ORIGINAL ARTICLE

NOUVEAU CHIFFREMENT VIDEO CHAOTIQUE DANS UN SYSTEME DE COMMUNICATION MOBILE

NEW CHAOTIC VIDEO ENCRYPTION IN A MOBILE COMMUNICATION SYSTEM

| Tianandrasana Romeo Rajaonarison *[1] | et | Paul Auguste Randriamitantsoa [1] |

[1.] Université d'Antananarivo | Ecole Doctorale en Sciences et Technique de l'Ingénierie et de l'Innovation (ED-STII) | Laboratoire de Recherche en Télécommunication, Automatique, Signal et Images (TASI) | B.P 1500 Antananarivo101 | Madagascar |

| Received May 03, 2022 | | Accepted May 11, 2022 | | Published May 16, 2022 | | ID Article | Romeo-Ref08-ajira080522 |

RESUME

Introduction : Le cryptage est un moyen très efficace pour assurer la sécurité et la confidentialité des données devant les menaces. **Objectif** : Dans cet article, nous avons proposé une méthode de chiffrement de vidéo basée sur le chaos. **Méthodes** : L'approche de chiffrement de vidéo basée sur le chaos est implémentée sous Matlab pour crypter des vidéos de format AVI. Le choix d'un algorithme basé sur le chaos nous a permis d'améliorer le rapport temps/robustesse en exploitant au mieux les séquences chaotiques. **Résultats** : Les résultats des tests et évaluations montrent l'efficacité de nos systèmes. Parmi ces résultats, nous avons obtenu des valeurs de PSNR allant jusqu'à 8.14, des valeurs de NPCR autour de 99.99%, des valeurs de l'entropie allant jusqu'à 7.99.
Mots-clés: Cryptage, chaos, attracteur, diffusion, confusion

ABSTRACT

Introduction: Encryption is a very effective way to ensure the security and privacy of data in the face of threats. **Objective**: In this paper, we have proposed a chaos based video encryption method. **Methods**: The chaos-based video encryption approach is implemented in Matlab to encrypt AVI format videos. The choice of a chaos-based algorithm allowed us to improve the time/robustness ratio by exploiting the chaotic sequences. **Results**: The results of the tests and evaluations show the efficiency of our systems. Among these results, we obtained PSNR values up to 8.14, NPCR values around 99.99%, entropy values up to 7.99.
Keywords: *Encryption, chaos, attractor, diffusion, confusion.*

1. INTRODUCTION

Suite à ces réalités, beaucoup de vidéo circulent sans protection dans le système de communication mobile via les réseaux ou encore dans le cloud. Pourtant, cela provoque beaucoup de risques comme : la diffusion des informations intimes dans le réseau par des pirates qui peut conduire vers des délits graves comme les suicides, ou l'accès des vidéos payantes par des personnes qui n'ont pas le droit ou encore la découverte par les concurrents des idées des entreprises qui font leurs réunions en visio-conférence. La cryptographie est un art déjà utilisé par les hommes depuis très longtemps pour dissimuler ou cacher des informations et des messages secrets. L'évolution de l'informatique nous permette d'utiliser cet art pour sécuriser tous nos informations numériques et de faire face à ces menaces. De nombreuses techniques de cryptographies sont alors inventées. On peut citer celles qui sont basées sur la théorie de nombres comme le chiffrement Rivest Shamir Adleman (RSA) ou celle qui sont basées sur des opérations logiques et mathématiques comme l'Advanced Encryption Standard (AES). Le principal problème de ces systèmes est lié au temps car on a besoin d'augmenter les nombres d'opération pour augmenter la robustesse et échapper aux menaces causées par accroissement de vitesse de calcul des ordinateurs. Cet article propose l'utilisation de cryptage vidéo basé sur le chaos pour remédier à ces problèmes et ces menaces. Le non linéarité, la sensibilité à la condition initiale, l'aspect aléatoire tout en étant déterministe d'une suite chaotique lui permet de devenir l'une de meilleure option pour la cryptographie. C'est pour cela que nous allons utiliser cette approche pour établir notre algorithme de cryptage de vidéo.

2. MATERIELS ET METHODES

2.1 Systèmes dynamiques chaotiques discrets:

Les générateurs de signaux chaotiques font parties des attracteurs étranges. Et on peut classer ces attracteurs en deux : les attracteurs qui fournissent des signaux chaotiques continus et les attracteurs qui fournissent des signaux chaotiques discrets.Cet article n'étudie que les attracteurs discret.

2.1.1 Suite logistique ou logistic map :

Cette fonction est donnée par l'équation **(1)** suivante [1,2]:

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

Xn est compris entre 0 et 1 et r est un nombre positif compris entre 1 et 4. Le comportement est chaotique à partir de r égal à 3.6. La **figure 1** illustre le diagramme de bifurcation (Xn en fonction de r).

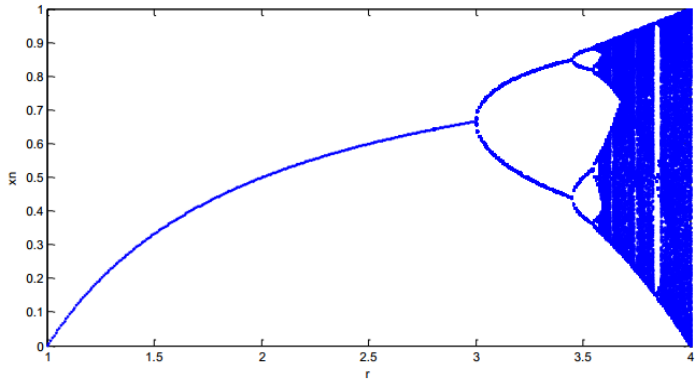


Figure 1: Diagramme de bifurcation.

2.1.2 Suites chaotiques linéaires par morceaux:

Il existe plusieurs récurrences chaotiques linéaires par morceaux, dont on cite les suivantes :

Tent Map

$$f(x) = \begin{cases} r\,x, & 0 \le x < 0.5 \\ r\,(1-x), & 0.5 \le x \le 1 \end{cases} \quad (2)$$

Skew tent map

$$f(x) = \begin{cases} x/r, & 0 \le x < r \\ (1-x)/(1-r), & r \le x < 1 \end{cases} \quad (3)$$

2.1.3 Récurrence de Hennon:

Elle constitue un système dynamique à temps discret introduit par l'astronome Michel Hénon en 1976. Il est présenté par le système d'**équation (4)** [3].

$$\begin{cases} X_{n+1} = Y_n + 1 - aX_n^2 \\ Y_{n+1} = bX_n \end{cases} \quad (4)$$

La **figure 2** représente l'attracteur de Hénon avec a=1.4 et b=0.3.

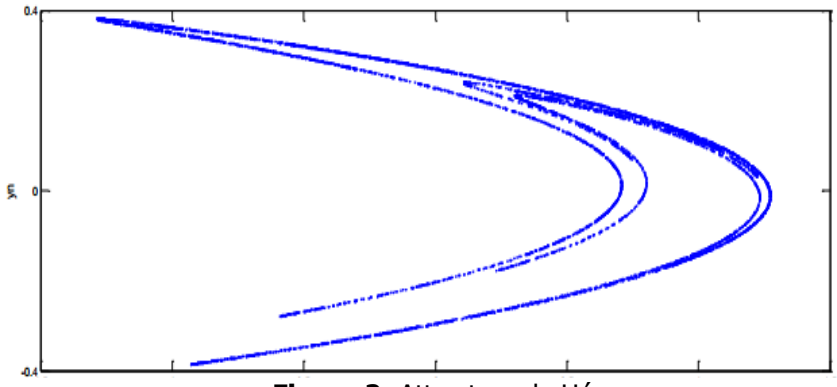


Figure 2: Attracteur de Hénon.

2.1.4 Gingerbreadman map:

Cet attracteur est présenté par le système d'**équations (5)** [3].

$$\begin{cases} X_{n+1} = 1 + |X_n| - Y_n \\ Y_{n+1} = X_n \end{cases} \quad (5)$$

La **figure 3** représente l'attracteur de Gingerbreadman.

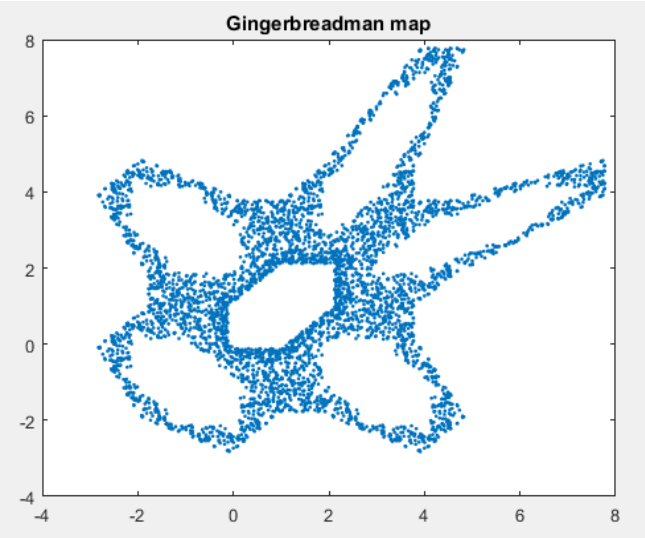


Figure 3: Attracteur de Gingerbreadman.

