REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique

École Nationale Polytechnique

Département d'Électronique

Projet de fin d'études

en vue de l'obtention du diplôme
d'Ingénieur en Électronique

Thème

Watermarking Audio

Encadré par :
M. L. ABDELOUEL

Étudié par :
Melle. Mounira BENZITOUNI
M. Liès M. CHOUITER

Promotion : juin 2002

Remerciements

Nous tenons à adresser nos plus vifs remerciements à notre promoteur M. ABDELOUEL pour son suivi et la qualité des conseils qu'il nous a prodigués tout au long de l'année.

Nos remerciements vont également à tous les enseignants de l'École Nationale Polytechnique qui ont contribué à notre formation.

Nous remercions enfin tous ceux, qui de prés ou de loin, nous ont soutenus et aidés dans la réalisation de ce travail.

Watermarking Audio

Résumé

Avec l'apparition de codeurs permettant de compresser beaucoup plus efficacement les fichiers numériques et l'accessibilité aux systèmes de reproduction, les copies illégales d'œuvres audio ont pris une ampleur préoccupante. Pour lutter contre ce phénomène qui cause un préjudice commercial substantiel aux auteurs et aux ayants droit, des techniques sont mises au point pour protéger la propriété intellectuelle et les droits d'auteurs. Le Watermarking Audio est une de ces techniques.

Après la présentation de quelques-uns des algorithmes de watermarking existants et des attaques qui pourraient éventuellement les cibler, ce travail expose une méthode de tatouage basée sur l'étalement du spectre et les propriétés perceptuelles de l'audition humaine, pour prendre en charge les exigences de robustesse et d'imperceptibilité des watermarks insérés.

Abstract

With the appearance of effective coders making it possible much more to compress effectively the numerical files and accessibility with the systems of reproduction, the illegal copies of audio œuvres are taking an alarming dimension . To fight against this phenomenon which causes a substantial commercial harm to the authors and the entitled beneficiaries, techniques are developed to protect the intellectual property and the copyrights. Watermarking Audio is one of these techniques.

After the presentation of some existing algorithms of watermarking and of the attacks which could possibly target them, this work presents a method of tattooing based on the spread spectrum and on the perceptual properties of human hearing, to take charge of the requirements of robustness and imperceptibility of the inserted watermarks.

ملخص

مع ظهور مرمزات فعالة تسمح بتقليص حجم الملفات الرقمية و تسهيل عمليات إستنساخ المؤلفات السمعية ، أصبحت النسخ الغير شرعية منتشرة بصورة تثير القلق. لمحاربة هذه الآفة التي تسبب أضرارا تجارية معتبرة للمؤلفين و ذوي الحقوق ، أحدثت تقنيات لحماية الملكية الفكرية و حقوق المؤلفين ، منها تقنية تعرف بـ : **Watermarking Audio**

بعد تقديم بعض مناهج هذه التقنية و الهجومات التي يمكن أن تستهدفها ، هذا البحث يعرض طريقة وشم ترتكز على نشر الطيف و خصائص السمع البشري ، تضمن متانة الوشم و تحافظ على النوعية الصوتية للمؤلفات

SOMMAIRE

INTRODUCTION

Avec le développement des technologies le l'information et de la communication, et plus particulièrement d'Internet, la diffusion des données numériques, entre autres les médias audio, a pris une ampleur sans précédent. Des œuvres audiovisuelles circulent en abondance sur la toile. Cependant, faute de moyens de contrôle et de protection efficaces, elles font l'objet d'un piratage constant. Aussi, tant les éditeurs traditionnels que les producteurs de musique se montrent réticents à proposer leurs oeuvres sur le réseau. Le rétablissement de la confiance indispensable au développement de la diffusion des œuvres artistiques sur le net passe par la mise au point de techniques efficaces permettant la protection des droits d'auteur et de la propriété intellectuelle. Le Watermarking Audio est une des techniques proposées à cette fin. Il consiste à tatouer les œuvres à protéger en y insérant un filigrane numérique. Les filigranes insérés, ou watermarks, sont conçus pour contenir des informations, soit sur les propriétaires des œuvres (copyright), soit sur les ayants droit (fingerprint).

Le présent travail est consacré à l'étude du watermarking audio et à la présentation de quelques-unes des techniques qu'il utilise, avec l'objectif d'implémenter une méthode permettant l'insertion d'informations sur le copyright.

Dans la mesure où ce domaine de recherche est un domaine relativement récent, pour lequel les publications sont beaucoup moins nombreuses que pour le watermarking vidéo, d'importants développements sont consacrés aux aspects théoriques. Ils sont indispensables pour une correcte compréhension des algorithmes présentés et de la méthode développée.

Après un bref historique et quelques généralités sur le watermarking et les caractéristiques d'un bon tatouage - principalement la robustesse et l'imperceptibilité - (chapitre 1), des développements sont consacrés à l'étude du système d'audition humain et à la psychoacoustique, dont la connaissance est nécessaire pour la mise au point de techniques d'insertion de watermarks inaudibles (chapitre 2). Dans le même ordre d'idées, le chapitre 3 est consacré à la présentation et à l'étude du format de compression le plus utilisé sur Internet, MPEG audio, et des modèles psychoacoustiques qu'il utilise. Sachant que la compression avec pertes constitue en soi une attaque, l'étude du MPEG audio aidera à la mise au point d'une technique d'insertion de watermarks robustes. D'autres techniques d'attaque susceptibles d'altérer les filigranes sont présentées dans le chapitre 4. Dans le chapitre 5, consacré à l'état de l'art dans le domaine du watermarking audio, sont présentées quelques-unes des méthodes de tatouage récemment mises au point. Enfin, le chapitre 6 sera consacré, d'une part, à l'exposé de la méthode de tatouage étudiée et de l'application implémentée qui lui correspond et, d'autre part, aux résultats des différentes expérimentations effectuées pour en tester les performances et la robustesse.

Chapitre 1

GÉNÉRALITÉS

- **Historique**
- **Qu'est ce que le Watermarking ?**
- **Modélisation de l'insertion d'un Watermark**
- **Classification des systèmes de Watermarking**
- **Applications**

GÉNÉRALITÉS

Ce chapitre sera consacré à un bref historique sur la stéganographie et sur le « data hiding » en tant qu'origine du watermarking, à la définition de ce dernier, à la classification des différentes approches, avant de conclure par un bref aperçu sur ses différentes applications.

1.1. HISTORIQUE [1]

A partir du 16^{e} siècle, une littérature de plus en plus abondante traitait du « **data hiding** » et de la **"stéganographie"**. Différentes méthodes ont ainsi été proposées pour la dissimulation de l'information. Dans son ouvrage intitulé **Schola Steganographica**, **Schott (1608-1666)** expliquait comment dissimuler des messages dans des tablatures musicales où chaque note correspondait à une lettre. Une autre méthode basée sur le nombre d'occurrences des notes a été utilisée par **Bach**. Schott a également amélioré le code de **"Ave Maria"** proposé par **Trithemius (1462-1516)** dans **Steganographiæ**, un des premiers ouvrages consacré au sujet. Ce code comprenait 40 tables, chacune contenait 24 entrées (une pour chaque lettre de l'alphabet de l'époque) dans quatre langues: Latin, Allemand, Italien, et Français. Chaque lettre du texte en entrée était remplacée par le mot ou l'expression qui apparaissait dans la table correspondante.

Wilkins (1614-1672), *Master of Trinity College* (Cambridge), montra comment deux musiciens sont parvenus à converser en jouant sur leurs instruments de musique. Il expliqua également comment on pouvait dissimuler un message au moyen de figures géométriques en utilisant des points, des lignes ou des triangles.

Une autre méthode qui fut très largement répandue est l'**acrostic**. Dans son ouvrage **The Codebreakers**, **Kahn** montra comment un moine a pu dissimuler un message dans les premières lettres des chapitres successifs d'un livre qu'il avait écrit. Ce procédé a été également utilisé par des prisonniers de guerre qui dissimulaient des messages dans les lettres qu'ils envoyaient à leur famille. De nombreux messages ont cependant été interceptés du fait de la difficulté de les concevoir tout en gardant une cohérence qui n'éveillerait pas de soupçons.

Différentes techniques ont été proposées pour la dissimulation de l'information. Les plus anciennes d'entre elles remontent à l'Antiquité :

- **La sécurité par obscurité (cryptographie)**

En 1883, **Kerckhoffs** a énoncé les premiers principes de l'ingénierie cryptographique en considérant que la méthode employée pour le chiffrement des données est connue par l'ennemi. La sécurité dépendra ainsi uniquement du bon choix de la clé de chiffrement. Dès

lors, la sécurité par obscurité qui part du principe que la méthode de cryptographie mise en œuvre est « obscure » à l'ennemi est devenue obsolète. Cependant, de nos jours encore, des systèmes stéganographiques se limitent à l'insertion de données dans le bit le moins significatif d'un média (audio ou vidéo), ce qui rend aisé pour un éventuel ennemi leur détection et leur suppression.

- **Le camouflage**

Depuis les temps anciens, les artistes se sont aperçus que les œuvres de sculpture ou de peinture apparaissent différentes selon l'angle de vision. C'est ce qui les a amenés à établir les règles de la perspective. Pendant les XVI^{eme} et XVII^{eme} siècles, les images amorphes fournissaient un moyen idéal pour camoufler des idées politiques dangereuses ou hérétiques. Un chef d'œuvre de dissimulation d'image « **The Vexierbild** » a été créé en 1530 par un graveur de Nürnberg, **Shö**. De face, cette œuvre n'apparaît que comme un paysage étrange, mais de côté, elle laisse apparaître les portraits des rois célèbres.

Dans ses mémoires, **Herodotus (486-425 A.J.C)** expliquait comment, aux environs de 440 A.J.C., **Histiæus** a rasé la tête à nombreux de ses esclaves de confiance pour y tatouer un message caché par la repousse des cheveux. Son but était d'inciter à une révolte contre les Persans. Curieusement, cette même méthode a été également employée par des espions allemands au début du vingtième siècle.

Vers 1860, les problèmes relatifs à la réduction des images ont été résolus. En 1857, **Brewster** avait déjà suggéré la dissimulation de messages dans des espaces *« pas plus grands que des points »*. Pendant la guerre qui opposa la France et la Prusse (1870-1871) et alors que Paris était assiégée, des messages sur microfilm ont été envoyés grâce à des pigeons voyageur. De même, durant la guerre entre la Russie et le Japon (1905), des images microscopiques ont été cachées dans des oreilles, narines, et sous les ongles des doigts.

L'équivalent numérique de ces techniques de camouflage consiste en l'utilisation d'algorithmes de masquage. Comme la plupart des techniques de codage-source, ces techniques sont généralement basées sur les propriétés des systèmes de perception humains. Le masquage audio, par exemple, est un phénomène dans lequel un bruit interfère sur la perception d'un autre bruit. Comme ces effets sont pris en considération dans des standards de compression tels que MPEG, les systèmes de dissimulation insèrent les données dans les composantes les plus significatives en terme de perception afin qu'elles survivent à la compression.

- **Dissimulation de l'endroit où l'information est insérée**

Dans un protocole de sécurité développé en Chine ancienne, puis repris par un mathématicien italien **Jerome Crano (1501-1576)**, l'expéditeur appliquait une grille spécialement trouée sur une feuille, puis écrivait les lettres ou les mots de son message dans

les trous. En retirant la grille, il complétait alors la feuille par des lettres prises au hasard pour rendre ce message incompréhensible, ou par des mots de manière à former une lettre anodine. Le destinataire n'avait plus qu'à appliquer le même grille pour faire apparaître le message dissimulé.

L'équivalent numérique de cette technique consiste à introduire des distorsions en des endroits choisis par des générateurs pseudo-aléatroires. Caméléon, une de ces techniques, à été utilisée notamment dans la diffusion de CD et dans la télévision payante. Elle consiste à diffuser un contenu ayant le même message chiffré tout en donnant à chaque utilisateur une clé de déchiffrement légèrement différente pour que le message déchiffré soit légèrement modifié.

- **Etalement de l'information dissimulée**

Utilisé dans les télécommunications militaires depuis 1940, l'étalement de spectre est une technique qui consiste à envoyer un message sur un grand spectre de fréquences de telle sorte qu'à toutes les fréquences, la puissance du signal reste inférieure à celle du bruit. Localement, l'émission est ainsi toujours imperceptible; ce n'est qu'en écoutant sur l'ensemble du spectre d'émission et en connaissant le procédé utilisé que l'on pourra retrouver le message émis.

Tirkel et al. étaient les premiers à noter que les techniques d'étalement de spectre pouvaient être appliquées au watermarking. Plus tard, un certain nombre de chercheurs ont développé des techniques stéganographiques basées sur l'étalement de spectre. Ces techniques tirent profit de la largeur de la bande passante du canal par rapport à la bande relativement étroite utilisée pour la dissimulation des données.

Certaines de ces techniques opèrent directement sur des données compressées. MP3Stego, par exemple, dissimule l'information dans des bits de la couche III du MPEG pendant le processus de compression.

Une nouvelle technique de codage par transformation est la dissimulation de l'écho : « **echo hiding** ». Elle est basée sur le fait que l'oreille humaine ne peut pas percevoir les échos courts (de l'ordre de la milliseconde). Ainsi, la dissimulation des données se fait par l'insertion de deux types d'échos courts avec différents retards pour coder les "1" et les "0". Ces échos sont insérés à des endroits séparés par des espaces pseudo-alétoires.

- **Techniques spécifiques à l'environnement**

La dissimulation par écho compte parmi les techniques de dissimulation de l'information exploitant les particularités des phénomènes naturels. La « communication utilisant le passage des météorites », technologie issue du domaine militaire, utilise le canal radio transitoire engendré par l'ionisation des traînées des météorites pénétrant dans l'atmosphère, pour envoyer des paquets de données entre une station mobile et une base.

La non-permanence de ces canaux rend la localisation des stations mobiles très difficile.

Les techniques de dissimulation de l'information et de marquage sont également utilisées pour protéger les documents écrits. Le watermark sur papier est une technique très ancienne utilisée pour faire face aux contrefacteurs. Des innovations plus récentes utilisent une encre fluorescente sensible aux UV pour les chèques de voyage.

Actuellement, la tendance est de plus en plus à l'incorporation des informations à dissimuler dans des bandes magnétiques, comme pour les cartes bancaires où chaque carte possède son propre numéro de série difficile à reproduire. Cette technique est aussi utilisée pour les cartes téléphoniques. Des fibres magnétiques sont insérées dans du papier ou du carton, donnant ainsi à chaque copie du document une empreinte digitale unique.

1.2. QU'EST CE QUE LE WATERMARKING ? [1], [2]

Depuis quelques années, l'information a changé de forme de représentation. Les supports analogiques sont progressivement remplacés par des supports numériques. Les réseaux d'échange d'information véhiculent désormais des signaux numériques, et l'utilisation d'images, de sons ou de séquences vidéo fait généralement intervenir leur représentation digitale. C'est notamment le cas sur l'Internet.

Le numérique a des avantages considérables mais n'est pas sans dérive : en rendant la copie à l'identique très aisée, le numérique a participé au boom récent de la piraterie sur la propriété intellectuelle. Les autres causes de ce phénomène sont l'explosion du réseau d'échange de données qu'est l'Internet et l'accession du grand public à la fabrication de supports de stockage numérique (graveurs de CD...).

C'est pour cette raison qu'aujourd'hui, certains auteurs et distributeurs de documents digitaux hésitent à autoriser l'accès à leur propriété intellectuelle. Ils sont en permanence à la recherche de solutions leur permettant d'assurer la protection de leurs droits d'auteurs.

De nombreuses techniques du data hiding ont été proposées comme le montre la figure 1.1 ci après.

Le ***tatouage numérique*** a été proposé comme un moyen permettant d'identifier le propriétaire et le distributeur d'une information digitale. Ce procédé consiste en l'insertion d'un copyright dans un média digital (image, son ou vidéo), en apportant de petites modifications au niveau des échantillons du signal.

A la différence du cryptage, le tatouage ne limite pas l'accès à l'information tatouée. Aussi, Une fois qu'un document crypté est déchiffré, le média n'est plus protégé, alors que dans le tatouage, le watermark fait partie intégrante du signal hôte.

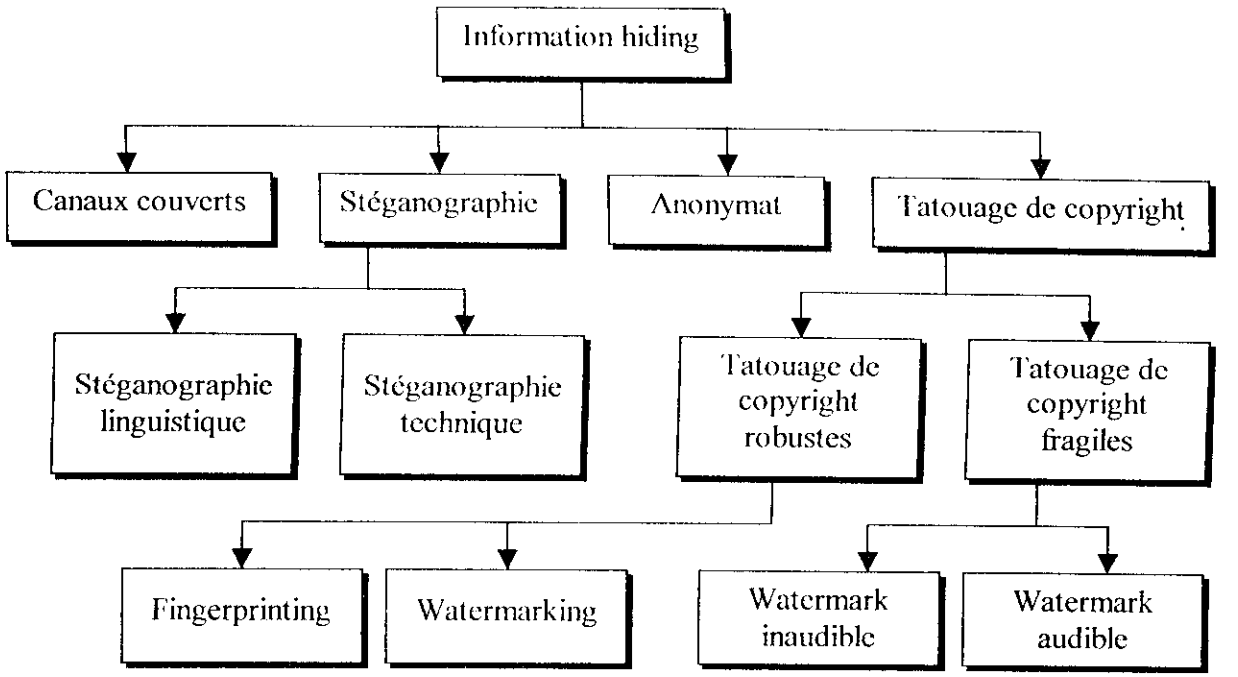


Figure 1.1 : *Classification des techniques de data hiding*

D'une manière générale, l'information à cacher est habituellement insérée dans une autre servant de couverture. L'objet ainsi marqué (image, vidéo ou audio) est appelé **stego-objet.**

La **stego-clé** ***(stego-key)*** est, quant à elle, utilisée pour le contrôle du procédé de tatouage, afin d'éviter la détection et la localisation de l'information dissimulée.

Les critères d'appréciation[1] d'un watermark sont les suivants :

- **Imperceptibilité :** Le filigrane numérique inséré ne doit pas être perceptible à l'oreille humaine pour conserver une bonne qualité du signal audio.

