Codage hybride cryptage-marquage-compression pour la sécurisation de l'information médicale

William Puech, Gouenou Coatrieux

Chapitre 10

Codage hybride cryptage-marquage-compression pour la sécurisation de l'information médicale

10.1. Introduction

De nos jours, de plus en plus d'images numériques sont transférées sur les réseaux informatiques. Les travaux présentés dans ce chapitre montrent comment des algorithmes de cryptage et de marquage permettent la sécurisation des images médicales. Pour ce faire, les images peuvent être cryptées au niveau des codages source afin de faire remonter cette fonctionnalité au niveau des couches hautes (applications). De cette manière, les fonctionnalités cryptage et marquage d'images sont insérées au niveau d'un logiciel. La protection est alors assurée pendant la transmission des images médicales mais aussi pour l'archivage de ces données numériques. L'enjeu consiste ensuite à faire résister ces chiffrements à des traitements avals comme la compression. En effet, la quantité d'information (entropie) à transmettre augmente fortement entre l'image originale et l'image cryptée. Dans le cas particulier de certains types d'images médicales, de grandes zones homogènes apparaissent. Ces zones perturbent l'efficacité des algorithmes de chiffrement. Paradoxalement ces zones homogènes, inutiles pour le diagnostic, peuvent être utilisées en toute sécurité pour le marquage des images médicales.

Aujourd'hui lorsqu'un médecin reçoit un patient il a souvent besoin de l'avis d'un spécialiste avant de prononcer son diagnostic. Une solution possible est de transmettre par liaison informatique, les images du patient avec le compte-rendu du

---

Chapitre rédigé par William PUECH et Gouenou COATRIEUX.

spécialiste. Cependant les réseaux informatiques sont complexes et les écoutes illégales possibles. Il se pose donc un réel problème quant à la sécurité lors de la transmission de données. Pour des raisons éthiques, le transfert des images médicales ne peut se faire avec un tel risque et doit donc se protéger. La protection la plus adaptée pour ce type de communication réside dans la cryptographie. Beaucoup de techniques de cryptage de texte ont été développées. Depuis l'antiquité, les hommes ont toujours essayé de coder des messages secrets pour se prévenir des oreilles malveillantes. Dans les premières esquisses de cette science du secret, la sécurité résidait dans la confidentialité de l'algorithme qui permettait le chiffrement et le déchiffrement. C'est au fil du temps qu'est apparue progressivement la notion de clef. Aujourd'hui, les systèmes de cryptage reposent sur des algorithmes mis à disposition de tous et c'est la clef, code secret particulier, qui est confidentielle et qui permet de crypter ou de décrypter le message [KER 83].

Dans la section 10.2. nous mettrons en évidence la nécessité de sécuriser les images et les données médicales. Puis, dans la section 10.3. nous présenterons les algorithmes standard de cryptage et montrerons, dans la section 10.4. comment les adapter aux images médicales. Enfin, nous exposerons section 10.5. comment il est possible de dissimuler des données dans des images tout en gardant une très haute qualité des images.

10.2. Protection des images et des données médicales

L'évolution des techniques de traitement, de partage et de communication des images médicales et plus généralement de l'information médicale, s'accompagne d'une évolution des risques pour l'information alors sous forme numérique. L'information médicale de manière générale est composée principalement de résultats d'analyses, d'examens cliniques, d'examens para cliniques et de renseignements individuels [DUS 97]. Les possibilités d'accès distants et de communication de l'information sont améliorées mais les possibilités de fuites, de détournement et de modification de l'information sont aussi plus importantes, voire même facilitées par la mise à disposition d'outils de surveillance des réseaux et d'outils d'édition avancée de l'information comme les logiciels d'imagerie.

Cependant, ce sont les conséquences liées à la survenance de ces risques qui introduisent le besoin de protection de l'information médicale. Ces conséquences, perceptibles, concernent un individu et en particulier sa santé partie intégrante de sa vie privée. C'est pourquoi de nombreux pays ont traduit ce besoin de manière déontologique et aussi législative en donnant des droits au patient et par conséquent en imposant aux professionnels et aux établissements de santé d'assurer la protection des données qu'ils ont en leur possession.

10.2.1. *Législation et droits du patient*

Le législateur et le code de déontologie médical suivent l'évolution des techniques et au travers d'un nombre important de textes de lois, reconnaissent des droits au patient. Le premier, le plus connu, s'inscrit précisément dans le cadre de la relation patient/médecin, il s'agit du secret médical (article 4 du code de déontologie). La garantie de la confidentialité des informations que le patient échange ou a pu échanger avec les divers interlocuteurs du système de soins, permet d'instaurer une relation de confiance, relation qui permet aussi au praticien de mieux apprécier la situation du patient.

L'informatisation du système de santé et les possibilités offertes aussi bien en termes d'automatisation des traitements que de diffusion de l'information, a vu l'arsenal législatif s'élargir et d'autres lois sont à considérer par les praticiens. En particulier la loi n°78-17 du 6 janvier 1978 « loi relative à l'informatique, aux fichiers et aux libertés » (plus connue sous le nom « informatique et liberté ») complétée par celle du 1er juillet 1994 « loi relative au traitement des données nominatives ayant pour fin la recherche dans le domaine de la santé ». La CNIL (Commission Nationale de l'Informatique et des Libertés) est chargée de surveiller le bon respect de ces lois (articles 6 à 13). Ces lois, outre la collecte d'information, confèrent à tout citoyen, et en conséquence à tout patient, le droit d'exercer un contrôle sur l'exploitation des informations qui le concernent [DUC 96]. Notamment, un droit à la sécurité est donné au patient, et pour toute information nominative l'article 29 impose au responsable du fichier, donc au professionnel de santé, qui ordonne ou effectue un traitement, de s'engager à prendre « toutes précautions utiles afin de préserver la sécurité des informations et notamment d'empêcher qu'elles ne soient déformées, endommagées ou communiquées à des tiers non autorisés ». Si ce droit n'est pas respecté, des poursuites pénales peuvent être engagées (article 226-17 du code pénal). Du point de vue pratique et technique, des groupes de travail comme celui mis en place par la Comité Européen de Normalisation TC/251 Informatique médicale (*Working Group III*), montrent que pour satisfaire à cette obligation de sécurité, il revient de satisfaire à la fois [ALL 94] :

– la « confidentialité » des données en limitant l'accès de l'information aux seuls ayant droits (le patient et les professionnels de santé en charge de sa santé considérant l'aspect collégial de la pratique médicale ainsi que les dérogations prévues par la loi) ;

– l'« intégrité » de l'information, en apportant les preuves que l'information n'a pas été modifiée autrement que par une personne habilitée et dans les conditions prévues à cet effet ;

– la « disponibilité » dont l'objet est de garantir l'accès à l'information dans les conditions de fonctionnement normalement prévues.

10.2.2. *Un arsenal de mesures de protection*

Quelque soit la nature de l'information, nous pouvons distinguer trois types de mesures pour la protection des données stockées, traitées et communiquées par un système d'information : la législation en vigueur, les politiques de sécurité et les mécanismes de protection. Ces mesures sont à considérer conjointement pour satisfaire aux exigences DIC (Disponibilité, Intégrité, Confidentialité) des données. Néanmoins, ces mesures sont modulées entre elles suivant le système d'information cible et son contexte. Nous pouvons distinguer le système placé chez le praticien libéral du système d'un établissement de santé couplé ou non avec des applications communicantes telles que les applications de télémédecine.

Le premier type de mesures de nature législative a pour objet de décourager les personnes qui volontairement ou par accident pourraient attenter à la confidentialité, l'intégrité et la disponibilité des données et des systèmes d'information (Loi Godfrain n°88-19 du 05/01/1988). Cependant l'efficacité de telles mesures n'est envisageable que s'il est possible de détecter l'intrusion d'un tiers dans un système d'information. La politique de sécurité d'un établissement de santé a pour objet de fixer la stratégie de mise en œuvre et de maintien d'un niveau de sécurité optimal. Cette politique décide entre autres des divers protocoles d'usage de l'information et des systèmes, en considérant : les risques pour l'intégrité, la disponibilité et la confidentialité des données ; les rôles particuliers des divers acteurs du dispositif hospitalier, etc. Il revient en particulier à la politique de sécurité de positionner et de paramétrer la panoplie d'outils ou mécanismes de protection. Ces mécanismes physiques ou logiques sont nombreux et de mise en œuvre plus ou moins complexe. Les premiers ont trait à la protection physique des matériels (accès sécurisé aux salles, mesures contre les dégâts naturels, anti-vol, etc.) et les seconds sont intégrés au niveau des systèmes d'information. Les outils que nous étudierons dans ce chapitre font partie de cette seconde classe, il s'agit d'outils cryptographiques (chiffrement et signature numérique) et de marquage des images. Parmi les autres mécanismes de protection logique, nous pouvons ajouter [COA 03] :

– le contrôle d'accès qui couple une politique précisant les ayant droits à l'information et à l'accès aux fonctionnalités d'un système, avec des solutions techniques d'authentification des utilisateurs tels que les systèmes de carte à puce ou de vérification biométrique ;

– les pare-feu (*Firewall*) dont la mission première est de contrôler les accès au système en entrée et en sortie dès que le système est connecté à un réseau ;

– les antivirus ;

– l'audit, qui permet de garder une trace des accès à l'information par des utilisateurs ou des programmes informatiques.

Il est important de souligner que ces mécanismes sont complémentaires, c'est pourquoi ils sont exploités en parallèle. Par ailleurs, certains de ces mécanismes comme le contrôle d'accès, s'appuient sur des techniques cryptographiques. Le marquage des images est d'usage plus récent et vient s'ajouter à cette gamme d'outils. Avant de discuter de ces techniques, revenons aux images médicales. Ces images sont le plus souvent produites, stockées et communiquées en tenant compte du standard DICOM[1] décrit chapitre 4. Ce standard est plus qu'un simple format de stockage et intègre des « profiles » ou procédures bien spécifiques dont l'objectif est de garantir pour les données images les exigences DI dans le stockage et les échanges entre systèmes compatibles DICOM. Ces profiles s'appuient sur des mécanismes de type cryptographiques.

