



Université Abou Bekr Belkaid de Tlemcen  
Faculté de Technologie  
Département de Génie Biomédical  
Laboratoire de Génie Biomédical



**Master En**  
**IMAGERIE MEDICALE**  
**(2<sup>ème</sup> Année)**

**"BM 941"**  
**POLYCOPIE DES TRAVAUX PRATIQUES**  
**INFORMATIQUE MEDICALE**

**Filière de Génie Biomédical**

Elaboré par :

**Dr BOUKLI HACENE Ismail**

Courriel: [ismail.bouklichacene@mail.univ-tlem.dz](mailto:ismail.bouklichacene@mail.univ-tlem.dz)

Année universitaire : 2016 – 2017

## **Préface :**

L'informatique médicale s'est progressivement affirmée en tant que discipline scientifique.

Ce polycopié est constitué de textes de travaux pratiques élaborés pour étudier et gérer pratiquement des systèmes d'information médicale hospitaliers et en santé. Il permet d'offrir une formation interdisciplinaire généraliste qui aborde les différents enjeux de la mise en œuvre et de l'utilisation appropriée d'outils informatiques pour soutenir la pratique médicale et soignante, et plus largement le fonctionnement des systèmes de santé (Système d'information en Santé 'SIS', Système d'information Hospitalier 'SIH', Système d'Information en Radiologie 'SIR').

Cet enseignement qui se déroule au cours d'un semestre est destiné aux étudiants de MASTER (2) LMD option Imagerie Médicale. Les textes de travaux pratiques que nous proposons, comportent chacun une partie théorique détaillée suivi de simulation explicatives.

Les deux premiers TP permettra aux étudiants de concevoir et d'implémenter une base de données médicale en passant par les différents niveaux (conceptuel, logique et physique). Ils auront l'occasion de bien assimiler l'apport des Systèmes de Gestion de Base de Donnée Médicale (SGBDM) avec Open office et SQL pour gérer des données médicale.

L'objectif de TP N° 3 est d'apprendre, manipuler, traiter, filtrer des images médicales, avec le logiciel DICOM WORKS 'Digital Imaging and Communication in Medicine' ; car il permet d'obtenir des images du patient ainsi que toutes les informations associées dans un format identique permettant l'interconnexion et l'interopérabilité des équipements et le transfert des données médicales.

Afin de Familiariser l'étudiant sur les techniques de codage et de compression des images médicale ; le TP 4 viser a données aux étudiants les fondements de base pour l'évaluation des avantages et les inconvénients des différentes techniques de compression des images médicales en utilisant la transformée en ondelette par les codeurs de sous bande (EZW « Embedded Zerotrees of Wavelet transforms » ,SPIHT « Set Partitioning In Hierarchical Trees ») ; ainsi que les critères de choix d'une technique de compression des images médicales.

En fin, l'objectif des deux dernier TP est de sensibiliser l'étudiant sur l'importance de la sécurité informatique et le lui apprendre à maîtriser les technologies et les algorithmes robustes utilisés en sécurité informatique dont le but est la protection des images médicale.

Tout commentaire ou proposition ou critique constructive permettant l'amélioration des textes ainsi élaborés sera recueillie avec grand intérêt.

Je tiens à remercier Mr Hadj Slimene Zine eddine et Mr Abderahim Mohamed El-Amine pour tout l'intérêt qu'ils ont témoigné pour examiner ce travail.

***BOUKLI HACENE Ismail***

## Table des matières

<b>Préface</b>	3
<b>TP n°1: Systeme de Gestion de Base de Donnée « Open Office »</b>	4
I. Introduction.....	4
II. Quelques definitions.....	4
III. Objectifs.....	6
III.1.Ouverture de la Base de donnée Open Office.....	6
III.2.Création des Tables.....	6
III.3.Définitions des Liens entre les Tables .....	7
III.4.Insertions des Données .....	7
V. Travail à Réaliser.....	7
VI. Travail Demandé « Interrogation de la base de données » .....	9
<b>TP N °2: MANIPULATION SQL.....</b>	11
I. Introduction .....	11
II. Partie Théorique .....	11
II.1.Qu'est ce que SQL.....	11
II.2. Comment est construite une instruction SQL.....	12
II.3.Les Contrainte d'intégrité .....	13
II.4. Types de Données .....	13
II.5. Syntaxes pour la description des données .....	13
II.6. MySQL : L'outil PHPMyAdmin .....	14
II.6.1. Qu'est-ce que PHPMyAdmin.....	14
II.7.Avantage de SQL.....	17
III. Partie Pratique.....	18
III.1.Installation de l'environnement de travail .....	18
III.2. But de ce TP .....	18
III.3.Déscipion de données .....	18
III.4.Travail demandé dans la séance de TP.....	18
III.5.Requettes « Intérogation de la nouvelle base de donnée médicale en SQL....	23

<b>TP N °3: DICOM WORKS</b> .....	24
I. Introduction .....	24
II. Partie Théorique .....	24
II.1.Transformation DICOM des Images .....	24
III. Partie Pratique.....	26
III.1.But de ce TP .....	26
III.2.Interface de Logiciel DICOM WORKS.....	26
III.3.Travail Demandé .....	27
<b>TP N °4 : Compression des images médicales par des codeurs de sous bandes (EZW, SPIHT)</b>	28
I. Introduction .....	28
II. Partie Théorique .....	29
II.1.Méthode de Codage sans Perte .....	29
II.2. Méthode de Codage avec Perte .....	29
II.3.Transformée en Ondelette .....	30
II.4.Méthode de Codage de Sous Bandes .....	30
II.4.1.L'Algorithme de Codage EZW .....	31
II.4.2.L'Algorithme de Codage SPIHT.....	31
II.5. Evaluation de la Qualité de Compression .....	32
II.5.1.Technique Subjectives .....	32
II.5.2.Technique Objectives .....	32
II.5.2.1.Erreur Moyenne Quadratique (MSE) .....	32
II.5.2.2.Rapport Signal sur Bruit crête (PSNR) .....	33
III.Partie Pratique .....	33
III.1. But de TP .....	33
III.2.Compression des Images Médicales par EZW .....	33
a) Script MATLAB N°1.1.....	33
b) Travail Demandé sur la partie 1 (EZW) .....	34
III.3. Compression des Images Médicales par SPIHT.....	35
c) Script MATLAB N°1.2.....	35
d) Travail Demandé sur la partie 2 (SPIHT) .....	35

## **TP N °5 : Cryptage des Images Médicales par l'Algorithme de Chiffrement par Flot Asynchrone**

I. Partie Théorique .....	37
I.1. Introduction .....	37
I.2. Généralités sur le Cryptage .....	37
I.2.1. Cryptage et Décryptage .....	37
I.2.2. Cryptographie .....	37
I.2.3. Cryptosysteme .....	38
I.2.3.1. Cryptographie à clefs Privés .....	38
I.2.3.2. Cryptographie à clefs Publiques.....	38
I.3. Cryptage d'image Médicale .....	38
I.4. Algorithme de chiffrement par Flot Asynchrone .....	39
II. Partie Pratique .....	40
II.1. Script MATLAB Préliminaire .....	40
II.2. Partie d'exécution de Programme .....	41
II.3. Partie d'Analyse de Sécurité .....	42
II.3.1. Analyse des Histogrammes .....	42
II.3.2. Calcul d'Entropie .....	42
II.3.3. Analyse de la Corrélation .....	42

## **TP N°6 : TATOUAGE ROBUSTE DES IMAGES MEDICALES PAR LA TRANSFORMEE EN ONDELETTE**

I. Introduction .....	44
II. Etude Théorique .....	44
II.1. Définitions et Propriétés générales des processus de Tatouage.....	44
II.1.1. Définitions.....	44
II.1.2. Processus d'insertion de la Marque.....	45
II.1.2.1. Schéma d'insertion.....	45
II.1.2.2. Formalisme.....	45
a) le tatouage Fragile.....	45
b) le tatouage Robuste.....	46
II.2. Algorithme de Xie.....	46
III. Partie Simulation .....	48
III.1. Objectif de TP.....	48

III.2.Implémentation Sous MATLAB.....	48
III.2.1. Analyser la propriété d'imperceptibilité.....	52
III.2.2. Choix de seuil de décision.....	52
III.2.3. Evaluation de la propriété de fragilité par rapport aux attaques.....	52
<b>ANNEXE</b>	54
<b>REFERENCES BIBLIOGRAPHIQUE</b>	57



# TP n°1: Systeme de Gestion Base de Donnée « Open Office »

## I. Introduction

L'informatique évolue vers le traitement de masses d'informations de plus en plus grandes dans des environnements répartis géographiquement où doivent cohabiter des matériels hétérogènes. Dans ce contexte, les bases de données sont utilisées de façon intensive pour de nombreux domaines d'application tels que le domaine médical, les administrations ou les associations. Les applications concernées par l'utilisation d'un SGBD (Système de Gestion de Base de Données) possèdent des caractéristiques différentes tant au niveau du volume de données concernées qu'au niveau de la complexité de ces données et des traitements informatiques à réaliser. Néanmoins, le regroupement des données dans une base de données gérée par un système de gestion de base de données apporte de nombreux avantages dans la plupart des cas d'utilisation. Il a été mis au point la notion de base de données répondant aux questions suivantes :

- L'accès aux données selon les multiples critères ;
- L'intégration des données ;
- La relation entre les données.

La notion qui remplace avantageusement celle de fichiers.

- L'ordre dans le stockage de données ;
- L'utilisation simultanée des données par différents utilisateurs.

## II. Quelques définitions :

Le concept de Base de Données (BDD) est apparu vers 1960, face au nombre croissant d'informations que les entreprises devaient gérer et partager :

### **Base de données (BDD) :**

Un ensemble organisé d'informations avec un objectif commun. Peu importe le support utilisé pour rassembler et stocker les données (papier, fichiers, etc.), dès lors que des données sont rassemblées et stockées d'une manière organisée dans un but spécifique, on parle de base de données. Plus précisément, on appelle base de données un ensemble structuré et organisé permettant le stockage de grandes quantités d'informations afin d'en faciliter l'exploitation (ajout, mise à jour, recherche de données). Bien entendu, dans le cadre de ce TP, nous nous intéressons uniquement aux bases de données informatisées.



- ✚ **Base de données informatisée** est un ensemble structuré de données enregistrées sur des supports accessibles par l'ordinateur, représentant des informations du monde réel et pouvant être interrogées et mises à jour par une communauté d'utilisateurs .
- ✚ **Un Système de Gestion de Base de Données (SGBD)** est un système informatique permettant de créer et de manipuler des bases de données.

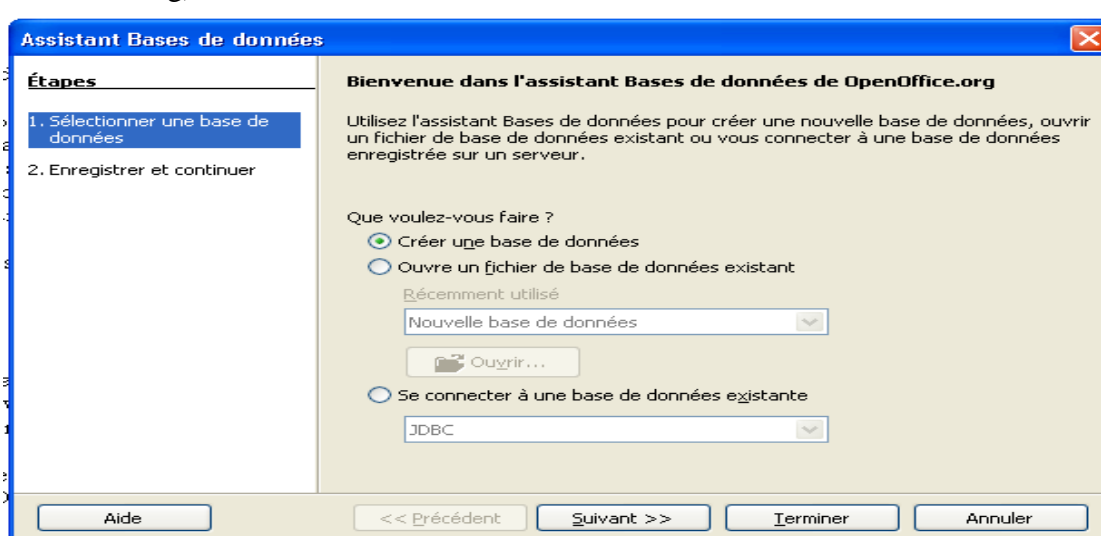
L'objectif d'une base de données est le stockage structuré des données et la mise à disposition d'outils permettant d'afficher ces données de manière pertinente. Une base est constituée de tables.

Chaque table est constituée de champs, qu'on appelle aussi parfois colonnes. On peut envisager la structure d'une table (la description des champs qu'elle contient) comme un modèle de fiche à remplir. D'une certaine manière, une table peut être vue comme une feuille de classeur : les champs comme colonnes et les enregistrements comme lignes. Mais il-y-a une différence fondamentale : dans une table, le type de contenu des colonnes est indiqué dans la structure de la table, et est le même pour toutes les lignes d'une colonne. Chaque fois que l'on saisit une nouvelle fiche, on crée un enregistrement. Un champ est un espace réservé aux données auquel on a imposé certaines règles.

### III.Objectifs : Dans ce TP, vous allez apprendre à :

- Créer une base de données médicale sous Base d'Open Office.Org.
- Modifier la structure de votre base de données//structure d'une table (la description des champs qu'elle contient).
- Éditer les données de la base de données.
- Interroger une base de données.

**III.1) Ouverture de Base d'OOO :** Lancer OpenOffice.Org Base (Démarrer – Open Office.Org). L'assistant s'ouvre :



**Figure I.1 :** création de la Base de donnée de openOffice





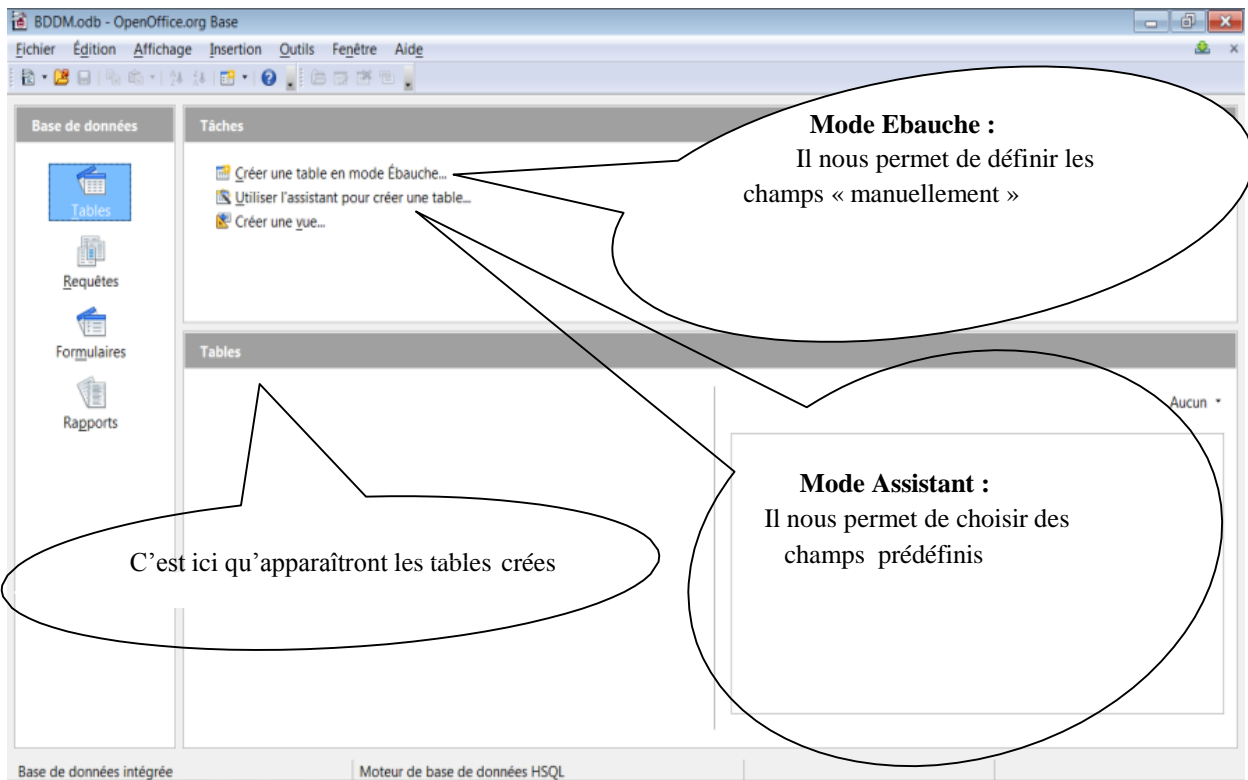
- Sélectionnez **Créer une base de données** puis **Terminer**
- Enregistrez sous le nom **BDDM** dans votre dossier personnel.