- **Sécurité :** La marque doit être statistiquement indétectable pour qu'il soit impossible de la localiser et/ou la retirer, et ce quand bien même le schéma de dissimulation serait connu. Si le filigrane peut être localisé, toute tentative de l'extraire ou l'altérer doit rendre le média audio complètement inutilisable.

- **Robustesse :** Le watermark ne doit pas être altéré par les distorsions que pourrait subir le signal hôte. Il doit être résistant à tous types de manipulations (codage avec pertes, ajout d'un bruit, filtrage, conversions A/D-D/A...etc.).

- **Spécificité :** La marque doit être bien spécifique pour être parfaitement identifiée lors de son extraction.

[1] Les critères d'appréciation d'un watermark dépendent des domaines d'application.

- **Traitement au niveau du domaine compressé :** Il est évident que pour des raisons pratiques, les distributeurs stockent leurs médias dans le domaine compressé. Aussi, doit-il être possible de dissimuler le watermark dans le domaine compressé.

- **Interopérabilité** : Même si le tatouage numérique se fait de plus en plus sur les médias compressés, il serait souhaitable de pouvoir tatouer les médias non-compressés sans devoir au préalable procéder à leur compression.

- **Bit rate constant :** Le tatouage d'un média ne doit pas augmenter le bit rate, particulièrement dans les applications où la bande passante doit être respectée.

- **Coût** : La mise en place d'un système de marquage doit être d'un coût raisonnable.

1.3. MODÉLISATION DE L'INSERTION D'UN WATERMARK [3]

De manière générale, le processus de tatouage (figure 1.2) se déroule de la manière suivante :

Soit un signal audio A, dans lequel on veut insérer une marque M. L'opération est paramétrée par une clé K (une suite de nombres générée de préférence de manière aléatoire).

Le processus d'insertion peut être exprimé par la forme : $(A \times K \times M \rightarrow A')$.

Quant au processus de détection de la marque, il peut être représenté par la formule suivante : $(A' \times K (\times A \text{ "optionnel"}) \rightarrow M)$.

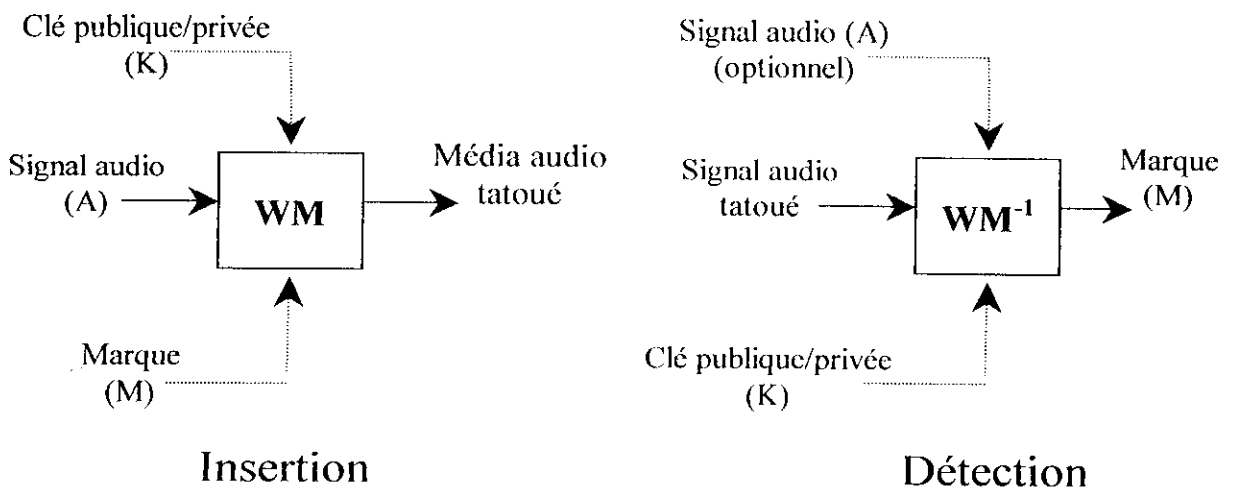


Figure 1.2 : *Schéma général d'insertion et de détection d'un watermark* [3].

1.4. CLASSIFICATION DES SYSTÈMES DE WATERMARKING [1]

Il existe différents systèmes de marquage dits robustes, et qui sont :

- **Systèmes privés de marquage :** Cette classe se subdivise en deux types, qui nécessitent tout deux le signal audio original pour l'extraction de la marque.

  Type 1 : Ces systèmes utilisent le signal original (A) pour localiser et extraire la marque (M) du signal audio potentiellement distordu (A').

  Type 2 : Ces systèmes nécessitent, en plus du signal original (A), une copie de la marque insérée (M), et répondent par « oui » ou par « non » à la question : « est-ce que (A') contient la marque (M) ? ». On peut représenter de tels systèmes par la formule :

  $(A \times A' \times K \times M \rightarrow \{0,1\})$. Ces systèmes sont plus robustes que les autres puisqu'ils fournissent peu d'informations sur le watermark.

- **Systèmes semi-privés de marquage :** Ces systèmes ont les mêmes caractéristiques que les systèmes privés de type 2, sauf qu'ils ne nécessitent pas le signal audio original $(A' \times K \times M \rightarrow \{0,1\})$.

- **Systèmes publiques de marquage :** Ces systèmes n'ont besoin ni de la marque (M), ni du média original (A). En effet, ces systèmes extraient la marque à partir du média marqué et de la clé seulement $(A' \times K \rightarrow M)$.

- **Systèmes asymétriques de marquage (systèmes à clé publique) :** Ces systèmes ont la propriété essentielle d'avoir un watermark qui peut être « lu » sans possibilité d'être extrait.

1.5. APPLICATIONS [4]

Les informations que le data hiding cherche à dissimuler peuvent être de différentes natures, ce qui offre un domaine d'application assez vaste.

Les militaires et les services de renseignements ont besoin de communications discrètes. Même si le contenu de ces communications est chiffré, leur simple détection peut rapidement conduire à une attaque. Pour cette raison, les militaires utilisent des techniques permettant de rendre la détection de leurs signaux impossible, à l'exemple de la modulation par étalement de spectre ou l'exploitation du passage de météorites.

Les organisations criminelles, également, recourent aux technologies du data hiding, pour pouvoir communiquer sans être repérées.

De plus, le fait que les gouvernements réglementent assez sévèrement l'utilisation de la cryptographie à des fins personnelles a incité au développement de nouvelles techniques permettant d'assurer la confidentialité des communications.

Dans le domaine commercial, le watermarking connaît plusieurs applications, la plupart destinées à la protection des droits d'auteurs.

1.5.1. La vérification du copyright

A fin Mars 2002, le terme « MP3 » est le huitième mot le plus recherché sur Internet, et était même en première position au début de 1999[2].

Du fait de la facilité de lecture et d'écriture en MP3, les copies frauduleuses mais de qualité néanmoins exceptionnelle sont en abondance sur Internet. Le développement du watermarking audio a été ainsi motivé par le souci des artistes et de leurs éditeurs de protéger leur droit d'auteur en insérant des tatouages numériques permettant d'identifier le propriétaire du document. Toute personne qui s'en réclamera propriétaire illégalement pourra être condamnée, la marque faisant office de preuve devant les tribunaux.

1.5.2. L'empreinte digitale (Fingerprinting)

Une autre application, dite « **Fingerprinting** », consiste à insérer dans le watermark **l'ayant droit au document**. Chaque copie du document contient une marque différenciée (l'empreinte) permettant d'identifier l'acquéreur autorisé. En cas de découverte de copies illégales, leur origine pourra être ainsi retracée.

1.5.3. Le contrôle de copies

Les watermarks peuvent aussi contenir les permissions attachées au document. Ils peuvent indiquer si le document en question est marqué en copie illimitée, copie interdite ou une seule copie autorisée. Les système de copie sont, de plus en plus, appelés à tester les documents avant de procéder à leur copie.

1.5.4. La surveillance de la diffusion

En plus de la protection du copyright, les watermarks peuvent être utilisés pour la surveillance de la diffusion par l'insertion d'un filigrane secret lors de la production ou de la diffusion d'un média audio. Des réseaux de stations de surveillance guettent les signaux diffusés, à la recherche de morceaux protégés et des copies ou diffusions illégales. La surveillance requiert un label unique contenu dans chaque seconde d'un signal audio.

[2] Selon " http://www.searchterms.com".

Chapitre 2

AUDITION HUMAINE ET PSYCHOACOUSTIQUE

- **Définitions**
- **Propriétés du système auditif**
- **Filtre auditif et bandes critiques**
- **Le modèle psychoacoustique**
- **Principes généraux du codage psychoacoustique**

AUDITION HUMAINE ET PSYCHOACOUSTIQUE

Les algorithmes de marquage s'appuient en majorité sur les propriétés perceptuelles du système auditif humain pour insérer des filigranes inaudibles.

La connaissance de ces propriétés (seuil absolu d'audition, masquages temporel et fréquentiel, filtre auditif...) est indispensable à la modélisation de l'oreille humaine (modèle psychoacoustique).

2.1. DÉFINITIONS [5]

Quelques définitions de base sont nécessaires à la compréhension des développements qui suivent :

- **Son pur (ou son tonal) :** son modélisable par une sinusoïde de période donnée. Son spectre est discontinu.
- **Bruit (ou son non tonal) :** son dont le spectre est continu.
- **Son complexe :** tout son est constitué d'une superposition de sons purs et de bruits d'amplitudes distinctes. Par conséquent, tout son peut être décomposé en ses composantes tonales (sons purs) et non tonales (bruits). Cette décomposition sur l'ensemble des fréquences forme le spectre du son.
- **Puissance acoustique :** puissance vibratoire transmise par une source sonore (émetteur) à l'air. Elle est exprimée en watt.
- **Intensité acoustique :** puissance acoustique reçue par unité de surface du récepteur. Elle est liée à l'amplitude des ondes sonores et est exprimée en Wm^{-2}.
- **Niveau d'intensité acoustique :** directement lié à la sensation auditive de l'intensité d'un son, ce niveau, noté N, donne l'intensité acoustique par rapport à une intensité de référence. Il est donné par :

$$N = 10.\mathrm{Log}\left(\frac{I}{Io}\right) \qquad (2.1)$$

avec I : intensité acoustique du son considéré (Wm^{-2}),

Io : intensité acoustique de référence (minimum du seuil d'audibilité de l'oreille pour un son tonal de fréquence 1 kHz). Ce minimum est égal à 10^{-12} Wm^{-2}.

2.2. PROPRIÉTÉS PERCEPTUELLES DE L'AUDITION HUMAINE

Les propriétés principales qui caractérisent le système auditif humain sont le *seuil absolu d'audition* et le *phénomène de masquage*.

2.2.1. Le seuil absolu d'audition [6]

Le seuil absolu d'audition est le niveau d'intensité acoustique à partir duquel un son, à une fréquence donnée, est entendu dans des conditions de silence total. Ce seuil n'est pas constant mais varie en fonction de la fréquence du son (entre 20 Hz et 20.000 Hz) et des individus. Ainsi, comme le montre la courbe de Fletcher-Munsen (figure 2.1.) ci-dessous, un son de fréquence 2 kHz et de puissance acoustique 20 dB sera audible alors qu'un son de même puissance (20 dB) et de fréquence 13 kHz ne le sera pas.

Le seuil absolu d'audition (sa) est calculé par la formule suivante :

$$sa = 3{,}64.f^{-0.8} - 6.5.\exp(-0{,}6.(f-3{,}3)^2) + 0{,}001.f^4 \qquad (2.2)$$

où f est en kHz.

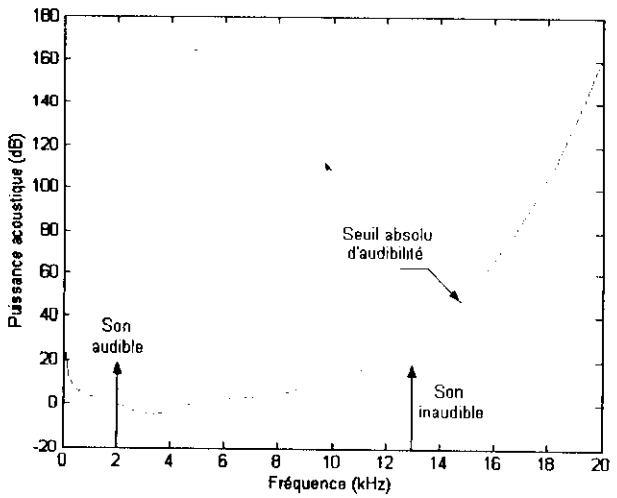


Figure 2.1 : *Courbe de Fletcher-Munsen (seuil absolu d'audition).*

2.2.2. L'effet de masque

Il y a masquage lorsqu'un son habituellement audible est rendu inaudible par un autre son d'amplitude plus importante. On distingue deux types de masquages : le *masquage temporel* (non-simultané) et le *masquage fréquentiel* (simultané).

- **Le masquage temporel [6]**

Ce type de masquage intervient lorsque avant ou après un son d'une puissance acoustique élevée, l'oreille perd pendant un court délai sa sensibilité normale. Ainsi, après le passage d'un

avion à basse altitude, l'oreille humaine ne retrouvera sa sensibilité normale lui permettant à nouveau d'entendre une conversation, qu'après un certain délai.

On distingue entre :

- le masquage appelé « *effet de précédence* » (ou *effet de Haas*) qui apparaît lorsque le signal masqué est émis *juste avant* le signal masquant,
- et le « *masquage de postériorité* » lorsque le son masqué est émis après le signal masquant.

La figure 2.2 ci-dessous donne une représentation de ce phénomène, pour des sons brefs et de type impulsionnel.

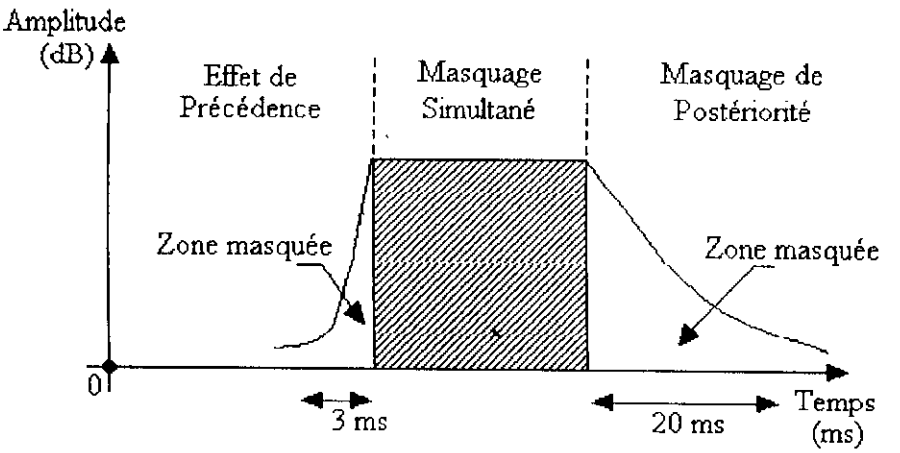


Figure 2.2 : Les *masquages temporels*.

Comme le montre la figure 2.2, l'effet de précédence est beaucoup plus « tolérant » que le masquage de postériorité. Un son théoriquement masqué, émis plus de 2 ou 3 ms avant le son masquant, devient un **pré-écho** audible. Au contraire, dans le cas du masquage de postériorité, un son restera masqué sur les 20 ms (environ) qui suivent l'émission du son masquant.

➢ **Le masquage fréquentiel [7], [8]**

Le masquage fréquentiel intervient lorsqu'un fort pic d'énergie audio, à une fréquence donnée, rend inaudible les sons aux fréquences voisines de ce pic et d'amplitude inférieure (cf. figure 2.3).

Pour reprendre l'exemple de l'avion, le masquage fréquentiel intervient, non pas après, mais pendant le passage de l'avion : une conversation sera rendue inaudible par le bruit provoqué par ce dernier.

Autre exemple : Une faible sinusoïde à 4000 Hz, audible dans des conditions normales, sera masquée par une autre sinusoïde à 1000 Hz d'amplitude nettement supérieure.

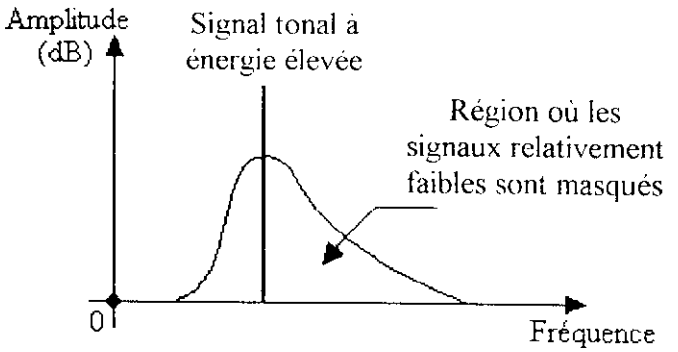


Figure 2.3 : *Le masquage fréquentiel.*

Le masquage se répand autour de la fréquence masquante de manière asymétrique. Il s'étend plus vers les fréquences supérieures à la fréquence masquante que vers les fréquences inférieures. Les bruits intenses à basse fréquence masquent beaucoup plus les sons à haute fréquence que l'inverse [6].

Le masquage fréquentiel, qui dépend de plusieurs facteurs (fréquence, puissance et type du son masquant), entraîne la modification du seuil d'audition, comme le montre la figure 2.4. ci-après (pour un son tonal de fréquence 1 kHz et d'amplitude 60 dB).

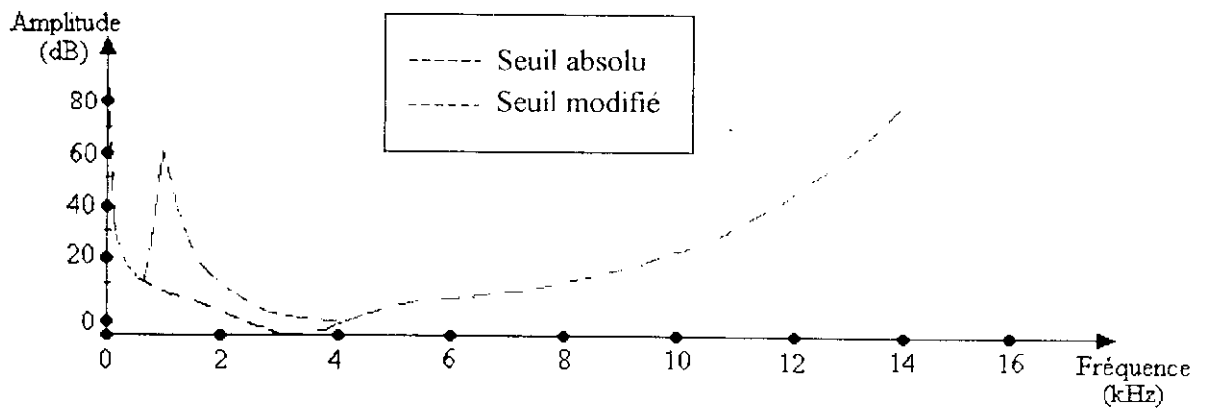


Figure 2.4 : *Changement du seuil d'audition en présence d'un son tonal de fréquence 1kHz et d'amplitude 60 dB.*

2.3. FILTRE AUDITIF ET BANDES CRITIQUES [9]

Le concept de « *filtre auditif* », développé par Fletcher en 1940, est né de la découverte du phénomène de masquage. Selon Fletcher, le système auditif se comporte comme un banc de filtres qui se chevauchent continûment. Il a modélisé cet ensemble par une série de filtres rectangulaires dont la largeur a été appelée ***bande critique*** .

