additionnelle. La transformée ROP [26] est connue pour avoir un algorithme rapide et efficace pour son calcul et un nombre de paramètres indépendants largement supérieur à celui de la transformée TFR. Malgré cela, les méthodes de cryptages DRPE basées sur la transformée ROP n'ont pu être expérimentées dans le cas du cryptage d'une seule ou plusieurs images en même temps.

Dans ce chapitre, nous proposons une nouvelle méthode de cryptage numérique collectif des images en utilisant la transformée paramétrique ROP [50]. Ce chapitre est divisé en deux parties, la première traite le cas du cryptage d'une seule image en utilisant la transformée ROP avec une fonction de permutation chaotique, quant à la deuxième, nous proposons le cryptage de deux images en même temps en utilisant la transformée ROP avec une fonction de permutation complexe adaptée [50].

3.2 Cas d'une seule image

3.2.1 Motivation

Dans la méthode de cryptage DRPE dans le domaine de la transformée ROP présentée dans [26], [27], la clé secrète est constituée des nombreux paramètres indépendants de la transformée ROP et du masque de phases aléatoires. Ce masque est connu pour être volumineux, car il a une taille équivalente à celle de l'image qu'on souhaite crypter. Cela peut être désavantageux dans le cas où le masque doit être stocké ou synchronisé entre l'émetteur et le récepteur [34]. Cette problématique a été déjà traitée précédemment dans le cas de la méthode de cryptage DRPE dans le domaine de la transformée TFR [33], [36] par la substitution du masque de phases aléatoires du

domaine de la transformée TFR par une fonction de permutation. De ce fait, il est fortement souhaitable d'introduire ce concept de cryptage dans la méthode DRPE basée sur la transformée ROP.

En conséquence, nous proposons dans un premier temps de crypter une seule image en utilisant la transformée ROP avec une fonction de permutation chaotique. Cette méthode consiste à remplacer le masque de phases aléatoires de la méthode de cryptage DRPE dans le domaine de la transformée ROP par une fonction de permutation chaotique

3.2.2 Description de la méthode

3.2.2.1 Fonction de permutation chaotique

Cette fonction de permutation chaotique est proche de la fonction de permutation chaotique proposée dans [36]. Une image **I** de taille $2N \times 2N$ peut être permutée selon les étapes suivantes:

1) Former un vecteur **X** de $4N^2$ nombres réels aléatoires en utilisant l'équation (1.3) de la suite logistique avec une condition initiale x_0 et un paramètre de contrôle μ. La condition initiale x_0 est sélectionnée aléatoirement de l'intervalle $x_0 \in (0,1)$. Contrairement à la fonction de permutation chaotique proposée dans [36], le paramètre de contrôle μ est fixé à 4. Cela permet d'assurer une suite logistique entièrement chaotique (voir figure 1.6, chapitre 1).

2) Trier les éléments du vecteur **X** dans un ordre ascendant ou dans un ordre descendant, en parallèle, le changement effectué dans l'index des positions des éléments du vecteur est enregistré dans un vecteur **S** qu'on désigne comme le vecteur de permutation.

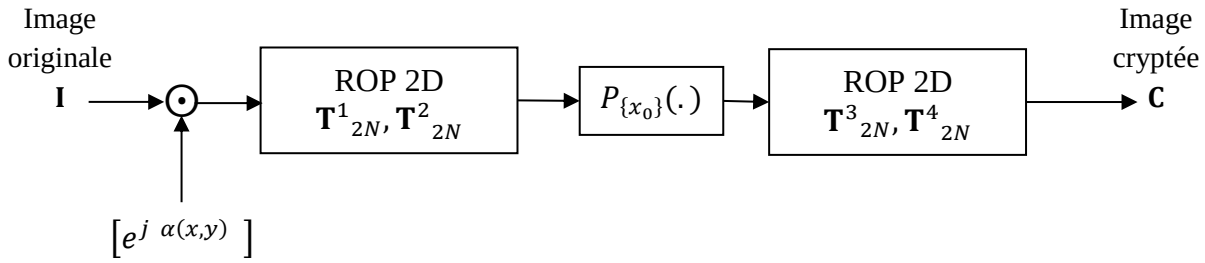
3) Redimensionner la matrice de l'image **I** en un vecteur, puis effectuer une permutation entre ses éléments en utilisant le vecteur de permutation **S** pour former un nouveau vecteur **C**, où $c_n = i_n(s_n)$, $n = 1,2,3, \dots , 1 \times 4N^2$.

4) Convertir le vecteur **C** obtenu lors de l'étape précédente en une matrice de taille $2N \times 2N$.

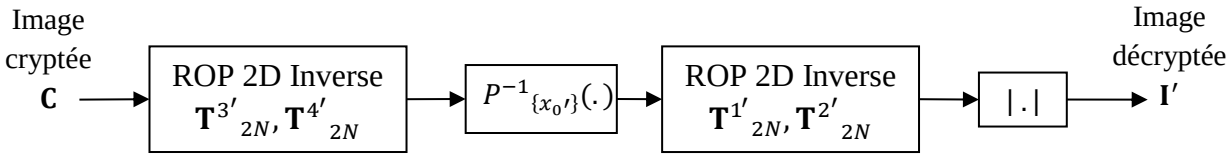
Ces étapes peuvent être résumées par une fonction $P_{\{x_0\}}(.)$, où x_0 est la condition initiale de la suite logistique. Les étapes de permutation inverse sont l'inverse des étapes précédentes, et peuvent être résumées par la fonction $P_{\{x_0\}}{}^{-1}(.)$. Il y a lieu de noter que lors de l'étape 3 en décryptage, l'image originale **I** est obtenue seulement si $i_n = c_n(s_n)$, $n = 1,2,3, \dots, 1 \times N^2$.

3.2.2.2 Algorithmes de cryptage et de décryptage

La méthode de cryptage proposée est illustrée dans la figure 3.1.



a)



b)

Figure 3.1 Méthode proposée pour le cryptage d'une image en utilisant la transformée ROP et une fonction de permutation, a) algorithme de cryptage, b) algorithme de décryptage.

Supposons une image **I** en forme de matrice de taille $2N \times 2N$, et $[e^{j.\alpha(x,y)}]$ un masque de phases aléatoires de taille identique à la taille de l'image **I**, où $\alpha(x,y)$ est une fonction ayant une distribution aléatoire et uniforme dans l'intervalle $[0,2\pi]$.

Soient $\mathbf{T}^1{}_{2N}, \mathbf{T}^2{}_{2N}$, $\mathbf{T}^3{}_{2N}$, et $\mathbf{T}^4{}_{2N}$ des matrices ROP d'ordre $2N$ construites en utilisant l'équation (2.14) de la transformée ROP avec $N\ log_2\left(\frac{N}{2}\right) + 1$ paramètres indépendants choisis aléatoirement du plan complexe.

Ainsi, l'image **I** est cryptée avec la méthode proposée selon les étapes suivantes :

1) Multiplier le masque de phases aléatoires $\left[e^{j\,.\alpha(x,y)}\right]$ élément par élément avec l'image **I** dans le domaine spatial.

2) Appliquer une transformée ROP 2D sur le résultat du produit en utilisant les matrices ROP $\mathbf{T^1}_{2N}$ et $\mathbf{T^2}_{2N}$ suivant une configuration lignes-colonnes.

3) Permuter l'image obtenue dans le domaine de la transformée ROP en utilisant la fonction de permutation chaotique $P_{\{x_0\}}$ avec une condition initiale x_0 aléatoirement choisie de l'intervalle $(0,1)$.

4) Appliquer une autre transformée ROP 2D sur l'image permutée résultante en utilisant les matrices ROP $\mathbf{T^3}_{2N}$ et $\mathbf{T^4}_{2N}$ suivant une configuration lignes-colonnes.

Finalement, on obtient une image cryptée d'amplitude complexe que l'on note **C**. Ces étapes de cryptage peuvent être résumées par l'équation suivante :

$$\mathbf{C} = \frac{1}{4\mathrm{N}^2}\left(\mathbf{T^3}_{2N}\left(P_{\{x_0\}}\left(\mathbf{T^1}_{2N}\left(\mathbf{I}\ \odot\ \left[e^{j\ \alpha(x,y)}\ \right]\right)\mathbf{T^2}_{2N}\right)\right)\mathbf{T^4}_{2N}\right) \qquad (3.1)$$

où $\odot$ dénote la multiplication élément par élément.

Dans la méthode proposée, la clé secrète que l'on note K est finalement composée des $4\left(N\ log_2\left(\frac{N}{2}\right)+1\right)$ paramètres indépendants des matrices ROP $\mathbf{T^1}_{2N}$, $\mathbf{T^2}_{2N}$,$\mathbf{T^3}_{2N}$, et $\mathbf{T^4}_{2N}$, et du paramètre x_0 de la fonction de permutation chaotique $P_{\{x_0\}}(.)$.

Supposons à présent une clé secrète K' composée de ${x_0}'$ comme condition initiale de la fonction de permutation chaotique inverse ${P^{-1}}_{x_0'}(.)$, et des $4\left(N\ log_2\left(\frac{N}{2}\right)+1\right)$ paramètres indépendants des matrices ROP $\mathbf{T^{1'}}_{2N}$, $\mathbf{T^{2'}}_{2N}$,$\mathbf{T^{3'}}_{2N}$, et $\mathbf{T^{4'}}_{2N}$.

De ce fait, les étapes de décryptage de l'image cryptée **C** avec la clé secrète K' consistent à prendre l'inverse des étapes précédentes de cryptage. Ces étapes de décryptage peuvent être résumées par l'équation suivante :

$$\mathbf{I}' = \left| \frac{1}{4N^2}\left(\left(\mathbf{T}^1{}_{2N}{}'\right)^{\mathrm{RT}}\left(P^{-1}{}_{\{x'_0\}}\left(\left(\mathbf{T}^3{}_{2N}{}'\right)^{\mathrm{RT}}\,\mathbf{C}\,\left(\mathbf{T}^4{}_{2N}{}'\right)^{\mathrm{RT}}\right)\right)\left(\mathbf{T}^2{}_{2N}{}'\right)^{\mathrm{RT}}\right) \right| \qquad (3.2)$$

où $(.)^{\mathrm{RT}}$ indique l'opération de la réciproque-transpose et $|.|$ indique le module.

Comme nous pouvons le remarquer, la méthode proposée est une méthode de cryptage symétrique, et l'image finale décryptée $\mathbf{I}'$ est identique à l'image originale $\mathbf{I}$ seulement si $x_0{}' = x_0$, et $\mathbf{T}^{n'}{}_{2N} = \mathbf{T}^n{}_{2N}$, où $n = 1,2,3,4$.

3.2.2.3 Résultats et discussions

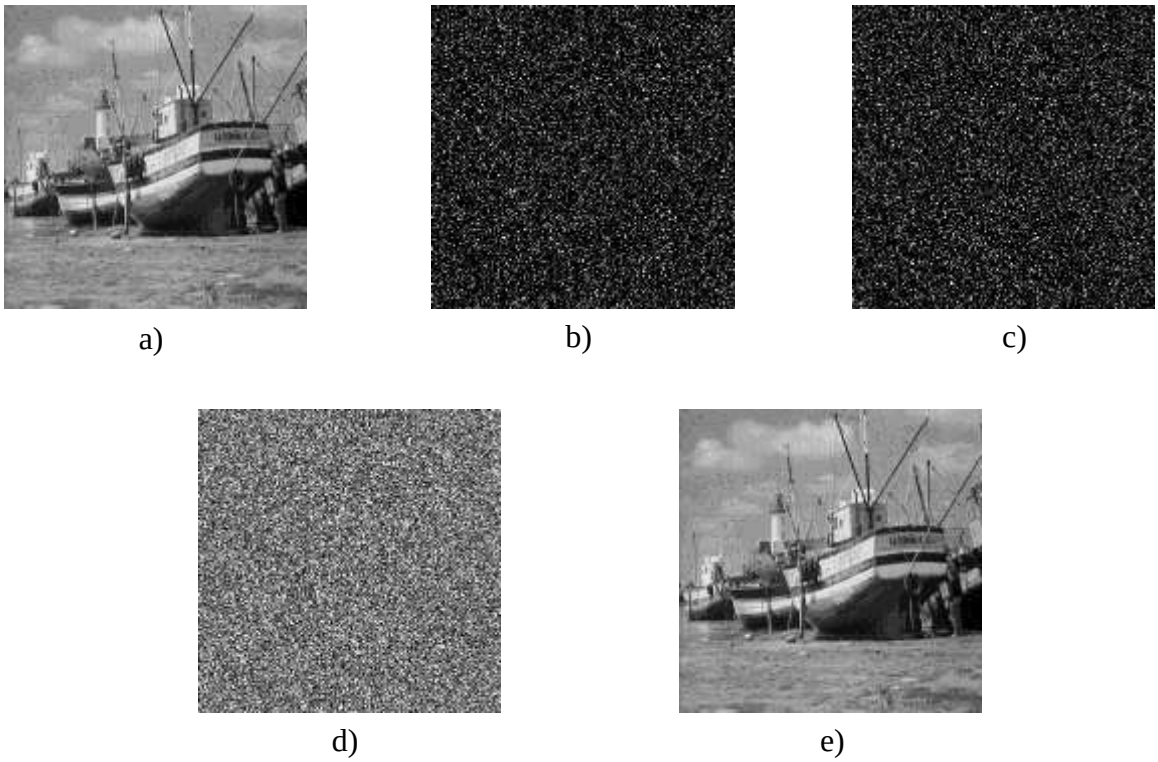
Dans cette section, nous allons présenter les résultats de simulation de la méthode proposée avec des images de tests standards de niveau gris (8bits). Une évaluation détaillée de sa sécurité est également discutée.

Soit une image de test de taille 256×256, et une clé secrète K qui est définie par :

- Le paramètre de la fonction de permutation chaotique : $x_0 = 0.57$.
- Pour $2N = 256$, les paramètres des matrices ROP sont au nombre de $4\left(N\,log_2\left(\frac{N}{2}\right) + 1\right) = 3076$ paramètres indépendants aléatoirement choisis du plan complexe.

Les résultats de simulation sont illustrés dans la figure 3.2 pour une image de test Boat de taille 256×256 cryptée avec la clé secrète K précédemment définit.

La figure 3.2(a) montre l'image Boat originale. Les figures 3.2(b) et 3.2(c) montrent, la partie réelle et la partie imaginaire de l'image cryptée. Les figures 3.2(d) et 3.2(e) montrent l'image Boat décryptée avec une clé secrète incorrecte et avec une clé secrète correcte. Les mêmes résultats ont été obtenus pour d'autres images de test.



a) b) c)

d) e)

Figure 3.2 Résultats de simulation: a) image originale, b) et c) partie réelle et partie imaginaire de l'image cryptée, d) et e) image décryptée avec une clé incorrecte et une clé correcte.

Ces résultats montrent que l'image originale ne peut être décryptée sans la connaissance de la clé secrète. De plus, nous constatons visuellement que l'image est correctement cryptée, car aucune information sur l'image originale n'est présente dans l'image cryptée. En conséquence, la méthode de cryptage proposée a une sécurité perceptuelle satisfaisante.

Afin d'évaluer la qualité de cryptage de manière objective, nous calculons le coefficient de corrélation entre une image cryptée et une image originale. Etant donné que l'image cryptée est complexe, le coefficient de corrélation c_r de la partie réelle et le coefficient de corrélation c_i de la partie imaginaire ont été calculés séparément. Les résultats sont présentés dans le tableau 3.1 pour différentes images de test.

Tableau 3.1 Coefficient de corrélation entre l'image originale et l'image cryptée.

Image	c_r	c_i
Boat	- 0.0021	0.0006
Cameraman	-0.0003	0.0004
Lenna	0.0004	0.0006
Barbara	0.0013	- 0.0019
Mandrill	-0.0005	-0.0007
Aerial	-0.0005	0.0013

D'après le tableau 3.1, la corrélation entre l'image cryptée et l'image originale est très proche du zéro. Cela signifie que l'image cryptée est entièrement indépendante de l'image originale, de ce fait, la méthode proposée a une qualité de cryptage satisfaisante.

A présent, pour évaluer la sensibilité de la clé secrète de la méthode proposée, nous supposons qu'une image Boat est cryptée avec la clé secrète K déjà définit au début, ensuite l'image Boat cryptée est décryptée avec une autre clé secrète K' identique à la clé K originale, mais qui comporte une erreur dans l'un de ses différents paramètres. Les résultats sont illustrés dans la figure 3.3.

La figure 3.3(a) montre l'image Lenna décryptée lorsque $x_0{}' = x_0 + 10^{-16}$. Les figures 3.3(b), 3.3(c), et 3.3(d) montrent respectivement l'image décryptée lorsque 50%, 25%, et 12% des paramètres des matrices ROP sont incorrects. Nous remarquons que l'image reste correctement cryptée dans tous les cas malgré une erreur insignifiante dans le paramètre de permutation x_0.

Pour déterminer la sensibilité des paramètres des matrices ROP nous calculons l'EQM entre l'image Boat décryptée et l'image Boat originale en fonction d'une erreur de déviation δ dans les paramètres des matrices ROP dans le cas de la méthode proposée. De la même manière, nous calculons l'EQM de la méthode de cryptage DRPE dans le domaine de la transformée ROP présentée par Bouguezel et al. dans [26]. Les résultats de calculs sont tracés dans la figure 3.4.

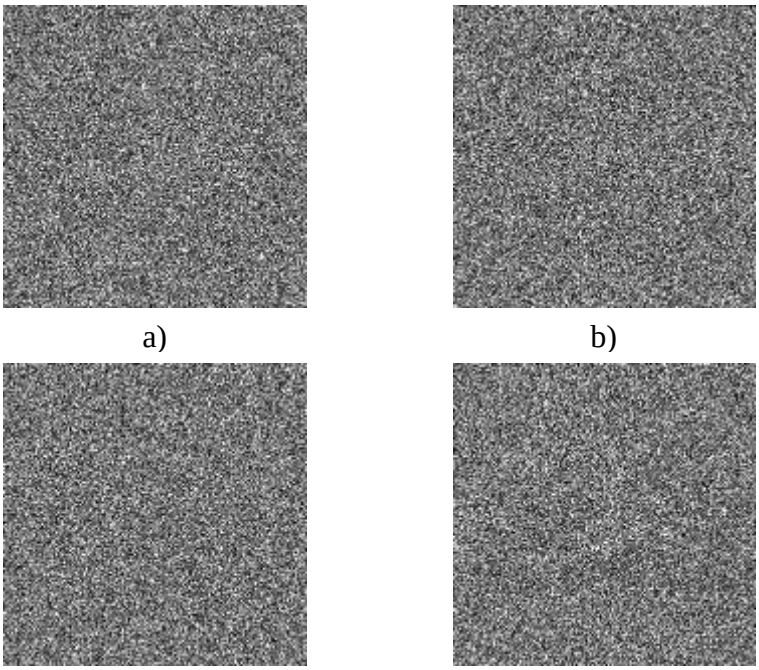


Figure 3.3 Image Boat décryptée en fonction des paramètres de la clé secrète, a) $x_0' = x_0 + 10^{-16}$, b), c) et d) 50%, 25%, et 12% des paramètres ROP sont incorrects.

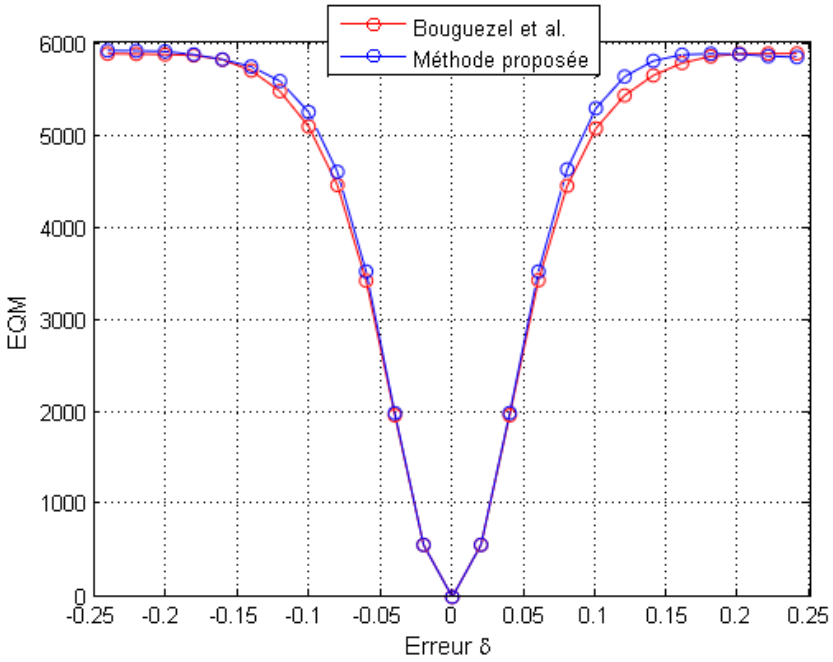
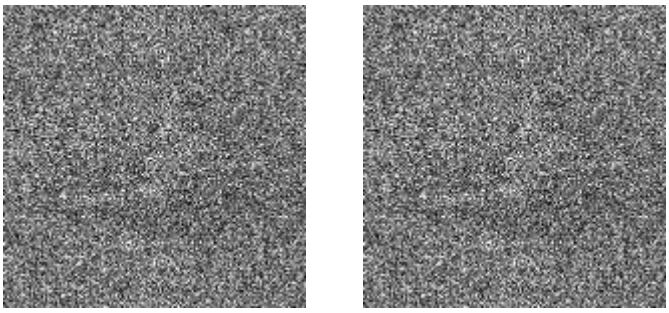


Figure 3.4 EQM en fonction de l'erreur δ dans les paramètres des matrices ROP.

On remarque que la méthode proposée a la même sensibilité que la méthode de Bouguezel et al. Pour mieux vérifier ces résultats, la figure 3.5 montre l'image Boat décryptée pour un seuil d'erreur minimum $\delta = 0.1$ dans les deux méthodes. On remarque que la clé secrète dans la méthode proposée a la même sensibilité aux erreurs que la méthode de Bouguezel et al. sans avoir a utilisé un masque $2N \times 2N$ de phases aléatoires. Cela a pour avantage de réduire la taille de la clé sans compromettre la sécurité du cryptage [34]. De plus, cela évite les problèmes de synchronisations des clés entre l'émetteur et le récepteur [33], [36].



a) b)

Figure 3.5 Image décryptée avec une erreur $\delta = 0.1$ dans les paramètres ROP, a) méthode proposée, b) méthode de Bouguezel et al.

Nous pouvons estimer à présent l'espace réel de la clé qui est d'environ $10^{3076} \times 10^{16}$ combinaisons de clés possibles. Ce nombre est largement supérieur au minimum recommandé qui est de 2^{100}. En conséquence, la méthode proposée est robuste contre les attaques par force brute.

A présent, pour évaluer la sécurité de la méthode proposée contre l'analyse statistique par histogramme, la figure 3.6 montre les histogrammes de quelques images de tests.

La figure 3.6(a) montre l'histogramme de l'image originale. Les figures 3.6(b) et 3.6(c) montrent respectivement l'histogramme du module et de la phase de l'image cryptée complexe. Nous constatons que peu importe l'image originale, celle cryptée correspondante possède un histogramme complétement différent de l'originale. De plus, ils ont une distribution aléatoire et identique. Cela permet d'interdire l'analyse statistique et protège l'image originale d'une éventuelle attaque statistique, car aucune information sur l'image originale n'a fuitée. Ainsi, la méthode proposée est robuste contre l'analyse statistique par histogramme.

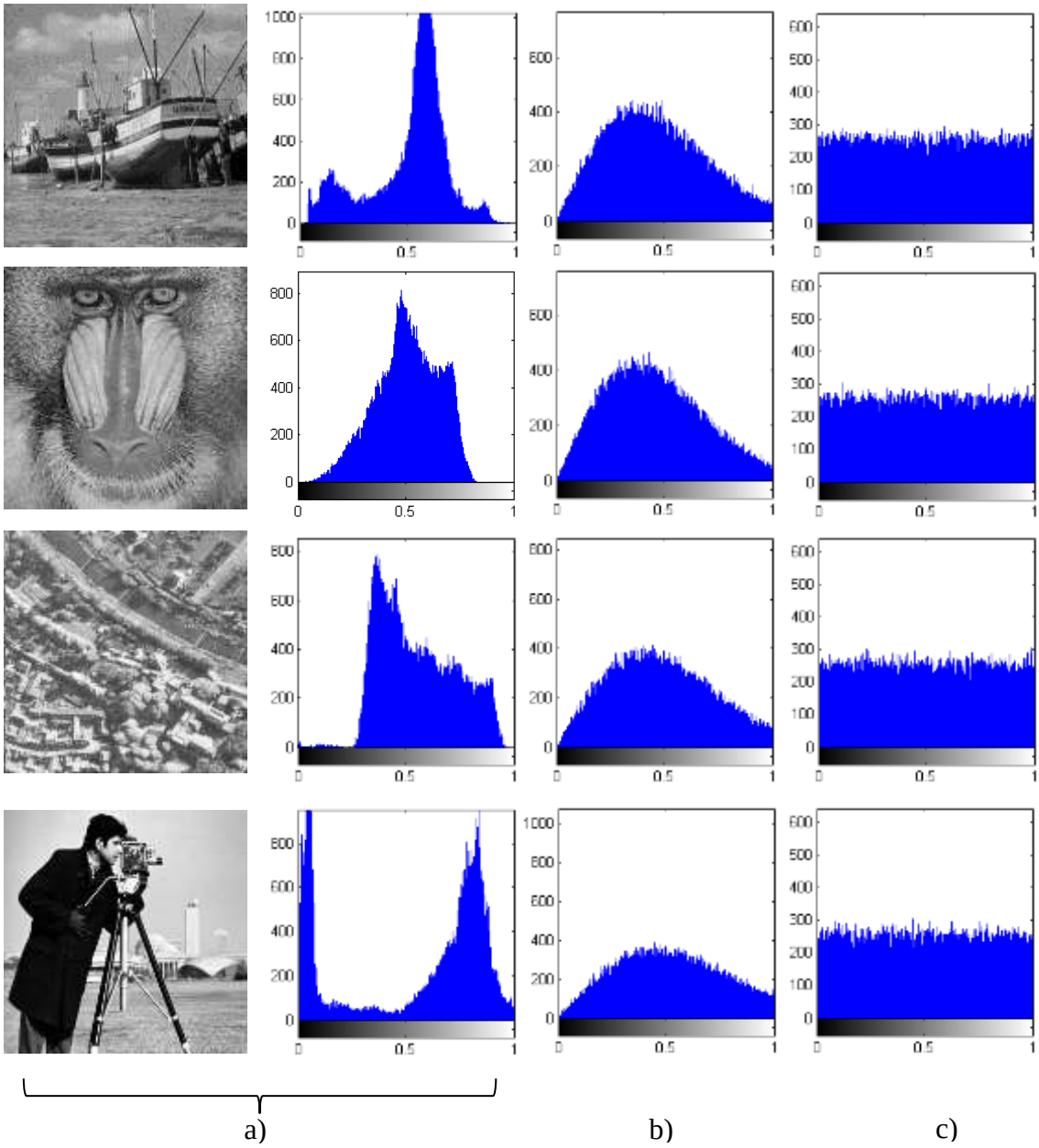


Figure 3.6 Histogrammes de quelques images de test:
a) l'image originale, b) et c) le module et la phase de l'image cryptée.

Pour vérifier la résistance de la méthode proposée contre l'attaque du bruit, nous avons ajouté à une image Boat cryptée un bruit blanc Gaussien de coefficient de puissance σ (voir chapitre 1, section 2.6.5). Ensuite, nous avons calculé l'EQM entre l'image Boat originale et l'image Boat décryptée en fonction du coefficient de puissance du bruit σ. Les résultats de simulations sont illustrés dans la figure 3.7 pour la méthode proposée, et pour la méthode de Bouguezel et al.