2.1.5 Tinkerbell map:

Il est donné par les **équations (6).**

$$\begin{cases} X_{n+1} = X_n^{\,2} - Y_n^{\,2} + aX_n + bY_n \\ Y_{n+1} = 2X_nY_n + cX_n + dY_n \end{cases} \quad (6)$$

La **figure 4** représente l'attracteur de Tinkerbell avec a=0.9, b=0.6013, c=2, d=0.5.

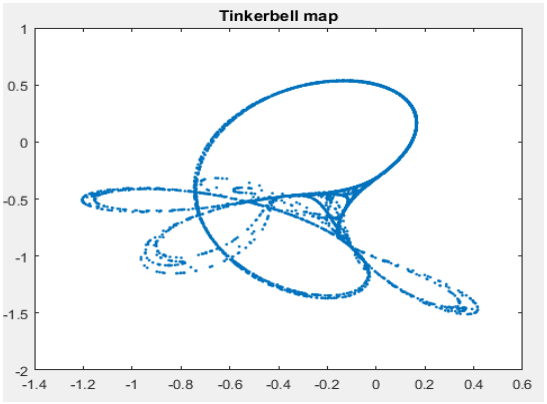


Figure 4: Attracteur de Tinkerbell.

2.1.6 Burger's map:

Ce système chaotique est présenté comme suit :

$$\begin{cases} X_{n+1} = (1-\nu)X_n - Y_n^{\,2} \\ Y_{n+1} = (1+\mu)Y_n + X_nY_n \end{cases} \quad (7)$$

La **figure 5** représente l'attracteur de Burger's.

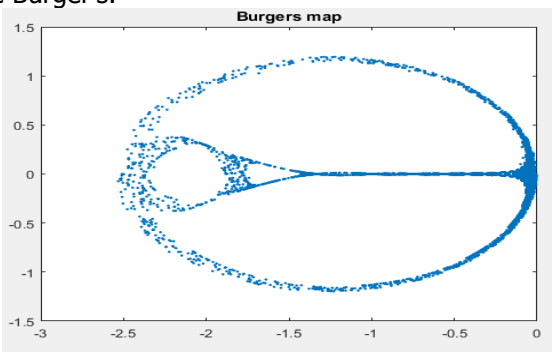


Figure 5: Attracteur de Burger.

2.1.7 Ricker's map:

L'attracteur de Ricker's est généré par le système d'**équations (8)** :

$$X_{n+1} = aX_n e^{X_n} \quad (8)$$

La **figure 6** représente l'attracteur de Ricker's avec a=20.

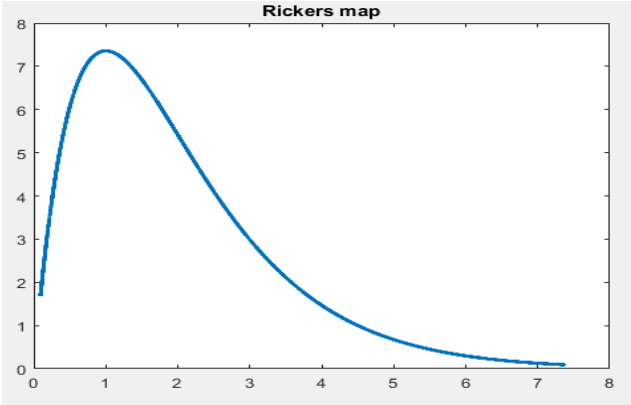


Figure 6: Attracteur de Ricker.

2.1.8 Cubic map:

L'attracteur de Cubic est généré par le système d'**équations (9)** :

$$\begin{cases} X_{n+1} = \alpha X_n + \beta X_n^{\,3}, 0 \le X_n \le 1/2 \\ X_{n+1} = \alpha(1 - X_n) + \beta(1 - X_n)^3, 1/2 \le X_n \le 1 \end{cases} \quad \textbf{(9)}$$

La **figure 7** représente l'attracteur de Cubic.

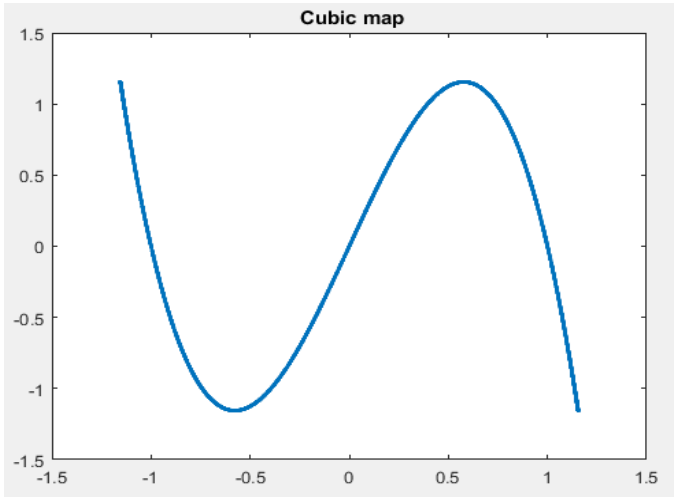


Figure 7: Attracteur de Cubic.

2.1.9 Sin map:

Ce système chaotique est présenté comme suit [1,2,4] :

$$X_{n+1} = \mu \sin(\pi X_n) \text{ avec } X \in [0,1], \mu > 0 \quad \textbf{(10)}$$

La **figure 8** représente l'attracteur de Sin.

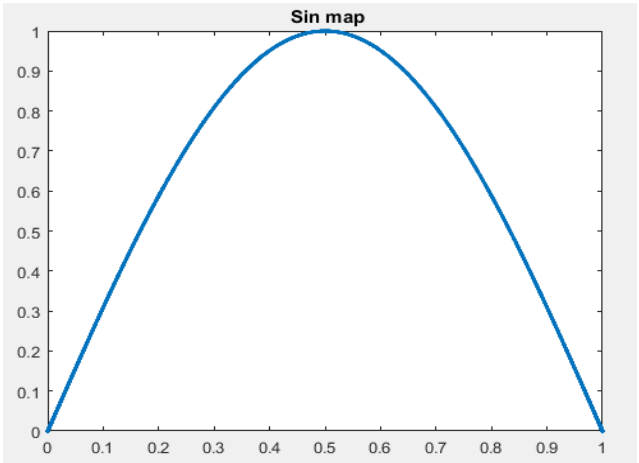


Figure 8: Attracteur de Sin.

2.2 Différence entre le chaos et la cryptographie classique:

Depuis les années 90, plusieurs chercheurs ont noté qu'il existe un rapport intéressant entre le chaos et la cryptographie. En effet, plusieurs propriétés des systèmes chaotiques présentent des correspondances similaires ou presque, avec des systèmes cryptographiques traditionnels. Le tableau suivant illustre parfaitement cette correspondance.

Tableau 1: Correspondance entre la théorie du Chaos et la cryptographie.

Théorie du chaos	Cryptographie
Systèmes chaotiques	Système pseudo-aléatoire
Transformation non linéaire	Transformation non linéaire
Nombre infini d'états	Nombre fini d'états
Nombre infini d'itérations	Nombre fini d'itérations
Etat initial	Plaintext
Etat final	Ciphertext
Condition(s) initiale(s) et/ou paramètre(s)	Clé(s)
Indépendance asymptotique des états initiaux et finaux	Confusion
Sensibilité aux conditions initiales et paramètres	Diffusion

2.3 Cryptages basées sur le chaos:

Généralement, un système de chiffrement par chaos est composé de deux parties : la partie confusion et la partie diffusion qui consistent respectivement la permutation des éléments de l'information originel et le remplacement ou substitution de ces éléments [2,5,6].

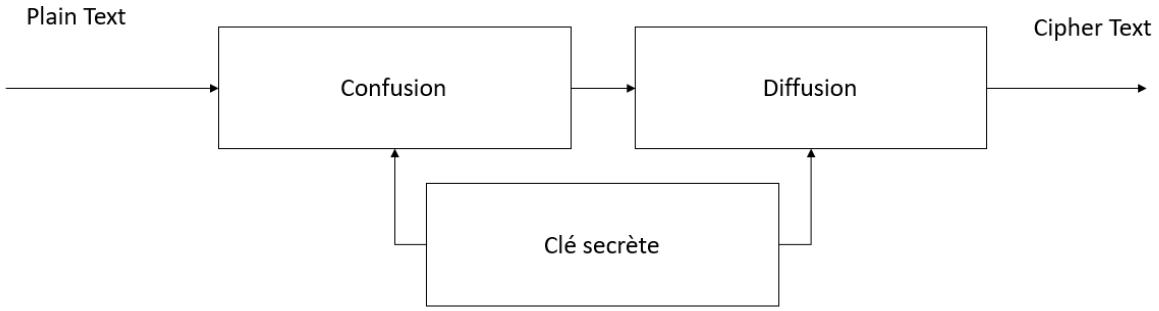


Figure 9: Principe générale.

