REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Rech

Université de Jijel
Faculté des Sciences et de la Technologie
Département d'Automatique

Laboratoire d'Automatique de Jijel -LAJ-

Mémoire

de Projet de Fin d'Etudes pour l'obtention du diplôme de Master en

Automatique et informatique industrielle

Thème

Hachage perceptuel des images numériques Application aux images médicales

Présenté par :

Mr Boussayoud Tahar & Bouridah Mohammed Salah

Soutenu le 23/06/2014 devant le jury compose de :

Mr : T. Bouden	MCA	Président
Mme : S. Bouatmane	MCB	Examinateur
Mme : S. Biad	MAA	Encadreur

Promotion 2014

Merci à dieu :

<< On ne peut ni penser, ni juger

Sans que tu sois avec nous

Tu es bon seigneur

Bon sans mesure

Car tu nous laisse vivre et travailler

Laisse-nous te remercier jour et nuit

A chaque instant

Notre seigneur et notre dieu

Christophe COLOMB

Remerciement

Au terme de ce travail, il nous est agréable d'exprimer nos remerciements à tous ceux qui ont contribué de prés ou de loin à l'élaboration de ce mémoire

Nos remerciements vont tout particulièrement à Mme. Biad Souad, qui a bien voulu assurer notre encadrement. Nous lui devons une immense reconnaissance et un très grand respect.

Nos remerciements vont aussi à tous les enseignants de notre département sans exception pour son aide et son soutien moral

Dédicaces

Je dédie ce modeste travail à :

- Mes chers parents
- Mes chers frères
- Tous ce qui porte les noms de BOUSSAYOUD et YAKHLEF
- Tous mes amis sans exception

Tahar

Dédicaces

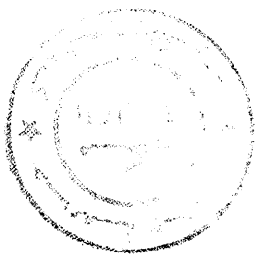
Je dédie ce modeste travail à :

- la mémoire de mon cher père
- Ma chere mère
- Mon cher frère : bilal
- Mes chères sœurs
- Mes chers neveux : zine elaabidine, norhane, ahmedhaytem, yahya anes ,douaa
- Tous ce qui porte les noms de BOURIDAH et MAKHLOUF et BOUMEHRAZ
- Tous mes amis sans exception

MOHAMMED SALAH

Sommaire

Liste des tableaux



Liste des figures et graphes

Introduction Générale

Introduction générale

Contexte général

L'importance que revêt l'imagerie médicale tient d'abord du fait qu'une image est un concentré d'information bien plus efficace qu'un texte ou qu'une explication verbale. L'interprétation des images médicales est l'un des domaines de recherche les plus encourageants, étant donné qu'il offre des facilités pour le diagnostic et les décisions thérapeutiques d'un grand nombre de maladies tel que le cancer. Avec l'évolution des maladies, plusieurs diagnostiques restent insuffisante, d'où la nécessite de la coopération de plusieurs confrères afin d'aboutir à un diagnostique correct. C'est ce qu'on appel, l'aide au diagnostic médical (télémédecine). Cette dernière technique ne cesse de prendre une place importante dans les différentes applications médicales, mais le problème majeur reste au niveau de l'échange des données, sur le réseau Internet, tout en conservant leurs intégrités ainsi que leurs confidentialités contre l'apparition considérable des pirates.

Alors et dans ces circonstances, il est devenu nécessaire d'élaborer des outils performants adaptés à ces nouvelles menaces. Malgré les mécanismes de sécurité classiques, tels que la cryptographie, qui protègent les données multimédia lors de leur acheminement, les risques de fraude, de manipulation et de piratage constituent de réelles menaces. En effet, ces données sont faciles à pirater, à modifier et à rediffuser sans aucune perte de qualité perceptible.

Dans ce contexte plusieurs solutions informatiques basées sur l'utilisation des techniques de contrôle d'accès existent. Dans le domaine de la sécurité multimédia, deux types d'approches ont été proposés pour répondre à ces exigences ces dernières années : le tatouage (watermarking) et le hachage perceptuel.

Dans ce mémoire, nous nous intéressons aux fonctions de hachage perceptuel pour l'authentification et le contrôle d'intégrité des images numériques et plus précisément les images médicales. Les fonctions de hachage perceptuel sont inspirées des fonctions de hachage cryptographique pour authentifier les données multimédia. Traditionnellement, la vérification d'intégrité des données est traitée par des fonctions de hachage cryptographique, telles que MD5 [1] et la famille SHA [2] qui sont très sensibles à chaque bit du message d'entrée. Par conséquent, l'intégrité du message est validée que lorsque chaque bit du message est inchangé [3]. Cela présente le principal inconvénient de ces techniques

cryptographiques pour authentifier les images. En effet, le problème de l'intégrité des images se pose en termes de son contenu sémantique plutôt que des valeurs de ses pixels. Pour cela, l'authentification des images devrait se baser sur leurs contenus visuels et non pas sur leurs contenus binaires. Par conséquent, pour authentifier une image, il faut tolérer des manipulations acceptables que pourrait subir une image telles que la compression JPEG, l'ajout du bruit, la rotation et le filtrage par exemple. En effet, ces manipulations préservent l'aspect visuel de l'image. En même temps, un système de hachage perceptuel doit être suffisamment fragile pour détecter les manipulations malveillantes qui modifient l'interprétation du contenu sémantique de l'image comme l'ajout de nouveaux objets, la suppression ou la modification majeure d'objets existants par exemple.

Les fonctions de hachage perceptuel sont des solutions potentielles dans ces cas-là permettant d'établir une "correspondance perceptuelle" entre l'image originale et l'image à authentifier. Ces dernières années ont vu beaucoup de chercheurs se pencher sur cette nouvelle approche de sécurité des données multimédia.

Contribution

Dans ce mémoire, nous avons proposé un schéma de hachage perceptuel pour les images numériques médicales. Un tel schéma se compose habituellement de quatre étapes essentielles ; étape de prétraitement, étape d'extraction de caractéristiques, étape de quantification et étape de crypto compression. Pour l'extraction de caractéristiques, nous nous sommes basés sur le descripteur SIFT (Scale Invariant Feature Transform) introduit par D. Lowe [13]. C'est une technique qui permet de calculer à partir de l'image, un descripteur invariant à la mise à l'échelle et aux transformations géométriques tel que la rotation. Pour la quantification on s'est basé sur la quantification uniforme où on a proposé une méthode pour la recherche du pas de quantification. Enfin pour la crypto compression nous avons utilisé l'algorithme SHA1 [2]. L'objectif principal de ce travail est de proposer une méthode de hachage perceptuel basée sur des caractéristiques extraites de l'image et sur des concepts cryptographiques. A la fin, le schéma proposé doit générer une signature:

- Courte : la signature doit être courte de l'ordre de quelques centaines de bits.
- Robuste : avoir la même signature pour des données multimédia de même contenus visuels.
- Sécurisée : impossible de générer les données originales à partir de leurs signatures et en même temps avoir des signatures totalement différentes pour des données multimédia n'ayant pas le même contenu visuel.

Organisation

Cette étude est composée des trois chapitres suivants ainsi d'une conclusion générale et des perspectives :

- Chapitre 1 : ce chapitre est dressé pour exposer le schéma général de hachage perceptuel avec ses différentes étapes ainsi que les différentes notions liées.

- Chapitre 2 : dans ce chapitre nous allons expliquer en détail les outils utilisés pour l'élaboration de l'algorithme de hachage perceptuel où nous exposons en détail l'étape d'extraction de caractéristique basée sur le descripteur SIFT et l'étape de quantification.

- Chapitre 3 : Dans ce chapitre, nous exposons les résultats d'analyse de la robustesse des signatures perceptuelles calculées à partir du processus général proposé.

Chapitre I

I.1 Introduction

Actuellement l'évolution explosive des nouvelles technologies a rendu l'échange de l'information, sous ses différentes formes (images, audio, vidéos....), à travers les réseaux publics de plus en plus facile et rapide. Le besoin alors de la sécurisation de ces données circulant à travers ces réseaux de communication est primordiale. En effet, ces données sont faciles à pirater, à modifier et à rediffuser sans aucune perte de qualité perceptible. Cela a suscité l'intérêt de développer des algorithmes et des techniques robustes pour vérifier la sécurité de la confidentialité, de l'authenticité et de l'intégrité des données multimédia échangées. Dans le domaine de la sécurité multimédia, deux types d'approches récentes ont été proposées pour répondre à ces exigences ces dernières années : le tatouage(Watermarking) et le hachage perceptuel qui constitue le thème de notre étude. Dans ce chapitre, nous allons exposer en détail la technique de hachage perceptuel ainsi que les différentes notions liées.

I.2 Différents aspects de sécurité

La protection des données numériques concerne principalement trois aspects, la confidentialité, l'intégrité et l'authenticité. Ces aspects sont assurés par différents mécanismes de sécurité tels que la cryptographie, le tatouage et le hachage perceptuel. Dans ce qui suit, nous allons définir ces différents aspects.

I.2.1 Confidentialité

La confidentialité permet de protéger le contenu des informations sauvegardées ou transmises sur un réseau. Cet aspect est assuré par la cryptographie. Assurer la confidentialité d'une information revient à la rendre limitée à certains correspondant est inintelligible pour les personnes non autorisées. En cryptographie, la confidentialité des données est obtenue principalement à travers des schémas de chiffrement basés sur une clef. Un tel système transforme un document ou une image claire en document chiffré, de telle façon que seuls les utilisateurs ayant accès à la clef soient en mesure de récupérer l'information à partir du message chiffré.

I.2.2 Intégrité

Le contrôle d'intégrité de données revient à vérifier leur contenu contre toute modification accidentelle ou intentionnelle. La vérification d'intégrité est une preuve de leur protection contre toute tentative de modification par une personne non autorisée. Le contrôle d'intégrité de données est assuré par les fonctions de hachage. Ces dernières génèrent des empreintes numériques pour chaque donnée. Ainsi, aucune personne ne peut modifier le message en conservant la même empreinte.

I.2.3 Authenticité

Le concept d'authenticité s'applique à la fois aux personnes et aux documents. On parle de l'identification de l'émetteur (une preuve de son identité) et de l'authentification des messages. La notion d'authentification est proche de celle d'intégrité. L'intégrité garantit le contenu d'un message alors que l'authentification garantit l'origine du message en prouvant son caractère authentique. Il s'agit là pour une personne de prouver qu'un message a bien été envoyé par elle. Nous citons les codes d'authentification de message (en anglais *MAC* pour *Message Authentification Code*) par clef secrète, et la signature électronique en clef publique comme les techniques les plus utilisées pour l'authentification des messages.

I.3 Définition du hachage cryptographique (HC)

Avant d'entamer le hachage perceptuel ainsi que les différentes notions liées nous devons premièrement définir le hachage cryptographique qui à l'origine de cette nouvelle approche.

En effet le hachage cryptographique est une fonction qui prend comme arguments une chaine de bits de longueur arbitraire finie (le message) et restitue en sortie une chaine de bits de longueur fixée, c'est l'empreinte. Cette opération représente une compression (la taille de l'empreinte est très inférieur à celle de la chaine initiale) et bien sure un moyen de sécurité des données.

Nous pouvons décrire une telle fonction de la façon suivante :

$h : \{0,1\}^* \longrightarrow \{0,1\}^n$

Etend donner la fonction h et l'ensemble de données x $\epsilon\{0,1\}^*$ on peut calculer efficacement h(x).

- $\{0,1\}^*$ désigne l'ensemble de chaine de longueur finie.
- $\{0,1\}^n$ désigne l'ensemble de chaine de longueur fixée.

I.4 Définition du hachage perceptuel (HP)

Les fonctions de hachage perceptuel se basent sur l'aspect visuel des données à hacher et sont fortement inspirées des fonctions de hachage cryptographique.

Un système de hachage perceptuel se compose principalement :

- Génération de la fonction de HP H : cette fonction calcule les valeurs hachées h de longueur l_h à partir des donnée multimédia (image) d'entrée noté x et une clé optionnelle k : $h=H_k(x)$. Généralement les valeurs hachées h sont notées la signature ou ce qu'on appelle une empreinte non réversible pour assurer l'intégrité et l'authenticité de l'image.
- Vérification de la fonction HP V : cette fonction compare deux signatures et renvoie une décision.

Les fonctions de hachage perceptuel doivent générer une signature :

– **Courte** : la signature doit être courte de l'ordre de quelques centaines de bits.

– **Robuste** : avoir la même signature pour des données multimédia de même contenus visuels.

– **Sécurisée** : impossible de générer les données originales à partir de leurs signatures et en même temps avoir des signatures totalement différentes pour des données multimédia n'ayant pas le même contenu visuel.

I.5 Comparaison entre le hachage perceptuel et cryptographique

Les fonctions de hachage cryptographique et les fonctions de hachage perceptuel ont les mêmes objectifs. Les deux types de fonctions de hachage vérifient l'authenticité et contrôlent l'intégrité des données à hacher. Quand une authentification d'un fichier exécutable est exigée, il est très important que toutes les valeurs des bits correspondent exactement aux valeurs originales. Dans ce cas, les fonctions de hachage cryptographique sont les plus adéquates à utiliser. Pour authentifier une donnée multimédia, il est nécessaire de

vérifier son contenu visuel sans tenir compte de sa représentation numérique. Dans ce cas, les fonctions de hachage cryptographique ne présentent pas une bonne solution. Pour cela, les fonctions de hachage perceptuel sont proposées pour satisfaire les besoins particuliers de sécurité des images numériques.

I.6 Manipulations acceptables vs. Manipulations malveillantes

Une image numérique une fois diffusée peut subir différentes formes de transformations ou de manipulations qui peuvent affecter son contenu binaire et/ou visuel. De ce faite on peut constater l'utilité du hachage perceptuel afin de garantir l'intégrité de l'image et pouvoir authentifier le document originale. Ces différentes transformations se composent en deux classes : les manipulations acceptables et malveillantes.