Il a été établi expérimentalement qu'il existait 25 bandes critiques couvrant le spectre des fréquences audibles pour l'homme (20Hz à 20kHz). Exprimée en hertz, la largeur de ces bandes critiques n'est pas linéaire et varie selon la fréquence considérée (tableau 2.1.).

Bande critique	Fréquence (Hz)		
	Basse	*Haute*	*Largeur*
1	0	100	100
2	100	200	100
3	200	300	100
4	300	400	100
5	400	510	110
6	510	630	120
7	630	770	140
8	770	920	150
9	920	1080	160
10	1080	1270	190
11	1270	1480	210
12	1480	1720	240
13	1720	2000	280

Bande critique	Fréquence (Hz)		
	Basse	*Haute*	*Largeur*
14	2000	2320	320
15	2320	2700	380
16	2700	3150	450
17	3150	3700	550
18	3700	4400	700
19	4400	5300	900
20	5300	6400	1100
21	6400	7700	1300
22	7700	9500	1800
23	9500	12000	2500
24	12000	15500	3500
25	15500	22050	6550

Tableau 2.1 : *Représentation des 25 bandes critiques les plus significatives* [10].

En raison de cette non linéarité, l'échelle choisie pour la représentation des bandes critiques est le ***bark***. Cette échelle linéaire est plus adéquate, la largeur de chaque bande critique étant égale à 1 bark.

Le hertz et le bark sont liés par la relation suivante (figure 2.5.) :

$$Barks = 13.Arctg(0{,}76.f) + 3{,}5.Arctg\left(\frac{f}{7{,}5}\right)^2 \qquad (2.2)$$

où f est en kHz.

L'échelle en bark s'avérera d'autant plus pratique lors de l'implémentation du modèle psychoacoustique dans la mesure où elle permet une séparation rapide des différentes bandes critiques.

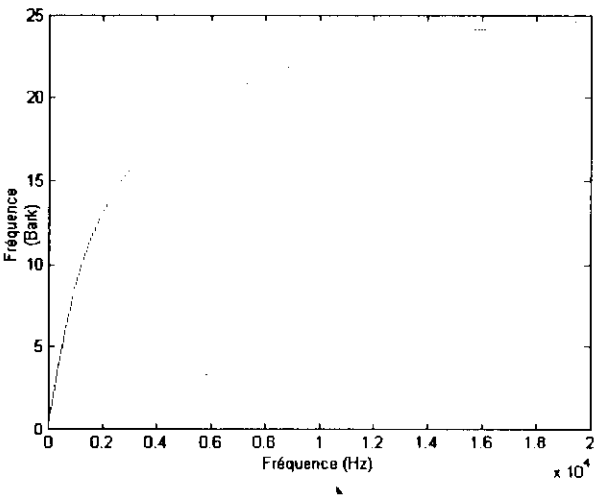


Figure 2.5 : *Relation entre les fréquences en Hertz et en Bark.*

2.4. LE MODÈLE PSYCHOACOUSTIQUE

Le modèle psychoacoustique est une modélisation de l'oreille humaine. Son rôle est de déterminer le seuil de masquage pour chaque sous-bande de fréquence, en fonction des composantes de celle-ci et des composantes des sous-bandes voisines. Autrement dit, il permet de déterminer les composantes qui seront audibles (puissance acoustique au dessus du seuil de masquage) et celles qui seront masquées (puissance acoustique en dessous du seuil de masquage).

De nombreux modèles psychoacoustiques ont été proposés [11]. Ils sont généralement utilisés dans les codeurs pour réduire le débit des fichiers audio.

2.5. PRINCIPES GENERAUX DU CODAGE PSYCHOACOUSTIQUE [12]

D'une manière générale, un codec (codeur-décodeur) psychoacoustique a une structure semblable à celle présentée dans la figure 2.6. En premier lieu, le signal d'entrée est divisé en plusieurs sous-bandes de fréquences par un banc de filtres passe-bandes. Le modèle

psychoacoustique va ensuite, pour chaque sous-bande, déterminer le seuil de masquage et calculer le rapport Signal/Masque (SMR). Les composantes spectrales dont le SMR (en dB) est négatif seront considérées comme inaudibles. A l'inverse, celles dont le SMR est positif seront considérées comme audibles.

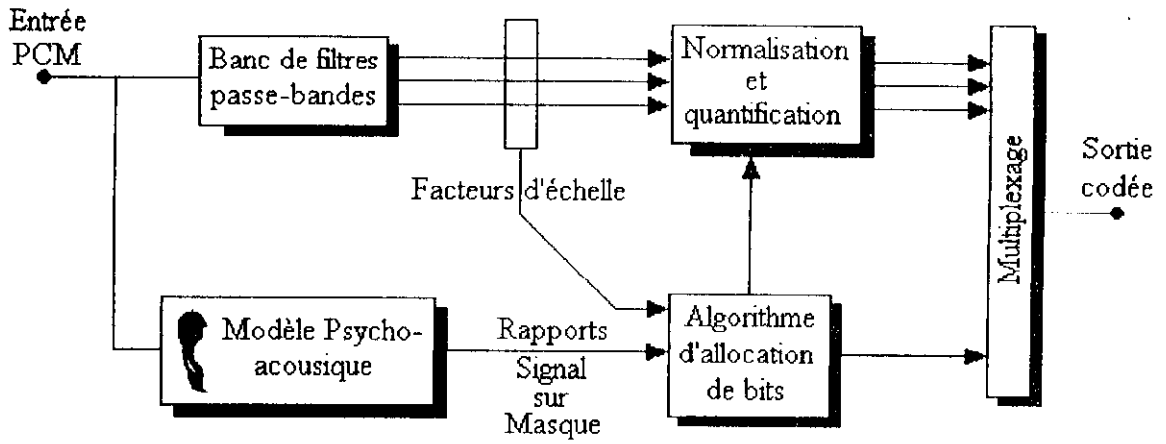


Figure 2.6 : *Structure générale d'un codeur psychoacoustique.*

La valeur du SMR constitue par ailleurs un paramètre déterminant dans le processus d'allocation de bits. Le nombre de bits attribués à chaque sous-bande de fréquence détermine la précision du quantificateur ainsi que la quantité de bruit[3] (qui résultera de la quantification) dans chaque plage de fréquences. Le bruit introduit doit être dans la région spectrale qui est masquée pour ne pas altérer la partie audible du signal audio.

Dans chaque sous-bande (issue du banc de filtres, l'amplitude du signal est, avant sa quantification, normalisée à l'unité. Les facteurs d'échelle nécessaires à la reconstitution du signal original lors du décodage sont conservés. Dans le cas où le signal est au-dessous du seuil de masquage, le facteur d'échelle ainsi que la sortie du quantificateur seront négligées. Ceci permettra de réduire le débit du signal original.

Le décodeur utilise le procédé inverse en générant le signal dans chaque sous-bande à partir des valeurs quantifiées et en multipliant chaque composante du signal par le facteur d'échelle approprié. A la fin, les sorties de toutes les sous-bandes de fréquences sont sommées pour constituer le signal décodé.

[3] Le bruit de quantification est donné par la relation suivante [5] :

$B^2 = k \times P^2 \times 2^{-2b}$ avec $k = \frac{\sqrt{3}}{2}$

où : B : bruit de quantification (en dB),

P : puissance maximale du signal à quantifier,

$b \in N$: nombre de bits sur lequel est réalisée la quantification.

La qualité du son après le codage dépend de la précision du modèle psychoacoustique. Dans le cas où ce dernier venait à considérer qu'une composante spectrale est inaudible alors qu'en réalité elle ne l'est pas, un auditeur percevra des distorsions ajoutés par le codec dans la région où se trouve cette composante. Cependant, même dans le cas où le modèle psychoacoustique prédit parfaitement les composantes qui sont audibles et celles qui ne le sont pas, il reste possible que le bruit soit perceptible si le signal est codé avec un débit trop faible.

Il arrive parfois que le module d'allocation de bits octroie à certaines bandes de fréquences plus de bits que disponible. Dans un tel cas, et pour respecter le débit imposé[4], le nombre de bits octroyé à certaines bandes sera réduit, avec comme conséquence l'introduction de plus de bruit. Le codage à débit variable permet d'éviter un tel problème en affectant à chaque sous-bande le nombre de bits nécessaire à sa quantification, tout en gardant le niveau du bruit au-dessous du seuil de masquage. Cette technique de codage permettra de réduire considérablement le débit lors du codage des passages « silencieux » et de l'augmenter pour le codage des passages plus « bruyants ».

[4] Dans une compression audio à débit constant le nombre de bits par seconde est fixe.

Chapitre 3

LE STANDARD MPEG AUDIO 1

- **Naissance du standard MPEG Audio**
- **Présentation du standard MPEG 1**
- **Les différentes couches du standard**
- **Le banc de filtres**
- **Les modèles psychoacoustiques du standard MPEG 1**
- **Le module d'allocation de bits MPEG**
- **Conclusion**

LE STANDARD MPEG AUDIO 1

Le codeur MPEG audio et de manière générale tous les codeurs perceptuels sont considérés en Watermarking comme des systèmes d'attaque. Ils suppriment en effet les parties inaudibles d'un signal audio pour en réduire le bitrate et peuvent, en conséquence, altérer le watermark éventuellement présent dans ce signal et le rendre indétectable.

Aussi, semble-t-il indispensable de bien maîtriser le mode de fonctionnement du codeur MPEG Audio, et particulièrement son modèle psychoacoustique pour pouvoir effectuer des tatouages robustes, résistants à la compression et qui n'altèrent pas la qualité sonore des fichiers traités.

3.1. NAISSANCE DU STANDARD MPEG AUDIO [5]

L'Organisation Internationale de Normalisation (ISO) et la Commission Electrotechnique Internationale (CEI) constituèrent en 1988 le groupe "ISO/CEI/MPEG[5]" pour mettre au point un standard de compression de haute qualité pour les vidéos. C'est au cours de la même année qu'ils décidèrent d'étendre leurs recherches au domaine de l'audio. Ils créèrent à cet effet le groupe "MPEG Audio". Il aura cependant fallu quatre ans (en 1992) pour que le résultat de ces recherches devienne un standard reconnu.

Le standard mis au point est un format de compression des sons numériques permettant, dans ses versions les plus récentes, d'obtenir un très bon compromis entre la qualité du son compressé et sa taille. A la différence de certains autres formats de compression, le MPEG Audio constitue un format de compression physiologique, dans la mesure où il exploite les caractéristiques de l'oreille humaine pour compresser le son.

3.2. PRÉSENTATION DU STANDARD MPEG 1 [12],[5]

Le standard MPEG Audio comprend trois couches, qui ont permis, au fur et à mesure de leur apparition, d'améliorer les performances du codage. Pour une fréquence d'échantillonnage donnée (32 kHz, 44,1 kHz ou 48 kHz), le MPEG permet de coder les fichiers audio avec des débits allant de 32 à 448 kbit/s par canal. Pour cela, l'algorithme de compression procède de la manière suivante :

- A l'aide d'un banc de filtres, le signal à compresser est, tout d'abord, décomposé dans le domaine fréquentiel en 32 sous-bandes de largeur égale.

[5] MPEG : Moving Pictures Experts Group (en français : Groupe d'experts en images animées).

- Parallèlement, le signal est étudié par le modèle psychoacoustique qui détermine les seuils de masquage et les rapports Signal/Masque[6] (SMR) nécessaires à l'allocation de bits.
- Le module d'allocation de bits utilise les contraintes du SMR et du bitrate pour attribuer à chaque bande le nombre de bits qui lui est nécessaire pour quantifier ses échantillons tout en s'assurant que le niveau du bruit qui résultera de cette quantification soit au dessous du seuil d'audition.
- Les échantillons codés sont ensuite paquetés, associés à des entêtes (headers) et formatés pour constituer les frames du signal compressé.

Ces différentes étapes du processus de compression sont retracées par la figure 3.1 ci-dessous.

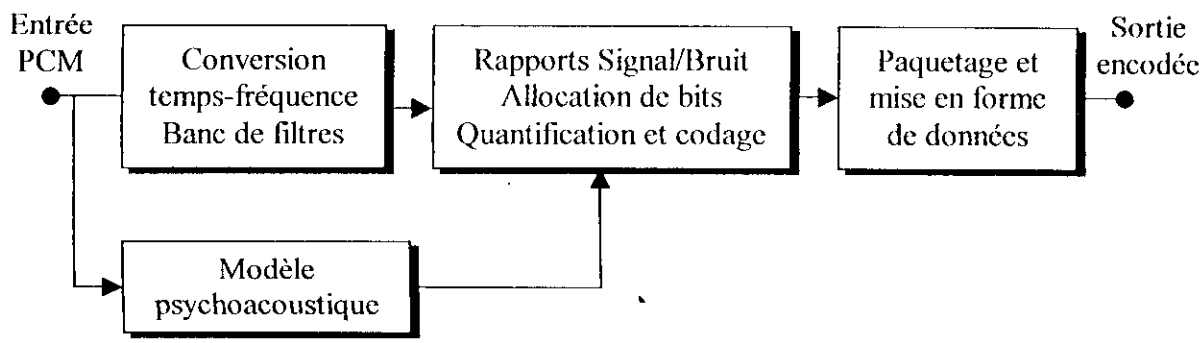


Figure 3.1 : *Schéma fonctionnel de l'algorithme d'encodage MPEG.*

Quant au processus de décompression, il s'opère selon les étapes suivantes :

- le décodeur reçoit les données codées qu'il décompose en frames grâce aux headers,
- il extrait de chaque frame les informations relatives à l'allocation de bits pour pouvoir restaurer les échantillons quantifiés,
- il repasse les valeurs restaurées de chaque sous-bande à travers un banc de filtres de synthèse pour reconstituer le signal audio d'origine.

Le schéma fonctionnel de l'algorithme de décodage est représenté sur la figure suivante.

[6] SMR : différence en dB entre le niveau maximum du signal et le niveau minimum du seuil de masquage.

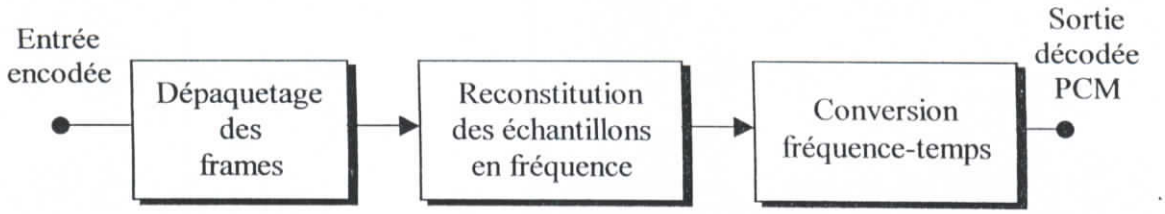


Figure 3.2 : *Schéma fonctionnel de l'algorithme de décompression.*

3.3. LES DIFFÈRENTES COUCHES DU STANDARD MPEG 1 [13]

Les spécificités du standard diffèrent selon les couches.

- La couche 1 (MP1) effectue un codage transparent avec un débit au dessus de 192 kbit/s. Elle utilise des segments de 384 échantillons à raison de 12 échantillons par sous-bande. Dans chacune de ces sous-bandes, les échantillons sont normalisés en utilisant un facteur d'échelle de sorte que la composante dont l'amplitude est la plus élevée soit inférieur ou égal à 1. Ces échantillons sont en suite quantifiés sur un même nombre de bits.

- La couche 2 (MP2) arrive à de bons résultats avec un débit de 128 kbits/s. Son algorithme améliore les performances de compression de la couche 1 en effectuant le traitement sur des segments plus grands contenant 1152 échantillons (3×12 échantillons dans chacune des 32 sous-bandes). Pour éviter les distorsions audibles, le MP2 utilise le plus souvent trois facteurs d'échelle dans une seule sous-bande; ce nombre est cependant réduit dans le cas où les facteurs d'échelle sont de valeurs proches ou quand le masquage temporel permet à lui seul de rendre ces distorsions inaudibles.

  Le MP2 a, par ailleurs, une meilleure résolution[7] spectrale que le MP1. En effet, il effectue des FFT sur 1024 points alors que son prédécesseur ne les effectue que sur 512 points seulement.

La structure du codec MPEG Audio1 pour les couches 1 et 2 est représentée par la figure 3.3 ci-après.

[7] La résolution spectrale est le rapport entre la fréquence d'échantillonnage f_s et le nombre de points de la FFT.

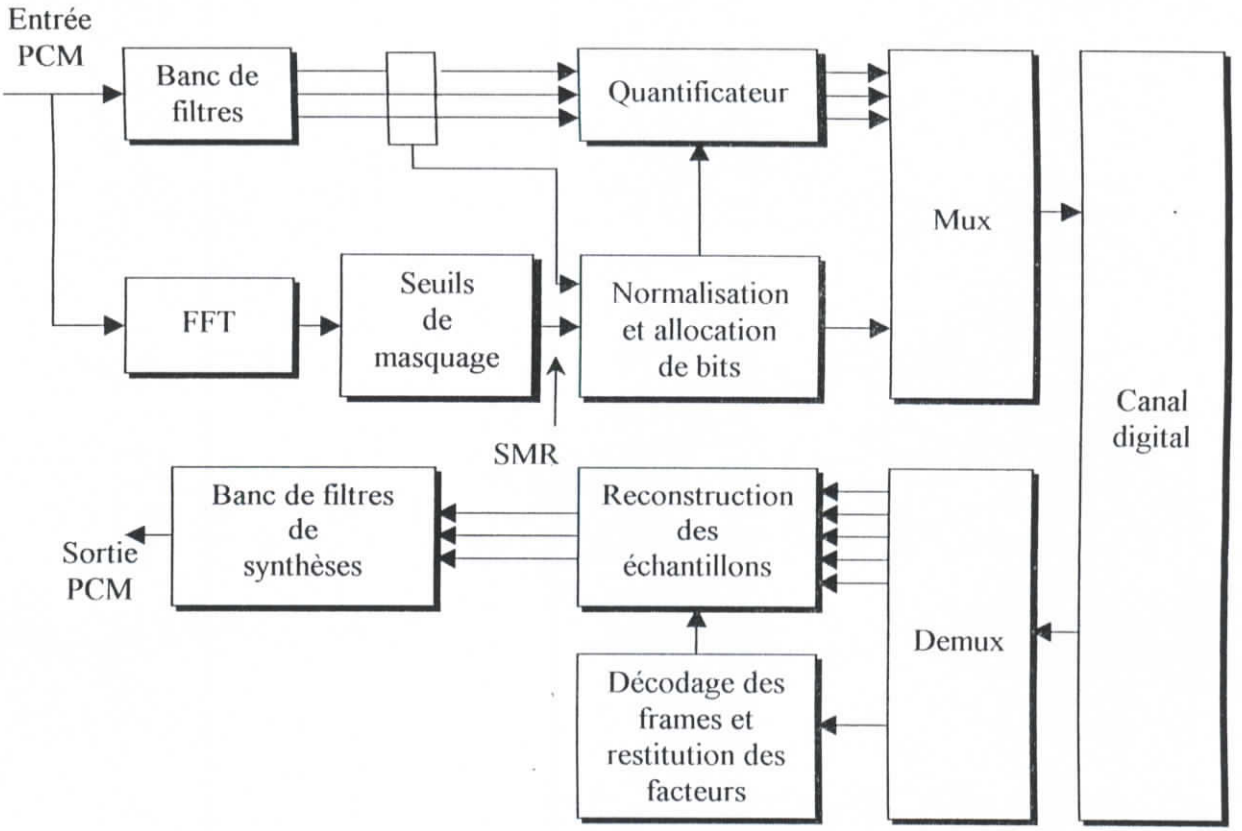


Figure 3.3 : *Structure du codec MPEG Audio1 pour les couches 1 et 2* [12].