2.3.1 Algorithme proposé par Chen et Mao

Chen a employé une version 3D de la carte Arnolad's Cat pour la substitution, la carte logistique pour la diffusion et le système chaotique de Chen comme un générateur des clefs. L'algorithme de chiffrement est illustré dans la **figure 10** suivante [2, 5,6].

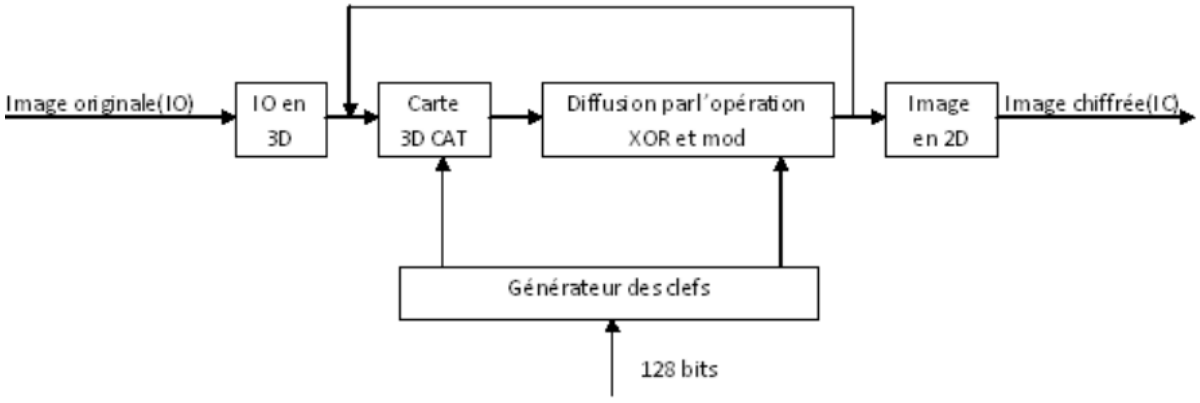


Figure 10: Principe de chiffrement de Chen.

Après la conversion de l'image originale en 3D, la carte 3D Arnolad's Cat definie comme suit :

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \\ Z_{n+1} \end{bmatrix} = A \begin{bmatrix} X_n \\ Y_n \\ Z_n \end{bmatrix} \mod N \qquad (11)$$

Où

$$A=\begin{bmatrix} 1+a_x a_z b_y & a_z & a_y+a_x a_z+a_x a_y a_z b_y \\ b_z+a_x b_y+a_x a_z b_y b_z & a_z b_z+1 & a_y a_z+a_x a_y a_z b_y b_z+a_x a_z b_z+a_x a_y b_y+a_x \\ a_x b_x b_y+b_y & b_x & a_x a_y b_x b_y+a_x b_x+a_y b_y+1 \end{bmatrix} \quad \textbf{(12)}$$

A est employée pour créer la confusion. Ensuite, l'**équation (13)** ci-après est utilisée pour créer la diffusion.

$$c(k)=\phi(k)\oplus\{[i(k)+\phi(k)]\bmod N\}\oplus c(k-1) \quad \textbf{(13)}$$

Où:

Φ(k) est généré en utilisant la carte logistique, i(k) représente la valeur du pixel en cours et c(k)est la nouvelle valeur du pixel en cours.

L'algorithme de Mao a la même idée que chen et al sauf qu'ils ont employé la carte 3D Baker à l'étape de substitution au lieu de la carte 3D Cat.

2.3.2 Algorithme proposé Lian :

Lian a prouvé qu'il existe quelques clefs faibles (problème de sécurité) dans les techniques de chiffrement utilisant les cartes chaotiques Baker et Cat, et que l'espace de clef de la carte chaotique standard est assez grand que ces deux dernières cartes. Ils ont utilisé la carte standard pour la substitution et la fonction suivante pour la diffusion :

$$C_i=v_i\oplus q[f(c_{i-1}),L) \quad \textbf{(14)}$$

Avec

$$q[f(c_{i-1}),L]=2^L\times f(c_{i-1}) \quad \textbf{(15)}$$

Où : Vi représente la valeur du pixel de l'image permutée, Ci désigne la valeur du pixel de l'image diffusée et la fonction f représente la carte logistique. Ils ont également recommandé au moins quatre rondes de la substitution et de la diffusion. L'algorithme de Lian est bien illustré par la **figure 11**.

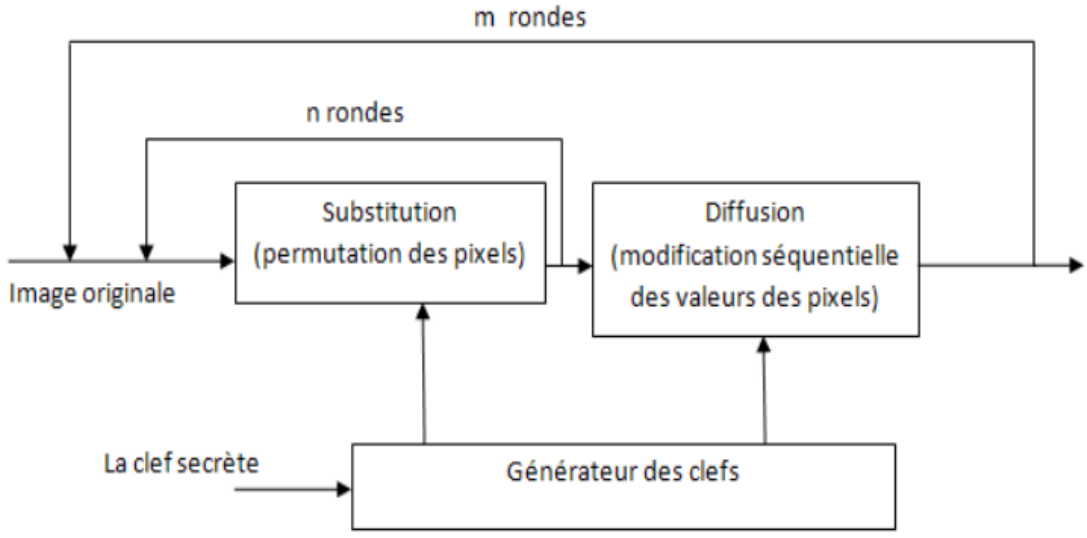


Figure 11: Principe de chiffrement de Lian.

2.3.3 Algorithme proposé par V. Patidar:

Ils ont proposé un nouvel algorithme de chiffrement en utilisant la carte chaotique standard et la carte logistique avec une clef secrète de 157 bits pour chiffrer des images couleurs. La condition initiale, le paramètre système de la carte standard et le nombre d'itération constituent ensemble la clef secrète. La première ronde de confusion est effectuée par l'intermédiaire des XORing keys calculé à partir de la clef secrète. Ensuite, dans les deux rondes de diffusion les propriétés des pixels horizontalement et verticalement adjacents sont mélangées respectivement. Dans la quatrième ronde une confusion robuste et efficace est réalisée à l'aide de la carte standard et logistique [2,5,6].

Cet algorithme est bien détaillé dans la **figure 12**
.

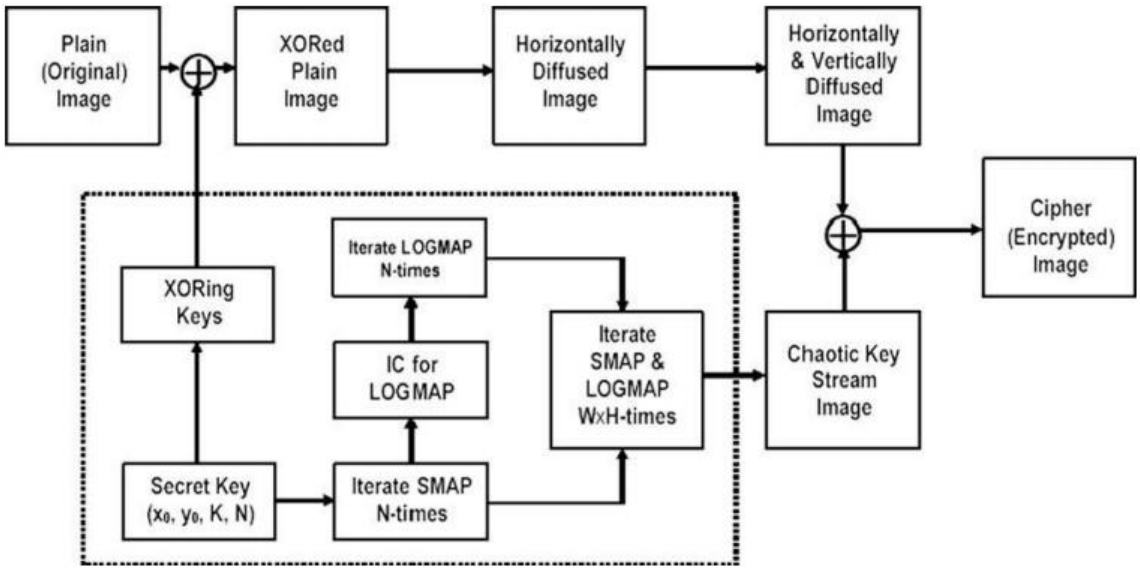


Figure 12: Principe de chiffrement de Patidar.

