Nouveau Schema de crypto-compression des images médicales

Mohamed Salim BOUHLEL, Moez ABDELMOULA,
Mourad ELLOUMI, Lotfi KAMOUN
Laboratoire d'Electronique et des Technologies de l'Information (LETI)
Ecole Nationale d'Ingénieurs de Sfax; B.P.W, 3038 Sfax, Tunisie
E-mail :medsalim.bouhlel@enis.rnu.tn
abd.moez@voila.fr mourad.elloumi@ipeis.rnu.tn
lotfi.kamoun@enis.rnu.tn

Introduction

Le développement remarquable des technologies de l'information et des télécommunications durant ces dernières années a engendré une évolution considérable dans le domaine de la Télémédecine. Le télédiagnostic, qui est actuellement parmi les secteurs potentiels en Télémédecine, est une discipline qui permet à deux ou plusieurs équipes médicales d'échanger des images médicales et de les commenter dans une démarche d'aide au diagnostic. Elle permet aussi d'apporter à un médecin qui se trouve éloigné des grands centres une aide à la décision par des spécialistes facilitant ainsi l'accès aux soins de proximités. Ceci aura pour conséquence l'amélioration de la qualité des soins et l'actualisation des compétences et des pratiques professionnelles.

L'efficacité de ce type d'applications dépend essentiellement de deux critères fondamentaux qui sont, le degré de sécurité et le temps exigé pour la transmission, le stockage et la consultation de ces images.

Dans ce papier, on présente une nouvelle approche répondant à ces deux exigences. Elle assure conjointement le cryptage et la compression (Crypto-Compression) de ce type d'images.

1.Position du problème

Dans les applications médicales, la taille des images numérisées est très importante. Donc, on doit les comprimer afin d'améliorer la capacité de stockage et de réduire le temps de transmission à travers les réseaux (rapidité dans la transmission et diminution de l'encombrement dans les réseaux).

En plus, le cryptage des images médicales s'impose afin d'assurer la confidentialité de ces données durant leur stockage et leur transfert sur réseau.

Pour satisfaire à ces deux conditions, l'approche classique (cf.figure 1) consiste à appliquer un algorithme de cryptage indépendant sur les données après l'étape de compression [1].

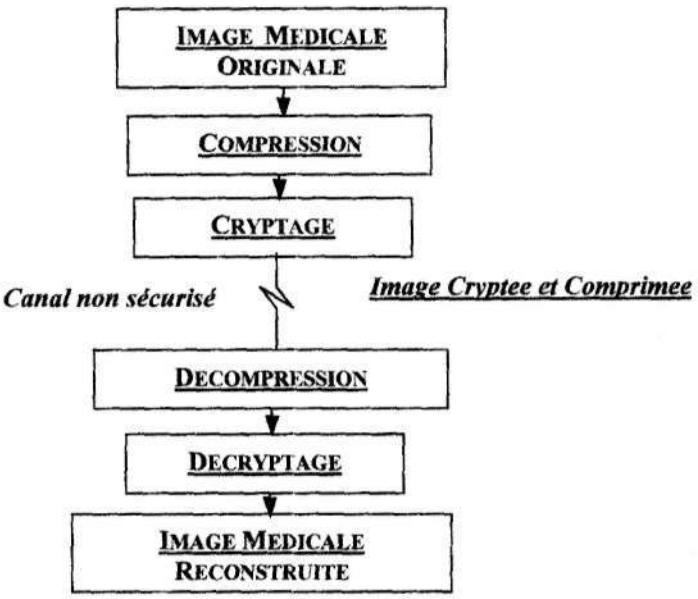


Figure 1: *Approche Classique*

Cette approche est justifiée car l'application d'un algorithme de compression sur des données, préalablement cryptées, ne peut être appropriée. En effet, le cryptage élimine toute redondance spatiale se trouvant dans la structure des données de l'image réduisant ainsi la qualité de compression. De plus, l'altération des données, cryptées, par une compression non conservative rend le décryptage impossible.

On signale que l'approche classique n'est pas très efficace. Elle exige un temps de cryptage (et de décryptage) relativement important par rapport à un cryptage (et un décryptage) visant les composantes les plus significatives de l'image. En effet de telles méthodes combinant les opérations de compression et de cryptage se développent [2-6].