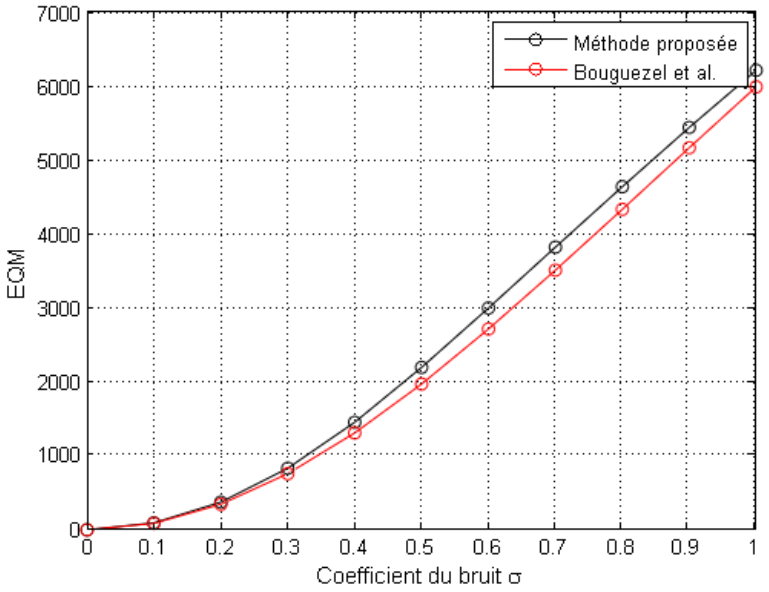


Figure 3.7 Comparaison de l'EQM en fonction du coefficient de puissance du bruit.

Nous constatons que la résistance de la méthode proposée est presque identique à celle de la méthode de Bouguezel et al. Nous remarquons aussi que l'EQM est proportionnelle au coefficient du bruit σ. De plus, la figure 3.8 montre l'image Boat décryptée par la méthode proposée ainsi que son PSNR pour différentes valeurs du coefficient de puissance σ du bruit additif. Nous constatons que l'image originale Boat reste reconnaissable malgré la présence du bruit. Ainsi, ces résultats démontrent la résistance de la méthode proposée contre le bruit additif.

Enfin, pour tester la résistance de la méthode proposée contre les erreurs dans le canal de communication, nous supposons qu'une partie des pixels de l'image Boat cryptée a été perdue au cours du transit. L'image Boat décryptée est illustrée dans la figure 3.9 avec son PSNR afin d'évaluer sa qualité. Nous remarquons que malgré un PSNR modeste, l'image Boat originale reste reconnaissable. En conséquence, ces résultats démontrent la résistance de la méthode proposée contre les erreurs de transmission.

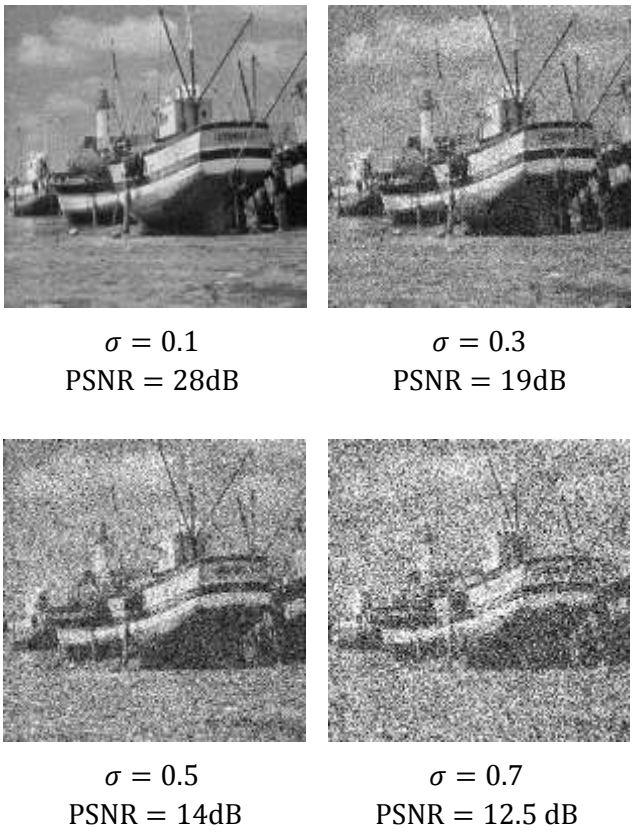


Figure 3.8 Image décryptée en fonction du coefficient de puissance du bruit.

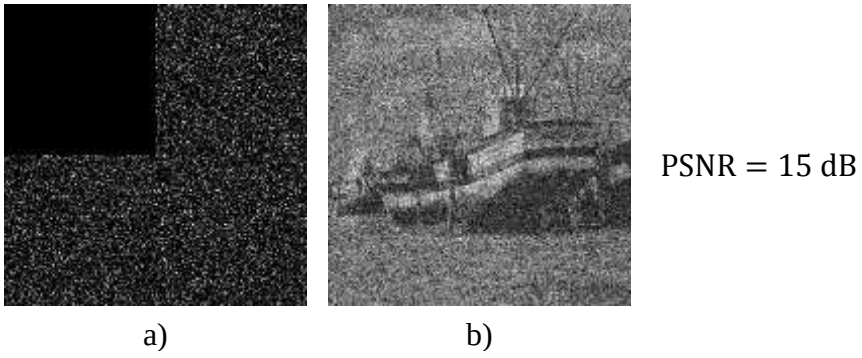


Figure 3.9 Image décryptée après la perte d'une partie des pixels, a) image cryptée, b) image décryptée.

3.3 Cas de deux images

3.3.1 Motivation

Dans la section précédente, nous avons vu l'intérêt de l'utilisation des fonctions de permutation chaotique avec la transformée ROP dans le cas d'une seule image en se basant sur des travaux antérieurs dans ce domaine, sauf que il est souvent important de crypter deux images simultanément, par exemple, lorsque deux images ont un contenu réciproquement relatif [44]-[48].

En effet, plusieurs méthodes de cryptage de deux images simultanément ont été déjà proposées en utilisant des fonctions de permutations chaotiques et un cryptage DRPE dans le domaine de la transformée TFR [44]-[48]. Ces méthodes consistent en général à encoder deux images réelles en une image complexe. L'une des images réelles est donc considérée comme le module de l'image complexe, et l'autre image réelle comme sa phase. Jusqu'aujourd'hui il n'existe pas de méthode de cryptage collectif d'images basée sur la transformée ROP. De ce fait, il est fortement souhaitable d'introduire ce concept de cryptage dans la méthode DRPE basée sur la transformée ROP.

Dans cette section, nous proposons une méthode de cryptage de deux images simultanément en utilisant un cryptage DRPE dans le domaine de la transformée ROP où nous proposons une fonction de permutation complexe [50]. Cette fonction de permutation complexe est une fonction de permutation chaotique réversible qui permet de réaliser une permutation entre les éléments de la partie réelle et de la partie imaginaire de l'image complexe en utilisant une suite logistique.

Contrairement au cas précédent d'une seule image, les masques de phases aléatoires sont maintenus, car l'image est complexe, et les deux masques serviront pour le cryptage de l'image réelle cryptée en tant que phase. Comme ces masques sont volumineux, on emploie des masques CRPM [34]. Ces derniers sont identiques aux masques de phases aléatoires sauf qu'ils sont générés au niveau de l'émetteur et au niveau du récepteur par l'emploi de deux suites logistiques indépendantes. Donc uniquement les paramètres des suites logistiques ont besoin d'être synchronisés entre l'émetteur et le récepteur [34]. Pour améliorer ce principe, dans notre cas, nous proposons de générer deux masques CRPM de manière à ce qu'ils soient dépendant l'un de l'autre afin d'améliorer la sensibilité de la clé. Il faut noter que dans cette section nous utilisons

la transformée ROP dans sa nouvelle définition [26] au lieu de l'ancienne définition [51] utilisée dans notre publication [50].

3.3.2 Description de la méthode proposée

3.3.2.1 Fonction de permutation complexe

Supposons nous souhaitons effectuer une permutation des éléments de la matrice d'une image complexe $\mathbf{X}$ de taille $2N \times 2N$. Les étapes de la fonction de permutation complexe proposée sont décrites dans les étapes suivantes :

1) Générer un vecteur de $1 \times 4N^2$ nombres réels en utilisant l'équation (1.3) de la suite logistique avec une condition initiale y_0 et un paramètre de contrôle γ, où $y_0 \in (0,1)$ et $\gamma \cong 4$ afin d'augmenter l'espace de la clé.

2) Trier les éléments du vecteur résultant de l'étape précédente dans un ordre ascendant, en parallèle, enregistrer dans un vecteur de permutation complexe $\mathbf{V}$ les changements dans l'index des positions.

3) Convertir la matrice $\mathbf{X}$ en un vecteur complexe $\mathbf{C} = \mathbf{R} + j\,\mathbf{I}$ de taille $1 \times N^2$ avec $\mathbf{R}$ comme la partie réelle et $\mathbf{I}$ comme la partie imaginaire du vecteur complexe, où $j = \sqrt{-1}$.

4) Accomplir une permutation entre les éléments de la partie réelle et les éléments de la partie imaginaire du vecteur complexe $\mathbf{C}$ en utilisant le vecteur de permutation $\mathbf{V}$ pour obtenir un nouveau vecteur complexe $\mathbf{C}' = \mathbf{R}' + j\,\mathbf{I}'$ comme suit :

$$\begin{cases} \mathbf{I}' = \mathbf{R}(\mathbf{W}) \\ \mathbf{R}'(\mathbf{W}) = \mathbf{I} \end{cases} \tag{3.3}$$

5) Convertir le vecteur $\mathbf{C}'$ en une matrice complexe $\mathbf{X}'$ de taille $2N \times 2N$.

Les étapes précédentes peuvent être résumées par une fonction que l'on note $P_{c\{y_0,\gamma\}}$. De plus, la fonction proposée a la propriété d'être réversible où $P_c{}^{-1}{}_{\{y_0,\gamma\}} = P_{c\{y_0,\gamma\}}$.

3.3.2.2 Algorithmes de cryptage et de décryptage

La méthode proposée est illustrée dans la figure 3.10.

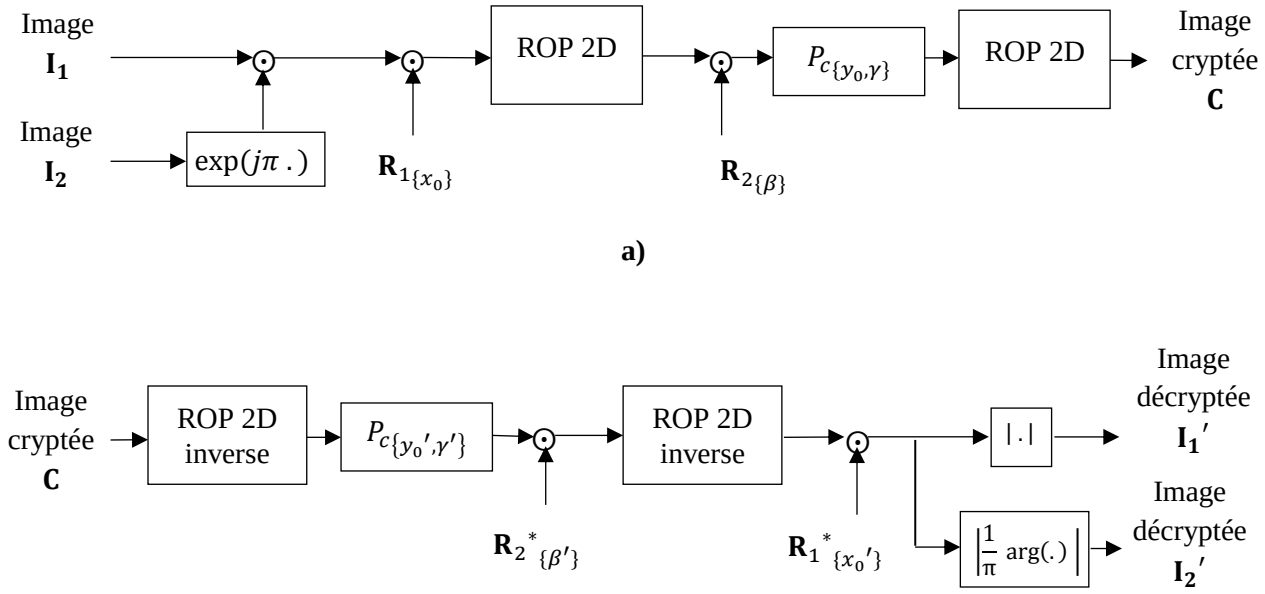


Figure 3.10 Méthode proposée de cryptage de deux images en utilisant la transformée ROP, a) algorithme de cryptage, b) algorithme de décryptage

Soit $\mathbf{R}_{1\{x_0\}} = e^{j2\pi\, x_{m,n}}, 1 \le m \le 2N,\ 1 \le n \le 2N$ un masque CRPM de taille $2N \times 2N$ généré de manière similaire que dans [34]. $x_{m,n}$ est un élément d'une matrice de nombres réels générée en utilisant l'équation (1.3) de la suite logistique avec une condition initiale $x_0 \in (0,1)$ et un paramètre de contrôle α égale à 4.

Soit $\mathbf{R}_{2\{\beta\}} = e^{j2\pi\, z_{m,n}}, 1 \le m \le 2N,\ 1 \le n \le 2N$, un autre masque CRPM, où $z_{m,n}$ est un élément d'une matrice de nombres réels générée en utilisant l'équation (1.3) de la suite logistique. Dans ce cas, le paramètre de contrôle est considéré variable $\beta \cong 4$ et la condition initiale $z_0 = x_{2N,2N}$ est égale à la dernière itération de la suite logistique utilisée pour la génération du premier masque $\mathbf{R}_{1\{x_0\}}$. Cela a pour but d'améliorer la diffusion de l'erreur dans la méthode proposée afin d'améliorer la sécurité du cryptage.

Soient $\mathbf{T^1}_{2N}, \mathbf{T^2}_{2N}, \mathbf{T^3}_{2N}$, et $\mathbf{T^4}_{2N}$ des matrices ROP d'ordre $2N$ construites en utilisant l'équation (2.14) de la transformée ROP avec $N\, log_2\left(\frac{N}{2}\right) + 1$ paramètres indépendants choisis aléatoirement du plan complexe.

Pour crypter une paire d'images $\mathbf{I_1}$ et $\mathbf{I_2}$ de taille $2N \times 2N$ en utilisant la méthode proposée, les étapes suivantes doivent être suivies :

1) Former une matrice complexe $\mathbf{M} = \mathbf{A}\, e^{j\,\boldsymbol{\varphi}}$, où $\mathbf{A} = \mathbf{I_1}$ est le module de la matrice complexe, et $\boldsymbol{\varphi} = \pi\,.\,\mathbf{I_2}$ est la phase de la matrice complexe.

2) Multiplier la matrice complexe $\mathbf{M}$ élément-par-élément avec le premier masque CRPM $\mathbf{R}_{1\{x_0\}}$.

3) Appliquer une transformée ROP 2D sur le résultat du produit en utilisant les matrices ROP $\mathbf{T^1}_{2N}$ et $\mathbf{T^2}_{2N}$ suivant une structure lignes-colonnes.

4) Multiplier la matrice obtenue dans le domaine de la transformée ROP par le second masque CRPM $\mathbf{R}_{2\{\beta\}}$.

5) Effectuer une permutation complexe sur le produit résultant en utilisant la fonction de permutation complexe $P_{c\{\,y_0,\gamma\}}$ avec $y_0 \in (0,1)$ et $\gamma \cong 4$.

6) Appliquer une autre transformée ROP 2D sur la matrice résultante de l'étape précédente en utilisant les matrices ROP $\mathbf{T^3}_{2N}$ et $\mathbf{T^4}_{2N}$ suivant une structure lignes-colonnes.

Les étapes précédentes peuvent être résumées par l'équation suivante :

$$\mathbf{C} = \frac{1}{4\mathrm{N}^2}\left(\mathbf{T^4}_{2N}\;\; P_{c\{\,y_0,\gamma\}}\left[\left(\mathbf{T^2}_{2N}\,(\mathbf{M}\;\odot\;\mathbf{R}_{1\{x_0\}})\;\;\mathbf{T^1}_{2N}\right)\odot\;\mathbf{R}_{2\{\beta\}}\right]\;\;\mathbf{T^3}_{2N}\right) \qquad (3.4)$$

où $\odot$ indique l'opération de multiplication élément-par-élément.

Dans la méthode proposée, la clé secrète que l'on note K est composée des $4\left(N\, log_2\left(\frac{N}{2}\right) + 1\right)$ paramètres indépendants des matrices ROP $\mathbf{T^1}_{2N}$, $\mathbf{T^2}_{2N}$,$\mathbf{T^3}_{2N}$, et $\mathbf{T^4}_{2N}$, des paramètres x_0,

et β des masques CRPM $\mathbf{R}_{1\{x_0\}}$, $\mathbf{R}_{2\{\beta\}}$, et des paramètres y_0 et γ de la fonction de permutation complexe $P_{c\{y_0,\gamma\}}$.

A présent, supposons que nous avons une clé secrète K' composée des paramètres x_0', et β' des masques CRPM $\mathbf{R}_{1\{x_0'\}}$, $\mathbf{R}_{2\{\beta'\}}$, des paramètres y_0' et γ' de la fonction de permutation complexe $P_{c\{y_0',\gamma'\}}$, et de $4\left(N\ log_2\left(\frac{N}{2}\right)+1\right)$ paramètres indépendants des matrices ROP $\mathbf{T}^{1'}{}_{2N}$, $\mathbf{T}^{2'}{}_{2N}$,$\mathbf{T}^{3'}{}_{2N}$, et $\mathbf{T}^{4'}{}_{2N}$ d'ordre $2N$.

De ce fait, l'image cryptée $\mathbf{C}$ est décryptée avec la clé secrète K' suivant l'équation suivante

$$\mathbf{M}' = \frac{1}{4N^2}\ \mathbf{R}^*{}_{1\{x_0'\}} \odot \left[\left(\mathbf{T}^{2'}{}_{2N}\right)^{\mathrm{RT}} \left(P_{c\ \{y_0',\gamma'\}}\left[\left(\mathbf{T}^{4'}{}_{2N}\right)^{\mathrm{RT}}\ \mathbf{C}\ \left(\mathbf{T}^{3'}{}_{2N}\right)^{\mathrm{RT}}\right] \odot \mathbf{R}^*{}_{2\{\beta'\}}\right)\left(\mathbf{T}^{1'}{}_{2N}\right)^{\mathrm{RT}}\right] \quad (3.5)$$

où $\mathbf{M}' = \mathbf{A}'\ e^{j\,\boldsymbol{\varphi}'}$ est l'image complexe décryptée, $(.)^{\mathrm{RT}}$ l'opération de la réciproque-transpose, et $(.)^*$ le conjugué complexe.

L'image décryptée $\mathbf{I_1}'$ est obtenue en prenant le module de $\mathbf{M}'$ où $\mathbf{I_1}' = |\mathbf{M}'|$, et l'image décryptée $\mathbf{I_2}'$ est obtenue en prenant le module de l'argument ou la phase de $\mathbf{M}'$, où $arg(\mathbf{M}') = \boldsymbol{\varphi}'$, et $\mathbf{I_2}' = \left|\frac{1}{\pi}\boldsymbol{\varphi}'\right|$

Enfin, la méthode proposée est une méthode de cryptage symétrique, car l'image décryptée $\mathbf{I_1}'$ et l'image décryptée $\mathbf{I_2}'$ sont correctement décryptées seulement si ${y_0}' = y_0$, $\gamma = \gamma'$, ${x_0}' = x_0$, $\beta' = \beta$,et $\mathbf{T}^{n'}{}_{2N} = \mathbf{T}^{n}{}_{2N}$, où $n = 1, 2, 3, 4$.

3.3.2.3 Résultats et discussions

Dans cette section, nous présentons les résultats de simulation de la méthode proposée avec des images de tests standards de niveau gris (8bits). Une évaluation détaillée de sa sécurité est également présentée.

Soit une paire d'images de test de taille identique 256×256. Soit K une clé secrète définit par :

- Les paramètres de la fonction de permutation complexe : $y_0 = 0.371$ et $\gamma = 3.983$.
- Les paramètres des deux masques CRPM : $x_0 = 0.7$ et $\beta = 3.944$.

- Pour $2N = 256$, les paramètres indépendants des matrices ROP constitues de $4\left(N\ log_2\left(\frac{N}{2}\right) + 1\right) = 3076$ paramètres aléatoirement choisis du plan complexe.

Pour une paire d'images réelles Barbara/Mandrill cryptée avec la clé secrète K, les résultats de simulation sont illustrés dans la figure 3.11.

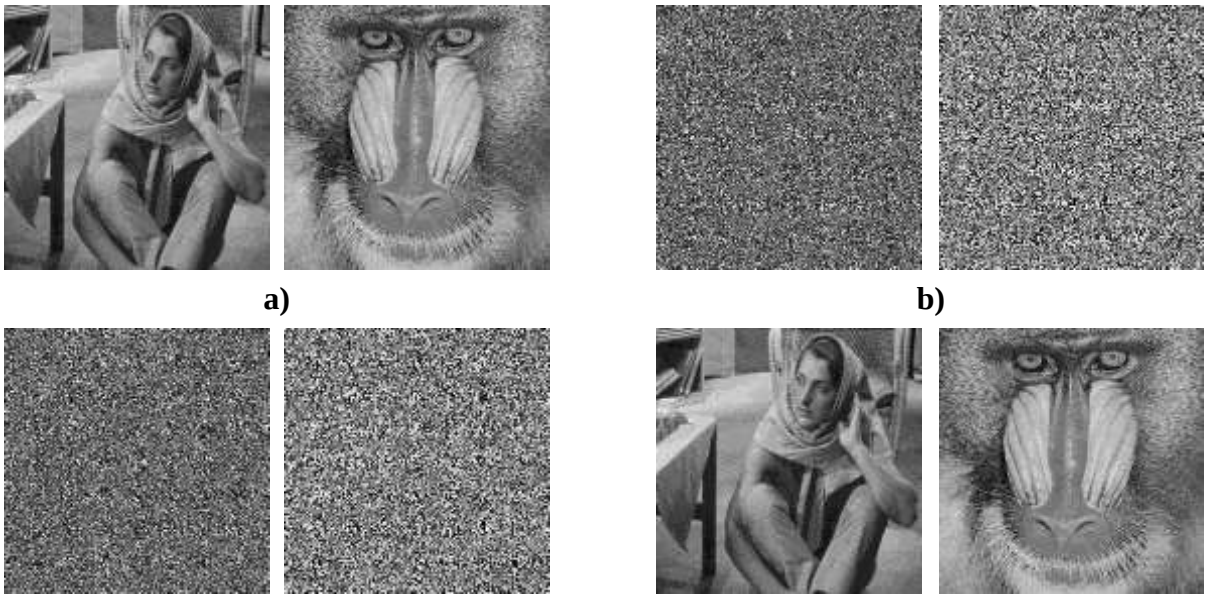


Figure 3.11 Résultats de simulation: a) l'image complexe originale, b) l'image complexe cryptée, c) et d) l'image complexe décryptée avec une clé incorrecte et avec une clé correcte.

La figure 3.11(a) montre le module (Barbara) et la phase (Mandrill) de l'image complexe originale. La figure 3.11(b) montre le module et la phase de l'image complexe cryptée. La figure 3.11(c) montre l'image complexe décryptée avec une clé secrète incorrecte et la figure 3.11(d) montre la paire d'images réelles Barbara/Mandrill correctement décryptée.

Nous constatons que la paire d'images réelles (Barbara/Mandrill) est correctement cryptée dans une image complexe. De plus, aucune des deux images réelles ne peut être décryptée sans la bonne clé secrète. Cela démontre que la méthode de cryptage proposée a une sécurité perceptuelle considérable.

Afin d'évaluer la qualité du cryptage de la méthode proposée de manière objective, nous calculons le coefficient de corrélation c_m entre le module de l'image complexe cryptée et l'image

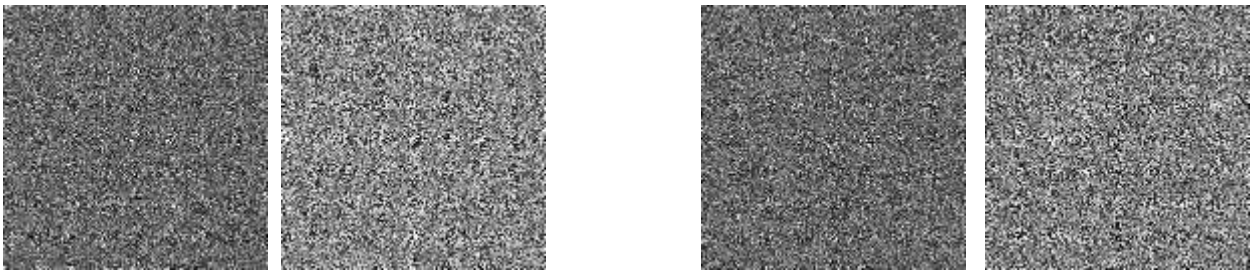
complexe originale. Ensuite, nous calculons le coefficient de corrélation c_p entre la phase de l'image complexe cryptée et l'image complexe originale. Les résultats sont illustrés dans le tableau 3.2 en considérant plusieurs paires d'images de test.

Tableau 3.2 Coefficient de corrélation de plusieurs paires d'images de tests.

Paire d'images	c_m	c_p
Barbara/Mandrill	- 0.0026	- 0.0016
House/ Lenna	- 0.0019	- 0.0004
Boat/Cameraman	- 0.0021	- 0.0006
Aerial/Bridge	- 0.0001	0.0001

Nous remarquons que dans tous les cas, le coefficient de corrélation est proche de zéro. Cela démontre de manière objective la qualité de cryptage de la méthode proposée.

A présent, pour évaluer la sensibilité de la clé secrète, nous supposons que la paire d'images Barbara/Mandrill précédente est décryptée avec une clé secrète K' identique à la clé K mais qui comporte une erreur dans l'un de ses différents paramètres. Les figures 3.12(a) et 3.12(b) montrent la paire d'images Barbara/Mandrill décryptée lorsque les paramètres de la permutation $y_0' = y_0 + 10^{-16}$, et $\gamma_0' = \gamma_0 + 10^{-15}$.



a) **b)**

Figure 3.12 Paire d'images décryptée en fonction des paramètres de la fonction de permutation complexe: a) $y_0' = y_0 + 10^{-16}$, b) $\gamma_0' = \gamma_0 + 10^{-15}$.

De plus, les figures 3.13(a) et 3.13(b) montrent la paire d'images Barbara/Mandrill décryptée lorsque les paramètres des masques CRPM sont égale à $x_0' = x_0 + 10^{-16}$, et $\beta_0' = \beta_0 + 10^{-15}$.

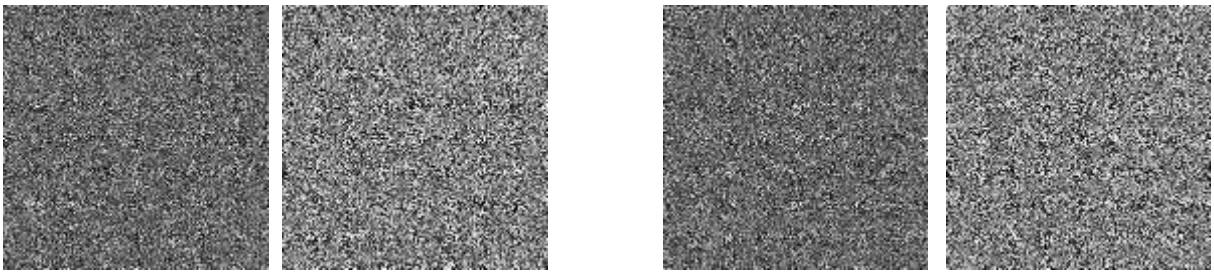


Figure 3.13 Paire d'images décryptée en fonction des paramètres des masques CRPM: a) $x_0' = x_0 + 10^{-16}$, b) $\beta_0' = \beta_0 + 10^{-15}$.