2.4 Fonctionnement du nouveau chiffrement à base chaotique :

2.4.1 Chiffrement :

Lecture : Cette étape consiste à lire les frames qui composent la vidéo pour la partie image, et les échantillons de son pour celle de l'audio. Dans cette étape on enregistre les informations concernant le nombre de frame par second et le nombre d'échantillon par second pour synchroniser la vidéo cryptée.

Cryptage : Cette étape renferme toutes les transformations effectuées sur la vidéo pour la rendre inintelligible.

Synchronisation : Cette étape consiste à réécrire le fichier vidéo selon ses paramètres initiales (frames par seconde, échantillon par seconde) mais avec des frames et échantillons crypté.

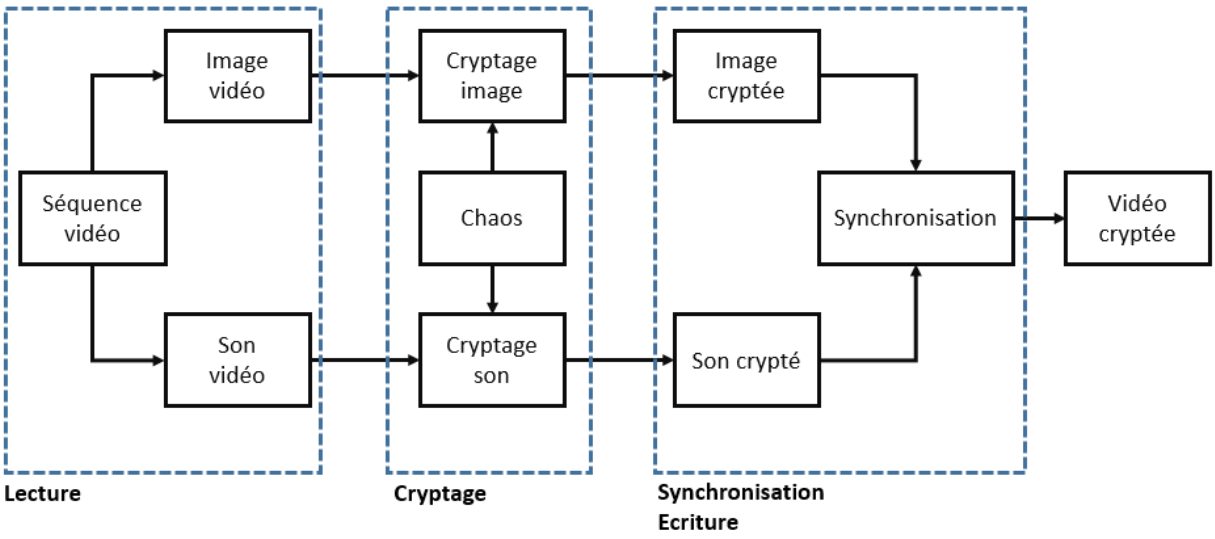


Figure 13: Schéma synoptique de cryptage de vidéo.

2.4.2 Déchiffrement :

L'opération de décryptage se fait comme l'opération de cryptage sauf qu'ici l'entrée est la vidéo cryptée.

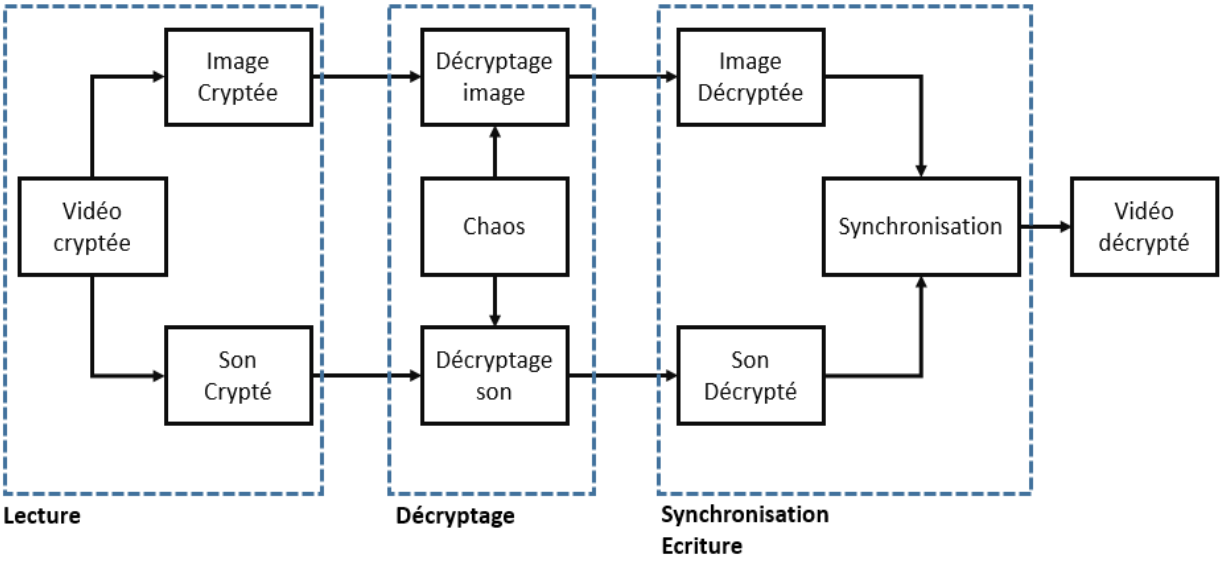


Figure 14: Schéma synoptique de décryptage de vidéo.

2.4.3 Principe de cryptage :

- **Génération de clé :**

Nous allons utiliser des attracteurs chaotiques pour générer les suites que nous utiliserons comme clés de chiffrements. Les éléments qu'on doit garder secrète sont donc : le ou les attracteur(s) utilisés, ces conditions initiales et les paramètres de ces équations. Après sa génération, ses clés seront préparées/transformées pour pouvoir les employées dans chaque partie de chiffrement. Dans la partie confusion on doit créer les couples (m_i, k_i) où m_i un élément du message et k_i un élément de la clé. Dans la partie diffusion on doit linéariser les éléments de la clé entre 0 et 255. La **figure 15** montre les étapes de la génération de clé.

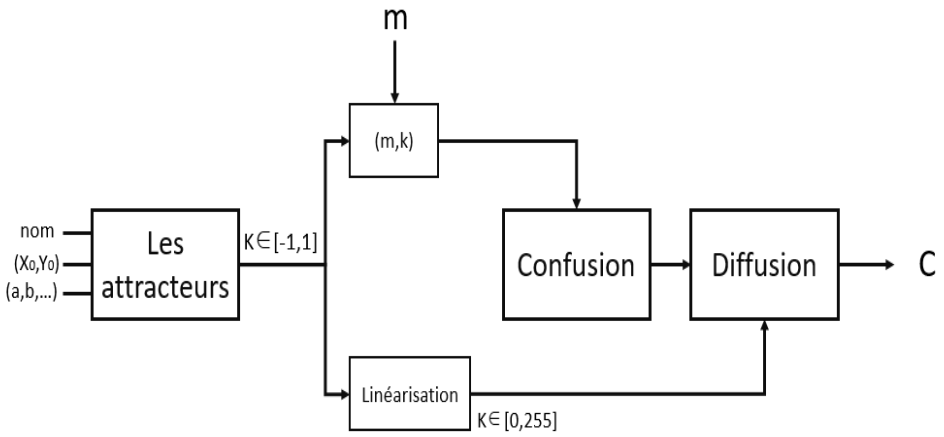


Figure 15: Génération de clé.

Avec les conditions initiales suivantes :

Attracteur : Tinkerbell
Conditions initiales : x0= 0, y0=0.5, a= 0.9, b= -0.6, c= 2, d= 0.5.

On obtient la suite chaotique suivante :

0; -0.5500; -0.4050; -1.0130; 0.1894; -0.8727;-0.2201; -0.2240; -0.0947;-0.0162
La linéarisation consiste projeter cette suite dans l'intervalle [0 ; 255].

Après la linéarisation, la suite chaotique devient :

36 ; 164; 77 ; 138; 0; 45; 36; 31; 181; 17
On utilise cette suite comme clé de chiffrement.

- **Confusion à partir du chaos:**

L'objectif de cette étape est de permuter l'image pour éliminer toutes les indices visuels qui peuvent révéler le contenu de l'image. Pour effectuer cette permutation nous allons suivre les étapes suivantes : former d'abord les couples (m,k) puis ranger les éléments de k dans l'ordre croissant sans séparer chaque couple.

$$(m_i, k_i) \Rightarrow (m_j, k_j) \quad \textbf{(16)}$$

Où, $i = \{1,2,...,N\}$,

$j = \{1,2,...,N\}$,

$k_j < k_{j+1}$

La **figure 16** montre 3 moyens de permuter une image : par ligne (a), par colonne(b), par colonne sur l'image entière(c).