- La couche 3 (MP3) atteint une haute qualité de codage avec un débit allant de 64 à 128 kbits/s. Elle utilise une MDCT (Modified Discrete Cosine Transform) à la sortie du banc de filtres pour affiner et corriger les défauts de ce dernier. A la différence des couches précédentes, la couche 3 effectue un encodage avec un débit variable lui permettant d'utiliser le minimum de bits sans contrainte de débit fixe. Pour améliorer les taux de compression, ce format utilise une quantification dynamique (non uniforme) et le codage de Huffman.

La structure du codec MPEG Audio1 pour la couche 3 est représentée par la figure 3.4.

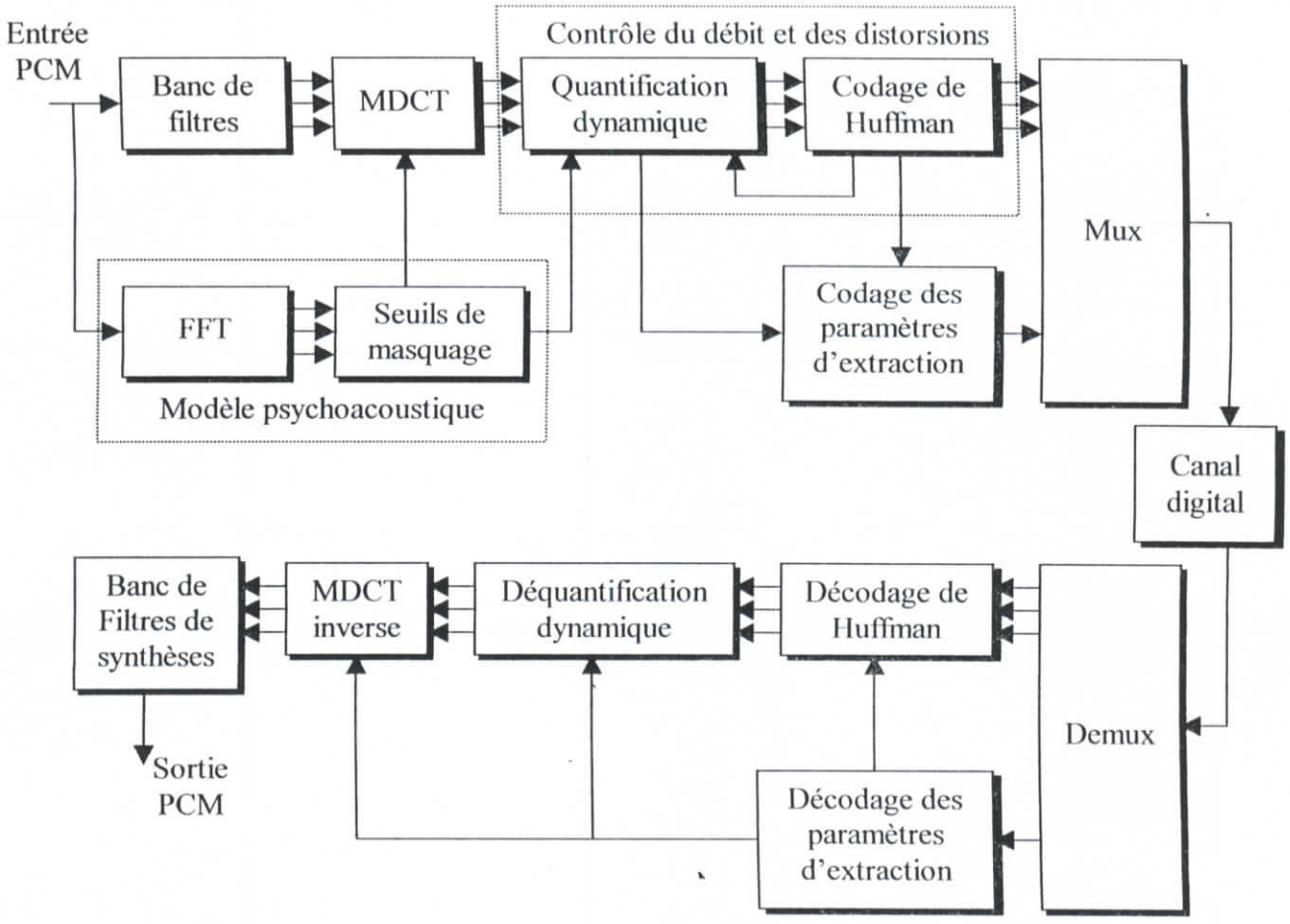


Figure 3.4 : *Structure du codec MPEG Audio1 pour la couche 3* [12].

3.4. LE BANC DE FILTRES [7]

Les couches 1 et 2 du standard MPEG 1 utilisent un banc de filtres qui décompose l'axe fréquentiel en 32 sous-bandes permettant de faire passer le signal audio du domaine temporel vers le domaine fréquentiel. Dans la couche 3, une MDCT est ajoutée à la sortie de ce banc de filtres pour en améliorer la résolution. C'est cette décomposition en fréquences qui permet d'effectuer un codage psychoacoustique dans la mesure où le système auditif humain perçoit les signaux audio dans le domaine fréquentiel avec une résolution correspondant à celle des bandes critiques. Cependant, et comme le montre la figure 3.5 ci-après, le banc de filtres décompose l'axe fréquentiel en 32 sous-bandes de largeur égale alors que la décomposition spectrale à laquelle se livre l'oreille humaine décompose l'axe fréquentiel en sous-bandes critiques de largeur variable et proportionnelle à la fréquence. Ainsi une sous-bande du banc de filtres va recouvrir plusieurs sous-bandes critiques aux basses fréquences.

Aussi, et pour approcher au maximum la résolution en fréquences de l'oreille humaine, le modèle est conçu de telle manière à calculer le seuil de masquage de chaque sous-bande issue du banc de filtres en tenant compte des sous-bandes critiques qui y sont incluses.

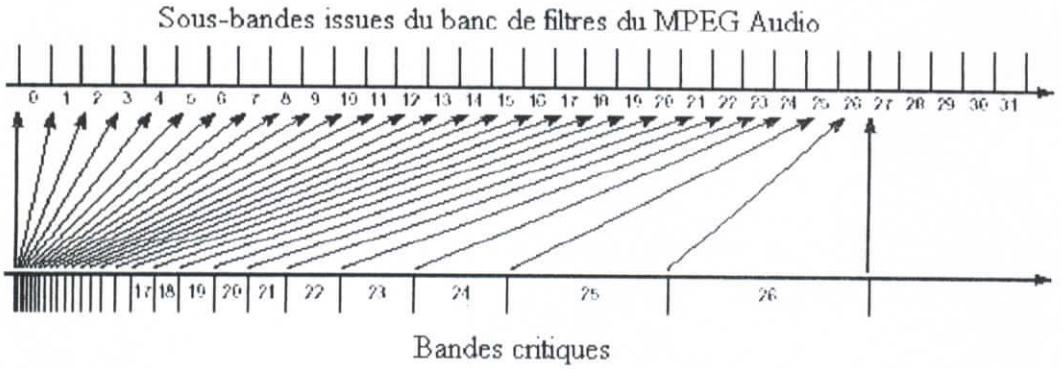


Figure 3.5 : *Correspondance entre la largeur des sous-bandes issues du banc de filtres et celle des sous-bandes critiques* [12].

3.5. LES MODÈLES PSYCHOACOUSTIQUES DU STANDARD MPEG 1 [7], [5]

Le standard MPEG 1 propose deux modèles psychoacoustiques pouvant être utilisés au niveau du codeur. Ils ont pour but de déterminer, pour chacune des 32 sous-bandes issues du banc de filtres, les SMR qui permettront de déterminer quelles composantes sont audibles et lesquelles ne le sont pas. Les SMR sont également indispensables à l'allocation de bits.

3.5.1. Le modèle psychoacoustique 1

Le modèle psychoacoustique 1 est utilisé dans les couches 1 et 2 du standard MPEG Audio[8].

Dans la mesure où la résolution du banc de filtres n'est pas la même que celle de l'oreille humaine (cf. 3.4), le modèle psychoacoustique détermine les seuils de masquage, non en étudiant directement le signal en sortie du banc de filtres, mais en utilisant une transformation temps-fréquence (FFT) indépendante permettant de décomposer ce signal en sous-bandes de largeur variable. Ces sous-bandes, approximation des sous-bandes critiques de l'oreille, permettent un calcul plus approprié des seuils de masquage.

Pour chaque sous-bande critique (notée SBC dans la suite des développements), le modèle 1 différencie les composantes tonales des composantes non-tonales. Cette différenciation s'opère selon les conditions suivantes:

[8] Rien n'empêche que le modèle psychoacoustique 1 soit utilisé dans la couche 3 du MPEG 1.

Soit $s(n)$ ($n \in N$, $1 \leq n \leq 512$) une composante de la fenêtre d'analyse (de taille 512 échantillons). Le modèle 1 calcule ensuite sa puissance $S(k)$.

Cette composante $e(k)$ sera considérée comme tonale par le modèle 1 si et seulement si S(k) satisfait aux trois conditions suivantes :

- $S(k) > S(k+1)$
- $S(k) \geq S(k-1)$
- $S(k) - S(k+j) \geq 7$ dB avec :

  $j \in \{-2,+2\}$ si $2<k<63$

  $j \in \{-3,-2,2,3\}$ si $63 \leq k<127$

  $j \in \{-6,-5,-4,-3,-2,2,3,4,5,6\}$ si $127 \leq k \leq 250$.

Cette différenciation est rendue nécessaire du fait que les caractéristiques de masquage pour chacun de ces deux types de composantes sont différentes.

A. Calcul du seuil de masquage d'une composante tonale

Le seuil de masquage d'une composante tonale est calculé par l'équation suivante :

$$LT_{tm}(j,i) = X_{tm}(j) + av_{tm}(j) + vf(j,i) \qquad (3.1)$$

dans laquelle :

$LT_{tm}(j,i)$: seuil de masquage de la composante i (bark) en tenant compte des composantes voisines j (bark). $LT_{tm}(j,i)$ exprimé en dB.

$X_{tm}(j)$: niveau de pression acoustique en dB de la composante masquante j,

$av_{tm}(j)$: seuil absolu d'audition à la fréquence j,

$vf(j,i)$: fonction de masquage de la composante j.

Dans l'équation 3.1, le seuil absolu $av_{tm}(j)$, exprimé en dB, est défini par :

$$av_{tm}(j) = -1{,}525 - 0{,}275.j - 4{,}5 \qquad (3.2)$$

La fonction de masquage $vf(j,i)$, quant à elle, est définie par :

$$vf(j,i) = \begin{cases} 17.(dz+1) - (0{,}4.X(j)+6) \text{ (dB)} & -3 \leq dz < -1 \text{ Bark} \\ (0{,}4.X(j)+6).dz \text{ (dB)} & -1 \leq dz < 0 \text{ Barks} \\ -17.dz \text{ (dB)} & 0 \leq dz < 1 \text{ Bark} \\ -(dz-1).(17-0{,}15.X(j)) - 17 \text{ (dB)} & 1 \leq dz < 8 \text{ Barks} \end{cases} \qquad (3.3)$$

où : $dz = i-j$ en bark et $X(j)$ représente $X_{tm}(j)$ et $X_{nm}(j)$.

La fonction de masquage n'a pas d'effet considérable en dehors de la zone s'étalant de –3 à +8 barks par rapport à la composante masquante.

B. Calcul du seuil de masquage d'une composante non tonale

Le seuil de masquage des composantes non tonales est calculé par l'équation suivante :

$$LT_{nm}(j,i) = X_{nm}(j) + av_{nm}(j) + vf(j,i), \qquad (3.4)$$

avec le seuil absolu $av_{nm}(j)$, exprimé en dB donné par :

$$av_{nm}(j) = -1{,}525 - 0{,}175.j - 0{,}5 \qquad (3.5).$$

Pour déterminer le seuil de masquage *Sm* d'une sous-bande issue du banc de filtre (par abréviation : SBF), le modèle sélectionne le minimum des seuils de masquage des SBC incluses dans cette sous-bande. Si cette approche donne de bons résultats aux basses fréquences où les SBC sont relativement étroites par rapport à la SBF, elle peut s'avérer inappropriée aux hautes fréquence où les SBC sont relativement larges et couvrent plusieurs SBF.

3.5.2. Le modèle psychoacoustique 2

Le modèle psychoacoustique 2, utilisé par la couche 3 du standard MPEG Audio, est plus complexe que le modèle 1, mais plus réaliste. En effet, à l'inverse de ce dernier, il n'opère pas une séparation stricte des composantes tonales et non tonales, mais établit un index de tonalité en fonction de la fréquence. Cet index mesure si une composante est plus tonale que non tonale ou l'inverse.

Après avoir calculé les seuils de masquage des différentes SBC, le modèle 2 compare chacun de ces seuils avec le seuil absolu et retient le plus élevé des deux.

Pour déterminer le seuil de masquage des 32 SBF, le modèle psychoacoustique 2 utilise pour les basses fréquences le même principe que le modèle 1. Pour les hautes fréquences, il calcule la moyenne du seuil de masquage dans la partie de la SBC incluse dans la SBF. Ce procédé est expliqué par la figure 3.6 ci-après.

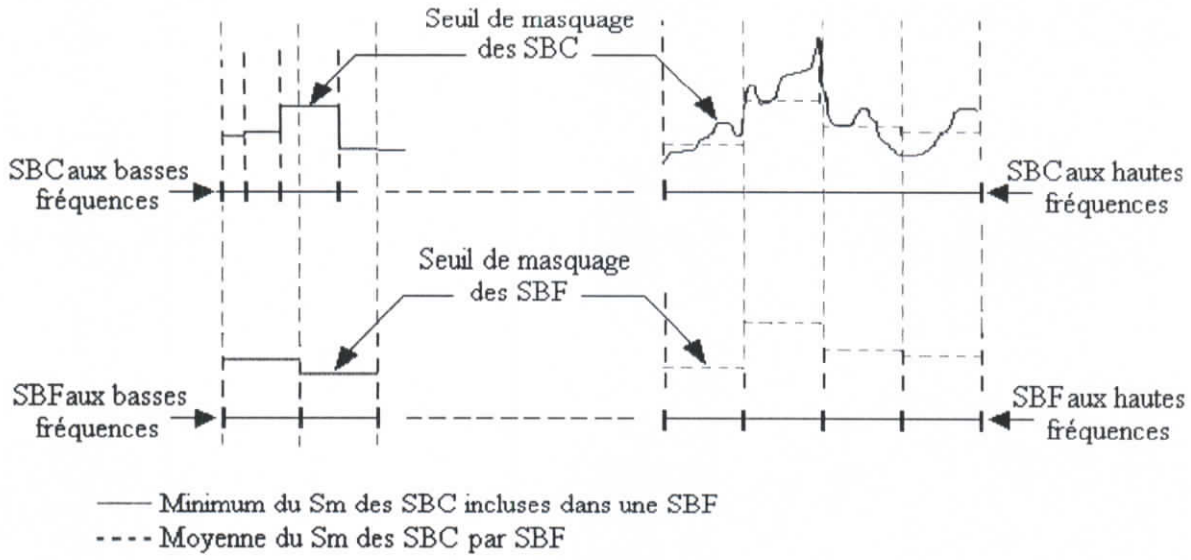


Figure 3.6 : *Détermination du seuil de masquage des sous-bandes issues du banc de filtre selon le modèle psychoacoustique 2.*

Une fois le seuil de masquage de chaque sous-bande de fréquence i ($Sm(i)$) déterminé, le modèle l'utilise pour calculer le rapport Signal/Masque de la SBF i :

$$SMR(i) = E(i)/Sm(i) \qquad (3.6)$$

avec $E(i)$: puissance acoustique maximale à l'intérieur de la SBF i.

Ce rapport constitue une donnée fondamentale pour le module d'allocation de bits.

3.6. LE MODULE D'ALLOCATION DE BITS DU MPEG [5], [13]

Le rôle du module d'allocation de bits est de déterminer le nombre de bits à allouer à chaque SBF pour la quantification de ses échantillons, en réalisant le meilleur compromis possible entre les contraintes du bitrate et le rapport Signal/Masque. Ce nombre varie de 0 à 15 (excepté 1dans la mesure où un bit ne peut jamais être alloué seul) et est fixé de manière à obtenir un bruit de quantification juste inférieur au seuil de masquage. Pour cela, le module calcule le rapport Masque/Bruit (MNR) pour chaque SBF en fonction du rapport Signal/Bruit (SNR) et du rapport Signal/Masque (SMR). Ce calcul est opéré au moyen de la formule suivante :

$$MNR(i) = SNR(kb) - SMR(i) \text{ (dB) } 1 \le i \le 32 \qquad (3.7).$$

Le SMR(i) est donné par le modèle psychoacoustique alors que le SNR(kb) est fourni par une table d'estimations du rapport Signal/Bruit résultant d'une quantification sur un certain nombre de bits kb ($0 \le kb \le 15$, $kb \ne 1$).

Une fois le MNR calculé pour chacune des sous-bandes, le module recherche la SBF dont le MNR est le plus petit et lui alloue le nombre de bits *kb* qui lui correspond.

Si le nombre de bits *kb* alloué à une SBF s'avérait plus élevé que le nombre de bits disponibles, le module procède à sa réduction (*kb*-1 au lieu de *kb*) et recalcule un nouveau MNR pour la sous-bande. Ce procédé est répété jusqu'à ce qu'il n'y ait plus de bits qui puissent être alloués.

Une fois l'allocation de bits achevée, le codeur MPEG quantifie les composantes de chaque sous-bande puis formate les paquets de données (frames) selon des normes précises, variables d'une couche à une autre.

3.7.CONCLUSION

Le MPEG Audio constitue ainsi un format de haute compression du son qui en préserve la qualité. Il est fondé sur le principe du « codage perceptuel » en vue de réduire au maximum, par une cascade d'opérations, la quantité d'informations pour ne garder que celles nécessaires à la perception intégrale du son par l'oreille humaine.

Sa couche 3, plus connue sous le nom MP3 (MPEG Layer III) est de loin la plus utilisée, notamment sur Internet dans la mesure où elle offre un rapport Poids de données/Qualité d'écoute assez exceptionnel. Elle constitue de ce fait une attaque des plus courantes quand bien même elle n'est pas toujours effectuée avec une intention malhonnête. Aussi, est-t-il impératif que les techniques de tatouages auxquelles on peut avoir recours puissent lui résister.

Outre la compression MP3, diverses autres techniques peuvent également être utilisées pour attaquer les systèmes de watermarking. Le chapitre qui suit leur est consacrée.