Nous remarquons que la clé secrète dans la méthode proposée est très sensible aux erreurs dans les paramètres de la fonction de permutation complexe proposée de même que les paramètres des deux masques CRPM qui sont dépendant l'un de l'autre. Pour déterminer le degré de sensibilité de la clé sécrète aux paramètres de la transformée ROP, la figure 3.14 montre la paire d'images Barbara/Mandrill décryptée lorsque 12% des paramètres des matrices ROP sont incorrects. Dans ce cas aussi,nous remarquons que les deux images restent correctement cryptées. Ces résultats démontrent la capacité de confusion et de diffusion de la méthode proposée.

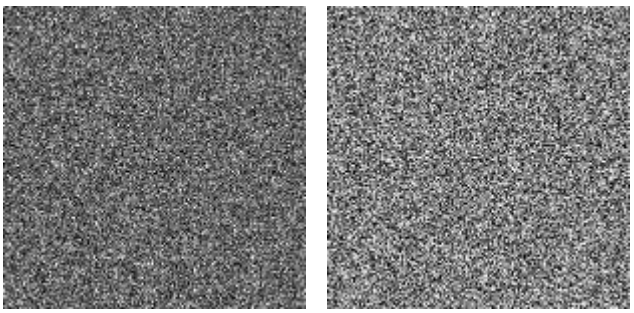


Figure 3.14 Paire d'images décryptée avec 12% des paramètres ROP incorrects.

Comme la méthode présentée dans la section précédente pour le cas d'une seule image est basée sur des permutations chaotiques, nous proposons de comparer la méthode proposée dans le cas de deux images avec la méthode proposée dans le cas d'une seule image. Ainsi, nous avons calculé l'EQM entre l'image décryptée et l'image originale en fonction d'une erreur de déviation

δ dans les paramètres des matrices ROP dans les deux cas. Les résultats de calcul de l'EQM sont tracés dans la figure 3.15.

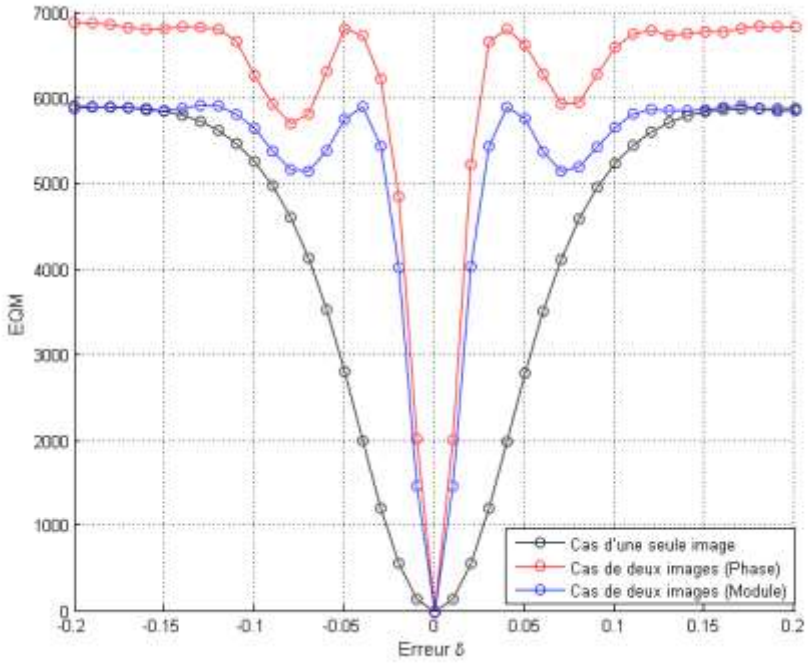
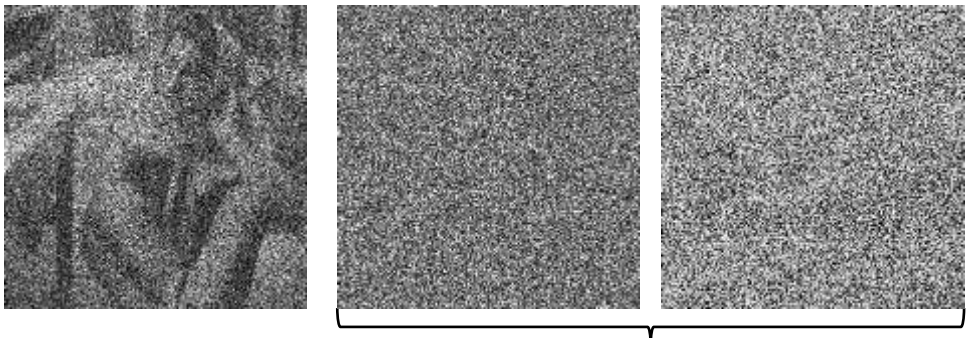


Figure 3.15 Comparaison de l'EQM en fonction d'une erreur δ dans les paramètres ROP.

Nous constatons que la méthode proposée dans le cas de deux images possède une clé secrète plus sensible aux erreurs dans les paramètres des matrices ROP que la méthode proposée dans le cas d'une seule image. Aussi, pour un seuil d'erreur fixe, la figure 3.16 montre la paire d'images décryptée lorsque $\delta = 0.05$ dans la méthode de Bouguezel et al. [26].

Ainsi, ces résultats démontrent l'efficacité de la méthode proposée. A présent, l'espace réel de la clé peut être estimé et qui est de l'ordre de $20^{3076} \times 10^{16} \times 10^{15} \times 10^{16} \times 10^{15}$ combinaisons de clés possibles. Ce résultat est largement supérieur au minimum recommandé de 2^{100}. En conséquence, la méthode proposée est robuste contre les attaques par force brute.



a) b)

Figure 3.16 Images décryptée avec une erreur $\delta = 0.05$ dans les paramètres ROP, a) Méthode de Bouguezel et al. , b) méthode proposée (cas de deux images).

Afin d'évaluer la sécurité de la méthode proposée contre l'analyse statistique par histogramme, la figure 3.17(a) montre les histogrammes d'une image Barbara et une image Mandrill considérées comme le module et la phase d'une image complexe. La figure 3.17(b) montre les histogrammes du module et de la phase de l'image complexe cryptée.

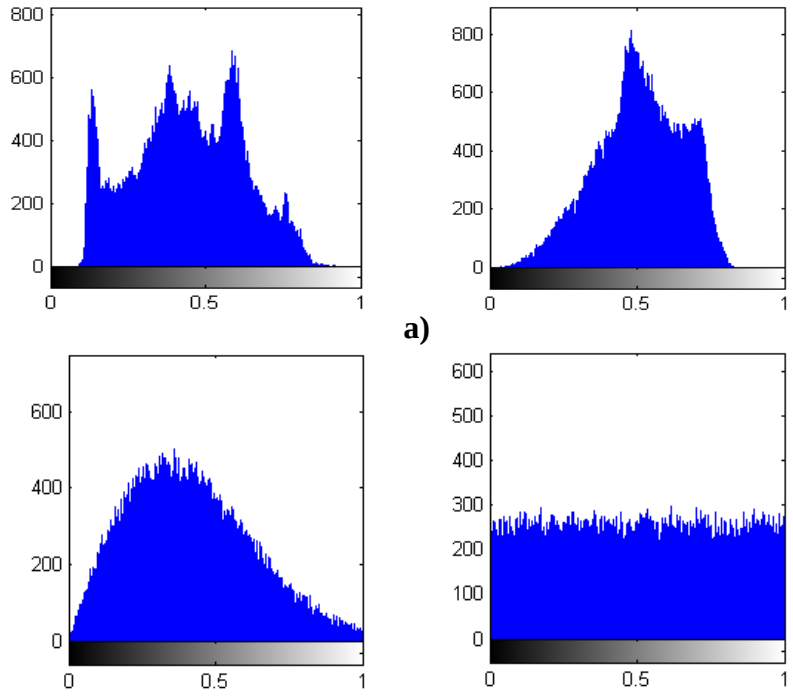


Figure 3.17 Histogrammes d'une paire d'images: a) paire d'image originale, b) paire d'images cryptée.

Nous remarquons que l'histogramme du module et l'histogramme de la phase de l'image complexe cryptée sont différents de ceux de l'image complexe originale. Les mêmes résultats ont été obtenus avec d'autres images de tests. En conséquence, la méthode proposée est robuste contre l'analyse statistique par histogramme.

Pour vérifier la résistance de la méthode proposée contre le bruit additif, nous avons calculé pour une paire d'images Barbara/Mandrill l'évolution de l'EQM en fonction du coefficient de puissance du bruit σ. Les résultats de simulations sont illustrés dans la figure 3.18.

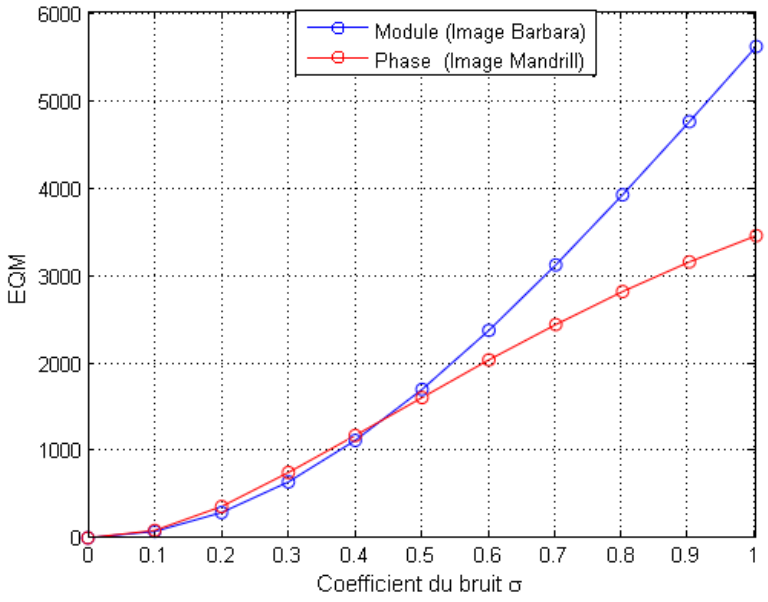


Figure 3.18 EQM du module et de la phase en fonction du coefficient du bruit additif σ.

Nous remarquons que lorsque $\sigma > 0.5$, l'image réelle cryptée en tant que module est plus susceptible aux erreurs du bruit additif que l'image cryptée en tant que phase, néanmoins, une vérification visuelle de ces résultats s'impose. Pour cela, la figure 3.19 montre la paire d'images Barbara/Mandrill avec son PSNR correspondant lorsque le coefficient du bruit σ est égal à 0.5 et 0.7, respectivement. Nous observons que l'image Barbara ainsi que l'image Mandrill reste reconnaissable malgré la présence d'un bruit additif. Par conséquent, ces résultats démontrent la résistance de la méthode proposée contre le bruit additif.

$\sigma = 0.5$ PSNR Module = 16dB PSNR Phase = 16dB

$\sigma = 0.7$ PSNR Module = 13dB PSNR Phase = 14dB

Figure 3.19 Paire d'images décryptée en fonction du coefficient du bruit additif σ.

Enfin, pour tester la résistance de la méthode proposée contre les erreurs de transmission, nous supposons qu'une partie des pixels de l'image cryptée a été perdue au cours du transit dans le canal de communication. La paire d'images Barbara /Mandrill décryptée est donc illustrée avec son PSNR dans la figure 3.20.

D'après ces résultats, nous remarquons que les deux images restent reconnaissables à un certain degré malgré que l'image complexe ait perdue une partie de ses pixels. Ainsi, ces résultats démontrent la résistance de la méthode proposée contre les erreurs de transmission.

PSNR Module = 16dB PSNR Phase = 15dB

Figure 3.20 Paire d'images décryptée lorsqu'une partie des pixels a été perdue.

3.4 Conclusion

Dans ce chapitre, nous avons proposé une méthode de cryptage collectif des images en utilisant la transformée ROP et une fonction de permutation complexe. Cette méthode a été développée au début dans le cas du cryptage d'une seule image par l'utilisation de la transformée ROP avec une permutation chaotique connue, s'en est suivi un élargissement d'idée dans le cas du cryptage de deux images en même temps, où nous avons proposé d'utiliser une fonction de permutation chaotique complexe réversible. Les résultats de simulation ont montré l'efficacité et la robustesse de la méthode proposée contre les attaques par force brute, l'analyse statistique par histogramme et la résistance contre le bruit additif et les erreurs de transmission. De plus, les résultats de comparaison montrent clairement que la méthode proposée est plus efficace que les méthodes existantes basées sur la transformée ROP en termes de sensibilité et d'espace de la clé secrète.

Chapitre 4

Proposition d'un nouveau prétraitement non linéaire pour le cryptage d'images

4.1 Introduction

La méthode de cryptage DRPE reste une méthode linéaire malgré l'utilisation des transformées paramétriques, car ces transformées sont également considérées comme linéaires. Cette linéarité laisse la méthode DRPE vulnérable à certaines attaques complexes en cryptanalyse [53]-[63].

Plusieurs méthodes de cryptage basées sur une structure modifiée de la méthode DRPE classique ont été développées afin d'améliorer sa sécurité [33]-[43]. Ces méthodes sont en général basées sur l'utilisation d'un cryptage hybride opto-numérique qui consiste à remplacer les masques de phases aléatoires de la méthode DRPE classique par des fonctions de cryptage numérique. Hennelly et al. ont proposé dans [33] de remplacer les masques de phases aléatoires dans la méthode DRPE par une fonction de permutation secrète dans le domaine de la transformée TFR. Liu et al. ont proposé dans [42] d'améliorer l'espace de la clé de la fonction de permutation précédente en utilisant la suite du chat d'Arnold qui est une suite chaotique. De plus, Lang et al. ont proposé dans [36] une méthode de cryptage hybride opto-numérique basée sur la transformée TFRD à paramètres multiples en remplaçant les masques de phases aléatoires par deux fonctions de permutation chaotiques basées sur les suites logistiques. Bien que les fonctions de permutation permettent d'améliorer la sécurité du cryptage DRPE, elles restent aussi des fonctions linéaires qui peuvent être réduites à une simple matrice de permutation quel que soit le type de permutation [63], [64].

Dans ce chapitre, nous proposons un nouveau prétraitement non-linéaire pour le cryptage DRPE afin de résoudre le problème de sa linéarité et renforcer sa sécurité. Le prétraitement non-linéaire est introduit dans le domaine spatial avant l'application d'une méthode de cryptage DRPE en utilisant une combinaison de suites chaotiques associées à une fonction OU exclusif (XOR).

Vu que la méthode de cryptage DRPE peut être réalisée numériquement et optiquement, ce chapitre sera divisé en deux parties. Dans la première, nous proposons d'introduire l'idée du prétraitement non-linéaire pour le cas de la méthode DRPE dans le domaine de la transformée ROP en utilisant un cryptage purement numérique [65]. Quant à la deuxième partie, nous proposons d'élargir cette idée au cryptage hybride opto-numérique en utilisant la transformée TFRD à paramètres multiples et des suites PLCM [66].

4.2 Cryptage numérique dans le domaine de la transformée ROP

4.2.1 Description de la méthode

Le masque spatial de la méthode DRPE dans le domaine de la transformée ROP [26] n'a pas d'impact réel sur la sécurité, car il est négligé lors de l'étape de décryptage en prenant simplement le module. De ce fait, nous proposons une méthode de cryptage DRPE dans le domaine de la transformée ROP suivant une structure modifiée. Pour cela, nous remplacerons ce masque de phases aléatoires par un prétraitement non linéaire dans le domaine spatial afin d'améliorer la sécurité du cryptage. Ce prétraitement non-linéaire est réalisé sur l'image en entrée en utilisant une matrice chaotique construite par une combinaison de suites logistiques. Cette matrice chaotique est ensuite associée à l'image en entrée en utilisant une opération logique XOR. Il faut noter que dans cette section nous avons utilisé la transformée ROP dans sa nouvelle définition [26] au lieu de l'ancienne définition [51] utilisée dans notre communication [65].

4.2.1.1 Construction de la matrice chaotique

Une matrice chaotique est appelée ainsi, car elle est construite en utilisant des suites chaotiques. De ce fait, nous proposons de construire une matrice chaotique **M** de taille $2N \times 2N$ selon les étapes suivantes:

1) Créer un vecteur de $4N^2$ nombres réels aléatoires que l'on note **X** en utilisant l'équation (1.3) de la suite logistique avec une condition initiale $x_0 \in (0,1)$, et un paramètre de contrôle $\mu \cong 4$ afin d'avoir une suite entièrement chaotique. De la même manière, créer un autre vecteur de $4N^2$ nombres réels que l'on note **Y** en générant une suite logistique avec un paramètre de contrôle $\rho \cong 4$ et une condition initiale y_0 égale à la dernière valeur d'itération du vecteur **X**, où $y_0 = x_{4N^2}$.

2) Convertir les éléments des deux vecteurs **X** et **Y** en nombres entiers de 8bits, puis former un troisième vecteur **Z** en effectuant une opération logique XOR entre les éléments des deux vecteurs de sorte que $z_n = x_n \oplus y_n,\ 1 \leq n \leq 4N^2$.

3) Redimensionner le vecteur **Z** obtenu lors de l'étape précédente en une matrice chaotique de taille $2N \times 2N$ que l'on note **M**.

Notez que ces étapes peuvent être répétées autant de fois afin d'améliorer la sécurité, cependant, un compromis rapidité de calcul / sécurité de cryptage doit être considéré.

4.2.1.2 Algorithmes de cryptage et de décryptage

La méthode proposée est illustrée dans la figure 4.1.

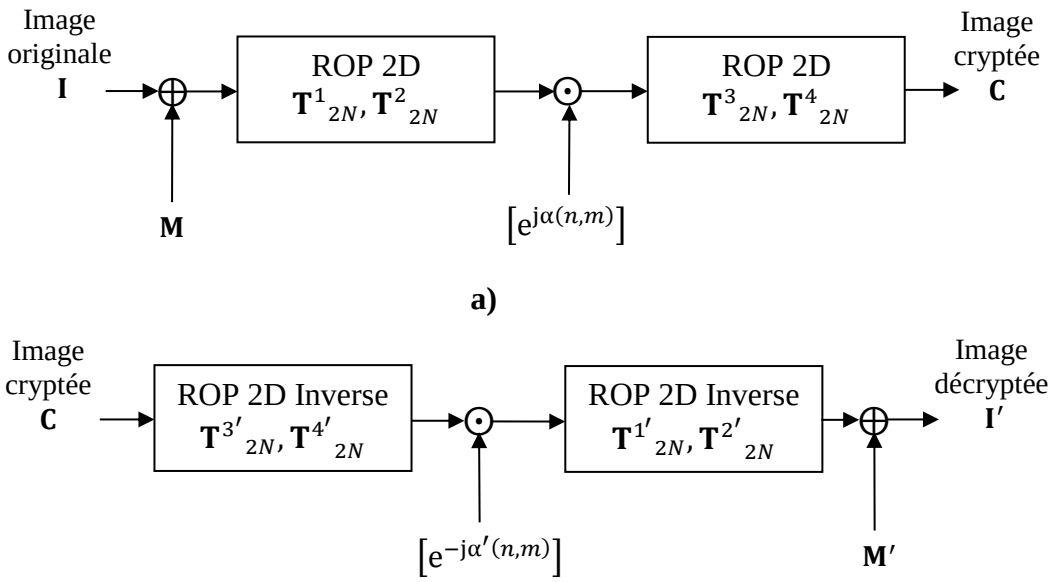


Figure 4.1 Méthode proposée de cryptage d'image basée sur la transformée ROP et un prétraitement non-linéaire, a) algorithme de cryptage, b) algorithme de décryptage.

Soit une matrice chaotique **M** de taille $N \times N$ construite selon les étapes décrites dans la section précédente avec une condition initiale $x_0 \in (0,1)$, et des paramètres de contrôle $\mu \cong 4$, et $\rho \cong 4$.

Soient $\mathbf{T}^1{}_{2N}$, $\mathbf{T}^2{}_{2N}$, $\mathbf{T}^3{}_{2N}$, et $\mathbf{T}^4{}_{2N}$ des matrices ROP d'ordre $2N$ construites en utilisant l'équation (2.14) de la transformée ROP avec $N\, log_2\left(\frac{N}{2}\right) + 1$ paramètres indépendants choisis aléatoirement du plan complexe. Soit une image **I** en forme de matrice de taille $2N \times 2N$ et $[e^{j\alpha(n,m)}]$ un masque de phases aléatoires de taille équivalente à la taille de l'image **I**, où $\alpha(x, y)$ est une fonction ayant une distribution aléatoire et uniforme dans l'intervalle $[0,2\pi]$.

Ainsi, les étapes du cryptage numérique de l'image **I** sont décrites dans ce qui suit:

1) Effectuer un prétraitement non-linéaire sur l'image **I** en utilisant une opération OU logique (XOR) entre la matrice chaotique **M** et l'image **I** comme suit :

$$\mathbf{R}(n,m) = \mathbf{M}(n,m) \oplus \mathbf{I}(n,m),\ 0 \le n,m \le N-1 \tag{4.1}$$

2) Appliquer la transformée ROP 2D sur la matrice **R** résultante de l'étape précédente en utilisant les matrices ROP $\mathbf{T^1}_{2N}$ et $\mathbf{T^2}_{2N}$ suivant une configuration lignes-colonnes.

3) Multiplier la matrice résultante élément par élément avec le masque de phases aléatoires $\left[e^{j\alpha(n,m)}\right]$.

4) Appliquer une autre transformée ROP 2D sur le résultat du produit en utilisant les matrices ROP $\mathbf{T^3}_{2N}$ et $\mathbf{T^4}_{2N}$ suivant une configuration lignes-colonnes.

Ainsi, nous obtenons une image cryptée d'amplitude complexe que l'on note **C**. Ces étapes de cryptage peuvent être résumées par l'équation suivante :

$$\mathbf{C} = \frac{1}{4N^2}\left(\mathbf{T^4}_{2N}\left(\left[\mathbf{T^2}_{2N}\ \mathbf{R}\ \mathbf{T^1}_{2N}\right] \odot \left[e^{j\alpha(n,m)}\right]\right)\mathbf{T^3}_{2N}\right) \tag{4.2}$$

où $\odot$ indique l'opération de multiplication élément-par-élément.

De ce fait, la clé secrète de cryptage que l'on note K est composée des $4N^2$ phases aléatoires du masque $\left[e^{j\alpha(n,m)}\right]$, des $4\left(N\ log_2\left(\frac{N}{2}\right)+1\right)$ paramètres indépendants des matrices ROP $\mathbf{T^1}_{2N}$, $\mathbf{T^2}_{2N}$,$\mathbf{T^3}_{2N}$, $\mathbf{T^4}_{2N}$ et des paramètres x_0, μ , et ρ de la matrice chaotique **M** utilisée lors du prétraitement non-linéaire.

Supposons à présent qu'on ait une clé secrète K' composée d'un masque $\left[e^{j\alpha'(n,m)}\right]$ de $4N^2$ phases aléatoires, des matrices ROP $\mathbf{T^{1'}}_{2N}$, $\mathbf{T^{2'}}_{2N}$,$\mathbf{T^{3'}}_{2N}$, $\mathbf{T^{4'}}_{2N}$ de $4\left(N\ log_2\left(\frac{N}{2}\right)+1\right)$ paramètres indépendants et d'une matrice chaotique $\mathbf{M'}$ avec x'_0, μ' , et ρ' comme paramètres.

Ainsi, les étapes de décryptage de l'image cryptée **C** avec la clé secrète K' consistent à effectuer l'inverse des étapes précédentes de cryptage, de ce fait, les étapes de décryptage peuvent être résumées par l'équation suivante

$$\mathbf{R}' = \frac{1}{4N^2}\left(\mathbf{T}^{2'}{}_{2N}\right)^{\mathrm{RT}} \left(\left[\left(\mathbf{T}^{4'}{}_{2N}\right)^{\mathrm{RT}} \mathbf{C} \left(\mathbf{T}^{3'}{}_{2N}\right)^{\mathrm{RT}}\right] \odot \left[\mathrm{e}^{-\mathrm{j}\alpha'(n,m)}\right]\right) \left(\mathbf{T}^{1'}{}_{2N}\right)^{\mathrm{RT}} \quad (4.3)$$

où $(.)^{\mathrm{RT}}$ indique l'opération de la réciproque-transpose, cependant, l'image décryptée finale $\mathbf{I}'$ est obtenue seulement si on effectue un prétraitement non-linéaire inverse entre la matrice décryptée $\mathbf{R}'$ et la matrice chaotique $\mathbf{M}'$ en utilisant l'équation suivante :

$$\mathbf{I}'(n,m) = \mathbf{M}'(n,m) \oplus \mathbf{R}'(n,m),\ 0 \leq n,m \leq N-1 \quad (4.4)$$

Enfin, nous remarquons que la méthode proposée est celle d'un cryptage symétrique, car l'image finale décryptée $\mathbf{I}'$ est identique à l'image originale $\mathbf{I}$ seulement si $x_0{}' = x_0, \mu' = \mu$, $\rho' = \rho$, $\mathbf{T}^{k'}{}_{2N} = \mathbf{T}^{k}{}_{2N}$ avec $k = 1,2,3,4$, et $\alpha'(n,m) = \alpha(n,m)$ où $0 \leq n,m \leq N-1$.

4.2.1.3 Résultats et discussions

Dans cette section, nous présentons les résultats de simulation de la méthode proposée avec des images de tests standards de niveau gris (8bits). Aussi, nous élaborons une évaluation détaillée de sa sécurité.

Soit une image de test de taille 256×256, et une clé secrète K qui est définie par :

- Les paramètres de la matrice chaotique $\mathbf{M}$ utilisée pour le prétraitement non linéaire : $x_0 = 0.248$, $\mu = 3.997$, et $\rho = 3.951$.
- Les $4\left(N\ log_2\left(\frac{N}{2}\right) + 1\right) = 3076$ paramètres indépendants des matrices ROP aléatoirement choisis du plan complexe.
- Un masque de phases aléatoires $\left[\mathrm{e}^{\mathrm{j}\alpha(n,m)}\right]$ de taille 256×256.

Les résultats de simulation sont illustrés dans la figure 4.2 pour une image de test Lenna.

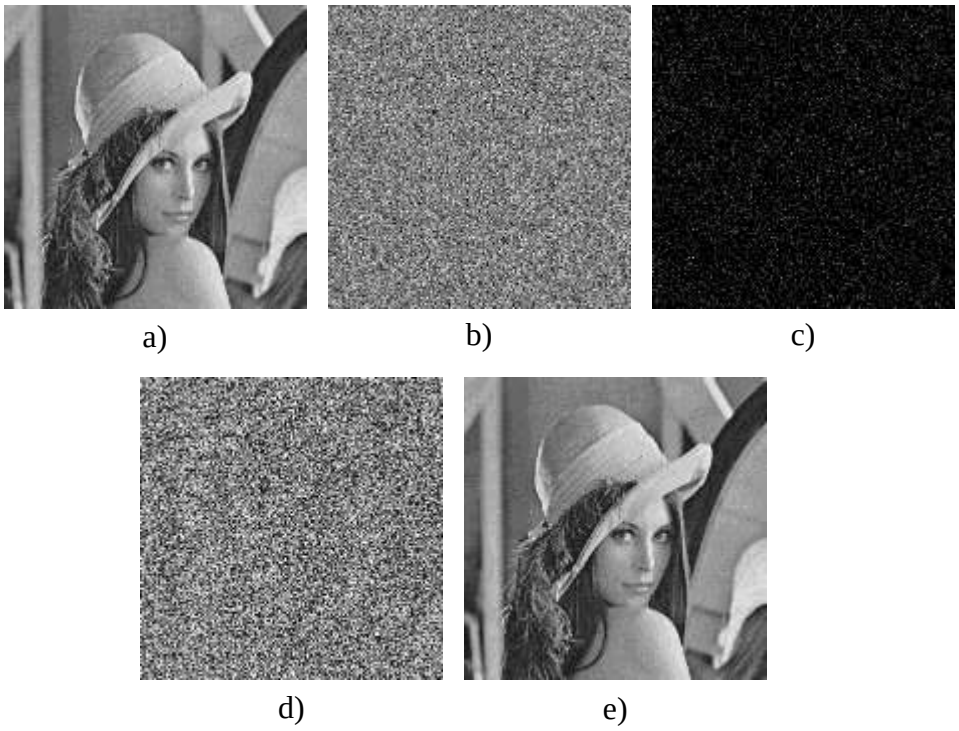


Figure 4.2 Résultats de simulation: a) image originale, b) et c) partie réelle et partie imaginaire de l'image cryptée, d) et e) image décryptée avec une clé incorrecte et une clé correcte.