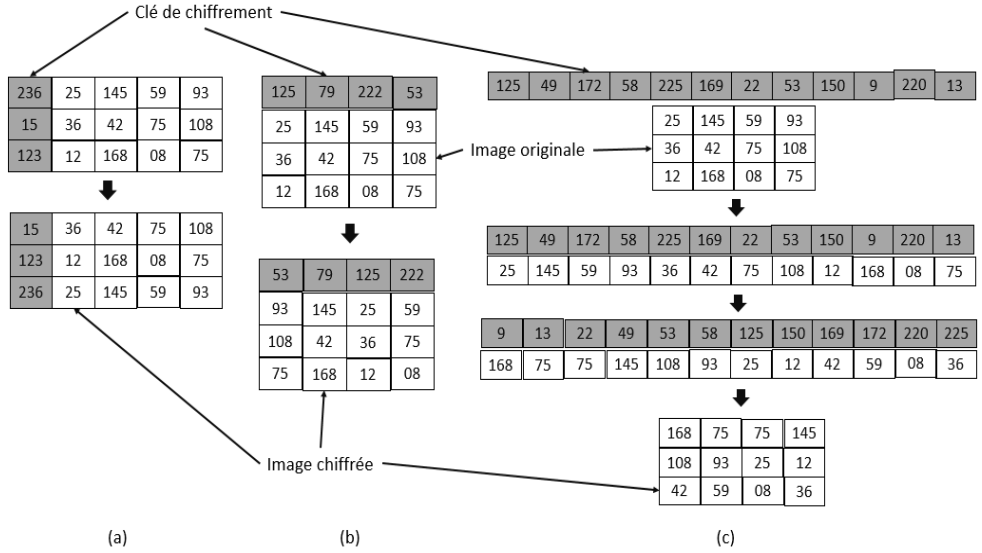


Figure 16: Principe de confusion par ordre croissante.

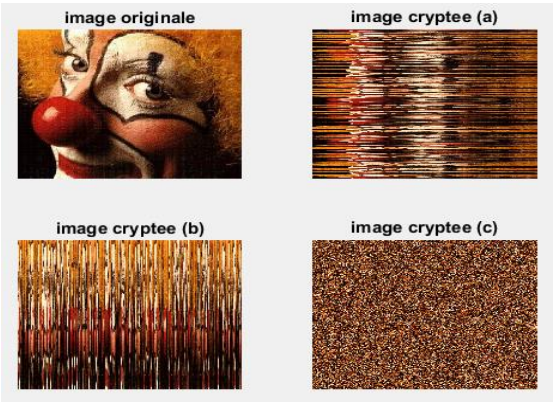


Figure 17: Exemples de confusion par le chaos.

Pour effectuer le déchiffrement, nous allons utiliser une clé intermédiaire $k_{\text{int}} = 1,2,...N$. Cette clé doit être permutée par la clé k utilisée pour le chiffrement. Et cette k_{int} permuté est utilisé pour le décryptage du message. La **figure 18** montre les étapes de déchiffrement de la « confusion 1-(a) ».

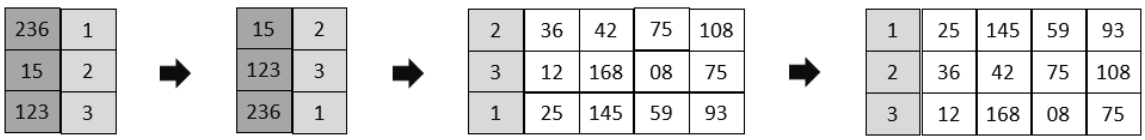


Figure 18: Principe de déchiffrement de la confusion par chaos.

- **Confusion à partir de l'image:**

L'objectif de cette étape de confusion est de crée une dépendance entre les pixels de l'image crypté pour éviter les attaques différentiels. On effectue donc une permutation des pixels voisins en fonction de la valeur d'un pixel courant suivant l'algorithme :

Si $((p(i,1,1) \oplus p(i,1,2) \oplus p(i,1,3)) \bmod 3) \neq 1$

Alors $p(i+1,:,1) \Leftrightarrow p(i+2,:,1)$

$p(i+1,:,2) \Leftrightarrow p(i+2,:,2)$

$p(i+1,:,3) \Leftrightarrow p(i+2,:,3)$

Figure 19: Exemple de confusion par l'image lui-même.

Le déchiffrement suit le même algorithme que le chiffrement mais dans le sens inverse cela veut dire que si le chiffrement se fait de haut en bas alors le déchiffrement de bas en haut.

- **Principe de diffusion:**

Pendant cette étape on va changer les valeurs de chaque pixel pour égaliser la probabilité d'apparition de chaque niveau de couleur. Pour une image de taille $l \times c \times k$; où l,c,k indique respectivement le nombre de ligne, le nombre de colonne et la couleur du composant; cette opération de diffusion est définie par l'**équation (17)**. La valeur de k sera 1 pour la couleur rouge, 2 pour la couleur verte, et 3 pour la couleur bleue. La table de vérité de l'opération OU-EXCLUSIF est représentée par le **Tableau 2.**

Tableau 2: Table de vérité de l'opération xor.

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

Exemples:
A=10 = 1010; B=12= 1100 donc A XOR B = 0110

$$P(i,j,k) = C(i,j) \oplus P(i,j,k) \quad \textbf{(17)}$$

Où : $i = 1,2,...l$

$j = 1,2,...c$

$k = 1,2,3$

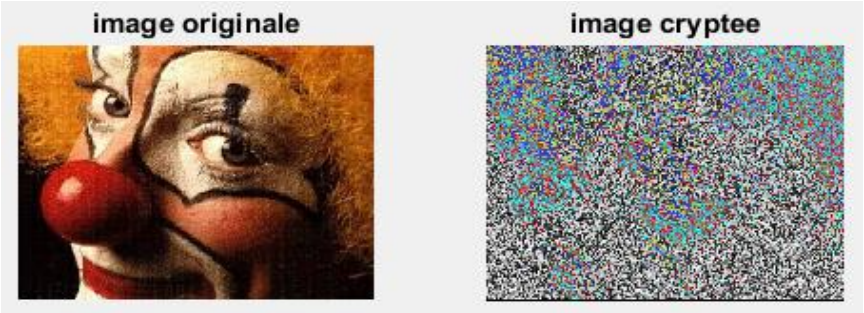


Figure 20: Exemple de diffusion.

2.5 Algorithme proposé :

Les schémas que nous allons proposer serons basées sur ces 3 types de chiffrement sauf que : la génération de clé, le principe de confusion ou de diffusion, l'ordre d'utilisation des étapes seront permuter pour avoir des meilleurs résultats. Après plusieurs tests réalisés, on propose les schémas suivants qui peuvent être utilisés selon nos besoins. Pour faciliter l'appellation : nous allons nommer la confusion à partir du chaos « confusion 1 » et celle à partir du contenu de l'image « confusion 2 ».

- **Confusion totale:**

Dans ce schéma nous avons privilégié le temps de chiffrement au détriment de la robustesse. Pour réaliser ce chiffrement nous allons utiliser un attracteur pour générer une clé de chiffrement. Cette clé sera utilisée pour effectuer la « confusion 1-(c)» sur le frame courant et sur les échantillons audio. Après, à la sortie de cette première confusion, les échantillons audio rejoignent la sortie tandis que le frame passe encore par la « confusion 2 ». La **figure 21** montre le fonctionnement de ce schéma. Ces étapes sont répétées pour tous les frames de la vidéo.

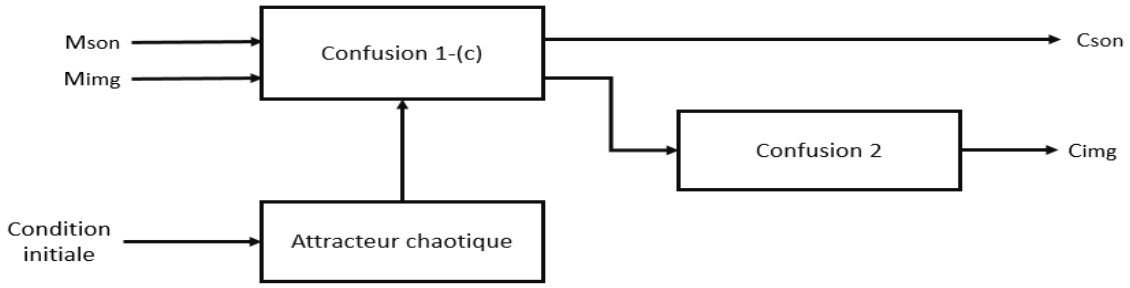


Figure 21: Fonctionnement de l'algorithme confusion totale.

- **Diffusion totale et confusion partielle:**

Ce schéma possède un rapport temps robustesse équilibré par rapport aux deux autres. Ce deuxième schéma utilise un attracteur pour générer les clés suivant la méthode décrite par la **figure 15**). Après, l'image subit l'étape de « diffusion » avant de passer par l'étape de « confusion 2 », et enfin par une étape de « confusion 1-(a) ». Les échantillons audio sont cryptés par la méthode de « confusion1-(c) ». La **figure 22** montre le fonctionnement de ce schéma.