Nous partageons l'avis de Chen[7] pour signaler que ces méthodes sont soit non sécurisantes soit très exigeantes en terme de temps de calcul. En exploitant les particularités que présentent les images médicales, on propose dans ce papier une approche qui associe à l'opération de compression basée sur la TCD, un niveau de cryptage dépendant du degré de sécurité voulu (cf.figure 2).

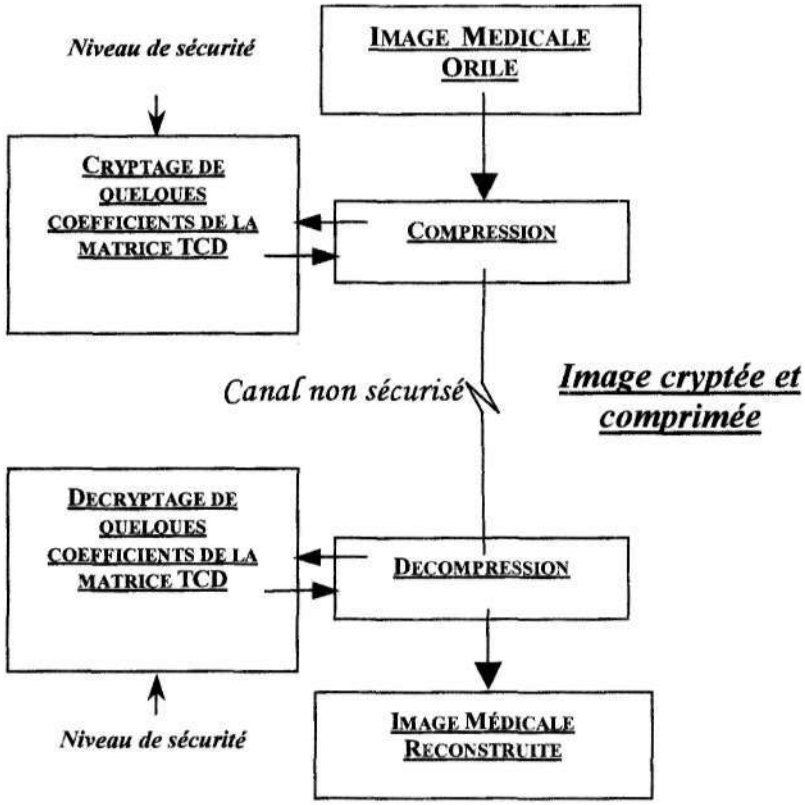


Figure 2: *Notre Approche*

2. Schéma de principe

La découverte de la transformée en cosinus discret en 1974 [8] fut un exploit très important pour la communauté scientifique du fait qu'elle permet la décorrélation entre les différents pixels en passant du domaine spatial au domaine fréquentiel. Le JPEG (Joint Photographic Experts Group) fut la première norme de compression d'images basé sur la TCD. Elle a été retenue par le CCITT en 1990 [9]. Dans le processus de compression JPEG, l'image originale est décomposée en blocs de 8*8 pixels. Chaque bloc subit la transformée en cosinus discret bidimensionnelle (TCD-2D) et génére à la sortie un autre bloc de 8*8 pixels.

Chacun des coefficients des blocs (8*8) obtenu, est ensuite quantifié en utilisant la valeur correspondante dans une table de quantification. La table de quantification est choisis en fonction de la qualité de restitution voulu, et selon les caractéristiques particulières des images. Les coefficients quantifiés sont ensuite traitées par un processus de codage entropique.

Dans la matrice TCD, la majorité des informations concernant l'image se trouve dans les coefficients représentant la partie basse fréquence. Notre approche consiste à profiter de cette particularité dans l'opération de cryptage.

En se basant sur le processus de compression de la norme JPEG, la figure 3.