Certaines applications peuvent avoir besoin d'appliquer certaines manipulations acceptables afin d'améliorer la qualité de l'image originale tels que le filtrage, la compression, ou même d'effectuer d'autres opérations permettant l'amélioration de l'image en question. Certaines applications peuvent également nécessiter une compression avec pertes pour satisfaire les Contraintes de ressources sur la bande passante ou d'espace de stockage. Ces manipulations acceptables modifient uniquement les valeurs de pixels, qui se traduisent par différents niveaux de distorsion visuelle de l'image, mais le contenu de l'image, qui porte la même information visuelle vers le récepteur, est encore conservé. D'autre part, les manipulations malveillantes changent le contenu de l'image originale afin de porter une information visuelle différente pour le récepteur. Un exemple typique de modification malveillante est de remplacer certaines parties de l'image avec des contenus différents pour une utilisation malveillante. Une classification, non exhaustive, des manipulations acceptables préservant le contenu et les manipulations malveillantes changeant le contenu est présentée dans le tableau. 1 ci-dessus :

Manipulations acceptables	Manipulations malveillantes
✓ Erreurs de transmission ✓ Ajout de bruit ✓ Compression et Quantification ✓ Mise à l'échelle ✓ Réduction de résolution	✓ Suppression des objets sur l'image ✓ Déplacement des éléments de l'image pour changer leurs positions ✓ Ajout de nouveaux objets ✓ Changements des caractéristiques de

✓ Rotation ✓ Filtrage ✓ Conversion de couleurs ✓ Réglage de contraste ✓ Changements de luminosité ✓ teinte et de saturation	l'image : couleur, textures, structure, impression, etc.... ✓ Modifications du contexte de l'image : de jour ou d'emplacement ✓ Les changements de conditions d'éclairage : manipulations d'ombre ✓ Dégradation de la qualité

Tableau. I.1: Manipulations acceptables et manipulations malveillantes. [4]

I.7 Schéma générale d'un système de hachage perceptuel

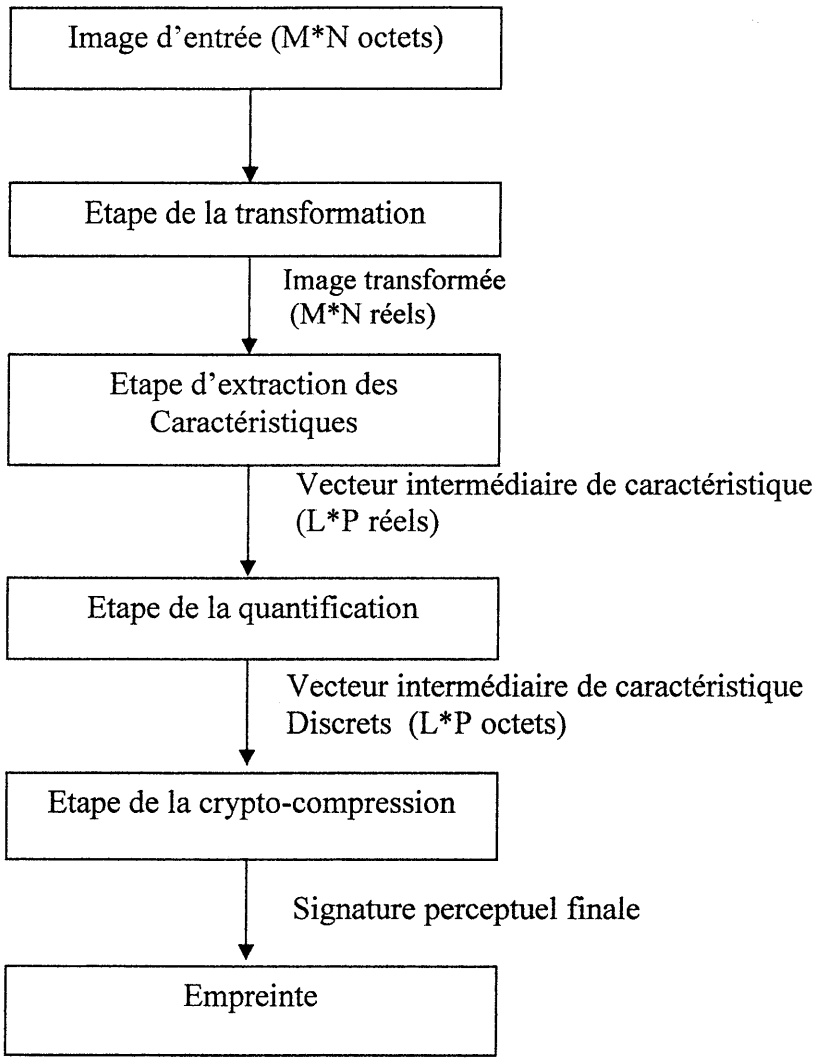


Figure I.1 Schéma général d'un système de hachage perceptuel

Le schéma général d'un système de hachage perceptuel se compose de quatre étapes élémentaires qui sont : l'étape de transformation considérée comme une étape de prétraitement suivi de l'étape d'extraction des caractéristiques. Dans cette étape on extraite les caractéristiques de l'image à partir de leur transformation avec des algorithmes bien définie comme le SIFT pour générer le vecteur de caractéristiques réelles. Ensuite on va faire la quantification sur le vecteur de caractéristique, après que la quantification est finie le nouveau vecteur doit contenir des valeurs discrètes (entiers), à la fin, l'étape de la crypto-compression va faire la sécurité du système et sa longueur finale, cette étape est assurée par des fonctions de hachage cryptographique comme l'algorithme SHA 1.

I.8 Définition de chaque étape du schéma général

I.8.1 Etape de la transformation

En effet cette étape peut être aussi considérée comme étape de prétraitement. Elle comprend les opérations typiques comme la conversion en niveau de gris, égalisation d'histogramme, filtrage de bruit, remise à l'échelle....Son objectif principal est :

- Réduction de la taille des données de la variable d'entrée (image) et par la suite la taille des variables de hachage.
- Réduction du temps de calcul.
- Filtrage de bruits et augmentation de la robustesse.

Pour la réduction du nombre des données d'entrée, l'image de départ subit une transformation spatiale (telles que la transformation de couleur, le lissage) et/ou fréquentielle (comme la transformée en cosinus discrète (DCT) ou la transformée en ondelettes (DWT)). Pour le cas de la transformée en ondelettes discrètes par exemple, la plupart des systèmes de hachage perceptuel ne prennent que la sous-bande LL en compte. En effet, la sous-bande LL est une version grossière de l'image originale et contient toutes les informations perceptuelles de l'image avec une dimension réduite à la moitié. L'objectif principal de ces transformations est de rendre toutes les caractéristiques extraites, qui sont à la base de la génération de la fonction de HP, dépendantes des valeurs de pixel ou des coefficients fréquentiels de l'image d'entrée.

I.8.2 L'extraction des caractéristiques

Dans l'étape d'extraction des caractéristiques, le système de hachage perceptuel extrait les caractéristiques de l'image à partir de l'image transformée pour générer le vecteur intermédiaire de caractéristiques réelles de L éléments, où L≪M×N. À noter que chaque caractéristique peut contenir p éléments de type réel, ce qui signifie que le vecteur de caractéristiques est composé de L × p réels à cette étape. Une autre sélection de caractéristiques peut être ajoutée à cette étape, les caractéristiques les plus pertinentes sont sélectionnées. Elles sont statistiquement plus résistantes contre certaines manipulations spécifiques tolérées, comme l'ajout de bruit, la compression JPEG et le filtrage. Les caractéristiques sélectionnées peuvent être présentées comme un vecteur intermédiaire de caractéristiques de K × p réels, où K <L. Notez que les caractéristiques visuelles sélectionnées sont généralement connues du public et peuvent donc être modifiées. Cela pourrait menacer la sécurité, du fait que la valeur de la signature perceptuelle pourrait être ajustée malicieusement pour correspondre à celle d'une autre image. Pour le cas de notre travail, nous avons utilisé l'algorithme SIFT (Scale Invariant FeatureTransform) pour l'extraction des caractéristiques. En effet ce choix est justifié par le faite que cet outil est invariant au changement d'échelle, à la rotation, aux points de vue et aux conditions d'éclairage [4].

I.8.3 La quantification

L'étape de quantification dans un système de hachage perceptuel permet la discrétisation du vecteur de hachage intermédiaire continu contenant des caractéristiques à virgule flottante dans un vecteur de hachage intermédiaire discret. Cette étape est très importante pour diminuer la taille des données à hacher. Aussi elle permet l'amélioration des propriétés de la robustesse en minimisant les probabilités de collisions dans un système de hachage perceptuel. La quantification est la manière convenable pour atteindre ce but.

Cette étape, dans un système de hachage perceptuel, est un processus difficile parce que la manière dont les caractéristiques du vecteur de hachage intermédiaire continu changent, après les manipulations tolérables, reste imprévisible dans chaque intervalle de quantification de taille Q. Cette difficulté pour assurer une correcte quantification augmente plus quand elle est suivie par une étape de crypto compression, *i.e.* SHA-1 [5].

Dans cette étape aussi, le vecteur intermédiaire de caractéristiques réelles est quantifié et peut être codé sur K×p octets. Le nouveau vecteur intermédiaire de caractéristiques contient des

valeurs discrètes. Il existe deux types de quantification : la quantification uniforme et adaptative. La quantification uniforme peut être appliquée sur chaque composant du vecteur de caractéristiques réelles pour obtenir des entiers par un simple schéma d'arrondissement. La quantification adaptative [6] est également un schéma de quantification largement utilisé dans les techniques de hachage perceptuel. La différence entre les deux techniques est que la partition des intervalles de quantification uniforme est basée sur la taille des valeurs du vecteur de caractéristiques, tandis que la partition des intervalles de quantification adaptative est basée sur la fonction de densité de probabilité des valeurs du vecteur de caractéristiques.

I.8.4 la crypto-compression

L'étape de la crypto-compression est la dernière étape d'un système de hachage perceptuel qui garantit à la fois la sécurité du système et la longueur fixe de la signature perceptuelle finale. Le vecteur intermédiaire de caractéristiques discrètes est compressé et crypté dans une courte signature perceptuelle de taille fixe de l octets, où $l \ll K \times p$, qui présente la signature perceptuelle permettant la vérification et l'authentification d'image au niveau du récepteur. Cette étape peut être assurée par les fonctions de hachage cryptographique comme, par exemple, la fonction de hachage cryptographique SHA-1 générant une signature de taille 160-bits. Dans le cadre de notre travail nous allons utiliser l'algorithme SHA-1.

I.8.4.1 L'algorithme SHA-1

SHA (Secure Hash Algorithm) : c'est la norme du gouvernement Américain pour le hachage. **Le SHA-1** est une amélioration de SHA qui produit une empreinte de 160 bits à partir d'un message de longueur maximale de 2^{64} bits.

I.8.4.1.1 Principe de fonctionnement

Le fonctionnement de L'algorithme de hachage SHA-1 est comme suite :

- il prend comme entrée un message de taille max 2^{64} bits, ensuite il ajoute à la fin de ce message un bit à 1 suivi d'une série de bits à 0, telle que la nouvelle longueur du message devient multiple de 512 bits, après il code le message en entrée sur 64 bits.

le SHA1 décompose le message en des blocs de 512 bits pour chacun ensuite il calcule 80 tours successifs et applique une série de transformations sur l'entrée. Il calcule 80 valeurs sur 32 bits :

Les 16 premières sont obtenues directement à partir du bloc. Les 64 autres sont calculées successivement, Le SHA-1 les obtient grâce à une rotation qui est appliquée sur le résultat d'un XOR, il utilise pour cela 4 mots obtenus dans les étapes précédentes. On définit ensuite 5 variables qui sont initialisées avec des constantes (spécifiées par le standard), le SHA-1 utilise encore 4 autres constantes dans ses calculs. Si un bloc de 512 bits a déjà été calculé auparavant, les variables sont initialisées avec les valeurs obtenues à la fin du calcul sur le bloc précédent, Il s'ensuit 80 tours qui alternent des rotations, des additions entre les variables et les constantes. Selon le numéro du tour, le SHA-1 utilise une des quatre fonctions booléennes. L'une de ces fonctions est appliquée sur 3 des 5 variables disponibles. Les variables sont mises à jour pour le tour suivant grâce à des permutations et une rotation. En résumé, le SHA-1 change sa méthode de calcul tous les 20 tours et utilise les sorties des tours précédents.

À la fin des 80 tours, on additionne le résultat avec le vecteur initial. Lorsque tous les blocs ont été traités, les cinq variables concaténées ($5 \times 32 = 160$ bits) représentent la signature. Sur la figure I. 2, on trouve un schéma explicatif de l'algorithme SHA-1.

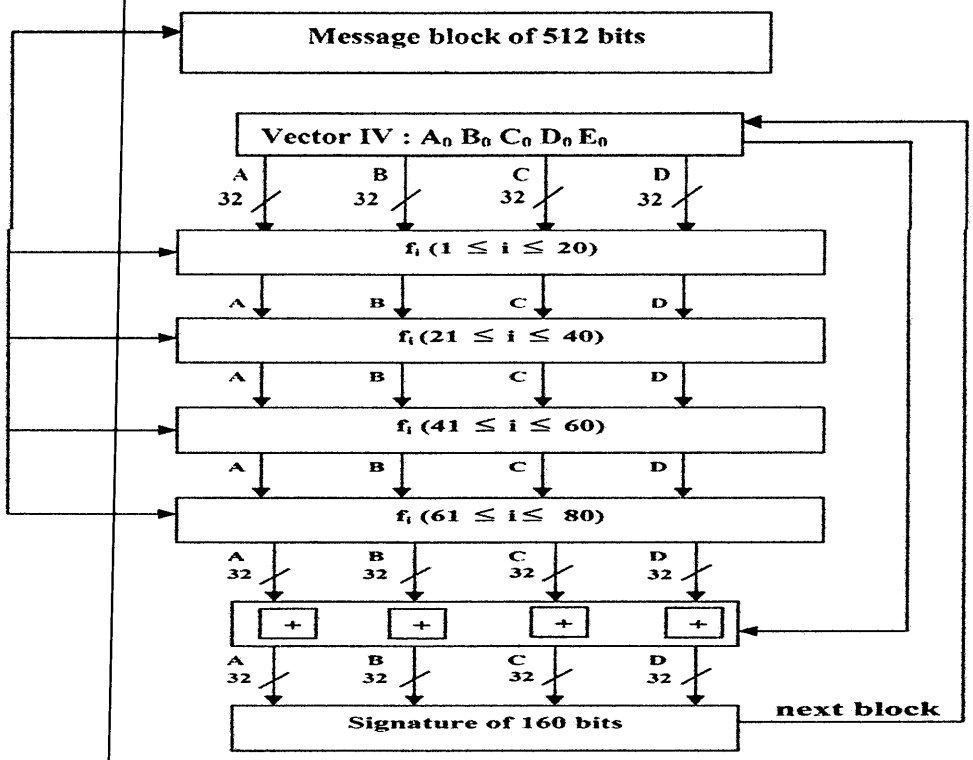


Figure I. 2: L'algorithme SHA-1. [4]

I.9 Classification des méthodes de hachage perceptuel

L'extraction de caractéristiques est l'étape la plus importante dans un système de hachage perceptuel. Ces caractéristiques doivent résister à certaines distorsions comme la compression, l'ajout de bruit..., mais très sensibles aux modifications malveillantes du contenu de l'image. La plupart des méthodes existantes de hachage perceptuel des images se concentrent principalement sur cette étape. Donc l'objectif principal pour ces méthodes est d'extraire des caractéristiques visuelles qui restent relativement constantes face à un type précis de manipulations acceptables. Ces méthodes réduisent un système de hachage perceptuel des images à un système qui se satisfait de l'extraction de quelques caractéristiques résistantes à certaines manipulations. Ces méthodes peuvent être grossièrement classés en quatre catégories [7] :

– Méthodes basées sur les statistiques d'image.
– Méthodes basées sur la relation.
– Méthodes basées sur la préservation grossière de l'image.
– Méthodes basées sur les caractéristiques bas niveaux.

C'est dans le cadre de la quatrième catégorie qu'on peut classer le l'algorithme étudié dans ce mémoire. Les caractéristiques de l'image bas niveau sont les contours ou les points caractéristiques ou points d'intérêts [8] du fait qu'elles captent l'information pertinente de l'image. L'inconvénient majeur de ces méthodes est leurs sensibilités à certaines modifications insignifiantes, comme la quantification et la réduction de résolution.

I.10 Robustesse et sécurité d'un système de hachage perceptuel

D'après ce qu'on a vu dans la section (IV), les exigences les plus importantes d'un système HP sont la robustesse et la sécurité. En effet assurer la robustesse d'un système de hachage perceptuel revient à extraire des caractéristiques visuelles robustes, par différentes techniques de traitement d'image, durant le stage d'extraction de caractéristiques comme on a discuté dans la section précédente. Dans cette direction, assurer l'extraction d'un ensemble de caractéristiques visuelles robustes qui résistent (ou qui restent relativement constantes) aux manipulations acceptables/non-malveillantes est le challenge à résoudre dans ce domaine de recherche.