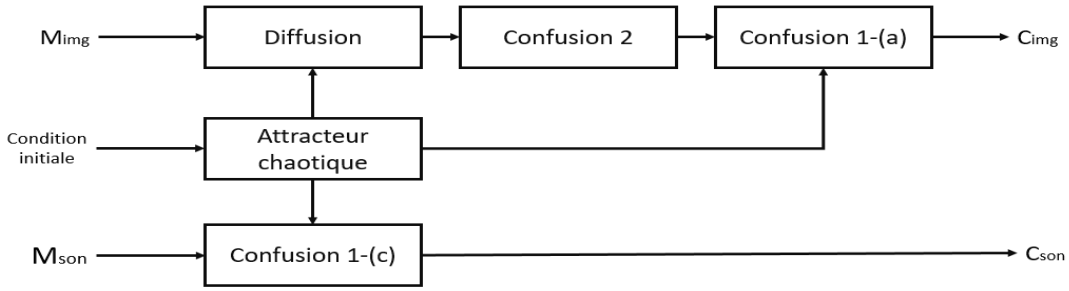


Figure 22: Fonctionnement de l'algorithme diffusion totale et confusion partielle.

- **Diffusion et confusion**:

Ce troisième schéma permet un chiffrement robuste mais qui est assez gourmand en temps. Dans ce schéma, nous allons utiliser la méthode de génération de clé de la **figure 15.** Après, on effectue successivement les étapes de « diffusion », « confusion 2 » et « confusion 1-(c) ». Les échantillons audio sont cryptés par la méthode de « confusion1-(c) ». La **figure 23** montre le principe de ce chiffrement.

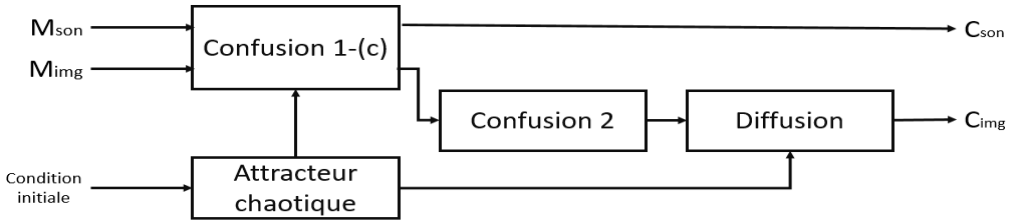


Figure 23: Fonctionnement de l'algorithme de diffusion et confusion totale.

3. RESULTATS

3.1 Mesure d'analyse statistique:

Dans cette section, nous allons présenter les résultats de nos méthodes de cryptage. Pour analyser les performances de chaque approche, nous allons utiliser un vidéo couleur de durée 2 seconde de format « avi » de dimension 640X480 pixels. Les conditions initiales utilisées pendant le test sont les suivantes :

Attracteurs : Tinkerbell map avec Conditions initiales : x0= 0, y0=0.5, a= 0.9, b= -0.6, c= 2, d= 0.5.

3.1.1 Analyse d'uniformité:

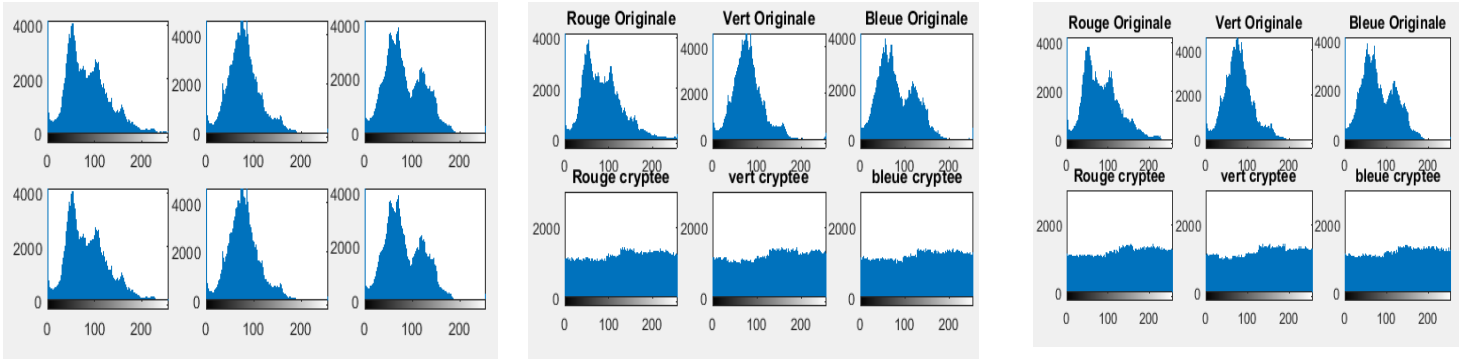
L'histogramme est une métrique couramment utilisée en tant que vérification qualitative de la distribution des données, dans le but d'évaluer la robustesse du cryptosystème contre les attaques statistiques. Pour un cryptosystème bien conçu, une telle métrique doit masquer toute information notable sur l'image simple ou la relation entre cette dernière et l'image de chiffrement. Un histogramme d'image est une représentation graphique présentant la distribution des valeurs de pixels, en traçant le nombre de pixels dans chaque valeur de niveau de gris. La représentation graphique de l'histogramme de l'image chiffrée est importante, mais pas suffisante pour garantir l'uniformité de la distribution des pixels chiffrés. À cette fin, le test du Chi-square est effectué pour indiquer la caractéristique d'uniformité de l'histogramme d'image de chiffrement donné. Ce test statistique est calculé par l'**équation (18)** [6,7]:

$$x^2_{\exp} = \sum_{i=1}^{N_v} \frac{(o_i - e_i)^2}{e_i} \quad \textbf{(18)}$$

Où N_v désigne le nombre total de niveaux ($N_v = 256$ pour une image à niveaux de gris), o_i sont les fréquences d'occurrence de chaque niveau de gris (0-255) et e_i représente la fréquence d'occurrence attendue de la distribution uniforme, calculé par l'**équation (19).**

$$e_i = \frac{M \times N \times P}{256} \quad \textbf{(19)}$$

Où : M est le nombre de lignes, N nombre de colonnes et P le nombre de plans, pour l'image en niveaux de gris $P = 1$.



Méthode 1 Méthode 2 Méthode 3

Figure 24: Fonctionnement de l'algorithme de diffusion et confusion totale.

3.1.1 Analyse de corrélation:

Une image avec son contenu visuel significatif se caractérise par sa forte corrélation et redondance entre les pixels voisins, que ce soit dans les directions horizontale, verticale ou diagonale. Un crypto-système bien conçu devrait dissimuler de telles relations entre pixels adjacents et présenter zéro corrélation. Pour évaluer l'immunité d'un crypto-système donné à ce type d'attaques, les N premières paires de pixels voisins sont sélectionnées de manière aléatoire à partir de l'image en clair et de sa version de chiffrement correspondante dans chaque direction. Ensuite, les corrélations de pixels adjacents dans une image donnée sont quantifiées en calculant le coefficient de corrélation r_{xy} de chaque paire au moyen des équations suivantes [8,9]:

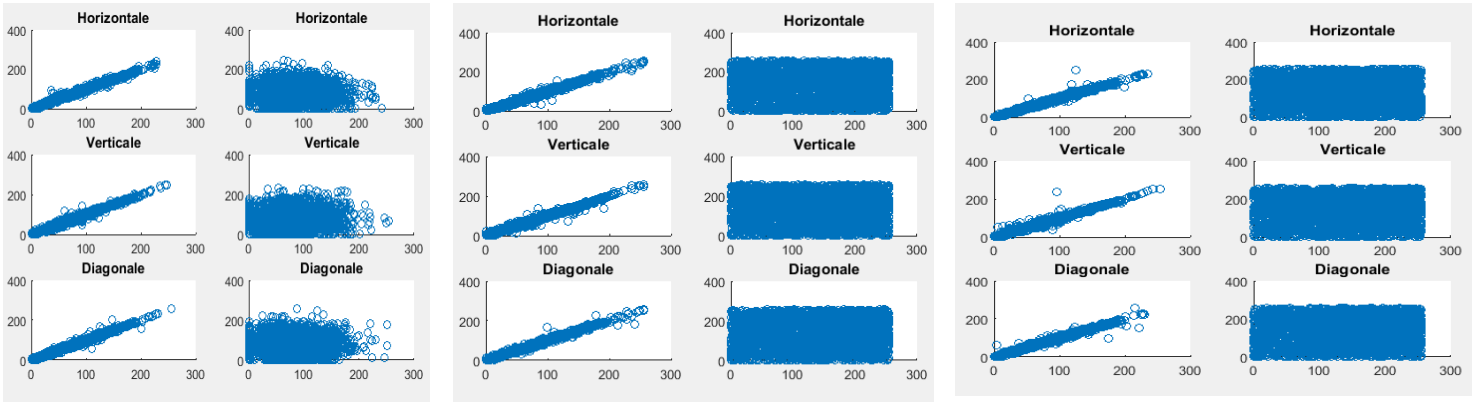
$$r_{xy} = \frac{\operatorname{cov}(x,y)}{\sqrt{D(x)} \times \sqrt{D(y)}} \quad \textbf{(20)}$$

$$\operatorname{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \quad \textbf{(21)}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \quad \textbf{(22)}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \quad \textbf{(23)}$$

Où x_i et y_i représentent les valeurs de niveau de gris de la $i^{ème}$ paire de pixels voisins choisis dans l'image, et N est le nombre entier d'échantillons.