Chapitre 4

ATTAQUES CONTRE LES SYSTÈMES DE MARQUAGE

- **Définitions et classification des techniques d'attaque**
- **Attaques visant à enlever le watermark**
- **Attaques cryptographiques**
- **Attaques sur le protocole**
- **Attaques de désynchronisation**
- **Conclusion**

ATTAQUES CONTRE LES SYSTÈMES DE MARQUAGE

4.1. DÉFINITIONS ET CLASSIFICATION DES TECHNIQUES D'ATTAQUE

Au cours de ces dernières années, de nombreuses techniques de watermarking ont été proposées pour empêcher ou du moins réduire les copies illégales des médias numériques. Le plus difficile a été de trouver le moyen d'insérer des tatouages qui soient robustes. En effet, les fichiers tatoués sont susceptibles de subir de nombreuses manipulations, qu'elles soient volontaires ou fortuites : compression avec pertes, amplification du signal, conversion analogique-digitale (A/D) puis digitale-analogique (D/A), ajout du bruit, etc. Ces manipulations peuvent, à la longue, causer des dégradations aux fichiers tatoués et altérer, en conséquence, les watermarks qui y sont insérés.

D'une manière générale, tous les traitements susceptibles d'altérer les watermarks ou de provoquer des ambiguïtés lors de leur extraction sont appelés *attaques*. Celles-ci sont considérées comme réussies si elles parviennent à rendre la détection du watermark impossible, sans altérer la qualité sonore du fichier attaqué.

De nombreuses méthodes de tatouage récemment proposées, pourtant qualifiées de « robustes » par leurs inventeurs, ont montré leurs limites face à des attaques plus ou moins "basiques". Les critères utilisés pour démontrer cette robustesse étaient en effet souvent inadéquats [14].

Sans cesse remises en cause, les techniques de tatouage numérique suivront inévitablement, comme pour la cryptographie, une évolution itérative : des algorithmes sont élaborés, des attaques sont trouvées, de nouveaux algorithmes apparaîtront et ainsi de suite.

Dans le domaine du watermarking, les systèmes d'attaques peuvent être différenciés en deux grandes catégories : les *attaques générales* et les *attaques spécifiques*.

Les *attaques générales*, également appelées attaques aveugles *(blind attacks)*, sont les premières auxquelles un attaquant éventuel a recours dans la mesure où elles ne tiennent pas compte des algorithmes utilisés lors du tatouage et peuvent être appliquées contre n'importe quel outil de marquage

Dans le cas où les attaques générales s'avèrent inefficaces, l'attaquant peut recourir aux attaques spécifiques, à condition cependant de connaître l'algorithme utilisé. En effet, chaque attaque de ce type vise un type d'algorithmes particulier. L'attaque contre l'*echo hiding* [15] en est un exemple.

Les différents types d'attaques peuvent être regroupés en quatre grandes classes, suivant le domaine visé [16] :

- les attaques visant à enlever le watermark,
- les attaques cryptographiques,
- les attaques sur le protocole,
- les attaques de désynchronisation.

4.2. ATTAQUES VISANT A ENLEVER LE WATERMARK *(Removal attacks)* [16]

Ces attaques visent à rendre le watermark indétectable dans le fichier tatoué, sans casser la sécurité de l'algorithme d'insertion (i.e sans utiliser la clé). Parmi ces attaques, on trouve :

4.2.1. Introduction du bruit dans le signal audio *(Attack by noise addition)* [17]

L'attaquant ajoute au signal audio un autre signal qui consiste en un bruit blanc; cet ajout a pour effet de modifier légèrement les échantillons du signal audio et peut éventuellement rendre le watermark inséré indétectable.

4.2.2. Filtrage numérique *(Digital filtering)* [3]

Ce type d'attaque regroupe celles qui peuvent être représentées par des filtres mathématiques simples, tels que les *filtrages passe-bas, décalages en temps (time shifting), amplifications, ajouts ou suppressions d'échantillons (cropping)*, etc.

4.2.3. Codage avec pertes *(Lossy coding)* [3]

Certains formats de compression des fichiers audio, comme le format MPEG par exemple, consistent en une compression avec une perte d'informations. Cette perte d'information est considérée comme une attaque dans la mesure où l'encodeur, en supprimant les parties inaudibles d'un média audio, peut également supprimer le filigrane qui, par définition, est conçu de manière à être inaudible.

Ces changements de format ne résultent pas forcement d'une intention malhonnête : ils sont souvent simplement utilisés pour optimiser les capacités de stockage.

4.2.4. Réechantillonnage et requantification *(Resampling and requantization)* [3]

La conversion des médias audio du numériques à l'analogique puis leur reconversion au numérique est une autre forme d'attaque. Elle consiste à faire jouer un média audio numérique sur un haut-parleur et l'enregistrer grâce à un microphone, puis le numériser à nouveau. Lors des deux conversions, requantification et réechantillonnage peuvent causer la destruction des données du filigrane.

4.2.5. Attaque par collusion *(Collusion attack)* [16]

L'attaque par collusion n'est applicable que lorsqu'il est mis à la disposition de l'attaquant un certain nombre de copies d'un fichier, tatouées chacune par un watermark différent ou au

moyen d'une clé différente. En comparant les différentes copies, l'attaquant peut, dans certains cas, localiser le watermark et ainsi, le modifier ou le supprimer.

Des études récentes ont montré qu'une dizaine de copies peut suffire pour conduire avec succès son attaque.

4.2.6. Attaques par restauration *(Attack by restoration)* [14]

Les techniques de restauration audio ont, durant de longues années, fait l'objet d'études et se sont avérées très utiles pour supprimer les dégradations (clics, éraflures, crackles, etc...) localisées dans les enregistrements anciens [18]. Après localisation des ces dégradations, l'algorithme de restauration supprime les échantillons concernés et interpole le signal en utilisant les échantillons voisins. C'est en se basant sur une telle démarche que de nombreuses attaques ont été développées : le signal filigrané[9] est « reconstruit » bloc par bloc en utilisant un segment du signal considéré. Des paramètres auto-régressifs sont déterminés et utilisés pour estimer les autres blocs.

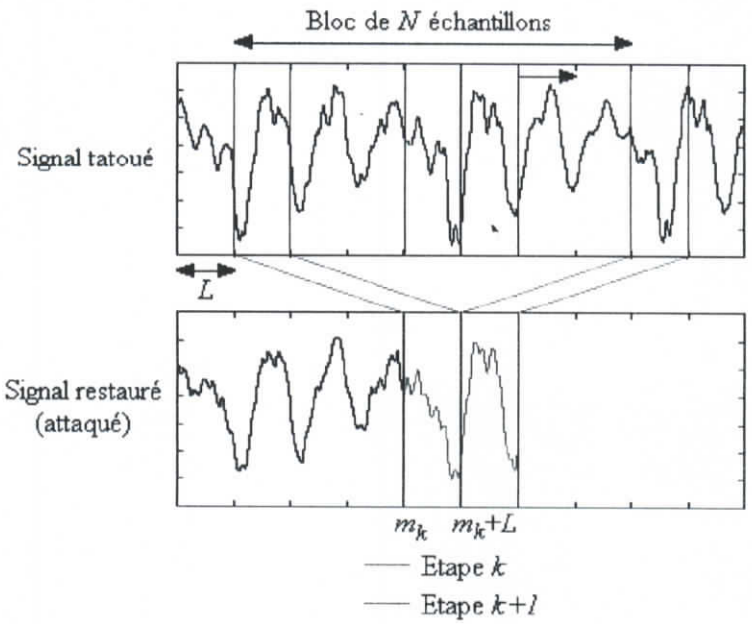


Figure 4.1 : *Représentation de deux étapes de l'algorithme de restauration.*

L'attaque utilise un signal audio filigrané comme entrée et le reconstruit entièrement bloc par bloc. En pratique, l'algorithme est appliqué sur un bloc de $N = 1000$ échantillons provenant de l'entrée et leur applique un algorithme de restauration pour reconstruire un bloc de $L = 80$ échantillons commençant à l'échantillon $m = 460$ (par conséquent le bloc reconstruit se trouve au milieu des 1000 échantillons). Ce bloc est apposé sur la sortie comme illustré

[9] La méthode suppose que le signal à interpoler est auto-régressif d'ordre fini.

dans la figure 4.1. Un autre ensemble de N échantillons est de nouveau extrait à partir de l'entrée (après décalage de L échantillons de la fenêtre coulissante) et un nouveau bloc reconstruit est produit, et ainsi de suite.

Pour chaque restauration, ce sont les échantillons originaux qui sont utilisés pour éviter une dérive du processus entier et la perte importante de qualité qui s'ensuivrait.

La restauration sans erreur est théoriquement possible. Mais à la différence des applications habituelles de la restauration audio, la restauration comme moyen d'attaque ne peut se concevoir sans erreurs : en effet, l'attaque contre le filigrane suppose que la sortie soit différente de l'entrée. Pour cela, la qualité de la sortie est ajustée en diminuant le nombre L des échantillons inconnus.

Cette technique d'attaque a été utilisée contre la méthode de *BlueSpike* [19], une des méthodes de tatouage auxquelles recourt l'*International Federation of the Phonographic Industry* (*IFPI*). Après que des fichiers tatoués avec la méthode de *BlueSpike* aient été reconstruits, le détecteur n'a retrouvé les filigranes insérés dans aucun des fichiers attaqués.

Les différentes attaques décrites ci-dessus n'arrivent cependant pas toujours à supprimer entièrement le watermark. Elles arrivent néanmoins à lui causer de sérieux dommages qui rendent impossible sa détection.

4.3. ATTAQUES CRYPTOGRAPHIQUES *(Cryptographic attacks)* [16]

Contrairement aux précédentes, les attaques cryptographiques ont pour but de casser la sécurité de l'algorithme de tatouage en tentant de reconstituer la clé qui permet d'enlever le watermark dissimulé. Différentes techniques sont utilisées pour atteindre cet objectif.

4.3.1. La recherche exhaustive *(Brute-force search)*

Cette technique d'attaque nécessite l'utilisation de calculateurs très puissants qui tenteront de retrouver la clé utilisée pour le tatouage, en essayant toutes les clés possibles jusqu'à en trouver la bonne. Cette technique reste cependant peu utilisée du fait qu'elle nécessite beaucoup de temps, particulièrement dans le cas où la longueur de la clé n'est pas connue par l'attaquant.

4.3.2. La reconstitution de la clé secrète *(Key recovery)*

Des attaques par corrélation peuvent être employées pour reconstituer la clé utilisée lors du tatouage du média audio original. Si un watermarker utilise la même clé pour coder différents médias audio numériques, les résultats de ces encodages peuvent parfois être corrélés pour identifier l'endroit où le filigrane a été inséré et/ou pour retrouver la clé secrète qui a été utilisée lors du tatouage.

4.4. ATTAQUES SUR LE PROTOCOLE *(Protocol attacks)* [20], [21]

A la différence des autres types d'attaques, les attaques sur le protocole ne cherchent ni à détériorer le watermark ni à empêcher sa détection. Ces attaques, qui visent le protocole, ont pour but de créer des ambiguïtés lors du processus d'extraction.

4.4.1. Attaque par copies *(Copy attack)*

Kutter et al [20] ont montré qu'il était possible de copier une marque à partir d'un fichier tatoué et de la mettre dans un autre fichier-cible sans avoir à utiliser les séquences de la clé ou du watermark. Dans cette technique, appelée attaque par copies, l'attaquant estime en premier le fichier original à partir du fichier tatoué et en déduit le watermark. Il insère ensuite ce watermark dans un autre fichier (fichier-cible), après l'avoir modelé de sorte qu'il soit juste au dessous du seuil d'audibilité. Cette attaque (figure 4.2) peut être formulée de la manière suivante [20] :

Soit $T(I,D)$ l'algorithme utilisé pour tatouer le signal original S_1 avec le watermark w et la clé k; le signal tatoué étant S_1'.

Le processus d'insertion I est donné par :

$$I(S_1,w,k)=S_1'$$

Le processus de détection D peut être formulé comme suit :

$$D(S_1,S_1',w,k)=True.$$

L'algorithme de tatouage T sera susceptible d'une attaque par copies dès lors qu'il existe un procédé permettant d'obtenir à partir de S_1' et S_2 (S_2 étant un signal quelconque) un signal S_2' tatoué avec k et w, tel que :

$$D(S_2, S_2',w,k)=True.$$

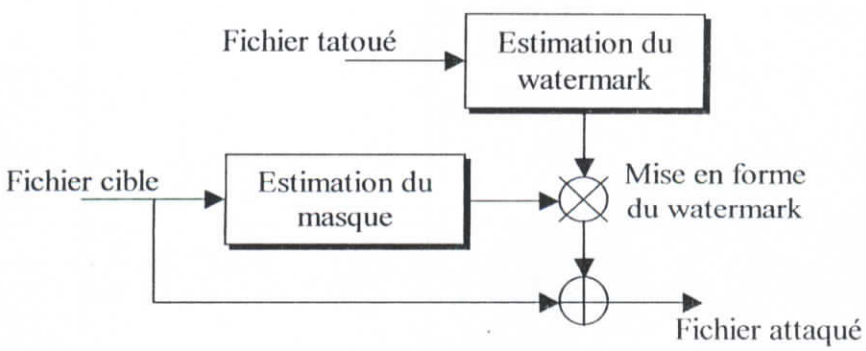


Figure 4.2 : *Attaque par copies.*

Pour résister à ce type d'attaques, un algorithme doit générer des signaux watermark étroitement liés au signaux originaux. Ainsi, même dans le cas où un attaquant arrive à estimer le watermark, il lui sera impossible de l'insérer dans un autre signal sans dégrader la qualité de celui-ci.

4.4.2. Attaque par inversion : Problème de l'impasse *(Deadlock problem)* [21]

Un autre type d'attaque sur les protocoles a pour cible les systèmes de marquage inversibles. Du fait que la plupart des outils de marquage ne permettent pas de savoir quel tatouage parmi plusieurs a été inséré le premier, un attaquant peut insérer son propre watermark d'un fichier déjà tatoué et prétendre qu'il en est le propriétaire. Ce problème, dit « problème de l'impasse », peut être formulé comme suit :

Soit $T(I,D)$ l'algorithme utilisé pour tatouer le signal original S avec le watermark w_1 et la clé k_1; le signal tatoué étant S'.

Le processus d'insertion I est donné par :

$$I(S,w_1,k_1)=S'$$

Le processus de détection D peut être formulé comme suit :

$$D(S,S',w_1,k_1)=True.$$

On dira que l'algorithme T est susceptible d'une attaque par inversion si un attaquant peut, à partir de S', inverser le processus et trouver un autre signal S'', une clé k_2 et un watermark w_2 de sorte que lors de la détection on ait :

1. $\boldsymbol{D(S'',S',w_2,k_2)=True}$: l'attaquant proclamera qu'il a tatoué le signal S'' avec la clé k_2 et le watermark w_2, et que le signal tatoué obtenu est S',

2. $D(S'',S,w_2,k_2)=True$: l'attaquant peut extraire son watermark du signal original du vrai propriétaire,

3. $D(S,S'',w_1,k_1)=True$: le vrai propriétaire peut extraire son watermark du signal original de l'attaquant.

Dans ce cas précis, une ambiguïté se posera : personne ne pourra donc dire qui est le véritable propriétaire du signal tatoué S'.

4.5. ATTAQUES DE DÉSYNCHRONISATION [16]

Dans la plupart des systèmes de marquage, une synchronisation est nécessaire avant le processus de détection du watermark. Cette synchronisation permet au détecteur de repérer l'emplacement de la marque dans le fichier tatoué avant de procéder à son identification. Les attaques de désynchronisation visent justement à mettre en défaut cette synchronisation et empêcher ainsi l'extracteur de repérer le watermark.

Plusieurs techniques de synchronisation sont utilisées. L'une d'elles consiste à rajouter un en-tête *(header)* à la séquence du watermark. En utilisant une copie de cet en-tête, le détecteur va pouvoir localiser le watermark et procéder à son identification. L'attaque contre ce type de synchronisation consistera à repérer puis à enlever l'en-tête pour empêcher la localisation et la détection du watermark.

Une autre technique de synchronisation est basée sur la localisation des maximums dé la séquence du watermark dans le domine fréquentiel. Il suffira au détecteur de retrouver ces points pour pouvoir extraire ces derniers. Une des attaques mises au point contre cette méthode (figure 4.3) consiste successivement à estimer le watermark inséré, à le transposer du domaine temporel au domaine fréquentiel (en utilisant une transformée de Fourier), à détecter les amplitudes maximales de la séquence estimée et enfin à supprimer ces dernières. Pour minimiser l'effet des distorsions introduites par ces modifications, le watermark est interpolé au niveau des composantes qui ont été supprimées. L'oreille humaine étant sensible aux variations de phases, cette interpolation est effectuée en tenant compte des arguments des composantes supprimées.

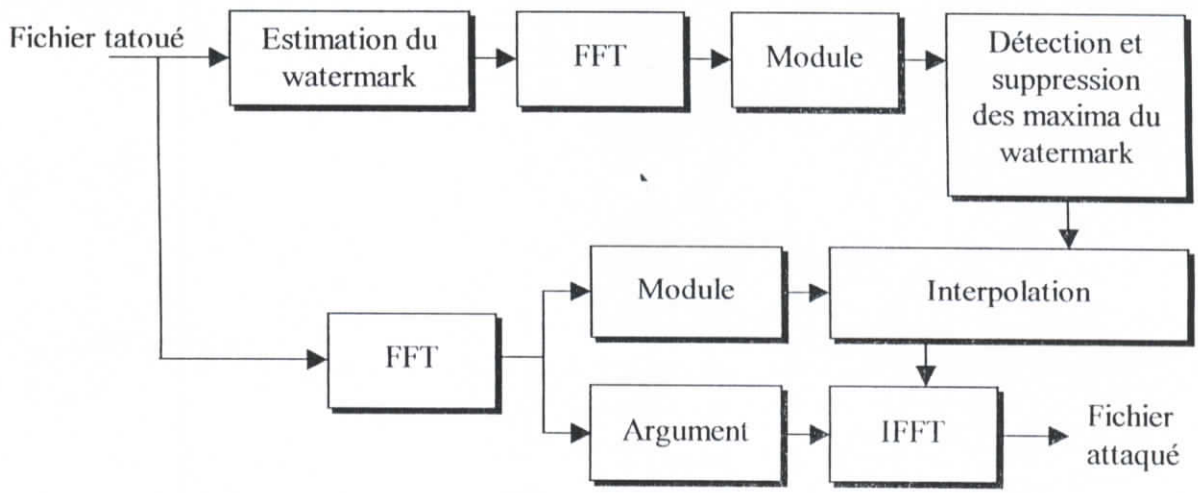


Figure 4.3 : *Attaque de désynchronisation.*

4.6. CONCLUSION

Si les systèmes de marquage sont généralement résistant aux attaques de base, ils le sont beaucoup moins contre des combinaisons de celles-ci. La connaissance des principales techniques d'attaque utilisées s'avère en conséquence indispensable pour l'élaboration et la mise au point d'algorithmes de marquage pouvant leur résister. L'idéal serait cependant de disposer de bancs d'essai, comme *StirMark audio* [22], qui permettent, grâce à une batterie d'attaques d'évaluer la robustesse et la qualité des algorithmes de marquage. Des outils de l'espèce spécifiques au domaine de l'audio restent, cependant, encore rare sur le marché.