10.3. Généralités sur les algorithmes de cryptage

10.3.1. *Classification des algorithmes de cryptage*

Il existe quatre grands objectifs pour le cryptage des données numériques :

– « la confidentialité » ou masquage des données, caractéristique la plus utilisée, vise à rendre le cryptogramme inintelligible pour celui qui n'est pas en possession de la clef ;

– « l'authentification » permet à l'émetteur de signer son message, ainsi, le récepteur n'aura pas de doute sur l'identité du premier ;

– « l'intégrité » quant à elle va assurer au récepteur que le contenu du message n'a pas pu être malencontreusement falsifié depuis son écriture ;

– « la non répudiation » est la garantie qu'aucun des deux individus ayant effectué une transaction ne pourra nier avoir reçu ou envoyé les messages ;

La caractéristique majeure en imagerie médicale est bien sûr la première, à savoir la confidentialité. Mais la caractéristique intégrité décrite section 10.2., ainsi que les deux autres sont aussi importantes pour la protection des images médicales.

Les algorithmes de cryptage peuvent être séparés en fonction de plusieurs caractéristiques : les systèmes à clef secrète (systèmes symétriques) illustrés figure 10.1. et ceux à clefs publique/privée (systèmes asymétriques) illustrés figure 10.2. [DIF 76] [STI 96]. Les systèmes à clef secrète sont ceux qui permettent de crypter et de décrypter avec la même clef. Il va de soi que l'émetteur et le récepteur doivent s'être auparavant partagé le secret de la clef par un moyen de communication sécurisé. Les systèmes à clef publique ou asymétriques permettent de pallier à cette incommodité en utilisant une clef pour crypter, et une autre clef pour décrypter.

1. www.dicom.org.

Chaque individu détiendra un couple de clefs, dont une sera confidentielle (la clef privée) et l'autre connue de tous (la clef publique). Pour écrire à *B*, il suffit de chiffrer le message avec la clef publique de *B* que l'on connaît. A la réception, seul *B* pourra déchiffrer avec sa clef privée. Dans cette section nous présentons plusieurs systèmes de cryptage de données ; systèmes par bloc à clef secrète (DES et AES) ; par bloc à clef publique (RSA) ; et système de chiffrement par flot.

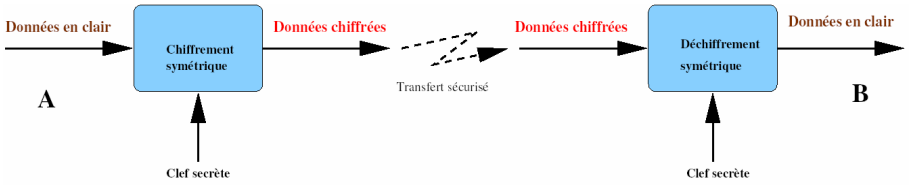

Figure 10.1. *Principe du chiffrement symétrique*

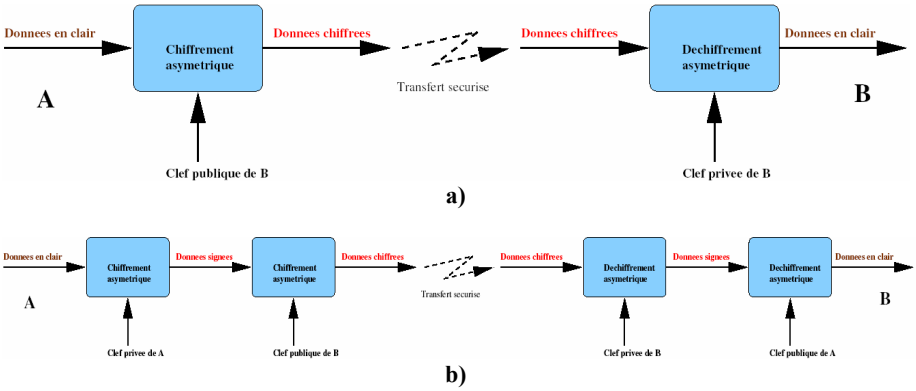

Figure 10.2. a) *Principe du chiffrement asymétrique.* **b)** *Double chiffrement asymétrique garantissant confidentialité et authenticité*

10.3.2. *L'algorithme de cryptage DES*

L'algorithme de chiffrement DES (*Data Encryption Standard*) fait partie des systèmes standard de chiffrement par bloc (figure 10.1.). Sa sécurité réside uniquement dans le secret de la clef, l'algorithme est public. C'est en 1974 que l'algorithme DES deviendra le premier standard de la cryptographie moderne [SCH 97]. L'algorithme DES repose sur 16 rondes (ensemble d'étapes répétées 16 fois) au cours desquelles un bloc de données de 64 bits va se mélanger avec la clef *K*, qui est codée sur 64 bits également. A chacune de ces rondes, une sous-clef k_i est calculée à partir de la clef de départ *K* (cette sous-clef servira à mélanger les bits du

bloc). Une fois que les 16 sous-clefs sont générées à partir de la clef secrète, il est possible de chiffrer (ou déchiffrer) un bloc de données de 64 bits. Le processus commence par une permutation initiale (PI) qui modifie l'ordre des bits du bloc de départ, avant de scinder le résultat en deux blocs de 32 bits, L_0 et R_0. Une fois les 16 rondes passées, avant de retourner le résultat, une permutation finale (PF) est appliquée au bloc. Cette permutation n'est autre que l'opération inverse de la PI. Pour le décryptage, le procédé est le même, mis à part que les sous-clefs sont utilisées dans l'ordre inverse.

Aujourd'hui, même si l'algorithme est toujours aussi robuste, il souffre quelque peu du fait de la longueur de sa clef limitée à 64 bits. En effet, la performance actuelle des machines en terme de rapidité de calcul rend le DES cassable. L'attaque dite brutale qui consiste à essayer toutes les 2^{64} clefs potentielles, est désormais abordable par de gros calculateurs. Une solution a été apportée pour augmenter la sécurité : elle s'appelle le triple-DES. Le triple-DES consiste à crypter le bloc d'entrée 3 fois avec des clefs différentes K_1, K_2 et K_3. Il existe plusieurs variantes, mais en général les première et troisième opérations sont des opérations de cryptage, tandis que la seconde est une opération de décryptage. Souvent, on choisit $K_1 = K_3$, ce qui permet à la clef totale de ne pas dépasser 128 bits.

10.3.3. *L'algorithme de cryptage AES*

L'algorithme de chiffrement AES (*Advanced Encryption Standard*) est le système standard de chiffrement par bloc et a pour objectif de remplacer le DES qui devient vulnérable. Le nombre de rondes de l'algorithme AES dépend de la taille de la clef et de la taille des blocs de données. Par exemple, le nombre de rondes est 9 si les blocs et la clef sont de longueur 128 bits. Pour crypter un bloc de données avec AES (figure 10.3.) il faut d'abord effectuer l'étape nommée *AddRoundKey* qui consiste à appliquer un « OU exclusif » (XOR) entre une sous-clef et le bloc. Après, nous entrons dans l'opération d'une ronde. Chaque opération régulière de ronde implique quatre étapes. La première est l'étape nommée *SubByte*, où chaque octet du bloc est remplacé par une autre valeur issue d'une *S-box*. La seconde étape est l'étape nommée *ShiftRow* où les lignes sont décalées cycliquement avec différents *offsets*. Dans la troisième étape, nommée *MixColumn*, chaque colonne est traitée comme un polynôme, multipliée sur GF(2^8) (*Galois Field*) par une matrice. La dernière étape d'une ronde est à nouveau l'étape nommée *AddRoundKey*, qui est un simple « OU exclusif » entre la donnée actuelle et la sous-clef de la ronde courante. L'algorithme AES effectue une routine supplémentaire finale qui est composée des étapes *SubByte, ShiftRow* et *AddRoundKey* avant de produire le chiffrement final. Le processus sur les données en clair est indépendant de celui appliqué sur la clef secrète, et cette dernière est appelée *KeySchedule*. Celle-ci est formée de deux composantes : la *KeyExpansion* et la *RoundKeySelection* [DAE 02] [AES 01].

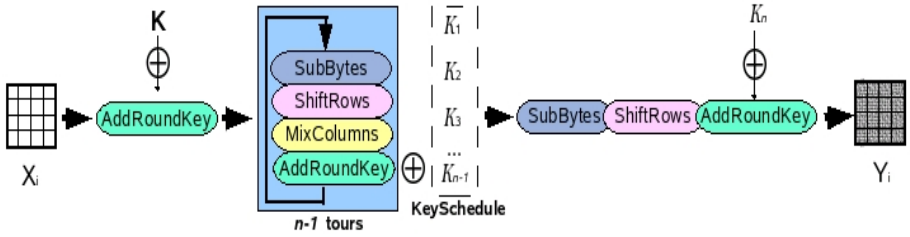

Figure 10.3. *Le schéma général d'AES*

L'algorithme AES peut supporter les modes de chiffrement suivants : ECB, CBC, OFB, CFB et CTR. Le mode ECB (*Electronic CodeBook*) est le mode de l'algorithme standard AES comme décrit dans la documentation 197 du standard FIPS (*Federal Information Processing Standards*). A partir d'une séquence binaire X_1, X_2,..., X_n de blocs en clair, chaque X_i est chiffré avec la même clef secrète k afin de produire les blocs chiffrés Y_1, Y_2,.., Y_n. Le mode CBC (*Cipher Block Chaining*) rajoute au chiffrement par bloc un mécanisme de retour. Chaque bloc chiffré Y_i est additionné par un « OU exclusif » avec le bloc clair rentrant X_{i+1} avant d'être crypté avec la clef k. Un vecteur d'initialisation (*VI*) est utilisé pour la première itération. En fait tous les modes (sauf ECB) ont besoin d'un *VI*. Dans le mode CFB (*Cipher FeedBack*) $VI = Y_0$. La clef dynamique Z_i est générée par $Z_i = E_k(Y_{i-1})$, $i > 1$ et le bloc chiffré est produit par $Y_i = X_i \oplus Z_i$. Dans le mode OFB (*Output FeedBack*), comme dans CFB, $Y_i = X_i \oplus Z_i$ mais $VI = Z_0$ et $Z_i = E_k(Z_{i-1})$, $i > 1$. Les données en entrée sont cryptées par un « OU exclusif » avec la sortie Z_i. Le mode CTR (*Counter*) a des caractéristiques très similaires à OFB, mais en plus il autorise une propriété d'accès aléatoire pour le décryptage. Il génère la clef dynamique suivante par cryptage de valeur successive d'un compteur. Ce compteur peut être une fonction simple qui produit une séquence pseudo aléatoire. Dans ce mode, la sortie du compteur est l'entrée de AES.