Pour la deuxième exigence, l'étape clé pour garantir la sécurité du système est l'utilisation du module de crypto compression permettant d'avoir une signature perceptuelle finale de taille fixe et sécurisée (du faite qu'elle est non réversible). Quand l'étape de crypto-

compression manque dans un système de hachage perceptuel, les propriétés de sécurité du système sont menacées.

I.11 Conclusion

Les fonctions de hachage perceptuel assure d'associer à une image de taille absolue, une empreinte de taille fixe et non réversible **une empreinte de 160 bits**. L'empreinte générée représente l'image d'une manière précise et permet la simplification de détection des changements dans ce dernier. Les fonctions de hachage perceptuel sont utilisées dans des domaines où la sécurité des données traitées est critiquée pour assurer l'authentification et le contrôle d'intégrité.

Chapitre II

II.1 Introduction

Dans ce chapitre nous allons expliquer en détail les outils utilisés pour l'élaboration de l'algorithme de hachage perceptuel étudié dans ce mémoire.

Le premier concerne l'étape d'extraction de caractéristiques où nous avons utilisé le descripteur SIFT. La particularité de SIFT est de générer un grand nombre de caractéristiques recouvrant l'image par un ensemble complet d'échelles et de localisations. Par exemple une image de résolution500* 500 pixels, il est possible d'extraire environ 2000 caractéristiques stables. Chose qui permet d'améliorer d'avantage la robustesse du schéma de hachage perceptuel.

Le deuxième outil présenté dans ce chapitre concerne la troisième étape du schéma proposé dans ce travail ; c'est l'étape de quantification. Cette dernière permet la discrétisation du vecteur de hachage intermédiaire continu contenant des caractéristiques à virgule flottante dans un vecteur de hachage intermédiaire discret. Elle est très importante pour diminuer la taille des données à hacher. Aussi elle permet l'amélioration des propriétés de la robustesse en minimisant les probabilités de collisions dans un système de hachage perceptuel [4].

II.2 Description de l'algorithme SIFT

Le SIFT (Scale-Invariant Feature Transform), qui peut être traduit par -Caractérisation d'images par descripteurs locaux invariants à l'échelle- a été proposé par David Lowe en 1999 [9], puis amélioré ensuite 2004 [10]. C'est un descripteur de bas niveau largement utilisé en traitement d'image, et permet de représenter une image par un ensemble de caractéristiques locales invariantes à l'échelle. Il intègre un détecteur de points d'intérêt et un descripteur pour les points détectés. Ce descripteur est exploité pour trouver les similarités entre deux images numériques indépendamment de l'échelle.

L'étape fondamentale de la méthode est de calculer ce qu'on appelle les « descripteur SIFT »des images à étudier, il s'agit d'information numérique dérivé de l'analyse local d'une image et qui caractérise le contenu visuel de cette image de la façon la plus indépendante possible de l'échelle du cadrage, de l'angle d'observation et de l'exposition (luminosité). Ce descripteur par rapport aux détecteurs de points d'intérêts, qui existaient avant, est invariant aux changements d'illumination, d'échelle, de rotation, au bruit de l'image et aux petits changements d'angles lors de la prise d'image.

La méthode consiste en trois étapes fondamentales : la détection des points d'intérêts, le calcul des descripteurs et enfin la correspondance entre images. Dans ce qui suit nous allons présenter en détail ces étapes.

II.2.1 Détection des points d'intérêts

Les points d'intérêts SIFT sont les extrema de la pyramide de différence de gaussiennes vérifiant certaines conditions de contraste et de courbure. Dans ce qui suit nous allons voir premièrement ce qu'est la pyramide de différence de gaussiennes ou l'espace-échelle Gaussien et pourquoi ses extrema sont des points particuliers de l'image. Puis nous expliquerons comment sont détectés les points d'intérêts. Enfin nous expliquerons comment sont rejetés certains points d'intérêts à l'aide d'arguments de contraste et de courbure.

II.2.1.1 Construction de l'espace-échelle Gaussien :

L'objectif de cette étape est de réduire le niveau de détail d'une image pour ne garder que les détails grossiers et rien de plus. La construction de cet espace de différences de Gaussiennes se fait en deux temps. Tout d'abord l'image est convoluée à plusieurs filtres Gaussiens, comme suit :

$$L(x,y,\sigma) = G(x,y,\sigma)*I(x,y) \qquad \text{(II.1)}$$

Avec : $G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}}$

- I(x, y) : l'image en niveaux de gris.
- $G(x,y,\sigma)$: le filtre gaussien.
- σ : Ecart-type du filtre Gaussien et représente le facteur d'échelle de chaque filtre. Il détermine le niveau de flou : plus σ est grand plus l'image résultante de la convolution est flou.

-Le facteur d'échelle est fixé à une valeur initiale σ_0, et augmenté à chaque fois. Plus exactement, l'échelle est multipliée par un facteur k. Ce facteur est déterminé par le nombre s d'images qu'on veut obtenir par octave dans l'espace-échelle Gaussien suivant la relation : $k = 2^{1/s}$. D. Lowe recommande la valeur initiale $\sigma_0 = 1.6 * k$ [7].

- L'axe des échelles est l'ensemble $\sum = \{\sigma_0, k\sigma_0.., k^{ne-1}\sigma_0\}$ (n_e est le nombre total des octaves)

Lorsqu'on obtient la valeur $2\sigma_0$:

- Les images filtrées jusqu'à présent constituent une octave. Une octave est une zone d'échelle entre une échelle σ et l'échelle 2σ.
- Les dimensions de l'image sont réduites de moitiés, et l'algorithme est reproduit de nouveau avec cette image réduite pour obtenir une seconde octave.
- Le calcul des images filtrées et par la suite la constitution des octaves est arrêté lorsque les dimensions de l'image deviennent très petites.

A la fin de cette étape nous obtenons un ensemble d'images flouées à des échelles différentes, comme illustrées sur la figure 1 (à gauche).

Une fois, les octaves obtenues, l'étape suivante consiste à calculer les différences entre chaque deux image successive :

$$D(x, y, \sigma) = L(x, y, (k+1)\sigma) - L(x, y, k\sigma) \qquad \text{(II.2)}$$

L'ensemble des différences de Gaussienne D(x, y, σ) constitue l'espace-échelle Gaussien. Comme illustré sur la figure 1 (à droite).

L'espace échelle est donc bien un empilement de n_e fois la même image où plus la hauteur σ est grande, plus le niveau de détail est grossier.

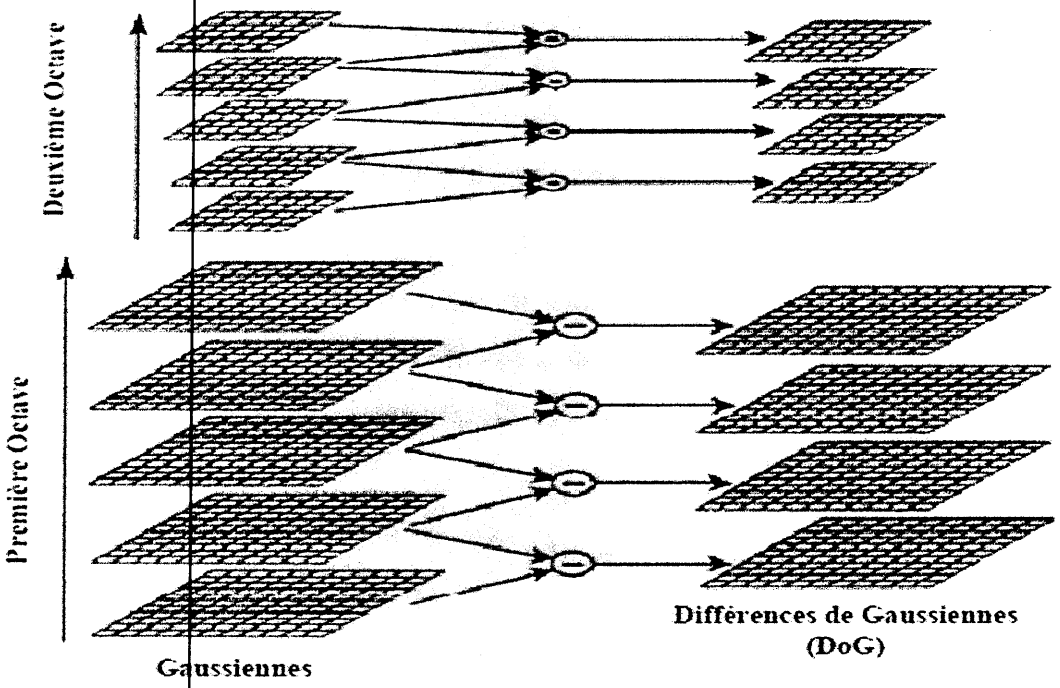


Figure II.1: Illustration de l'espace-échelle Gaussiennes et différence de Gaussiennes

II.2.1.2 Localisation des extrema locaux

Une fois l'espace des différences de Gaussiennes obtenu, il suffit de calculer les extrema locaux. Ces extrema sont sélectionnés de la manière suivante : on compare chaque pixel à ses 26 voisins dans la pyramide de différence de gaussiennes (9 dans le plan du dessous, 8 dans le même plan, 9 dans le plan du dessus. Voir figure 2). Le pixel est sélectionné seulement si c'est le maximum ou le minimum de tous ses voisins.

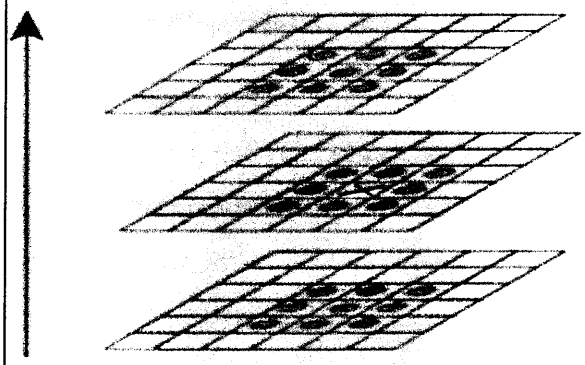


Figure II.2: Localisation des extrema aux locaux

Ces extrema sont des points clés potentiels. Les étapes suivantes vont permettre de les relocaliser avec précision et d'en éliminer un certain nombre.

II.2.1.3 Amélioration de la précision par interpolation des coordonnées :

Lorsqu'un point d'intérêt est localisé sur un étage de l'espace échelle, différent du premier étage, une optimisation sous-pixel est effectuée, afin de positionner au mieux ce pixel sur l'image de taille initiale. Cela s'obtient par un développement de Taylor d'ordre 2 de la fonction D(x,y,σ), en prenant comme origine les coordonnées du point d'intérêt candidat [4]:

$$D(x) = D + \frac{\partial D^T}{\partial x} X + \frac{1}{2} x^T \frac{\partial^2 D^T}{\partial^2 x^2} X \qquad \text{(II.3)}$$

où $x = (x, y, \sigma)^T$ au voisinage du point d'intérêt

La position précise de l'extremum $\hat{x}$ est déterminée en résolvant l'équation annulant la dérivée de cette fonction par rapport à x :

$$\hat{X} = -\frac{\partial^2 D^{-1}}{\partial x^2} \cdot \frac{\partial D}{\partial x} \quad \text{(II.4)}$$

Si le $\hat{x} > 0.5$ dans l'une des trois dimensions, cela signifie que le point est plus proche d'un des voisins dans l'espace des échelles discret. Dans ce cas, le point d'intérêt candidat est mis à jour et l'interpolation est réalisée à Partir des nouvelles coordonnées. Sinon, le delta est ajouté au point candidat initial qui gagne ainsi en précision.

II.2.1.4 Élimination des points d'intérêts de faible contraste

On appelle *contraste* d'un point de coordonnées (x, y, σ) la valeur $|D_h(x, y, \sigma)|$. Avec h est le numéro de l'octave où il a été sélectionné le point d'intérêt. Les points d'intérêt de faibles contrastes sont rejetés car ils sont instables (le bruit peut faire apparaître, disparaître ou déplacer les extrema de faible contraste).

David Lowe définit un seuil de contraste $C_t > 0$ de tel sorte que tous les points en dessous de ce seuil soient rejetés. Tous les extrema (x, y, σ) tel que $|D_h(x, y, \sigma)| < C_t$ sont rejetés.

En utilisant l'amélioration discutée dans la section en dessus nous trouvons [4] :

$$D_h(\hat{x}) = D + \frac{1}{2}\frac{\partial D^T}{\partial x}\hat{x} \quad \text{(II.5)}$$

Un seuillage absolu ($|D(\hat{x})| < 0.03$) est effectué pour éliminer les points instables, à faible contraste.

II.2.1.5 Élimination des points situés sur les arêtes

Les points situés sur les arêtes (ou contours) doivent être éliminés. Ceci, car la fonction D(x; y; σ), y prend des valeurs élevées, ce qui donne naissance à des extrema locaux instables, très sensibles au bruit. Un point instable aura une grande courbure le long du contour, mais une faible courbure dans la direction perpendiculaire. Pour déterminer si un point est le long d'une arête ou non il faut étudier les courbures en ce point, ce sont les valeurs propres de la matrice hessienne H en ce point :

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad \text{(II.6)}$$

Les valeurs propres de la matrice Hessienne sont proportionnelles aux courbures principales de D. Mais comme nous nous intéressons uniquement au rapport des deux valeurs propres, il

est inutile de les calculer, puisque la trace et le déterminant de la matrice permettent de déduire respectivement la somme et le produit de ces deux valeurs [4].

$$T_r(H) = D_{xx} + D_{yy} = \alpha + \beta \tag{II.7}$$

$$Det(H) = D_{xx} + D_{yy} - (D_{xy})^2 = \alpha\beta \tag{II.8}$$

En supposant que α est la plus grande valeur propre et r le rapport entre la plus grande et la plus petite valeur propre $r = \alpha / \beta$.

Nous avons :

$$\frac{T_r(H)^2}{Det(H)} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r\beta+\beta)^2}{r\beta^2} = \frac{(r+1)^2}{r} \tag{II.9}$$

Comme la fonction : $\frac{(r+1)^2}{r}$ est strictement croissante sur [1, +∞], donc pour vérifier que le rapport r est au-dessus d'un certain seuil r seuil, il suffit de vérifier que :

$$\frac{T_r(H)^2}{Det(H)} < \frac{(r_{seuil}+1)^2}{r_{seuil}} \tag{II.10}$$

Ceci est beaucoup moins coûteux en nombre d'opérations, que le calcul des valeurs propres. Lowe recommande de fixer r seuil à 10, et donc d'éliminer les points clés où le rapport des deux principales courbures est supérieur à 10.

II.2.2 Calcul des descripteurs

Comme nous l'avons déjà mentionné, l'algorithme proposé par D. Lowe est composé de deux phases importantes : la détermination des points clés et le calcul des descripteurs correspondants à chaque point. Donc et d'autant que les points clés sont déterminés, dans ce qui suit nous allons présenter les différentes étapes du calcul du descripteur SIFT.

II.2.2.1 Assignation d'orientation :

La présente étape est la dernière avant le calcul des descripteurs. Elle permet d'attribuer à chacun des points détectés une ou plusieurs orientations déterminées localement sur l'image. C'est ce qui assurera l'invariance de la méthode par rapport à la rotation et au changement d'échelle.