La figure 4.2(a) montre l'image Lenna originale, puis les figures 4.2(b) et 4.2(c) montrent la partie réelle et la partie imaginaire de l'image cryptée. De plus, les figure 4.2(d) et 4.2(e) montrent l'image décryptée avec une clé secrète incorrecte et avec une clé secrète correcte, respectivement.

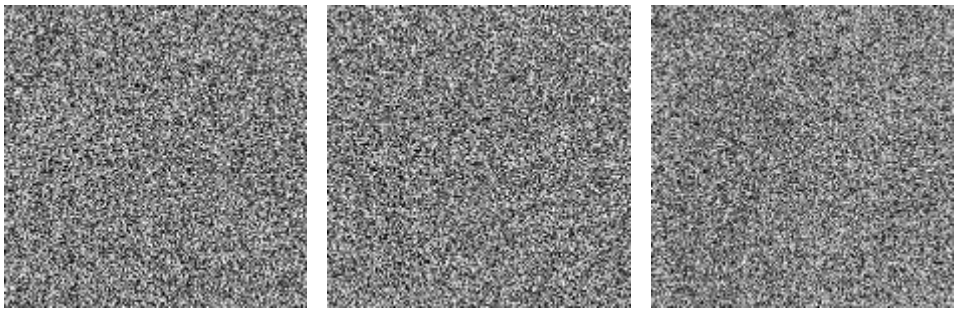
Nous remarquons que l'image ne peut être décryptée si on ne connait pas la clé secrète. De plus, visuellement, nous constatons que l'image est correctement cryptée et aucun détail visuel de l'image originale n'est visible sur l'image cryptée. Cela démontre que la méthode de cryptage proposée possède une sécurité perceptuelle satisfaisante, néanmoins, pour évaluer la qualité du cryptage de manière objective sur plusieurs images de tests, le tableau 4.1 montre les résultats du calcul du coefficient de corrélation c_r entre la partie réelle de l'image cryptée et celle originale et du coefficient de corrélation c_i entre la partie imaginaire de l'image cryptée et celle originale.

Tableau 4.1. Coefficient de corrélation entre l'image originale et l'image cryptée.

Image	$\mathbf{c_r}$	$\mathbf{c_i}$
Lenna	0.0006	-0.0015
Cameraman	-0.0004	0.0001
Barbara	-0.0004	- 0.0009
Mandrill	- 0.0018	- 0.0022
Aerial	0.0015	- 0.0001

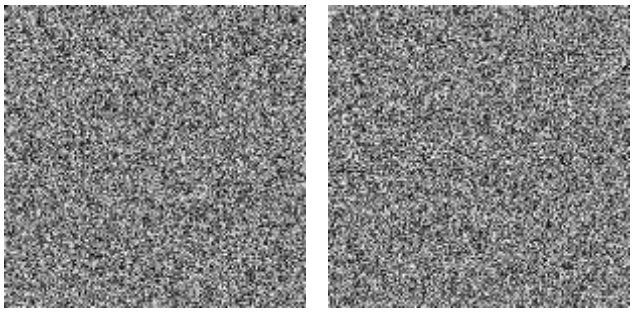
Nous remarquons que l'image cryptée avec la méthode proposée à une corrélation qui tend vers le zéro peu importe l'image originale, cela veut dire que l'image cryptée est indépendante de celle originale. Ainsi ces résultats démontrent de manière objective la qualité de cryptage de la méthode proposée.

Pour évaluer davantage la sécurité de la méthode proposée, la sensibilité de la clé secrète dans la méthode proposée doit être évaluée. Pour cela, supposons qu'une image Lenna soit cryptée avec la clé secrète K précédente, par la suite cette image est décryptée avec une clé secrète K' qui comporte une erreur dans l'un de ses différents paramètres. Ainsi, les figure 4.3(a), 4.3(b), et 4.3(c) montrent l'image Lenna décryptée lorsque les paramètres de la matrice chaotique sont, respectivement, $x_0' = x_0 + 10^{-16}$, $\mu' = \mu + 10^{-15}$, et $\rho' = \rho + 10^{-15}$.



a) b) c)

Figure 4.3 Image décryptée en fonction des paramètres de la matrice chaotique, a) $x_0' = x_0 + 10^{-16}$, b) $\mu' = \mu + 10^{-15}$, c) $\rho' = \rho + 10^{-15}$.



a) b)

Figure 4.4 Image décryptée en fonction des paramètres des matrices ROP, a) 50%, et b) 25% des paramètres ROP sont incorrects.

De plus, les figures 4.4(a) et 4.4(b) montrent l'image Lenna décryptée lorsque 50% et 25% des paramètres ROP sont incorrects. D'après ces figures, il est clair que la clé secrète est très sensible aux erreurs des paramètres du prétraitement non-linéaire proposé ainsi qu'aux paramètres des matrices ROP, cependant, pour illustrer l'intérêt du prétraitement non-linéaire dans la méthode proposée, nous comparons la sensibilité de la clé secrète dans le cas de la méthode proposée et dans le cas de la méthode de Bouguezel et al. [26] par le calcul de l'EQM entre l'image Lenna décryptée et celle originale en fonction d'une erreur de déviation δ variable introduite dans chacun des paramètres des matrices ROP. Les résultats sont illustrés dans la figure 4.5.

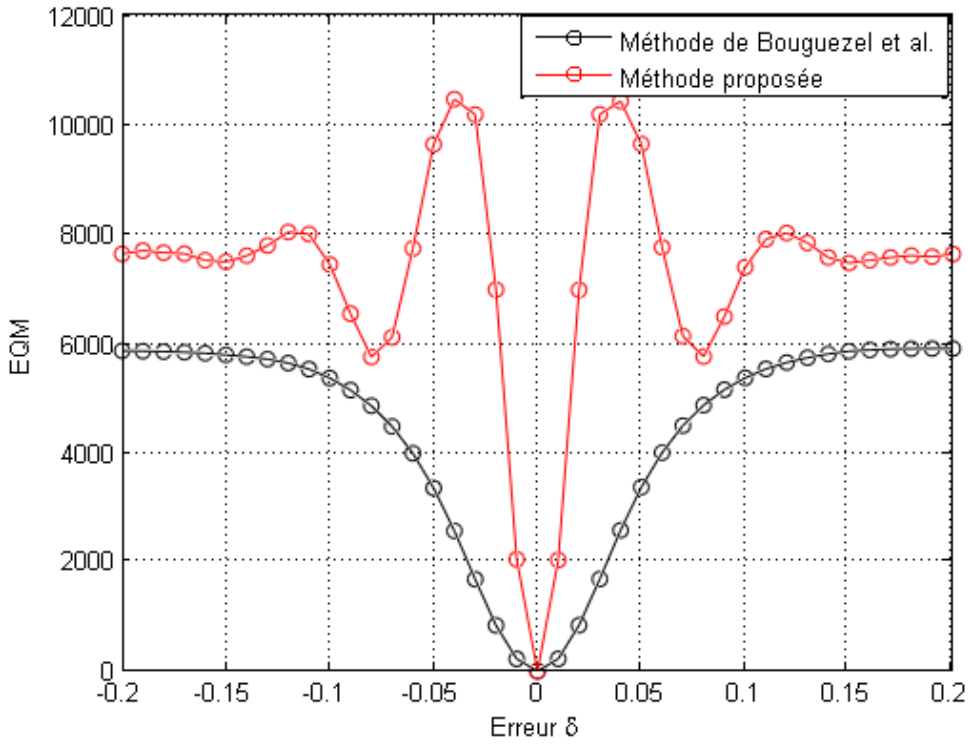
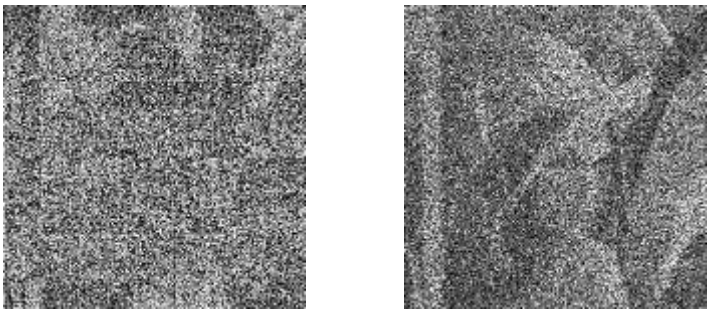


Figure 4.5 Comparaison de l'EQM en fonction d'une erreur δ dans les paramètres des matrices ROP

Nous constatons que dans la méthode proposée, la clé secrète est devenue plus sensible aux erreurs dans les paramètres des matrices ROP que dans la méthode linéaire proposée par Bouguezel et al. grâce au prétraitement non-linéaire proposée dans le domaine spatial. Par exemple, la figure 4.6 montre l'image Lenna décryptée avec un seuil d'erreur minimal de $\delta = 0.06$ dans le cas de la méthode proposée et dans le cas de la méthode de Bouguezel et al.

Ainsi, nous pouvons estimer que l'espace de la clé dans la méthode proposée est approximativement égale à $10^{3076} \times 10^{16} \times 10^{15} \times 10^{15}$ clés possibles ce qui est largement plus grand que le minimum 2^{100} pour résister à une attaque par force brute. En conséquence, la méthode proposée améliore significativement la sensibilité et l'espace de la clé secrète, et résiste aux attaques par force brute.



a) b)

Figure 4.6 Image décryptée en fonction d'une erreur minime $\delta = 0.06$ dans les paramètres des matrices ROP, a) méthode proposée, b) méthode de Bouguezel et al.

A présent, pour évaluer la sécurité de la méthode proposée contre l'analyse statistique par histogramme, la figure 4.7 montre les histogrammes de quelques images de tests, et ceux correspondants à leurs versions cryptées en utilisant la même clé de cryptage K. Vu que l'image cryptée est d'amplitude complexe, l'histogramme du module et celui de la phase sont illustrés séparément.

D'après ces figures, nous remarquons que les histogrammes de l'image cryptée sont aléatoires, identiques, et complétement différents de ceux des images originales. Cela permet de protéger l'image originale d'une éventuelle attaque statistique du fait qu'aucune information sur l'image originale ne peut être distinguée sur l'histogramme de l'image cryptée. En conséquence, ces résultats démontrent que la méthode proposée est robuste contre l'analyse statistique par histogramme.

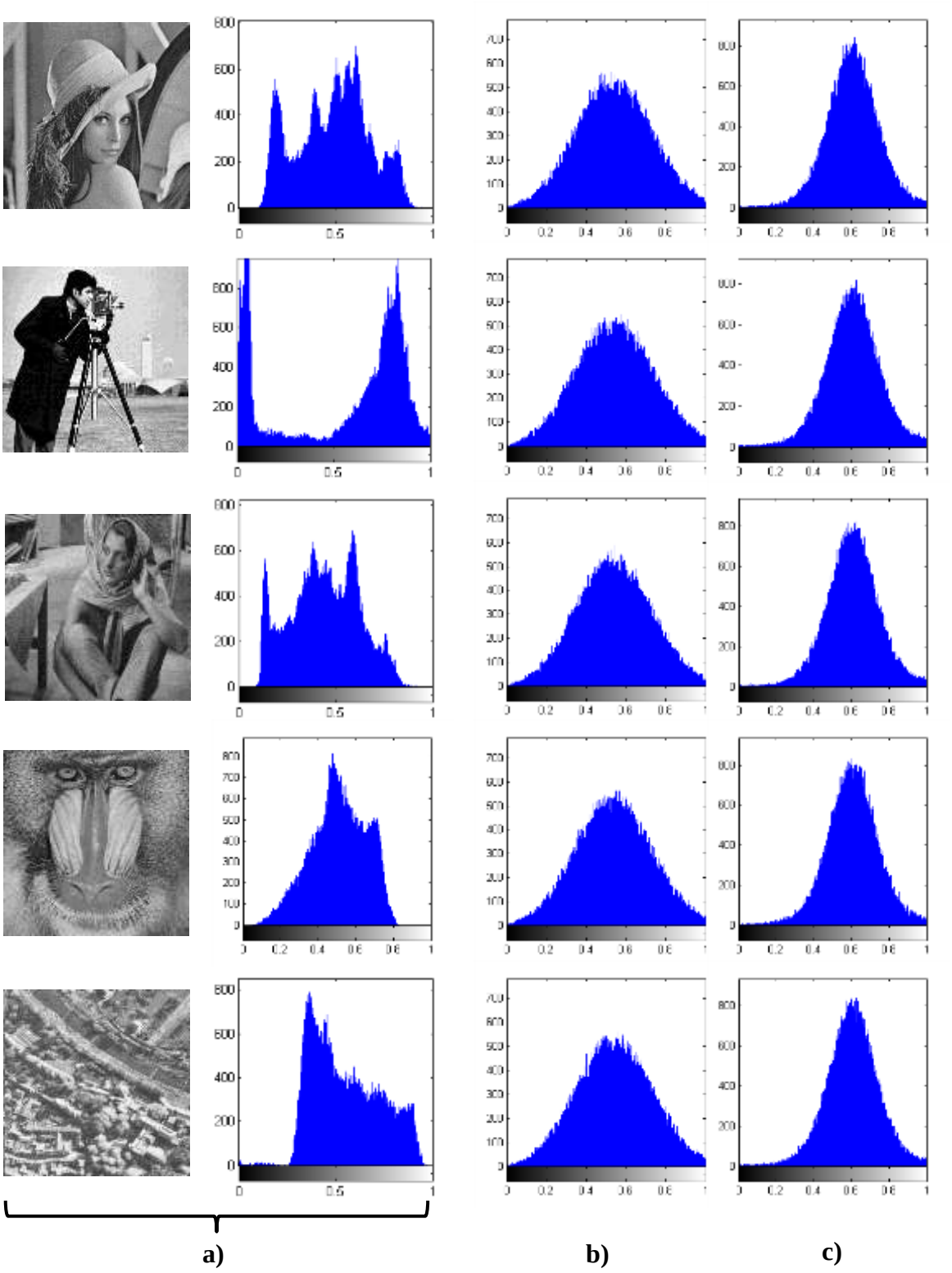


Figure 4.7 Histogrammes de quelques images de tests:

a) l'image originale, b) et c) le module et la phase de l'image cryptée.

Afin d'étudier la résistance de la méthode proposée contre un bruit blanc Gaussien additif, nous calculons l'EQM entre l'image Lenna originale et l'image Lenna décryptée en fonction du coefficient de puissance du bruit σ. Les résultats de simulations sont illustrés dans la figure 4.8.

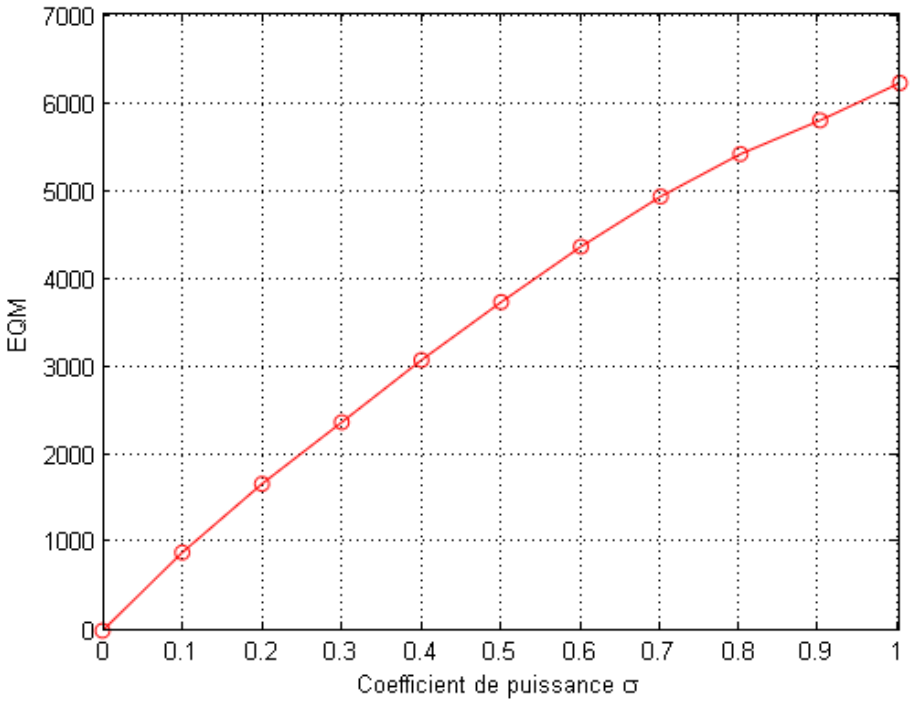


Figure 4.8 EQM en fonction du coefficient de puissance σ du bruit additif.

D'après cette figure, nous remarquons que l'erreur est proportionnelle au coefficient de puissance du bruit, cependant, il faudra effectuer une inspection visuelle de l'image Lenna décryptée en fonction des différents niveaux de puissance du bruit afin d'observer la dégradation résultante. Pour cela, la figure 4.9 montre l'image Lenna décryptée et son PSNR lorsque le coefficient du bruit σ est égal à 0.1, 0.3, 0.5, et 0.7, respectivement.

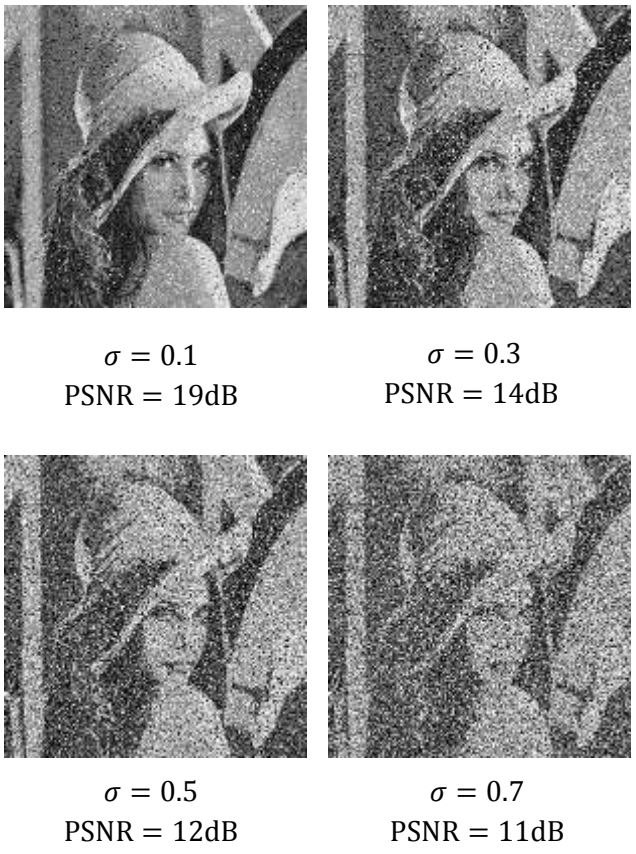
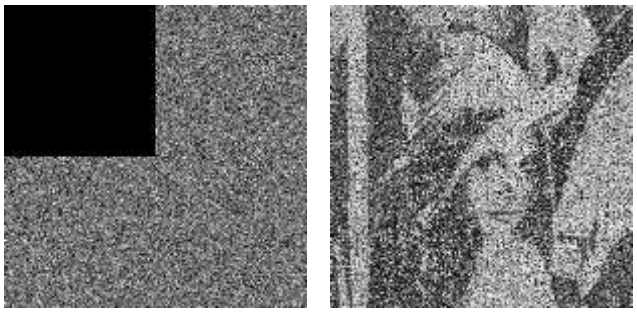


Figure 4.9 Image décryptée en fonction du coefficient du bruit additif.

Nous constatons que l'image originale Lenna reste reconnaissable à un certain point malgré la dégradation causée par le bruit. Ainsi, ces résultats démontrent la résistance de la méthode proposée contre le bruit additif.

Enfin, pour tester la résistance de la méthode proposée contre les erreurs de transmission, nous supposons qu'une partie des pixels de l'image Lenna cryptée a été perdu. L'image décryptée correspondante est illustrée avec son PSNR dans la figure 4.10.

Nous remarquons que l'image Lenna originale reste reconnaissable malgré que celle cryptée ait perdu une grande partie de ses pixels. En conséquence, ces résultats démontrent la résistance de la méthode proposée contre les erreurs de transmission.



PSNR = 12 dB

a) b)

Figure 4.10 Image décryptée après avoir perdu une partie des pixels:
a) image cryptée, b) image décryptée.

4.3 Cryptage opto-numérique dans le domaine de la transformée TFRD à paramètres multiples

4.3.1 Description de la méthode

Dans cette section, nous proposons dans [66] une nouvelle méthode de cryptage opto-numérique basée sur un prétraitement non-linéaire dans le domaine spatial, et des fonctions de permutation chaotique basées sur les suites PLCM. Ce prétraitement non-linéaire consiste à générer un vecteur de flux en utilisant une suite PLCM, puis ce vecteur de flux est associé à une opération logique XOR en entrée.

4.3.1.1 Fonction de permutation chaotique basée sur les suites PLCM

La fonction de permutation chaotique proposée par Lang et al. dans [36] est basée sur l'utilisation des suites logistiques. Nous proposons de modifier cette fonction de permutation en utilisant une suite PLCM au lieu d'une suite logistique afin d'améliorer la sécurité, car la suite logistique possède un paramètre de contrôle $\mu \epsilon (3.57, 4)$ qui est restreint. Ce paramètre de contrôle μ est entièrement chaotique seulement si $\mu \cong 4$ (voir figure 1.6). De ce fait, l'espace effectif de la clé est considérablement limité. Cependant, la suite PLCM selon la définition de Zhou et al. [71]-[73] possède un paramètre de contrôle entièrement chaotique sur tout l'intervalle de définition $(0,0.5)$. Par conséquent, en partant de la définition de la fonction de permutation

proposée dans [36], nous proposons une fonction de permutation chaotique basée sur les suites PLCM.

Soit une image **I** de taille $N \times M$. Les étapes de permutation peuvent être décrites comme suit:

1) Générer un vecteur z, $\{z_k, k = 1,2,3, \dots\}$, en utilisant l'équation (1.4) de la suite PLCM avec une condition initiale z_0, et un paramètre de contrôle λ.

2) Extraire $N \times M$ éléments du vecteur précèdent v pour constituer un autre vecteur t, $\{t_k, k = \tau + 1, \tau + 2, \dots, \tau + N \times M\}$, où $\tau \in \mathbb{N}^*$ est un entier naturel non nul appelé paramètre de troncation.

3) Ordonner le vecteur t dans un ordre croissant pour obtenir un nouveau vecteur a et les changements causés par cette nouvelle disposition des éléments du vecteur t sont enregistrés en parallèle dans un vecteur de permutation m où $\{ a_k = t_k(m_k),\ k = 1,2, \dots ,\ N \times M\}$.

4) Redimensionner l'image **I** en un vecteur i de taille $1 \times N \times M$, puis permuter ce vecteur en utilisant le vecteur de permutation m généré lors de l'étape précédente pour obtenir au final un vecteur permuté c, où $\{ c_k = i_k(m_k),\ k = 1,2, \dots, 1 \times\ N \times M\}$.

Enfin, le vecteur c est redimensionné en une matrice de taille $N \times M$ pour obtenir l'image permutée finale **C**. Ces étapes peuvent être résumées par une fonction de permutation que l'on note $S_{\{z_0,\lambda,\tau\}}(.)$. Les étapes de permutation sont l'inverse des étapes précédentes, où le vecteur original i de l'image est récupéré seulement si $\{ i_k = c_k(m_k),\ k = 1,2, \dots ,\ N \times M\}$ et la fonction de permutation inverse peut être notée $S^{-1}{}_{\{z_0,\lambda,\tau\}}(.)$

4.3.1.2 Algorithmes de cryptage et de décryptage

La méthode proposée est illustrée dans la figure 4.11. Supposons que nous avons une image **I** de taille $N \times M$ et $\left[e^{j\alpha(n,m)}\right]$ un masque de phases aléatoires de taille équivalente à la taille de l'image **I**, où $\alpha(x, y)$ est une fonction ayant une distribution aléatoire et uniforme dans l'intervalle $[0,2\pi]$.

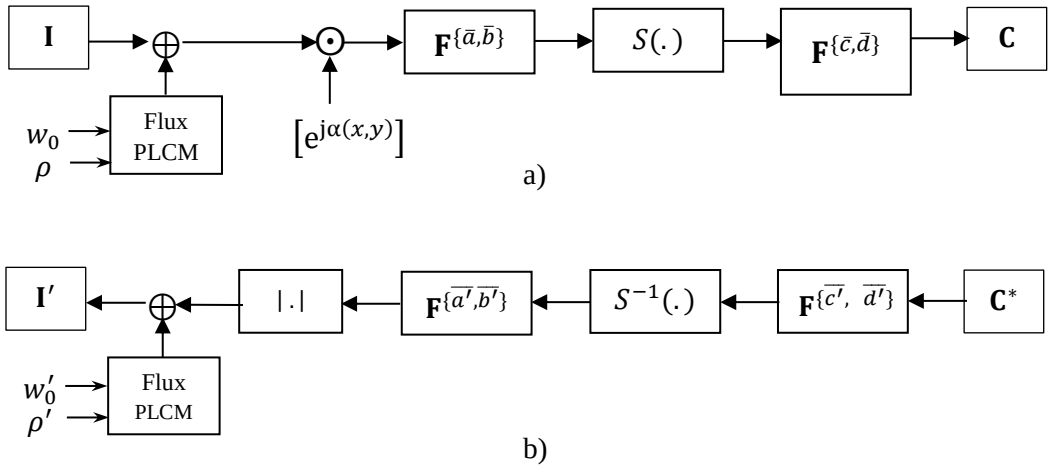


Figure 4.11 Méthode proposée de cryptage d'image basée sur la transformée TFRD à paramètres multiples et un prétraitement non-linéaire, a) algorithme de cryptage, b) algorithme de décryptage.

Soient $\mathbf{F}^{\bar{a}}$ et $\mathbf{F}^{\bar{c}}$ des matrices d'ordre N de la transformée TFRD à paramètres multiples construite en utilisant l'équation (2.7) avec des vecteurs paramétriques $\bar{a}$ et $\bar{c}$ de taille $1 \times N$. De la même manière, soient $\mathbf{F}^{\bar{b}}$ et $\mathbf{F}^{\bar{d}}$ des matrices d'ordre M de la transformée TFRD à paramètres multiples construite avec des vecteurs paramétriques $\bar{b}$ et $\bar{d}$ de taille $1 \times M$. Il faut noter que les vecteurs paramétriques $\{\bar{a}, \bar{b}, \bar{c}, \bar{d}\}$ sont constitués d'ordres fractionnaires distincts aléatoirement choisis de l'intervalle $(0,2)$.

Ainsi, les étapes de cryptage hybride opto-numérique de l'image **I** avec la méthode proposée sont décrites dans ce qui suit:

1) Générer numériquement un vecteur w, $\{w_k, k = 1,2, \dots, 1 \times N \times M\}$, en utilisant l'équation (1.4) de la suite PLCM avec une condition initiale w_0, et un paramètre de contrôle ρ.

2) Convertir les éléments réels du vecteur w en nombres entiers de 8 bits dans un nouveau vecteur s que l'on appelle un vecteur de flux, car il servira comme générateur de flux de bits pour l'étape du prétraitement non-linéaires.

3) Convertir l'image **I** en un vecteur i de taille $1 \times N \times M$, puis effectuer un prétraitement non-linéaire en accomplissant une opération XOR bit-par-bit entre le vecteur de l'image i et le vecteur de flux s comme suit :

$$\{x_k = i_k \oplus s_k, k = 1,2, \dots, 1 \times N \times M\} \tag{4.5}$$

4) Redimensionner le vecteur x obtenu lors de l'étape précédente en une matrice **X** de taille $N \times M$, puis multiplier cette matrice élément-par-élément avec le masque de phases aléatoires $\left[\mathrm{e}^{\mathrm{j}\alpha(n,m)}\right]$.