Votre base de données sera enregistrée dans un seul fichier avec l'extension **.odb**

A la fin de ces opérations, vous aurez la fenêtre suivante:

Vous trouverez dans cette fenêtre tous les objets de la base de données :

- Les tables
- Les requêtes
- Les formulaires
- Les rapports



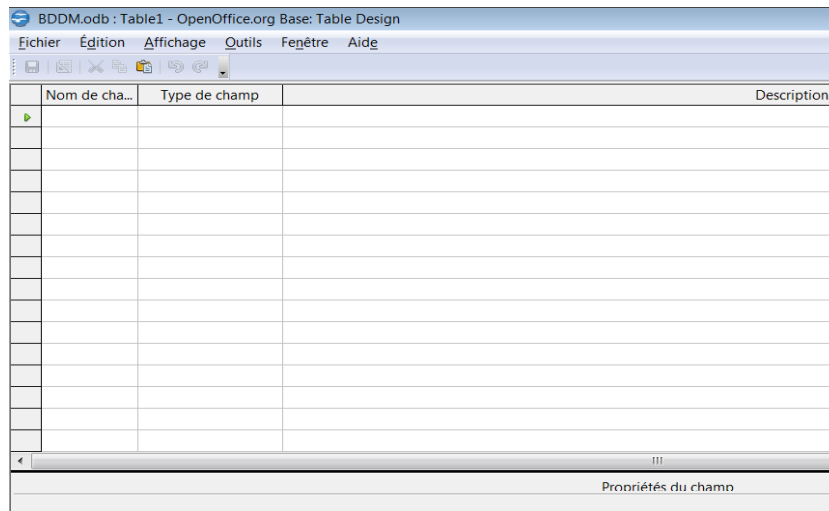
**Figure I.2 :** Création des tables de la Base de donnée de openOffice



**III.2) Création des tables :** Pour créer une table, vous devez définir tous les champs qu'elle comporte.

- Pour pouvoir choisir nous-même le **nom** et le **type** des **champs** à créer, nous allons utiliser le mode **Ebauche**.

- **Attention :** pour le **nom des champs**, évitez les accents et les espaces !



**Figure I.3 :** Définir les champs des tables de la Base de donnée de openOffice

➤ **Définition de la clef primaire :** cliquez droit sur la ligne et choisir *Cle primaire*.

**III.3) Définition des liens entre les tables :** Dans Base, les relations ont pour rôle d'établir des liens entre les tables d'une base de données.

Les deux conditions suivantes sont requises pour définir une relation entre 2 tables :

1. Les 2 tables doivent avoir au moins 1 champ en commun
2. Chaque enregistrement doit pouvoir être identifié de manière unique.

**Création d'une relation entre les tables :** Allez dans *Outil – Relation*, **Ajoutez** les 2 tables puis fermez la fenêtre *Ajouter des tables*, Repérez les champs à liés.

Allez dans *Insertion – Nouvelle relation* sélectionnez les tables puis les champs concernés.


**III.4) Insertion de données :** Pour insérer des données, il suffit de double-cliquez sur la table concernée.

**Le domaine d'application** de ce TP est d'informatiser la gestion des séjours des patients d'un Etablissement hospitalier, ainsi que la gestion des interventions effectuées par les médecins. Jusqu'à présent, cette gestion s'est effectuée à l'aide des fiches suivantes :



### Etablissement hospitalier

#### Gestion des séjours et interventions



<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td><b>PATIENT</b></td></tr> <tr><td>No Matricule:</td></tr> <tr><td>Nom:</td></tr> <tr><td>Prénom:</td></tr> <tr><td>Adresse :</td></tr> </table>	<b>PATIENT</b>	No Matricule:	Nom:	Prénom:	Adresse :	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td><b>SEJOUR</b></td></tr> <tr><td>No Séjour:</td></tr> <tr><td>Date Arrivée:</td></tr> <tr><td>Date Sortie:</td></tr> <tr><td>No Chambre :</td></tr> </table>	<b>SEJOUR</b>	No Séjour:	Date Arrivée:	Date Sortie:	No Chambre :									
<b>PATIENT</b>																				
No Matricule:																				
Nom:																				
Prénom:																				
Adresse :																				
<b>SEJOUR</b>																				
No Séjour:																				
Date Arrivée:																				
Date Sortie:																				
No Chambre :																				
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td><b>INTERVENTION(S):</b></td></tr> <tr> <td>Code intervention:</td> <td>Description:</td> <td>Date:</td> <td>Code Médecin:</td> <td>Nom:</td> <td>Prénom:</td> </tr> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </table>		<b>INTERVENTION(S):</b>	Code intervention:	Description:	Date:	Code Médecin:	Nom:	Prénom:												
<b>INTERVENTION(S):</b>																				
Code intervention:	Description:	Date:	Code Médecin:	Nom:	Prénom:															

Figure 1.4 : exemple d'application : Gestion des séjours des patients d'un Etablissement hospitalier, ainsi que la gestion des interventions effectuées par les médecins

- Nous supposons qu'un patient occupe la même chambre pendant toute la durée de son séjour.
- A part des informations concernant les médecins, qui se trouvent déjà sur les fiches, on désire stocker dans la base de données le numéro de téléphone et la spécialité de chaque médecin.
- Les interventions sont identifiées par un code et une description. L'établissement hospitalier dispose d'une liste d'interventions prédéfinies.

**VI. Travail à Réaliser :** Créer la base de données qui permettra de stocker les informations concernant les entités.

**Tableau 1.1 : Dictionnaire des données sur les Patients**

Description de la donnée	Nom du champ	Type de données	Taille
Numéro de Sécurité Sociale du patient	NSS	Nombre [NUMERIC]	14
Nom	Nom	Texte [VARCHAR]	30
Prénom	Prenom	Texte [VARCHAR]	30
Adresse	Adresse	Texte [VARCHAR]	50
Civilité	Civilite	Texte [VARCHAR]	1
Date de naissance	Date_Naiss	Date [DATE]	–

**Tableau 1.2 : Dictionnaire des données sur les Médecins**

Description de la donnée	Nom du champ	Type de données	Taille
Code médecin	Code_m	Nombre [NUMERIC]	4
Nom	Nom	Texte [VARCHAR]	30
Prénom	Prenom	Texte [VARCHAR]	30
Numéro de téléphone	N_TEL	Texte [VARCHAR]	14
Spécialité	SPECIALITE	Texte [VARCHAR]	25

**Table: « Patient » :**

NSS	NOM	PRENOM	Adresse	Civilite	DateNaiss
70123	Zaghdoudi	Ali	03,rue de l'independance	m	03/01/70
82321	Boulimi	Nadjet	N° 40,bloc B	m	23/01/82
90231	Djebali	Nacer	78, cité 30 logts	c	30/12/90

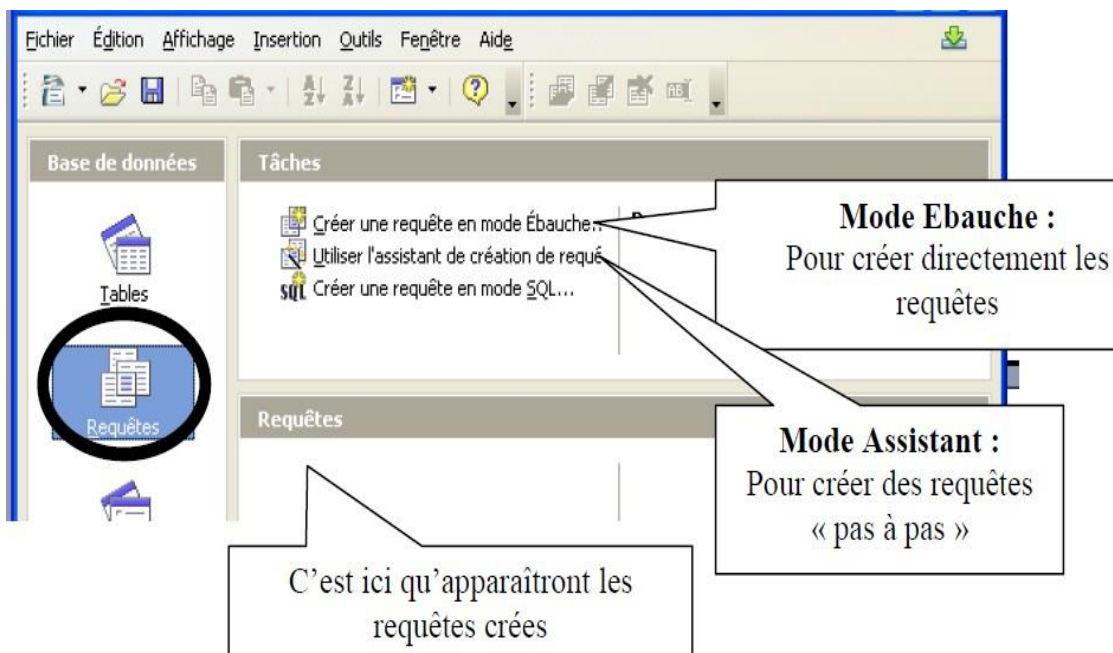
**Table « Médecin » :**

code_m	nom	prenom	N_TEL	SPECIALITE
58	SAHRAOUI	DJALLAL	07-77-77-77-77	Pneumologie
81	BENTOUIR	RIADH	06-66-66-66-66	Cardiologie
102	HADDAM	MOHAMED	05-55-55-55-55	Orthopédie

**Table « Intervention » :**

code_interv	description	date	code_medecin	nom_m	prenom_m	nss
12	Cardiopathie congénitale	23/11/13	81	BENTOUIR	RIADH	70123
15	Os naviculaire	03/01/14	102	HADDAM	MOHAMED	82321
17	Valve cardiaque	13/02/14	81	BENTOUIR	RIADH	90231

**V. Travail demandé : ‘ ‘ Interrogation de la base de données’ ’**



**Figure I.5 :** Définir les requêtes de la Base de donnée de openOffice



1. Créez une requête permettant d'afficher les noms, les prénoms et les N° des téléphones des médecins ayant la spécialité « « Pneumologie ».
2. Créez une requête permettant d'afficher la liste des médecins ayant effectué des interventions le 03 janvier 2014.
3. Créez une requête permettant d'afficher la liste des patients qui ont eu une intervention d'« Os naviculaire ».
4. Quelles sont les coordonnées (nom et adresse) des patients ayant « Dr.BENTOUIR » comme médecin intervenant ?



## TP n°2 : Manipulations SQL

### I. Introduction

La plupart des logiciels de gestion de données nous offre la possibilité de créer, interroger et modifier aisément nos bases de données grâce à des outils graphiques. On verra également qu'à ces outils peut s'ajouter la possibilité d'écrire l'ensemble des requêtes SQL permettant de réaliser les tâches nécessaires à la gestion de bases de données médicale.

### II. Partie théorique :

#### II.1. Qu'est-ce que SQL ?

SQL (sigle de Structured QueryLanguage, en français langage de requête structurée) est un langage informatique normalisé servant à exploiter des bases de données relationnelles. La partie langage de manipulation des données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données relationnelles.

Outre le langage de manipulation des données, la partie langage de définition des données permet de créer et de modifier l'organisation des données dans la base de données, la partie langage de contrôle de transaction permet de commencer et de terminer des transactions, et la partie langage de contrôle des données permet d'autoriser ou d'interdire l'accès à certaines données à certaines personnes.

Créé en 1974, normalisé depuis 1986, le langage est reconnu par la grande majorité des Systèmes de Gestion de Bases de Données Relationnelles (abrégé SGBDR) du marché.

SQL a différentes fonctions :

- ✓ **langage de définition et de validation de données** : pour créer, modifier et supprimer des tables dans une base de données, ou encore pour définir des valeurs par défaut pour certaines zones et règles de contrôle pour l'encodage,
- ✓ **langage de manipulation de données**: pour sélectionner, modifier, insérer, combiner, trier ou supprimer des données dans les tables d'une base de données ou encore pour lier des tables entre elles via des zones clés,



- ✓ **langage de contrôle d'accès aux données:** pour définir les permissions accordées aux différents utilisateurs de la base de données.

**Il permet :**

- ✿ La maintenance des tables (création, modification de la structure, suppression) ; il s'agit de la subdivision DDL (Data DefinitionLanguage) du SQL. Nous ne verons qu'une petite partie de cette subdivision.
- ✿ La manipulation des données contenues dans les tables : ajouter des enregistrements, les modifier, les supprimer ; c'est la subdivision DML (Data Modification Language) du SQL.
- ✿ La gestion des droits d'accès (qui ne sera pas traitée dans ce document) : subdivision DCL (Data Control Language).
- ✿ d'éviter des problèmes d'incohérence dans les données : TCL (subdivision Transaction Control Language) ; nous aborderons succinctement ces instructions.
- ✿ Il existe enfin un « embedded SQL » que nous n'aborderons pas.

**II.2. Comment est construite une instruction SQL ?**

Une instruction SQL est une chaîne de caractères, comprenant des mots-clés (qui indiquent au moteur de la base ce qu'il doit faire) et des paramètres sur lesquels il va travailler. Ces paramètres peuvent être des identificateurs (de tables, d'enregistrements, de requêtes, ...) ou des expressions.

**Tableau II.1 : Les commandes SQL**

Catégorie	Commandes SQL	
<i>Description des données</i>	<b>CREATE</b>	Création de tables
	<b>ALTER</b>	Modification de tables
	<b>DROP</b>	Suppression de tables
<i>Manipulation des données</i>	<b>INSERT</b>	Insertion de lignes dans une table
	<b>UPDATE</b>	Mise à jour de lignes dans une table
	<b>DELETE</b>	Suppression de lignes dans une table
<i>Contrôle des données (DCL)</i>	<b>GRANT</b>	Attribution de droits d'accès
	<b>REVOKE</b>	Suppression de droits d'accès
	<b>COMMIT</b>	Prise en compte des mises à jour
	<b>ROLLBACK</b>	Suppression des mises à jour
<i>Interrogation des données</i>	<b>SELECT</b>	Interrogations diverses





### II.3. Les contraintes d'intégrité

Une contrainte d'intégrité est une clause permettant de contraindre la modification de tables, faite par l'intermédiaire de requêtes d'utilisateurs, afin que les données saisies dans la base soient conformes aux données attendues. Ces contraintes doivent être exprimées dès la création de la table grâce aux mots clés suivants :

- ✚ PRIMARY KEY
- ✚ FOREIGN KEY
- ✚ REFERENCES
- ✚ DEFAULT
- ✚ NOT NULL
- ✚ UNIQUE
- ✚ CHECK

### II.4. Types de donnée (Tableau II.2)

Type de donnée	Syntaxe	Description
Type alphanumérique	CHAR(n)	Chaîne de caractères de longueur fixe <b>n</b> ( $n < 16383$ )
Type alphanumérique	VARCHAR(n)	Chaîne de caractères de <b>n</b> caractères maximum ( $n < 16383$ )
Type alphanumérique	TEXT	chaînes de caractères de longueur variable
Type numérique	DECIMAL(n)	Nombre de <b>n</b> chiffres
Type numérique	SMALLINT	Type numérique Entier signé de 16 bits (-32768 à 32757)
Type numérique	INTEGER	Entier signé de 32 bits (-2E31 à 2E31-1)
Type numérique	FLOAT	Nombre à virgule flottante
Type horaire	DATE	Date sous la forme 16/07/99
Type horaire	TIME	Heure sous la forme 12:54:24

### II.5. Syntaxes pour la description des données

#### ✚ Création d'une Table:

```
CREATE TABLE `Table1`
(`NumTable1` INT PRIMARY KEY,
`colonne2` TEXT(25),
`colonne3` DATE,
`colonne4` DOUBLE,
`NumTable2` INT REFERENCES Table2);
```

- ✚ Suppression d'une table :                    « DROP TABLE Nom\_de\_la\_table »
- ✚ Suppression des données d'une table :   « TRUNCATE TABLE Nom\_de\_la\_table »



### + Suppression de colonnes :

```
ALTER TABLE Nom_de_la_table  
DROP COLUMN Nom_de_la_colonne
```

### + Ajout de colonnes :

```
ALTER TABLE Nom_de_la_table  
ADD Nom_de_la_colonne Type_de_donnees
```

### + Modification de colonnes :

```
ALTER TABLE Nom_de_la_table  
CHANGE Nom_de_la_colonne TO nouveau nom colonne Type_de_donnees
```

### + Modification du type de données d'une colonne :

```
ALTER TABLE Nom_de_la_table  
MODIFY Nom_de_la_colonne Type_de_donnees
```

### + Changement du nom d'une table :

```
RENAME TABLE Ancien_Nom TO Nouveau_Nom
```

## II.6. MySQL : L'outil PHPMyAdmin :

### II.6.1. Qu'est-ce que PHPMyAdmin ?

PHPMyAdmin est une interface écrite en PHP permettant de gérer les données d'un serveur MySQL sous forme "graphique". Par des interactions avec des clics de souris, il permet de choisir une table, choisir un enregistrement à éditer, etc.