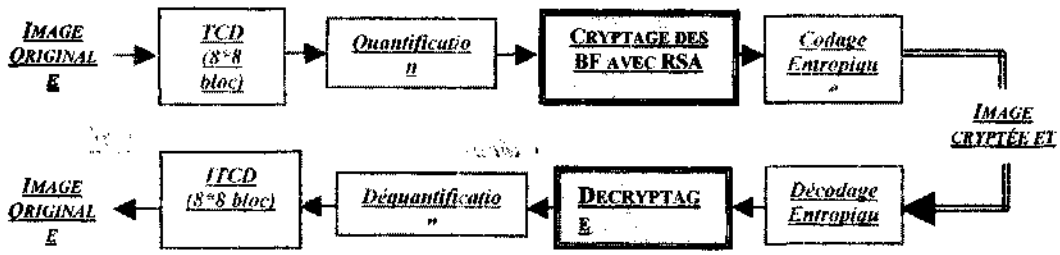


Figure 3 : Schéma de Principe de Notre Approche

illustre notre schéma de principe qui consiste à effectuer un cryptage après l'étape de quantification et juste avant l'étape de codage. Pour restituer l'information de départ, on décode d'abord les coefficients quantifiés de la matrice TCD par le décodeur entropique. Ensuite, on les décrypte avant l'étape de déquantification. Enfin, on applique la ITCD (inverse de la TCD) pour restituer l'image originale.

Les principaux avantages de notre schéma sont la flexibilité et la réduction du temps de traitement lors des opérations de cryptage et de décryptage. En effet, dans notre approche, on peut faire varier le temps de traitement suivant le degré de sécurité voulu selon des critères qu'on développera dans la partie (3.2).

3.Developpement de NOTRE Approche

Dans cette partie, on présente tout d'abord l'algorithme de cryptage adopté, on définira par la suite la Transformée en Cosinus Discret afin de pouvoir développer notre approche, et on détaillera enfin les différents avantages de notre schéma.

3.1 Définitions :

3.1.1 L'Algorithme de Cryptage RSA

L'algorithme RSA [10] est un algorithme de cryptage asymétrique qui a été baptisé du nom de ses inventeurs R. Rivest, A. Shamir et L. Adleman.

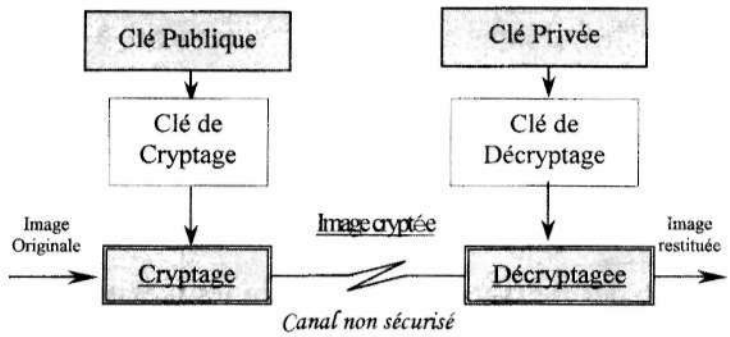


Figure 4 : *Principe du Cryptage Asymétrique*

Il utilise deux clés distinctes : une clé publique pour le chiffrement du document et une clé privée (secrète) pour l'opération inverse (cf.figure 4). La clé publique, peut être communiquée librement sur un canal non sécurisé à tout correspondant susceptible d'envoyer des documents chiffrés au détenteur de la clé privée. Par contre, la clé privée doit rester confidentielle. Sans cette dernière, il est " théoriquement" impossible de déchiffrer les données cryptées avec la clé publique [11].

La sécurité de cet algorithme repose sur la difficulté de factoriser les grands nombres. Quoique n'ayant jamais été défini comme norme internationale, ce procédé est aujourd'hui considéré comme une norme de fait, notamment en raison de sa grande diffusion.

Pour expliquer le fonctionnement de l'algorithme RSA, on suppose que la personne A veut envoyer une donnée cryptée à la personne B

Le cryptage et le décryptage avec l'algorithme RSA nécessite la générations de deux clés : une publique, et l'autre privée. Pour cela, A doit générer deux nombres premiers grands distincts p et q, puis, il doit déterminer n et $phi(n)$ tel que [12] :

$n=p.q$ et $phi(n)=(p-1).(q-1)$

Ensuite, il doit choisir un entier e tel que :

- $e < phi\ (n)$

- e premier avec *phi (n)*

$\Leftrightarrow$ PGCD *(e,phi (n))=1*

Puis, il doit déterminer un entier d tel que :

- *(d.e-1)* multiple de *phi (n)*

n et e forment la Clé Publique alors que n et d forment la Clé privée ou Secrète. Le cryptage d'un nombre x ($x< n$) de taille k (bits) tel que : $2^k < n$ se fait en appliquant la fonction f_c suivante :

$$f_c : Z_n \longrightarrow Z_n$$
$$x \longrightarrow (x^e) \bmod n \ (= y)$$

Lors du décryptage des messages y on récupère les messages x avec la fonction f_d suivante :

$$f_d : Z_n \longrightarrow Z_n$$
$$y \longrightarrow (y^d) \bmod n \ (= x)$$

On rappelle que Z_n, est la classe d'entiers appartenant à l'ensemble : *{0,1,2...,n-1}* Les additions, les soustractions et les multiplications dans Z_n se font *modulo n*.