Méthode 1 Méthode 2 Méthode 3

Figure 25: Fonctionnement de l'algorithme de diffusion et confusion totale.

3.2 Test de sensibilité:

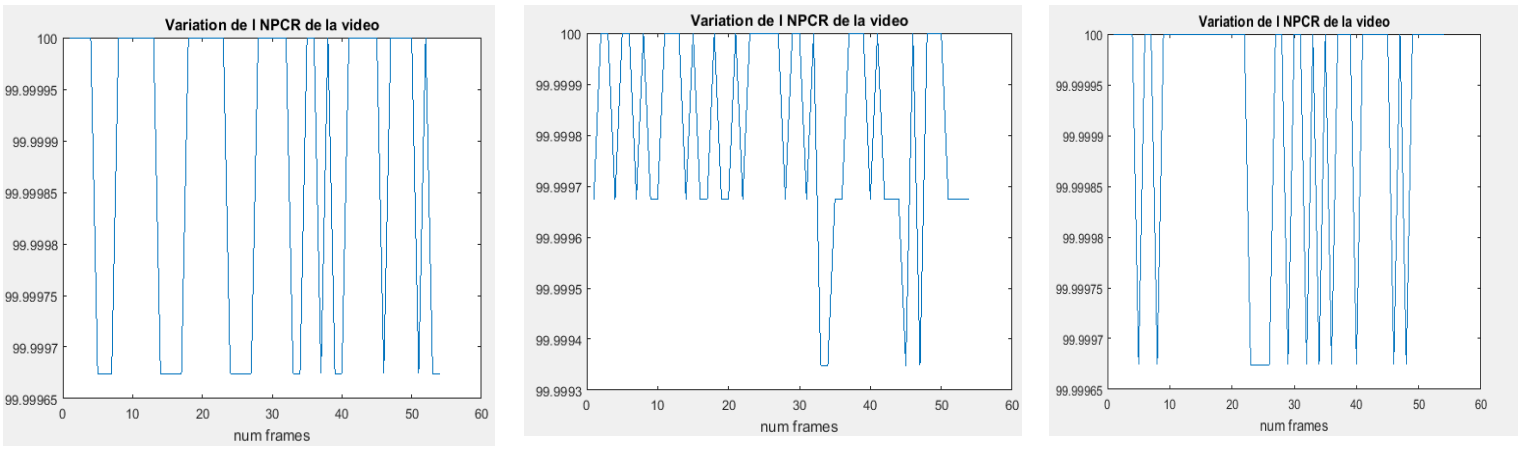
3.2.1 Robustesse contre les attaques différentielles:

Par souci de récupération de clé secrète, un attaquant pourrait tenter de distinguer toute information notable entre l'image normale et sa version de chiffrement, en observant l'influence d'un changement d'un pixel sur l'ensemble de la sortie du système de chiffrement. Un cryptosystème bien conçu est un système dans lequel une modification mineure de son image simple entraîne une transformation majeure de son image chiffrée et, par conséquent, ce type d'attaques est rendu nul. Pour réaliser l'expérience, la procédure suivante doit être adoptée:
1. L'image en clair P1 est cryptée pour avoir une image de chiffrement C1.
2. L'image simple P2 ici est obtenue en appliquant un changement mineur sur un pixel sélectionné de manière aléatoire, l'image ordinaire altérée P2 est chiffrée sous l'emploi de la même clé secrète pour produire l'image de chiffrement correspondante C2.
L'influence est mesurée quantitativement au moyen des deux métriques couramment utilisées est:
- le nombre de taux de changement de pixel (NPCR): il calcule le nombre de différences de pixels entre deux images chiffrées ci-après C1 et C2, au moyen des équations suivantes [7]:

$$NPCR = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} D(i,j)}{H \times L} \times 100 \quad (24)$$

Où N et M désignent respectivement la largeur et la hauteur de l'image, et D (i; j) est défini comme suit:

$$D(i,j) = \begin{cases} 0 & \text{Si } C_1(i,j) = C_2(i,j) \\ 1 & \text{Si } C_1(i,j) \neq C_2(i,j) \end{cases} \quad (25)$$

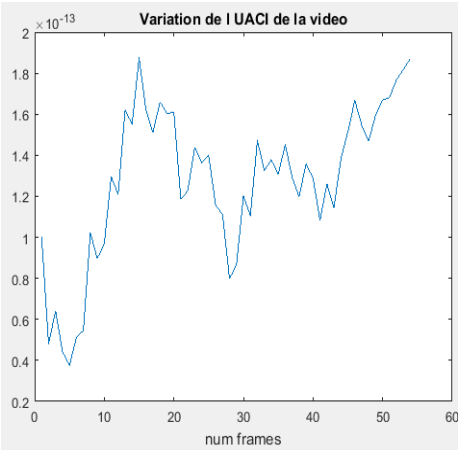


Méthode 1 Méthode 2 Méthode 3

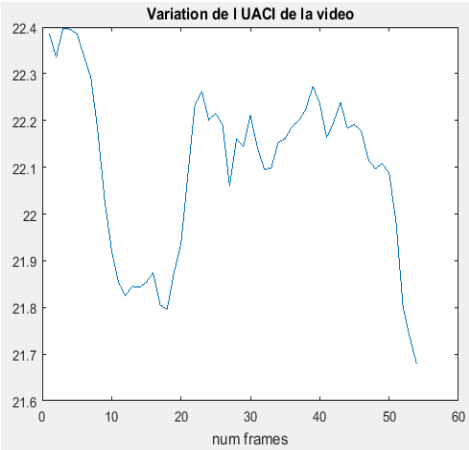
Figure 26 : Variation de la valeur de NPCR entre les images originales C1 et cryptées C2.

- Intensité de changement moyenne unifiée (UACI): elle calcule l'intensité moyenne des différences entre l'image originale C1 et l'image chiffrées C2, au moyen de l'**équation (26)** suivante [7] :

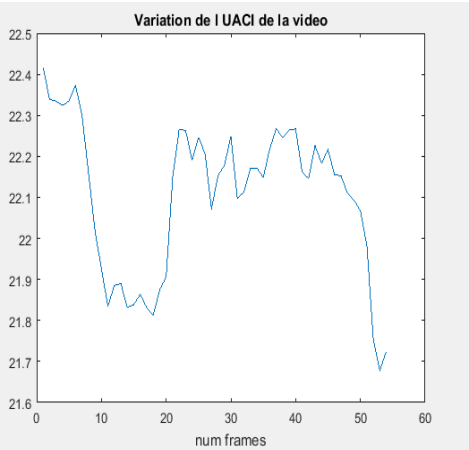
$$UACI = \sum_{i=1}^{N}\sum_{j=1}^{M}\frac{|C_1(i,j)-C_2(i,j)|}{H\times L\times 255}\times 100 \quad \textbf{(26)}$$



Méthode 1



Méthode 2



Méthode 3

Figure 27 : Variation de la valeur de NPCR entre les images originales C1 et cryptées.

3.2.2 Mesure d'analyse de performance

- **Estimations de la vitesse de calcul**

Chaque figure ou tableau doit avoir une légende décrivent les principaux résultats.
La prise de temps est également un facteur crucial à prendre en compte, en ce qui concerne le niveau de sécurité, dans la conception d'un crypto-système sécurisé. Par conséquent, une bonne combinaison entre les performances de calcul et un niveau de sécurité satisfaisant est plus qu'essentiel, en particulier pour les scénarios d'applications en temps réel. Les performances d'un crypto-système donné sont déterminées par l'évaluation de la vitesse d'exécution, cette dernière étant calculée au moyen des temps moyens de cryptage et décryptage [8].
Résultat de la méthode 1 : 26.52 sec
Résultat de la méthode 2:13.39 sec
Résultat de la méthode 3: 30.42 sec

- **Analyse du rapport de signal sur le bruit:**

Le rapport signal de crête sur le bruit, souvent abrégé en PSNR, est un terme technique désignant le rapport entre la puissance maximale possible d'un signal et la puissance de corruption du bruit affectant la fidélité de sa représentation. Du fait que de nombreux signaux ont une plage dynamique très large, le PSNR est généralement exprimé en échelle logarithmique de décibels. Le PSNR se définit le plus facilement via l'erreur quadratique moyenne (MSE). Étant donné une image monochrome $m\times n$ sans bruit I et son approximation bruyante K, MSE est définie comme suit [9]:

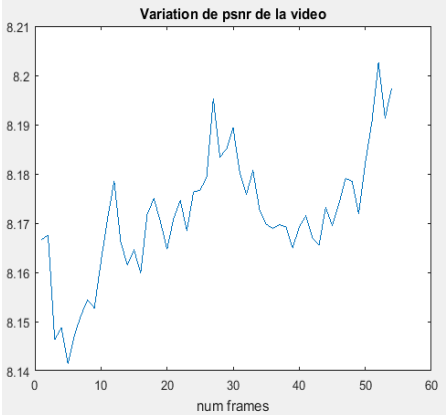
$$MSE = \frac{1}{m\times n}\sum_{i=1}^{m}\sum_{i=1}^{n}\left[I(i,j)-K(i,j)\right]^2 \quad \textbf{(27)}$$