Chapitre 5

ÉTAT DE L'ART DES TECHNIQUES DE TATOUAGE AUDIO

- **Travaux réalisés dans le domaine du watermarking audio**
- **Watermarking audio dans le domaine temporel**
- **Watermarking audio dans le domaine fréquentiel**

ÉTAT DE L'ART DES TECHNIQUES DE TATOUAGE AUDIO

5.1. TRAVAUX RÉALISÉS DANS LE DOMAINE DU WATERMARKING AUDIO

Plusieurs articles font la synthèse des différentes techniques du tatouage numérique [23], [24], [25]. Il ressort de ces articles que les recherches menées dans le cadre de la protection des images et des vidéos ont été engagées bien plus tôt que celles destinées aux médias audio et sont de ce fait bien plus en avance. Cependant, avec l'apparition d'algorithmes de compression audio particulièrement performants qui ont permis la diffusion de la musique numérique par Internet, l'élaboration et la mise au point de techniques permettant la protection des fichiers musicaux est devenue une préoccupation majeure, compte tenu de ses implications au plan commercial. C'est ainsi que de nombreuses techniques de protection sont apparues ces dernières années.

Une technique des plus rudimentaires utilise le principe du tatouage par modification d'amplitude. Elle consiste à modifier l'amplitude du signal original en remplaçant le bit le moins significatif de chaque échantillon audio par la donnée à dissimuler (le watermark) qui généralement est une séquence pseudo-aléatoire [26].

Dans [27], une méthode de tatouage par codage de phase est employée pour inclure un signal inaudible. Le signal audio est divisé en segments et la phase du premier segment est remplacée par une phase de référence tandis que la phase des segments suivants est ajustée pour la conservation des phases relatives[10].

Les techniques présentées dans [28], [29] et [30] opèrent dans le domaine fréquentiel afin de tirer profit des caractéristiques du système auditif humain. Elles ont pour objet d'insérer des watermarks inaudibles, même dans le cas où ceux-ci sont d'amplitude élevée.

Dans [31], la technique proposée est basée sur l'étalement de spectre : elle considère le signal original comme un canal de transmission et le filigrane comme le signal à transmettre. Ses concepteurs insistent sur le fait que le filigrane, pour être robuste et résistant aux attaques, doit être placé dans les parties du spectre du signal à tatouer les plus significatives en terme de perception auditive.

Dans [32], l'information à dissimuler est incluse dans les signaux audio sous forme d'échos étroitement espacés. Les « 0 » et les « 1 » sont codés en employant différents retards entre le signal original et l'écho. Ces retards (de l'ordre de la milliseconde) sont situés au-dessous du seuil à partir duquel le signal original et l'écho sont perçus par l'oreille humaine comme provenant de deux sources différentes. Pour un délai inférieur à ce seuil, l'écho est considéré comme une distorsion ou une simple redondance du signal original. Bien que cette

[10] l'oreille n'est pas sensible aux valeurs absolues des phases, mais elle l'est pour leurs variations relatives.

méthode soit assez résistante à la compression MPEG et à la conversion D/A, son utilisation n'est cependant pas recommandée pour les signaux contenant des périodes de silence.

Toutes les méthodes citées précédemment opèrent sur les fichiers audio non compressés (au format WAV par exemple). Certaines d'entre elles insèrent des watermarks résistants à la compression-décompression-recompression et peuvent donc permettre de tatouer des fichiers compressés en procédant successivement à leur décompression, à leur tatouage, puis à leur recompression. Si cette procédure assure la robustesse du marquage, elle nécessite cependant beaucoup de temps, du fait de la lenteur du processus de compression. Elle ne convient donc pas aux transactions en ligne. Pour réduire le temps d'insertion, certains algorithmes opèrent directement sur des fichiers compressés.

Dans [33], Sandford et al, proposent un algorithme qui tatoue directement des fichiers compressés. Cette méthode n'est cependant pas robuste car le watermark peut être supprimé au cours du processus de décompression-recompression sans altérer la qualité du son.

Une autre approche a été proposée dans [34] : l'insertion du watermark se fait non pas après mais pendant la compression. Cette méthode (MP3stego) se caractérise cependant par deux grandes faiblesses : son manque de robustesse et sa lenteur. En effet, elle n'insère pas le watermark directement sur des données compressées mais pendant la compression et nécessite, en conséquence, beaucoup de temps.

Dans [35], K. Nahrstedt expose deux techniques permettant d'insérer des filigranes lors d'une compression MP3, selon le même principe que MP3stego. Dans la première, l'insertion du watermark s'opère en modifiant les facteurs d'échelle; dans la seconde, elle passe carrément par la modification de certains échantillons codés, et ce dans chaque frame. La question de la robustesse de ces méthodes n'a cependant pas été soulevée.

5.2. WATERMARKING AUDIO DANS LE DOMAINE TEMPOREL

Les méthodes de watermarking audio opérant dans le domaine temporel permettent d'insérer les watermarks en procédant directement à une modification de l'amplitude des échantillons du signal audio. Quelques méthodes entrant dans cette catégorie sont présentées ci-après.

5.2.1. Watermarking audio basé sur la distribution de l'énergie du signal dans le temps

Dans [36], une nouvelle méthode basée sur la relation existant entre les énergies de trois blocs d'échantillons adjacents et permettant l'insertion de tatouages dans le domaine temporel est présentée. Cette méthode prend en considération les propriétés du système auditif humain (HAS) pour faire en sorte que les distorsions insérées dans le signal hôte soient inaudibles.

A. Insertion du watermark

Dans une première étape, l'algorithme d'insertion procède à une segmentation du signal original $f(x)$ en blocs de même longueur (L). Il opère ensuite, comme le montre la figure 5.1 ci-dessous, sur des ensembles de trois blocs successifs, notés respectivement seg_1, seg_2 et seg_3. Leurs énergies respectives E_1, E_2 et E_3 sont définies et calculées comme suit :

$$E_1' = \sum_{x=x_i}^{x_i+L-1} |f(x)|^2,$$
$$E_2' = \sum_{x=x_i+L}^{x_i+2L-1} |f(x)|^2, \qquad (5.1)$$
$$E_3' = \sum_{x=x_i+2L}^{x_i+3L-1} |f(x)|^2,$$

où x_i est le premier échantillon du segment seg_1. Les énergies E_1, E_2 et E_3 sont ensuite classées et re-notées E_{max}, E_{mid} et E_{min} pour le calcul des deux paramètres A et B :

$$A = E_{max} - E_{mid} \qquad (5.2)$$

$$B = E_{mid} - E_{min} \qquad (5.3)$$

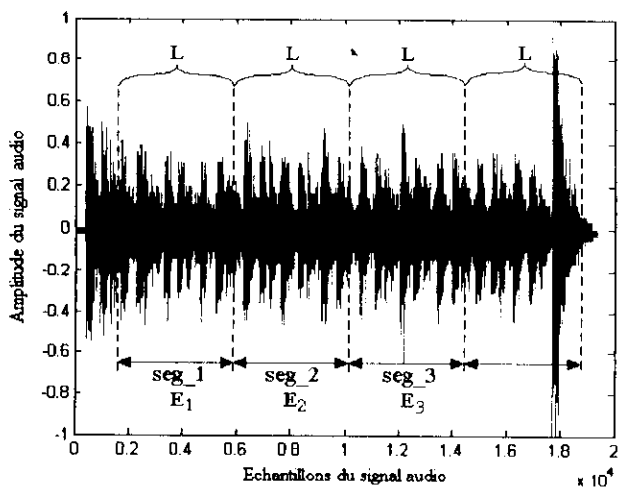


Figure 5.1 : *Segmentation du signal original.*

L'algorithme consiste à insérer un bit du watermark pour chaque groupe de trois blocs successifs du signal original, en fonction des paramètres A et B. Cette insertion s'opère selon la procédure décrite ci-après.

Pour insérer un « **1** », l'algorithme teste la condition suivante :

$$(A\text{-}B) \geq (E_{max} + 2.E_{mid} + E_{min}).d' \qquad (5.4)$$

Dans le cas où l'équation 5.4 est vérifiée, l'algorithme passe directement au traitement du groupe suivant. Dans le cas contraire, il augmentera l'énergie E_{max} ou diminuera E_{mid} jusqu'à ce que l'équation 5.4 soit vérifiée.

Une procédure analogue est suivie pour l'insertion d'un « **0** ». Si la condition :

$$(B\text{-}A) \geq (E_{max} + 2.E_{mid} + E_{min}).d' \qquad (5.5)$$

est vérifiée, l'algorithme passe au traitement du groupe suivant. Dans le cas contraire, E_{mid} sera augmentée ou E_{min} diminuée jusqu'à ce que cette condition soit satisfaite.

Dans les équations 5.4 et 5.5, d' est un paramètre qui fixe la différence qui doit exister entre A et B pour pouvoir représenter un « 0 » ou un « 1 ». Ce paramètre, qui ne doit pas dépasser un certain seuil pour ne pas affecter la qualité du signal tatoué, détermine également la quantité d'énergie à ajouter ou à soustraire pour insérer le watermark. A cet effet et selon le cas, les énergies E_{max}, E_{mid} et E_{min} sont modifiées en multipliant le segment correspondant par un facteur d'échelle k dont la valeur dépend de d'.

Le fait d'augmenter ou de diminuer l'énergie de certains blocs en les multipliant par un facteur d'échelle constant, engendrerait des discontinuités aux limites de ces blocs et altèrerait, en conséquence, la qualité d'écoute du fichier audio. Pour palier ce problème et préserver la qualité sonore du fichier, les différents blocs sont multipliés par des facteurs d'échelle dont la valeur varie progressivement aux limites, permettant ainsi d'avoir une forme d'onde continue (figure 5.2).

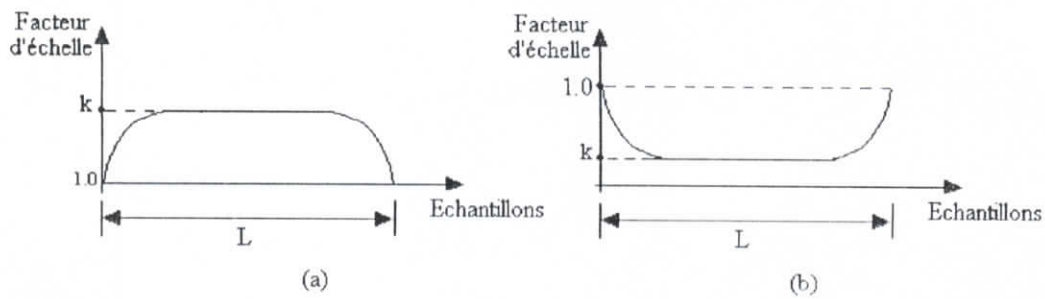


Figure 5.2 : *Variations progressives du facteur d'échelle k aux limites de la fenêtre : (a) amplification, (b) atténuation.*

B. Extraction du watermark

Pour ce qui est de l'extraction, le watermark peut être identifié sans avoir recours au fichier original. Comme pour la procédure d'insertion, l'algorithme procède à la segmentation du signal tatoué en blocs de même longueur L et effectue des tests sur chaque groupe de trois blocs successifs dont il calcule les énergies respectives E', E'_2 et E'_3 (équation 5.1).

Selon leur valeur, ces énergies sont renommées E'_{max}, E'_{mid} et E'_{min} pour le calcul des paramètres A' et B' :

$$A' = E'_{max} - E'_{mid} \qquad (5.6)$$

$$B' = E'_{mid} \quad E'_{min} \qquad (5.7)$$

La comparaison entre A' et B' permettra de déterminer si le bit composant le watermark présent dans le groupe de blocs traité est un « **1** » ou un « **0** ».

Si $A' \geq B'$, le bit sera un « **1** ».

Si $A' < B'$, le bit sera un « **0** ».

5.2.2. Watermarking audio basé sur les périodes de silence [37]

Dans [37], Khalid A. Kaabneh et Abdou Youssef présentent une nouvelle approche pour le tatouage audio, basée sur le fait que la plupart des signaux audio (parole, chants, ou musique instrumentale) contiennent des périodes de silence (figure 5.3). Toutes ces périodes sont considérées comme faisant partie intégrante du signal audio d'origine; elles ne peuvent par conséquent être négligées au risque de causer d'importantes distorsions dans le signal.

Kaabneh et Youssef ont également montré qu'il était possible d'insérer des watermarks en augmentant légèrement la longueur de chacune de ces périodes de silence dans des limites qui permettent d'éviter toute distorsion audible.

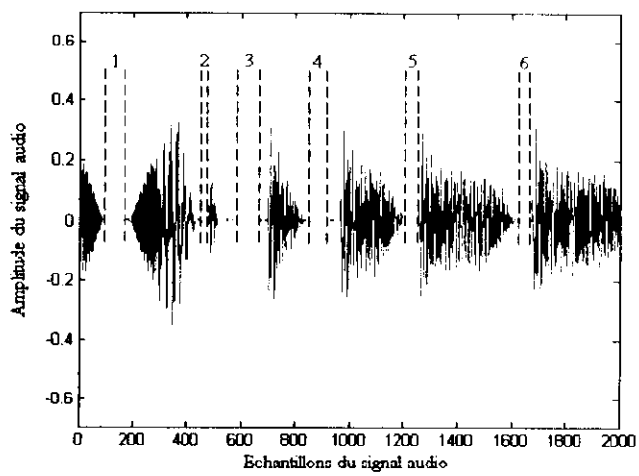


Figure 5.3 : *Les périodes de silence dans un signal audio.*

Ces périodes de silence présentent l'avantage d'apparaître dans les signaux audio de manière aléatoire et de ne pas être supprimées lors d'une éventuelle compression, du fait qu'elles constituent des intervalles temporels réels.

A. Insertion du watermark

La technique de tatouage présentée dans [37] s'applique aux signaux audio codés sur 8 ou 16 bits, avec une fréquence d'échantillonnage de 44.1kHz en stéréo. L'algorithme commence par repérer, dans le fichier à tatouer, toutes les périodes de silence : celles dont l'amplitude est inférieure à un certain seuil pré-déterminé. Il calcule ensuite les durées $M(i)$ de chaque période (i) repérée. Ces durées correspondent au nombre d'échantillons couverts par chaque période.

L'insertion du watermark s'effectue par le rajout d'échantillons dans chacune des périodes de silence dans lesquelles le bit à insérer est un « 1 ». Les échantillons rajoutés doivent être d'une amplitude inférieure à celle d'un seuil préalablement fixé. Lorsque le bit à insérer est un « 0 », les périodes de silence ne subissent aucune modification.

L'insertion du watermark peut être représentée par la formule suivante :

$$M_w(i) = M_o(i) + \delta(i) \qquad (5.8)$$

où : $M_w(i)$ est la durée de la i^e période de silence du fichier tatoué,

$M_o(i)$ est la durée de la i^e période de silence du fichier original,

$\delta(i)$ correspond à la modification éventuellement apportée à la i^e période de silence du fichier original.

B. Extraction du watermark

Dans cette méthode, l'extraction du watermark ne requiert pas le signal original. Il suffit de connaître la longueur des périodes de silence $M_o(i)$ du signal original et celles des périodes de silence $M_w(i)$ du fichier tatoué. En fonction de $\delta(i)$, calculé au moyen de la formule (5.9) ci-dessous, le bit inséré dans chaque période de silence pourra être identifié. Le watermark sera ainsi reconstitué dans son intégralité.

$$\delta(i) = M_w(i) - M_o(i) \qquad (5.9)$$

5.2.3. Watermarking audio robuste dans le domaine temporel [38]

La méthode de tatouage présentée dans [38] opère dans le domaine temporel. Elle permet l'insertion de watermarks en modifiant légèrement l'amplitude des échantillons constituant le signal hôte en veillant à ne pas générer de distorsions audibles. Lors de l'extraction, le signal d'origine n'est pas requis.

A. Insertion du watermark

Soit $x(i)$ le signal audio à tatouer constitué de M composantes ($i=1,\ldots,M$), ce signal étant divisé en N_s segments de N échantillons chacun.

$x_k(i)$ représente le i^e échantillon du segment k; il est donné par :

$$x_k(i)=x(k.N+i) \quad i=0,\ldots,N\text{-}1,\ k=0,\ldots,N_s\text{-}1. \qquad (5.10)$$

Chacun des k segments du signal sera tatoué par la même séquence bipolaire $w(i)\in\{-1,1\}$, avec $i=0,\ldots,N\text{-}1$. Le processus d'insertion du watermark se fait selon les 3 étapes suivantes :

1. L'algorithme commence par moduler le watermark $w(i)$ avec le signal hôte pour générer un watermark dépendant du signal $w_k'(i)$:

$$w'_k(i) = \alpha.|x_k(i)| \oplus w(i) \quad i = 0,\ldots,N-1 \quad k = 0,\ldots,N_s-1 \qquad (5.11)$$

Dans l'équation précédente :

- le symbole $\oplus$ représente une loi de superposition qui peut être une multiplication, une élévation en puissance...etc.
- α représente une constante qui contrôle l'amplitude du watermark. Cette dernière ne doit pas dépasser le seuil à partir duquel le watermark causerait des distorsions audibles.

2. L'algorithme filtre la séquence $w'_k(i)$ par un filtre passe-bas de Hamming d'ordre L suivant l'équation :

$$w''_k(i) = \sum_{l=0}^{L-1} b_l w'_k(i-l) \qquad (5.12)$$

où les b_l représentent les coefficients du filtre de Hamming.

3. L'algorithme insère le watermark modulé et filtré $w_k''(i)$ dans chaque segment x_k pour obtenir le segment tatoué y_k :

$$y_k(i) = x_k(i) + w_k''(i) \qquad (5.13)$$

Le signal tatoué $y(i)$ est obtenu par la concaténation des différents segments $y_k(i)$.

B. Extraction du watermark

Pour procéder à la détection du watermark, l'extracteur fractionne le signal y en N_s segments de N échantillons puis calcule les sommes $S_k(n)$. Celles-ci sont données par l'équation suivante :

$$S_k(n) = \frac{1}{N}\sum_{i=0}^{N-1} y_k[(i+n) \bmod N].w(i) \qquad (5.14)$$

où : $y_k(i)$ est le i^e échantillon du k^e segment de y.

Les $S_k(n)$ représentent les corrélations entre le signal y_k et le watermark $w(i)$; ces corrélations sont évaluées pour tout décalage circulaire de y_k.

La combinaison des équations 5.13 et 5.14 donne :

$$S_k(n) = \frac{1}{N}\left(\sum_{i=0}^{N-1} x_k[(i+n) \bmod N].w(i) + \sum_{i=0}^{N-1} w_k''[(i+n) \bmod N].w(i)\right) \qquad (5.15)$$

D'après l'équation 5.15, la corrélation $S_k(n)$ est égale à zéro dans le cas où la moyenne du watermark m_w ou la moyenne du signal m_x est nulle. Si m_w diffère de zéro, l'on pourra déduire les « –1 » et les « 1 » ne seront pas présents en même nombre dans le watermark et l'écart qui les sépare (Δw) devra être pris en compte pour les traitements ultérieurs. Cet écart est calculé par l'équation 5.16 ci-près.

$$\Delta w = \sum_{i=0}^{N-1} w(i) \qquad (5.16)$$

En supposant que A soit l'ensemble des indices pour lesquels :

$$\sum_{i \in A} w(i) = 0 \qquad (5.17)$$

et B l'ensemble du reste des indices pour lesquels :

$$\sum_{i \in B} w(i) = \Delta w \qquad (5.18)$$

il sera possible d'écrire :

$$S_k(n) = \frac{1}{N}\left(\sum_{i \in A} x_k[(i+n) \bmod N].w(i) + \sum_{i \in B} x_k[(i+n) \bmod N].w(i) + \sum_{i=0}^{N-1} w_k''[(i+n) \bmod N].w(i)\right) \qquad (5.19)$$

ou bien :

$$S_k(n) = T_{1,k}(n) + T_{2,k}(n) + T_{3,k}(n) \qquad (5.20)$$

avec :

$$\begin{aligned} T_{1,k}(n) &= \frac{1}{N}\sum_{i \in A} x_k[(i+n) \bmod N].w(i), \\ T_{2,k}(n) &= \frac{1}{N}\sum_{i \in B} x_k[(i+n) \bmod N].w(i), \\ T_{3,k}(n) &= \frac{1}{N}\sum_{i=0}^{N-1} w_k''[(i+n) \bmod N].w(i). \end{aligned} \qquad (5.21)$$

Il peut facilement être démontré que $E(T_{1,k}(n)) = 0$. Si N_A est grand, il sera possible de dire que $T_{1,k}(n) \approx 0$.