Sécurité : Le choix des clés s'effectue à partir de la formule : *(d.e-1)=a.(p-1)(q-1)* (avec a entier quelconque). Donc, le forçage de l'algorithme RSA revient principalement à déterminer la clé secrète d à partir de la clé publique e. Il suffit pour cela de connaître la valeur de *(p-1).(q-1)*. La seule façon d'obtenir *(p-1).(q-1)* à partir de n (rappelons que n est public), est de décomposer n en ses facteurs premiers p et q. De ce fait, la factorisation de n en ses facteurs premiers représentent le point faible de la RSA si la taille de la clé est faible. Mais, si la taille de la clé est suffisamment importante, la factorisation devient non plus le point faible mais le point fort dans RSA [13].

3.1.2 L'Algorithme de la Transformation en Cosinus Discret (TCD)

La transformée en cosinus discrète (TCD), utilisée dans l'algorithme JPEG, permet de transformer les pixels d'un bloc 8x8 d'une image en un autre bloc de 8x8 contenant les composantes fréquentielles correspondantes.

Cette transformée, qui est spécialement étudiée pour la compression des images, est d'autant plus efficace que les données sont corrélées.

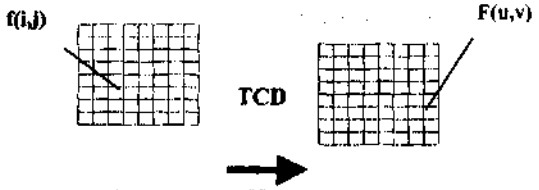
Soit f(i,j) l'intensité du pixel ayant pour coordonnée i et j, et F(u,v) la valeur du coefficient de la matrice TCD ayant pour coordonnées u et v. La valeur de F(u,v) est donnée par [8] :

$$F(u,v) = \frac{c(u)c(v)}{4} \sum_{i=0}^{7} \sum_{j=0}^{7} f(i,j) \cos\left(\frac{(2i+1)u\pi}{16}\right) \cos\left(\frac{(2j+1)v\pi}{16}\right)$$

Où : i, j, u, v = 0, 1, ..., 7

i et j sont les coordonnées dans le domaine spatial

u et v sont des coordonnées dans le domaine fréquentiel



La valeur des coefficients c*(k)* est :

$$c(k) = \begin{cases} \frac{1}{\sqrt{2}} & pour \quad k = 0 \\ 1 & pour \quad k \neq 0 \end{cases}$$

La transformée inverse de la TCD 8x8 est définie par :

$$f(i,j) = \frac{1}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} c(u)c(v) F(u,v) \cos\left(\frac{(2i+1)u\pi}{16}\right) \cos\left(\frac{(2j+1)v\pi}{16}\right)$$

3.2 Développement de notre approche

En effet, dans un bloc de 8*8 sur lequel on a appliqué la TCD, le coefficient F(0,0), appelé DC (Direct Component), est égal à la somme des 64 pixels du bloc, divisée par 8 [14]. Il représente donc une moyenne des intensités du bloc concerné. Les 63 autres coefficients qui sont appelés AC (Alternative Component), représentent les variations d'intensités entre les différents pixels du bloc. Ils caractérisent l'information liée aux détails de l'image. Parmi ces 63 coefficients, on trouve des coefficients qui caractérisent les basses fréquences et d'autres qui caractérisent les hautes fréquences [15] comme le montre la figure 5.

Vu que la variation des intensités des pixels dans un bloc 8*8 est très lente, la majorité de l'énergie se situe dans les fréquences basses. La transformée en cosinus discrète TCD permet donc de concentrer cette énergie dans quelques coefficients.