Même si AES est un algorithme de chiffrement par bloc, les modes OFB, CFB et CTR opèrent comme des chiffrements par flot (section suivante). Ces modes ne nécessitent aucune mesure particulière concernant la longueur des messages. Chaque mode a différents avantages et inconvénients. Dans les modes ECB et OFB par exemple tout changement dans le bloc du texte clair X_i provoque dans le bloc chiffré correspondant Y_i une modification, mais les autres blocs chiffrés ne sont pas affectés. D'un autre coté, si un texte clair du bloc X_i est changé dans les modes CBC et CFB, alors Y_i et tous les blocs chiffrés conséquents seront affectés. Ces propriétés signifient que les modes CBC et CFB sont utiles pour des problèmes d'authentification et les modes ECB et OFB traitent séparément chaque bloc. Par conséquent, nous pouvons noter que le mode OFB ne diffuse pas le bruit, alors que le mode CFB le diffuse.

10.3.4. *Système par bloc asymétrique : RSA*

L'algorithme RSA est le système asymétrique le plus utilisé. Sa sécurité réside sur la lenteur des ordinateurs actuels pour factoriser des très grands nombres en produits de facteurs premiers [SCH 97] [SHA 78]. Soient p et q deux très grands nombres premiers[2] distincts, et n un très grand nombre qui est le produit de p et q. On note $\phi(n)$ la fonction d'Euler en n (nombres entiers naturels inférieurs à n et premiers avec n, avec $\phi(n) = (p-1)(q-1)$.

La paire clef publique/clef privée va résider dans deux nombres d et e associés à n. e est d'abord calculé aléatoirement entre 2 et $\phi(n)$ et doit être premier avec $\phi(n)$. La paire (n,e) constitue la clef publique. Puis d est calculé tel que $d = e^{-1} \bmod \phi(n)$. L'algorithme d'Euclide étendu permet de calculer cet inverse instantanément, même avec de très grands nombres. La paire (n,d) constitue la clef privée. L'usage des clefs dans le cryptage et le décryptage est le suivant. Si m est le message clair (inférieur à n, sinon on le découpe), on le crypte avec la clef publique (n,e) en l'élevant à la puissance e, modulo n. On obtient le message chiffré $m' = m^e \bmod \phi(n)$. Pour le décryptage, nous avons besoin de la seconde clef (n,d). En élevant le message crypté à la puissance d modulo n et comme d et e sont inverses modulo n, on obtient :

$$(m')^d \bmod(n) = (m^e \bmod(n))^d \bmod(n) = m^{ed} \bmod(n) = m \qquad [10.1]$$

Par exemple, si Bob souhaite envoyer un message à Alice, il convertit son message en nombre et coupe le message en blocs de taille plus petite que n. Pour chaque bloc m_i, en utilisant la clef publique d'Alice, Bob calcule et chiffre le bloc de la manière suivante :

$$c_i = m_i^e \bmod(n), \qquad [10.2]$$

avec i, la position du bloc dans le texte, $i \in [1,N]$, si N est le nombre de blocs.

Alice, avec sa clef privée, peut alors décrypter le message en faisant :

$$m_i = c_i^d \bmod(n) \qquad [10.3]$$

Ainsi, la méthode RSA se distingue des systèmes de chiffrement symétriques en utilisant deux clefs différentes pour le cryptage et le décryptage (figure 10.2.). L'une de ces deux clefs, la clef publique, sera censée être connue de tous,

2. Deux nombres sont premiers entre eux si et seulement si leur plus grand commun diviseur est égal à 1.

et l'autre, la clef privée, connue par un seul individu. L'algorithme RSA peut permettre soit de crypter avec une clef publique, dans lequel cas seul le destinataire pourra déchiffrer le message avec sa clef privée, soit de crypter avec sa propre clef privée (signature). Dans ce cas, tout le monde peut lire le message grâce à la clef publique, mais l'émetteur a pu signer le message puisqu'il est potentiellement le seul à avoir pu crypter avec sa clef privée. Un double cryptage clef publique/clef privée permet alors de combiner signature et confidentialité (figure 10.2.b.).

Malheureusement, RSA est un algorithme très lent, beaucoup plus lent que n'importe quel système symétrique, et d'autant plus que les nombres utilisés sont grands. De plus, il est aujourd'hui facilement cassable, même pour des nombres de 512 bits[3]. Il est actuellement conseillé d'utiliser des clefs de longueur 1 024 bits. Il est donc préférable de l'utiliser pour envoyer de manière sécurisée une clef secrète, qui permettra de déchiffrer le message, avec AES plus rapide que RSA.

10.3.5. *Algorithmes de chiffrement par flot*

Les algorithmes de chiffrement par flot peuvent être définis comme étant des algorithmes de chiffrement par bloc, où chaque bloc est de dimension unitaire (1 bit, 1 octet, etc.) ou relativement petit. Leurs principaux avantages sont leur extrême rapidité et leur capacité à changer à chaque symbole du texte clair. Avec un algorithme de chiffrement par flot, il est possible de crypter séparément chaque caractère du message clair un par un, en utilisant une fonction de cryptage qui varie à chaque fois (ces algorithmes ont donc besoin de mémoires). Généralement, les algorithmes de chiffrement par flot sont composés de deux étapes : la génération d'une clef dynamique et la fonction de cryptage de sortie dépendant de la clef dynamique.

Quand la clef dynamique est générée indépendamment du texte clair et du texte chiffré, l'algorithme de chiffrement par flot est dit synchrone. Avec un chiffrement par flot, l'émetteur et le récepteur doivent se synchroniser en utilisant la même clef et en l'utilisant à la même position. Les chiffrements par flot synchrone sont utilisés principalement dans des environnements où les erreurs sont fréquentes car ils ont l'avantage de ne pas propager les erreurs [GUI 02]. Concernant les attaques actives comme l'insertion, la suppression et la copie de digits du texte chiffré par un adversaire actif, celles-ci produisent immédiatement une perte de synchronisation. Le processus de cryptage d'un

3. Un RSA avec une clef de 560 bits a été cassé en 2003 par une équipe internationale de chercheurs.

chiffrement par flot synchrone est décrit figure 10.4., où *f()* est la fonction qui détermine l'état suivant, *g()* est la fonction génératrice de la clef dynamique et *h()* la fonction de sortie de cryptage :

$$\begin{cases} s_{i+1} = f(K, s_i) \\ z_i \;\; = g(K, s_i) \\ c_i \;\; = h(z_i, m_i) \end{cases} \qquad [10.4]$$

où K est la clef, s_i, m_i, c_i et z_i sont respectivement le i^e état, le texte clair, le texte chiffré et la clef dynamique. Le processus de décryptage est illustré figure 10.4.

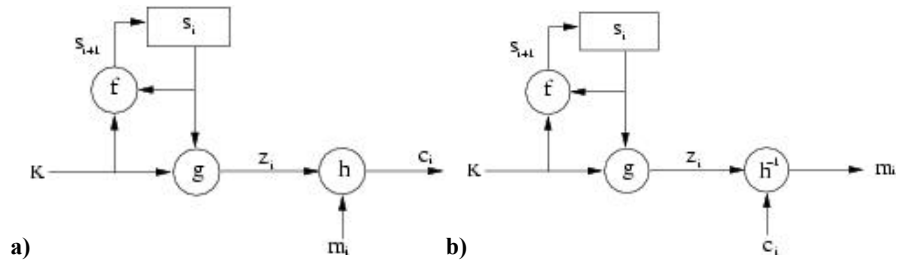

Figure 10.4. *Chiffrement par flot synchrone.* **a)** *Cryptage.* **b)** *Décryptage.*

Quand la clef dynamique est générée à partir de la clef et d'un certain nombre de digits précédemment cryptés, l'algorithme de chiffrement par flot est dit asynchrone, appelé aussi chiffrement par flot auto-synchronisant. La propagation des erreurs est limitée à la taille de la mémoire. Si des digits du texte chiffré sont effacés ou insérés en plus, le récepteur est capable avec la mémoire de se re-synchroniser avec l'émetteur. Concernant les attaques actives, si un adversaire actif modifie une part des digits du texte chiffré, le récepteur est capable de la détecter. Le processus de cryptage d'un chiffrement par flot asynchrone est décrit figure 10.5., où *g()* est la fonction génératrice de la clef dynamique et *h()* la fonction de sortie de cryptage :

$$\begin{cases} z_i \;\; = g(K, c_{i-t}, c_{i-t+1}, ..., c_{i-2}, c_{i-1}) \\ c_i = h(z_i, m_i) \end{cases} \qquad [10.5]$$

où K est la clef, m_i, c_i et z_i sont respectivement le i^e texte clair, le texte chiffré et la clef dynamique. Nous pouvons remarquer équations [10.5] que la clef dynamique dépend des t digits précédents du texte chiffré. Afin d'être robuste à de nombreuses attaques statistiques, la fonction génératrice de la clef dynamique *g()* doit produire une séquence d'une large période avec de bonnes propriétés statistiques qui peuvent être appelées séquences binaires pseudo aléatoires. Le processus de décryptage est illustré figure 10.5.

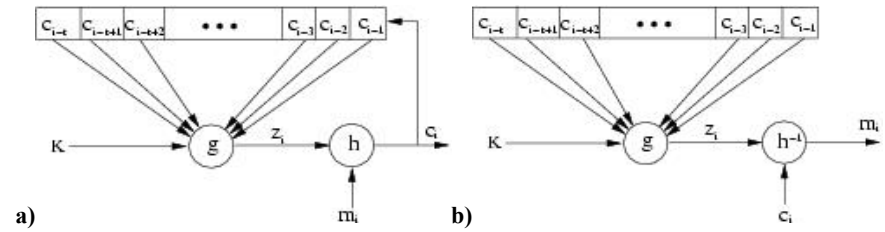

Figure 10.5. *Chiffrement par flot asynchrone* **a)** *Cryptage.* **b)** *Décryptage*

10.4. Cryptage d'images médicales

Dans cette section, nous exposerons comment il est possible d'appliquer les algorithmes présentés précédemment à des images médicales en niveaux de gris. Dans le cas de signaux médicaux 1D, les algorithmes standard de chiffrement peuvent être directement appliqués. Par contre, du fait de la notion de voisinage bidimensionnelle des images et de leur taille, ces algorithmes standard doivent être modifiés afin d'être applicables efficacement sur les images médicales. L'objectif du chiffrement d'images est d'obtenir une image de même format et de taille égale au maximum à la taille de l'image originale.

Le chiffrement d'images est considéré comme un codage source de manière à traiter cette fonctionnalité au niveau de l'application. De ce fait, si un utilisateur n'a pas la bonne clef, il a accès au moins à une image dans un format connu. En remontant le chiffrement au niveau de l'application, il est possible de procéder, par exemple, à un fenêtrage d'image. Dans le cas d'images de grande taille, il n'est alors pas utile de décrypter toute l'image si nous ne souhaitons en visualiser qu'une zone particulière. L'étape de compression sera également à prendre en compte dans la phase de chiffrement des images.