5) Appliquer optiquement une transformée TFRD à paramètres multiples $\mathbf{F}^{\{\bar{a},\bar{b}\}}[\,.\,]$ dans l'axe des x, ensuite suivant l'axe des y sur le résultat du produit de l'étape précédente en utilisant les vecteurs paramétriques $\{\bar{a}, \bar{b}\}$.

6) Permuter numériquement l'image obtenue dans le domaine de la transformée TFRD en utilisant la fonction de permutation chaotique $S_{\{z_0,\lambda,\tau\}}(.)$ avec les paramètres $z_0 \in (0,1)$, $\lambda \in (0,0.5)$, et $\tau \in \mathbb{N}^*$.

7) Appliquer optiquement une autre transformée TFRD à paramètres multiples $\mathbf{F}^{\{\bar{c},\bar{d}\}}[\,.\,]$ suivant l'axe des x, ensuite suivant l'axe des y sur l'image permutée résultante de l'étape précédente en utilisant les vecteurs paramétriques $\{\bar{c}, \bar{d}\}$.

Les étapes précédentes peuvent être résumées par l'équation suivante :

$$\mathbf{C} = \mathbf{F}^{\{\bar{c},\bar{d}\}}\Big[\ S_{\{z_0,\lambda,\tau\}}\Big(\ \mathbf{F}^{\{\bar{a},\bar{b}\}}\left[\mathbf{X} \odot \mathrm{e}^{\mathrm{j}\alpha(x,y)}\right]\Big)\Big] \tag{4.6}$$

où $\odot$ indique l'opération de multiplication élément-par-élément, et **C** indique l'image cryptée qui a une amplitude complexe. Ainsi, la clé secrète de cryptage que l'on note K est finalement composée des ordres fractionnaires des vecteurs paramétriques $\{\bar{a}, \bar{b}, \bar{c}, \bar{d}\}$ de la transformée TFRD à paramètres multiples, des paramètres w_0 et ρ que nous avons utilisé lors de l'étape du prétraitement non-linéaire, et des paramètres z_0, λ, τ utilisés avec la fonction de permutation chaotique $S_{\{z_0,\lambda,\tau\}}(.)$.

Notez que l'équation (4.6) peut être utilisée de manière itérative et plus il y aura d'itérations, plus la sécurité du cryptage sera meilleure. Cependant, un compromis entre la complexité de calcul et le niveau de sécurité souhaité doit être toujours envisagé au préalable.

Supposons à présent une clé secrète K' composée de vecteurs paramétriques $\{\bar{a}', \bar{b}', \bar{c}', \bar{d}'\}$ de la transformée TFRD à paramètres multiples, des paramètres ${z_0}', \lambda'$, et τ' de la fonction de permutation inverse ${S^{-1}}_{\{{z_0}', \lambda', \tau'\}}$, et des paramètres ${w'}_0$, ρ' du prétraitement non-linéaire.

Ainsi, les étapes de décryptage de l'image cryptée $\mathbf{C}$ avec la clé K' sont l'inverse des étapes précédentes de cryptage, toutefois, il est important de prendre le conjugué complexe de l'image cryptée avant de procéder au décryptage, de ce fait, les étapes de décryptage peuvent être résumées par l'équation suivante :

$$\mathbf{X}' = \left| \mathbf{F}^{\{\bar{a}', \bar{b}'\}} \left[{S^{-1}}_{\{{z_0}', \lambda', \tau'\}} \left(\mathbf{F}^{\{\bar{c}', \bar{d}'\}} [\, \mathbf{C}^*] \right) \right] \right| \tag{4.7}$$

où $\mathbf{C}^*$ indique le conjugué complexe de l'image cryptée $\mathbf{C}$, et $|\,.\,|$ indique le module de l'image cryptée.

Pour obtenir l'image décryptée finale $\mathbf{I}'$, la matrice $\mathbf{X}'$ doit être convertie en un vecteur x', puis un prétraitement non linéaire inverse est appliqué en utilisant une opération XOR bit-par-bit entre le vecteur x' et un vecteur de flux s' généré de manière similaire à l'étape (1) en cryptage de sorte que :

$$\{{i'}_k = {x'}_k \oplus {s'}_k \,, k = 1, \dots, 1 \times N \times M\} \tag{4.8}$$

L'image $\mathbf{I}'$ finale est obtenue en redimensionnant le vecteur i' en une matrice de taille $N \times M$.

Enfin, nous remarquons que la méthode proposée est une méthode de cryptage symétrique, car l'image finale décryptée $\mathbf{I}'$ est identique à l'image originale $\mathbf{I}$ seulement si les vecteurs paramétriques $\{\bar{a}', \bar{b}', \bar{c}', \bar{d}'\}$ sont équivalents aux vecteurs paramétriques $\{\bar{a}, \bar{b}, \bar{c}, \bar{d}\}$ et seulement si ${w_0}' = w_0$, $\rho' = \rho$, ${z_0}' = z_0$, $\lambda' = \lambda$, et $\tau' = \tau$.

4.3.1.3 Implémentation opto-numérique

Les algorithmes de cryptage et de décryptage de la méthode proposée peuvent être implémentés dans un montage hybride opto-numérique similaire à celui de Lang et al. dans [36] qui est illustré dans la figure 4.12.

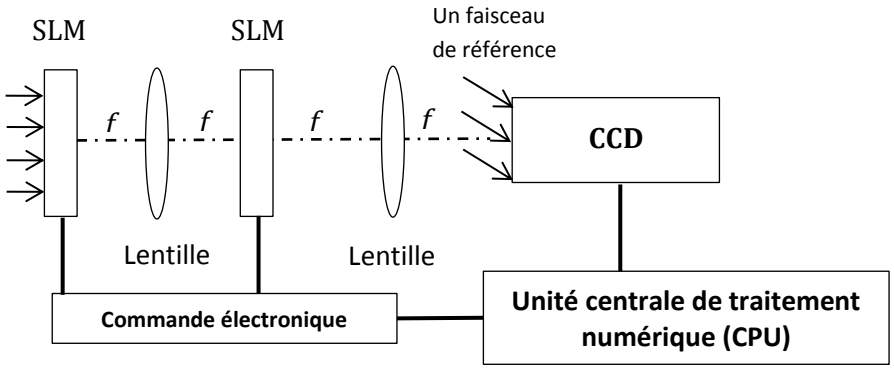


Figure 4.12 Implémentation opto-numérique

Cette configuration peut être divisée en une partie numérique et en une partie optique. Pour la partie optique, elle est basée sur un montage appelée 4-f en optique qui permet l'implémentation optique de la transformée TFRD 2D à paramètres multiples en utilisant deux lentilles et un modulateur spatial de lumière SLM ou « spatial light modulator » en anglais contrôlé électroniquement pour l'affichage du signal discret complexe lors des différentes étapes du cryptage et du décryptage [36], de plus, un faisceau de référence est utilisé afin de capturer les signaux complexes numériquement dans un capteur CCD (une caméra) en utilisant les techniques d'holographie numérique [36]. A présent, en ce qui concerne la partie numérique du montage, le prétraitement non-linéaire ainsi que la permutation basée sur les suites PLCM dans la méthode proposée sont effectués numériquement sur le signal image en utilisant une unité centrale de traitement numérique (CPU) avant de le réinjecté dans la partie optique du montage en utilisant une commande électronique [36]. Notez que l'opération XOR peut être implémentée numériquement, mais elle peut aussi être implémentée optiquement [86], cependant, il est recommandé d'utiliser une implémentation numérique, car un CPU est déjà nécessaire pour réaliser les permutations chaotiques. Par conséquent, l'utilisation d'une implémentation optique de l'opérateur XOR compliquera seulement le montage.

4.3.1.4 Résultats et discussions

Dans cette section, nous présentons seulement les résultats de simulation numérique de la méthode proposée avec des images de tests standards de niveau gris (8bits), puis une évaluation détaillée de sa sécurité est discutée.

Soit une image de test de taille carrée 256×256, et une clé secrète K qui est définie par :

- Les paramètres du processus de prétraitement non-linéaire:
  $w_0 = 0.2567$ et $\rho = 0.1428$.
- Les paramètres de la fonction de permutation chaotique:
  $z_0 = 0.9856$, $\lambda = 0.2857$, et $\tau = 4200$.
- Les $4N$ ordres fractionnaires des vecteurs paramétriques $\{\bar{a}, \bar{b}, \bar{c}, \bar{d}\}$ choisis aléatoirement de l'intervalle $(0,2)$.

Ainsi, pour une image Lenna cryptée avec la clé secrète K, les résultats de simulation sont illustrés dans la figure 4.13.

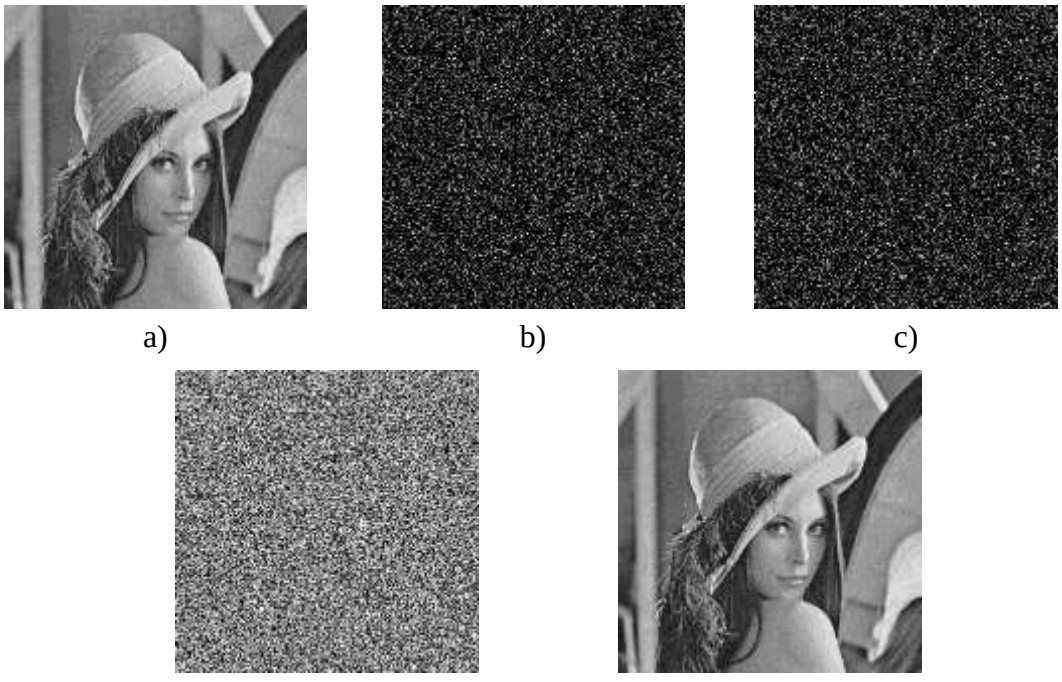


Figure 4.13 Résultats de simulation: a) image originale, b) et c) partie réelle et partie imaginaire de l'image cryptée, d) image décryptée avec une clé incorrecte, e) image décryptée avec une clé correcte.

La figure 4.13(a) montre l'image Lenna originale, et les figures 4.13(b) et 4.13(c) montrent la partie réelle et la partie imaginaire de l'image Lenna cryptée. De plus, les figures 4.13(d) et 4.13(e) montrent l'image Lenna décryptée avec une clé secrète incorrecte, ensuite avec une clé secrète correcte.

D'après ces résultats, nous remarquons que l'image est correctement cryptée visuellement sans aucun détail visuel visible sur l'image cryptée complexe, de ce fait, la méthode de cryptage proposée possède une sécurité perceptuelle satisfaisante, cependant, cette analyse reste subjective. Pour évaluer la qualité du cryptage de manière objective, le tableau 4.2 montre le coefficient de corrélation c_r entre la partie réelle de l'image cryptée et celle originale et le coefficient de corrélation c_i entre la partie imaginaire de l'image cryptée et celle originale et dans le cas de plusieurs images de tests.

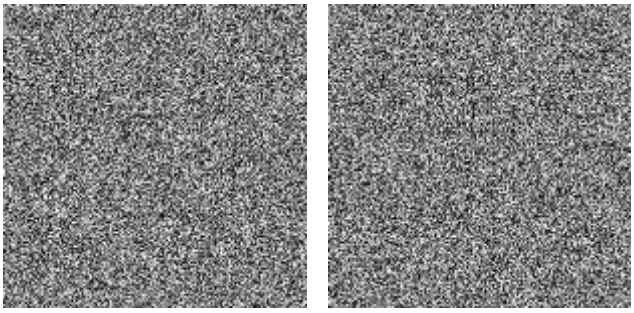
Tableau 4.2 Coefficient de corrélation entre l'image originale et l'image cryptée.

Image	$\mathbf{c_r}$	$\mathbf{c_i}$
Lenna	0.0009	- 0.0015
Barbara	- 0.0001	- 0.001
Cameraman	0.0003	0.0002
Mandrill	0.0008	- 0.0009
Aerial	0.001	- 0.001

D'après ce tableau, nous remarquons que dans tous les cas, le coefficient de corrélation reste très proche du zéro, et peu importe l'image originale. Cela veut dire que l'image cryptée a une relation très faible avec celle originale, ce qui est avantageux en cryptage. En conséquence, ces résultats démontrent la qualité du cryptage de la méthode proposée.

A présent, pour évaluer la sensibilité de la clé secrète dans la méthode proposée, nous supposons qu'une image Lenna est cryptée avec la clé secrète K définie précédemment, puis cette image est décryptée avec une autre clé secrète K' équivalente à la clé secrète originale K mais comportant une erreur dans l'un de ses différents paramètres. Ainsi, pour les paramètres w_0 et ρ

utilisés dans le processus de prétraitement non-linéaire, les figures 4.14(a) et 4.14(b) montrent l'image Lenna décryptée lorsque $w_0' = w_0 + 10^{-16}$, et lorsque $\rho' = \rho + 10^{-16}$.

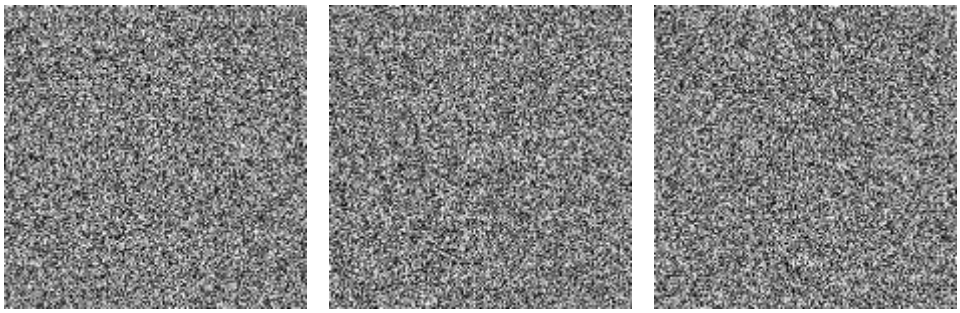


a) b)

Figure 4.14 Image décryptée en fonction des paramètres du prétraitement non-linéaire: a) $w_0' = x_0 + 10^{-16}$, b) $\rho' = \rho + 10^{-16}$.

Nous remarquons que l'image reste correctement cryptée malgré une erreur minime dans les paramètres du prétraitement non linéaire proposé.

Concernant les paramètres z_0, λ , et τ de la fonction de permutation chaotique proposée, les figures 4.15(a), 4.15(b), et 4.15(c) montrent, respectivement, l'image Lenna décryptée lorsque $z_0' = z_0 + 10^{-16}$, $\lambda' = \lambda + 10^{-16}$, et $\tau' = \tau + 1$.

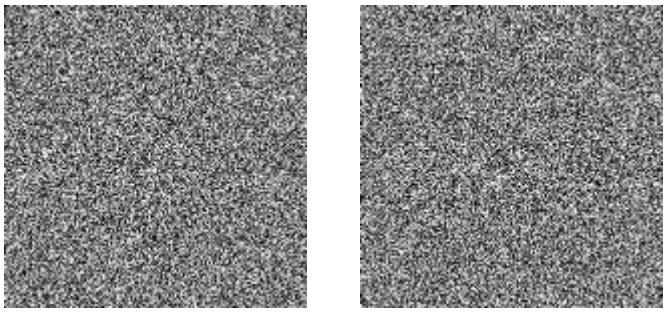


a) b) c)

Figure 4.15 Image décryptée en fonction des paramètres de la permutation: a) $z_0' = z_0 + 10^{-16}$, b) $\lambda' = \lambda + 10^{-16}$, c) $\tau' = \tau + 1$.

Nous remarquons également que la clé secrète reste très sensible aux erreurs dans la fonction de permutation chaotique proposée. Pour les ordres fractionnaires des vecteurs paramétriques $\bar{a}, \bar{b}, \bar{c},$ et $\bar{d}$ de la transformée TFRD à paramètres multiples, les figures 4.16(a) et 4.16(b)

montrent l'image décryptée lorsque 50% et 25% des vecteurs paramétriques $\{\bar{a}, \bar{b}, \bar{c}, \bar{d}\}$ sont incorrects. Nous constatons aussi dans le cas échéant que l'image reste correctement cryptée.



a) b)

Figure 4.16 Image décryptée en fonction des paramètres de la transformée TFRD à paramètres multiples: a) 50% , b) 25% des vecteurs paramétriques $\{\bar{a}, \bar{b}, \bar{c}, \bar{d}\}$ sont incorrects.

Pour déterminer de façon approfondie la sensibilité de la clé secrète aux vecteurs paramétriques $\{\bar{a}, \bar{b}, \bar{c}, \bar{d}\}$ de la transformée TFRD à paramètres multiples, une image Lenna cryptée est décryptée avec des vecteurs $\overline{\delta_k}$, où $\{\bar{a} + \overline{\delta_1}, \bar{b} + \overline{\delta_2}, \bar{c} + \overline{\delta_3}, \bar{d} + \overline{\delta_4}\}$.

Chaque vecteur $\overline{\delta_k}$ comporte une erreur de déviation δ aléatoirement sélectionnée de l'ensemble $\{\delta, -\delta\}$ et pour chaque erreur de déviation δ, nous calculons l'EQM entre l'image Lenna décryptée et celle originale. Les résultats de simulation sont tracés dans la figure 4.17.

D'après cette figure, nous remarquons que plus on diverge du zéro plus l'EQM est importante, de plus, les éléments des vecteurs paramétriques sont très sensible aux erreurs. Ces résultats peuvent être vérifiés visuellement, par exemple, la figure 4.18 montre l'image Lenna décryptée lorsque le seuil d'erreur $\delta = 0.01$.

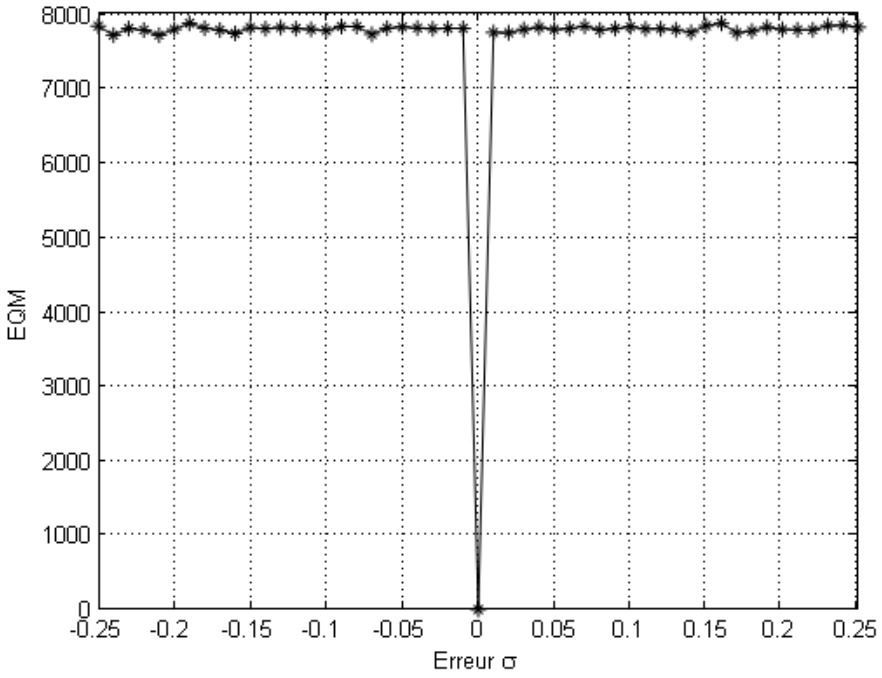


Figure 4.17 EQM en fonction d'une erreur δ dans les paramètres de la transformée TFRD à paramètres multiples.

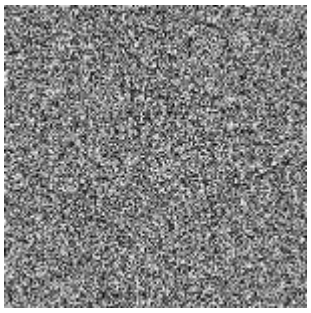


Figure 4.18 Image décryptée avec une erreur $\delta = 0.01$.

Ainsi, il est évident que la clé secrète est sensible aux erreurs, de plus, ces résultats nous permettent de déterminer de façon approximative l'espace de la clé qui est de l'ordre de $10^{4\times16} \times 10^{4N}$, ce qui est largement plus grand que le minimum 2^{100} requis pour résister à une attaque par force brute. En conséquence, la méthode proposée est robuste contre les attaques par force brute.

Afin d'illustrer l'intérêt du prétraitement non-linéaire proposé, nous comparons la méthode proposée avec d'autres méthodes de cryptage linéaire existantes qui sont basées sur la transformée TFR à paramètres multiples telle que la fameuse méthode de Lang et al. [36]. Pour cela, nous calculons l'EQM en fonction d'une erreur de déviation δ aléatoirement sélectionnée de l'ensemble $\{\delta, -\delta\}$ de sorte que $\{\bar{a}+\bar{\delta}, \bar{b}+\bar{\delta}, \bar{c}, \bar{d}\}$. De plus, pour illustrer l'intérêt de la fonction de permutation chaotique basée sur la suite PLCM, nous calculons de la même façon l'EQM dans la méthode de Hennely et dans la méthode de Liu lorsque les ordres fractionnaires de la transformée TFR sont :

- Dans la méthode de Hennelly [33] : $\{0.97+\delta, 0.75+\delta, 1.41, 1.1, 0.23, 0.8\}$.
- Dans la méthode de Liu [40] : $\{0.2,\ 1.8+\delta, 1.3,\ 0.6+\delta, 1.2, 1.7+\delta, 0.6, 0.1+\delta\}$.

Les résultats de simulation sont tracés dans la figure 4.19.

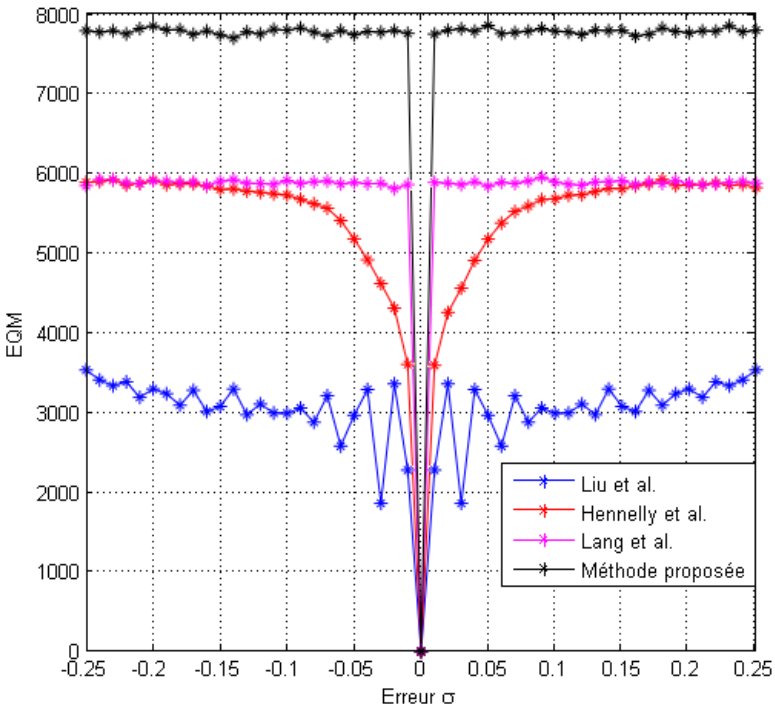


Figure 4.19 Comparaison de l'EQM avec d'autres méthodes existantes.

D'après ces résultats, il est évident que la méthode proposée est plus sensibles aux erreurs que les autres méthodes existantes. Cela est rendu possible principalement grâce au processus de prétraitement non linéaire proposé dans le domaine spatial et basé sur les suites PLCM.

De plus, la fonction de permutation chaotique dans la méthode proposée améliore significativement l'espace de la clé de sécurité grâce à l'utilisation de la suite PLCM de Zhou. Cet avantage peut être vu lorsque l'image Lenna dans la méthode de Lang et al. [36] est correctement décryptée dans la figure 4.20 malgré la présence d'une erreur de 10^{-4} dans le paramètre de contrôle de la suite logistique, alors qu'il est supposé avoir une sensibilité aux erreurs de 10^{-16} [36].

Figure 4.20 Image décryptée avec une erreur 10^{-4} dans le paramètre de contrôle de la fonction de permutation de Lang et al.

Cela est due au fait que le paramètre de contrôle de la suite logistique a une sensibilité irrégulière dans son intervalle de définition (3.57,4) et il est entièrement chaotique seulement si sa valeur est proche de 4 (voir figure 1.6). Cette irrégularité réduit considérablement l'espace effectif de la clé si le paramètre de contrôle est très inférieur à 4, alors que dans la fonction de permutation chaotique basée sur la suite PLCM de Zhou, le paramètre de contrôle de la suite est constamment chaotique quel que soit la valeur choisie à partir de son intervalle de définition (0,0.5) (voir figure 1.7, chapitre 1).

Ainsi, ces résultats viennent montrer l'efficacité du processus de prétraitement non-linéaire de même que la fonction de permutation basée sur la suite PLCM de Zhou dans la méthode proposée.

A présent, pour évaluer la sécurité de la méthode proposée contre l'analyse statistique par histogramme, la figure 4.21 montre les histogrammes de quelques images de test, et ceux correspondants à leurs versions cryptées en utilisant la même clé de cryptage K. Etant donné que l'image cryptée est d'amplitude complexe, l'histogramme du module et celui de la phase sont illustrés séparément.