Une telle constatation est à la base de notre approche. En effet, les coefficients des hautes fréquences contribuent seulement dans les détails fins de l'image et que la majorité des informations contenues dans l'image sont concentrées dans les coefficients qui sont localisés dans la zone des basses fréquences y compris le coefficient DC. Il suffit donc de crypter cette dernière partie pour obtenir une bonne qualité de brouillage sur toute l'image.

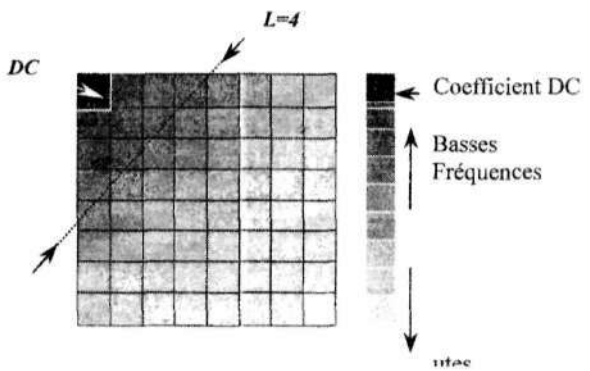


figure 5 *: Distribution des Fréquences dans la Matrice TCD*

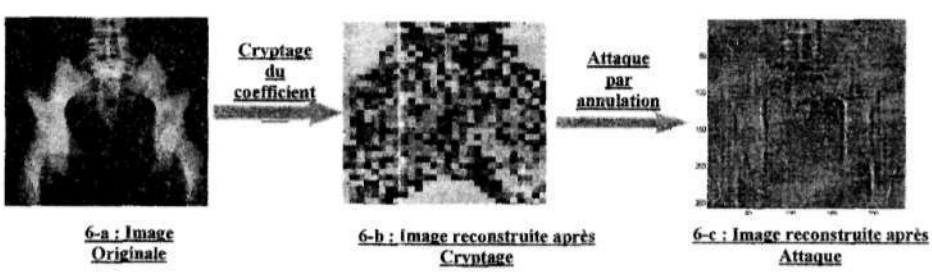


Figure 6 : Résultats du Cryptage du Coefficient DC

Si on crypte seulement le coefficient DC de la matrice TCD, on remarque que l'image obtenue est parfaitement illisible (Figure 6-b). Toutefois, il est impératif de signaler qu'il est possible d'extraire une image pouvant être significative en annulant le coefficient DC crypté (figure 6-c). Ceci constitue une éventuelle technique d'attaque à laquelle il faut parer.

Par conséquent, on a jugé qu'il est insuffisant de crypter seulement les coefficients DC. Il faut donc crypter aussi les coefficients AC significatifs pour améliorer la sécurité.

Puisque l'augmentation du nombre des coefficients à crypter engendre une diminution de la vitesse de l'opération de cryptage et de décryptage (cf.figure 8), on doit déterminer la valeur optimale L qui permet d'assurer, après attaque, une image ne contenant pas d'information utile (cf.figure 5). L étant un entier compris entre 1 et 15.

Soit Ci,j le coefficient de la matrice TCD qui est localisé dans la $i^{éme}$ ligne et la $j^{éme}$ colonne (avec i et j deux entiers compris entre 1 et 8). On propose d'annuler les coefficients Ci,j qui vérifient $i+j < L$ pour différentes valeurs de L (cf.figure 7).

A partir des résultats illustrés dans cette figure, on remarque que plus le niveau L se rapproche du coefficient DC, plus l'importance des informations pouvant être extraites est grande.

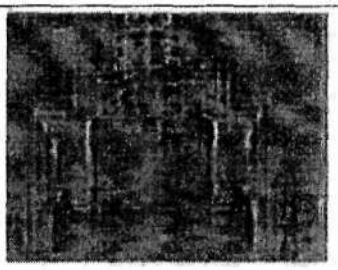
Image reconstruite après attaques	Niveau de L
	L=1
	L=2

Figure 7 : *Influence du Niveau L sur l'Image Restituée Après Attaque*

Par conséquent, et pour garantir un bon niveau de sécurité, on a intérêt à éloigner le plus possible la diagonale "L" du coefficient DC. Mais cela entraînera évidemment un temps de traitement plus important. La figure 8 représente l'influence du niveau L sur le temps de traitement relatif qui représente le rapport du temps mis pour crypter les coefficients appartenant au niveau L par le temps nécessaire pour crypter un seul pixel (niveau L=1).En effet ce temps est proportionnel aux nombre de coefficients de la TCD à crypter.