10.4.1. *Chiffrement d'images par bloc*

Dans le cas de chiffrement par bloc, la longueur des blocs est imposée et varie entre 64 bits (8 pixels) et 192 bits (24 pixels). Du fait de l'information bidimensionnelle d'une image, plusieurs solutions de regroupement de pixels sont possibles. En effet dans l'objectif de mieux résister à une compression aval ou de comprimer en même temps que le chiffrement, il est intéressant de regrouper les pixels avec leurs voisins les plus proches (par ligne, par colonne ou par bloc). Chaque bloc de pixels sera crypté indépendamment. Le bloc crypté obtenu viendra alors se substituer dans l'image au bloc original. Dans ce chapitre, le parcours de lecture des blocs sera exécuté uniquement de manière linéaire (*scan line*). Manniccam et Bourbakis montrent qu'il est souvent plus intéressant d'utiliser

d'autres types de parcours (spirale, zig-zag, etc.) afin de combiner cryptage et compression sans perte [MAN 01] [MAN 04].

10.4.2. *Chiffrement d'images par flot asynchrone*

Dans cette section nous présentons un algorithme de chiffrement par flot asynchrone appliqué aux images. Soit K une clef de longueur k bits b_i, $K = b_1 b_2 ... b_k$. L'unité de cryptage est le pixel (1 octet). La méthode réside dans le fait que pour chaque pixel de l'image le cryptage dépend du pixel original, de la valeur de la clef K, et des $k/2$ pixels précédemment cryptés. Pour utiliser les équations [10.5] nous avons $t = k/2$. Pour chaque pixel p_i de l'image originale, nous calculons la valeur du pixel p'_i de l'image chiffrée en utilisant l'équation suivante :

$$\begin{cases} z_i = \left(\sum_{j=1}^{k/2} \alpha_j \, p'_{i-j} \right) \bmod(256) \\ p'_i = (z_i + p_i) \bmod(256) \end{cases} \qquad [10.6]$$

avec $i \in [0,\ldots,N-1]$ où N est le nombre de pixels de l'image, k est la longueur de la clef avec $k \in [1,N]$, et α_j est une séquence de $k/2$ coefficients générée à partir de la clef secrète K [PUE 01a] [PUE 01b].

Le principe de chiffrement est le même que celui illustré figure 10.5. Les équations [10.6] ont une récurrence d'ordre $k/2$, correspondant à la moitié de la longueur de la clef [PUE 01c].

Les coefficients α_j sont des coefficients entiers compris entre −2 et +2 tels que :

$$\begin{cases} \alpha_j = \beta_j - 1 & \text{si } \beta_j \in \{0,1,2\} \\ \alpha_j = \pm 2 & \text{si } \beta_j = 3 \end{cases} \qquad [10.7]$$

avec $\beta_j = 2b_{2j-1} + b_{2j}$, où b_{2j-1} et b_{2j} sont deux bits voisins de la clef secrète K. De plus, la densité de probabilité des α_j doit être uniforme afin d'atténuer les erreurs de transmission durant l'étape de décryptage. Le signe devant les coefficients égaux à 2 dépend de la somme des coefficients α_j afin d'avoir :

$$\frac{1}{k/2} \sum_{j=1}^{k/2} \alpha_j \approx 0 \qquad [10.8]$$

En considérant que le chiffrement d'un pixel s'appuie sur les *k/2* pixels précédemment cryptés, nous ne pouvons pas chiffrer les *k/2* premiers pixels de l'image de la même manière. Il est nécessaire d'associer la séquence des coefficients α_i à une séquence de *k/2* pixels virtuels cryptés p'_{-i}, pour $i \in [1,\ldots,k/2]$, correspondant à un vecteur d'initialisation (*VI*). Par conséquent, un *VI* est codé dans la clef : *k/2* valeurs de pixels virtuels qui permettent de crypter les *k/2* premiers pixels de l'image comme s'ils avaient des prédécesseurs. La longueur *k* de la clef *K* doit être suffisamment grande afin de garantir une sécurité maximale.

L'équation [10.9] présente la procédure de décryptage. Dans la procédure de décryptage, nous devons appliquer le processus inverse. Nous pouvons noter que la fonction génératrice de la clef dynamique est la même qu'à l'équation [10.6] :

$$\begin{cases} z_i = \left(\sum_{j=1}^{k/2} \alpha_j p'_{i-j} \right) \bmod(256) \\ p_i = (p'_i - z_i) \bmod(256) \end{cases} \qquad [10.9]$$

10.4.3. *Application du cryptage aux images médicales*

A partir de l'image de la figure 10.6.a., nous avons appliqué l'algorithme DES par blocs de 8 pixels en ligne avec une clef de 64 bits pour obtenir l'image de la figure 10.6.c. De la même manière, avec l'algorithme AES par bloc avec une clef de 128 bits, nous obtenons l'image de la figure 10.6.e.

Nous constatons l'apparition de textures (figures 10.6.c-e.). La raison de ce phénomène se trouve dans l'apparition de grandes zones homogènes (en l'occurrence noires) sur les images médicales.

Au niveau des histogrammes (figures 10.6.d-f.) nous constatons la présence forte de niveaux de gris correspondant au cryptage des niveaux de gris de zones homogènes.

Le cryptage est alors très mauvais pour deux raisons : premièrement parce qu'il est facile de deviner la nature de l'image médicale (échographie), et surtout secondement parce que la connaissance de la valeur du bloc clair (les pixels clairs étaient tous noirs) et après cryptage (les niveaux de gris dominant dans l'image cryptée) est un indice précieux pour les cryptanalystes.

Les algorithmes de chiffrement par bloc posent donc des problèmes dans le cas d'images contenant des zones homogènes.

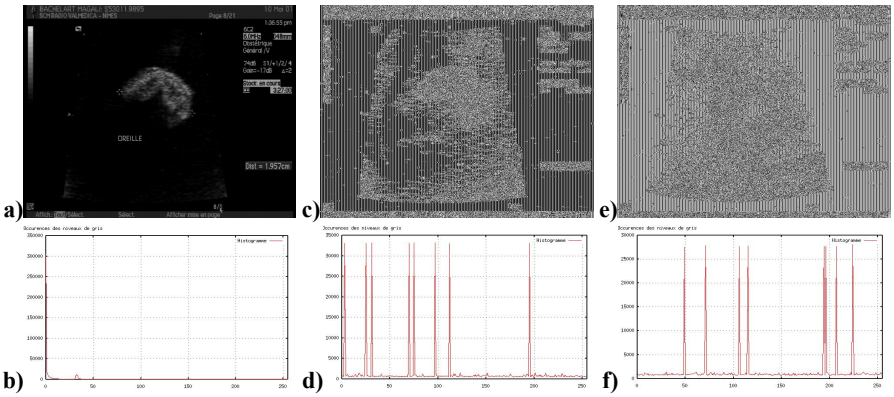

Figure 10.6. a) *Image médicale échographique (442 Ko), avec de grandes zones homogènes, image cryptée.* **c)** *Par l'algorithme DES (bloc de 8 pixels avec une clef de 64 bits),* **e)** *Par l'algorithme AES (bloc de 8 pixels avec une clef de 128 bits)* **b)***,* **d)** *et* **f)** *Histogrammes*

A partir de l'image originale, figure 10.7.a. (396*400 pixels), nous avons appliqué un algorithme de chiffrement par flot asynchrone avec une clef de 128 bits.

La figure 10.7.c. illustre les valeurs obtenues pour la clef dynamique z_i générée par l'équation [10.6]. Nous pouvons remarquer (figure 10.7.d.) que la probabilité d'apparition de chaque valeur est quasi uniforme.

Donc la fonction génératrice de la clef dynamique *g()* produit une séquence avec une grande période et de bonnes propriétés statistiques. A partir des équations [10.6] nous obtenons l'image cryptée (figure 10.7.e.), nous remarquons que l'image initiale n'est plus du tout visible. En comparant l'histogramme de l'image initiale (figure 10.7.b.) avec l'histogramme de l'image cryptée (figure 10.7.f.), nous remarquons que la densité de probabilité des niveaux de gris est quasi-uniforme.

Par conséquent, les entropies des images cryptées sont très élevées (proche de 8 bits/pixel).

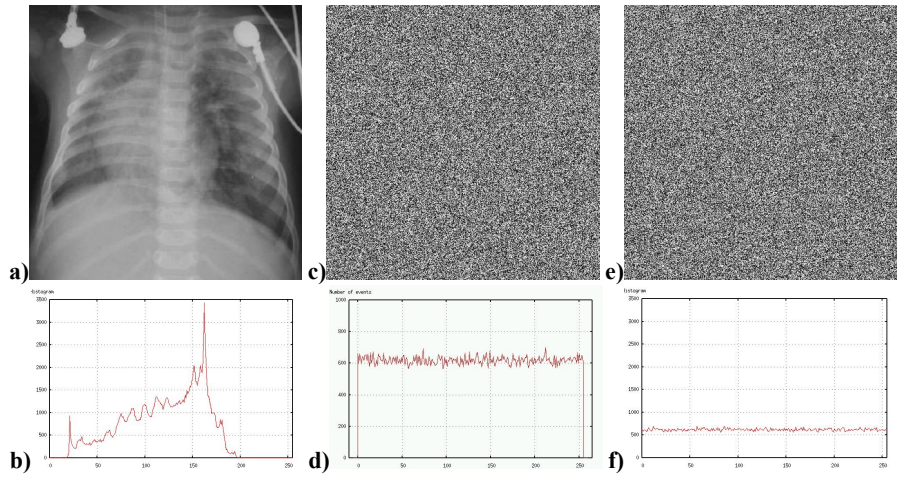

Figure 10.7. a) *Image originale.* **b)** *Histogramme de l'image originale.* **c)** *Image de la clef dynamique* z_i. **d)** *Histogramme des valeurs de la clef dynamique* z_i, **e)** *Image finale cryptée avec l'algorithme de chiffrement par flot asynchrone, avec une clef de 128 bits,* **f)** *Histogramme de l'image cryptée.*

Le chiffrement par flot présente un avantage majeur par rapport aux autres systèmes de cryptage utilisés pour ce qui est des images médicales. En effet, puisqu'on tient compte pour chaque pixel à crypter du résultat du cryptage des pixels précédents, le problème des zones homogènes est résolu. Nous ne sommes plus dans le cas des systèmes de chiffrement par bloc où deux blocs clairs identiques donnaient le même bloc crypté. Nous constatons que quelque soit le type d'image avec ou sans zones homogènes, aucune texture n'apparaît dans les images cryptées. En conclusion dans le cas d'un chiffrement par flot, les zones homogènes ne sont plus visibles ni au niveau de l'image, ni au niveau de l'histogramme. La méthode de chiffrement par flot présente un autre avantage : comme les calculs qui la composent sont peu nombreux, elle s'avère très rapide, plus encore que l'AES. Par exemple, une image de 7 Mo est cryptée (ou décryptée) en 5 secondes avec un PC standard au lieu de 15 secondes pour les algorithmes de chiffrement par bloc.