. Pour un point clé donné (x_0, y_0, σ_0), nous calculons d'abord la norme m(x, y) et l'orientation θ(x, y). Le calcul s'effectue au niveau de ses points voisins sur l'image filtrée, calculée au départ L(x; y; σ) avec σ_0 le plus proche facteur d'échelle du point :

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (11)$$

$$\theta(x,y) = tan^{-1} \frac{(L(x+1,y) - L(x-1,y))}{(L(x,y+1) - L(x,y-1))} \quad (12)$$

Une fois ce calcul préliminaire effectué, un histogramme des orientations est réalisé avec des intervalles couvrant chacun 10 degrés d'angle (figure 3). L'histogramme est doublement pondéré : d'une part, par une fenêtre circulaire Gaussienne de paramètre égal à 1,5*σ_0, d'autre part, par l'amplitude de chaque point.

Les pics dans cet histogramme correspondent aux orientations dominantes. Toutes les orientations dominantes permettant d'atteindre au moins 80% de la valeur maximale sont prises en considération. Ce qui provoque si nécessaire la création de points-clés supplémentaires ne différant que par leur orientation principale.

À l'issue de cette étape, un point-clé est donc défini par quatre paramètres(x, y, σ, θ).

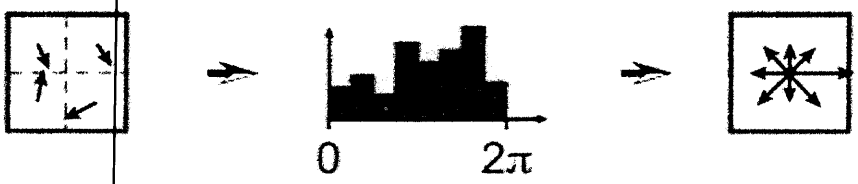


Figure II. 3: Construction de l'histogramme des orientations d'un point clé.

II.2.2.2 Descripteur SIFT du point d'intérêt

Après avoir désigné les points clés et leur orientation principale, il est maintenant temps de calculer le vecteur descripteur de chaque point clé. Tout comme l'étape précédente, le calcul qui suit s'effectue sur l'image lissée L(x, y, σ) avec σ le plus proche du facteur d'échelle du point.

Pour chaque point, on commence par modifier le système de coordonnées local, en utilisant une rotation d'angle égal à l'orientation du point-clé, mais de sens opposé. On considère ensuite, toujours autour du point-clé, une région de 16 *16 pixels, subdivisée en 4 *

4 zones de 4*4 pixels chacune. Sur chaque zone est calculé un histogramme des orientations comportant 8 intervalles.

En chaque point de la zone, l'orientation et l'amplitude du gradient sont calculés comme précédemment. L'orientation détermine l'intervalle à incrémenter dans l'histogramme, ce qui se fait avec, comme précédemment, une double pondération : Par l'amplitude et par une fenêtre Gaussienne centrée sur le point clé, de paramètre égal à 0, 5 fois le facteur d'échelle du point-clé comme l'illustre la figure 4.

Ensuite, les 16 histogrammes à 8 intervalles chacun sont concaténés et normalisés. Dans le but de diminuer la sensibilité du descripteur aux changements de luminosité, les valeurs sont plafonnées à 0.2 et l'histogramme est de nouveau normalisé, pour finalement fournir le descripteur SIFT du point-clé, de dimension 128.

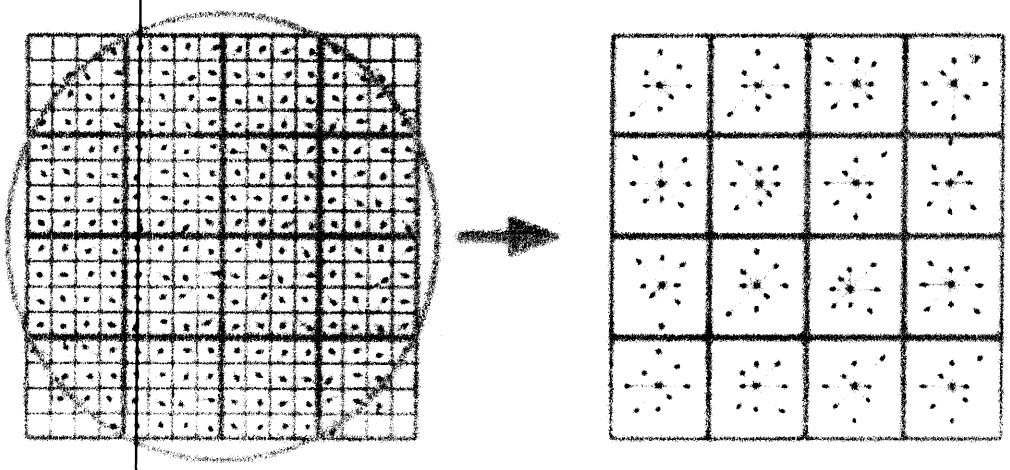


Figure II. 4: Calcul du descripteur d'un point clé

II.2.3 Correspondance entre images

La problématique de base pour laquelle la méthode SIFT a été conçue est la suivante : Peut-on trouver dans une image donnée (image requête), des objets déjà présents dans une collection d'images de référence préétablie ?

Afin de parvenir à cet objectif, il faut extraire de chaque image de la collection ces points clés et stocker les descripteurs. Ainsi au moment de la comparaison, il faut pour chaque point clé de l'image requête déterminer son plus proche voisin, en utilisant la distance Euclidienne, sur chaque image de la collection. Pour ce faire, Lowe utilise un algorithme d'approximation " Best-Bin-First "(BBF) afin d'éviter une recherche exhaustive.

II.3 Etape de Quantification

L'étape de la quantification, dans un système de hachage perceptuel, est un processus difficile parce que la manière dont les caractéristiques du vecteur de hachage intermédiaire continu changent, après les manipulations tolérables, reste imprévisible dans chaque intervalle de quantification de taille Q. Cette difficulté augmente plus quand elle suivie par une étape de crypto compression qui est très sensible au moindre changement même s'il est de l'ordre de 1 seul bit.

La raison de cette difficulté est que le vecteur de hachage intermédiaire continu doit être quantifié convenablement donnant le même vecteur de hachage intermédiaire discret pour toutes les images perceptuellement similaires. C'est-à-dire, dans le cas des manipulations tolérables/acceptables, le vecteur de hachage intermédiaire continu de l'image originale et le celui de l'image reçue doivent différer par une petite distance pour donner le même vecteur quantifié. Dans le cas des manipulations malicieuses, ces deux vecteurs doivent changer par une grande distance pour donner deux vecteurs quantifiés différents et par la suite deux signatures différentes.

Il existe deux types de quantification : la quantification uniforme et la quantification adaptative.

II.3.1 quantification uniforme : elle est basée principalement sur le passage du réel au discret par l'utilisation de la relation suivante :

$$x_Q = \left\lfloor \frac{x}{Q} \right\rfloor . Q \qquad (13)$$

Le symbole $\lfloor . \rfloor$ signifie l'entier arrondi.

Q représente le pas de quantification.

x la caractéristique originale et x_Q la caractéristique quantifiée.

Dans notre travail nous avons utilisé la quantification uniforme.

II.3.2 quantification adaptative : ou la quantification probabiliste "Probabilistic Quantization" est basée principalement sur le travail de [11]. Sa caractéristique clef est qu'elle prenne en compte la distribution des données à quantifier.

II.4 Conclusion

Dans ce chapitre, nous avons présenté en détail une étape importante du schéma général du HP, c'est l'extraction des caractéristiques. Cette étape utilise les différents aspects fondamentaux de l'algorithme SIFT. Ce dernier consiste en trois étapes fondamentales :

- Détection des points d'intérêts.
- Calcul des descripteurs.
- Correspondance entre images.

Nous avons vu que l'intérêt principal du descripteur SIFT réside dans le fait qu'il est invariant au changement d'échelle et aux transformations affines.

Chapitre III

III.1 Introduction

Depuis quelque année, nous assistons à un développement massif des techniques d'imagerie dans le domaine de la médecine. Le processus de numérisation permet à l'image médicale d'être traitée, formatée, travaillée sur une console informatique, et ensuite d'être remise au patient, au médecin demandeur ou transmise carrément à travers les réseaux de communication pour être télé diagnostiqué. De ce faite toutes ces opérations peuvent exposées ces images médicales à toutes sortes de piraterie et par la suite remet la confidentialité du patient en cause. L'une des solutions proposées pour ce type de situation est la technique de hachage perceptuel.

Dans ce chapitre, nous allons présenter un schéma de hachage perceptuel visant la protection des images numérique (images médicales). Il consiste en trois étapes élémentaires (extraction de caractéristique, quantification, crypto-compression) que nous allons les présenter en détails ainsi que les différents résultats de robustesse face aux attaques que peut subir ces images. Ce schéma proposé permet de sécuriser les images avec une signature empreinte irréversible.

III.2 base de données

Dans notre travail on a choisit une collecte d'images médicales à niveaux de gris qui se trouve dans la bibliothèque Matlab pour tester l'algorithme de hachage perceptuel élaboré.

Le niveau de gris est la valeur de l'intensité lumineuse en un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires. Donc pour représenter les images à niveaux de gris, on peut attribuer à chaque pixel de l'image une valeur correspondante à la quantité de lumière renvoyée. Cette valeur peut être comprise par exemple entre 0 et 255. Chaque pixel n'est donc plus représenté par un bit, mais par un octet. Pour cela, il faut que le matériel utilisé pour afficher l'image soit capable de produire les différents niveaux de gris correspondant. Le nombre de niveaux de gris dépend du nombre de bits utilisés pour décrire la " couleur "de chaque pixel de l'image. Plus ce nombre est important, plus les niveaux possibles sont nombreux.
Des échantillons d'images de la base utilisée sont représentés par la figure (III.1) :

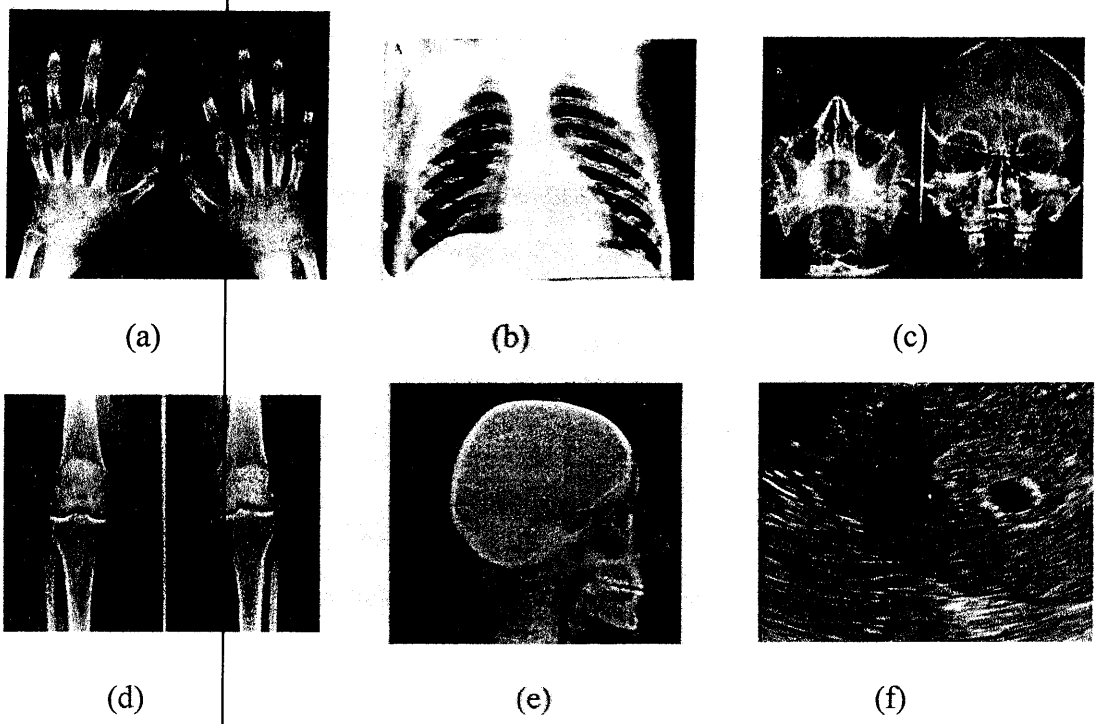


Figure III.1 : images de teste : (a) image issue des rayons X, (b) image issue des rayons X, (c), (e) image scanner, (d) image issue des rayons X, (f) image échographique

III.3 Algorithme de hachage perceptuel proposé

Comme nous l'avons déjà expliqué aux deux chapitres précédents, le schéma général du processus de hachage perceptuel proposé dans ce travail est représenté sur la figure (III. 2):

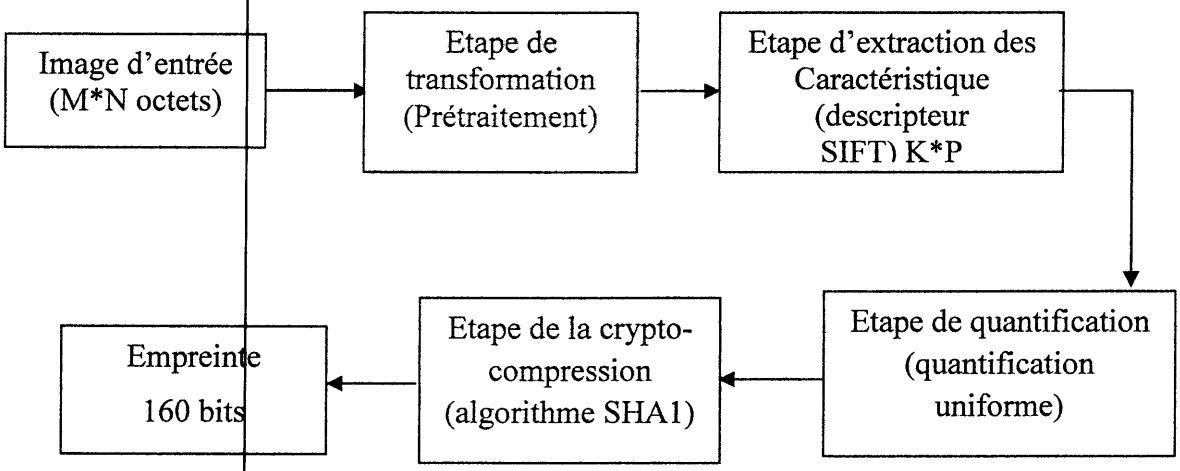


Figure III.2 : Organigramme de l'algorithme de hachage

Ce schéma est composé de ces différentes étapes :

III.3.1 Etape de transformation

En général c'est une étape de prétraitements. Elle vise principalement l'amélioration de la qualité de l'image. Dans notre travail, nous avons réalisé l'égalisation d'histogramme pour accentuer le contraste des images peu contrastée.