Choix du niveau de cryptage :

Suite à la consultation de deux spécialistes dans le domaine de la radiographie, on a tiré la conclusion suivante : sur un total de 150 images médicales et à partir du niveau 2, toutes les images reconstruites après attaque ne sont plus interprétables. Le tableau suivant détaille le nombre d'images devenues non interprétables en fonction du niveau L.

Tableau 1 : *Pourcentage d'images protéges en fonction de L*

Niveau de L	Nombre d'images protégées	pourcentage d'images protéges
1	127	84,7%
2	150	100%

Nous pensons que si cette constatation(L=2) peut être généralisée à l'ensemble des images médicales, elle ne peut l'être pour tous les types d'images ni à tous les domaines d'application. Elle ne peut l'être que moyenant le choix judicieux de L selon le choix du niveau de sécurité selon l'application envisagée. Nous estimons toutefois,suite a une étude éffectuée sur 80 image de types différents, que la valeur de L restera inférieur à 5.

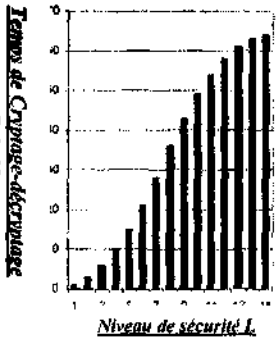


Figure 8 : Influence du Niveau L sur le Temps de Traitement

3.3 Les Avantages de notre schéma

Notre schéma est flexible. En effet, on peut, selon la vitesse de traitement tolérée, choisir le niveau L qui maximise la sécurité.

Notre schéma préserve, en grande partie, les performances de la norme JPEG en terme de taux de compression, parce que le cryptage peut se faire seulement sur les coefficients de la basse fréquence. En effet, le taux de compression dépend, principalement, du nombre de coefficients des hautes fréquences qui sont nuls après l'étape de la quantification.

Notre schéma est sécurisé du fait qu'on utilise un cryptosystème sûr (RSA) pour le cryptage. La sécurité de RSA est généralement équivalente au problème de la factorisation qui est considéré comme un problème très complexe. En effet, le dernier record de factorisation publié a été réalisé le 22 août 1999 par l'équipe de l'Institut National de Recherche en Mathématiques et en Science Informatique d'Amsterdam [16]. Cette équipe a cassé une clé RSA à 512 bits (clé de 155 chiffres décimaux) en trois mois et demie. Cette opération a nécessité trois cents ordinateurs qui fonctionnent en réseaux cumulant ainsi un total de calcul qui a été évalué à 8000 Mips année. Sachant que 1 Mips = un million d'instructions par seconde et que l'opération a durée trois mois et demie, on en déduit qu'il a fallu 72576000 milliards instructions pour casser une clé RSA a 512 bits (ce qui est irréalisable dans la pratique avec un ou même plusieurs PC classiques). Mais avec l'augmentation exponentielle des puissances de calcul qu'on peut envisager avec les nouvelles machines, nous devons être prudent dans le choix de la

dimension de la clé. Dans le futur proche, le dimensionnement de la clé dans la gamme de 1024 à 2048 bits nous paraît inévitable

Dans notre schéma, l'algorithme de cryptage RSA adopté est adapté à la transmission des images médicales en télédiagnostic. En effet, Le premier avantage de l'algorithme RSA est qu'il est asymétrique, c'est à dire qu'il ne nécessite pas la transmission de la clé de l'expéditeur au destinataire d'une façon sécurisée comme pour les algorithmes symétriques dans lesquels la même clé sert à la fois au cryptage et au décryptage [17]. En effet, pour qu'une personne puisse transmettre une donnée d'une façon confidentielle, il lui suffit tout simplement de procurer la clé publique du destinataire et de crypter les données avec. De cette façon, il est sûr que seul le destinataire pourrait décrypter les données qu'il a envoyé puisque le destinataire est le seul à posséder la clé privée [11].

Le deuxième avantage réside dans le fait que la technique RSA permet d'authentifier les données à envoyer (signature numérique). Ceci revient à la réversibilité des fonctions des deux clés publiques et privée puisqu'ils peuvent servir en même temps pour le cryptage et le décryptage [18].