The screenshot shows the PHPMyAdmin interface for a 'localhost' server. The 'Créer une base de données' section is active, with the 'utf8\_unicode\_ci' character set selected. A callout box on the left says 'Ecrivez le nom de la BDDM à créer' and points to the text input field for the database name. Another callout box on the right says '2. Sélectionnez le nom de la base à manipuler' and points to the 'Interclassement' dropdown menu. The interface also displays server version information and various management options like 'Afficher l'état du serveur' and 'Processus'.

Figure II.1 : Interface graphique de MySQL/PHPMyAdmin

Cet écran se divise en 2 parties : gauche et droite.

↳ **Partie gauche:**

A gauche, il y a 4 icônes et un menu déroulant. Les icônes permettent respectivement de :

- HOME : revenir sur cette page
- SQL : exécuter une ou des requêtes SQL sur la base de données
- HELP : Aide PHMyAdmin
- Doc SQL : Documentation SQL

En dessous se présente un menu déroulant qui liste toutes les bases de données présentes sur le serveur. Si l'hébergeur n'autorise qu'une seule base de données, il se peut que ce menu déroulant ne soit pas là.

Toujours dans la colonne de gauche, un long espace vide en bas : cet espace sera rempli, lors du choix d'une base, par une liste de liens qui seront toutes les tables de la base.

↳ **Partie droite:**

La partie droite sur l'accueil présente la version de MySQL disponible, ainsi que des liens pour créer une base de données, ou gérer les utilisateurs (lien "privilèges"). Selon les permissions du compte et les restrictions éventuelles de l'hébergeur, certains liens peuvent ne pas apparaître.

✓ **Sélection d'une base de données :**

Choisir une base dans le menu déroulant à gauche. Son nom apparaît en tête de liste en dessous, suivi de la liste de toutes les tables :

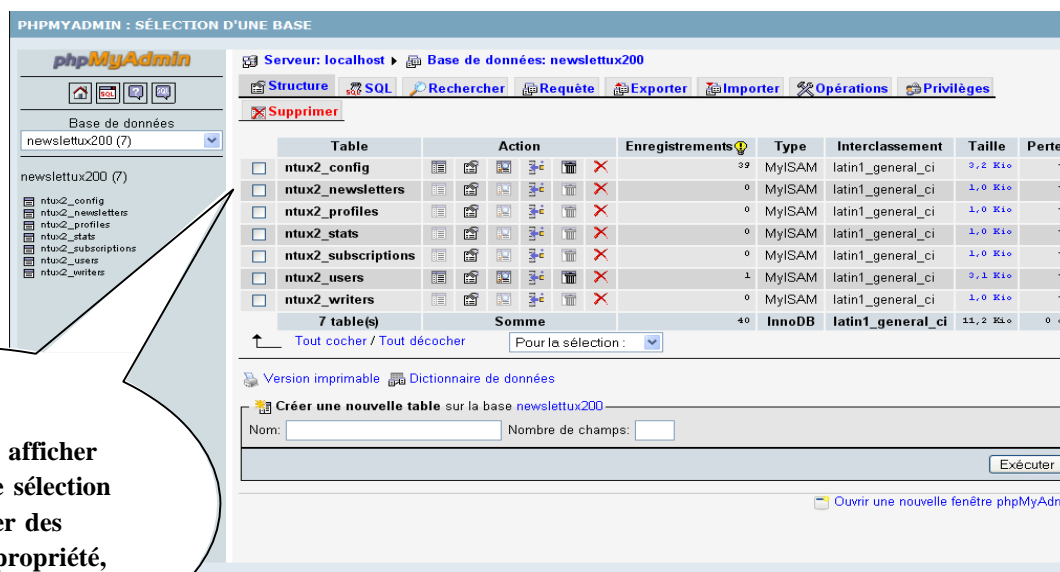


Figure II.2 : Sélection d'une Base de donnée PHMyAdmin

✓ **Structure d'une Table :**

Lorsqu'on clique, dans la colonne de gauche, sur le nom d'une table, ou bien à droite, sur l'icône "Structure" en regard du nom de la table, on arrive à l'écran Structure d'une table :

The screenshot shows the MySQL table structure interface with several callouts:

- Propriétés des attributs de la table**: Points to the table fields table.
- Quelques actions rapides : affichage, sélection, insertion d'un enregistrement, vidage, suppression**: Points to the top toolbar with buttons like 'Afficher', 'Sélectionner', 'Insérer', 'Vider', and 'Supprimer'.
- Quelques actions sur les attributs : modifier, supprimer ; plus les contraintes : clé primaire et unique ; et y mettre un index**: Points to the 'Action' column of the table fields table.
- Créer une nouvelle clé sur X attributs**: Points to the 'Créer une clé sur' section at the bottom left.
- Modifier ou supprimer plusieurs attributs en même temps**: Points to the 'Modifier' and 'Supprimer' buttons in the 'Pour la sélection' section.
- Quelques statistiques et infos**: Points to the 'Statistiques' section on the right side.

Figure II.3 : Structure de table de la Base de donnée MySQL/PHPMyAdmin

✓ **Manipuler des données :**

Sous PHPMyAdmin, la manipulation de données (ajout, édition, suppression) se fait via l'onglet "SQL" présent au-dessus de la structure de la table dans la colonne de droite de PHPMyAdmin.

The screenshot shows the PHPMyAdmin interface with the 'SQL' tab selected. The main area is a text input field for entering SQL queries, with an 'Exécuter' button below it. The left sidebar shows the database structure, and the top navigation bar includes options like 'Structure', 'SQL', 'Rechercher', 'Requête', 'Exporter', 'Importer', 'Opérations', and 'Privileges'.

Figure II.4 : Manipulation de données sous PHPMyAdmin

## ✓ Requête SQL

Pour effectuer une requête SQL, cliquer sur l'opération « SQL » en haut de la page d'accueil de la table concernée (ou de la page d'accueil de la base). La page qui s'ouvre alors présente une zone de saisie où vous devez écrire la requête à exécuter « 1 » (Voir figure ci-dessous).

Pour afficher par exemple, les nom, prénom et date de naissance des patients de sexe masculin, saisir la requête SQL suivante :

```
SELECT Nom_Patient, Prenom_Patient, DDN FROM tab_patient WHERE Sexe=1
```

Il est possible de saisir plusieurs requêtes en précisant quel caractère les sépare « 2 » (point-virgule par défaut). Pour afficher le résultat de la requête, cliquer sur « Exécuter » « 3 »

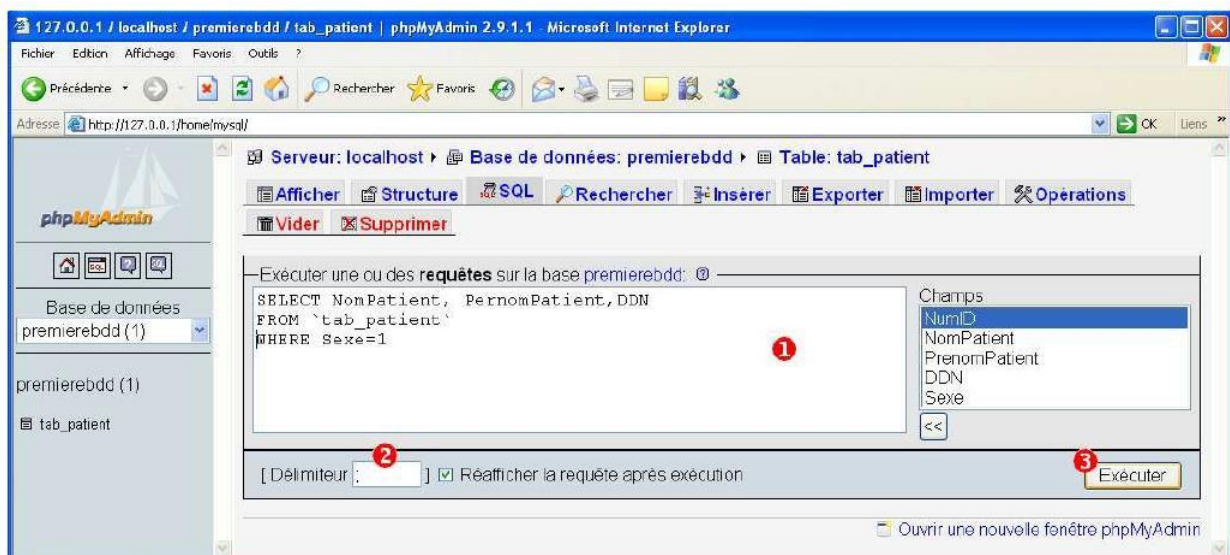


Figure II.5 : Requetes SQL sous PHPMysqlAdmin

Cette présentation du logiciel phpMyAdmin vise à vous présenter un autre logiciel de gestion de bases de données qu'Access, afin que vous puissiez constater que les connaissances que vous avez acquises vous serviront pour n'importe quel logiciel de gestion de bases de données. Moyennant le temps nécessaire à la prise en main du logiciel, vous serez opérationnel quel que soit le matériel que l'on vous demande d'utiliser.

## II.7. Avantages de SQL :

- ✚ **Rapide** : Le serveur MySQL est très rapide. Des tests de performances sont disponibles sur le site de MySQL
- ✚ **Facile à utiliser** : MySQL est beaucoup plus simple à utiliser que la plupart des serveurs de bases de données commerciaux.



- ✚ **API diverses** : On peut effectuer diverses opérations sur une base MySQL en utilisant des interfaces écrits en C, Perl, C++, Java, Python, PHP.
- ✚ **Connexion et Sécurité** : MySQL dispose d'un système de sécurité permettant de gérer les personnes et les machines pouvant accéder aux différentes bases.
- ✚ **Portabilité** : MySQL tourne sur divers systèmes tels que : Unix, Windows, Linux ou OS/2.
- ✚ **Distribution ouverte** : Les sources étant fournies, il est possible d'améliorer MySQL.

### III. Partie pratique

#### III.1. Installation de l'environnement de travail

Il est téléchargeable ici : <http://phpmyadmin.sourceforge.net>

**III.2. But de ce TP** : Dans ce TP, nous exprimons les principales instructions en SQL de MySQL (Création de tables, insertion de données, modification de données, suppression de données, ...etc), dans le but de **créer une base de données médicale** pour la gestion d'un hôpital qui contient 4 tables ,et qui sera **améliorée** par la suite dont nous ajoutons d'autre tables avec plus de données.

#### III.3. Description des données

Soit le schéma relationnel (Gestion d'un hôpital) :

- MEDECIN (num\_med, nom)
- PATIENT (num\_dossier, nom, adresse, ref\_med)
- ATTEINT (id, ref\_dossier, ref\_maladie, date)
- MALADIE (num\_maladie , nom\_maladie)

Les attributs soulignés forment la clé primaire de la relation où ils se trouvent.

Les attributs dont le nom est composé du préfixe « ref\_ » référencent les clés primaires.

Le sens de chaque attribut est exprimé par son nom. Le type des attributs est suffisamment évident pour ne pas être spécifié.

#### III.4. Travail demandé dans la séance de TP:

Exprimez en SQL les opérations suivantes :

1 – **Création de tables** : Créez les quatre tables constituant la base de données en définissant le nom et le type des attributs composant chaque table, ainsi que les clés primaires.





2 – **Insertion de données** : Insérez les enregistrements suivants dans la base de données nouvellement créée :

- MEDECIN : 10876, boukli ; 10877, oujdi ; 10878, fandi

- PATIENT : 1, ali, kifene, 10876 ; 2, mostapha, imama, 10876 ; 3, ahmed, chetouane, 10877

- ATTEINT : 1, 1, 100, 2004 ; 2, 2, 101, 2005 ; 3, 3, 102, 2004 ; 4, 1, 100, 2005

- MALADIE : 100, Otite ; 101, Grippe ; 102, Angine ; 103, Bronchite

3 – **Modification de données** : Modifiez l'enregistrement (ligne ou tuple) de la table MEDECIN qui contient le nom fandi : remplacer fandi par benosmane.

4 – **Suppression de données** : Supprimez l'enregistrement de la table MALADIE correspondant à la maladie Bronchite.

5 – **Ajout et suppression de colonne** : Ajoutez la colonne prénom avant la colonne nom à la table MEDECIN. Puis, supprimez cette colonne.

6 – **Requêtes de sélection** :

a) Quelles sont les villes de nos patients ?

b) Quelles sont les coordonnées (nom et adresse) du patient dont le numéro de dossier est 02 ?

c) Quelles sont les coordonnées (nom et adresse) des patients soignés par le Dr boukli ?

d) Quelles sont les maladies ayant atteint les patients en 2004 ?

e) Quels sont les patients (noms et numéros de dossiers) ayant été atteints par la grippe ?

7 – **Modification de nom de table** : Renommez la table MALADIE en PATHOLOGIE.

8 – **Suppression de données d'une table** : Supprimez en une opération SQL l'ensemble des données de la table PATHOLOGIE.

9 – **Suppression de table** : Supprimez la table PATHOLOGIE

10- **Amélioration de la base de données** : Améliorez le programme précédent afin qu'il gère toutes les exceptions possibles dans un centre hospitalier/ hôpital. Concevez un jeu de tests permettant de vérifier que votre programme est robuste et que la gestion des exceptions

fait bien son travail. Nous allons Créer d'autre tables tel que (chambre, hospitalisation, infirmier, service ,..etc) constituant la nouvelle base de données médical pour la gestion d'un hôpital ,et Insérez les nouveaux enregistrements cité dans les tables suivants :



Figure II.4 : présentation de la BDD et ces tables sous PHPMyAdmin

Table « chambre »

CODE_SERVICE	NB_CHAMBRE	NB_LITS/CHBR
110	4	5
100	6	4
500	7	4
440	8	4
550	9	3
200	10	6
300	12	3
400	15	7
111	16	2
220	20	8

Table « infirmier »

NUMERO	NOM	PRENOM	CODE_SERVICE
11	BOUDALIA	BOUBKER	400
22	TABET	ABDELLATIF	500
33	DERRAR	LAMIA	300
44	TRARI	MADJID	100
55	BRIXI	NORIA	440
66	MEBARKI	BACHIR	110
77	MECHRI	FOUZIA	200
88	FAKIRI	KADER	111
99	BOUMEDIENNE	RAHIM	220
111	BERHIL	KARIMA	550



**Table « Médecin »**

NUMERO	NOM	PRENOM	ADRESSE	TEL	SPECIALITE
1	DALI	RACHID	KIFENE,TLEMCEN	777111478	CARDIOLOGIE
5	MEZOUAR	ASMAA	BIROUANA,TLEMCEN	668374308	PEDIATRIE
10	BENAHBIB	RYAD	LES DALIAS,TLEMCEN	554786512	GENICOLOGIE
15	ZIANE	MOSTAPHA	HNAYA,TLEMCEN	661457526	PNEUMO-PHTISIOLOGIE
20	BENMANSOR	RAFIK	CHETOUANE,TLEMCEN	772146439	NEPHROLOGIE
25	BEMRAH	NABIL	OIJLIDA,TLEMCEN	559176439	CHIRURGIE GENERALE
30	GUELLIL	BOUMEDIENNE	SIDI SAID,TLEMCEN	666167349	DERMATOLOGIE
35	ARIS	HADJER	SIDI CHAKER,TLEMCEN	775420103	ONCOLOGIE
40	KORTI	FOUZI	BEL HORIZON,TLEMCEN	558413679	TRAUMATOLOGIE
45	WADAH	NACERA	ABOU TECHFIN,TLEMCEN	664197649	OPHTALMOLOGIE

**Table « hospitalisation »**

NO_MALADE	DATE	CODE_SERVICE	NO_CHAMBRE	NO_LIT
10	2015	110	15	5
19	2010	300	10	2
20	2012	400	2	5
22	2014	220	8	5
25	2015	200	8	1
28	2015	500	2	3
31	2013	220	7	2
32	2011	111	6	2
35	2009	440	5	4
39	2014	550	3	4
40	2013	220	5	1
45	2001	100	6	2
51	2015	110	7	3
55	2013	300	1	4
60	2005	400	11	2
66	2012	100	9	5
70	2014	111	15	2
88	2011	440	2	4