III.3.2 Etape d'extraction de caractéristique

Comme nous l'avons mentionné au chapitre 2, nous avons utilisé le descripteur SIFT pour extraire des caractéristiques invariantes des images utilisées afin de pouvoir garantir la robustesse du schéma proposé. En effet l'algorithme SIFT (*Scale-Invariant Feature Transform*) permet de détecter et de décrire les caractéristiques d'une image. Celui-ci comprend à la fois un algorithme de détection de points d'intérêt et d'extraction des caractéristiques distinctives de ces points pour la reconnaissance d'objet. Cette approche permet de transformer une image en une large collection de vecteurs de caractéristiques, chacun étant invariant aux transformations suivantes : translation de l'image, changement d'échelle (redimensionnement), rotation et partiellement invariant aux c

III.3.2.1 Description de la méthode proposée pour l'extraction

L'extraction des caractéristiques pertinentes se fait en 3 phases :

1. Détection des points d'intérêt à partir du SIFT en spécifiant les différents paramètres (tels que le seuil de contraste, le seuil de courbure ...).Dans [8], l'auteur précise qu'un faible contraste implique une forte sensibilité au bruit. Par conséquent nous pouvons déduire qu'un fort contraste serait moins sensible au bruit .Le contraste correspond théoriquement à la valeur de la différence de Gaussienne d'un point pour une octave et un intervalle spécifique. (Pour un seuil de 0.02 nous aurons entre 2000 et 3000 points d'intérêt).Nous avons choisi les 10 points ayant les plus grandes valeurs de contraste.
2. Choisir un nombre fixe de points parmi les points détectés en les ordonnant par leur valeur de contraste. Nous avons choisi la valeur de 0.09 comme seuil de contraste, cette valeur nous a permis de détecter une vingtaine de points d'intérêt tels que leur valeur de contraste est supérieure à 0.09.
3. Extraction de caractéristiques des points choisis. Dans notre étude on s'est basé sur les 3 pics du descripteur SIFT (de taille 128). Alors parmi les 128 orientations autours du point d'intérêts on choisit les 3 plus hauts sommets de l'histogramme d'orientation pour chaque point parmi les 10. Ces valeurs correspondent aux directions dominantes des gradients locaux.

A cette étape résulte le vecteur de caractéristiques réel de dimension 10*3afin de les quantifier dans l'étape suivante.

III.4 Etape de quantification

Dans l'étape de quantification, le vecteur intermédiaire de caractéristiques réelles de dimension 10×3 est quantifié et peut être codé sur 10×3 octets. Le nouveau vecteur intermédiaire de caractéristiques contient des valeurs discrètes. La quantification utilisée dans ce travail est basée sur la quantification uniforme, pour se faire on procède comme suit :

-d'abord on calcule le vecteur de caractéristique quantifié par la formule :

$$X_Q = \left\lfloor \frac{x}{Q} \right\rfloor \quad \text{(III.1)}$$

Avec X est la valeur originale du vecteur de caractéristiques réels, Qle pas de quantification et $X_Q$le quotient (entier arrondis), $\lfloor . \rfloor$ Signifie la valeur arrondie [12].

- ensuite pour pouvoir corriger les erreurs de quantification dues aux différentes attaques que peut subir l'image on calcule une variable Rqui représente le reste de l'opération de quantification :

$$R = x - X_Q * Q \quad \text{(III.2)}$$

D'après la référence [12], dans l'étape d'authentification de l'image en question, à la valeur$X_Q$on ajoute la valeur 0.25*Q si R<0 et on soustrait la valeur 0.25*Q si R>=0. Notons que le vecteur R a la même taille que le vecteur caractéristique extrait (vecteur des 3 premiers pics).

Maintenant et pour trouver le pas de quantification adéquat nous avons utilisé la formule de la quantification uniforme :

$$X_Q = \left\lfloor \frac{x}{Q} \right\rfloor * Q \quad \text{(III.3)}$$

Nous nous sommes basés sur la méthode présentée dans [1] où on cherche le pas Q expérimentalement. Cette dernière propose d'attaquer l'image par différentes valeurs de bruit gaussien dont son écart type σ varie de 0 à 12.5, puis essaye de trouver le pas qui permet de stabiliser les caractéristiques extraits:

- Nous testons premièrement le comportement du 1er pic du descripteur SIFT de chacune des 6 images de la base. La figure (III.3) montres la variation du 1er pic du premier point d'intérêt extrait pour les six images en fonction du bruit Gaussien ajouté aux images. Nous remarquons que les valeurs bruitées changent en fonction du bruit mais gardent presque la même variation

en fonction du bruit ajouté. D'où l'intérêt de les quantifier avec un pas qui néglige les petites modifications dues aux faibles densités de bruit qualifiées d'invisible.

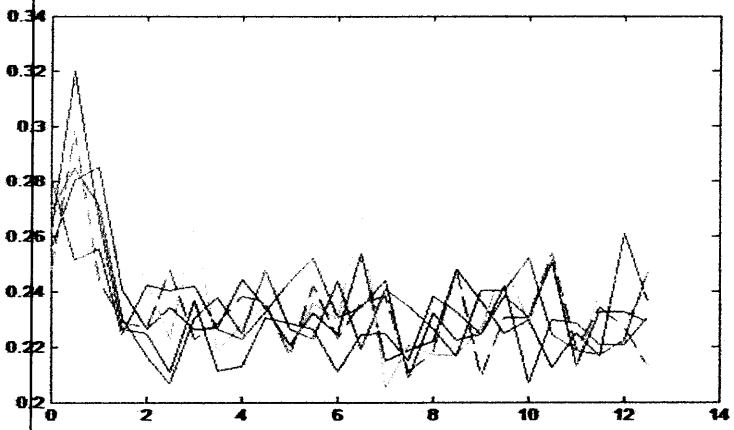


Figure III.3 : Variation du premier pic du 1er point de chaque image de base de données en fonction du bruit.

- Nous prenons maintenant le premier pic du 1erpoint parmi les dix sélectionnés et nous effectuons les différents tests de recherche du pas de quantification. Nous avons effectué la recherche du pas juste pour le premier point puis nous l'avons appliqué pour tous les autres points. Cette démarche est justifiée par le faite que les dix points possèdent un comportement similaire face à l'ajout de bruit comme le montre la figure (III. 4).

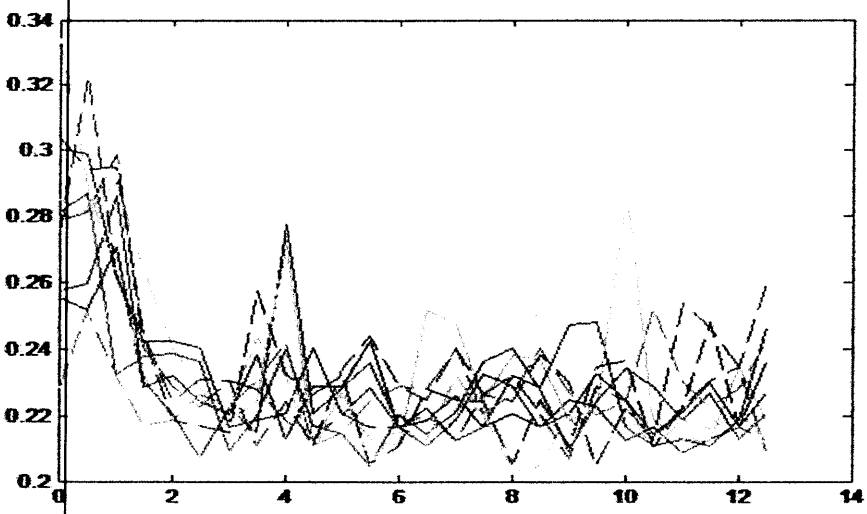


Figure III. 4 : Variation du premier pic des 10 points de l'image de la figure III. 1(a) en fonction du bruit.

L'approche suivie pour la recherche du pas Q est une méthode expérimentale basée sur la recherche manuelle du pas de quantification. Nous avons fait plusieurs tests pour plusieurs pas. Le but est de trouver un pas qui pour le bruit Gaussien dont l'écart-type $\sigma <= 12.5$ donne la même valeur de quantification. C'est-à-dire que la caractéristique quantifiée reste stable quelque soit les valeurs du bruit ajouté.

III.3.3.1 Recherche de la valeur du pas du 1er pic

La figure (III.4) présente les valeurs du premier pic du premier point avant et après quantification, et cela pour les niveaux de bruit Gaussien d'écart-type σ entre 0 et 12.5.D'après la figure 3. 4, Nous remarquons que, pour un pas de quantification Q=0.01, nous avons une stabilité des valeurs quantifiées jusqu'au bruit ayant σ =0.6, Pour un pas de Q=0.084, nous avons une valeur quantifiée σ= 1.1 quelque soit le niveau de bruit entre 0 et 12.5. Par contre pour le pas de quantification Q= 0.185, nous avons une stabilité de 0 à 12.5 du bruit σ. Donc pour quantifier le 1er pic, nous choisirons un pas de 0.185 qui donne une meilleure stabilité. Nous pouvons constater que la valeur du pas qui permet d'avoir une stabilité des valeurs quantifiées jusqu'au bruit ayant σ =12.5, correspond à **Q=0.185**.

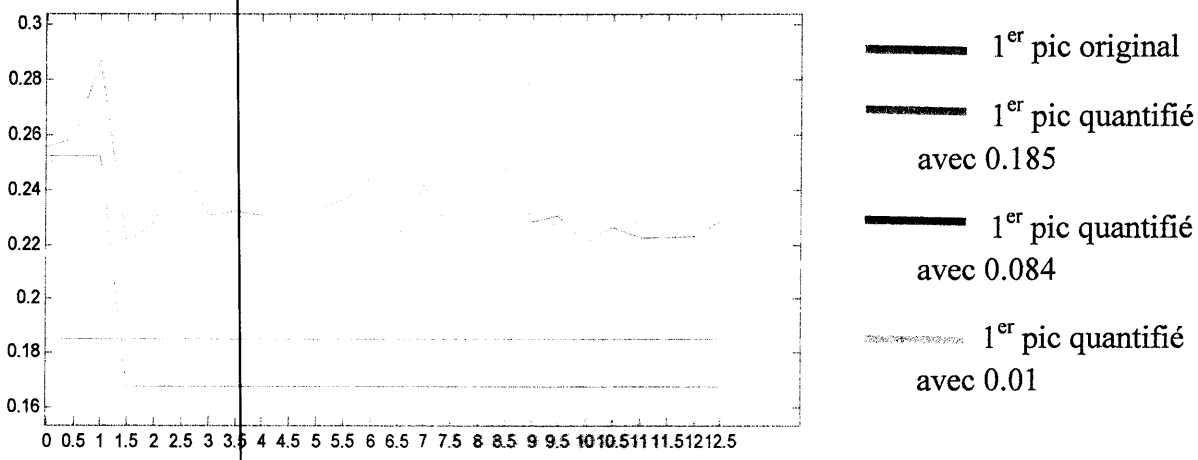


Figure III.5 : Quantification du premier pic du premier point pour l'image 1en fonction de bruit

III.3.3.2 Recherche de la valeur du pas du 2ème pic

La figure (III.6) présente les valeurs du premier pic avant et après quantification, et cela pour les niveaux de bruit Gaussien d'écart-type σ entre 0 et 12.5.

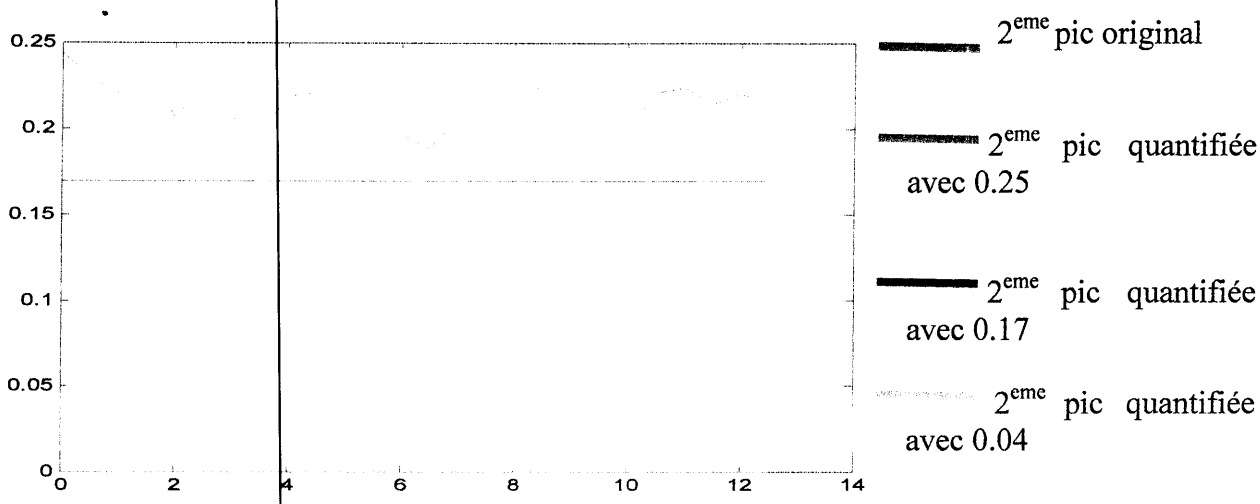


Figure III.6 : Quantification du premier pic du premier point pour l'image 1en fonction de bruit

Nous remarquons que, pour un pas de quantification de Q=0.04, nous n'avons pas la stabilité des valeurs quantifiées. Pour Q=0.25, nous avons toujours la même valeur quantifiée quelque soit le niveau de bruit entre 0 et 12.5.Une stabilité des valeurs quantifiées de sigma de 0 à 12.5 pour un pas Q=0.17. Donc pour quantifiée le 2eme pic on choisit un pas Q= **0.17**.

III.3.3.3 Recherche de la valeur du pas du 3èmepic

La figure (III7) montre les valeurs du 3eme pic du premier point avant et après quantification, et cela pour les niveaux de bruit Gaussien variant de 0 à 12.5.

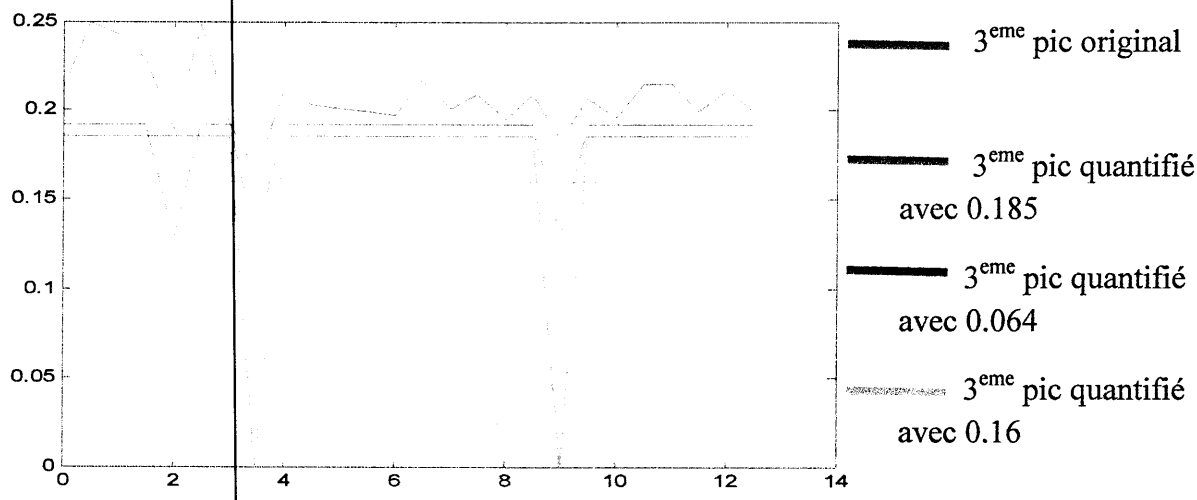


Figure III.7. : Quantification du troisième pic du premier point pour l'image 1 en fonction de bruit.