Dans le cas du télédiagnostic, on peut exploiter ces avantages pour sécuriser et authentifier la transmission des images médicales. En effet, On suppose que un docteur X veut envoyer une image médicale à un docteur Y de façon à s'assurer que seul Y va pouvoir lire l'image et de façon à ce qu'il garantie à Y que c'est bien lui qui a envoyé cette image. Dans ce cas, X doit utiliser d'abord sa propre clé privée (clé privée de X) pour crypter son message dans une première phase, puis il doit crypter l'image résultat par la clé publique de Y pour enfin l'envoyer à Y. Ce dernier devra d'abord décrypter l'image reçu avec sa clé privée personnelle (clé privée de Y) dans une première phase, puis Y devra utiliser la clé publique de X pour décrypter le résultat de la phase précédente et obtenir l'image médicale que X lui a envoyé. Ceci permet d'assurer aussi bien la confidentialité de l'image transmise que l'authentification de l'expéditeur [19] (cf.figure 9).

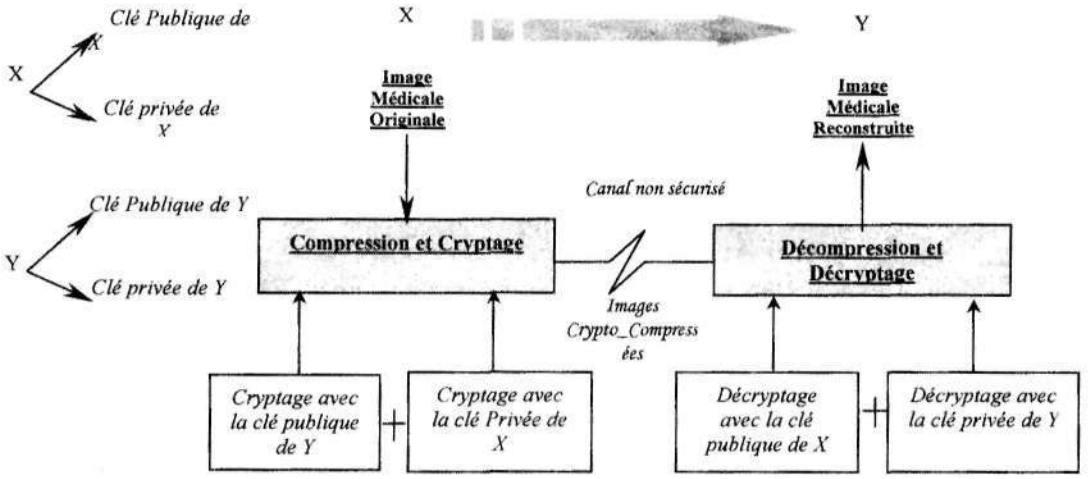


Figure 9 : *Schéma de Transmission Sécurisée et Authentifiée D'Images Médicales en Télédiagnostic*

Conclusion

Dans ce travail on a présenté un schéma efficace de crypto-compression destiné aux images médicales dans lequel on a développé une nouvelle approche concernant l'intégration du cryptage à l'intérieur des algorithmes de compression basés sur la TCD. Ce schéma est très bien adapté au secteur du télédiagnostic. Il permet de réduire le temps de cryptage et de décryptage (pouvant atteindre 1 à 2 sur 64) d'une façon flexible en fonction du niveau de sécurité exigé, améliorant ainsi la gestion et la vitesse dans la transmission des images médicales d'une part, et la capacité de stockage d'autre part. Notre approche possède l'avantage d'être applicable dans plusieurs secteurs d'activité liés à la télémédecine comme la télé expertise et la téléconsultation.

Bibliographies

[1] M.S.Bouhlel, M. Abdelmoula, M. Elloumi et L.Kamoun."Implémentation de la technique RSA pour la sécurisation des dossiers médicaux" JTEA 2002, Deuxième Conférence Internationale des Journée Tunisiennes d'Electrotechniques et d'Automatique, 21-23 Mars 2002, Sousse Nord, Tunisie.

[2] N. Bourbakis and C. Alexopoulos. "Picture data encryption using scan patterns" Pattern Recognition, 567-581, 1992.

[3] H. Cheng and X. Li "A linear quadtree compression scheme for image encryption" Signal Processing: Image Communication, 279-290, Sep 1997.

[4] D. Jones. "Applications of splay trees to data compression" Commun. ACM, 996-1007,Aug 1988.