Le terme $T_{2,k}(n)$ quant à lui peut être exprimé selon l'équation 5.22 qui suit :

$$T_{2,k}(n) = sign(\Delta w)\frac{1}{N}\sum_{i\in B} x_k[(i+n) \bmod N] \qquad (5.22)$$

$$= \frac{\Delta w}{N}\frac{1}{N_B}\sum_{i\in B} x_k[(i+n) \bmod N]$$

avec : N_B=cardinal(B).

De l'équation 5.22, il découle que :

$$E\left(T_{2,k}(n)\right) = \frac{\Delta w}{N} m_x \qquad (5.23)$$

Si N_B est suffisamment grand, il sera possible d'écrire : $T_{2,k}(n) \approx \frac{\Delta w}{N} m_x$.

Dans le cas où le signal n'est pas tatoué, le terme $T_{3,k}(n) = 0$. L'équation 5.20 sera donc de la forme : $S_i(n) \approx T_{2,k}(n)$.

Si par-contre le signal est tatoué, l'équation 5.20 s'écrira comme suit :

$$S_i(n) \approx T_{2,k}(n) + T_{3,k}(n) \qquad (5.24)$$

La détection du watermark se fait à partir de l'équation 5.24 par le calcul d'un ratio $r_k(n)$:

$$r_k(n) \overset{\Delta}{=} \frac{S_k(n) - T_{2,k}(n)}{T_{3,k}(n)} \qquad (5.25)$$

Ce ratio tendra vers 1 si le watermark recherché est présent dans le signal y. Il tendra vers 0 dans le cas contraire.

N'étant pas requis pour la procédure d'extraction, le signal original $x(t)$ nécessaire au calcul de $T_{2,k}(n)$ et $T_{3,k}(n)$ est remplacé par $y(t)$. Les tests effectués par les concepteurs de cette technique ont montré que cette substitution n'introduisait pas d'erreurs significatives.

Les valeurs de $r_k(n)$ sont calculées pour tout n=0,…,N-1 et ce, dans chaque segment k=1,…,N_s. Si des composantes sont ajoutées ou supprimées au signal y, le nombre de blocs présents lors du processus d'insertion pourra différer du nombre de blocs présents lors du processus d'extraction. Dans ce cas, les limites du segment k lors de l'insertion seront différentes de celles du segment k lors de l'extraction. Néanmoins, le fait que le calcul de $S_k(n)$ soit réalisé pour tous les décalages circulaires possibles de n dans y_k, assure que la synchronisation entre y et w, nécessaire pour la détection du watermark, sera réalisée pour certaines valeurs n_w de n. Dans ce cas, $r_k(n_w)$ sera très proche de 1. Au contraire, si le signal y n'est pas tatoué ou tatoué avec un autre watermark ou lorsque $n \neq n_w$, le ratio $r_k(n)$ sera approximativement égal à 0.

L'algorithme calcule ensuite le maximum des $r_k(n)$:

$$R_k = \max_{n=0}^{N-1} r_k(n) \qquad (5.26)$$

Puisque le watermark est inséré dans chaque segment du signal original, le facteur de détection final sera évalué par l'équation 5.27 qui suit :

$$R = \max_{n=0}^{N_s-1} R_k \qquad (5.27)$$

Dans le cas idéal, R est égal à 1. Le watermark est cependant supposé identifié si R est supérieur à un certain seuil prédéfini.

5.2.4. Watermarking des médias audio compressés [39]

Toutes les méthodes présentées ci-dessus ont pour objet le tatouage de fichiers audio non-compressés. Si elles se caractérisent par leur robustesse, la lenteur de leur processus d'insertion ne permet cependant pas leur utilisation en ligne.

D'autres méthodes ont été proposées qui opèrent directement sur des fichiers compressés (cf. [33] et [34]). Si, à l'inverse des précédentes, leur processus d'insertion est rapide, leur robustesse laisse cependant à désirer.

Partant des constats précédents, C. Xu, J. Wu et D.D. Feng proposent une approche qui réunit la robustesse du tatouage sur les fichiers non-compressés et la rapidité du tatouage sur les fichiers compressés [39]. A cette fin, ils proposent d'insérer le watermark (copyright ou fingerprint) dans le domaine « partiellement non-compressé ». Cette insertion, qui dépendra du contenu du fichier audio à tatouer, s'opère selon la procédure décrite ci-après (figure 5.4).

Le fichier compressé est, dans une première étape, segmenté en frames[11]. Une fois décompressées, ces frames seront testées par un dispositif de sélection et d'extraction des paramètres et par un modèle psychoacoustique :

- le *dispositif de sélection et d'extraction des paramètres* a pour rôle, d'une part de sélectionner les frames qui seront tatouées, en vérifiant pour chacune d'elles si elle remplit la (les) condition(s) fixée(s) par l'utilisateur, et d'autre part, de calculer certains paramètres nécessaires à l'insertion du watermark,
- le *modèle psychoacoustique* a pour rôle de déterminer le seuil de masquage de chaque frame pour garantir que le tatouage soit inaudible.

[11] Ces frames sont les mêmes que celles issues du processus de compression.

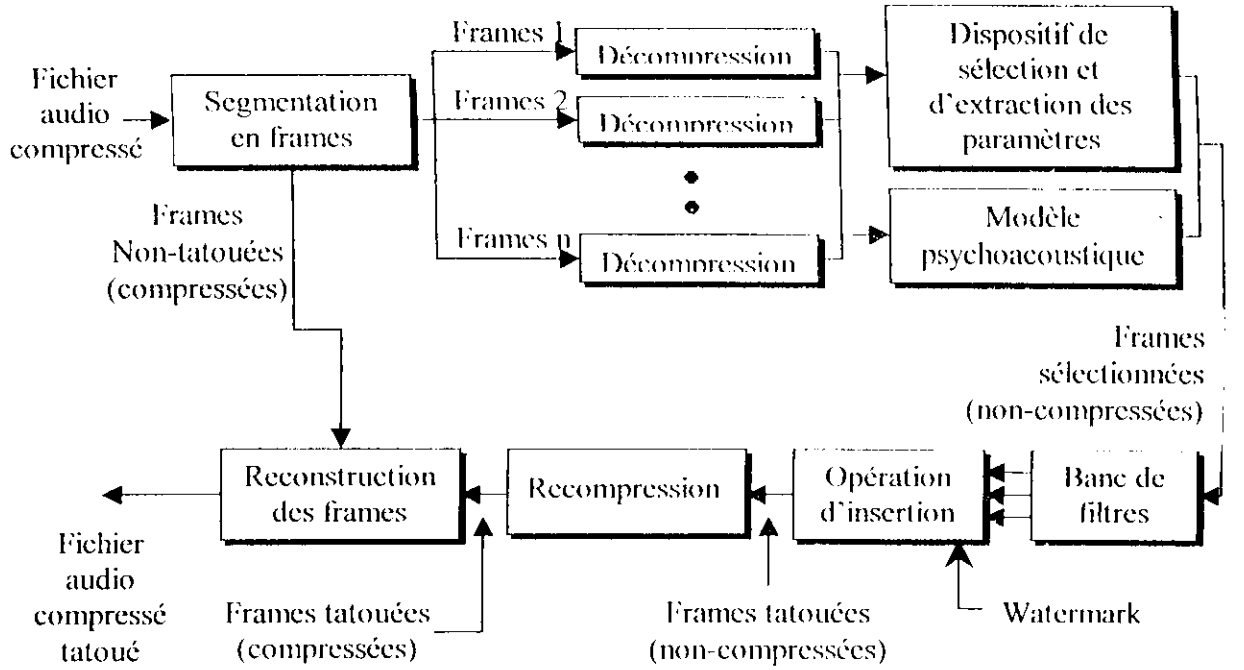


Figure 5.4 : *Schéma bloc du processus d'insertion.*

A. Insertion du watermark

L'algorithme d'insertion de cette méthode utilise la technique du « *Echo hiding* » qui consiste à introduire dans chaque frame sélectionnée des échos (délais) en relation avec le watermark à insérer, ce dernier étant une suite binaire prédéfinie.

D'une manière générale, les algorithmes utilisant l'*Echo hiding* introduisent dans le signal deux échos de durées différentes[12], un pour le codage des « 1 » et un autre pour celui des « 0 ». Xu, Wu et Feng, recourent à cette même technique sauf que, pour en augmenter la robustesse contre les attaques statistiques, ils introduisent des délais différents pour coder les « 1 », ainsi que pour coder les « 0 ».

Dans cette technique, 4 paires de délais différentes sont utilisées. Pour chaque frame sélectionnée et selon les informations fournies par le dispositif de sélection et d'extraction des paramètres, l'algorithme choisira la paire de délais à insérer.

A titre d'exemple, le test des frames et le choix des paires de délais à insérer peut se faire en de la manière suivante :

Si $P_{f<1kHz} \geq 2.\ P_{f>1kHz}$ l'algorithme utilise la paire de délais 1,

Si $P_{f<1kHz} \leq P_{f<1kHz} < 2.P_{f>1kHz}$ l'algorithme utilise la paire de délais 2,

[12] Ces délais sont choisis de manière à ne pas être perceptibles par l'oreille humaine.

Si $P_{f<1kHz} \leq P_{f>1kHz} < 2.P_{f<1kHz}$ l'algorithme utilise la paire de délais 3,

Si $P_{f>1kHz} \geq 2.\ P_{f\leq 1kHz}$ l'algorithme utilise la paire de délais 4.

Au cours de l'opération d'insertion, l'algorithme insère dans chaque frame sélectionnée un bit du watermark. Cette insertion se fait selon le processus suivant :

- l'algorithme décompose la frame sélectionnée en N sous-frames,
- il affecte au bit à insérer un autre code binaire sur N bits[13].

  Ce code, qui diffère d'une paire de délais à une autre, permet de constituer le watermark final (cette technique est appelée « *multiple-bit hopping* »).
- il insère dans chacune des sous-frames un bit du watermark final. Ainsi, la j^e sous-frame de la i^e frame tatouée $S'_{ij}(n)$ peut être exprimée de la manière suivante :

$$S'_{ij}(n) = S_{ij}(n) + A_{ij}.S_{ij}(n-\varphi_{ij}) \qquad (5.28)$$
$$S_{ij}(k) = 0 \quad \text{si} \quad k < 0$$

où : $S_{ij}(n)$: sous-frame originale j de la frame i,

A_{ij} : facteur permettant de contrôler l'amplitude du délais,

ϕ_{ij} : délai correspondant au bit « 0 » ou au bit « 1 ».

Une fois tatouées, les frames sélectionnées sont recompressées puis combinées avec celles qui ne l'ont pas été pour former le fichier tatoué compressé.

B. Extraction du watermark

L'extraction du watermark ne nécessite pas le fichier audio original. En effectuant les mêmes opérations que lors du processus d'insertion, l'algorithme d'extraction commence par décomposer le signal tatoué en frames pour déterminer celles d'entre elles qui ont été tatouées. L'algorithme procédera alors à l'extraction du watermark.

Le processus d'extraction du watermark de la méthode présentée par Xu, Wu et Feng peut être schématisé comme dans la figure 5.5 ci-après.

[13] Par exemple, le bit « 1 » sera codé par la séquence {101010} et le bit « 0 » codé par la séquence {110011} (N=6).

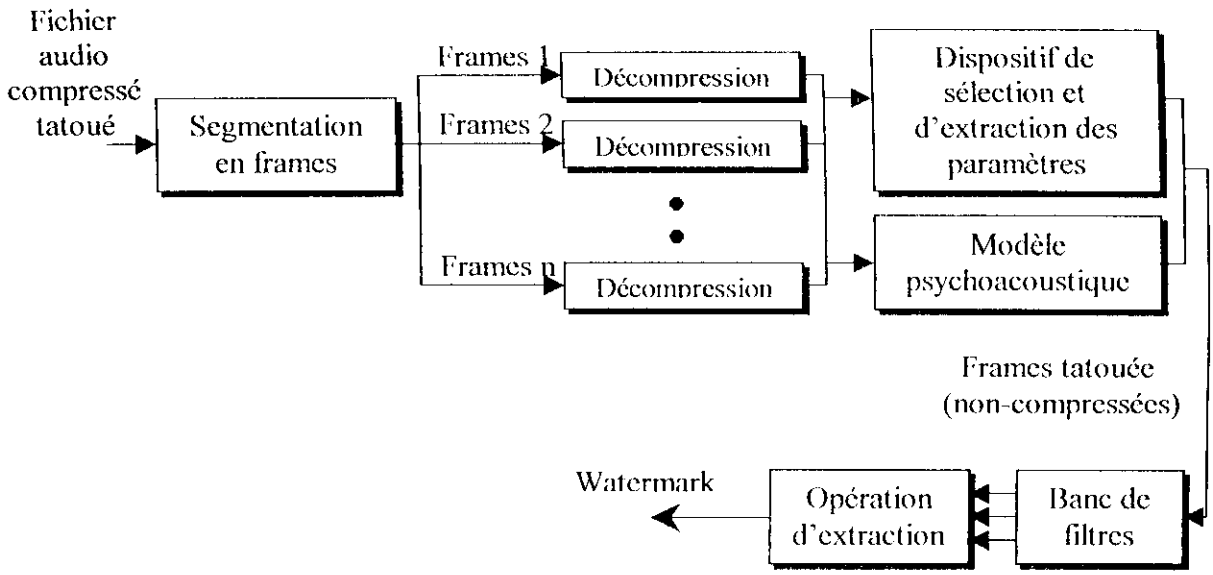


Figure 5.5 : *Schéma bloc du processus d'extraction.*

L'opération « extraction du watermark », phase finale du processus d'extraction décrit ci-dessus, consiste à retrouver les durées qui séparent les échos du signal original. L'algorithme a recours à cet effet au module de la fonction d'autocorrélation du cepstre du signal tatoué.

L'analyse cepstrale, qui permet de convertir une convolution en une somme, permettra, lorsqu'elle est appliquée au signal tatoué, de séparer les échos et le signal original. Après avoir calculé la fonction d'autocorrélation du cepstre, l'algorithme détectera les bits insérés qui correspondent aux pics d'énergie de cette fonction.

La détection du watermark dans cette méthode peut être récapitulée comme suit :

1. L'algorithme commence par calculer la FFT de chaque segment $s_i(n)$:

$$S_i(e^{j\Omega}) = FFT(s_i(n)) \qquad (5.29)$$

2. Il calcule ensuite le logarithme complexe de chaque $S_i(e^{j\Omega})$:

$$\log\{S_i(e^{j\Omega})\} = \log\{F(s_i(n))\} \qquad (5.30)$$

3. Il applique sur chaque logarithme complexe une FFT inverse (cepstre) :

$$\bar{s}_i(n) = IFFT\left[\log\{FFT(si(n))\}\right] \qquad (5.31)$$

4. Il calcule la fonction d'autocorrélation du cepstre :

$$R_{ss}(n) = \sum_{m=-\infty}^{\infty} \bar{s}(n+m).\bar{s}(m) \qquad (5.32)$$

5. Il détermine les délais φ_{ij} correspondants aux pics d'énergie de $R_{ss}(n)$

6. Il détermine le code correspondant à chaque délai φ_{ij} et en déduit le bit du watermak inséré dans le segment traité.

En répétant ces opérations pour tous les segments du signal audio, l'algorithme pourra reconstituer l'intégralité du watermark inséré.

5.3. WATERMARKING AUDIO DANS LE DOMAINE FRÉQUENTIEL

Les techniques de watermarking entrant dans cette catégorie, généralement plus complexes que celles qui procèdent dans le domaine temporel, se basent sur les caractéristiques fréquentielles du signal hôte pour faire en sorte que les watermarks insérés soient inaudibles, même dans les cas où leur amplitude est élevée. Elles s'appuient, pour cela, sur des modèles psychoacoustiques.

Dans [29] est présentée une procédure de watermarking permettant d'intégrer une protection de droits d'auteur dans des données audio-numériques, par modification directe des échantillons audio. Cette procédure exploite directement les masquages psychoacoustiques (fréquentiel et temporel) pour garantir que le filigrane numérique inséré soit inaudible. A cette fin, l'algorithme prend en considération le masquage fréquentiel en utilisant le *modèle psychoacoustique 1* défini dans la norme *ISO-MPEG Audio* [40] et fait une approximation du masquage temporel en utilisant l'enveloppe du signal audio (voir figure 5.6).

A. Insertion du watermark

Le signal audio original de longueur N est segmenté en blocs $s_i(k)$ de 512 échantillons, avec i=0,1,...,[N/512] – 1, et k=0,1,...,511.

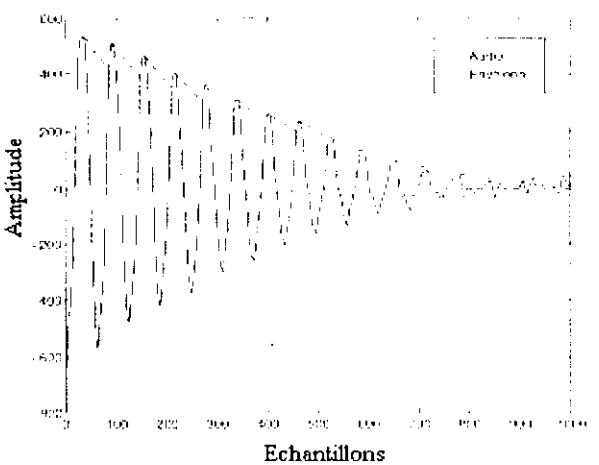


Figure 5.6 : *Enveloppe d'un signal audio.*

Cette procédure de tatouage consiste à insérer le même watermark dans chaque segment du signal audio en utilisant une clé y comme le montre la figure 5.7.

Pour chaque segment $s_i(k)$, l'algorithme applique les opérations suivantes :

1. il calcule le spectre de puissance $S_i(k)$ de chaque segment $s_i(k)$;

2. calcule les seuils de masquage en fréquence $M_i(k)$ du spectre $S_i(k)$;

3. modèle l'énergie des clés $y_i(k)$ de manière qu'elles soient au dessous des seuils $M_i(k)$ pour obtenir la signature de l'auteur $P_i(k) = Y_i(k).M_i(k)$;

4. calcule la transformée de Fourier inverse de la signature de l'auteur : $p_i(k) = \text{IFFT}(P_i(k))$;

5. calcule les seuils de masquage temporels $t_i(k)$ du segment $s_i(k)$;

6. modèle $p_i(k)$ de manière qu'elle soit au dessous des seuils de masquage temporel pour obtenir $w_i(k)$ par : $w_i(k) = t_i(k).p_i(k)$;

7. crée le bloc tatoué $s'_i(k) = s_i(k) + w_i(k)$.

Le signal tatoué résulte de la concaténation de tous les blocs $s'_i(k)$.