**Table « soigne »**

NUMERO	NOM	PRENOM	SEX	DATE_NAISS	NO_ASSURANCE
10	ZERROUKI	KARIM	H	1970	70123
19	BENYAHYA	NADJET	F	1982	82321
20	LAKERMI	AMINE	H	1948	48231
22	FELLAH	SABAH	F	1988	82335
25	MEFTAH	IMENE	F	2012	80935
28	KADOURI	KAMEL	H	1965	65894
31	BENCHLIL	KHEIRA	F	1977	77385
32	LATRACH	AMINE	H	1978	78124
35	GUENDOOUZ	YAHYA	H	1981	81666
39	LAARIBI	SALIHA	F	1950	50115
40	KACIMI	RABEA	F	1988	88564
45	KERNAF	WAFAA	F	1990	90256
51	SAHLI	YASMINE	F	1978	78521
55	MCHERNEN	NASRO	H	1960	60214
60	KARAOUZAN	TOUFIK	H	1994	64456
66	BEKHCHI	MOHAMMED	H	1950	50125
70	TCHOUAR	YASSINE	H	2013	84147
88	BENOSMAN	ZOLIKHA	F	1940	40723

**Table « malade »**

NO_DOCTEUR	NO_MALADE
1	45
1	66
5	25
5	70
10	22
10	31
10	40
15	10
15	51
20	19
20	55
25	20
25	31
25	60
30	28
30	66
35	32
35	70
40	35
40	88
45	39

**Table « service »**

CODE_SERVICE	NOM	BATIMENT	CHEF
100	CARDIOLOGIE	A	MEZIANE
110	PNEUMO-PHTISIOLOGIE	A	ZIANE
111	ONCOLOGIE	A	GHOUMARI
200	PEDIATRIE	B	BENDEDDOUCHE
220	GENICOLOGIE	B	BELARBI
300	NEPHROLOGIE	C	BENMANSOR
400	CHIRURGIE GENERALE	D	ABI AYAD
440	TRAUMATOLOGIE	D	KHERBOUCH
500	DERMATOLOGIE	F	BOUDGHENE S
550	OPHTALMOLOGIE	F	WADAH



### III.5 REQUETES « interrogation de la nouvelle base de donnée médicale en SQL »:

1. Quels sont les services se trouvant dans le bâtiment ‘A’ ?
2. Qui est le chef de service de ‘Cardiologie’ ?
3. Afficher les noms et les prénoms des malades opérés par Dr Bemrah ?
4. Afficher les noms et les prénoms des infirmiers du bâtiment B :
5. Combien de chambre contient le service de pédiatrie ?
6. Quelles sont les malades hospitalisées en gynécologie ?
7. Afficher le code\_service, nb de chambre et nb de lits des services du bâtiment « F » .
8. Afficher nom/prénom et la spécialité des médecins du bloc « A » .
9. Afficher les noms, prénoms et dates de naissance des malades femmes :
10. Donner le NO\_CHAMBRE et NO\_LIT des patients suivis par Dr KORTI :
11. Afficher les noms, prénoms, sex et n° d’assurance des patients hospitalisés
12. Afficher les noms, prénoms et sex des malades ayants moins de 40 ans.
13. Afficher le nom, prénom et date de naissance des enfants hospitalisés.
14. Afficher les noms des services dont le nombre de chambres est inférieur à 10 .
15. Afficher les noms des médecins et leurs spécialités qui ont suivis des patients hommes âgés moins de 66 ans
16. Afficher les noms, prénoms et date de naissance des malades hospitalisés en néphrologie en 2011 .
17. Afficher les noms, prénoms et date de naissance des malades hospitalisés en néphrologie entre 2009 et 2014 .
18. Dans quel service travaille l’infirmier n° 22 ?
19. Quel est le nom, le prénom et le n° d’assurance des patients hommes hospitalisés en pneumo-phtisiologie en 2015 ?



## TP N°3 : DICOM Works

### I. Introduction

Le domaine de l'imagerie médicale a beaucoup évolué depuis ses débuts au courant des années 1970. On a vu au cours des années l'émergence de plusieurs modalités (rayons X, ultrasons, résonance magnétique...) utilisées dans divers services de la médecine. Cette évolution a inévitablement attiré les manufacturiers d'équipements électriques et électroniques à s'intéresser aux équipements médicaux car ils voyaient là un marché prometteur pour l'avenir. Chaque constructeur offrait donc une large gamme d'équipements.

Un problème est malheureusement survenu avec la vulgarisation des équipements : la communication entre équipements de différentes marques était impossible. Ceci entraînait donc une complexité des systèmes d'information des établissements médicaux. C'est là que l'urgence d'une normalisation des équipements et des données médicales est apparue. Parmi les différentes normes créées par ces différents organismes, la plus aboutie et la plus prépondérante est la norme DICOM. Ce format est adopté par les cardiologues, les dentistes, les anatomopathologistes et les gastro-entérologues.

### II. PARTIE THEORIQUE:

#### II.1. Transformation DICOM des images :

« Pourquoi un format spécifique pour les images médicales ? »

Dans un Service de Radiologie qui produit plusieurs milliers d'images chaque jour, ainsi, un scanner, travaillant au rythme de 3 patients par heure produit environ 150 images par heures, et il n'est pas possible de classer ces images dans un format courant de type JPEG ou GIF car il aurait un risque de pertes des données démographiques de l'image, (nom du patient, type d'examen, hôpital, date d'examen, type d'acquisition etc...). Le format DICOM permet de rendre unique chaque image produite et de leur associer des informations spécifiques. Ainsi chaque image est autonome, si elle est perdue, reproduite ou renommée, il est toujours possible d'identifier formellement son origine, le patient, la date, la série d'où elle provient, les paramètres d'acquisition etc...



La transformation DICOM ou encore DICOMISATION va consister à placer les données d'image brutes dans un fichier conteneur DICOM contenant toutes les métadonnées indispensables à l'identification du patient, de l'origine du cliché (modalité initiale, date et heure de l'examen, injection de contraste ou non, orientation particulière, et de sa technique de numérisation (type de numériseur utilisé, méthode de compression des données, nouvelle matrice, dynamique d'image, nom de la personne ayant numérisé les images, date et heure de numérisation, etc..).

Tableau III.1 : Vue générale de la norme DICOM

<b>L'identification de la machine</b>	<ul style="list-style-type: none"> <li>• Date d'examen</li> <li>• Type d'examen</li> <li>• Fabricant de la machine</li> <li>• Hôpital ou institution</li> <li>• Identification de la machine (numéro, type )</li> </ul>
<b>Les informations sur le patient :</b>	<ul style="list-style-type: none"> <li>• Nom</li> <li>• Identification</li> <li>• Date de naissance</li> <li>• Sexe</li> </ul>
<b>Les informations sur l'acquisition de l'information :</b>	<ul style="list-style-type: none"> <li>• Bolus</li> <li>• Kilovolts</li> <li>• Inclinaison du statif</li> <li>• Epaisseur de coupe</li> <li>• etc.. variable suivant le type d'examen</li> </ul>
<b>Les informations en rapport avec l'examen :</b>	<ul style="list-style-type: none"> <li>• Série,</li> <li>• Orientation du patient</li> <li>• Plan de références</li> <li>• Nombres d'images dans l'acquisition</li> <li>• Commentaires</li> </ul>
<b>Les informations concernant l'image elle-même et la façon dont elle est codée.</b>	<ul style="list-style-type: none"> <li>• Largeur</li> <li>• Hauteur</li> <li>• Codage de l'image (niveaux de gris, couleur ..)</li> <li>• Compression</li> <li>• Représentation numérique de l'image</li> </ul>
<b>Les pixels ;</b>	<ul style="list-style-type: none"> <li>• Représentation numérique de l'image elle même.</li> </ul>

La transformation DICOM des images va attribuer à chaque image des numéros d'instance, de série et d'examen uniques (respectivement SOP instance UID, SeriesInstance UID et StudyInstance UID), qui l'identifieront définitivement dans n'importe quel système PACS ( Picture Archiving and Communication System) ou de téléradiologie.

Tableau III.2 : Propriétés des fichiers DICOM

<b>SOP Class UID</b>	Identifie le type de Service auquel est destiné l'image. <ul style="list-style-type: none"> <li>• Storage Service Class</li> <li>• Query/Retrieve Service Class.</li> </ul>
<b>Study Instance UID</b>	Identifie un examen entier , en temps et lieu.
<b>Series Instance UID</b>	Identifie une série d'images au sein de l'examen.
<b>SOP Instance UID ou Image UID</b>	Identifie l'image associée au fichier.



Cette transformation est une mise à la norme des images. Elle diminue grandement le risque d'erreur médicale car si l'image est correctement identifiée au départ, elle est définitivement associée au patient, à la différence d'une image brute (de type JPEG par exemple) pour laquelle le titre peut être modifié ou confondu pendant les multiples étapes de sa transmission.

La numérisation des images au format DICOM s'inclut dans certains protocoles de traçabilité et de sécurisation des données.

### **III. Partie pratique :**

**III.1. But de ce TP :** On utilise le logiciel DICOM Works pour effectuer certain traitement sur des images médicales.

#### **III.2. Interface du logiciel DICOM Works:**

Le DICOM WORK est un logiciel qui est inclus dans une solution de téléradiologie permettant la dicomisation, la télétransmission, la gestion et la lecture d'images à distance destiné aux radiologues, en vue d'importer des différents formats bureautiques (jpeg, tiff, dicom, avi, bmp, etc...) ou exporter des données en html, ftp, mail, PowerPoint, etc.,

La visionneuse permet de zoomer, déplacer, filtrer, lisser etc. Il est aussi possible de sauvegarder, compresser les archives.

Ces logiciels permettant de saisir dans un formulaire des données simples, concernant le patient, l'examen, la série d'images, et de compiler l'ensemble dans un ou plusieurs fichiers DICOM.

Il permet de se connecter à un système d'informations médicales (SIH) ou radiologiques (RIS) et de sélectionner un patient dans le système d'informations hospitalier et de créer un nouvel examen (unique) à la date convenue pour qu'il apparaisse dans une liste de travail. Cette liste de travail est accessible dans le logiciel de numérisation qui peut en extraire les données afin de les intégrer directement dans les fichiers DICOM. Les fichiers ainsi créés seront parfaitement synchronisés avec le système d'informations hospitalier.

Le traitement d'image en radiographie est essentiellement utilisé pour le but d'avoir une meilleure qualité de visibilité des organes sur l'image et permettre ainsi de lever des indéterminations. Les médecins sont aujourd'hui confrontés à l'utilisation d'ordinateurs pour faire un diagnostic plus complet lors de la suspicion d'une anomalie anatomique sur l'image.



### III.3 Travail à réaliser :

Nous utilisons le logiciel DICOM Works afin d'effectuer certain traitement sur des images, chaque fois que cela est nécessaire, une modification de la taille des images, leur mise en gris, la normalisation et l'égalisation des niveaux de gris.

Cliquer sur Répertoire DICOM, et enregistrer le Répertoire des images IRM.

Les principaux traitements réalisés par DICOM sont (voire affichage) :

1- Changement de niveau de gris (différentiation d'organe de densité proche).

2 - Changement du contraste et de la luminosité (amélioration de la qualité d'image).

3 - Inversion de niveau de gris.

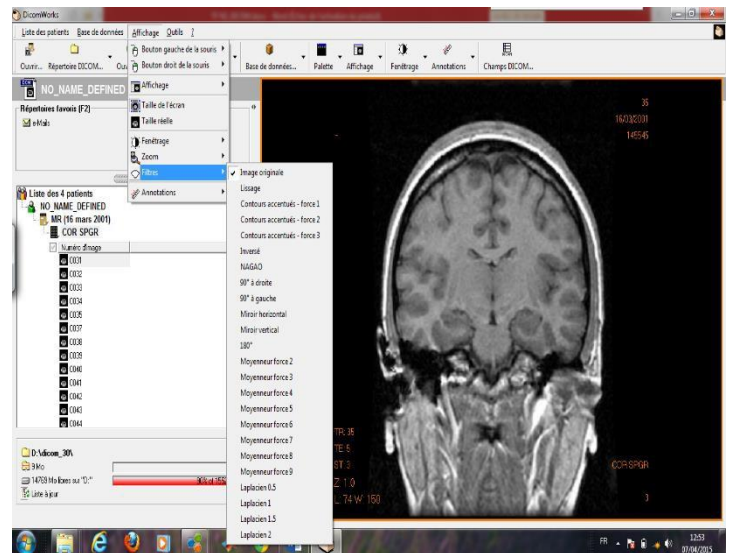
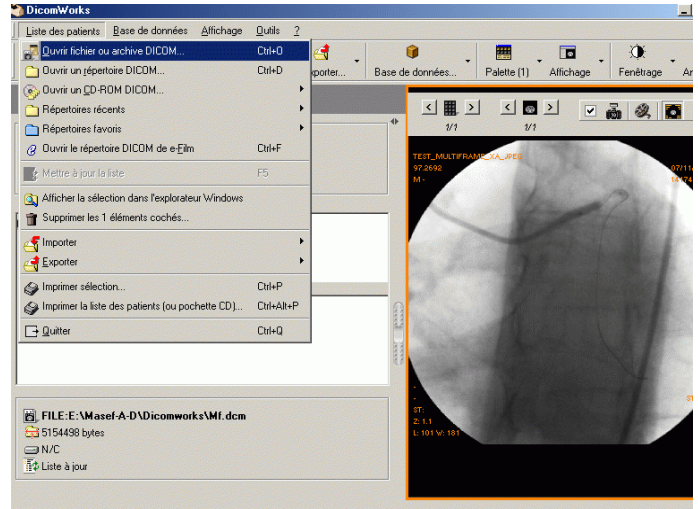
4- Application de filtre de convolution pour accentuer le contour. Appliquer un filtre laplacien, filtre moyen, filtre de NAGAO,..etc).

5- Appliquer un zoom sur l'image sans déformation.

6-Tracé l'histogramme des pixels de l'image IRM (voire Annotation statistique)

7- Enregistré notre image sous forme JPEG, et en fichier vidéo (exporter la série en AVI).

8-Donner moi toutes les Informations de fichier DICOM ' cliquer sur outil \_éditer le champ DICOM' .





## TP n°4 : Compression des images médicales par des Codeurs de Sous Bande (EZW, SPIHT)

### I.1 INTRODUCTION

La compression d'image concerne plusieurs domaines de la science et de la technologie, notamment le domaine médical. En effet, les praticiens disposent d'une base de données en image très dense et très variée. L'utilisation conjointe de ces images et les examens cliniques et biologiques aide considérablement les médecins à affiner leur diagnostic. Cette quantité intense d'images nécessite des moyens de stockage de plus en plus importants, à cette allure le seuil de saturation est très vite atteint. Les études ont prouvé que le service de radiologie d'un hôpital peut produire plus de 20 téraoctets de données d'image par année.

Pour remédier à ces problèmes, la compression des images numériques est alors une solution incontournable.

L'origine de la préférence des médecins pour la compression sans perte par rapport à la compression avec pertes est d'éviter les erreurs médicales liées à une mauvaise reconstruction de l'image. En effet, le principal problème de la compression avec pertes pour les images médicales est dû au fait que des détails importants pourraient disparaître (d'autres pourraient éventuellement apparaître). Ces détails sont généralement des structures difficiles à discerner car elles entraînent de faibles changements de contraste. Ainsi par exemple, des images peuvent révéler des lésions à travers des détails potentiellement sensibles à la compression avec perte puisqu'ils sont petits et ont des bords faiblement définis (comme par exemple : des micro calcifications dans des mammographies, le contour d'un pneumothorax, un faible nodule d'une image pulmonaire, etc).

Pour autant, la compression avec pertes est plus que jamais à l'ordre du jour en imagerie médicale, et ce pour les raisons suivantes. Tout d'abord, les études bien que peu nombreuses ont montrées que les images médicales possédaient des tolérances (maximum taux de compression) à la compression avec perte.



Enfin la qualité de la compression des images médicales doit tenir compte des éventuelles distorsions pouvant influencer l'interprétation qualitative des images et la valeur des paramètres anatomiques ou fonctionnels reflétant l'état de l'organe étudié.

## II. PARTIE THEORIQUE

### II.1 METHODES DE CODAGE SANS PERTE

La notion de codage sans perte ((lossless) d'une source encore appelé codage entropique, a pour objectif d'atteindre une limite théorique du gain de compression caractérisée par l'entropie de la source.