. Nous remarquons que, pour un pas de quantification Q=0.185, nous avons une stabilité des valeurs quantifiées jusqu'au bruit ayant σ =3. Pour un pas de quantification Q = 0.064, nous avons une stabilité des valeurs quantifiées jusqu'au bruit d'écart-type σ =1.8. Le pas de quantification Q=0.16 permet d'avoir une stabilité de 0 jusqu'au 12.5 et semble être adapté aux critères fixés auparavant. Alors pour quantifier le 3^{eme} pic on va choisit un pas **Q= 0.16**.Les résultats présentés dans cette section montrent que le choix d'un pas de quantification adéquat assure une stabilité des caractéristiques extraites qui à leur tour assure une stabilité des signatures perceptuelles générées. Cette méthode est une méthode manuelle pour la recherche du pas.

III.4 Modèle de teste de la méthode proposée

Dans cette section, nous présentons le processus global de la génération de la signature perceptuelle basée sur les points d'intérêt SIFT ainsi que la comparaison des deux signatures issues de l'image originale et celle attaquée. Le processus d'analyse est représenté sur la figure 3.8. Les entrées du système sont l'image originale et l'image attaquée. La première étape est l'extraction des points d'intérêt par les descripteurs SIFT. Ensuite, seulement les points d'intérêt les plus pertinents sont considérés pour la génération de la signature perceptuelles après l'étape de quantification. Ces mêmes étapes seront également appliquées sur l'image attaquée afin de générer une nouvelle signature.

Une fois le vecteur de caractéristiques obtenu, l'étape de la quantification permet d'obtenir une chaîne binaire qui forme le vecteur de hachage intermédiaire. Ce vecteur intermédiaire est ensuite injecté dans un module de crypto-compression tel que l'algorithme SHA-1 afin de générer la signature perceptuelle finale. Cette dernière sera comparée avec la signature de l'image originale pour étudier la stabilité des signatures perceptuelles.

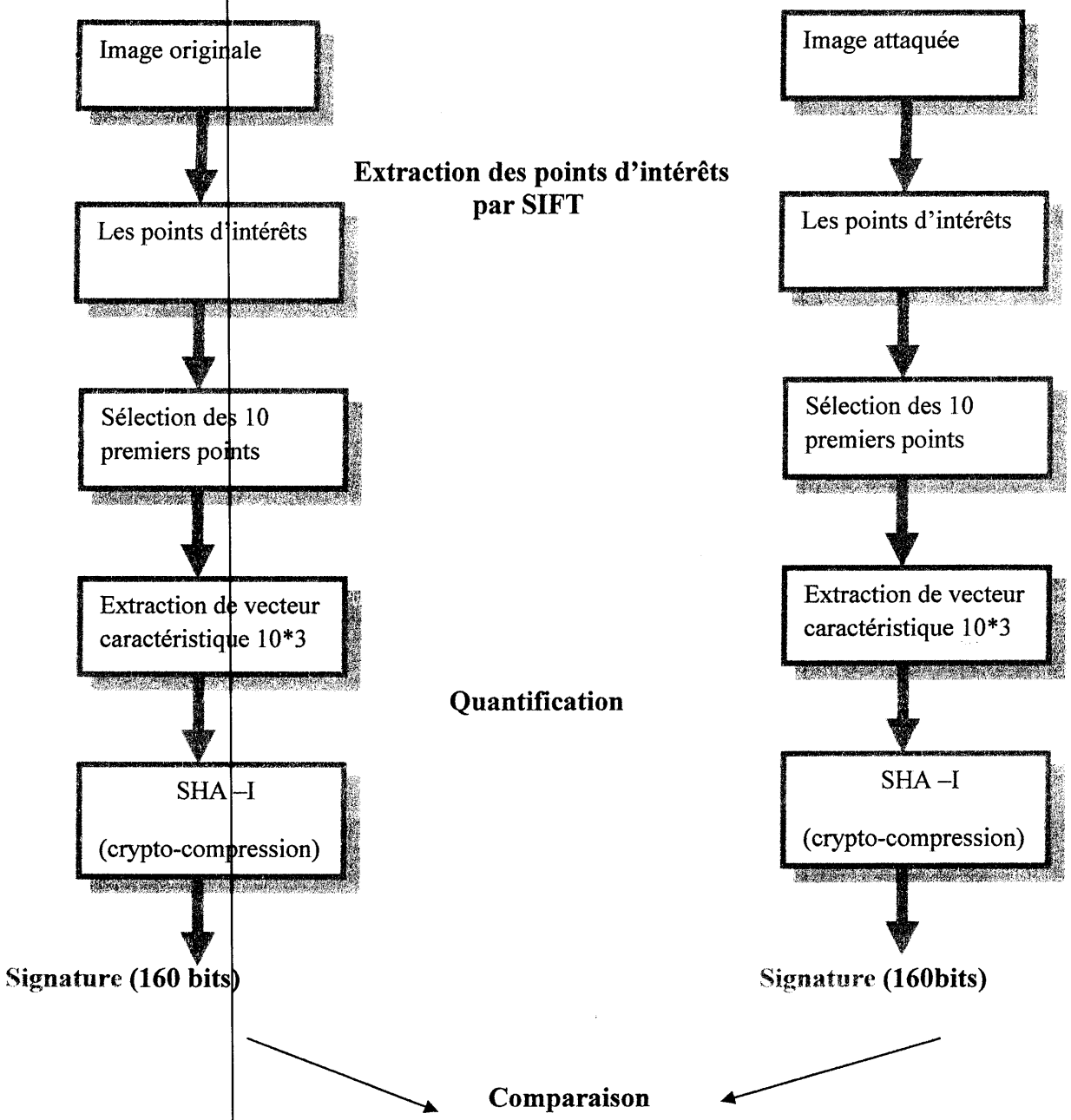


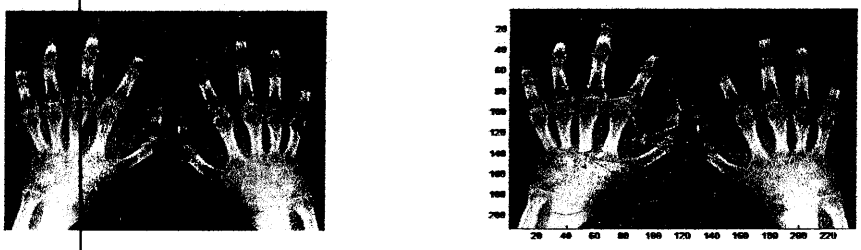
Figure III.8 : Processus de génération et de comparaison des signatures perceptuelles basées sur les descripteurs SIFT.

III.5 Résultats de simulation

Dans cette section, nous présentons les résultats obtenus concernant l'étude et l'analyse des signatures perceptuelles générées à partir des points d'intérêt SIFT basée sur le schéma proposé dans la section précédente. Nous allons quantifier les 3 premiers pics du descripteur SIFT des premiers 10 premiers points pour les deux images ; originale et attaquée.

III.5.1 L'empreinte des images originales

Nous appliquons l'algorithme proposé sur les différentes images de testes. Sur la figure (III.9) est représenté une image de teste originale ainsi que ses premiers dix points d'intérêts SIFT. La sélection de ces points est basé le paramètre contraste. (c'est les points les plus contrastés comme ça été déjà mentionné.



(a) (b)

Figure III.9: image originale (a) et points d'intérêts SIFT (b)

A l'étape de quantification c'est le passage du réel au discret du vecteur de caractéristique de dimension 10*3 ; les premiers 10 points avec leurs 3 premiers pics correspondants. Les valeurs du vecteur de caractéristiques sont données sur le tableau (III.1).

1^{er} pic	2^{eme} pic	3^{eme} pic
9538	9538	9600
9538	9538	9600
10462	9538	9600
9538	9538	9600
9538	9538	9600
9538	9538	9600
9538	9538	9600
10462	9538	9600
9538	9538	9600
9538	9538	9600

Tableau III.1 : vecteur des 10 premiers points quantifiés de l'image originale.

La dernière phase dans le processus de la génération de la marque est l'application de l'étape de crypto-compression où on a utilisé l'algorithme SHA-I. On obtient alors l'empreinte de 160 bits de l'image originale.

f050b83f869681085e2feb239af9143a839bb812

Sur le tableau (III.2), on représente les différentes signatures obtenues par le processus de hachage perceptuel proposé des six images de la base de données.

Image	Empreinte digitale
(a)	f050b83f869681085e2feb239af9143a839bb812
(b)	1da779c15e41b58b2239024cd34acbd700b94d46
(c)	7aa9349691c222a8340cefb14c8956ff7419d757
(d)	ec8074c48a5e14ad5645e7e5e7473653b427542f
(e)	b9f2382d79cd1722011ff91f1cca3e85087862a6
(f)	b385b3f692620623e6183d4d8a31e6aa1b5ad037

Tableau III. 2 : différentes empreintes obtenues

III.5.2 Testes de robustesses

Afin de tester la robustesse de notre algorithme, nous allons effectuer différentes attaques sur les images de testes. Le système proposé est dit robuste si :

- pour deux images différentes, il permet d'avoir deux empreintes différentes.
- pour la même image, l'empreinte de l'image originale et celle de l'image attaquée on obtient la même empreinte.

Pour le premier critère, et d'après le tableau (III. 2) le système proposé est robuste d'autant que toutes les empreintes obtenues des différentes images de tests sont différentes.

Pour le deuxième critère, nous allons attaquer les images de testes par les différents bruits ci-dessous et nous testons les différentes empreintes obtenues :

a-Ajout de bruit (Salt and Pepper, poisson)

b-Rotation

c-Filtre médian (3*3)

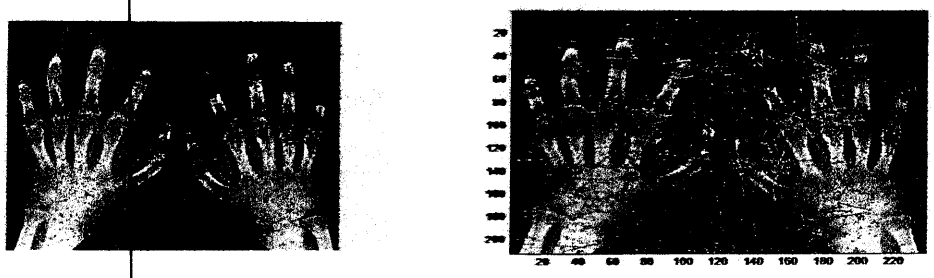
d-Compression

e-Niveau de gris

III.5.2.1 Ajout de bruit

Nous avons testé deux types de bruit : Salt and Pepper et Poisson.

a. Salt and Pepper : sur l'image originale de la figure (III. 1(a)), nous ajoutons le bruit Salt and Pepper avec un paramètre 0.03 et on calcule le descripteur SIFT



(a) (b)

Figure III.10 : image attaquée par Salt and Pepper avec un paramètre de 0.03(a) et détection de points SIFT (b).

Après quantification et crypto compression, nous obtenons l'empreinte de 160 bits suivante :

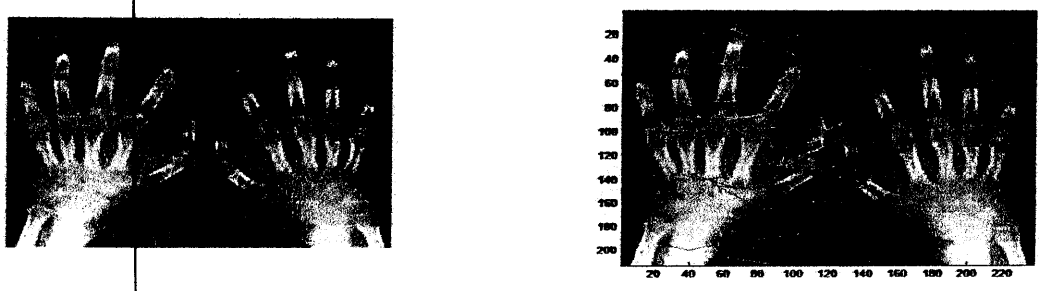
f050b83f869681085e2feb239af9143a839bb812

Nous appliquons l'algorithme sur différentes images et pour différentes valeurs de bruits, les résultats obtenus sont illustré sur le tableau (III.3)

Paramètre du bruit	Image (a)	Image (d)	Image (b)
0.03	Similaire	Similaire	Similaire
0.05	Similaire	Différent	Similaire
0.3	Similaire	Différent	Similaire
0.5	Similaire	Différent	Similaire

Tableau III. 3: Les empreintes générées à partir du descripteur SIFT pour différentes valeurs de la densité du bruit ajouté.

b. Poisson : Après une attaque sur l'image originale par le bruit poisson on retrouve l'image suivante montré dans la figure (III.11).



(a) (b)

Figure III.11 : l'image attaquée par le bruit poisson (a) et détection de points SIFT(b)

Nous appliquons le même traitement précédent, nous retrouvons alors l'empreinte suivante :

f050b83f869681085e2feb239af9143a839bb812

Nous appliquons l'algorithme sur différentes images attaquée par le bruit poisson, les résultats obtenus sont illustré sur le tableau (III.4)

	Image (a)	Image (d)	Image (f)
Bruit poisson	Similaire	Similaire	Similaire

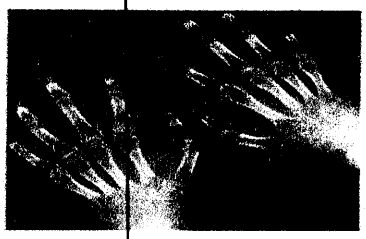
Tableau III.4: Les empreintes générées à partir du descripteur SIFT des images attaquée par le bruit poisson.

Nous remarquons que cet algorithme est robuste car malgré l'attaque ajouté à l'image, cette dernier garde la même empreinte ce qui signifie que cet algorithme garde les mêmes caractéristiques invisible de celle de l'image originale et l'image attaquée.

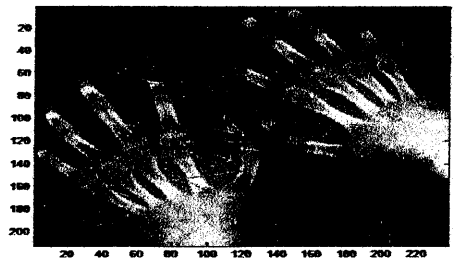
On conclu que la robustesse de cet algorithme est très efficace sur l'attaque de l'ajout de bruit (Salt and Pepper, poisson).

III.5.2.2 Rotation

Pour ce type d'attaque, nous allons utiliser différentes valeurs d'angles de rotation sur les images originales. La figure (III. 12) montre un exemple



(a)



(b)

Figure III.12 : l'image attaquée par une rotation d'angle 35(a), détection de points (b)

La signature générée avec cette attaque est donnée par :

f050b83f869681085e2feb239af9143a839bb812

Nous appliquons l'algorithme sur différentes images attaquées par différentes angles de rotation, les résultats obtenus sont illustrés sur le tableau (III.5)

Angle de rotation	Image (a)	Image (d)	Image (f)
90	Similaire	Similaire	Différent
180	Similaire	Différent	Différent
270	Similaire	Différent	Différent

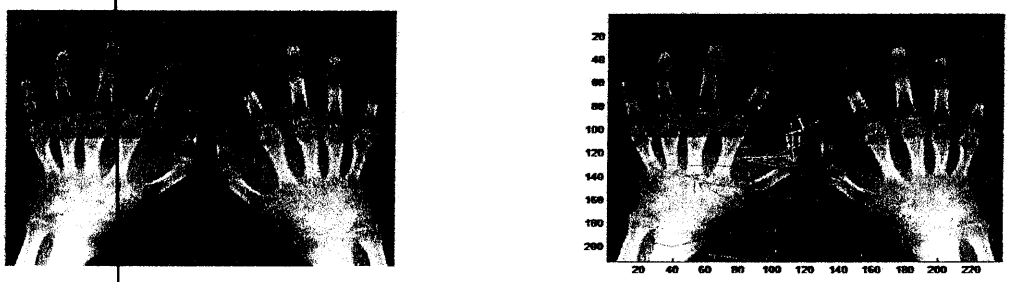
Tableau III.5: Les empreintes générées à partir du descripteur SIFT pour différentes angle de rotation.