10.4.4. *Cryptage sélectif d'images médicales*

Une autre manière d'assurer la confidentialité est d'adapter le niveau de protection en fonction de l'application et du temps disponible. C'est dans cette seconde approche que nous trouvons le cryptage sélectif où les utilisateurs peuvent appliquer une sécurité proportionnelle ou réglable en fonction du niveau de protection désiré [NOR 03]. De nombreuses applications peuvent se contenter d'un cryptage sélectif ; les images sont alors partiellement visibles sans révéler

complètement toute l'information. Le cryptage sélectif peut être intéressant dans le cas des images médicales prises depuis un appareil médical et devant être envoyées sur le réseau afin d'établir un diagnostic à distance. De plus, l'appareil d'acquisition d'images médicales peut se trouver dans une ambulance ou dans tout autre véhicule mobile, et dans ce cas la transmission est effectuée par l'intermédiaire de réseaux sans fil. Pour des raisons vitales, dans ce type d'applications, les images doivent être transmises rapidement et sûrement, et dans ce cas un cryptage sélectif semble être la meilleure solution (compromis temps/sécurité).

Dans cette section nous présentons une méthode de cryptage sélectif pour des images médicales comprimées avec JPEG [PUE 06]. Cette méthode est basée sur l'algorithme AES, utilisant le mode de chiffrement par flot OFB (*Output Feedback Block*) dans l'étape du codage de Huffman de l'algorithme JPEG. La combinaison du cryptage sélectif et de la compression permet de gagner du temps de calcul et de conserver le format JPEG et le taux de compression initial. Du point de vue sécurité, le cryptage sélectif garantit un certain niveau de confidentialité. Beaucoup de méthodes de cryptage sélectif ont été développées pour des images codées par DCT. Tang [TAN 96] a proposé une technique appelée permutation zigzag applicable à des vidéos ou des images. Bien que sa méthode offre une bonne confidentialité, elle diminue le taux de compression. [DRO 02] décrit une technique qui crypte un nombre sélectionné de coefficients AC ; les coefficients DC ne sont pas cryptés car ils portent une information visible importante et sont hautement prédictibles. Pour cette méthode, le taux de compression est constant (par rapport à la compression seule) et conserve le format du flux binaire. Par contre la compression et le cryptage sont faits séparément, la méthode est donc plus longue que la compression seule. [FIS 04] présente une méthode telle que les données sont organisées dans une forme de flux binaire réglable. Récemment, Said a montré la force des méthodes de cryptage partiel en testant des attaques qui exploitent l'information non cryptée de l'image associée à une image de petite taille [SAI 05].

Soit $E_K(X_i)$ le cryptage d'un bloc X_i de n bits utilisant la clef secrète K avec l'algorithme AES en mode OFB. Dans la description de la méthode, nous supposerons $n = 128$ et X_i un texte clair non vide. Soit $D_K(Y_i)$ le décryptage d'un texte chiffré Y_i en utilisant la clef secrète K. Le cryptage est appliqué en même temps que le processus de codage entropique durant la création du vecteur de Huffman. La méthode travaille en trois étapes illustrées dans la figure 10.8. : la construction du texte clair X_i, le cryptage de X_i pour créer Y_i et la substitution du vecteur original de Huffman par l'information cryptée [ROD 06]. Il est important de mentionner que ces opérations sont appliquées séparément pour chaque bloc DCT quantifié.

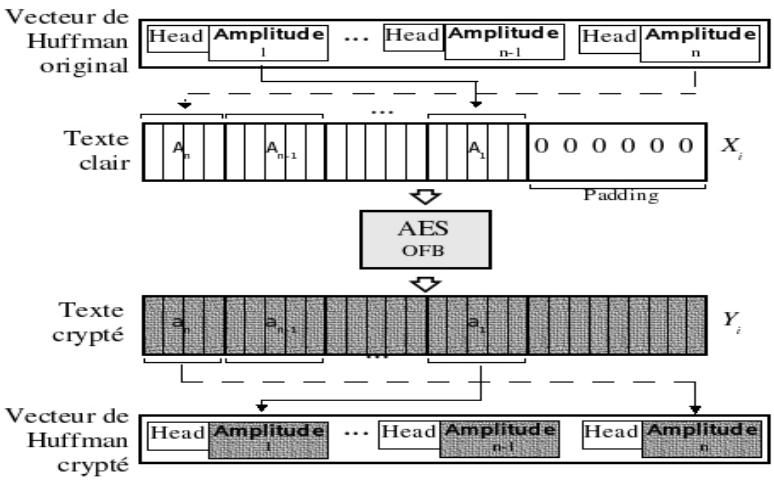

Figure 10.8. *Présentation générale de la méthode proposée*

Pour construire le texte clair X_i, nous prenons les coefficients AC non nuls du bloc courant i en accédant au vecteur de Huffman de la fin vers le début afin de créer des paires {HEAD, AMPLITUDE}. De chaque entête HEAD nous obtenons la longueur de l'Amplitude en bits. Seules les Amplitudes (A_n, A_{n-1},..., A_1) sont prises en compte pour construire le vecteur X_i. La longueur finale du message en clair L_{Xi} dépend à la fois de l'homogénéité ρ du bloc et d'une contrainte donnée C, avec $C \in$ {128, 64, 32, 16, 8} bits. Cette contrainte C spécifie la quantité maximale de bits qui doit être prise en compte dans chaque bloc. C'est donc par l'intermédiaire de C qu'est graduée l'importance du cryptage. D'un autre coté, l'homogénéité dépend du contenu de l'image et spécifie la quantité minimale de bits. Cela signifie qu'un bloc avec un grand ρ va produire un petit L_{Xi}. Le vecteur de Huffman est traité tant que $L_{Xi} \leq C$ et que le coefficient DC n'est pas atteint. Ensuite, la fonction de remplissage (*padding*) est appliquée, $p(j) = 0$, où $j \in \{ L_{Xi}, \ldots, 128\}$, afin de remplir si nécessaire avec des zéros le vecteur X_i.

Dans l'étape de chiffrement de X avec AES en mode OFB, la clef dynamique Z_{i-1} est utilisée comme entrée pour le cryptage par AES afin d'obtenir une nouvelle clef dynamique Z_i. Pour la première itération, le vecteur VI est créé à partir de la clef secrète K avec la stratégie suivante : la clef secrète K est utilisée comme une semence pour un générateur de nombres pseudo aléatoire (GNPA). Ce K est divisé en 16 portions de 8 bits chacun. Le GNPA produit 16 nombres aléatoires qui définissent l'ordre de formation du VI. Ensuite chaque Z_i est additionné par un « OU exclusif » avec le texte en clair X_i pour générer le bloc chiffré Y_i.

L'étape finale est la substitution de l'information initiale par l'information chiffrée dans le vecteur de Huffman. Comme dans la première étape (construction du texte clair X_i), le vecteur de Huffman est lu depuis la fin vers le début mais le vecteur chiffré Y_i est lu du début vers la fin. Connaissant la longueur en bits de chaque Amplitude (A_n, A_{n-1},..., A_1), ces portions sont coupées dans Y_i pour remplacer l'Amplitude dans le vecteur de Huffman. La quantité totale de bits doit être L_{Xi}. Cette procédure est faite pour chaque bloc. Les blocs homogènes ne sont pas ou peu chiffrés. L'utilisation du mode OFB pour le chiffrement permet une génération de clef dynamique Z_i indépendante. Dans la phase de décryptage en mode OFB, la clef dynamique Z_i est additionnée par un « OU exclusif » avec le bloc chiffré Y_i afin de régénérer le texte en clair X_i. Le vecteur résultat du texte en clair X_i est coupé en parties de la fin vers le début afin de remplacer les Amplitudes dans le chiffré de Huffman pour générer le vecteur de Huffman.

Cette méthode a été appliquée sur plusieurs dizaines d'images médicales en niveau de gris (exemple donné figure 10.9.).

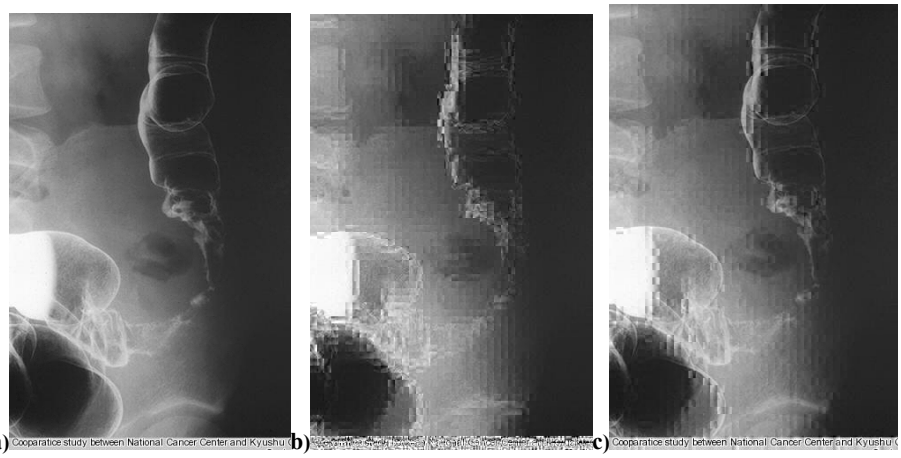

Figure 10.9. a) *Image médicale originale d'un cancer du colon, 320*496 pixels.*
b) *Image cryptée pour C = 128.* **c)** *Image cryptée pour C = 8*