Nous citons quelques codeurs sans perte par exemple : codeur entropique (codage de Huffman, codeur arithmétique, codage RLC (*Run Length Coding*), codeur lempel ziv...).

La méthode de codage sans perte reste limitée car elle donne un taux de compression qui ne dépasse pas 4% pour les images médicales. La compression avec pertes maîtrisées peut donc être une réponse à cette limitation.

### II.2. METHODES DE CODAGE AVEC PERTE

Les méthodes avec perte (lossy) ou irréversibles sont des méthodes qui tirent parti d'une corrélation (ou redondance) existante dans l'image. L'information perdue est due à l'élimination de cette redondance, ceci rend possible une compression plus importante. La perte d'information est toujours discutable et nous nous posons alors la question de la limite acceptable. Cette limite est définie par le type d'application, comme les images médicales par exemple. La quantification est un des mécanismes utilisé dans les algorithmes de compression, qui produit des pertes d'information.

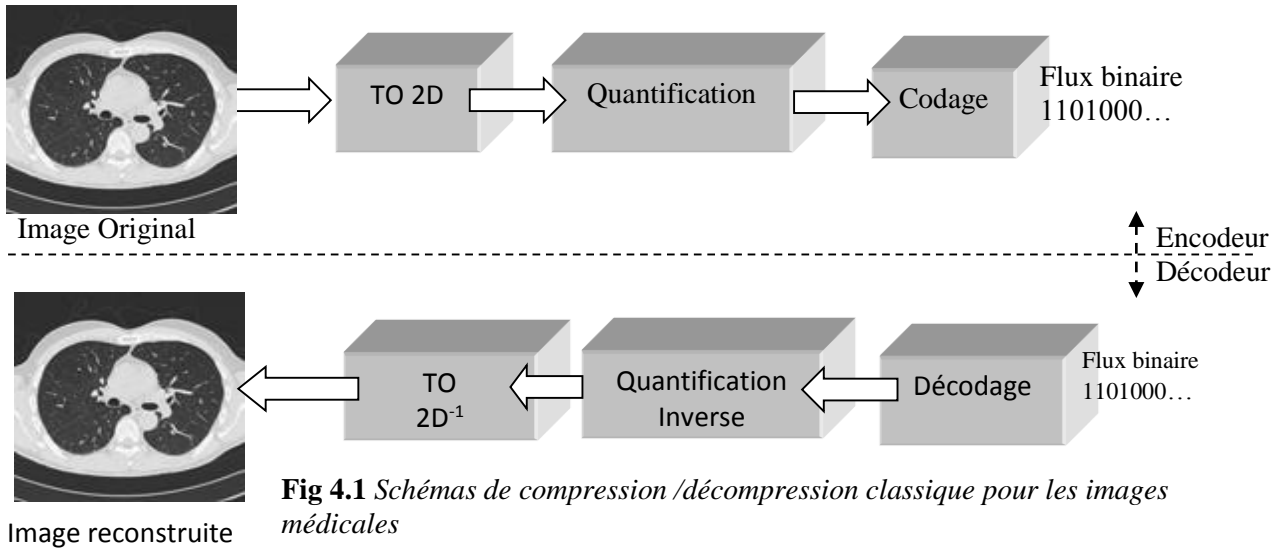
La figure (4.1) représente le schéma général d'un système de compression avec perte.

Dans un premier temps, afin de mieux compresser l'information, la source est transformée en groupe de coefficients. Les transformations les plus utilisées, que ce soit pour les images fixes ou les séquences d'images, sont la Transformée en Cosinus Discrète (DCT), la Transformée en Ondelettes Discrète (DWT) ou la décomposition Pyramidale.

Dans un second temps, les coefficients obtenus après la transformation sont quantifiés (tronqués). La phase de quantification introduit l'erreur dans le système de codage.

La dernière étape consiste à coder les coefficients quantifiés par le codage entropique.





### II.3 Transformation en ondelettes

La transformation en ondelettes permet d'obtenir une représentation temps-fréquence ou temps échelle. Elle a des propriétés d'adaptation ou de flexibilité très attrayantes notamment le choix des fonctions de bases des ondelettes (orthogonales ou non à support compact ou infini, ect...) et des paramètres de dilatation et translation. Les transformées en ondelettes conservent l'énergie du signal et possèdent notamment des algorithmes rapides, elles sont donc bien adaptées à la compression d'image.

L'intérêt des ondelettes par rapport aux sinus et aux cosinus se situe surtout à deux niveaux :

- Contrairement aux sinus et cosinus qui ne sont bien localisés qu'en fréquence, les ondelettes le sont également en temps. Par conséquent tout changement de fréquence dans la transformée en ondelettes ne produira de changements que sur une certaine partie du domaine temporel.

- Les ondelettes permettent de représenter de manière compacte un grand nombre de fonction : ainsi les fonctions formées de pics très prononcés nécessitent beaucoup moins d'ondelettes que de sinus/cosinus pour être représentées

Les ondelettes sont utilisées dans les deux catégories de techniques de compression que sont la compression sans perte et avec perte.

### II.4. METHODES DE CODAGE DES SOUS-BANDES

Le but des transformées en ondelette est de décorrélérer les données brutes de l'image représentées par ses pixels. Cette décorrélation n'est cependant pas parfaite et les coefficients obtenus après transformée restent dépendants statistiquement.



De plus, les transformées en ondelettes offrant naturellement une représentation progressive de l'image, il est intéressant de conserver cette propriété lors du codage des sous-bandes. Nous allons dans ce TP tester deux algorithmes EZW et SPIHT.

#### **II.4.1. L'algorithme de codage EZW (Embedded Zerotrees of Wavelet transforms):**

L'idée de base de cet algorithme qui est proposé par Shapiro est de trouver le meilleur ordre de transmission des coefficients de représentation en ondelettes.

Il est clair que la transmission des coefficients dans l'ordre décroissant de leur valeur absolue est la meilleure solution, puisque les coefficients les plus significatifs sont ceux dont la valeur absolue est la plus élevée. Shapiro proposa de transmettre les coefficients sous forme d'une suite de bits obtenue par enchaînement progressif des bits des coefficients les plus significatifs en commençant par les bits les plus importants. Cette nouvelle conception offre l'avantage à l'algorithme EZW de faire la transmission progressive d'image puisque le décodeur peut s'arrêter au niveau de n'importe quelle suite de bits. De surcroît nous aurons une meilleure image reconstruite avec cette suite de bits tronquée. Cet algorithme présente en plus l'avantage de ne nécessiter ni phase d'apprentissage, ni dictionnaire, ni l'information sur l'image source.

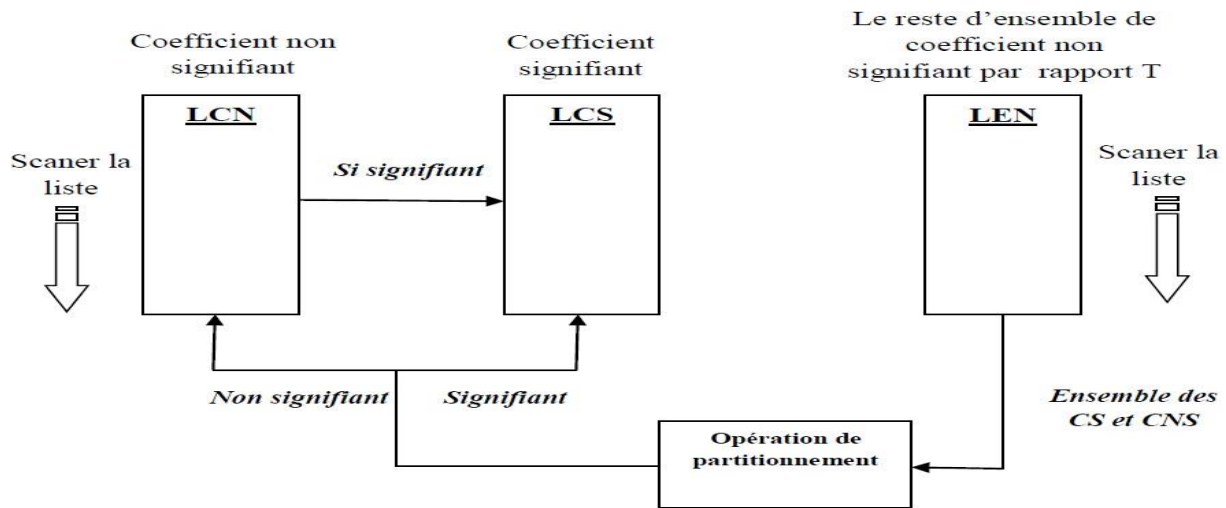
#### **II.4.2. L'algorithme de codage SPIHT :**

L'algorithme SPIHT ( Set Partitioning In Hierarchical Trees ) a été proposé par Saïd et Pearlman en 1996 pour la compression avec perte et sans perte . Il a été étendu au 3D pour la vidéo et pour la compression d'images volumiques .

La modification majeure de SPIHT par rapport à EZW réside dans la mise à jour d'une 3<sup>ème</sup> liste permettant de créer des ensembles non significatifs de grande taille. Ces ensembles non significatifs permettent de connaître l'état d'une descendance même si le coefficient n'est pas la racine d'un zérotree.

##### ***Algorithme de SPIHT :***

L'algorithme est cependant plus sophistiqué : contrairement à l'algorithme EZW qui n'utilise qu'un seul ensemble décrivant la signifiante des coefficients, le SPIHT utilise trois listes, de coefficients significatifs (LSP), de coefficients insignifiants (LIP) et d'ensembles insignifiants (LIS).



**Figure 4.2 Organigramme de l'Algorithme de SPIHT**

## II.5. EVALUATION DE LA QUALITE DE COMPRESSION

Les techniques irréversibles de compression modifient l'image en y introduisant une distorsion. Il faut donc évaluer le niveau de cette distorsion, qui permettra de contrôler la qualité des images reconstruites, d'évaluer et comparer les différentes approches. Dans la pratique, plusieurs techniques subjectives et objectives sont utilisées.

**II.5.1. Techniques subjectives :** Les mesures de la qualité subjective se fait par des tests psychophysiques ou par des questionnaires avec notations réalisés par des professionnels de la radiologie .Dans ce cas, on demande à un ensemble de radiologues de noter la qualité des images comprimées (généralement sur une échelle de 1 à 5) selon que celles-ci leur permettent ou non d'effectuer un diagnostic (ou une mesure particulière d'organe).

### II.5.2 Techniques objectives :

Les mesures objectives sont basées sur des critères mathématiques pour évaluer la qualité des images. Les critères de qualité utilisés pour mesurer les performances des instruments optiques sont, par exemple, le rapport signal/bruit (SNR), l'erreur quadratique moyenne (MSE).

#### II.5.2.1 .Erreur moyenne quadratique (MSE) :

Le plus simple paramètre de la mesure de qualité d'image est le MSE. La grande valeur de MSE signifie que l'image est de mauvaise qualité. Le MSE est défini comme suit:

$$MSE^2 = \frac{1}{M \times N} \sum_{i=1}^{i=N} \sum_{j=1}^{j=M} (I(i, j) - \hat{I}(i, j))^2 \quad 4.1$$

Avec  $I(i, j)$  : représente l'image originale,  $\hat{I}(i, j)$  : représente l'image dégradée .  $M$  et  $N$  sont le nombre de lignes et de colonnes.



### II.5.2.2. Rapport signal sur bruit crête (PSNR)

La faible valeur du PSNR signifie que l'image est de mauvaise qualité. Le PSNR est défini comme suit:

$$PSNR = 10 \log_{10} \left( \frac{\text{dynamique de l'image}}{MSE} \right)^2 \quad 4.2$$

Généralement une image est codée sur 8 bits. Elle est représentée par 256 niveaux de gris qui varient entre 0 et 255, l'étendu ou la dynamique de l'image est alors 255.

- Le PSNR d'une image couleur RGB est défini par l'équation:

$$PSNR = 10 \log_{10} \left( \frac{255^2 \times 3}{MSE_R + MSE_G + MSE_B} \right) \quad 4.3$$

## III. PARTIE PRATIQUE :

**III.1. But de TP :** On se donne comme but de réaliser une compression d'images médicale en niveau de gris par la transformée on ondelette utilisons des codeurs de sous-bandes tel que EZW et SPIHT.

### Préambule :

Les programmes nécessaires pour faire ce TP sont : « func\_DWT.m, func\_ezw\_enc.m, func\_huffman\_encode.m , func\_huffman\_decode .m ; func\_ezw\_dec.m ; func\_InvDWT .m ; func\_SPIHT\_Enc.m ; func\_SPIHT\_dec.m ». Ces fonctions sont réalisées par [Kanchi](https://www.mathworks.com/matlabcentral/fileexchange/19296-ezw--embedded-zero-tree-wavelet-) (<https://www.mathworks.com/matlabcentral/fileexchange/19296-ezw--embedded-zero-tree-wavelet->). Et (<https://www.mathworks.com/matlabcentral/fileexchange/4808-spiht>)

Les images médicales données par l'enseignant (voir annexe)

### III. 2. Compression des images médicales par EZW

Le programme MATLAB suivant permet de faire la compression d'image par le codeur EZW:

#### a) Script MATLAB N°1.1

```
>> clear all; close all; clc;
>> fprintf('----- Welcome to EZW Matlab Demo! -----\n');
>> infilename = 'Sagital_IRM.bmp';
>> outfile = 'Sagital_IRM_compressé.bmp';
>> img_orig = double(imread(infilename));
>> fprintf('----- Décomposition par la transformée ondelette -----\n');
>> n = size(img_orig, 1);
>> n_log = log2(n);
>> level = n_log; % niveau de décomposition d'ondelette
>> type = 'haar'; % type d'ondelette
```



```
>> [Lo_D,Hi_D,Lo_R,Hi_R] = wfilters (type);
%img_wavedata: wavelet coefficients of the input image
>> [img_wavedata, S] = func_DWT(img_orig, level, Lo_D, Hi_D);
% Partie de codage par EZW
>> fprintf('----- EZW Encoding -----\n');
>> ezw_encoding_threshold = 50; % le seuil de codage
>> [img_enc_significance_map, img_enc_refinement] = func_ezw_enc(img_wavedata,
ezw_encoding_threshold);
>> img_ezw_stream_bit = func_huffman_encode (img_enc_significance_map, img_enc_refinement);
>> fprintf('----- EZW Decoding -----\n');
>> treshold = pow2(floor(log2(max(max(abs(img_wavedata))))));
>> [img_dec_significance_map, img_dec_refinement] = func_huffman_decode(img_ezw_stream_bit);
>> img_wavedata_dec = func_ezw_dec(n, treshold, img_dec_significance_map, img_dec_refinement);
>> fprintf('----- Inverse Wavelet Decomposition -----\n');
>> img_reconstruct = func_InvDWT(img_wavedata_dec, S, Lo_R, Hi_R, level);
>> fprintf('The bitrate is %.2f bpp (with threshold %d in the encoding)\n',
length(img_ezw_stream_bit)/size(img_orig,1)/size(img_orig,2), ezw_encoding_threshold);
>> fprintf('----- parameter d\'évaluation -----\n');
>> Q = 255;
>> MSE = sum(sum((img_reconstruct-img_orig).^2))/size(img_orig,1)/size(img_orig,2)
>> fprintf('The psnr performance is %.2f dB\n', 10*log10(Q*Q/MSE));
```

**b) Travail demander sur la partie 1 (EZW) :**

Nous allons compresser 4 images médicales en niveau de gris de différentes modalité (IRM, CT, echo et mamographie ) par le codeur EZW :

1. Augmenter le niveau de décomposition de la transformée en ondelette de 1 jusqu'au 7 et calculer les paramètres d'évaluation. Quel est le meilleur niveau pour une bonne compression EZW
2. fixé le niveau idéal de la décomposition par ondelette (par exemple : level=3).tester ce algorithme pour différents type d'ondelette (db5 ;symlet3 ;coif5 ;bior1.3 ;bio4.4 et bior6.8) pour les 4 images médicales pour un seuil de 50.

**Tableau 1 :** compression d'image IRM par EZW avec un niveau ' ? '

Type d'ondelette	Db5	Sym3	Coif5	Bior1.3	Bior4.4	Bior6.8
MSE						
PSNR						

3. Quel est la meilleurs ondelette pour compressé ces quatre images médicales ?pourquoi ?
4. On varie maintenant le seuil de 10 à 50 avec le meilleur type d'ondelette choisit dans la question 2 et on va remplir ce tableau pour les quatre images médicales testées.