Nous remarquons que cet algorithme est n'est pas robuste car l'empreinte est changé après l'ajout de l'attaque malgré l'attaque ajouté à l'image, cette dernier garde la même empreinte ce qui signifie que cet algorithme ne garde pas les même caractéristiques de celle de l'image originale et l'image attaquée. On conclu que la robustesse de cet algorithme n'est pas efficace dans cette attaque.

III.5.2.3 niveau de gris

C'est une attaque que nous l'avons simulé nous même. Nous changeons les valeurs d'intensité de certains pixels, donc nous pouvant la classer dans la catégorie des attaques malveillantes. Si le processus de hachage proposé est robuste l'empreinte doit changer, c.-à-d.

qu'il doit être sensible à tout traitement suspect. Cette attaque est illustrée par la figure (III.13).



(a) (b)

Figure III.13 : image attaquée par le niveau de gris (a) et points d'intérêt SIFT (b).

Après avoir appliqué le processus de hachage perceptuel, nous obtenons la signature suivante :

f050b83f869681085e2feb239af9143a839bb812

Les différentes empreintes générées par les différentes images de testes sont comparées à ceux des images originales. Les résultats sont montrés sur le tableau (III.6).

Attaque	Image (a)	Image (d)	Image (f)
Niveau de gris	Similaire	Différente	Différente

Tableau III.6: Les empreintes générées à partir du processus proposé pour l'attaque niveau de gris.

Nous remarquons que l'empreinte de l'image attaquée a changé par rapport à celle de l'image originale. Pour ce type d'attaque tout dépend de la dimension de la plage de variation des pixels. Si elle coïncide avec un point d'intérêt l'empreinte change sinon on obtient les mêmes empreintes. Cela signifie que cet algorithme n'est pas robuste à l'attaque de niveau de gris car cette attaque change les caractéristique invisible de l'image c'est pour cela notre empreinte a été changée.

III.5.2.4 compression

. Pour ce type d'attaque, nous allons utiliser différentes valeurs de facteur de compression de 10 sur les images originales. La figure (III.14) montre un exemple

(a) (b)

Figure III.14 : image attaquée par un facteur de compression (a) et points d'intérêt SIFT (b).

La signature générée avec cette attaque est donnée par :

f050b83f869681085e2feb239af9143a839bb812

Nous appliquons l'algorithme sur différentes images attaquées par différentes angles de rotation, les résultats obtenus sont illustrés sur le tableau (III.7)

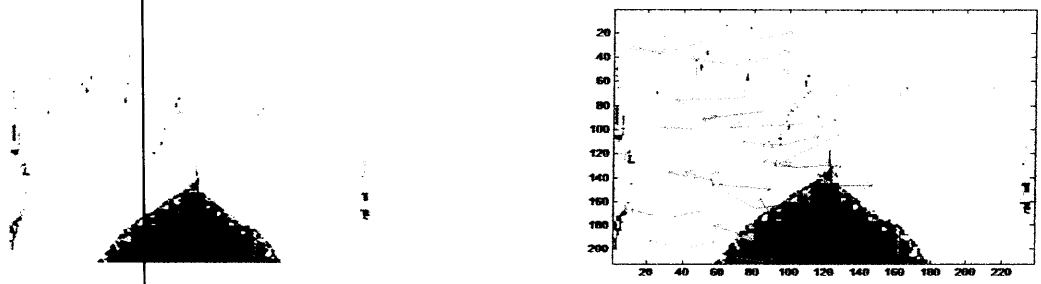
Facteur de compression	Image (a)	Image (d)	Image (f)
80	Similaire	Similaire	Similaire
50	Similaire	Similaire	Similaire
20	Similaire	Similaire	Similaire
0	Similaire	Similaire	Similaire

Tableau III.7: Les empreintes générées à partir du processus proposé pour l'attaque de compression JPEG.

D'après ce tableau, on peut conclure que cet algorithme est robuste contre l'attaque JPEG

III.5.2.5 filtre médian

Après une attaque sur l'image originale par le filtre médian on retrouve l'image suivante montré dans la figure (III.15)



(a) (b)

Figure III.15 : image attaquée par un filtre médian (a) et points d'intérêt SIFT (b).

Après avoir appliqué le processus de hachage perceptuel, nous obtenons la signature suivante :

f050b83f869681085e2feb239af9143a839bb812

Nous appliquons l'algorithme sur différentes images attaquée par le bruit poisson, les résultats obtenus sont illustré sur le tableau (III.8)

	Image (a)	Image (d)	Image (f)
Filtre médian	Similaire	Différente	Similaire

Tableau III.8 : Les empreintes générées à partir du processus proposé pour l'attaque du filtre médian

Les résultats obtenus nous permettent de confirmer la robustesse de cet algorithme car ces résultats nous ont donné la même empreinte que celle de l'image originale.

III.6 Conclusion

Dans ce chapitre nous avons présenté une nouvelle méthode de hachage perceptuel des images pour générer une signature perceptuelle robuste et sécurisée.

Dans le schéma proposé, nous nous sommes basés sur le descripteur SIFTdans l'étape d'extraction de caractéristiques. Nous avons utilisé trois valeurs de ce descripteur pour les dix points sélectionnés. De ce faite notre vecteur réel de caractéristique est de dimension 10*3. Pour la quantification nous avons procédé par la quantification uniforme en plus d'une technique de correction qui a permit de garder les mêmes valeurs quantifiées mêmes en présence des différents attaques.

Le système proposé a permit de garantir la robustesse en général contre la compression JPEG, l'ajout de bruit, la rotation et le filtrage du faite qu'il nous donne la même empreinte.

Nous avons remarqué qu'il est sensible à la nature de l'image. Par exemple pour l'image échographique et contre l'attaque de rotation, le schéma n'est pas robuste pour les différents testes effectués.

Conclusion Générale

Conclusion générale

Dans ce travail, nous avons proposé un système de hachage perceptuel des images numériques. Nous l'avons appliqué sur les images médicales.

La plupart des méthodes existantes de hachage perceptuel des images se concentrent principalement sur l'étape d'extraction des caractéristiques. L'objectif principal pour ces méthodes est d'extraire des caractéristiques visuelles qui restent relativement constantes face à un type précis de manipulations acceptables. Ces méthodes réduisent un système de hachage perceptuel des images à un système qui se satisfait de l'extraction de quelques caractéristiques résistantes à certaines manipulations. Par conséquent, ni la taille réduite de la signature est respectée, ni la sécurité du système est prise en considération.

Pour notre schéma nous avons concentré notre étude non seulement sur l'extraction de caractéristiques mais aussi sur la quantification et la crypto compression. Ceci est pour avoir à la fois un système robuste, sécurisé (l'empreinte obtenue est irréversible) et de taille de signature réduite (160 bits).

Pour l'extraction de caractéristiques nous avons utilisé le détecteur de points d'intérêts SIFT ainsi que son descripteur. Le vecteur de caractéristique se compose des trois premiers pics du descripteur des dix premiers points les plus forts en contraste.

Pour choisir la méthode de quantification et par la suite un choix approprié du pas de quantification, nous avons mené une analyse du comportement statique des caractéristiques extraites sous l'effet d'un bruit additif (bruit uniforme ou Gaussien). Cette analyse vise à simuler le comportement des transformations tolérables que peut subir l'image et par la suite trouver le meilleur pas qui permet de garantir la stabilité des caractéristiques extraites après quantification.

Les résultats expérimentaux obtenus révèlent que le schéma proposé offre une bonne robustesse contre la compression JPEG, l'ajout de bruit, la rotation et le filtrage. Seulement il faut noter qu'il n'est pas robuste avec le niveau de gris.

Perspectives

Dans le schéma proposé, nous avons utilisé une méthode manuelle pour la recherche du pas de quantification. En perspectives nous proposons d'utiliser les méthodes adaptatives qui permettent le calcul automatique du pas de quantification.

D'après les résultats expérimentaux obtenus, nous avons noté que pour les images échographiques le schéma proposé n'est pas robuste contre la rotation. Ceci est dû au caractère texturé de ce type d'image. Donc et en perspectives nous pensons qu'il serait intéressant d'utiliser une autre technique pour l'extraction de caractéristiques telles que les estimateurs de Bayes.

Bibliographie

Bibliographie

[1] R. L. Rivest : TheMD5Message Digest Algorithm, 1992.

[2] NIST : FIPS PUB 180-3, Federal Information Processing Standard (FIPS), Secure Hash Standard (SHS), Publication 180-3. Rapport technique, National Institute of Standards and Technology, Department of Commerce, October 2008a.

[3] A. J.Menezes, S. A. Vanstone et P. C. V. Oorschot: *Handbook of Applied Cryptography*. CRC Press, Inc., Boca Raton, FL, USA, 1stédition, 1996. ISBN *0849385237*.

[4] Azhar Hadmi. Protection des données visuelles Analyse des fonctions de hachage perceptuel. Thèse de doctorat .Université de Montpellier II. Science et technique de Languedoc. 2012.

[5] http://csrc.nist.gov.

[6] M. K. Mihçak et R. Venkatesan : A Perceptual Audio Hashing Algorithm : A Tool for Robust Audio Identification and Information Hiding. *In Proceedings of the 4th International Workshop on Information Hiding*, IHW '01, pages 51–65, London, UK, UK, 2001. Springer-Verlag. ISBN 3-540-42733-3.

[7] Zhu, J. Huang, S. Kwong et J. Yang : Fragility Analysis of Adaptive Quantization-Based Image Hashing. *IEEE Transactions on Information Forensics and Security*, 5:133–147,March 2010. ISSN 1556-6013.

[8] Dittmann, A. Steinmetz et R. Steinmetz : Content-Based Digital Signature for Motion Pictures Authentication and Content-Fragile Watermarking. *In Proceedings of the IEEE International Conference on Multi Media Computing and Systems - Volume 2*, ICMCS '99,Pages 209–213. IEEE Computer Society, 1999. ISBN 0-7695-0253-9.

[9] David.G. Lowe. Object recognition from local scale-invariant features. In Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, volume 2, pages 1150–1157 vol.2, 1999.
Doi 10.1109/ICCV.1999.790410.

[10] David G. Lowe. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision, 60(2) :91–110, November 2004. ISSN 0920-5691. doi : 10.1023/B:VISI.0000029664.99615.94.URL
http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94.

[11] V. Monga : *Perceptually Based Methods for Robust Image Hashing*. Phd dissertation, University of Texas at Austin, 2005.

[12] Q. Sun et S. F. Chang : A robust and secure media signature scheme for jpeg images. *VLSI Signal Processing*, 41(3):305–317, 2005.

[13] D. G. Lowe : Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110,November 2004.

Annexe

Annexe

Pour faciliter l'utilisation du schéma proposé, nous avons conçu une interface graphique à base du logiciel Matlab.

La version finale de cette interface est donnée sur la figure (a), qui exprime l'exécution finale de notre algorithme avec toutes les étapes.

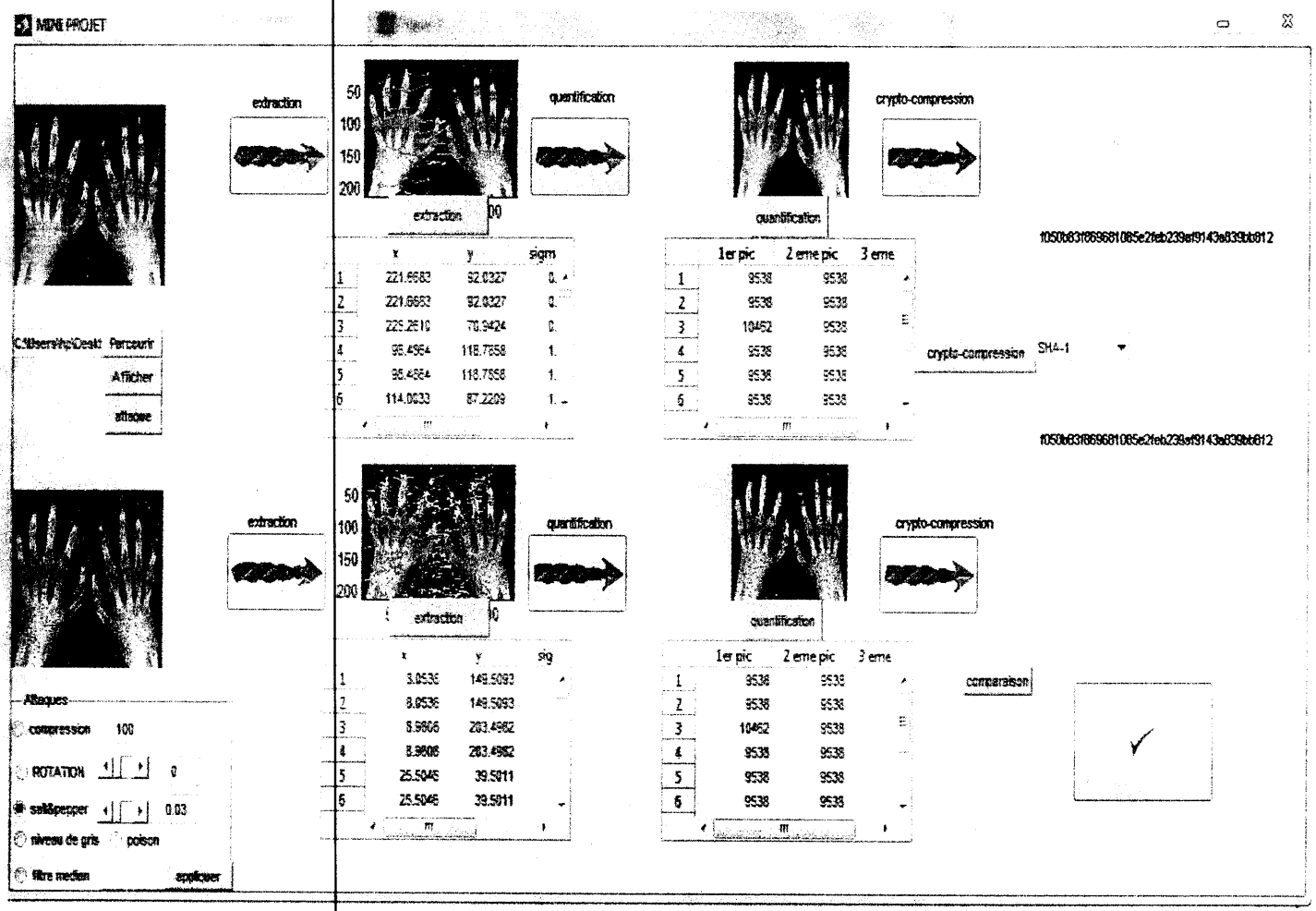


Figure (a) : interface utilisateur après l'exécution finale.

- La figure (b) exprime premièrement l'affichage de l'image originale et attaquée.