L'algorithme JPEG a été utilisé avec le système de codage en ligne séquentiel pour un facteur de qualité (FQ) de 100%. Cinq valeurs ont été appliquées pour la contrainte C (128, 64, 32, 16 et 8). Pour le chiffrement, l'algorithme AES a été employé avec le mode de chiffrement par flot OFB avec une clef de longueur 128 bits. L'image médicale originale de taille 320*496 pixels (figure 10.9.), comprimée ainsi que toutes les images cryptées ont la même taille soit 43.4 Ko. Les coefficients cryptés sont répartis dans les 2 480 blocs 8*8 de l'image. Cela signifie qu'il n'y a aucun bloc totalement homogène. Pour C = 128, maximum de 128 bits chiffrés par

bloc, 26 289 coefficients AC ont été chiffrés soit une moyenne de 33 bits chiffrés par bloc. Le pourcentage de bits chiffrés dans l'image entière est de 22,99% soit, dans le domaine spatial, 136 038 pixels changés, ce qui correspond à 85,71% des pixels chiffrés. Le PSNR est de 23.39 dB pour C = 128. Pour C = 8, les quantités de coefficients AC et de bits codés sont respectivement 6 111 et 16 765. Le pourcentage de bits chiffrés par rapport à l'image entière est de 4,7%. Cette contrainte donne un nombre de pixels modifiés correspondant à 76,1% de tous les pixels de l'image. Le PSNR est alors de 30.90 dB. Comme le montre les images résultats, le cryptage sélectif sur toute l'image JPEG produit des artefacts par bloc. Ces artefacts sont au niveau des frontières des blocs, qui importunent souvent le SVH. Puisque la transformation fréquentielle et la quantification des blocs de pixels sont traitées séparément, la continuité des valeurs des pixels de blocs voisins est cassée durant le codage. Un des avantages de cette méthode est la possibilité de décrypter de manière individuelle les blocs 8*8 pixels de l'image (utilisation du mode par flot OFB pour le cryptage par AES). En effet, afin d'établir un diagnostic à distance, un médecin doit pouvoir visualiser une région d'intérêt en haute résolution d'une image où le fond peut-être partiellement chiffré. Il convient de noter que la confidentialité est liée à la capacité de deviner les valeurs des données chiffrées (cryptanalyse). D'un point de vue sécurité, il est donc préférable de chiffrer les bits qui semblent les plus aléatoire [PUE 05].

1.5. Marquage d'images médicales et chiffrement

10.5.1. *Tatouage des images et applications en santé*

Le marquage des images rentre dans le contexte plus général de la dissimulation d'information : il s'agit de dissimuler un message dans un document, un hôte qui peut être du texte, du son, de la vidéo et des images. Pour les images, le signal de différence entre l'image originale et sa version tatouée constitue la marque associée au message tatouée. On prendra soin à ce que le document hôte marqué ait la même valeur que le document hôte original. Nous donnons figure 10.10. l'exemple d'une chaîne de tatouage.

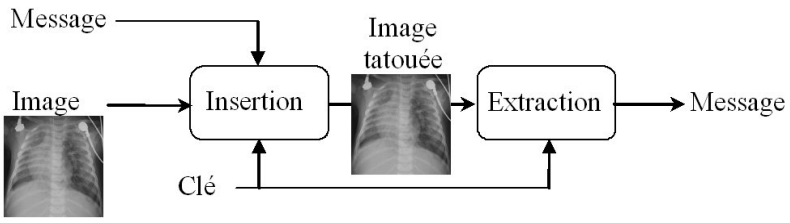

Figure 10.10. *Une chaîne de marquage*

En fonction du lien établi entre le message et son hôte nous distinguons la stéganographie, le tatouage et le *fingerprinting*. La stéganographie s'apparente à une communication secrète dans laquelle l'hôte, qui constitue le canal de communication caché, n'a que peu d'intérêt pour le destinataire du message. Le tatouage et le *fingerprinting* ont quant à eux pris leur essor dès 1995 pour répondre aux problèmes de gestion des droits d'auteurs de documents accessibles ou diffusés sur Internet[4]. Le tatouage vise l'insertion d'un code identifiant le propriétaire et le *fingerprinting* l'insertion d'une trace liée à l'acheteur. Depuis, d'autres applications ont été proposées sur la base de la dissimulation d'information comme le contrôle de copie, d'accès ou d'intégrité. Pour plus de détails sur les applications et les méthodes de tatouage y compris pour les images médicales, le lecteur pourra se référer à l'ouvrage « Tatouage de documents audiovisuels numériques » dans la même collection [DAV 04]. Dans la suite de ce paragraphe, nous utiliserons indifféremment les termes marquage et tatouage.

Ainsi, pour les images médicales, plusieurs scénarios utilisant le tatouage ont été identifiés [COA 00] :

– l'authentification des images avec l'insertion d'une information certifiant l'origine et l'attachement d'une image à un patient donné ;

– le contrôle d'intégrité des images, en plaçant dans l'image une information de contrôle comme par exemple une signature numérique (section 10.3.) ;

– l'ajout de méta-données (*data-hiding*) permettant d'enrichir le contenu des images en lui associant une description sémantique de son contenu [COA 05] ;

Un autre scénario plus complet combine à la fois l'authentification et le contrôle d'intégrité des images et cherche à établir un lien sûr entre ces images et le compte rendu d'examen associé [COA 06]. Dans la protection de l'information [COA 03], le tatouage apparaît comme complémentaire des mécanismes discutés précédemment car il associe l'information de protection et l'image à protéger en une seule entité : une image marquée.

10.5.2. *Techniques de marquage et images médicales*

10.5.2.1. *Propriétés*

Les techniques de dissimulation de données proposées pour les images sont nombreuses et différentes dans leur mise en œuvre. Elles ont néanmoins des caractéristiques communes qui, en fonction de l'application, sont à considérer pour choisir la technique la plus appropriée :

4. DRM en anglais pour *Digital Right Management.*

– la *robustesse* : une méthode est qualifiée de robuste si après modification de l'image tatouée (attaque de « lessivage » ou simple traitement d'image) l'information dissimulée est toujours accessible et compréhensible. Cette propriété est essentielle s'il s'agit d'authentifier des images qui sont *a priori* amenées à subir des traitements ou une compression avec pertes ;

– la *capacité* : cette mesure traduit le taux d'insertion exprimé en nombre de bits de message enfoui par pixel de l'image (bpp pour « bit per pixel ») et donc donne une indication sur la taille du message qu'il est possible d'enfouir dans une image. Les techniques de *data hiding* cherchent à optimiser ce paramètre et ainsi ajouter de l'information utile aux images ;

– la *sécurité* : pour quelques situations, il est nécessaire de contrôler l'accès à la marque et à son contenu ; comme pour la cryptographie, il existe des méthodes de marquage symétrique et asymétrique ;

– la *complexité* : il s'agit d'une indication sur les temps de calcul nécessaires lors des étapes d'insertion et d'extraction ; la complexité joue un rôle important s'il s'agit de traiter des volumes d'images de grande taille ;

– l'*invisibilité* : pour les images médicales cette notion est importante car pour ne pas porter préjudice au diagnostic, la marque ne doit pas interférer avec l'interprétation de l'image ; sur cette base, des méthodes de marquage ont été spécifiquement proposées pour les images médicales ;

– la nécessité de l'image originale pour le décodage fait aussi partie de ces propriétés ; une méthode est dite « aveugle » si l'image originale n'est pas nécessaire pour extraire la marque ; les applications de contrôle d'intégrité ne peuvent pas être envisagées avec des méthodes de marquage non aveugles, la question de savoir si l'image originale n'a pas été modifiée est alors un non sens.

Une méthode peut satisfaire indépendamment différentes applications mais pas simultanément. Un compromis s'impose entre capacité, robustesse et invisibilité. Une marque plus forte résistera mieux aux altérations du signal induites par la compression ou des tentatives de fraude ; mais sa présence sera plus facilement perçue par l'utilisateur et la capacité en sera réduite.

10.5.2.2. *Les méthodes*

Dans leurs principes, les techniques de marquage proposées pour les images médicales ne diffèrent que très peu des autres méthodes que nous pouvons maintenant qualifier de « classiques ». Elles profitent de stratégies d'adaptation particulières fonctions de la spécificité des images médicales.

Pour les schémas usuels, deux familles d'algorithmes sont en général distinguées. La première correspond aux méthodes additives. A partir du message (une séquence de bits), elles engendrent un signal qui est ajouté à l'image ou à une

transformation de celle-ci (TCD, ondelettes, etc.). Une technique par étalement de spectre consiste à associer à chaque bit b_j du message la valeur $d_j = 1\text{-}2b_j$ puis à multiplier cette quantité par une porteuse w_j de faible énergie qui est ensuite ajoutée à l'image I pour produire l'image tatouée : $I_w = I + \alpha d_j w_j$. α est un paramètre de force d'insertion ou d'incrustation (paramètre de robustesse). L'insertion d'un message de N bits revient alors à ajouter à l'image la marque $W = \sum_{j=1}^{N} \alpha d_j w_j$. La présence de cette marque est vérifiée par des techniques de corrélation, ce qui implique l'orthogonalité des porteuses w_j. Le signe de chaque produit de corrélation donnera la valeur du bit enfoui. L'insertion d'un message de grande taille peut conduire à une marque W particulièrement visible. Pour mieux garantir l'invisibilité de la marque, des critères psycho-visuels sont utilisés de manière à adapter la force d'insertion localement à l'image.

La seconde classe comprend les méthodes substitutives qui, pour insérer un bit du message, remplacent une information liée à l'image (ses niveaux de gris ou une transformation de ceux-ci) par un symbole issu d'un dictionnaire. La détection se fait alors par une simple relecture. La méthode de substitution des bits de poids faible de l'image est la plus simple. Cette méthode remplace simplement les bits de poids faible des niveaux de gris de l'image par ceux du message à tatouer. Pour un pixel de niveau de gris $p(n,m)$, cela revient à associer aux valeurs de $p(n,m)$ impaires la valeur binaire 1 et aux valeurs paires la valeur 0. Cette méthode n'est absolument pas robuste (elle est donc fragile) mais propose une capacité d'1 bpp. Depuis, des versions plus élaborées de cette approche ont vu le jour, elles suivent le schéma de Costa [COS 03]. Ces méthodes qualifiées d'informées [DAV 04] s'appuient sur des dictionnaires structurés qui contiennent les valeurs que des blocs de pixels prendront pour porter une information.

Pour les images médicales, trois stratégies ont été identifiées avec pour objectif premier la préservation de la qualité diagnostique des images. Il s'agit des méthodes précédentes, de méthodes de tatouage par zone et de méthodes de tatouage avec une propriété de réversibilité.