**Tableau 2 :** Compression d'image IRM par EZW avec un niveau 'Q1', type d'ondelette 'Q2'

seuil	10	20	30	40	50
Rate (bpp)					
MSE					
PSNR					

### III. 3. Compression des images médicales par SPIHT

Le programme Matlab suivant permet de faire la compression d'image par SPIHT:

**c) Script MATLAB N°1.2 à terminer par les étudiants (regarde Script MATLAB N°1.1)**

```
>> fprintf('----- Load Image -----\n');
>> rate = 1; % la valeur de bitrate est égale à 1bpp
>> OrigSize = size(Orig_I, 1);
>> max_bits = floor(rate * OrigSize^2);
>> OutSize = OrigSize;
>> image_spiht = zeros(size(Orig_I));
>> [nRow, nColumn] = size(Orig_I);
>> fprintf('----- Décomposition par la transformée ondelette -----\n');
>> -----
>> fprintf('----- SPIHT Encoding -----\n');
>> img_enc = func_SPIHT_Enc(I_W, max_bits, nRow*nColumn, level);
>> fprintf('----- SPIHT Decoding -----\n');
>> img_dec = func_SPIHT_Dec(img_enc);
>> fprintf('----- Reconstruction par la transformée ondelette -----\n');
>> -----
>> fprintf('----- Parameter d'évaluation et l'affichage -----\n');
>> -----
>> -----
```

**d) Travail demandé sur la partie 2 (SPIHT)**

1. Augmenter le niveau de décomposition de la transformée en ondelette de 1 jusqu'au 7 et calculer les paramètres d'évaluation. Quel est le meilleur niveau (level) pour une bonne compression par SPIHT ? pourquoi ?



2. Fixé le niveau idéal de la décomposition par ondelette (par exemple : level=3). Tester cet algorithme pour différents type d'ondelette (db5 ; symlet3 ; coif5 ; bior1.3 ; bio4.4 et bior6.8) pour les 4 images médicales pour un bit rate de 1 (TC=75%).
3. Quel est la meilleurs ondelette pour compressé ces quatre images médicales ?pourquoi ?
4. Fixé le meilleur type d'ondelette choisit dans la question 2.on varie maintenant le bit rate de 0.125 :0.25 :0.5 :0.75 :1 : et on va calculer les paramètres d'évaluation pour les quatre images médicale testées ?commenter les résultats obtenus ?





# TP n°5: Cryptage des images médicales par l'algorithme de Chiffrement par flot asynchrone

## I. PARTIE THEORIQUE :

### I.1. Introduction

L'utilisation des réseaux informatiques pour la transmission d'informations médicales pose le problème de la sécurisation. Pour pallier à ce problème, des techniques de chiffrement de messages plus ou moins robustes ont été développées. Ces algorithmes utilisent des clefs de chiffrement et de déchiffrement soit identiques, soit différentes. Parmi les plus courantes, nous pouvons citer le chiffrement de Vigenère à une seule clef, l'algorithme DES à clefs secrètes et l'algorithme RSA à clefs publiques et privées .

Le but de ce TP est de donner une introduction à la cryptographie moderne utilisée dans la transmission et le stockage sécurisé de données dont nous examinerons un algorithme cryptographique appliqué aux images médicales.

### I.2. Généralités sur le cryptage

#### I.2.1. Cryptage et décryptage

Le cryptage ou (chiffrement) peut être défini comme une fonction réversible de transformation des données en envisageant la protection d'information contre toute prise de connaissance (confidentialité) ou de modification (intégrité) du contenu.

Le cryptage consiste à transformer une information selon un algorithme donné afin de la rendre illisible sans avoir appliqué l'algorithme inverse.

Le décryptage est logiquement l'opération inverse du cryptage.

#### I.2.2. Cryptographie

La cryptographie est la science qui utilise les mathématiques pour le cryptage et le décryptage de données.

Elle vous permet ainsi de stocker des informations confidentielles ou de les transmettre sur des réseaux non sécurisés (tels que l'internet), afin qu'aucune personne autre que le destinataire ne puisse les lire.





### I.2.3. Cryptosystème

#### I.2.3.1. rypographie à clefs privés

La cryptographie à **clefs privées**, appelée aussi cryptographie **symétrique** est utilisée depuis déjà plusieurs siècles. C'est l'approche la plus authentique du chiffrement de données et mathématiquement la moins problématique.

La clef servant à chiffrer les données peut être facilement déterminée si l'on connaît la clef servant à déchiffrer et vice-versa. Dans la plupart des systèmes symétriques, la clef de cryptage et la clef de décryptage sont une seule et même clef.

Les principaux types de cryptosystèmes à clefs privés utilisés aujourd'hui se répartissent en deux grandes catégories : les cryptosystèmes par flots et les crytosystèmes par blocs.

#### I.2.3.2. Cryptographie à clefs publiques

Tous les algorithmes évoqués jusqu'à présent sont symétriques en ce sens que la même clef est utilisée pour le chiffrement et le déchiffrement. Le problème essentiel de la cryptographie symétrique est la distribution des clefs : pour que  $n$  personnes puissent communiquer de manière confidentielle il faut  $n(n-1)/2$  clefs.

L'idée de base des cryptosystèmes à clefs publiques a été proposée dans un article fondamental de Diffie et Hellman en 1976. Le principe fondamental est d'utiliser des clefs de chiffrement et déchiffrement différentes, non reconstituables l'une à partir de l'autre :

- une clef publique pour le chiffrement
- une clef secrète (privée) pour le déchiffrement

Ce système est basé sur une fonction à sens unique, soit une fonction facile à calculer dans un sens mais très difficile à inverser sans la clef privée.

Le gros avantage de ce système est qu'il n'y ait pas besoin d'avoir partagé un secret au préalable pour s'échanger des messages cryptés. En revanche les implémentations de tels systèmes (RSA) ont un inconvénient majeur : leur lenteur par rapport aux chiffrements symétriques.

### I.3. Cryptage d'images médicales

Dans cette section, nous exposerons comment il est possible d'appliquer un algorithme de cryptage à des images médicales en niveaux de gris. Dans le cas de signaux médicaux 1D, les algorithmes standards de chiffrement peuvent être directement appliqués. Par contre, du fait de la notion de voisinage bidimensionnelle des images et de leur taille, ces algorithmes standard doivent être modifiés afin d'être applicables efficacement sur les images médicales.

L'objectif du chiffrement d'images est d'obtenir une image de même format et de taille égale au maximum à la taille de l'image originale.

#### I.4. Algorithme de chiffrement par flot asynchrone

Les algorithmes de chiffrement par flot sont composés de deux étapes : la génération d'une clef dynamique et la fonction de cryptage de sortie dépendant de la clef dynamique. Quand la clef dynamique est générée à partir de la clef et d'un certain nombre de digits précédemment crypté, l'algorithme de chiffrement par flot est dit asynchrone.

Les processus de cryptage et décryptage d'un chiffrement par flot asynchrone sont décrit figure 5.1, où  $g()$  est la fonction génératrice de la clef dynamique et  $h()$  la fonction de sortie de cryptage :

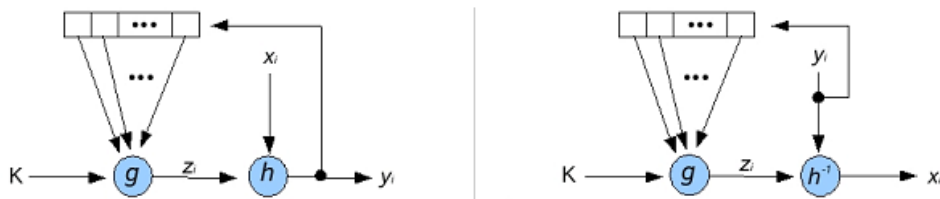


Fig. 5.1 : Cryptage et décryptage asynchrone.

$$\begin{cases} z_i = g(K, c_{i-t}, c_{i-t+1}, \dots, c_{i-2}, c_{i-1}) \\ c_i = h(z_i, m_i) \end{cases} \quad (5.1)$$

Où  $K$  est la clef,  $m_i$ ,  $c_i$  et  $z_i$  sont respectivement le  $i^{\text{ème}}$  texte clair, le texte chiffré et la clef dynamique. Nous pouvons remarquer équations (1) que la clef dynamique dépend des  $t$  digits précédents du texte chiffré.

Soit  $K$  une clef de longueur  $k$  bits  $b_i$ ,  $K = b_1 b_2 \dots b_k$ . L'unité de cryptage est le pixel (1 octet). La méthode réside dans le fait que pour chaque pixel de l'image le cryptage dépend du pixel original, de la valeur de la clef  $K$ , et des  $k/2$  pixels précédemment cryptés. Pour utiliser les équations (1) nous avons  $t = k/2$ . Pour chaque pixel  $p_i$  de l'image originale, nous calculons la valeur du pixel  $p'_i$  de l'image chiffrée en utilisant l'équation suivante :

$$\begin{cases} z_i = (\sum_{j=1}^{k/2} \alpha_j p'_{i-j}) \text{mod}(256) \\ p'_i = (z_i + p_i) \text{mod}(256) \end{cases} \quad (5.2)$$

avec  $i \in [0, \dots, N - 1]$  où  $N$  est le nombre de pixels de l'image,  $k$  est la longueur de la clef avec  $k \in [1, N]$ , et  $\alpha_j$  est une séquence de  $k/2$  coefficients générée à partir de la clef secrète  $K$ . Le principe de chiffrement est le même que celui illustré dans la figure 5.1.

Les équations (5.2) ont une récurrence d'ordre  $k/2$ , correspondant à la moitié de la longueur de la clef. Les coefficients  $\alpha_j$  sont des coefficients entiers compris entre  $-2$  et  $+2$  tels que :



$$\begin{cases} \alpha_j = \beta_j - 1 & \text{si } \beta_j \in \{0,1,2\} \\ \alpha_j = \pm 2 & \text{si } \beta_j = 3 \end{cases} \quad (5.3)$$

Avec  $\beta_j = 2b_{2j-1} + b_{2j}$ , où  $b_{2j-1}$  et  $b_{2j}$  sont deux bits voisins de la clef secrète  $K$ .

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

L'équation (5.4) présente la procédure de décryptage. Dans la procédure de décryptage, nous devons appliquer le processus inverse. Nous pouvons noter que la fonction génératrice de la clef dynamique est la même qu'à l'équation (2) :

$$\begin{cases} z_i = \left( \sum_{j=1}^{k/2} \alpha_j p'_{i-j} \right) \text{mod}(256) \\ p_i = (p'_i - z_i) \text{mod}(256) \end{cases} \quad (5.4)$$

## II. PARIE PRATIQUE :

Après avoir décrit en détail l'algorithme de chiffrement par flot asynchrone, nous appliquons notre méthode sur différentes images médicale en niveau de gris. Elles ont chiffrées avec une clef de 128 bits.

Le programme MATLAB suivant permet de faire le cryptage d'image médicale par flot asynchrone :

### **SCRIPT MATLAB Préliminaires:**

```
clc; clear all; close all;
%% image à marquer
nom=imgetfile;
image_a_crypter=(imread (nom));
[m,n,l]=size(image_a_crypter);
if l>1
    image_a_crypter=rgb2gray(image_a_crypter);
end
image_a_crypter_vec=reshape(image_a_crypter,1,m*n);
%% insérer la clef
taille_clef = input('entrez la taille de la clef: ');
clef=round(rand(1,taille_clef));
```



```
%% vecteur virtuel
vecteur_vrtuel=35*abs(round(randn(1,taille_clef/2)));
vecteur_vrtuell=vecteur_vrtuel;
%% clef dynamique
for j=1:taille_clef/2
    b(j)=2*clef((2*j)-1)+clef(2*j);
    if b(j)==0 || b(j)==1 || b(j)==2
        alpha(j)=b(j)-1;
    elseif b(j)==3
        alpha(j)=2;
    end
end
end
%% cryptage
for i=1:m*n
    zzz=uint8(alpha).*uint8(vecteur_vrtuel);
    zz=sum(zzz);
    z=mod(zz,256);
    image_cryptee_vec(i)=bitxor(z,image_a_crypter_vec(i));
    vecteur_vrtuel(1:end-1)=vecteur_vrtuel(2:end);
    vecteur_vrtuel(end)=image_cryptee_vec(i);
end
image_cryptee=reshape(image_cryptee_vec,m,n);
%% Partie de decryptage
for i=1:m*n
    ZZZ=uint8(alpha).*uint8(vecteur_vrtuell);
    ZZ=sum(ZZZ);
    Z=mod(ZZ,256);
    image_decryptee_vec(i)=bitxor(Z,image_cryptee_vec(i));
    vecteur_vrtuell(1:end-1)=vecteur_vrtuell(2:end);
    vecteur_vrtuell(end)=image_cryptee_vec(i);
end
end
image_decryptee=reshape(image_decryptee_vec,m,n);
taille_clef = input ('entrez la taille de la clef: ');
clef=round(rand(1,taille_clef));
```



%% crée un vecteur virtuel

```
vecteur_vrtuel=35*abs(round(randn(1,taille_clef/2)));
```

```
vecteur_vrtuell=vecteur_vrtuel;
```

%% insérer la clef dynamique

```
for j=1:taille_clef/2
```

```
    b(j)=2*clef((2*j)-1)+clef(2*j);
```

```
    if b(j)==0 || b(j)==1 || b(j)==2
```

```
        alpha(j)=b(j)-1;
```

```
    elseif b(j)==3
```

```
        alpha(j)=2; end
```

```
end
```

%% Partie de cryptage

```
for i=1:m*n
```

```
    zzz=uint8(alpha).*uint8(vecteur_vrtuel);
```

```
    zz=sum(zzz);
```

```
    z=mod(zz,256);
```

```
    image_cryptee_vec(i)=bitxor(z,image_a_crypter_vec(i));
```

```
    vecteur_vrtuel(1:end-1)=vecteur_vrtuel(2:end);
```

```
    vecteur_vrtuel(end)=image_cryptee_vec(i);
```

```
end
```

```
image_cryptee=reshape(image_cryptee_vec,m,n);
```

%% Partie de decryptage

```
for i=1:m*n
```

```
    ZZZ=uint8(alpha).*uint8(vecteur_vrtuell);
```

```
    ZZ=sum(ZZZ);
```

```
    Z=mod(ZZ,256);
```

```
    image_decryptee_vec(i)=bitxor(Z,image_cryptee_vec(i));
```

```
    vecteur_vrtuell(1:end-1)=vecteur_vrtuell(2:end);
```

```
    vecteur_vrtuell(end)=image_decryptee_vec(i);
```

```
end
```

```
image_decryptee=reshape(image_decryptee_vec,m,n);
```



## II.2. Partie d'exécution de Programme

1. Exécuter le programme et affiché l'image originale, image cryptée et l'image décryptée.
2. calculer le cryptage en utilisant l'équation (5.2) pour obtenir l'image cryptée.
3. Nous avons choisi le vecteur virtuel comme un bruit généré aléatoirement pour pouvoir crypter les  $k/2$  premiers pixels. Après décryptage, que constatez-vous ?

## II.3.Partie d'Analyse de sécurité

Un algorithme de cryptage est considéré comme sécurisé si un adversaire qui a intercepté des données chiffrées, n'a pas de techniques praticables permettant de retrouver des informations sur les données claires ou la clé utilisée, plus précisément, il doit résister aux attaques statistiques.

Dans ces conditions une série de tests est appliquée, sur plusieurs images médicales en utilisant une clé secrète de taille 128 bits. L'ensemble des résultats de ces tests donnera une idée du degré de robustesse de l'algorithme implémenté.

### II.3.1. .1. Analyse des histogrammes

L'histogramme d'une image représente la répartition des pixels en fonction de leur luminosité, pour cette raison, l'histogramme associé à l'image cryptée doit masquer cette répartition.

- ✚ Tracé les histogrammes de l'image originale ; Histogramme des valeurs de la clé dynamique  $z_i$  et l'Histogramme de l'image cryptée. Que remarquez-vous ?

### II.3.2.Calcul d'entropie

L'entropie est une mesure statistique du contenu d'information d'un message. Elle est donc adaptée à caractériser les images, chacune d'entre elles étant prise comme un message spécifique. Cette valeur est calculée à partir de la formule suivante :

$$h(k) = -\sum_{i=1}^n p(k_i) \log_2(p(k_i)) \quad (5.5)$$

Où  $p(k_i)$  désigne la probabilité de la valeur du pixel  $k_i$  dans l'image. L'entropie maximale

$h_{max} = \log_2(n)$ , la valeur idéale de  $h_{max}$  est 8bits/pixel.

- ✚ Calculer l'entropie de l'image originale et l'image cryptée ?