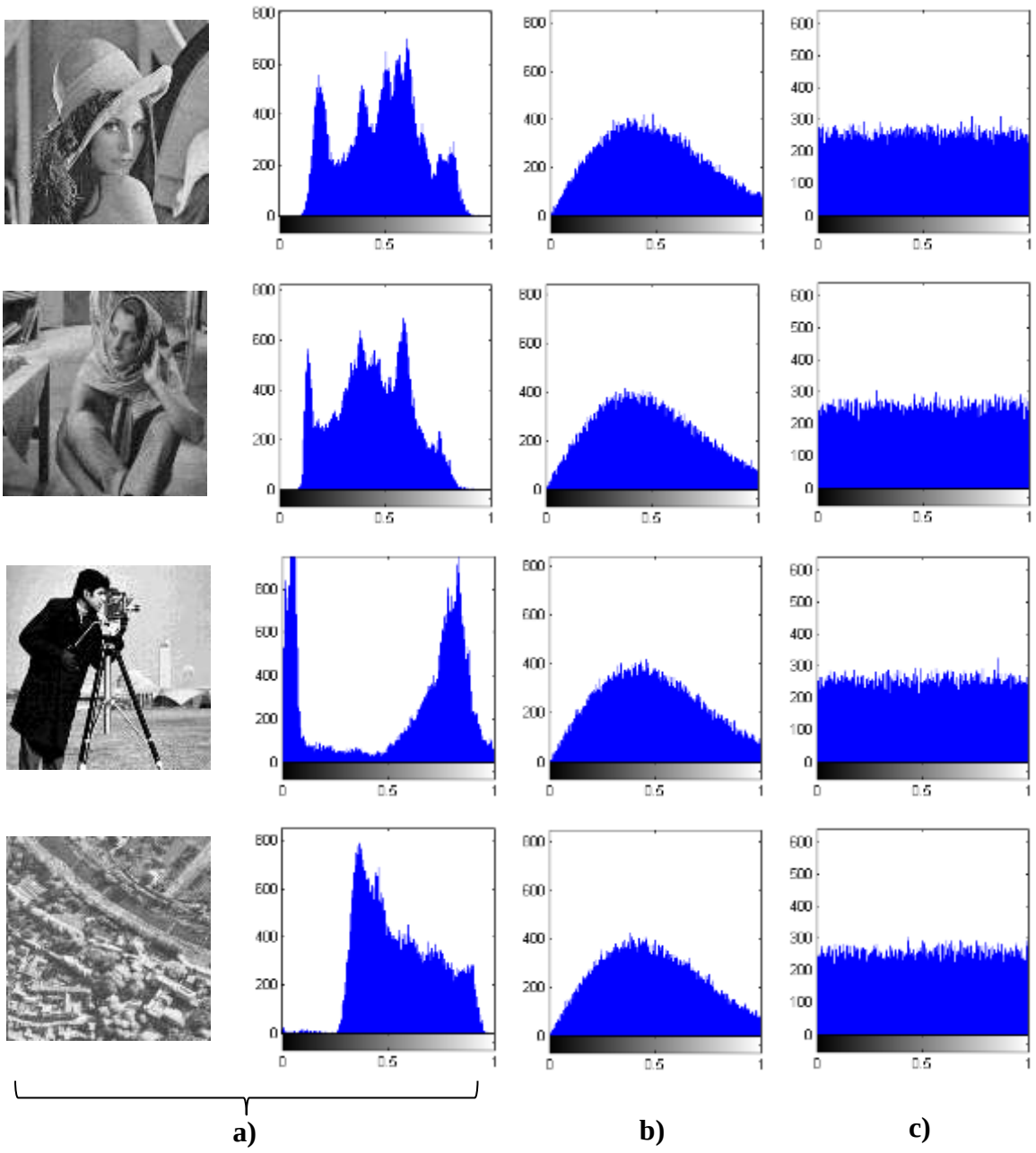
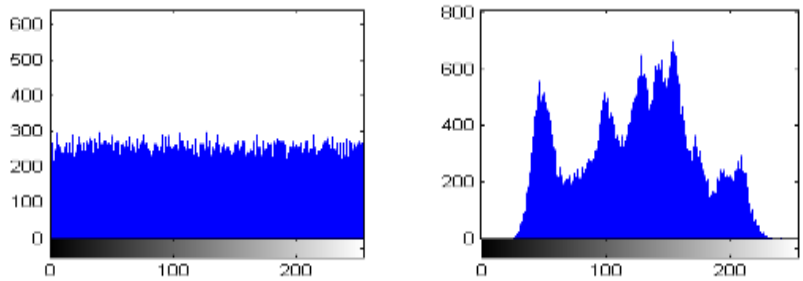


Figure 4.21 Histogrammes de quelques images de test: a) l'image originale, b) et c) le module et la phase de l'image cryptée.

Nous nous apercevons que quel que soit l'image de test utilisée, les histogrammes du module et de la phase de l'image cryptée sont aléatoires et absolument différents de l'histogramme de l'image originale, de plus, ils sont identiques et suivent la même distribution. Cela veut dire qu'aucune information sur l'image originale n'est présente dans l'histogramme de l'image cryptée.

Pour voir statistiquement l'utilité de l'utilisation du prétraitement non-linéaire proposée dans le domaine spatial, nous comparons la méthode proposée avec la méthode linéaire de Lang et al. [36]. Ainsi, la figure 4.22(a) montre l'histogramme de l'image Lenna décryptée lorsque l'attaquant ignore les paramètres du prétraitement non linéaire proposée dans le domaine spatial, et la figure 4.22(b) lorsque l'attaquant ignore les paramètres de la fonction de permutation spatiale.



a) **b)**

Figure 4.22 Histogramme de l'image Lenna décryptée avec une clé secrète incorrecte: a) méthode proposée, b) méthode de Lang et al.

Il est évident que l'histogramme de l'image Lenna décryptée a une distribution d'allure uniforme dans la méthode proposée, alors que dans la méthode de Lang, l'histogramme de Lenna décryptée est identique à celui de l'image Lenna originale. Cette information peut être exploitée dans d'éventuelles attaques statistiques sophistiquées. En conséquent, le prétraitement non linéaire proposé permet d'améliorer significativement les méthodes de cryptage DRPE.

Afin d'étudier la résistance de la méthode proposée contre le bruit additif du canal, une image Lenna cryptée et bruitée par la méthode proposée, ensuite l'EQM a été calculée entre l'image Lenna originale et celle décryptée dans la méthode proposée et dans la méthode de Lang

et al. [36] en fonction du coefficient de puissance du bruit σ . Les résultats de simulations sont illustrés dans la figure 4.23.

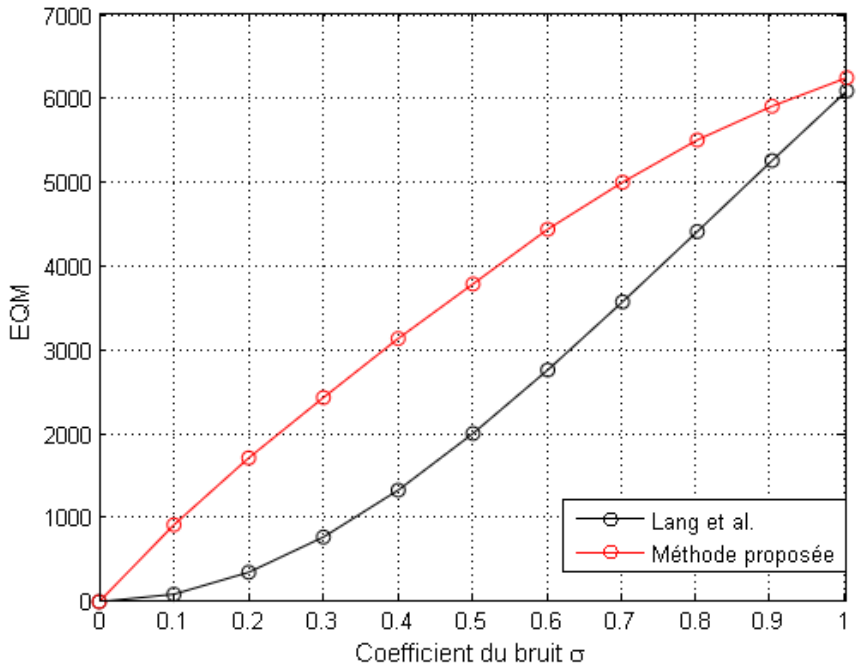
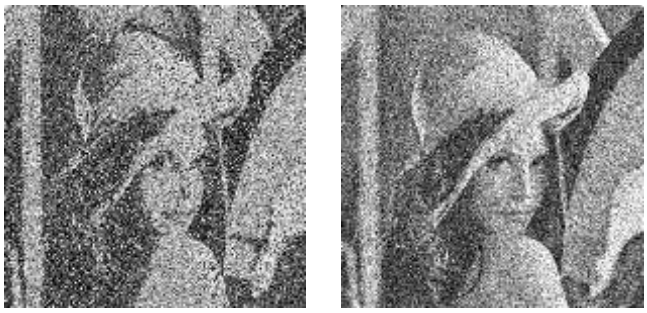


Figure 4.23 Comparaison de l'EQM en présence d'un bruit blanc Gaussien additif.

D'après ces résultats, l'EQM est proportionnelle au coefficient de puissance σ du bruit dans la méthode proposée ainsi que dans la méthode de Lang et al., cependant, l'évolution de l'EQM est plus rapide dans la méthode proposée que dans la méthode de Lang et al., cela est due au cryptage non-linéaire. Pour vérifier ces résultats visuellement, la figure 4.24 montre l'image décryptée ainsi que son PSNR dans la méthode proposée et dans la méthode de Lang lorsque le niveau de puissance du bruit est à 50%.

Nous remarquons que l'image Lenna originale peut être reconnue visuellement à un certain degré dans les deux cas, cependant, il y a une légère perte de qualité de l'image dans la méthode proposée, où le PSNR est inférieur dans la méthode proposée en comparaison avec la méthode de Lang, néanmoins, la qualité peut être éventuellement améliorée par des techniques numériques de posttraitement des images ou par l'introduction par exemple de codes correcteurs d'erreurs qui sont généralement utilisés dans de telles situations.

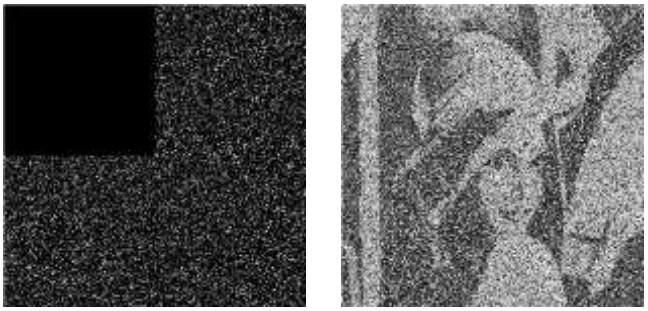


PSNR = 12dB PSNR = 15dB

a) b)

Figure 4.24 Comparaison de la qualité de l'image en présence de bruit additif : a) méthode proposée, b) méthode de Lang et al.

De plus, pour tester la résistance de la méthode proposée contre les erreurs de transmission, nous supposons qu'une partie des pixels de l'image Lenna cryptée ont été corrompus, ensuite l'image Lenna décryptée est illustrée avec son PSNR dans la figure 4.25.



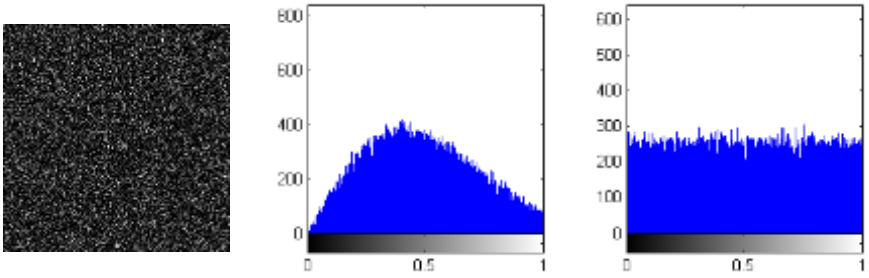
PSNR = 11dB

a) b)

Figure 4.25 Image décryptée après la perte d'une partie des pixels: a) image cryptée, b) image décryptée.

Nous remarquons que l'image Lenna originale reste reconnaissable malgré que l'image cryptée ait perdue une grande partie de ses pixels. En conséquence, ces résultats démontrent la résistance de la méthode proposée contre les erreurs de transmission.

Si nous considérons qu'une attaque en cryptanalyse est possible, telle que l'attaque à texte en clair choisi. Un attaquant peut choisir une image en entrée qui comporte que des zéros, et dans ce cas l'image cryptée résultante ainsi que son histogramme sont illustrées sur la figure 4.26.



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

Figure 4.26 Attaque à texte en clair choisi: a) image cryptée d'une image contenant que des zéros, b), et c) histogramme du module et de la phase de l'image cryptée

En effectuant une cryptanalyse préliminaire, on remarque que l'histogramme reste toujours identique à ceux que nous avons vus précédemment, de ce fait, l'image cryptée obtenue ne peut être utilisée comme une clé équivalente pour décrypter d'autres images cryptées par la clé secrète, car elle n'offre aucune information valide pour monter une attaque à texte en clair choisi. De plus, vu que la méthode proposée est une méthode opto-numérique, les suites PLCM du processus de prétraitement non-linéaire ainsi que le processus de permutation chaotiques sont gérés par un CPU. De ce fait, la clé secrète de cryptage peut être mise à jour après chaque opération de cryptage en choisissant une clé parmi $10^{4\times16} \times 10^{4N}$ clés disponibles. Ainsi, la nouvelle clé peut être échangée en temps réel entre l'émetteur et le récepteur en utilisant par exemple un cryptage asymétrique, ce qui nous amène à une classe d'algorithmes de cryptage hybride. Par conséquent, la méthode proposée interdit ce type d'attaque à texte en clair choisi tant que la clé secrète est dynamique.

4.4 Conclusion

Dans ce chapitre, nous avons proposé un nouveau prétraitement non-linéaire efficace dans le domaine spatial de la méthode de cryptage DRPE en utilisant le domaine de la transformée ROP ainsi que le domaine de la transformée TFRD à paramètres multiples. Ce prétraitement est basé sur l'utilisation d'une combinaison de suites chaotiques associées à une fonction XOR. Les résultats de simulation ont montré la faisabilité des méthodes proposées dans le cas du cryptage numérique, mais également dans le cas du cryptage hybride opto-numérique. De plus, les résultats de comparaison ont montré que le prétraitement non linéaire proposé permet d'améliorer significativement les méthodes de cryptage DRPE existantes basées sur les transformées paramétriques et les permutations chaotiques.

Chapitre 5

Proposition d'une nouvelle méthode de cryptage des séquences d'images vidéo

5.1 Introduction

Les méthodes de cryptage basées sur des transformées paramétriques sont en général conçues pour le cryptage de tous types de signal 2D. Ce signal peut être une image, mais aussi une séquence d'images vidéo. En effet, une séquence d'images vidéo comprend plusieurs images appelées trames. Ces trames peuvent être indépendantes ou dépendantes des autres trames adjacentes [49].

Jindal et al. ont proposé dans [49] de crypter individuellement chaque trame d'une séquence d'images vidéo en utilisant la méthode DRPE avec une transformée fractionnaire telle que la transformée TFRD afin de réduire le nombre de réitération de retransmission au niveau du récepteur, ce qui a pour avantage d'accélérer le processus du décryptage de la séquence vidéo lorsqu'elle comprend une ou plusieurs trames altérées ou perdues durant le transit sur le canal [49]. Bien que ce cryptage soit efficace, l'utilisation d'une transformée fractionnaire complexe et de masques de phases aléatoires est limitée en matière de débit de transmission, car une séquence d'images vidéo comporte plusieurs trames réelles qui doivent être cryptées et envoyées simultanément, et si les trames cryptées sont complexes, alors le volume de la séquence d'images vidéo à envoyer sera très important.

Dans ce chapitre, nous proposons une méthode de cryptage des séquences d'images vidéo basée sur la structure DRPE en utilisant la transformée TFRDR qui est une transformée fractionnaire réelle présentée dans [79]. Contrairement à la méthode proposée par Jindal et al. [49], les masques de phases aléatoires sont remplacés dans notre méthode par des fonctions de permutation. Par conséquent, les trames cryptées par la méthode DRPE proposée sont de type réel, ce qui remédie au problème exposé précédemment. De plus, nous proposons d'introduire une architecture de cryptage bloc par bloc qui consiste à diviser chaque trame à crypter en un nombre prédéfini de blocs rectangulaires afin d'augmenter le nombre d'ordres fractionnaires utilisés dans le cryptage.

5.2 Description de la méthode

5.2.1 Fonction de permutation par blocs

La fonction de permutation décrite dans cette section est la fonction utilisée dans le chapitre 3 mais avec une définition adaptée pour le cryptage bloc par bloc.

Soient $\mathbf{B}_k$, $k = 1, 2, \cdots, K$, des blocs de pixels de taille $n \times m$ avec $K \in \mathbb{N}^*$. Vu la présence de plusieurs blocs à permuter, il est donc préférable d'utiliser la suite logistique comme générateur de permutation au lieu de la suite PLCM de Zhou [75] utilisée dans le chapitre 4, car la suite logistique est plus rapide que la suite PLCM lorsqu'on a besoin de générer plusieurs suites chaotiques pour l'ensemble des blocs d'une trame vidéo. Néanmoins, pour assurer une permutation réellement chaotique, le paramètre de contrôle de la suite chaotique est fixé à 4 afin d'avoir des suites entièrement chaotiques. Ainsi, les pixels d'un bloc $\mathbf{B}_k$ peuvent être permutés chaotiquement selon les étapes suivantes:

1) Former un vecteur $\mathbf{X}_k$ de nombres réels aléatoires de taille $L = n \times m$ en utilisant l'équation (1.3) de la suite logistique avec une condition initiale $(x_0)_k$ sélectionnée aléatoirement de l'intervalle (0,1), et un paramètre de contrôle μ égale à 4.

2) Trier les éléments du vecteur $\mathbf{X}_k$ dans un ordre ascendant ou dans un ordre descendant, puis le changement occasionné dans l'index de position de chaque élément du vecteur est enregistré en parallèle dans un autre vecteur $\mathbf{V}_k$ qu'on désigne comme le vecteur de permutation.

3) Redimensionner le bloc $\mathbf{B}_k$ en un vecteur, puis effectuer une permutation entre ses éléments en utilisant le vecteur de permutation $\mathbf{V}_k$ pour former un autre vecteur $\mathbf{P}_k$, où $p_l = b_l(v_l)$, $l = 1,2,3, \ldots, 1 \times L$.

4) Convertir le vecteur $\mathbf{P}_k$ obtenu de l'étape précédente en une matrice ou un bloc $\mathbf{C}_k$ de taille $n \times m$.

Les étapes de (1)–(4) peuvent être désignées par une fonction $P_{\{(x_0)_k\}}(.)$. Les étapes de permutation inverse sont l'inverse des étapes précédentes, et peuvent être présentées par la fonction $P^{-1}_{\{(x_0)_k\}}(.)$.

5.2.2 Algorithmes de cryptage et de décryptage

Soit une séquence d'images vidéo qui compte R trames de taille $N \times M$ que l'on note $\mathbf{I}_r$, $r = 1,2, \cdots, R$. Chaque trame $\mathbf{I}_r$ est divisée au début en K blocs $\mathbf{B}_k$,$k = 1,2, \cdots, K \in \mathbb{N}^*$ de largeur n et de longueur m. Ainsi, chaque bloc $\mathbf{B}_k$ est crypté séparément tel qu'illustré dans la figure 5.1.

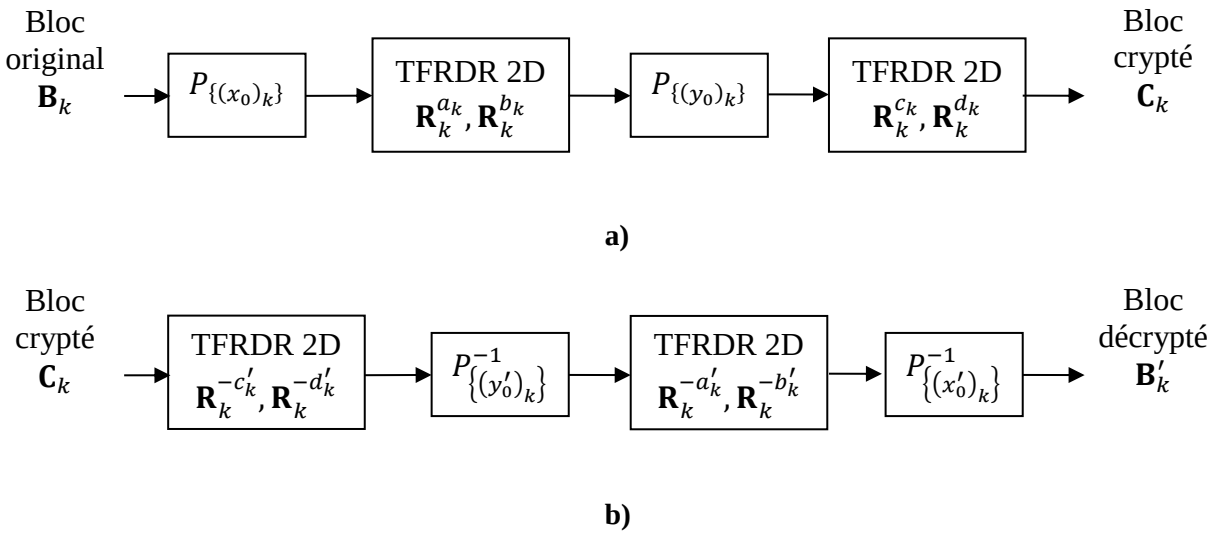


Figure 5.1 Méthode proposée de cryptage d'une séquence d'images vidéo,
a) algorithme de cryptage, b) algorithme de décryptage.

Soient $\mathbf{R}_k^{a_k}$, $\mathbf{R}_k^{b_k}$, $\mathbf{R}_k^{c_k}$, et $\mathbf{R}_k^{d_k}$ des matrices réelles de la TFRDR de taille $n \times m$ construites suivant l'équation (2.5) avec a_k, b_k, c_k, et d_k qui sont des ordres fractionnaires.

Ainsi, pour chaque valeur de $k = 1, 2, \cdots, K$, le bloc $\mathbf{B}_k$ est crypté selon les étapes suivantes :

1) Permuter le bloc $\mathbf{B}_k$ dans le domaine spatial par la fonction de permutation chaotique $P_{\{(x_0)_k\}}(.)$, avec une condition initiale $(x_0)_k$ aléatoirement choisie de l'intervalle (0,1).

2) Appliquer une transformée TFRDR 2D $\mathbf{F}^{\{a_k,b_k\}}[\,.\,]$ sur le bloc permuté de l'étape précédente en utilisant les ordres fractionnaires a_k et b_k aléatoirement choisis de l'intervalle (0,2).

3) Permuter le bloc résultant dans le domaine de la transformée TFRDR en utilisant la fonction de permutation chaotique $P_{\{(y_0)_k\}}(.)$, avec une condition initiale $(y_0)_k$ aléatoirement choisie de l'intervalle (0,1).

4) Appliquer une autre transformée TFRDR 2D $\mathbf{F}^{\{c_k,d_k\}}[\,.\,]$ sur le bloc résultant de l'étape précédente en utilisant les ordres fractionnaires c_k et d_k choisis aléatoirement de l'intervalle (0,2).

Finalement, nous obtenons un bloc crypté que nous le notons $\mathbf{C}_k$, et les étapes (1)–(4) peuvent être résumées par l'équation suivante :

$$\mathbf{C}_k = \mathbf{F}^{\{c_k,d_k\}}\left[P_{\{(y_0)_k\}}\left(\mathbf{F}^{\{a_k,b_k\}}\left[P_{\{(x_0)_k\}}(\mathbf{B}_k)\right]\right)\right] \quad (5.1)$$

Notez que pour $k > 1$, les conditions initiales $(x_0)_k$ et $(y_0)_k$ sont égales, respectivement, aux valeurs $(x_{n\times m})_k$ et $(y_{n\times m})_k$ des dernières itérations des deux suites logistiques utilisées dans le domaine spatial et fréquentiel lors de l'application des fonctions de permutation sur le bloc $\mathbf{B}_{k-1}$ précédent. Ce procédé a pour but de créer une diffusion de l'erreur ou un effet d'avalanche entre le bloc actuel et le bloc précédent d'une seule trame. Cette diffusion peut être assimilée à la propriété de diffusion définit en cryptographie (voir le chapitre 1).

Ainsi, la clé secrète finale que l'on note Q est finalement composée de :

- $4 \times K$ ordres fractionnaires a_k, b_k, c_k, et d_k.
- La largeur n et la longueur m du bloc.
- Les conditions initiales $(x_0)_k$ et $(y_0)_k$ des fonctions de permutations quand $k = 1$.

Il faut noter que cette clé secrète Q est utilisée pour le cryptage de l'ensemble des trames $\mathbf{I}_r$ de la séquence d'images vidéo.

Supposons à présent que nous avons une clé secrète Q' composée de :

- $4 \times K$ ordres fractionnaires a'_k, b'_k, c'_k, et d'_k.
- La largeur n' et la longueur m' du bloc.
- Les conditions initiales $(x'_0)_k$ et $(y'_0)_k$ des fonctions de permutations quand $k = 1$.

Ainsi, le décryptage d'un bloc crypté $\mathbf{C}_k, k = 1,2, \cdots, K$, avec la clé secrète Q' consiste à prendre l'inverse des étapes précédentes de cryptage. Cela peut être résumé par l'équation suivante :

$$\mathbf{B}'_k = P^{-1}{}_{\{(x'_0)_k\}}\left(\mathbf{F}^{\{-a'_k,-b'_k\}}\left[P^{-1}{}_{\{(y'_0)_k\}}\left(\mathbf{F}^{\{-c'_k,-d'_k\}}[\,\mathbf{C}_k\,]\right)\right]\right) \quad (5.2)$$

Ce processus de décryptage est répété pour chaque bloc crypté $\mathbf{C}_k$, $k = 1,2, \cdots, K$, afin de former la trame décryptée finale $\mathbf{I}'_r$.

Ainsi, il est clair que les trames décryptées $\mathbf{I}'_r$ sont entièrement identiques aux trames $\mathbf{I}_r$ originales de la séquence vidéo seulement si la taille $n' \times m'$ des blocs est semblable à la

taille $n \times m$ des blocs originaux, et que les ordres fractionnaires a'_k, b'_k, c'_k, et d'_k sont identiques aux ordres fractionnaires a_k, b_k, c_k, et d_k, $k = 1,2, \cdots, K$, de plus, lorsque $k = 1$, nous devons avoir $(x'_0)_k = x_0{}'$ et $(y'_0)_k = y_0{}'$.

L'avantage principale de la méthode proposée est que pour une seule trame $\mathbf{I}_r$ nous utilisons $4 \times K$ ordres fractionnaires a_k, b_k, c_k, et d_k au lieu de 4 ordres fractionnaires dans la méthode DRPE dans le domaine de la transformée TFRD [49], ce qui améliore significativement l'espace de la clé secrète. De plus, l'utilisation de la transformée réelle TFRDR et des permutations par bloc permet de rendre l'algorithme proposé de cryptage des séquences d'images vidéo plus efficace en termes de débit de transmission et de complexité de calculs.

5.3 Résultats et discussions

Dans cette section, nous présentons les résultats de simulation de l'application de la méthode proposée de cryptage par bloc sur quelques séquences d'images vidéo de tests de format standard CIF (352×288) et de niveau de gris (8bits). Une étude pour déterminer la taille adéquate pour les blocs ainsi qu'une évaluation détaillée de la sécurité sont présentées au fur et à mesure.

Soit une séquence d'images vidéo au format CIF composée de 100 trames. Supposons Q une clé secrète qui est définie par :

- $4 \times K$ ordres fractionnaires a_k, b_k, c_k, et d_k.
- La largeur n et la longueur m du bloc.
- Les paramètres des fonctions de permutations $(x_0)_{k=1} = 0.2431$, et $(y_0)_{k=1} = 0.7916$.

Notez que le nombre d'ordres fractionnaires est variable en fonction de la taille $n \times m$ des blocs, et donc du nombre de blocs K, de ce fait, le tableau 5.1 montre quelques tailles de blocs possibles en fonction du nombre d'ordres fractionnaires correspondants pour la division d'une trame de taille 352×288.

Tableau 5.1 Exemples des tailles de blocs possibles pour une trame au format CIF en fonction du nombre d'ordres fractionnaires

Nombre d'ordres fractionnaires $(4 \times K)$	**16**	**32**	**64**	**88**	**128**	**396**
Taille du bloc $n \times m$	88×288 352×72 176×144	88×144 44×288 352×36 176×72	44×144 88×72 176×36	32×144	44×72 88×36	32×32

Supposons que nous avons quatre trames tirées aléatoirement d'une séquence d'images vidéo Tennis. La figure 5.2 montre le cryptage de ces trames en utilisant la méthode proposée avec la clé secrète Q définit précédemment.

Les figures 5.2(b) et 5.2(c) montrent les trames Tennis cryptées en blocs de grande taille 176×144, ensuite en blocs de petite taille 32×32, puis les figures 5.2(d) et 5.2(e) montrent ces trames décryptées avec une clé secrète incorrecte, ensuite avec une clé secrète correcte.

Comme les trames de la séquence d'images vidéo dans la méthode proposée dépendent seulement de leurs blocs de pixels respectifs et non des trames adjacentes, la sensibilité de la clé secrète Q est donc évaluée dans le cas du cryptage d'une seule trame de la séquence Tennis. De ce fait, une trame Tennis cryptée avec la clé secrète Q est décryptée avec une clé secrète Q' équivalente à la clé Q originale mais qui comporte une erreur dans l'un de ses différents paramètres.