[5] X. Li, J. Knipe, and H. Cheng. "Image compression and encryption using tree structures" Pattern Recognition Letters, 1253-1259, Nov 1997.

[6] Y. Matias and A. Shamir. "A video scrambling technique based on spacelling curves" In CRYPTO '87, 398-417, 1988.

[7] H. Cheng and X. "Partial Encryption of Compressed Images and Videos". IEEE Transactions on Signal Processing, 48(8), 2439-2451, 2000.

[8] M.Bousselmi, "Etude de l'implémentation matérielle de la chaîne de compression d'image par TCD", Thèse de Doctorat d'université, ENIS, 2001

[9] W.Pennebacker et J-L.Mitchell "JPEG Still image compression standard ", Van Nostrand Rheinhold, NY, 1993

[10] D. Bleichenbacher, B. Kaliski, and J. Staddon "Recent Results on PKCS#1: RSA Encryption Standard" RSA Laboratories Bulletin, 24 June 1998

[11] RSA Laboratories "Frequently Asked Questions about to day's Cryptography FAQ" RSA Security Inc, version 4.1, May 2000

[12] M. Shand and J. Vuillemin, "Fast implementations of RSA cryptography", Proceedings of the 11th IEEE Symposium on Computer Arithmetic, IEEE Computer Society Press (1993)

[13] Public Key Cryptography Standards (PKCS), No. 1, "RSA Encryption standard" RSA Data Security, Inc. Défi de Clef de Secret 1997

[14] D. Boneh "Twenty years of attacks on the RSA cryptosystem" Notices of the American Mathematical Society, February 1999

[15] J. Kelsey, B. Schneier, and D. Wagner, "Key-Schedule Cryptanalysis of 3-WAY, IDEA, G-DES, RC4, SAFER, and Triple-DES", Advances in Cryptology CRYPTO '96 Proceedings, Springer-Verlag (1996), 237-251.

[16] T. Eude, H. Cherifi et R. Grisel, "Distribution statistique des coefficients TCD, application à la compression", In actes des journées de statistiques Ass.Stat.Ut, Neuchâtel, May 1994

[17] S. Cavallar, B. Dodson, A.K. Lenstra, W. Lioen, P.L. Montgomery, B. Murphy, H. Riele, K. Aardal, J. Gilchrist, G. Guillerm, P. Leyland, J. Marchand, F. Morain, A. Muffett, C. Putnam, C. Putnam, and P. Zimmermann "Factorization of a 512-bit RSA modulus" Eurocrypt 2000

[18] RSA Laboratories "PKCS #1: RSA Encryption Standard" RSA Laboratories Technical Note, v.1.5, 1993

[19] R. Cramer, V. Shoup "Signature schemes based on the strong RSA assumption" Proc. 6th ACM Conference on Computer and Communications Security, 1999

[20] E. Brickell, D. Pointcheval, S. Vaudenay, and M. Yung "Design validations for discrete logarithm based signature schemes" In Proc. of PKC '2000, volume 1751 of Lncs, 276-292. Springer-Verlag, 2000

Biographie

Prof. Mohamed-Salim BOUHLEL est né à Sfax (Tunisie) en Décembre 1955. Il a obtenu son Diplôme d'Ingénieur (Ingénieur Principal en Génie Electrique) de l'Ecole Nationale d'Ingénieurs de Sfax en 1981, le Diplôme d'Etudes Approfondies en Automatique de l'Institut Nationale des Sciences Appliquées (INSA) de Lyon en 1981, le Diplôme de Docteur Ingénieur de l'Institut Nationale des Sciences Appliquées de Lyon en 1983. Il est le directeur de l'Equipe des Technologies de l'Image et des Télécommunications au sein du Laboratoire d'Electronique et des Technologies de l'Information. Il enseigne depuis 1985 à l'Ecole Nationale d'Ingénieurs de Sfax (ENIS). Actuellement, il enseigne le traitement d'images, les systèmes électroniques et les circuits de Télécommunications. Il a reçu en 1999 la médaille d'or avec la mention spéciale du jury dans le Meeting Internationale d'Invention, d'Innovation et des Technologies, organisée par le East-West EURO INTELLECT à Dubai. Il a été le vice président de l'ASET. Il est actuellement le Vice Président de l'Association Tunisienne des spécialistes en Imagerie.