### II.2.3. Analyse de la corrélation



Pour tester la corrélation entre deux pixels adjacents horizontalement, verticalement et diagonalement d'une image cryptée, on calcule le coefficient de corrélation pour une séquence de pixels adjacents en utilisant les équations suivantes :

$$r_{xy} = cov(x, y) / (\sqrt{var(x)}\sqrt{var(y)}) \quad (5.6)$$

$$cov(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(y)) \quad (5.7)$$

$$\text{Avec } E(x) = \frac{1}{N} \sum_{i=1}^N x_i \text{ et } var(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2 \quad (5.8)$$

$cov(x, y)$  est la covariance,  $E(x)$  est l'espérance mathématique et  $var(x)$  est la variance. Pour une bonne sécurité, la corrélation de l'image cryptée doit être négligeable et très proche de zéro.

Calculée la corrélation de l'image crypté ?

Remplir le tableau suivant qui résume les valeurs des différentes mesures de corrélation et d'entropie, obtenues après les tests effectués sur les images originales (IO) et leurs versions cryptées (IC) exposées précédemment pour différente images médicale.

Interprétez ces résultats?

Tableau 5.1: mesures d'entropie et la corrélation.

Image	Entropie (IO)	Entropie (IC)	Corrélation (IC)
Scanner cérébral			
IRM cérébral 1			





## TP N°6 : TATOUAGE ROBUSTE DES IMAGES MEDICALES PAR LA TRANSFORMEE EN ONDELETTE

### I. Introduction

De nos jours, de plus en plus d'images numériques sont transférées sur les réseaux informatiques. Des études montrent comment des algorithmes de tatouage permettent la sécurisation des images médicales. Pour ce faire, les images peuvent être tatouées au niveau des codages source afin de faire remonter cette fonctionnalité au niveau des couches hautes (applications). De cette manière, les fonctionnalités tatouage d'images sont insérées au niveau d'un logiciel. La protection est alors assurée pendant la transmission des images médicales mais aussi pour l'archivage de ces données numériques. Nous allons présenter des solutions de sécurité pour les images médicales, basées sur des concepts avancés dans la section II.1. Ces solutions ont pour but d'assurer la confidentialité des données.

### II. Etude Théorique

#### II.1. Définitions et propriétés générales des processus de tatouage des images numériques

##### II.1.1. Définitions

Le tatouage d'image, que l'on peut sommairement décrire à l'aide de la figure 6.1, consiste à introduire, généralement de manière invisible, une information dans une image, puis à tenter de la récupérer après que l'image ait éventuellement subi des manipulations de natures variées ou non.

De la figure 6.1 le schéma illustre la définition précédente. Le signal d'entrée  $I$ , appelé signal hôte sera modifié par une application  $\mathcal{E}$ . Cette étape est l'insertion de la marque  $W$  dans  $I$ , avec l'intervention de la clef privée  $K$ . Le signal sortant  $I^*$ , est diffusé. Il est alors soumis à des transformations volontaire (attaque) ou non, cette version attaquée est notée  $I'$ . A la détection  $D$ , la marque est extraite (on obtient alors une marque  $W'$ ). Tous les processus de détection n'ont pas les mêmes entrées.

Nous allons maintenant présenter plus en détail les schémas d'implémentation et de détection de la marque.

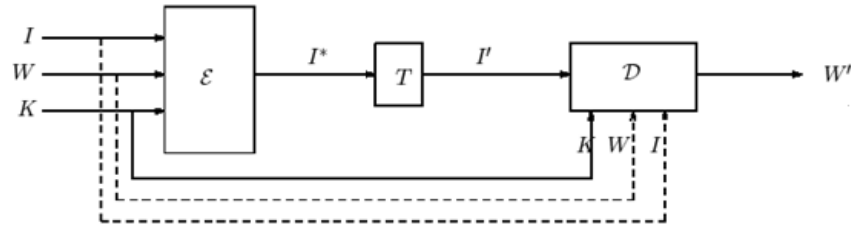


Fig. 6.1 : schéma général d'un processus de tatouage.

## II.1.2. Processus d'insertion de la marque

### II.1.2.1. Schéma d'insertion

La figure 6.2 présente le schéma général d'insertion de la marque. Une image hôte  $I$  est tatouée d'une marque  $W$  à l'aide d'une clé  $K$ . L'image résultante de  $\varepsilon; I^*$  l'application est perceptuellement similaire à  $I$  et  $W$  contient la marque.

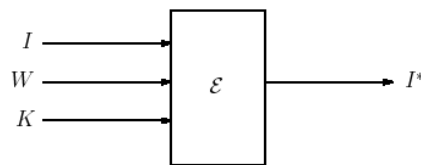


Fig. 6.2 : schéma général du processus d'insertion d'une marque

### II.1.2.2. Formalisme

La phase d'insertion génère en sortie une image tatouée  $I^*$ . Cette phase est modélisée par la fonction suivante :

$$I^* = \varepsilon(I, W, K) \quad (6.1)$$

Ce formalisme, très général, représente le processus d'insertion de la marque pour tous les processus de tatouage.

- Pour un tatouage robuste, si  $I^*$  est attaqué,  $W'$  doit correspondre à  $W$ .
- Pour un tatouage fragile,  $W'$  doit être significativement ou totalement différente à  $W$ , après des petites modifications (attaque) de  $I^*$ .

Le tatouage trouve plusieurs applications dans la protection des images Médicales.

**a) Le tatouage fragile:** permet de vérifier l'intégrité d'une image marquée. Il est très fragile aux modifications, et permet de vérifier que l'image n'a pas été retouchée et donc de l'authentifier.

b) **Le tatouage robuste** : dispose d'un large champ de théories et de résultats. Celui-ci cherche à préserver les données cachées face aux attaques. La marque doit donc être suffisamment résistante aux attaques afin de rester identifiable .

L'objectif est d'insérer directement dans l'image médicale les données pertinentes du patient d'une manière secrète et robuste. Ceci permettra en plus de faire transiter l'image seule, sans avoir besoin de l'accompagner d'un fichier textuel. Les méthodes utilisées doivent donc être robustes. Parmi les algorithmes de tatouages robustes présentés dans la littérature, nous avons choisi l'algorithme de Xie à implémenter et tester.

## II.2. Algorithme de Xie

Xie insère la marque dans l'image approximation d'une décomposition multi niveaux. Le niveau de décomposition dépend de la taille des informations que l'on souhaite insérer, de l'invisibilité et de la robustesse recherchées.

L'insertion s'effectue en faisant évoluer dans toute l'image approximation, une fenêtre  $b_j$  de taille 3x1 sans chevauchement. Pour chaque position de la fenêtre, les trois coefficients sont rangés dans l'ordre croissant de leur valeur. Ensuite, l'espace entre la valeur minimale  $b_1$  et la maximale  $b_3$  est divisé en intervalles de longueur  $S_\alpha$ :

$$S_\alpha = \alpha \frac{b_{(3)} - b_{(1)}}{2} \quad (6.2)$$

On obtient alors la configuration suivante:

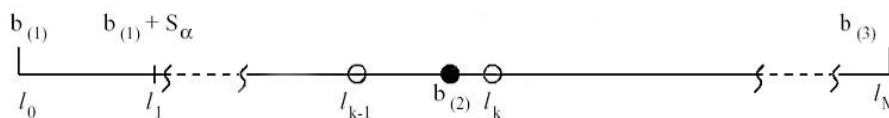


Fig. 6.3 : Ordonnancement des coefficients avant insertion. (algorithme de Xie)

Quatre cas peuvent alors se présenter, suivant que le bit à insérer ( $x$ ) soit "0" ou "1" et que  $k$  soit pair ou impair.

Si  $k$  est pair:

Si  $x = 0$ , alors  $b_{(2)}$  est mis à la valeur de  $l_k$

Si  $x = 1$ , alors  $b_{(2)}$  est mis à la valeur de  $l_{k-1}$

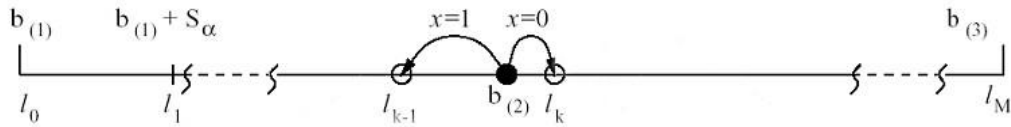


Fig. 6.4 : Cas où  $k$  est pair.

Si  $k$  est impair:

Si  $x = 0$ , alors  $b_{(2)}$  est mis à la valeur de  $l_{k-1}$

Si  $x = 1$ , alors  $b_{(2)}$  est mis à la valeur de  $l_k$

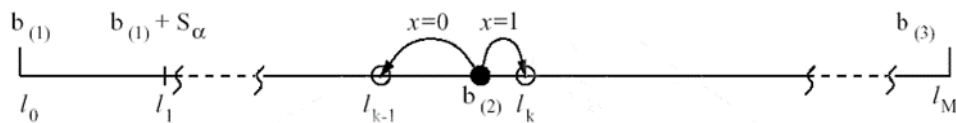


Fig. 6.5 : Cas où  $k$  est impair.

Enfin, les coefficients sont réinsérés à leur place initiale dans l'image approximation.

La détection s'effectue en réeffectuant, pour chaque position de la fenêtre (que l'on fait à nouveau évoluer dans toute l'image d'approximation à l'aide de la clé secrète), l'ordonnancement des coefficients et la division de l'espace entre les deux extrêmes. Il suffit alors de prendre le  $l_k$  le plus proche du  $b_{(2)}$ . Si  $k$  est pair, le bit inséré était "0", sinon c'était "1".

La détection ne nécessite donc ni l'image non marquée, ni la marque insérée. Cette méthode est donc *aveugle*.

Le mécanisme d'insertion de cet algorithme peut être réglé par certains paramètres qui modifient considérablement le nombre de bits de données insérés, la robustesse de la marque et l'impact visuel de l'insertion. Ces paramètres sont :

- le niveau de décomposition en ondelette ( De 1 à 4 par exemple)
- Le pas de quantification (déterminé par  $\alpha$ ).
- Il faut choisir la fenêtre  $b_{(j)}$  de telle manière, les trois coefficients sont différent entre eux. Ces trois paramètres combinés entraînent une palette de réglages très importante, et donc de résultats extrêmement divers.

### III. Partie Simulations

#### III.1. Objectif de TP

Notre but est la protection des données médicales. Pour cela il faut marquer l'image originale par des données utile du patient. Dans ce cas, le tatouage est robuste dans le but de résister aux modifications de l'image provenant de différentes attaques.

#### III.2. Implementation sous MATLAB

Pour implémenter cet algorithme de Xie il nous faut donc à développer 4 programmes :

Ces programmes ont été écrits en Matlab, c'est un langage rapide, facile et il contient beaucoup des fonctions prédéfinies, notamment pour le traitement des images. La figure 6.6 illustre l'espace de travail.

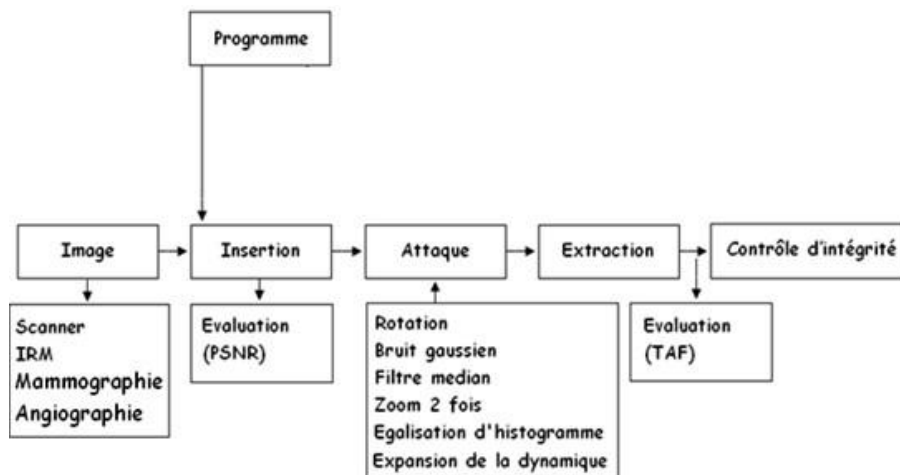


Fig. 6.6 : Architecture de l'espace de travail.

**1.Marque** : pour simuler une marque sous forme de damier .La marque insérée est : (lettre italique rouge) ; et le programme est donnée dans le Script N°6.1.

- Lieu d'examen : *Centre d'imagerie de Maghnia*
- Date d'examen : *10/10/2012*
- Nature : *Tomodensitométrie, angiographie rétinienne*
- Patient : *XXXXXX XXXXXXXX*
- Age : *45 ans*
- Radiologue : *Dr Jean MARTIN*
- Commentaire : *Occlusion veineuse rétinienne est déjà traitée en janvier 2011*



### Script MATLAB N°6.1.

```
Function [marque_binair,cle]=marque
marque1='Centre d'imagerie de Maghnia';
marque2='10/10/2012';
marque3='Tomodensitométrie,angiographie rétinienne';
marque4='XXXXXX XXXXXXXXX';
marque5='45 ans';
marque6='Dr Jean MARTIN';
marque7='occlusion veineuse rétinienne est déjà traitée en janvier 2011';
marque_binair1=uint8(double(marque1));
m1=size(marque_binair1,2);
for i=1:m1
    marque_binair11(i,:)=bitget(marque_binair1(1,i),8:-1:1); % insérer la valeur de BIT donnée
end
marque_binair2=uint8(double(marque2));
m2=size(marque_binair2,2);
for i=1:m2 marque_binair22(i,:)=bitget(marque_binair2(1,i),8:-1:1);
end
marque_binair3=uint8(double(marque3));
m3=size(marque_binair3,2);
for i=1:m3 marque_binair33(i,:)=bitget(marque_binair3(1,i),8:-1:1);
end
marque_binair4=uint8(double(marque4));
m4=size(marque_binair4,2);
for i=1:m4 marque_binair44(i,:)=bitget(marque_binair4(1,i),8:-1:1);
end
```

**2.Insertion** : pour insérer la marque dans l'image en utilisant l'algorithme choisit (Xie). Cette partie contient l'insertion de la marque, la décomposition et la reconstruction en ondelette et la partie tatouage et détatouage .(voire Script MATLAB Préliminaire N°6.2).