Les premières méthodes produisent des marques qui remplacent dans l'image une partie de l'information. Les utiliser nécessite une attention particulière pour que la marque n'interfère pas avec l'interprétation diagnostique. Les premières solutions alors proposées ont été des techniques modifiant secrètement des bits de poids faible de certains pixels ou de coefficients d'une transformée de l'image. Il s'agit là de méthodes à forte capacité, induisant une faible dégradation du signal original mais cependant très fragile. Plus récemment, des techniques robustes ont été testées avec, lors de l'expérimentation, une interaction avec un praticien précisant un seuil de force d'insertion à ne pas dépasser [PIV 05]. Plus généralement, le problème de l'évaluation automatique du maximum de distorsion autorisée pour une image se

pose. Ce problème est loin d'être réglé du fait de la diversité des signaux d'imagerie en santé (voir chapitre 3) et la mise à disposition des praticiens d'outils permettant par exemple d'isoler certaine partie de la dynamique des images. Suivant l'utilisateur ces plages de niveaux de gris varient et pour certains la marque peut devenir visible.

Une stratégie proposée pour optimiser les performances des méthodes précédentes en robustesse et capacité sans plus dégrader l'image, s'appuie sur l'existence dans l'image de zones d'interactions nulles ou tout au plus minimes avec l'information utile au diagnostic. Ces techniques dites de marquage par zone placent le plus souvent la marque dans le fond noir de l'image [COA 01]. Une certaine robustesse peut être acquise, la marque ne venant pas masquer d'information pertinente, cependant une marque de forte amplitude peut occasionner une certaine gêne pour le diagnostic. La force d'insertion doit être contrôlée.

La dernière approche regroupe les méthodes de tatouage réversible. L'idée est de pouvoir retirer la marque de l'image avec la récupération exacte des niveaux de gris de l'image originale. Ces techniques permettent d'envisager l'actualisation du contenu de la marque. Ce n'est pas le cas avec les méthodes précédentes où les marques s'ajouteront les unes aux autres. La contrepartie est que l'image tatouée n'est plus protégée une fois la marque retirée. La mise en œuvre de telles techniques a bénéficié d'un certain nombre d'efforts ces dernières années [COA 05] [COA 06] [CAV 04]. Les techniques développées offrent des performances relativement variables suivant les types de support marqués et en deçà des méthodes non réversibles. Par ailleurs, ces techniques sont très rarement robustes et vouloir maximiser la capacité conduit le plus souvent à des marques hautement visibles : la marque doit être retirée avant d'utiliser l'image.

Le marquage des images médicales n'en est qu'à ses débuts, la principale difficulté rencontrée étant la mesure perceptuelle de la distorsion alors introduite. Nous pouvons cependant espérer que les travaux conduits sur la mesure de la qualité de la compression d'image (chapitre 5) apporteront un début de réponse de manière à pouvoir bénéficier des avantages du tatouage.

10.5.3. *Confidentialité et intégrité des images médicales par cryptage et dissimulation de données*

Les applications du marquage en imagerie médicale sont nombreuses. Dans ce paragraphe, nous nous proposons d'illustrer la combinaison de la cryptographie et du marquage dans des échanges sécurisés d'images. Nous avons vu section 10.3. que les processus de cryptage pouvaient être symétriques ou asymétriques par bloc ou par flot. Alors que les algorithmes asymétriques ne sont pas appropriés au

cryptage des images à cause de leur temps de calcul, les algorithmes par bloc présentent des problèmes de sécurité (zone homogènes) et en particuliers d'intégrité des données. Les figures 10.11. illustrent ce problème. L'algorithme AES par bloc [AES 01] avec une clef de 128 bits a été appliqué sur l'image originale (figure 10.11.a.) afin d'obtenir l'image cryptée figure 10.11.b. Si l'image cryptée est modifiée durant le transfert il n'est pas forcément possible de détecter la modification. Par exemple dans la figure 10.11.c. une petite région de l'image cryptée a été copiée et collée sur une autre zone de l'image. Après décryptage, il est possible de visualiser les images mais il n'est pas possible de garantir l'intégrité comme illustrée figure 10.11.d.

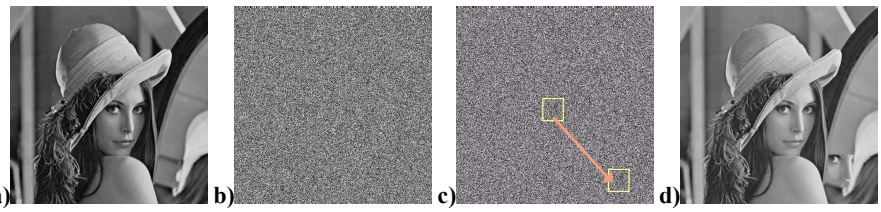

Figure 10.11. a) *Image originale de Lena.* **b)** *Image cryptée avec AES par bloc de 128 bits.* **c)** *Copie d'une région de l'image cryptée et collage sur une autre zone.* **d)** *Décryptage de* **d)**

Afin de résoudre ce problème d'intégrité, il est possible de combiner un algorithme de chiffrement par flot avec une clef secrète pour l'image et un algorithme asymétrique pour chiffrer la clef secrète. Une méthode de marquage de données substitutive (section 10.5.) permet alors d'insérer la clef cryptée dans l'image cryptée [PUE 04]. Si une personne *A* envoie par réseau une image à *B*, l'émetteur *A* utilisera l'algorithme de chiffrement par flot avec la clef secrète *K* pour crypter l'image. Afin de transmettre la clef *K*, *A* peut la chiffrer en utilisant un algorithme à clef publique tel que RSA. Soit *pub(e,n)* la clef publique et *priv(d,n)* la clef privée pour RSA avec $e = d^{-1} \bmod(n)$, alors *A* a ses clefs publique et privée $pub_a(e_a,n_a)$ et $priv_a(d_a,n_a)$, et B ses clefs publique et privée $pub_b(e_b,n_b)$ et $priv_b(d_b,n_b)$. Par conséquent *A* génère une clef secrète *K* pour cette session et chiffre l'image avec l'algorithme de chiffrement par flot. Ensuite *A* chiffre la clef *K* avec l'algorithme RSA en utilisant sa clef privée $priv_a$ afin d'obtenir un clef signée *K'* telle que :

$$K' = K^{d_a} \bmod(n_a) \qquad [10.10]$$

Cette clef signée *K'* est chiffrée une seconde fois avec RSA en utilisant la clef publique pub_b de son correspondant *B* afin de générer *K''* :

$$K'' = K'^{e_b} \bmod(n_b) \qquad [10.11]$$

La taille du message à insérer dans l'image dépend de la taille de la clef publique du récepteur et est connue par l'émetteur *A* et le récepteur *B*. Nous pouvons donc calculer le facteur d'insertion et calculer le nombre de blocs nécessaires pour la méthode de marquage. Cette clef *K''* est insérée dans l'image chiffrée. Finalement, *A* envoie l'image à *B* comme présentée figure 10.12. Cette procédure de cryptage *K* avec $priv_a$ et pub_b assure l'authenticité et seul *B* peut décrypter l'image envoyée. Le fait d'insérer la clef dans l'image rend la méthode autonome et garantit l'intégrité. En effet, si durant le transfert l'image est attaquée alors il n'est plus possible à la réception d'extraire la bonne clef et donc de décrypter l'image.

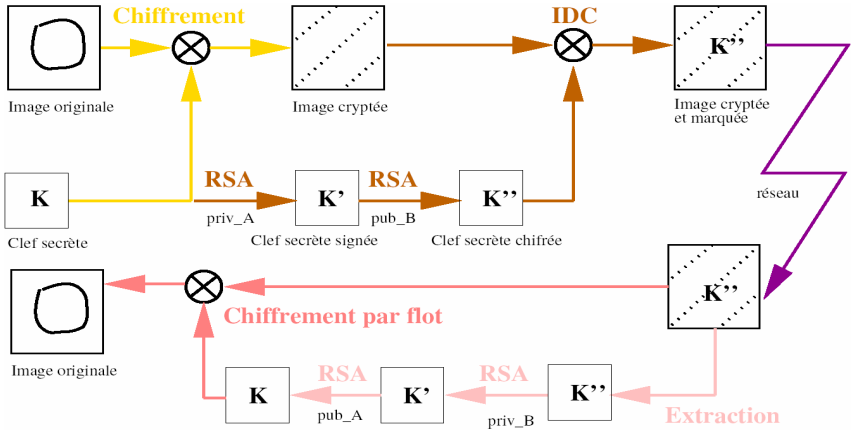

Figure 10.12. *Combinaison d'un cryptage à clef secrète, d'un cryptage à clef publique et d'une méthode d'IDC*

La personne *B* reçoit l'image cryptée et marquée et peut alors extraire la clef cryptée *K''*. Il peut alors authentifier *A* et décrypter la clef *K''* en utilisant sa clef privée $priv_b$ et la clef publique pub_a de *A* tel que :

$$K = (K''^{d_b} \bmod(n_b))^{e_a} \bmod(n_a) \qquad [10.12]$$

Avec la clef obtenue *K*, *B* peut déchiffrer l'image et la visualiser. A partir de l'image échographique originale (512*512 pixels), figure 10.13.a., nous avons appliqué la méthode de chiffrement par flot avec une clef *K* de 128 bits, afin d'obtenir l'image cryptée figure 10.13.b. Si cette image est décryptée, nous pouvons noter qu'il n'y a aucune différence entre celle-ci et l'originale. La clef *K* de 128 bits a été cryptée deux fois avec l'algorithme RSA afin d'obtenir *K''*. Du fait de la longueur de la clef publique de *B*, la longueur de *K''* est proche de 1 024 bits. Ensuite avec une méthode de marquage dans le domaine spatial basée sur les LSB, la clef *K''* est insérée dans l'image cryptée (figure 10.13.c.). Le facteur d'insertion

est de 1 bit tous les 256 pixels. La différence entre l'image cryptée et l'image cryptée marquée est présentée figure 10.13.d. Les pixels utilisés pour l'insertion sont visibles, le PSNR = 75.14 dB. Après décryptage de l'image cryptée et marquée (figure 10.13.c.), nous obtenons l'image finale illustrée (figure 10.13.e.) La différence entre l'image originale et l'image finale est présentée dans la figure 10.13.f. Cette figure montre que les différences entre les deux images (PSNR = 55.28 dB) ont été diffusées dans toute l'image. Cependant, du fait que la valeur moyenne des coefficients $\alpha(i)$ est égale à zéro l'erreur due au marquage n'est pas amplifiée durant la phase de décryptage.