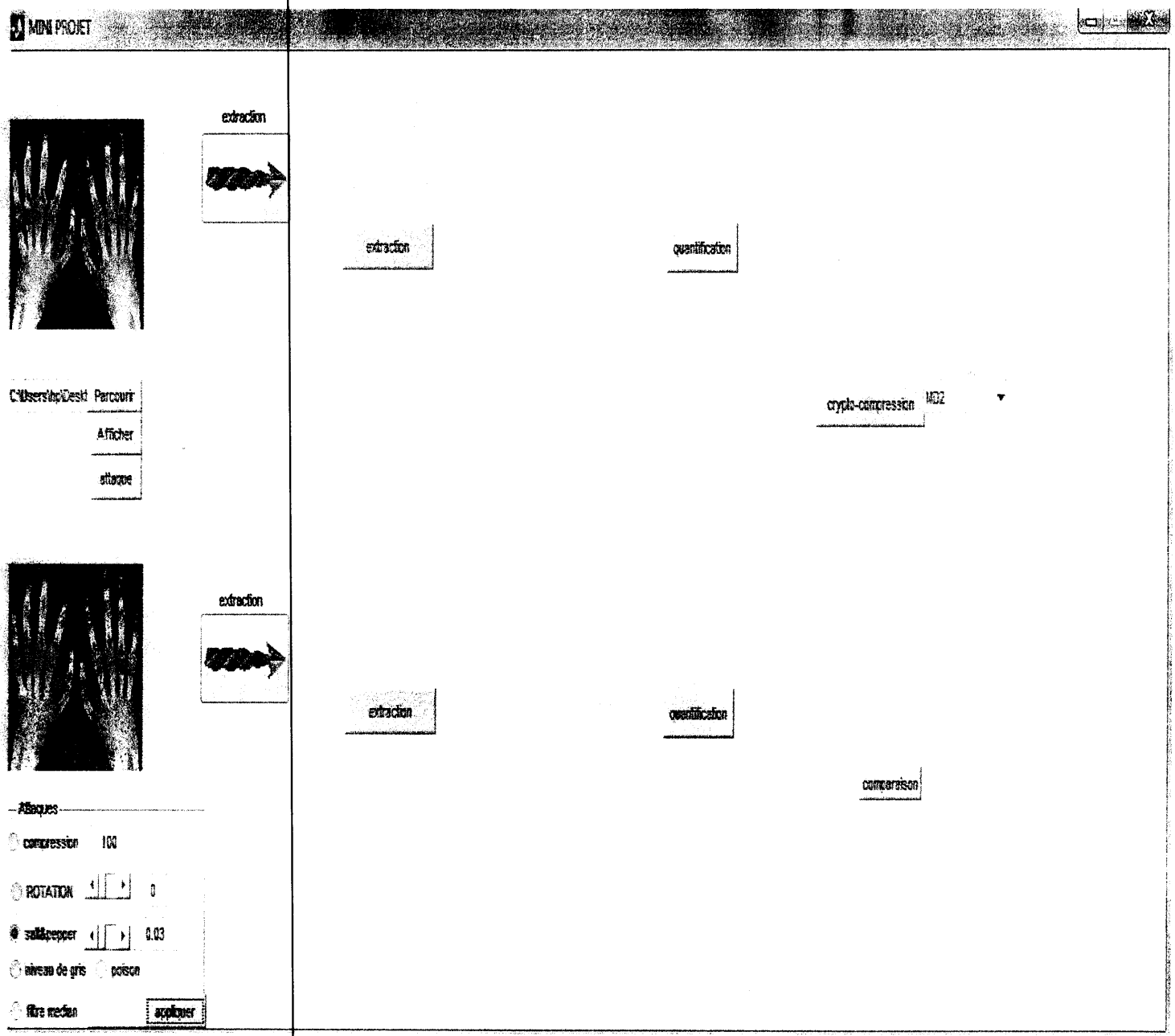


Figure (b)

• · La figure (c) montre l'étape de l'extraction de caractéristique des deux images elle nous donne le vecteur caractéristique réel des 3 premier pic.

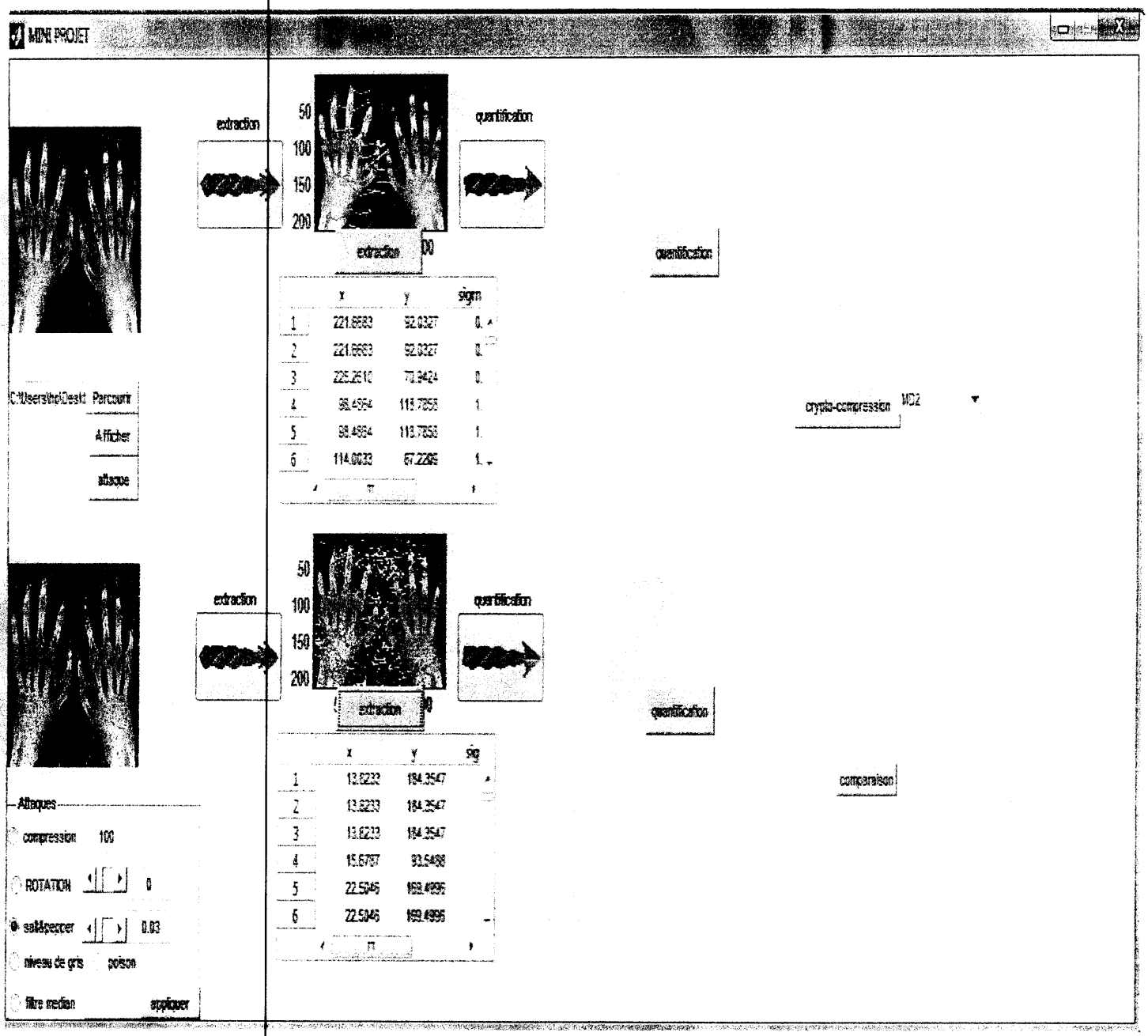


Figure (c)

- La figure (d) nous permet de donner le vecteur caractéristique des 3 premier pic entier après l'utilisation de l'étape de quantification

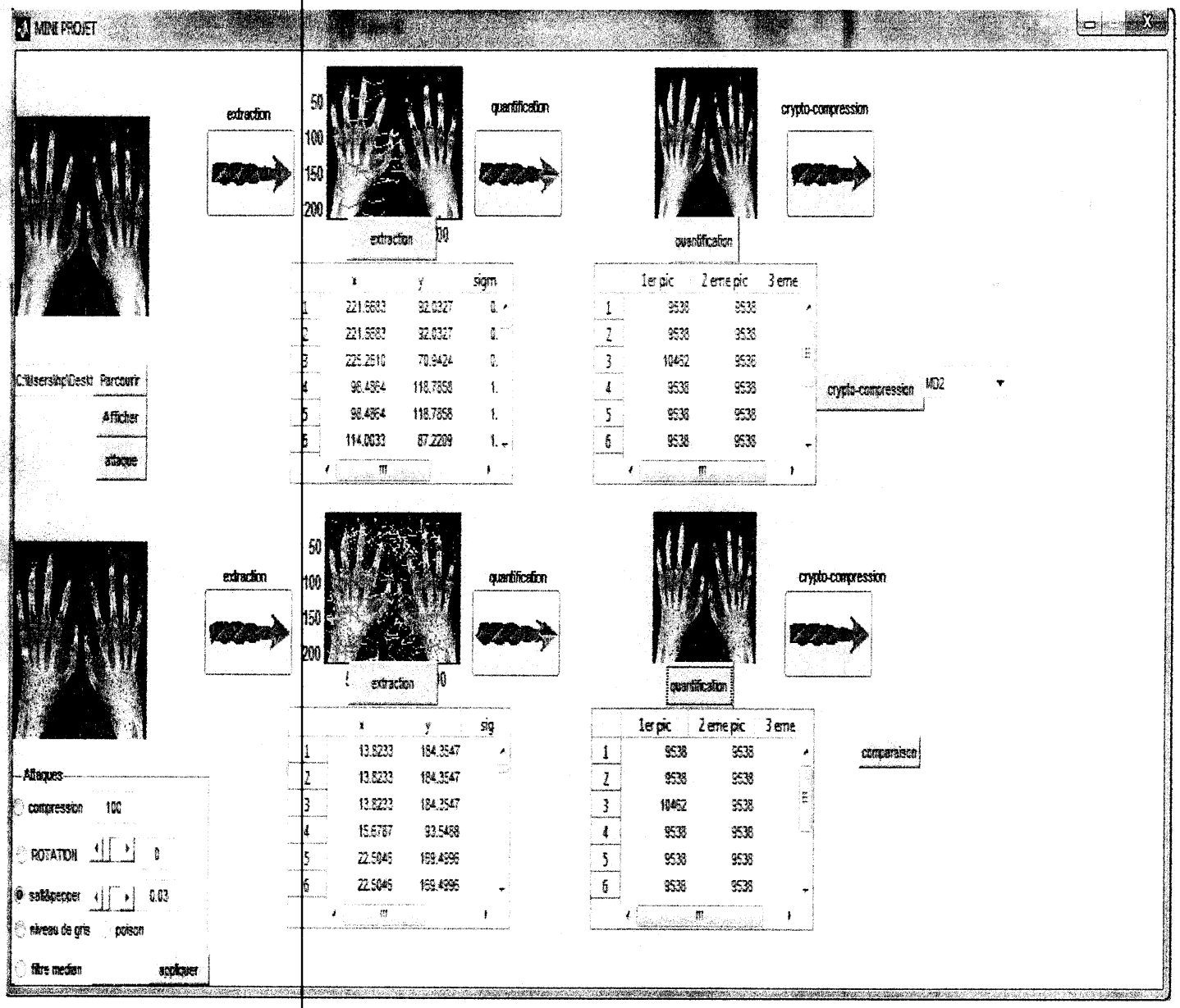


Figure (d)

- . Après l'étape de quantification on passe à la crypto compression, dans cette étape on va utiliser l'algorithme SHA-I pour crypter le vecteur caractéristique en une empreinte de 160 bits. La figure (e) montre tous ça.

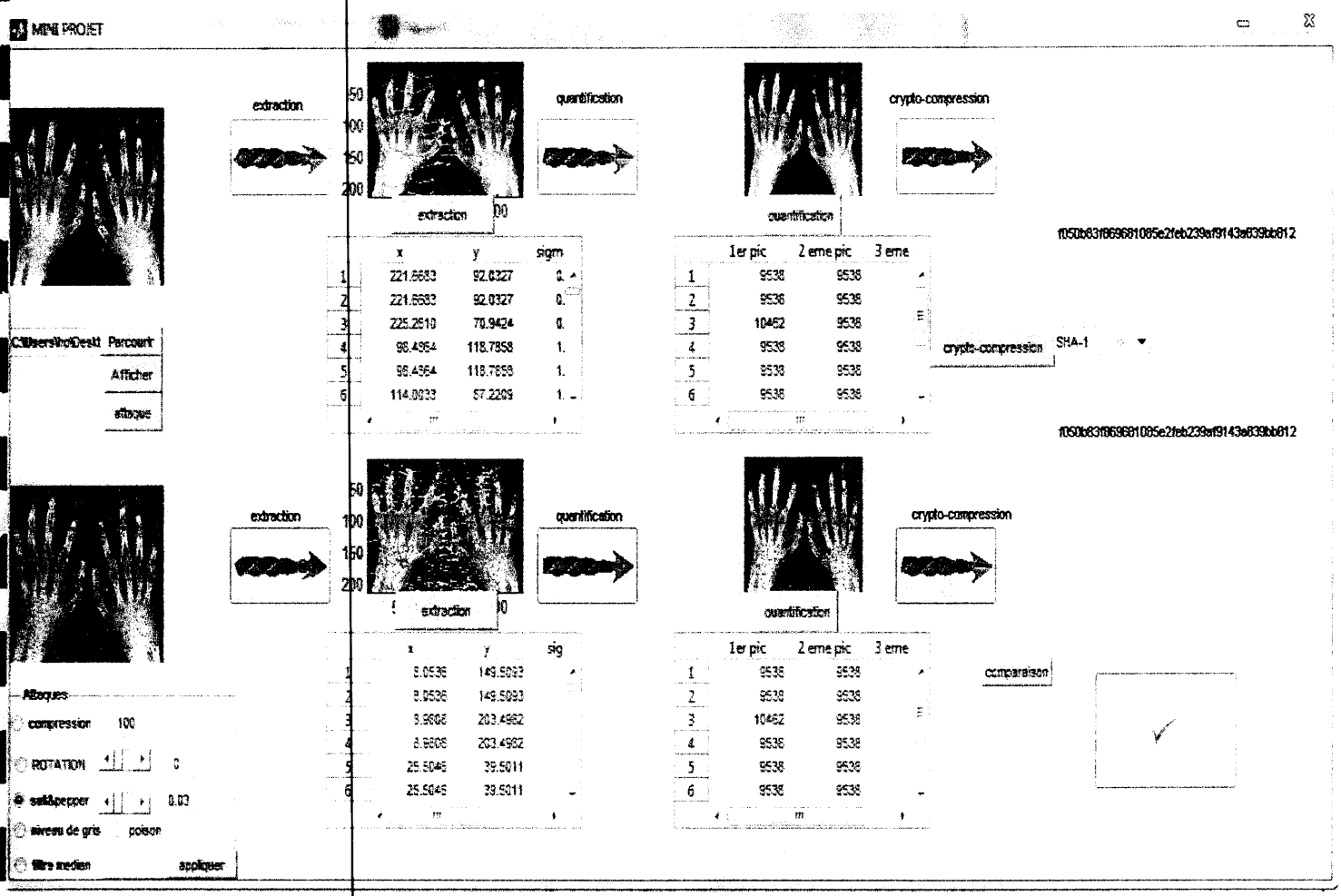


Figure (e)