### Script MATLAB Préliminaire N°6.2.

```
% insérer la marque [marque_binair,cle3]=marque; [l,k]=size(marque_binair);
marque_binair_vec=reshape (marque_binair, 1,l*k);
% Décomposition par ondelette de Haar avec le 1er niveau sur l'image a marqué
[ca1,ch1,cv1,cd1] = dwt2 (image_a_marquer,'haar');
[mm,nn]=size(ca1);
%% initiation des Paramètres de l'approximation
s=20; cle2=ca1>s;
```



cle2=imfill (cle2,'holes'); % Remplit les trous dans l'image d'entrée. Un trou est un ensemble %de pixels d'arrière-plan qui ne peuvent être atteints en remplissant le fond à partir du bord de l'image.

cle2=bwareaopen(cle2,100);% Supprime d'une image binaire tous les composants connectés (objets) qui ont moins de 100 pixels, produisant une autre image binaire cle2

alpha=2.8;WW=14;a=1;ci=1;cj=1;

% Partie de tatouage d'après l'algorithme de Xie for i=1:3:mm

for j=1:3:nn if ci<=(l\*k)

if cle2(i,j)==1

if ch1(i,j)~=cv1(i,j) && ch1(i,j)~=cd1(i,j) && cv1(i,j)~=cd1(i,j)

vec=[ch1(i,j),cv1(i,j),cd1(i,j)];

vec\_crois=sort(vec);

if abs(vec\_crois(1)-vec\_crois(2))>5 && abs(vec\_crois(2)-vec\_crois(3))>5 clei(ci)=i;

clej(cj)=j; ci=ci+1; cj=cj+1;

end end end end

end end

for i=1:l\*k x=clei(i); y=clej(i);

vec=[ch1(x,y),cv1(x,y),cd1(x,y)]; vec\_crois=sort(vec);

s\_alpha=((vec\_crois(3)-vec\_crois(1))/2)\*alpha;

kk=floor((vec\_crois(2)-vec\_crois(1))/s\_alpha); kk=kk+1;

if mod(kk,2)==0

if marque\_binair\_vec(i)==1

if vec\_crois(2)==ch1(x,y) ch1(x,y)=vec\_crois(1)+(kk-1)\*s\_alpha; elseif

vec\_crois(2)==cv1(x,y) cv1(x,y)=vec\_crois(1)+(kk-1)\*s\_alpha; elseif

vec\_crois(2)==cd1(x,y) cd1(x,y)=vec\_crois(1)+(kk-1)\*s\_alpha; end

elseif marque\_binair\_vec(i)==0 if vec\_crois(2)==ch1(x,y)

ch1(x,y)=vec\_crois(1)+kk\*s\_alpha; elseif vec\_crois(2)==cv1(x,y)

cv1(x,y)=vec\_crois(1)+kk\*s\_alpha; elseif vec\_crois(2)==cd1(x,y)

cd1(x,y)=vec\_crois(1)+kk\*s\_alpha; end

end

elseif mod(kk,2)~=0

if marque\_binair\_vec(i)==1 if

vec\_crois(2)==ch1(x,y)

ch1(x,y)=vec\_crois(1)+kk\*s\_alpha; elseif vec\_crois(2)==cv1(x,y)

cv1(x,y)=vec\_crois(1)+kk\*s\_alpha; elseif vec\_crois(2)==cd1(x,y)

cd1(x,y)=vec\_crois(1)+kk\*s\_alpha; end

elseif marque\_binair\_vec(i)==0 if vec\_crois(2)==ch1(x,y)





```
ch1(x,y)=vec_crois(1)+(kk-1)*s_alpha; elseif vec_crois(2)==cv1(x,y)
cv1(x,y)=vec_crois(1)+(kk-1)*s_alpha; elseif vec_crois(2)==cd1(x,y)
cd1(x,y)=vec_crois(1)+(kk-1)*s_alpha; end
end
end
```

%% Reconstruction par Ondelette de Haar l'image marquée

```
image_marquee=idwt2(ca1,ch1,cv1,cd1,'haar',[m,n]);
image_marquee=uint8(image_marquee);
```

%% Partie détatouage de l'image marquée

```
[cat1,cht1,cvt1,cdt1] = dwt2(image_marquee,'haar');
```

```
for i=1:l*k
```

```
x=clei(i); y=clej(i); vec=[cht1(x,y),cvt1(x,y),cdt1(x,y)]; vec_crois=sort(vec);
```

```
s_alpha=((vec_crois(3)-vec_crois(1))/2)*alpha;
```

```
kk=round((vec_crois(2)-vec_crois(1))/s_alpha); if mod(kk,2)==0;
```

```
marque_binair_extraite(i)=0; elseif mod(kk,2)~=0 marque_binair_extraite(i)=1;
```

```
end
```

```
marque_binair_extraite=reshape(marque_binair_extraite,l,k);
```

### 3. Attaque : une simulation des attaques contre les images marquées

%% insérer les différents types d'attaques (rotation,bruit,median,zoom,egalisation,expansion)

```
[rotation,bruit,median,zoom,egalisation,expansion]=attaque (image_marquee);
```

```
[caa1,cha1,cva1,cda1] = dwt2(rotation,'haar');
```

### 4. Extraction : pour extraire la marque insérée dans l'image marquée ou marquée et attaquée.

%%Partie detatouage pour l'attaque de rotation for i=1:l\*k

```
x=clei(i); y=clej(i);
```

```
vec=[cha1(x,y),cva1(x,y),cda1(x,y)]; vec_crois=sort(vec);
```

```
s_alpha=((vec_crois(3)-vec_crois(1))/2)*alpha;
```

```
kk=round((vec_crois(2)-vec_crois(1))/s_alpha); if mod(kk,2)==0
```

```
marque_binair_extraite_rotation(i)=0;
```

```
elseif mod(kk,2)~=0 marque_binair_extraite_rotation(i)=1;
```

```
end
```

```
end
```



```
marque_binair_extraite_rotation=reshape (marque_binair_extraite_rotation,l,k);
```

### 5. Evaluation : deux programmes pour calculer le PSNR et le TAF .

```
%% Parametre d'évaluation
```

```
difference=image_marquee-image_a_marquer; psnr=psnr (image_a_marquer,  
image_marquee);
```

```
%% calcul la TAF de la marque binaire pour les différents types d'attaque
```

```
taf1=taf(marque_binair,uint8(marque_binair_extraite));
```

```
taf2=taf(marque_binair,uint8(marque_binair_extraite_rotation)); mat(1,WW)=taf1;
```

```
mat(2,WW)=taf2;
```

```
%% partie d'affichage et de visualisation
```

Pour évaluer l'efficacité de notre algorithme en termes de degré de dégradation de l'image tatouée et la sensibilité et l'aptitude de détecter toute transformation dans l'image. Nous allons séparer les tests en trois parties :

#### **III.2.1. Analyser la propriété d'imperceptibilité :**

1.1. Calculer le PSNR de l'image tatouée pour le Niveau 1 jusqu'à le niveau 4. est ce que l'image taouée est imperceptible pour les différentes niveau de décomposition ? Que constaté vous ?

N.B/ Pour calculer le PSNR pour le 2eme jusqu'à 4eme niveau de décomposition ;il faut écrire un autre Programme en MATLAB

1.2. l'insertion de la marque dans notre algorithme est dans l'approximation .Nous varions maintenant le paramètre alpha ( $\alpha$ ) de 0.2 jusqu'à 2.8 , calculer le PSNR pour différents niveau de décomposition (level=1,2,3,4).

1.3. Nous insérons maintenant la marque dans les détails, et nous varions le paramètre alpha ( $\alpha$ ) de 0.2 jusqu'à 2.8 , calculer le PSNR pour différents niveau de décomposition (level=1,2,3,4).

1.4. Est-ce que le niveau de décomposition dégrade l'image tatouée pour les deux cas d'insertion ?

1.5. Quel est le rôle de paramètre alpha ( $\alpha$ ) ?

**III.2.2. C h o i x de seuil de décision :** d'après les réponses sur les questions 4 et 5 quel est le seuil de décision ?

**III.2.3. Evaluation de la propriété de fragilité par rapport aux attaques :** Une propriété très importante que doit garantir un algorithme de tatouage est la robustesse contre les attaques. Les tests de robustesse ont été effectués sur cet algorithme avec différentes attaques: Rotation, Ajout de bruit gaussien, filtrage médian, zoom (agrandissement de la taille puis retour à la taille originale), égalisation d'histogramme et expansion de la dynamique.

3.1. . Ecrire d'autre programme MATLAB pour ces différents types d'attaques ?

3.2. Varier le paramètre alpha ( $\alpha$ ) de 1 jusqu'à 2.8 et calculer les valeurs de TAF pour les différentes attaques citées précédemment, pour les différentes niveaux de la décomposition par Ondelette (level=1,2,3,4).( Type d'insertion au niveau de l'approximation)



**Note :** Pour dire et décider si la marque détectée correspond effectivement à la marque insérée ,nous prend la mesure de TAF, telle que :

$$TAF(w, w') = \frac{1}{N_w} \sum_{i=1}^{N_w} w(i) \oplus w'(i) \quad (6.3)$$

Où  $w$  la marque insérée,  $w'$  la marque extraite,  $N_w$  la longueur de la marque et  $\oplus$  est l'opérateur ou exclusif XOR.

La présence d'une altération est déterminé si  $TAF(w, w') \geq T$ ,  $T$  étant le seuil de décision notons que  $0 \leq T \leq 1$ . Si  $TAF < T$ , alors les modifications de l'image sont considérées négligeable ou les deux marque coïncident parfaitement.

3.3. . Même question pour le deuxième cas d'insertion ( au niveau de détails)

3.4. . Que constatez-vous d'après ces résultats ?

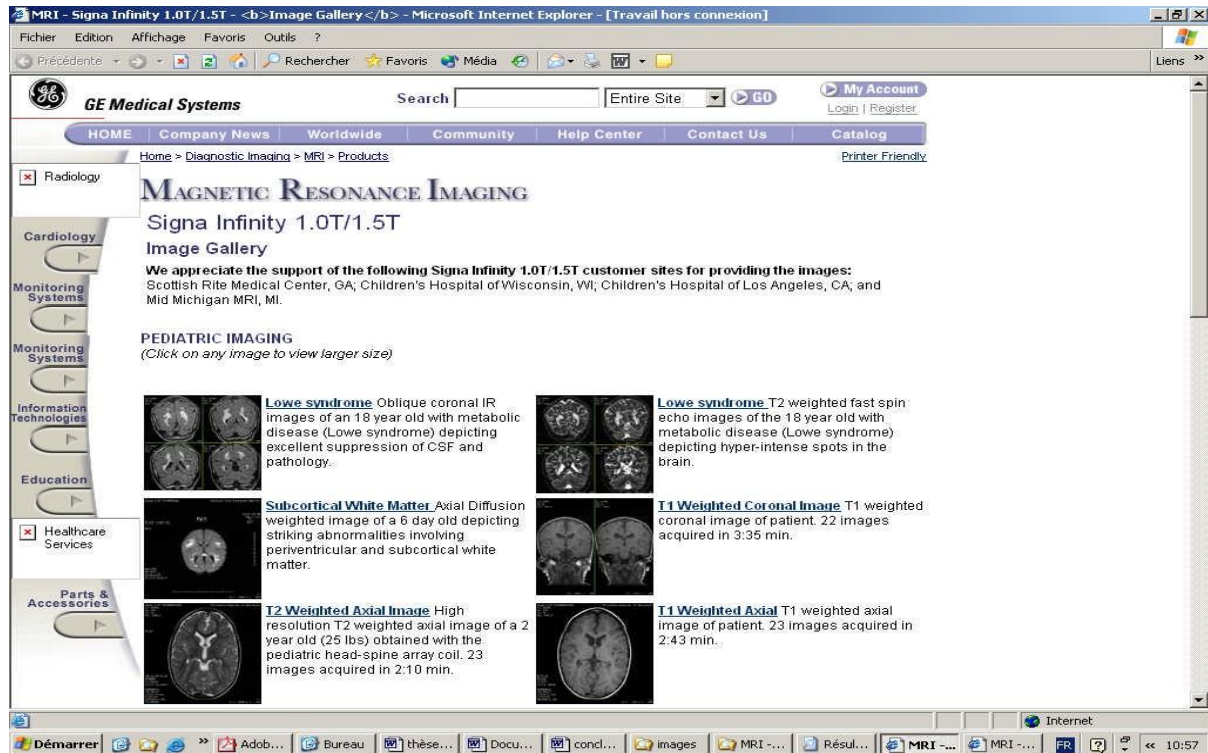
3.5. Donnez-moi la Marque extraite après une attaque (bruit gaussien) pour le 4<sup>eme</sup> niveau de la décomposition et  $\alpha = 2.2$ . Est-ce que notre algorithme est Robuste ?

3.6. Fixé le niveau idéal de la décomposition par ondelette (par exemple : level=4). Tester ce algorithme pour différents type d'ondelette (db5 ; symlet3 ;coif5 ;bior1.3 ;bio4.4 et bior6.8....etc) pour alpha ( $\alpha$ )=2.2 . Quel est la meilleure ondelette ?

3.7. Nous changeons maintenant dans notre Programme principale MATLAB l'instruction DWT et IDWT par PWT et IPWT dont notre but d'appliquer la transformée en paquet d'ondelette et voir l'impact de cette décomposition sur le tatouage des images médicale.

3.8. . Evaluer l'efficacité de ce nouvel algorithme ? (même questions des trois parties)

## ANNEXE A)- Base de Données des images médicales



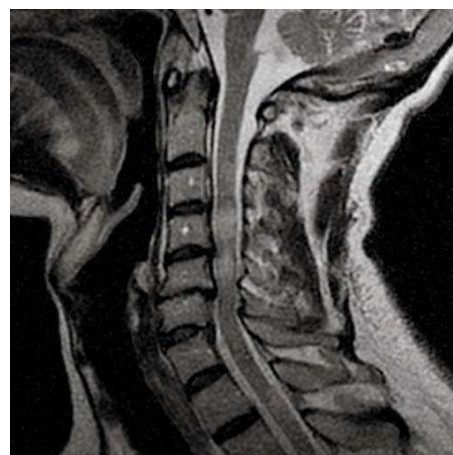
L'ensemble des images utilisées pour les tests au cours de ce TP sont des images Médicales de différentes modalités IRM, CT, MAMMOGRAPHIE, ECHOGRAPHIE, sont acquises de la base de données GE-SYSTEM présentées à l'adresse:

<http://www.gemedicalsystems.com>

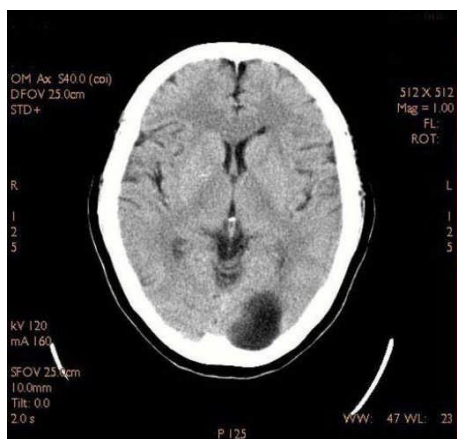
Taille de ces images 512× 512 (niveau de gris) codée sur 8bpp.



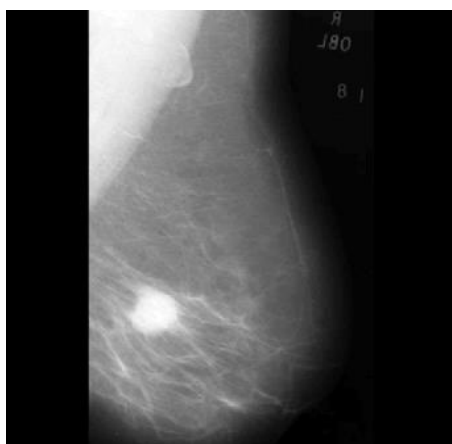
IRM1



IRM2



**CT1**



**Mammographie**



**ECHO**

Fig.A. Les images test

---

## Référence Bibliographie

---

- [1]. EL HADJ MIMOUNE Khadidja, MERABET Meriem ; « Etude de sécurité en base de données avec une application pour le contrôle d'accès ». Master en Informatique option système d'Information et de Connaissances (S.I.C) ; université de Tlemcen, 29 Septembre 2011.
- [2]. Laurent Audibert , « Bases de données de la modélisation au SQL, Conception des bases de données, modèle relationnel et algèbre relationnelle, langage SQL, programmation SQL » Cours et exercices, Edition 'Ellipses' ;254 pages, 25/09/2009, EAN: 9782729851200
- [3]. Simita Mishra and kailash chandra Mishra « Medical Informatics : An Exploration » ICFAI Books , University Press,2006,ISBN:81-314-0378-5
- [4]. Andreas MEIER : “Introduction pratique aux bases de données relationnelles”, Deuxième Edition, Collection IRIS, Springer-Verlag France 2006, ISBN 978-2-287-25205-1, DOI : 10.1007/2-287-31090-8.
- [5]. Ramez ELMASRI & Shamkant NAVATHE : “Conception et architecture des bases de données”,4<sup>ème</sup> Edition, Pearson Education, 21 Septembre 2004, ISBN : 978-2744070556.
- [6]. BOUKLI HACENE Ismail ; « Compression des images médicales par ondelettes de deuxième génération », Thèse de doctorat en électronique Biomédicale, Université de TLEMCEM ; 14 septembre 2014.
- [7]. J.SHAPIRO, "Embedded Image Coding using Zerotree of Wavelet Coefficients", IEEE trans. Signal processing. Vol. 41, pp. 3445-3463, December 1993.
- [8]. A. Said et W. Pearlman,"A new, fast, and efficient image codec based on set partitioning in hierarchical trees", IEEE Transactions Circuits Syst. Video Technol., vol. 6, pp. 243-250, juin 96.
- [9]. Chabriaïs J , Gibaud B . « DICOM , le standard pour l'imagerie médicale ». EMC ( Elsevier Masson SAS , Paris ) ,Radiologie et imagerie médicale ; principes et technique-radioprotection, 35-120-A-10,2010.
- [10]. Oleg S. Pinykh “ Digital Imaging and Communications in Medicine (DICOM); A Practical Introduction and Survival Guide”, Department of Radiology, BIDM CHarvard Medical School. USA. ISBN 978-3-540-74570-9. DOI 10.1007/978-3-540-74571-6. Springer 2008.
- [11]. BACHAR Mohamed el Amine « Gestion des images médicales dans un centre hospitalier universitaire. Application au cryptage et tatouage des images médicales » ; Master en génie biomédical ; spécialité signaux et images en médecine, Université de Tlemcen. 24.06.2013
- [12]. J. M. M. RODRIGUES, "Transfert sécurisé d'images par Combinaison de techniques de Compression, cryptage et marquage ", Thèse de doctorat, Université Montpellier II, Octobre 2006.