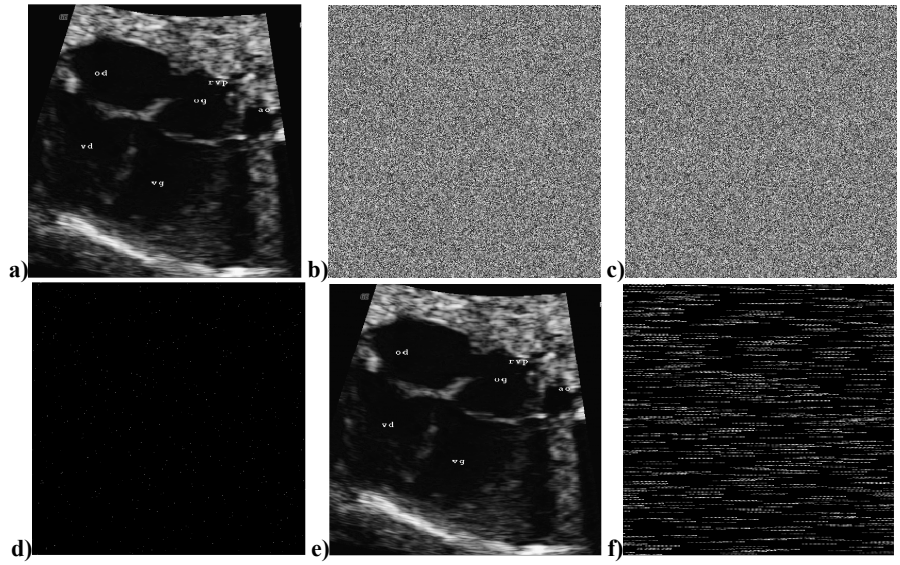

Figure 10.13. a) *Image originale*. **b)** *Image cryptée par flot avec une clef de 128 bits,* **c)** *Image* **b)** *marquée avec la clef secrète cryptée.* **d)** *Différence entre les images* **b)** *et* **c)**, **e)** *Décryptage de l'image* **c)**, **f)** *Différence entre l'image originale image* **a)** *et* **e)**

Afin de comparer les résultats de cette méthode hybride la méthode de marquage a été appliquée sur l'image médicale cryptée en utilisant l'algorithme AES avec les modes ECB et OFB (chiffrement par flot). Après décryptage de l'image marquée et chiffrée par AES en mode ECB l'image obtenue présente des variations très fortes par rapport à l'image originale (PSNR = 14.81 dB). Après décryptage de l'image marquée et chiffrée par AES en mode OFB l'image obtenue présente des variations qui n'ont pas été diffusées par ce mode. La qualité de l'image finale est bonne (PSNR = 52.81 dB) mais un problème de débordement subsiste avec le mode OFB d'AES. En effet des pixels noirs deviennent blancs et des pixels blancs deviennent noirs. En conclusion, la combinaison du cryptage et du marquage permet un système de transmission autonome et garantit l'intégrité des données.

10.6. Conclusion

Dans ce chapitre nous avons montré qu'il existait de nombreuses solutions afin de sécuriser la transmission et le stockage d'images médicales. Actuellement, en pratique les quelques solutions proposées afin de sécuriser les données médicales s'appuient sur des approches de protection très classiques. Ces approches anciennes nécessitent, soit la mise en place de dispositifs particuliers, soit un temps d'exécution relativement plus important. Ces approches classiques ne sont pas envisageables pour des utilisations en temps réel ou pour des accès depuis le cabinet d'un médecin. Certaines solutions présentées dans ce chapitre s'intégreront dans des systèmes de transmission d'images médicales à condition d'avoir prouvé leur robustesse. L'avantage principal de toutes ces approches hybrides est de pouvoir associer dans un même algorithme plusieurs types de codage. Dans les années futures, l'apparition de standards en cryptage et marquage d'images profiteront pleinement à la transmission sécurisée de données médicales.

10.7. Bibliographie

[AES 01] AES. Announcing the Advanced Encryption Standard. Federal Information Processing Standards Publication, 2001.

[ALL 94] ALLAËERT F.A, DUSSERRE L., « Security of health system in France. What we do will no longer be different from what we tell », *International Journal of BioMedical Computing*, vol. 1, p. 201-204, 1994.

[CAV 04] CAVARO-MÉNARD C., AMIARD S., « Reversible data embedding for integrity control and authentication of medical images », *In ISIVC'04, Proceedings of 2nd International Symposium on Image/Video Communications*, Brest, juillet 2004.

[COA 00] COATRIEUX G., MAITRE H., SANKUR B., ROLLAND Y., COLLOREC. R. « Relevance of Watermarking in medical imaging », *In ITAB'00, Proceedings of ITAB*, Washington, Etats-Unis, novembre 2000.

[COA 01] COATRIEUX G., SANKUR B., MAÎTRE H., « Strict integrity control of biomedical images », *In SPIE, Proc. Electronic Imaging, Security and Watermarking of Multimedia Contents*, p. 229-240, San Jose, Etats-unis, novembre 2001.

[COA 03] COATRIEUX G., H. MAITRE, « Images médicales, sécurité et tatouage », *Annales des Télécommunications, Numéro Spécial Santé*, vol. 58, p. 782-800, 2003.

[COA 05] COATRIEUX G., LAMARD M., DACCACHE, PUENTES W.J., ROUX. C., « A low distortion and reversible watermark: Application to angiographic images of the retina ». *In EMBC'05, Proceedings of Int. Conf. of the IEEE-EMBS*, p. 2224-2227, Shangaï, Chine, novembre 2005.

[COA 06] COATRIEUX G., PUENTES J., LECORNU L., CHEZE LE REST C., ROUX. C., « Compliant secured specialized electronic patient record platform », *In D2H2'00, Proceedings of D2H2*, Washington, Etats-Unis, novembre 2006.

[COS 03] COSTA M.H.M., « Writing on dirty paper », *IEEE Trans. on Information Theory*, vol. 58, p. 782-800, 2003.

[DAE 02] DAEMEN J., RIJMEN. V., AES, Proposal: The Rijndael Block Cipher. Technical report, Proton World Int.l, Katholieke Universiteit Leuven, ESAT-COSIC, Belgique, 2002.

[DIF 76] DIFFIE W., HELLMAN M.E., « New directions in cryptography ». *IEEE Transactions on Information Theory*, vol. 26, n°6, p. 644-654, 1976.

[DRO 02] VAN DROOGENBROECK M., BENEDETT R., « Techniques for a selective encryption of uncompressed and compressed images ». *In Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS) 2002*, Ghent, Belgique, septembre. 2002.

[DUC 96] DUCROT H., Le dosssier médical informatisé face à la Loi Française. Informatique et Santé : Aspects Déontologiques, Juridiques et de Santé Publique, vol. 8, p. 87-96, 1996.

[DUS 97] DUSSERE L., Recommandations déontologiques pour le choix de logiciels destinés aux cabinets médicaux. Ordre national des médecins, Conseil National de l'Ordre, Ethique et Déontologie, 1997.

[FIS 04] FISCH M.M., STGNER H., UHL A., « Layered encryption techniques for DCT-coded visual data », *In European Signal Processing Conference (EUSIPCO) 2004*, Vienne, Autriche, septembre, 2004.

[GUI 02] GUILLEM-LESSARD S., www.uqtr.ca/~delisle/Crypto. visité en 2002.

[KER 83] KERCKHOFFS A., « La cryptographie militaire », *Journal des sciences militaires*, vol. 9, p. 5-38, 1883.

[MAN 01] MANICCAM S.S., BOURBAKIS N.G., « Lossless image compression and encryption using SCAN », *Pattern Recognition*, vol. 34, p. 1229-1245, 2001.

[MAN 04] MANICCAM S.S., BOURBAKIS N.G., « Lossless compression and information hiding in images », *Pattern Recognition*, vol. 37, p. 475-486, 2004.

[NOR 03] NORCEN R., PODESSER M., POMMER A., SCHMIDT H.P., UHL A., « Confidential storage and transmission of medical image data », *Computers in Biology and Medicine*, vol. 33, p. 277-292, 2003.

[DAV 04] DAVOINE F., PATEUX S., Tatouage de documents audiovisuels numériques. HERMES, Lavoisier, Vuibert, Paris, 2004.

[PIV 05] PIVA A., BARNI M., BARTOLINI F., DE ROSA A., « Data Hiding Technologies for Digital Radiography », *IEEE Vision, Image and Signal Processing*, vol. 152, n°5, p. 604-610, 2005.

[PUE 01a] PUECH W., DUMAS M., « Transfert sécurisé d'images par combinaison de techniques de cryptographie et de tatouage », *In Proc. 7th Colloque Compression et Représentation des Signaux Audiovisuels, CORESA'01*, Dijon, France, novembre, 2001.

[PUE 01b] PUECH W., DUMAS M., BORIE J.C., PUECH M., « Tatouage d'images cryptées pour l'aide au Télédiagnostic », *In Proc. 18th. Colloque Traitement du Signal et des Images, GRETSI'01*, Toulouse, France, septembre. 2001.

[PUE 01c] PUECH W., PUECH M., DUMAS M., « Accés sécurisé distance d'images médicales haute résolution », *In Proc. 11th. Forum des Jeunes Chercheurs en Génie Biologique et Médical*, Compiègne, France, p. 72-73, juin. 2001.

[PUE 04] PUECH W., RODRIGUES J.M., « A new crypto-watermarking method for medical images safe transfer », *In EUSIPCO'04*, Vienne, Autriche, 2004.

[PUE 05] PUECH W.,. RODRIGUES J.M., « Crypto-Compression of medical images by selective encryption of DCT », *In EUSIPCO'05*, Antalya, Turquie, septembre, 2005.

[PUE 06] PUECH W., RODRIGUES J.M., DEVELAY-MORICE J.E., « Transfert sécurisé d'images médicales par codage conjoint : cryptage sélectif par AES en mode par flot et compression JPEG », *Traitement du signal (TS), numéro spécial Traitement du signal appliqué à la cancérologie*, vol. 23, n°5, 2006.

[ROD 06] RODRIGUES J.M., PUECH W., BORS A.G., « A selective encryption for heterogenous color JPEG images based on VLC and AES stream cipher », *In CGIV'06,* Leeds, Royaume-Uni, 2006.

[SAI 05] SAID A., « Measuring the strength of partial encryption scheme », *In ICIP 2005, IEEE International Conference in Image Processing*, Gènes, Italie, vol. 2, p. 1126-1129, 2005.

[SHA 78] SHAMIR A., RIVEST R.L., ADLEMAN L., « A method for obtaining digital signatures and public-key cryptosystems », *Communications of the ACM*, vol. 21, n°2, p. 120-126, 1978.

[SCH 97] SCHNEIER B., « Cryptographie appliquée : protocoles, algorithmes et codes sources en C ». Wiley, 1997.

[STI 96] STINSON D., Cryptographie - Théorie et pratique, Thompson Publishing, 1996.

[TAN 96] TANG L., « Methods for encrypting and decrypting MPEG video data efficiently », *In ACM Multimedia*, p. 219-229, 1996.