Ainsi, pour une taille des blocs égale à 176×144, les figures 5.3(a) et 5.3(b) montrent la trame Tennis décryptée quand $(x_0)'_{k=1} = (x_0)_{k=1} + 10^{-16}$ et $(y_0)'_{k=1} = (y_0)_{k=1} + 10^{-16}$, puis les figures 5.3(c) et 5.3(d) montrent la trame Tennis décryptée avec des ordres fractionnaires a_k, b_k, c_k, et d_k incorrects, ensuite avec la taille des blocs considérée incorrecte.

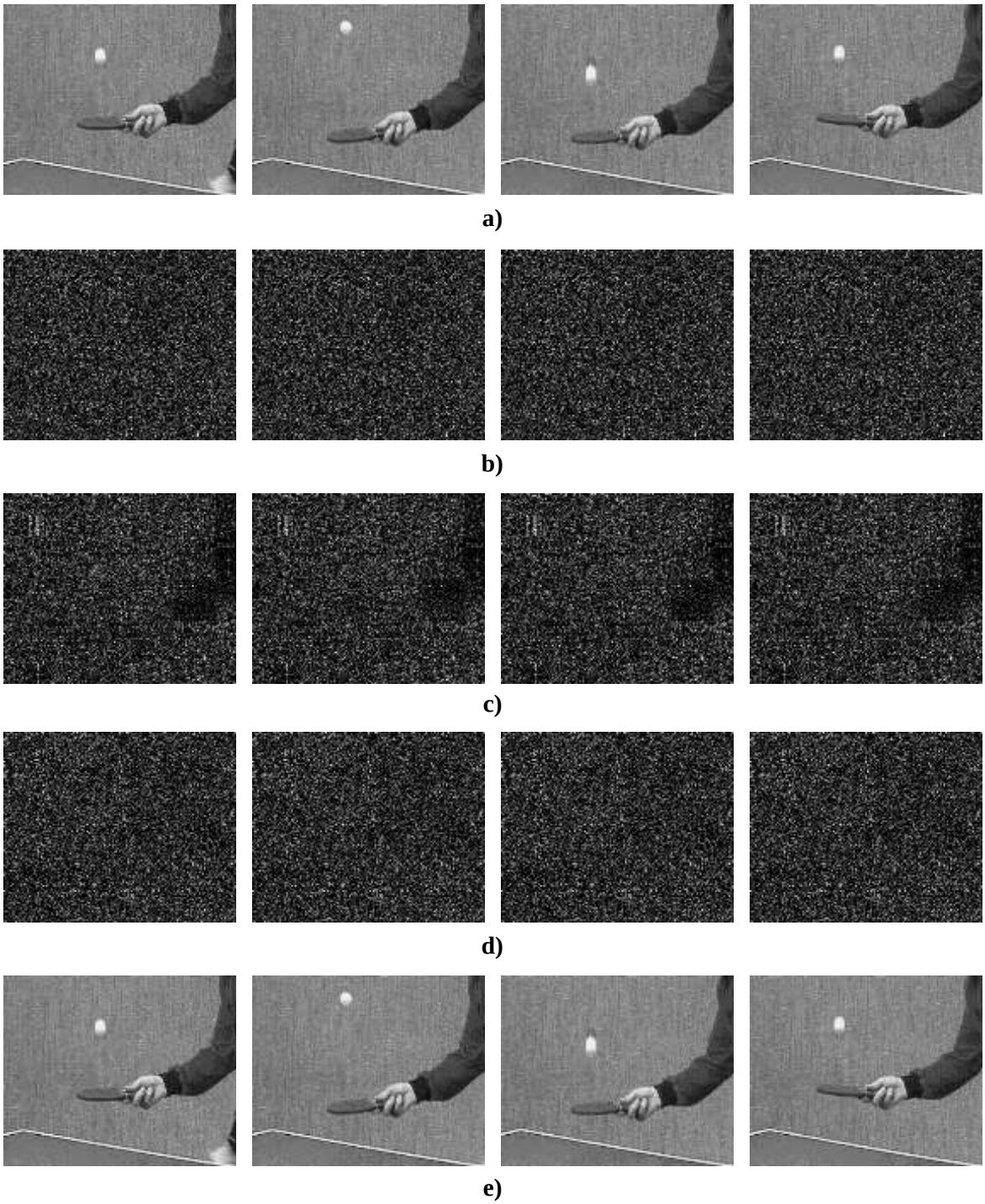
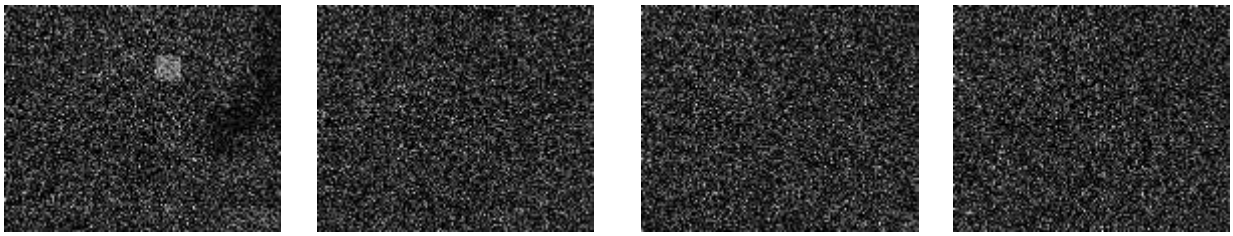


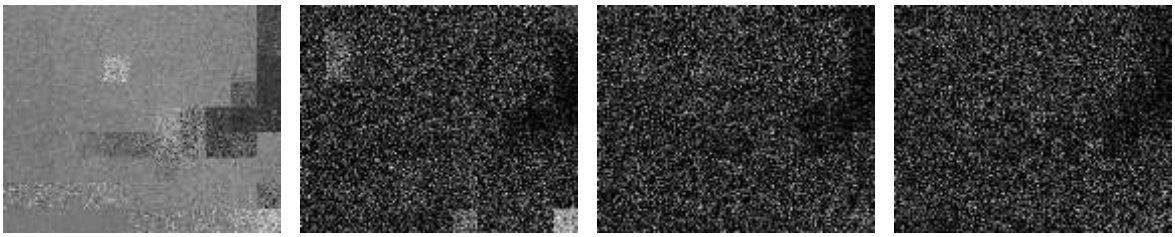
Figure 5.3 Résultats de simulation: a) les trames originales, b) les trames cryptées avec des blocs de taille 176×144, c) les trames cryptées avec des blocs de taille 32×32, d) les trames décryptées avec une clé incorrecte, e) les trames décryptées avec une clé correcte.



a) b) c) d)

Figure 5.3 Décryptage d'une trame en fonction des erreurs présentes dans les paramètres de la clé secrète avec un bloc de taille 176×144: a) $(x_0)'_{k=1} = (x_0)_{k=1} + 10^{-16}$, b) $(y_0)'_{k=1} = (y_0)_{k=1} + 10^{-16}$, c) les ordres fractionnaires a_k, b_k, c_k et d_k sont incorrects, d) la taille des blocs est incorrecte.

De plus, pour un bloc de taille relativement plus petite et égale à 32×32, les figures 5.4(a) et 5.4(b) montrent la trame Tennis décryptée quand $(x_0)'_{k=1} = (x_0)_{k=1} + 10^{-16}$ et $(y_0)'_{k=1} = (y_0)_{k=1} + 10^{-16}$, puis les figures 5.3(c) et 5.3(d) montrent la trame Tennis décryptée avec des ordres fractionnaires incorrects, ensuite avec une taille incorrecte des blocs.



a) b) c) d)

Figure 5.4 Décryptage d'une trame en fonction des erreurs présentes dans les paramètres de la clé secrète avec un bloc de taille 32×32: a) $(x_0)'_{k=1} = (x_0)_{k=1} + 10^{-16}$, b) $(y_0)'_{k=1} = (y_0)_{k=1} + 10^{-16}$, c) les ordres fractionnaires a_k, b_k, c_k, et d_k sont incorrects, d) la taille des blocs est incorrecte.

D'après les résultats des figures 5.3 et 5.4 nous remarquons que la méthode proposée est sensible aux erreurs dans les paramètres de la clé secrète, cependant, quand la taille des blocs est considérablement petite par rapport à la taille de la trame, nous constatons de visu que la trame décryptée est moins sensible aux erreurs, de ce fait et afin de comparer objectivement la sensibilité de la clé secrète en fonction de la taille des blocs, nous proposons de décrypter la

trame Tennis en fonction d'une erreur δ dans les paramètres a_k, b_k, c_k, et d_k de la transformée TFRDR, ensuite l'EQM entre la trame décryptée et la trame originale est calculée en fonction de l'erreur δ et les résultats de calculs sont tracés dans la figure 5.5.

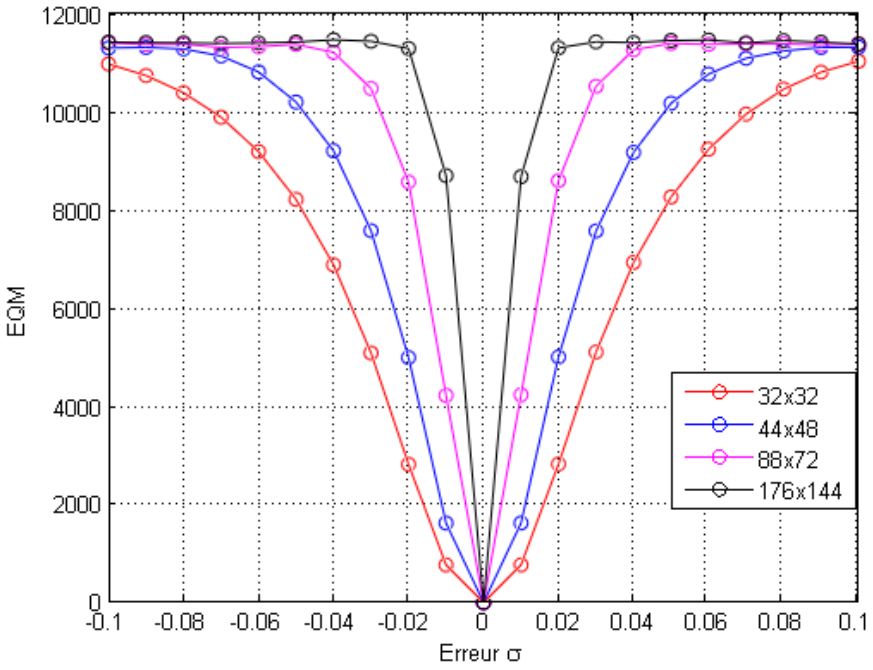


Figure 5.5 EQM en fonction d'une erreur δ dans les paramètres de la transformée TFRDR.

Nous remarquons que d'après ces résultats, la clé secrète est sensible aux erreurs dans les paramètres a_k, b_k, c_k, et d_k de la transformée TFRDR, cependant, la sensibilité est variable en fonction de la taille du bloc, plus le bloc est grand plus la clé est sensible. Afin d'avoir une sensibilité plus acceptable quel que soit la taille du bloc, nous proposons de restreindre la taille des blocs à $176 \times m$ avec $m \in \{16, 18, 24, 32, 36, 48, 72, 96\}$, ou bien à $n \times 144$ avec $n \in \{16, 22, 32, 44, 88\}$. Les résultats du calcul de l'EQM dans ce cas sont illustrés dans la figure 5.6 pour $m = 32$ et $n = 32$. Des résultats similaires ont été obtenu en considérant d'autres valeurs suggérées de n et m.

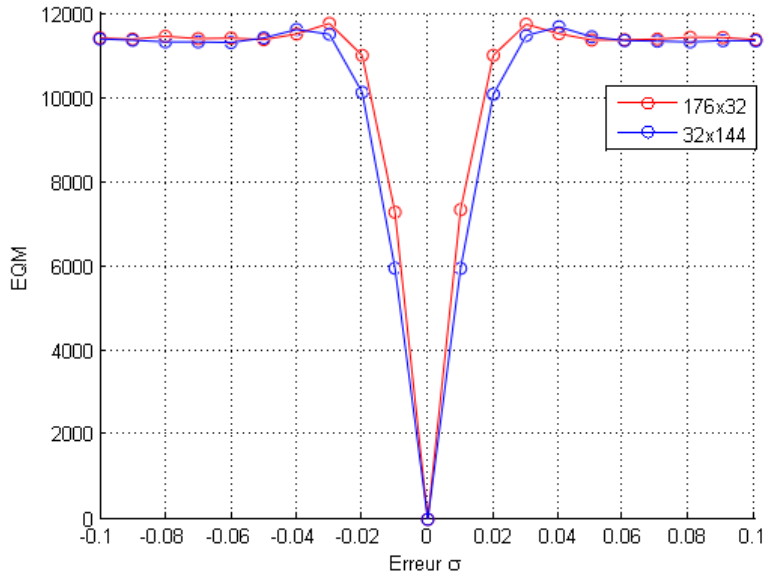
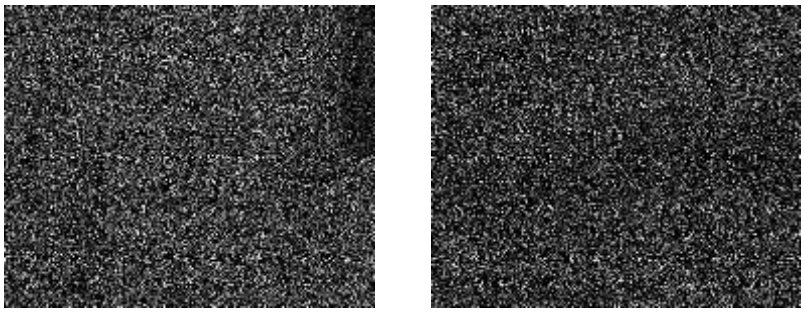


Figure 5.6 EQM en fonction de l'erreur δ pour des tailles prédéterminées.

Ces résultats peuvent être inspectés visuellement pour un seuil d'erreur $\delta = 0.02$ sur la figure 5.7 qui montre l'image décryptée lorsque la taille des blocs est de 32×144 et 176×32. Nous remarquons que dans les deux cas, la trame reste correctement cryptée. Cela est dû au fait qu'ils ont la même sensibilité aux erreurs dans les paramètres de la transformée TFRDR.



a) b)

Figure 5.7 Trame décryptée avec une erreur $\delta = 0.02$ dans les paramètres de la transformée TFRDR pour des: a) blocs de taille 32×144, b) blocs de taille 176×32.

Afin de comparer la méthode proposée avec celle de Jindal et al. [49], nous calculons l'EQM en fonction des paramètres de la transformée complexe TFRD dans le cas de la méthode dans [49] et en fonction des paramètres de la transformée TFRDR réelle utilisée dans notre méthode. Les résultats de cette comparaison sont illustrés dans la figure 5.8.

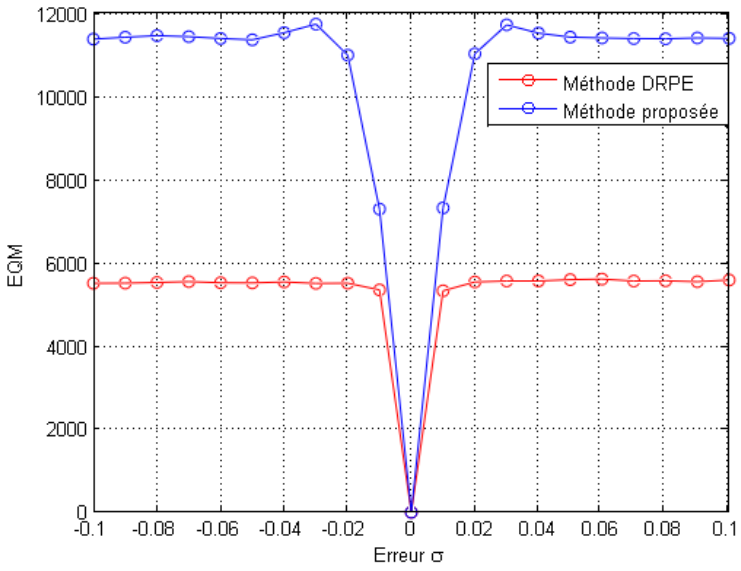


Figure 5.8 EQM résultant de la méthode proposée et la méthode DRPE dans [49].

Nous remarquons que la méthode proposée est plus sensible aux erreurs que la méthode DRPE dans [49]. De ce fait, nous pouvons estimer l'espace de la clé dans la méthode proposée qui est de l'ordre de $10^{16} \times 10^{16} \times (10^2)^{4K}$, où K est le nombre de blocs dérivé selon la taille $n \times m$ des blocs. Cet espace est largement supérieur à 2^{100} qui est le minimum requis pour résister à une attaque par force brute. En conséquence, la méthode proposée est robuste contre les attaques par force brute.

Maintenant, pour évaluer la sécurité de la méthode proposée contre l'analyse statistique par histogramme, les figures 5.9(a) et 5.9(b) montrent différentes trames tirées de la séquence vidéo Tennis ainsi que leurs histogrammes correspondants, puis la figure 5.8(c) montre les histogrammes des trames cryptées correspondantes.

De la même façon, les figures 5.10(a) et 5.10(b) montrent différentes trames tirées de la séquence vidéo CoastGuard ainsi que leurs histogrammes correspondants, puis la figure 5.10(c) montre les histogrammes des trames cryptées correspondantes. Du fait que l'utilisation de la transformée réelle TFRDR, les histogrammes des trames cryptées indiquent la distribution des valeurs absolues des pixels.

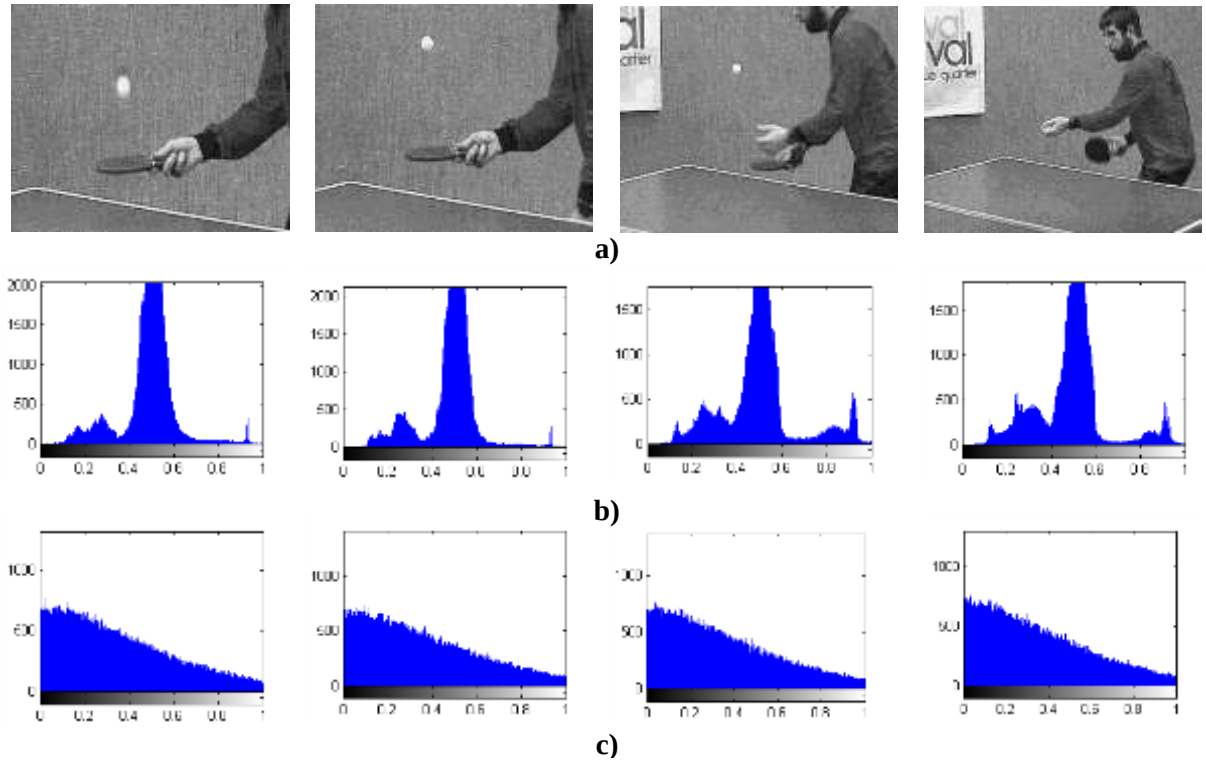


Figure 5.9 Histogrammes de quelques trames de la séquence vidéo Tennis: a) trames originales, b) histogrammes des trames originales, et c) histogrammes des trames cryptées.

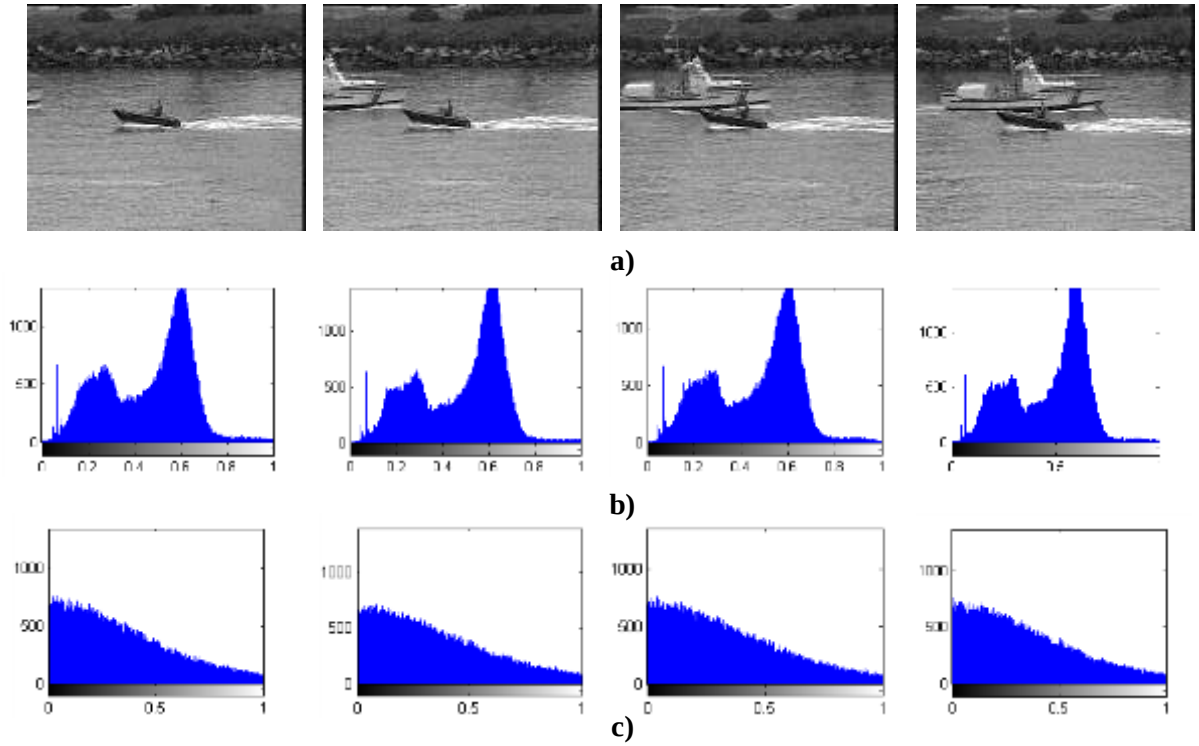
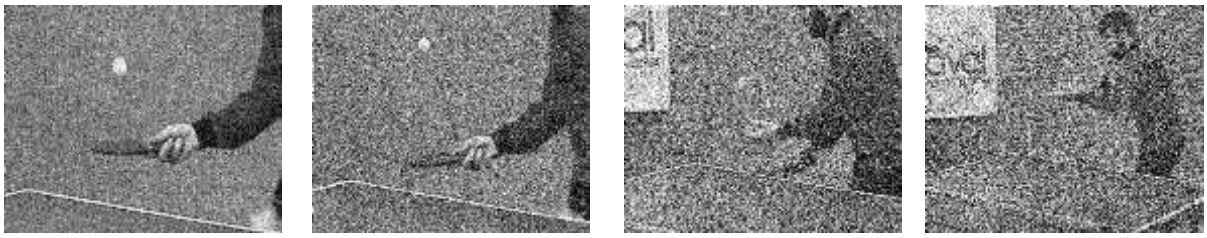


Figure 5.10 Histogrammes de quelques trames de la séquence vidéo CoastGuard: a) trames originales, b) histogrammes des trames originales, et c) histogrammes des trames cryptées.

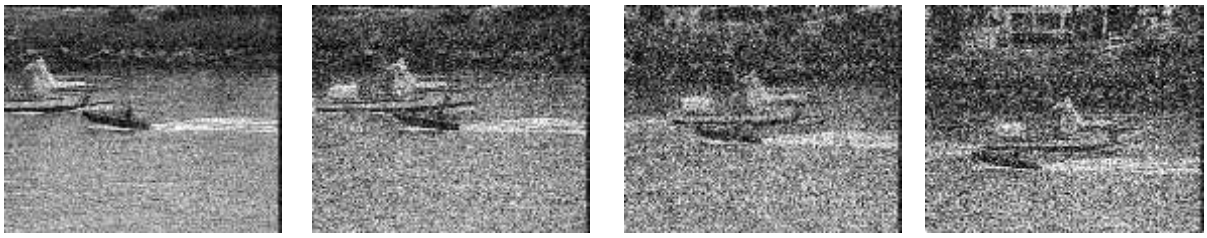
Nous remarquons que les histogrammes des trames cryptées sont entièrement différents de ceux des trames originales et ils sont généralement identiques peu importe la trame originale. Cela permet de réduire le risque des attaques statistiques basées sur la collecte d'informations sur les trames originales en analysant les histogrammes des trames cryptées. En conséquence, la méthode proposée est robuste contre l'analyse statistique par histogramme.

Pour voir la résistance de la méthode proposée contre le bruit, la figure 5.11(a) montre quatre trames d'une séquence vidéo Tennis ainsi que leurs PSNR lorsqu'elles sont décryptées avec un bruit blanc Gaussien additif de coéfficient de puissance variable σ, de la méme maniére, la figure 5.11(b) montre les résultas de décryptage des trames d'une séquence vidéo Coastguard. Nous remarquons que les trames restent identifiables malgré la présence du bruit. De plus, la figure 5.12 montre l'EQM calculée entre une trame Tennis décryptée et celle originale correspondante en fonction du coefficient de puissance variable σ. De mémes résultas ont été obtenus pour différentes tailles des blocs, de ce fait, la méthode proposée résiste aux erreurs du bruit.



$\sigma = 0.3$ PSNR $= 16$dB | $\sigma = 0.5$ PSNR $= 12$dB | $\sigma = 0.7$ PSNR $= 10$dB | $\sigma = 0.8$ PSNR $= 9.5$dB

a)



$\sigma = 0.3$ PSNR $= 16$dB | $\sigma = 0.5$ PSNR $= 13$dB | $\sigma = 0.7$ PSNR $= 11$dB | $\sigma = 0.8$ PSNR $= 10$dB

b)

Figure 5.11 Séquence d'images vidéo décryptée en fonction du coefficient du bruit additif : a) séquence Tennis, b) séquence Coastguard.