Le PSNR (en dB) est défini comme suit:

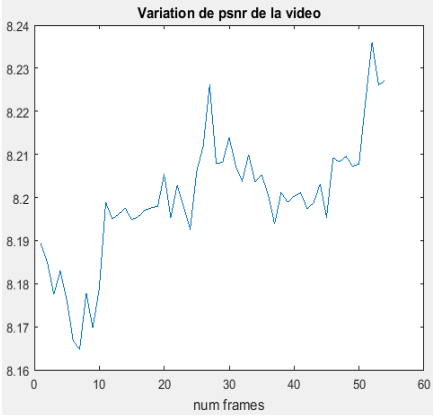
$$PSNR = 10\log\frac{255^2}{MSE} \quad \textbf{(28)}$$

Utilise cette forme simple de tableau ne pas utiliser la forme complexe:

Méthode 1



Méthode 2



Méthode 3

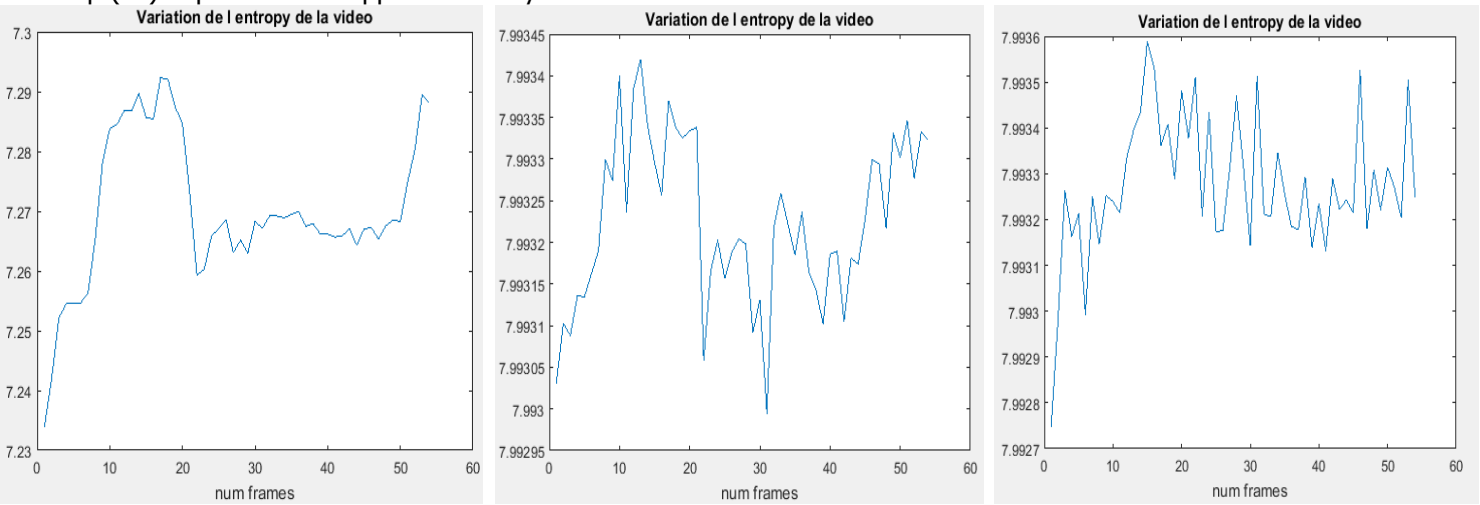
Figure 28 : Variation de la valeur de NPCR entre les images originales C1 et cryptées

- **Entropie:**

L'entropie H(m) peut être utilisée comme un indicateur de performance d'un crypto-système. Elle indique la quantité d'information contenue ou délivrée par un frame. Elle est liée à la distribution de probabilité d'apparition de chaque pixel et se calcule par l'**équation (29)**. La valeur de l'entropie pour un crypto-système idéal est 8.

$$H(m) = -\sum_{i=0}^{L-1} p(m_i)\log(p(m_i)) \quad \textbf{(29)}$$

Avec p (mi) la probabilité d'apparition du symbole mi



Méthode 1 Méthode 2 Méthode 3

Figure 29 : Variation de la valeur de l'entropie des images cryptées.

4. DISCUSSION

Les courbes de la **figure 24** représentent respectivement les histogrammes originales et cryptés des trois couleurs (Rouge, Vert, Bleu) des méthodes 1, 2, 3. L'axe des ordonnées indique le nombre d'occurrence de chaque pixel de l'axe des abscisses. Une image cryptée doit avoir un histogramme plat. L'histogramme de la **figure 25 méthode 1** ne change pas parce que la méthode ne fait que permuter les pixels sans changer leurs valeurs. Les méthodes 2 et 3 passent ce test.

Les courbes la **figure 25** représentent respectivement les comparaisons des courbes de corrélations des pixels voisines entre les images originales (à gauche) et les images cryptées (à droite). Les points de la figure de corrélation pour une image cryptée doivent être bien éparpillés. La corrélation des pixels adjacents de la méthode 1 n'est pas conforme. Par contre les deux autres méthodes passent ce test.

La valeur de référence de NPCR pour un cryptage réussi est de 99.6094 %. La **figure 26** montre que nos 3 méthodes passent ce test parce que nos valeurs sont toutes supérieurs à 99,9993 %. Cela veut dire que nos systèmes résistent bien aux attaques différentielles.

Les courbes de la **figure 27** montrent les variations de la valeur de UACI des 3 méthodes proposées. La valeur de référence pour ce test est de 33,4635%. La 1ère méthode a des résultats très loin de cette valeur. Par contre les deux autres méthodes ont des résultats plus proches.

Les courbes de la **figure 28** montrent les valeurs de PSNR. Le PSNR de la 1ère méthode est supérieur à ceux des autres. Cela veut dire que la 1ère méthode est moins performante que les autres car la valeur de référence est 0.
La valeur de référence de l'entropie est 8. Les 3 méthodes de la **figure 29** montrent que nos résultats ne sont pas loin de cette valeur. Cela montre l'efficacité de nos méthodes.

5. CONCLUSION

Le besoin de confidentialité, d'intégrité, d'authentification du système de communication mobile oblige les gens à courir vers le cryptage de ses données. Dans cet article nous avons créé une méthode de chiffrement basée sur le chaos pour assurer la protection de nos données vidéo et faire faces à plusieurs menaces liés à la diffusion ou accès non autorisé. Cette méthode réduit la complexité de calculer et de générer les clés de chiffrement par l'utilisation des attracteurs chaotiques. La propriété de diffusion est assuré par une opération ou exclusif entre la clé et les valeurs de pixels de

chaque frame. Tandis que la confusion se fait par une permutation des valeurs de pixels suivant l'ordre de grandeur de la clé de chiffrement. Après, nous avons implémenté notre méthode cryptage sur Matlab. A partir des résultats obtenus par la simulation, nous avons effectué les tests et les évaluations pour qualifier notre système.

6. REFERENCES

1. Eddine G.D.. Fonction logistique et standard chaotique pour le chiffrement des images satellitaires , Univérsité Mentouri de Constantine, 2010
2. Al Nahian, S., Hosen, Z. and Ahmed, P. An Elementary Study of Chaotic Behaviors in 1-D Maps. *Journal of Applied Mathematics and Physics*. 2019;7(5) :1149-1173. doi: 10.4236/jamp.2019.75077.
3. Laurent L. Cryptographie par chaos à l'aide des dynamiques non linéaires à retar, Université de Franche-Comté, 2002
4. Madani M., et Bentoutou Y. Cryptage d'images médicales à la base des cartes chaotique, University of Sidi-Bel-Abbes, Décembre 2015.
5. Nsreddine M. Cryptage basé sur l'attracteur de Clifford, Université Abou Bekr Belkaid, 2017.
6. Nassim A.S. Chaos based cryptography and image encryption, University of Applied Sciences Lucbeck Germamany, 2012.
7. Junqin Zhao, Weichuang Guo, and Ruisong Ye. A Chaos-based Image Encryption Scheme Using Permutation-Substitution Architecture. *International Journal of Computer Trends and Technology (IJCTT). Sep 2014.* 15(4):174-185. doi: 10.14445/22312803/IJCTT-V15P137
8. S. Ahadpour, Y. Sandra. A chaos-based image encryption scheme using chaotic coupled map lattices, University of Mohaghegh Ardabili Iran, 2012.
9. Sathishkumar G. A., Bagan K.B., and Sriraam N. Image encryption based on diffusion and multiple chaotic maps. *International Journal of Network Security & Its Applications (IJNSA)*. March 2011; 3(2):181-194. doi : 10.5121/ijnsa.2011.3214 181

Cite this article: Tianandrasana Romeo Rajaonarison and Paul Auguste Randriamitantsoa. NOUVEAU CHIFFREMENT VIDEO CHAOTIQUE DANS UN SYSTEME DE COMMUNICATION MOBILE**.** *Am. J. innov. res. appl. sci. 2022; 14(5): 188-*203.