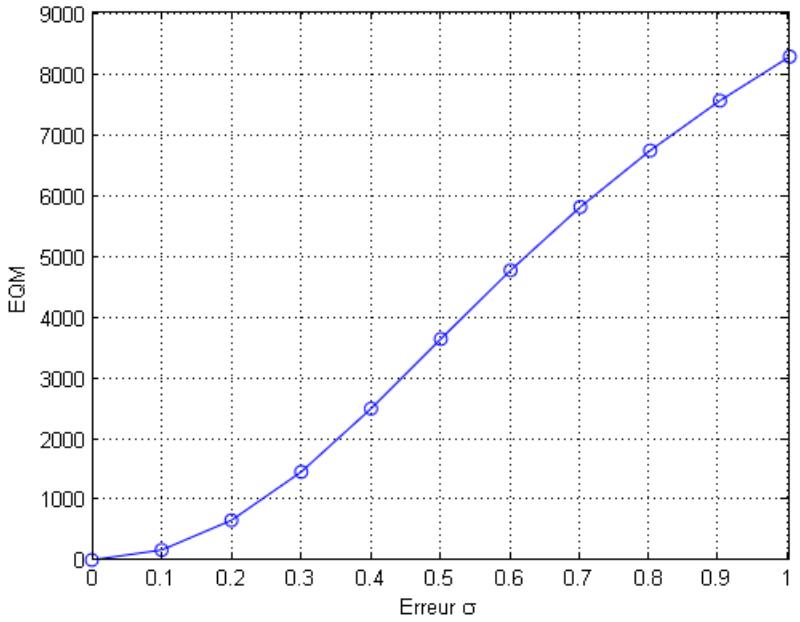
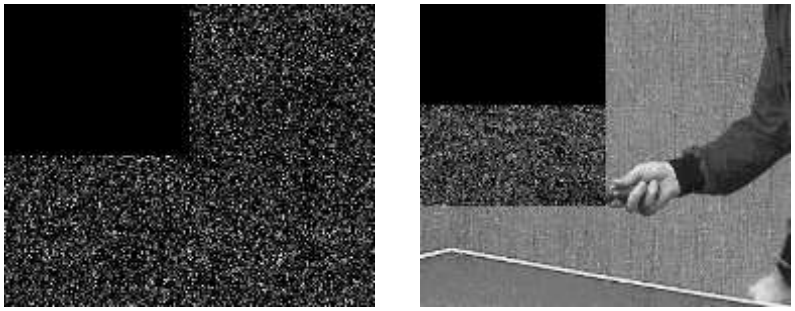


Figure 5.12 EQM en fonction du coefficient de puissance σ du bruit additif.

Au final, pour évaluer la résistance de la méthode proposée contre les erreurs de transmission, nous supposons qu'une partie d'une trame Tennis cryptée a été corrompu ou perdu au cours du transit dans le canal de communication. Les résultats de décryptage dans ce cas sont illustrés dans la figure 5.13.



a) b)

Figure 5.13 Une trame Tennis décryptée après qu'une partie des pixels est perdue:
a) Trame cryptée, b) Trame décryptée.

Nous remarquons que la trame décryptée a perdu une partie de ces paramètres ce qui est un inconvénient, cela est causé par l'utilisation de l'architecture de cryptage bloc par bloc, ainsi, un compromis entre sécurité et performances doit être considéré au préalable avant d'utiliser

la méthode proposée. Mais, généralement, cette attaque n'est pas considérée dans les séquences vidéo.

5.4 Conclusion

Dans ce chapitre, nous avons proposé une méthode efficace pour le cryptage des séquences d'images vidéo. Cette méthode permet de crypter individuellement chaque trame d'une séquence vidéo en utilisant une transformée réelle qui est la transformée TFRDR et des fonctions de permutation, ce qui est avantageux en débit de transmission et en complexité de calculs. De plus, un schéma de cryptage par bloc a été adopté et introduit afin d'améliorer la sensibilité et l'espace de la clé. Nous avons aussi effectué une étude détaillée sur le choix de la taille des blocs. Cette étude nous a permis de suggérer des tailles appropriées qui assurent une sensibilité acceptable de la clé. Enfin, les résultats de simulation montrent clairement la faisabilité de la méthode proposée ainsi que sa résistance contre les attaques statistiques, par force brute et du bruit.

Conclusion générale et perspectives

Conclusion générale et perspectives

Le développement et l'implémentation de nouvelles méthodes de cryptage d'images et vidéo basées sur les transformées paramétriques ont fait l'objet de notre thèse. Après avoir effectué des recherches avancées sur la méthode de cryptage DRPE dans le domaine des transformées paramétriques et étudié en détail ses différentes versions, nous avons pu élaborer de nouvelles méthodes de cryptage d'images et vidéo. Ces versions utilisent généralement le chaos pour la génération des masques de phases aléatoires de la méthode DRPE ou pour contrôler une fonction de permutation chaotique afin d'améliorer la sensibilité et l'espace de la clé secrète.

Ainsi, dans notre première contribution, nous avons proposé une méthode de cryptage d'images basée sur la transformée ROP. Cette méthode a été développée au début dans le cas de cryptage d'une seule image par l'utilisation de la transformée ROP avec une permutation chaotique, ensuite cette idée a été élargie pour le cryptage de deux images en même temps, où nous avons proposé d'utiliser une fonction de permutation chaotique complexe. Les résultats de comparaison montrent clairement que la méthode proposée est plus efficace que les méthodes existantes basées sur la transformée ROP en termes de sensibilité et d'espace de la clé secrète.

Notre deuxième contribution est la proposition d'un nouveau prétraitement non-linéaire pour le cryptage DRPE afin de résoudre le problème de sa linéarité et renforcer sa sécurité. Ce prétraitement non-linéaire qui est introduit dans le domaine spatial avant l'application d'une méthode DRPE est basé sur l'utilisation d'une combinaison de suites chaotiques associées à une fonction XOR. Nous avons exploité ce nouveau prétraitement non-linéaire pour développer deux méthodes différentes de cryptage d'images. La première méthode qui est un cryptage purement numérique utilise la méthode DRPE dans le domaine de la transformée ROP. La seconde méthode qui est un cryptage hybride opto-numérique a été développée en utilisant la transformée TFRD à paramètres multiples et des suites PLCM. Enfin, le nouveau prétraitement non-linéaire proposé a permis d'améliorer significativement les méthodes de cryptage existantes basées sur les transformées paramétriques, notamment en termes de sensibilité et d'espace de la clé secrète ainsi que de sa résistance aux attaques de cryptanalyse à texte en clair choisi.

Notre troisième contribution est la proposition d'une nouvelle méthode de cryptage des séquences d'images vidéo en introduisant une structure de cryptage DRPE par bloc basée sur la transformée TFRDR et des permutations chaotiques. Cette méthode remédie les problèmes de débit de transmission et de complexité de calculs de la méthode DRPE utilisée dans le cryptage des séquences d'images vidéo. Elle présente aussi une sensibilité et un espace de la clé secrète plus élevés.

Enfin, toutes les méthodes proposées dans ce travail ont été implémentées sur le logiciel MATLAB et appliquées pour le cryptage des images et des séquences vidéo de tests standards. Une évaluation méthodique de la sécurité en termes d'attaques par force brute et d'attaques statistiques a été suivie pour chacune des méthodes proposées. Nous avons vérifié également leurs résistances contre le bruit additif et les erreurs de transmission dans le canal de communication. Les résultats de simulation ont démontré l'apport considérable des méthodes proposées par rapport aux méthodes existantes ainsi que l'intérêt de leurs utilisations.

Dû au fait que la transformée ROP a un nombre très élevé de paramètres indépendants qui sont hautement désirés en cryptage, il est fortement souhaitable d'avoir son implémentation optique en plus de son implémentation numérique actuelle. Comme deuxième perspective, le développement de nouvelles fonctions non linéaires plus complexes peut aboutir à un prétraitement plus robuste que celui proposé dans ce travail.

Bibliographie

1. J. Dumas, J. Roch, E. Tannier, and S. Varrette, “Théorie des codes - Compression, cryptage, correction,” Dunod, France. 2007.

2. B. Furht, E. Muharemagic, and D. Socek, *Multimedia Encryption and Watermarking*, Springer Science & Business Media, 2005.

3. B. Schneier, *Cryptographie appliquée: algorithmes, protocoles et codes source en C*, Vuibert Informatique, 2001.

4. D. Stinson, *Cryptographie-Théorie et pratique*. International Thomson Publishing, France, 1996.

5. El-Samie, Fathi E. Abd, et al, *Image encryption: a communication perspective*, CRC Press, 2013.

6. S. Lian, *Multimedia content encryption: techniques and applications*, CRC press, 2008.

7. P. Refregier and B. Javidi, “Optical image encryption based on input plane,” *Opt. Lett.*, vol. 20, no. 7, pp. 767–769, 1995.

8. S. Liu, C. Guo, and J. T. Sheridan, “A review of optical image encryption techniques,” *Opt. Laser Technol.*, vol. 57, pp. 327–342, Apr. 2014.

9. G. Unnikrishnan, and K. Singh, “Double random fractional Fourier-domain encoding for optical security,” vol. 39, no. November, pp. 2853–2859, 2000.

10. S. Liu, Q. Mi, and B. Zhu, “Optical image encryption with multistage and multichannel fractional Fourier-domain filtering,” *Opt. Lett.*, vol. 26, no. 16, pp. 1242–1244, 2001.

11. S. Liu, L. Yu, and B. Zhu, “Optical image encryption by cascaded fractional Fourier transforms with random phase filtering,” *Opt. Commun.*, vol. 187, no. January, pp. 57–63, 2001.

12. Y. Zhang, C. Zheng, and N. Tanno, “Optical encryption based on iterative fractional Fourier transform,” vol. 202, no. February, pp. 277–285, 2002.

13. B. M. Hennelly and J. T. Sheridan, “Image encryption and the fractional Fourier transform,” Int. J. Light Electron Opt., vol. 114, no. 6, pp. 251–265, 2003.

14. S. Zhang and M. A. Karim, “Color Image Encryption Using Double Random Phase Encoding,” *Microw. Opt. Technol. Lett.*, vol. 21, no. 5, pp. 318–323, 1999.

15. M. Joshi and K. Singh, “Color image encryption and decryption using fractional Fourier transform,” Opt. Commun., vol. 279, no. 1, pp. 35–42, 2007.

16. M. Joshi and K. Singh, “Color image encryption and decryption for twin images in fractional Fourier domain,” Opt. Commun., vol. 281, no. 23, pp. 5713–5720, Dec. 2008.

17. G. Situ and J. Zhang, “Double random-phase encoding in the Fresnel domain.,” Opt. Lett., vol. 29, no. 14, pp. 1584–1586, 2004.

18. J. A. Rodrigo, T. Alieva, and M. L. Calvo, “Applications of gyrator transform for image processing,” *Opt. Commun.*, vol. 278, pp. 279–284, 2007.

19. Z. Liu, L. Xu, C. Lin, J. Dai, and S. Liu, “Image encryption scheme by using iterative random phase encoding in gyrator transform domains,” *Opt. Lasers Eng.*, vol. 49, no. 4, pp. 542–546, Apr. 2011.

20. M. Joshi, C. Shakher, and K. Singh, “Image encryption and decryption using fractional Fourier transform and radial Hilbert transform,” Opt. Lasers Eng., vol. 46, no. 7, pp. 522–526, Jul. 2008.

21. M. Joshi, C. Shakher, and K. Singh, “Image encryption using radial Hilbert transform filter bank as an additional key in the modified double random fractional Fourier encoding architecture,” *Opt. Lasers Eng.*, vol. 48, no. 5, pp. 605–615, May 2010.

22. M. Joshi, C. Shakher, and K. Singh, “Fractional Fourier plane image encryption technique using radial hilbert-, and Jigsaw transform,” *Opt. Lasers Eng.*, vol. 48, no. 7–8, pp. 754–759, Jul. 2010.

23. Z. Liu, M. A. Ahmad, and S. Liu, “Image encryption based on double random amplitude coding in random Hartley transform domain,” Opt. - Int. J. Light Electron Opt., vol. 121, no. 11, pp. 959–964, Jun. 2010.

24. S. Pei and W. Hsue, “The Multiple-Parameter Discrete Fractional Fourier Transform,” IEEE Signal Process. Lett., vol. 13, no. 6, pp. 329–332, 2006.

25. S. Bouguezel, “Image Encryption using the Reciprocal-Orthogonal Parametric Transform,” *Proc. 2010 IEEE Int. Symp. Circuits Syst.*, no. iii, pp. 2542–2545, 2010.

26. S. Bouguezel, “A Reciprocal-Orthogonal Parametric Transform and Its Fast Algorithm,” *IEEE Signal Process. Lett.*, vol. 19, no. 11, pp. 769–772, Nov. 2012.

27. S. Bouguezel, M. O. Ahmad, and M. N. S. Swamy, “A new involutory parametric transform and its application to image encryption,” *2013 IEEE Int. Symp. Circuits Syst.*, no. 3, pp. 2605–2608, May 2013.

28. R. Tao, J. Lang, and Y. Wang, “The multiple-parameter discrete fractional Hadamard transform,” *Opt. Commun.*, vol. 282, no. 8, pp. 1531–1535, Apr. 2009.

29. J. Wu, L. Zhang, and N. Zhou, “Image encryption based on the multiple-order discrete fractional cosine transform,” *Opt. Commun.*, vol. 283, no. 9, pp. 1720–1725, May 2010.

30. N. Zhou, T. Dong, and J. Wu, “Novel image encryption algorithm based on multiple-parameter discrete fractional random transform,” *Opt. Commun.*, vol. 283, no. 15, pp. 3037–3042, Aug. 2010.

31. X. Wang, H. Zhai, Z. Li, and Q. Ge, “Double random-phase encryption based on discrete quaternion fourier-transforms,” *Opt. - Int. J. Light Electron Opt.*, vol. 122, no. 20, pp. 1856–1859, Oct. 2011.

32. S. Bouguezel, M. O. Ahmad, and M. N. S. Swamy, “Binary Discrete Cosine and Hartley Transforms,” IEEE Trans. Circuits Syst. I Regul. Pap., pp. 1–1, 2012.

33. B. Hennelly and J. T. Sheridan, “Optical image encryption by random shifting in fractional Fourier domains.,” Opt. Lett., vol. 28, no. 4, pp. 269–71, 2003.

34. N. Singh and A. Sinha, “Optical image encryption using fractional Fourier transform and chaos,” Opt. Lasers Eng., vol. 46, no. 2, pp. 117–123, Feb. 2008.

35. N. Singh and A. Sinha, “Gyrator transform-based optical image encryption, using chaos,” *Opt. Lasers Eng.*, vol. 47, no. 5, pp. 539–546, May 2009.

36. J. Lang, R. Tao, and Y. Wang, “Image encryption based on the multiple-parameter discrete fractional Fourier transform and chaos function,” *Opt. Commun.*, vol. 283, no. 10, pp. 2092–2096, May 2010.

37. Z. Liu, H. Chen, T. Liu, P. Li, L. Xu, J. Dai, and S. Liu, “Image encryption by using gyrator transform and Arnold transform,” J. Electron. Imaging, vol. 20, no. 1, p. 013020, 2011.

38. N. Zhou, Y. Wang, L. Gong, H. He, and J. Wu, “Novel single-channel color image encryption algorithm based on chaos and fractional Fourier transform,” *OPTICS*, vol. 284, no. 12, pp. 2789–2796, 2011.

39. J. Lang, “Image encryption based on the reality-preserving multiple-parameter fractional Fourier transform,” Opt. Commun., vol. 285, no. 10–11, pp. 2584–2590, 2012.

40. S. Liu and J. T. Sheridan, “Optical encryption by combining image scrambling techniques in fractional Fourier domains,” *Opt. Commun.*, vol. 287, pp. 73–80, 2013.

41. Z. Liu, S. Li, W. Liu, and S. Liu, “Opto-digital image encryption by using Baker mapping and 1-D fractional Fourier transform,” *Opt. Lasers Eng.*, vol. 51, no. 3, pp. 224–229, Mar. 2013.

42. Z. Liu, S. Li, W. Liu, Y. Wang, and S. Liu, “Image encryption algorithm by using fractional Fourier transform and pixel scrambling operation based on double random phase encoding,” *Opt. Lasers Eng.*, vol. 51, no. 1, pp. 8–14, Jan. 2013.

43. H. Chen, J. Zhao, Z. Liu, and X. Du, “Opto-digital spectrum encryption by using Baker mapping and gyrator transform,” Opt. Lasers Eng., vol. 66, pp. 285–293, Mar. 2015.

44. R. Tao, Y. Xin, and Y. Wang, “Double image encryption based on random phase encoding in the fractional Fourier domain.,” *Opt. Express*, vol. 15, no. 24, pp. 16067–79, 2007.

45. H. Li and Y. Wang, “Double-image encryption based on discrete fractional random transform and chaotic maps,” *Opt. Lasers Eng.*, vol. 49, no. 7, pp. 753–757, Jul. 2011.

46. M. Shan, J. Chang, Z. Zhong, and B. Hao, “Double image encryption based on discrete multiple-parameter fractional Fourier transform and chaotic maps,” *Opt. Commun.*, vol. 285, no. 21–22, pp. 4227–4234, Oct. 2012.

47. Z. Zhong, J. Chang, M. Shan, and B. Hao, “Fractional Fourier-domain random encoding and pixel scrambling technique for double image encryption,” *Opt. Commun.*, vol. 285, no. 1, pp. 18–23, Jan. 2012.

48. Y.Zhang and D.Xiao, “Double optical image encryption using discrete Chirikov standard map and chaos-based fractional random transform,” Optics and Lasers in Engineering, vol. 4, pp. 472-480, 2013

49. N. Jindal and K. Singh, “Image and video processing using discrete fractional transforms,” *Signal, Image Video Process.*, Oct. 2012.

50. S. E. Azoug and S. Bouguezel, “Double Image Encryption Based on the Reciprocal- Orthogonal Parametric Transform and Chaotic Maps,” 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA), pp. 156–161, Algiers, Algeria, 2013.

51. S. Bouguezel, M. Ahmad, and M. N. S. Swamy, “A New Class of Reciprocal Orthogonal Parametric Transforms,” *Circuits Syst. I Regul. Pap. IEEE Trans.*, vol. 56, no. 4, pp. 795–805, 2009.

52. L. Kocarev and S. Lian, “Chaos-Based Cryptography - Theory, Algorithms and Applications”, Springer-Verlag Berlin Heidelberg, 2011.

53. A. Carnicer, M. Montes-Usategui, S. Arcos, and I. Juvells, “Vulnerability to chosen-ciphertext attacks of optical encryption schemes based on double random phase keys.,” Opt. Lett., vol. 30, no. 13, pp. 1644–6, Jul. 2005.

54. X. Peng, P. Zhang, H. Wei, and B. Yu, “Known-plaintext attack on optical encryption based on double random phase keys.,” Opt. Lett., vol. 31, no. 8, pp. 1044–6, Apr. 2006.

55. X. Peng, H. Wei, and P. Zhang, “Chosen-plaintext attack on lensless double-random phase encoding in the Fresnel domain.,” *Opt. Lett.*, vol. 31, no. 22, pp. 3261–3263, 2006.

56. X. Peng, H. Wei, and P. Zhang, “Chosen-plaintext attack on lensless double-random,” vol. 31, no. 22, pp. 3261–3263, 2006.

57. U. Gopinathan, D. S. Monaghan, T. J. Naughton, and J. T. Sheridan, “A known-plaintext heuristic attack on the Fourier plane encryption algorithm.,” Opt. Express, vol. 14, no. 8, pp. 3181–6, Apr. 2006.

58. Y. Frauel, A. Castro, T. J. Naughton, and B. Javidi, “Resistance of the double random phase encryption against various attacks.,” Opt. Express, vol. 15, no. 16, pp. 10253–65, Aug. 2007.

59. S. David, J. Thomas, T. John, D. S. Monaghan, G. Situ, and U. Gopinathan, “Role of phase key in the double random phase encoding technique : an error analysis,” 2008.

60. W. Qin and X. Peng, “Vulnerability to known-plaintext attack of optical encryption schemes based on two fractional Fourier transform order keys and double random phase keys,” J. Opt. A Pure Appl. Opt., vol. 11, no. 7, p. 075402, Jul. 2009.

61. Q. Ran, H. Zhang, J. Zhang, L. Tan, and J. Ma, “Deficiencies of the cryptography based on multiple-parameter fractional Fourier transform,” Opt. Lett., vol. 34, no. 11, pp. 1729–1731, 2009.

62. W. Qin, X. Peng, X. Meng, and W. He, “Improved Known-Plaintext Attack on Optical Encryption Based on Double Random Phase Encoding,” 2010 Symp. Photonics Optoelectron., pp. 1–4, Jun. 2010.

63. Y. Zhang, D. Xiao, W. Wen, and H. Liu, “Vulnerability to chosen-plaintext attack of a general optical encryption model with the architecture of scrambling-then-double random phase encoding.,” Opt. Lett., vol. 38, no. 21, pp. 4506–9, Nov. 2013.

64. S. Li, C. Li, G. Chen, N. G. Bourbakis, and K.-T. Lo, “A general quantitative cryptanalysis of permutation-only multimedia ciphers against plaintext attacks,” Signal Process. Image Commun., vol. 23, no. 3, pp. 212–223, 2008.

65. S. E. Azoug and S. Bouguezel, “A New Method Based on the ROP Transform and Chaotic Maps for Image Encryption,” International Congress on Telecommunication and Application’14, Bejaia, Algeria, 2014.

66. S. E. Azoug and S. Bouguezel, “A non-linear preprocessing for opto-digital image encryption using multiple-parameter discrete fractional Fourier transform,” Opt. Commun., vol. 359, pp. 85–94, 2016.

67. FIPS 197. Advanced Encryption Standard (AES). November 2001.

68. C. E. Shannon, “Communication Theory of Secrecy Systems, Bell System Technical Journal,” vol. 28, no. 4, pp. 656–715, 1949.

69. Auguste Kerckhoffs, “La cryptographie militaire, ” Journal des sciences militaires, vol. IX, pp. 5–38, 1883.

70. G. Alvarez and S. Li, “Some Basic Cryptographic Requirements for Chaos-Based,” Int. J. Bifurc. Chaos, vol. 16, no. 8, pp. 2129–2151, 2006.

71. H. Zhou, X. Ling, “Generating chaotic secure sequences with desired statistical properties and high security,” Int. J. Bifurc. Chaos., vol. 7, pp. 205–213, 1997.

72. S.Li, Q.Li, W.Li, X.Mou, and Y.Cai,”Statistical Properties of Digital Piecewise Linear Chaotic Maps and Their Roles in Cryptography and Pseudo-Random Coding,” Cryptography and Coding, Lect. Notes Comput. Sci., pp. 205-221, 2001.

73. H. Zhou, X. T. Ling, “Problems with the chaotic inverse system encryption approach,” IEEE Trans. Circuits and Systems, vol. 44, no. 3, pp. 268–271, 1997.

74. E. Sejdic, I. Djurović, and Lj. Stanković, “Fractional Fourier transform as a signal processing tool: An overview of recent developments,” Signal Processing, vol. 91, no. 6, pp. 1351–1369, Jun. 2011.

75. S. C. Pei and M. H. Yeh, “Improved discrete fractional Fourier transform.,” Opt. Lett., vol. 22, no. 14, pp. 1047–9, Jul. 1997.

76. S.C. Pei and M.H. Yeh, “Two dimensional discrete fractional Fourier transform,” Signal Processing, vol. 67, no. 1, pp. 99–108, May 1998.

77. C. Candan, M. A. Kutay, and H. M. Ozaktas, “The discrete fractional Fourier transform,” IEEE Trans. Signal Process., vol. 48, no. 5, pp. 1329–1337, 2000.

78. H.M.Ozaktas, Z.Zalevsky, and M.A. Kutay, “The Fractional Fourier Transform with Applications in Optics and Signal Processing,” Wiley, 2000.

79. I.Venturini and P.Duhamel, “Reality preserving fractional transforms,” IEEE Int. Conf. Acoust. Speech, Signal Process., vol. 5, pp. 205–208, 2004.

80. F. J. Marinho and L. M. Bernardo, “Numerical calculation of fractional Fourier transforms with a single fast-Fourier-transform algorithm,” J. Opt. Soc. Am. A, vol. 15, no. 8, p. 2111, 1998.

81. B. Javidi and T. Nomura, “Securing information by use of digital holography.,” Opt. Lett., vol. 25, no. 1, pp. 28–30, 2000.

82. B. O. Matoba, T. Nomura, E. Pe, and B. Javidi, “Optical Techniques for Information Security,” Proc. IEEE, vol. 97, no. 6, 2009.

83. J.Lang, R.Tao, and Y.Wang, “The discrete multiple-parameter fractional Fourier transform,” Science China Information Sciences, vol. 11, pp. 2287-229, 2010.

84. S. Bouguezel, M. O. Ahmad, and M. N. S. Swamy, “New Parametric Discrete Fourier and Hartley Transforms, and Algorithms for Fast Computation,” IEEE Trans. Circuits Syst. I Regul. Pap., vol. 58, no. 3, pp. 562–575, Mar. 2011.

85. M. H. Lee, X.-D. Zhang, W. Song, and X.-G. Xia, “Fast Reciprocal Jacket Transform With Many Parameters,” IEEE Trans. Circuits Syst. I Regul. Pap., vol. 59, no. 7, pp. 1472–1481, Jul. 2012.

86. C. Cheng and M. Chen, “Polarization encoding for optical encryption using twisted nematic liquid crystal spatial light modulators,” *Opt. Commun.*, vol. 237, pp. 45–52, 2004.

Abstract

In this thesis, we present three contributions in the field of image/video encryption based on the parametric transforms and chaos. In the first contribution, we propose an encryption DRPE method based on the ROP transform for the encryption of one or various images simultaneously. In the second contribution, we develop a new non-linear pre-processing for the DRPE encryption in order to solve the problem of its linearity and reinforce its security. In the third contribution, we propose a novel method for video sequences encryption by introducing a bloc-based DRPE encryption structure using the TFRDR in order to solve the problem of transmission rate and computational complexity of the DRPE method used in video encryption. Finally, we implement and apply the methods proposed in this work for encrypting test images and video sequences. Moreover, we methodically perform the security evaluation in terms of brute force and statistical attacks as well as comparisons with the existing methods in terms of secret key sensitivity and space. In addition, we verify their resistance against additive noise and channel communication transmission errors.

Keywords: Image Encryption, DRPE method, parametric transforms, chaotic permutation.

Résumé

Dans cette thèse, nous présentons trois contributions dans le domaine de cryptage d'image/vidéo basées sur les transformées paramétriques et le chaos. Dans la première contribution, nous proposons une méthode de cryptage DRPE basée sur la transformée ROP pour le cryptage d'une seule ou plusieurs images en même temps. Dans la deuxième contribution, nous développons un nouveau prétraitement non-linéaire pour le cryptage DRPE afin de résoudre le problème de sa linéarité et de renforcer sa sécurité. Dans la troisième contribution, nous proposons une nouvelle méthode de cryptage des séquences vidéo en introduisant une structure de cryptage DRPE par bloc basée sur la transformée TFRDR afin de pallier au problème du débit de transmission et de complexité de calculs de la méthode DRPE utilisée en cryptage vidéo. Enfin, nous implémentons et appliquons les méthodes proposées dans ce travail pour le cryptage des images et des séquences vidéo de tests standards. Aussi, nous effectuons une évaluation méthodique de la sécurité en termes d'attaques par force brute et d'attaques statistiques ainsi que des comparaisons avec des méthodes existantes en termes de sensibilité et d'espace de la clé secrète. De plus, nous vérifions également leurs résistances contre le bruit additif et les erreurs de transmission dans le canal de communication.

Mots-clés: Cryptage d'images, Méthode DRPE, Transformées paramétriques, Permutations chaotiques.

ملخص

في هذه الأطروحة، نقترح ثلاث طرق تشفير جديدة في مجال تشفير الصور و الفيديو, هذه الطرق تعتمد على استعمال التحويلات ذات معاملات عشوائية . في الطريقة الاولى نقترح طريقة تشفير عدة صور في وقت واحد باستعمال أسلوب التشفير DRPE و تحويلات ROP . في الطريقة الثانية، نقوم بتطوير معالجة غير خطية جديدة لحل مشكلة التشفير الخطي في طريقة التشفير DRPE لتعزيز أمنها .اما في الطريقة الثالثة، نقترح طريقة جديدة لتشفير الفيديو باستعمال التحويلات الحقيقية ذات معاملات عشوائية و التشفير بالمجموعات. كل هذه الطرق مرت بعدة اختبارات لتقييم امنها ضد عدة هجمات معروفة , كما قمنا بمقارنتها مع الطرق الحالية من حيث الحساسية و مجال المفتاح السري .بالإضافة إلى ذلك، قمنا بالتحقق من مقاومتها ضد الضوضاء المضافة وأخطاء الإرسال في قناة الاتصال.

كلمات البحث تشفير الصور ؛ أسلوب التشفير DRPE ؛ التحويلات ذات معاملات عشوائية حرة ؛ التبادلية