La signature de l'auteur y_i correspondante au bloc i est calculée à partir d'une clé privée x_1 et d'une autre clé x_2 dépendante du signal original. La clé x_2 est obtenue en appliquant une fonction de hachage à sens unique sur un paramètre du signal d'origine (par exemple, sur le seuil de masquage de chaque segment).

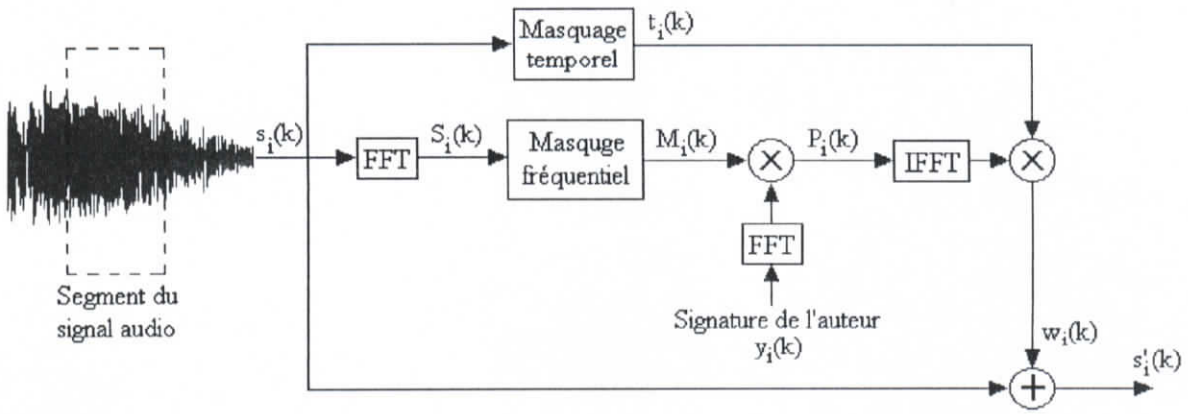


Figure 5.7: *Diagramme d'insertion du watermark.*

Le fait d'avoir recours aux deux masquages, temporel et fréquentiel, permet de contrôler le watermark dans les deux domaines. Le seul masquage fréquentiel ne garantit pas l'inaudibilité du tatouage. En effet, lorsque l'énergie du signal n'occupe pas toute la largeur de la fenêtre d'analyse dans le domaine fréquentiel, le watermark ne sera pas masqué à l'extérieur de la fenêtre correspondante dans le domaine temporel et les distorsions

engendrées seraient audibles. C'est pour cette raison qu'il est fait recours au masquage temporel qui garantit aux régions "calmes" de ne pas être perturbées par le watermark.

B. Extraction du watermark

Lors de l'extraction, le watermark devrait pouvoir être identifié même si il est ciblé par des attaques visant à l'altérer ou le supprimer. Le processus d'extraction du watermark, nécessite la connaissance du signal original et s'opère selon la démarche décrite ci-après :

- **En cas de synchronisation entre les signaux *s(i)* et *r(i)***

Soit $r(i)$, avec $0 \le i \le N-1$, le signal à analyser lors de l'extraction. Ce signal peut s'écrire sous la forme :

$$r(i) = s(i) + d(i), \text{ avec } 0 \le i \le N-1,$$

le signal $d(i)$ étant soit du bruit[14], soit du bruit plus le watermark. Le processus d'extraction suppose également que le propriétaire du fichier, ou l'arbitre, disposent des deux clés x_1 et x_2 nécessaires à la génération de la séquence pseudo-aléatoire y, ou qu'ils puissent les reconstituer.

Ces conditions étant réunies, les tests destinés à la détection du watermark vont reposer sur les hypothèses suivantes :

$H_1 : t(i) = r(i) - s(i) = n(i)$, $0 \le i \le N-1$ (absence de watermark),

$H_2 : t(i) = r(i) - s(i) = w'(i) + n(i)$, $0 \le i \le N-1$ (présence d'un watermark),

où $w'(i)$ est le watermark éventuellement modifié et $n(i)$ le bruit.

La comparaison de la similarité (équation 5.33) entre le signal extrait $t(i)$ et le watermark original $w(i)$ avec un seuil T préalablement fixé, permet de déterminer laquelle des deux hypothèses est la plus probable.

$$Sim(t,w) = \frac{\sum_{j=0}^{N-1} t(j).w(j)}{\sum_{j=0}^{N-1} w(j).w(j)} \qquad (5.33)$$

Des tests ont montré l'efficacité de cette équation pour la détection du watermark et qu'un haut niveau de performance était obtenu avec un seuil T=0.15.

Le watermark est supposé identifié si la similarité est supérieure ou égale au seuil T.

[14] Il est supposé que le bruit est blanc, gaussien et de moyenne nulle.

- **En cas de non-synchronisation entre les signaux *s(i)* et *r(i)***

Dans le cas où le signal à analyser *r(i)* n'est pas synchronisé avec le signal original *s(i)*, il s'écrira sous la forme :

$$r(i)=s(i+\tau)+d(i),\ 0\le i\le N-1,$$

où *d(i)* est tel qu'il a été défini plus haut, et τ un décalage inconnu (τ n'étant pas nécessairement un nombre entier).

Il suffit ensuite de calculer le ratio défini par l'équation 5.34, et le comparer avec *T*.

$$Sim(t,w)=\frac{\max_{\tau}\exp(-\sum_{i=0}^{N-1}(r(i)-(s(i+\tau)+w(i+\tau))^2)}{\max_{\tau}\exp(-\sum_{i=0}^{N-1}(r(i)-s(i+\tau))^2)} \qquad (5.34)$$

Dans ce cas également, le watermark est considéré comme identifié dès lors que la similarité est supérieure ou égale au seuil *T*.

De nombreuses autres méthodes de tatouage opérant dans le domaine fréquentiel ont été développées en se basant sur la technique de l'étalement du spectre. Une méthode de ce type est étudiée et développée dans le présent travail (chapitre 6). Elle permet l'insertion de watermarks dans les fichiers audio non-compressés. Outre l'étalement du spectre qui permet d'augmenter la robustesse des watermarks insérés, cette méthode recourt à la modélisation psychoacoustique de l'audition humaine pour faire en sorte que ces watermarks ne génèrent des distorsions audibles.

TABLEAU COMPARATIF DES DIFFERENTES TECHNIQUES DE TATOUAGE EXPOSEES

METHODE	DOMAINE DE DISSIMULATION	TYPE DE WATERMARK	FACTEUR D'AMPLI-FICATION	TYPE D'INSERTION	TYPE DE DONNEES DISSIMULEES	EXTRACTION	AVANTAGES	BANDE DE DISSIMULATION
WATERMARKING AUDIO BASE SUR LA DISTRIBUTION DE L'ENERGIE DANS LE TEMPS	Temporel	Additif	Oui	Blocs de longueur L	Copyright	Ne nécessite pas la connaissance de l'original	Grande qualité sonore du fichier tatoué	Tout le spectre des fréquences
WATERMARKING AUDIO BASE SUR LES PERIODES DE SILENCE	Temporel	Ajout d'échantillons	Non	Dépend des périodes de silence	Copyright	Ne nécessite pas la connaissance de l'original	Simplicité d'implé-mentation	Tout le spectre des fréquences
WATERMARKING AUDIO ROBUSTE DANS LE DOMAINE TEMPOREL	Temporel	Additif	Oui	Blocs de longueur L	Copyright	Ne nécessite pas la connaissance de l'original	Grande robustesse	Basses fréquences
WATERMARKING DES MEDIAS AUDIO COMPRESSES	Temporel partiellement non-compressé	Additif	Oui	Sur les frames MPEG	Tous types de données numériques	Ne nécessite pas la connaissance de l'original	Utilisation en ligne	Tout le spectre des fréquences
WATERMARKING AUDIO DANS LE DOMAINE FREQUENTIEL	Spectral	Additif	Non	Blocs de 512 échantillons	Copyright	Nécessite l'original	Résout le problème de l'impasse	Tout le spectre des fréquences

Chapitre 6

ÉTUDE ET IMPLÉMENTATION D'UNE MÉTHODE DE WATERMARKING AUDIO

- **Présentation de la méthode**
- **Modèle psychoacoustique utilisé**
- **Génération et insertion du watermark**
- **Détection du watermark**
- **Mise en œuvre de la méthode**
- **Expérimentations**
- **Résultats et perspectives**

ÉTUDE ET IMPLÉMENTATION D'UNE MÉTHODE DE WATERMARKING AUDIO

6.1. PRÉSENTATION DE LA MÉTHODE [41]

La méthode de watermarking audio étudiée et implémentée opère dans le domaine fréquentiel (figure 6.1) et permet l'insertion de watermarks dans les fichiers audio non-compressés. Elle utilise :

- d'une part, la technique de l'étalement de spectre afin de générer des watermarks résistants aux différentes attaques susceptibles de les détériorer,
- et d'autre part, un modèle psychoacoustique afin que les watermarks insérés n'introduisent pas de distorsions audibles.

Pour ce qui est du processus d'extraction du watermark, cette méthode ne requiert pas la connaissance du fichier original.

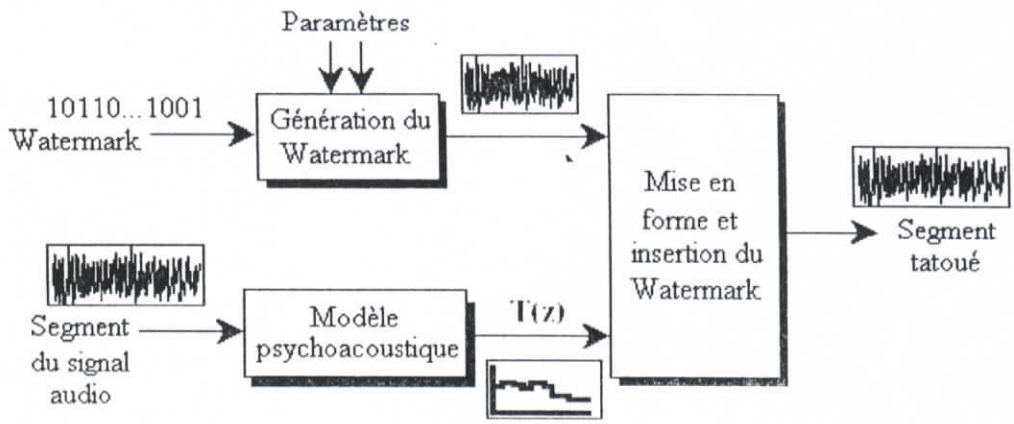


Figure 6.1 : *Schéma général de génération et d'insertion d'un watermark*

L'algorithme d'insertion procède, dans une première étape, à la division du signal audio original en segments de tailles égales puis transpose chaque segment du domaine temporel au domaine fréquentiel, en utilisant une transformée de Fourier rapide (FFT). Un modèle psychoacoustique est enfin appliqué à chaque segment du signal pour trouver son seuil de masquage. La connaissance de ce seuil permet d'insérer le watermark sans altérer la qualité sonore du signal hôte.

Pour garantir la robustesse du watermark, sa génération est paramétrée par une clé (séquence pseudo-aléatoire). De même, des codes correcteurs sont utilisés (code répéteur et Interleaving) pour réduire le nombre d'erreurs qui peuvent éventuellement affecter le watermark inséré, lors du processus d'extraction.

6.2. LE MODÈLE PSYCHOACOUSTIQUE UTILISÉ

Le modèle psychoacoustique utilisé dans la méthode développée est un algorithme qui approche le mécanisme de l'audition humaine (figure 6.2). Pour chaque segment du signal audio traité, il estime le seuil de masquage nécessaire au modelage du watermark à insérer de sorte à rendre celui-ci quasiment imperceptible et préserver ainsi la qualité sonore du signal original.

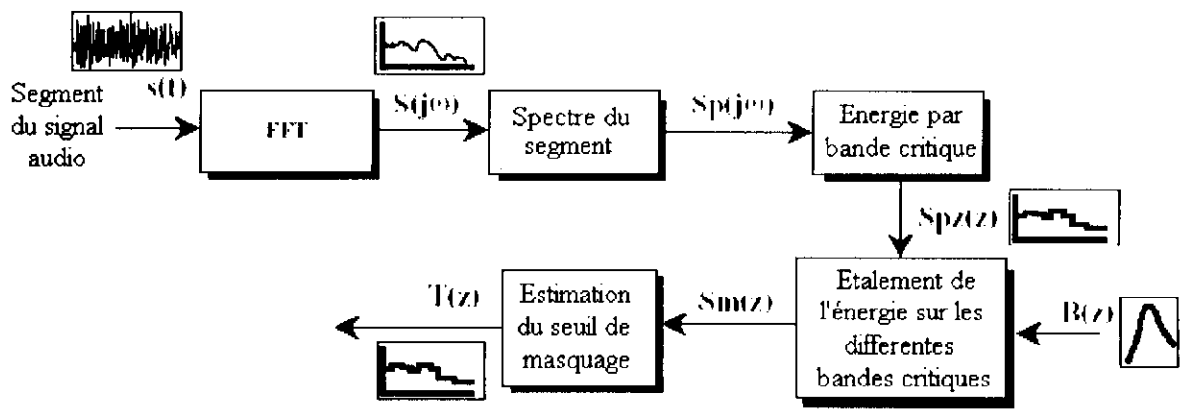
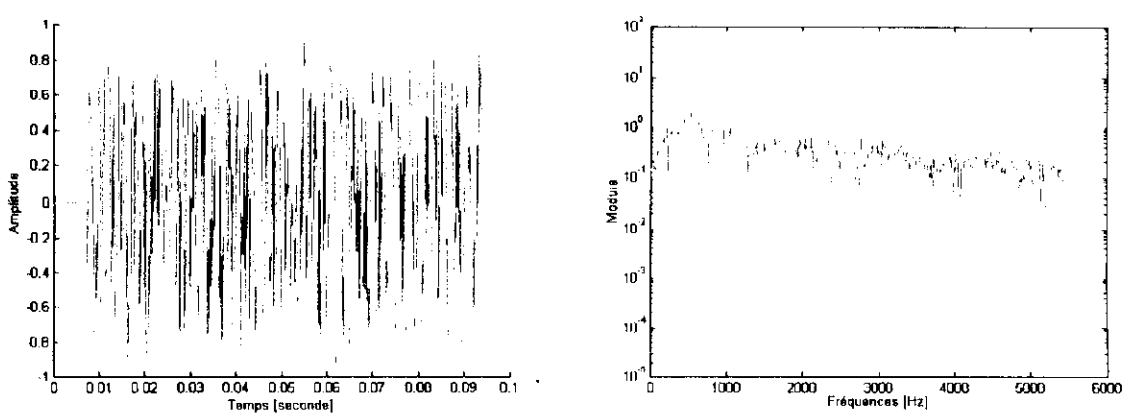


Figure 6.2 : *Modèle psychoacoustique utilisé.*

6.2.1. Spectre d'énergie

Le modèle psychoacoustique sera appliqué pour chaque segment du signal original (figure 6.3.a.). Une étape préliminaire à l'analyse consistera à transférer le segment audio du domaine temporel vers le domaine fréquentiel en utilisant une FFT (figure 6.3.b.).



(a) **(b)**

Figure 6.3 : (a) *Segment sw(t) du signal original*, (b) *spectre du signal sw(t).*

Dans le domaine fréquentiel, la première opération consistera à calculer le l'énergie de chaque composante spectrale du signal. Pour ce calcul, l'expression suivante est utilisée :

$$Sp(j\omega) = \mathrm{Re}\{Sw(j\omega)\}^2 + \mathrm{Im}\{Sw(j\omega)\}^2 = |Sw(j\omega)|^2 \qquad (6.1)$$

L'algorithme calculera ensuite l'énergie par bande critique *Sp(z)* qui est donnée par :

$$Spz(z) = \sum_{\omega=LBZ}^{HBZ} Sp(j\omega) \qquad (6.2)$$

où : z =1,2,…,nombre total de bandes critiques.

LBZ : la plus basse fréquence de la bande critique z

HBZ : la plus haute fréquence de la bande critique z.

Le spectre de l'énergie *Sp(jω)* et l'énergie par bande critique *Spz(z)* sont les éléments de base à partir desquels sera opérée l'analyse dans le domaine fréquentiel. Ces éléments (figure 6.4) permettront le calcul du seuil de masquage.

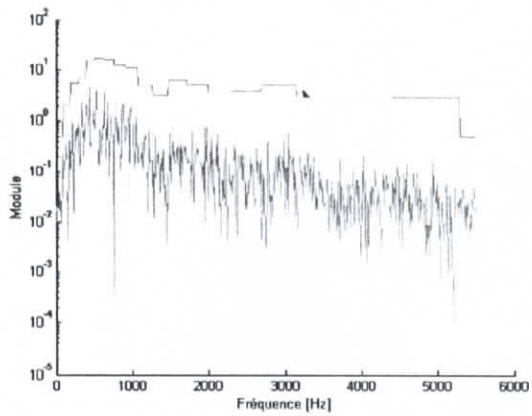


Figure 6.4 : *Spectre de l'énergie Sp(jω) (en bleu), et l'énergie par bande critique Spz(z) (en rouge).*

6.2.2. Modélisation de la fonction d'étalement de la membrane basilaire

Au niveau du système auditif humain, l'interprétation en fréquence des ondes sonores est effectuée par la membrane basilaire. Cette dernière vibre à différents endroit selon la fréquence du son qu'elle perçoit. Cette vibration ne se fait pas en un seul point; elle s'étale sur une certaine région de la membrane basilaire.

Dans la mesure où l'algorithme a pour objectif de se rapprocher le plus possible des mécanismes de l'audition humaine, cet étalement sera reconstitué à travers le modèle mathématique suivant : (voir figure 6.5.a)

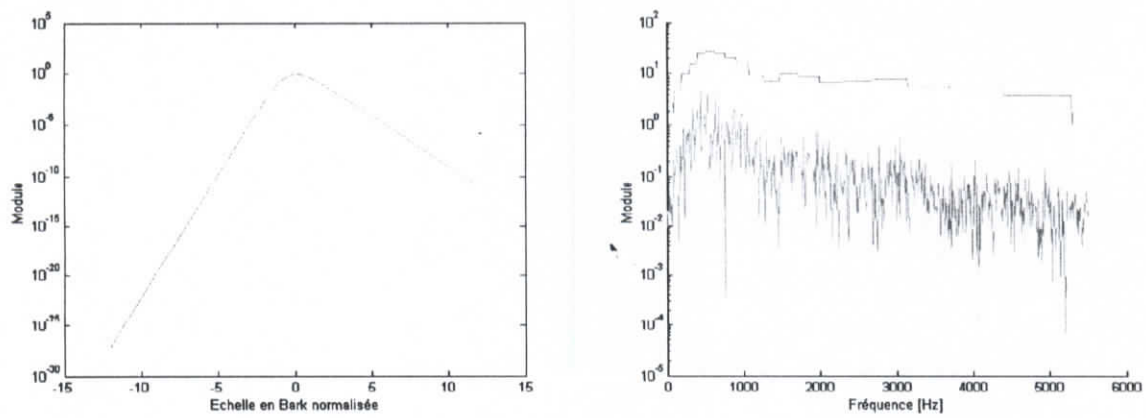
$$B(z) = 15.91 + 7.5(z + 0.474) - 17.5\sqrt{1 + (z + 0.474)^2} \qquad (6.3)$$

où : $B(z)$ est la fonction d'étalement,

z représente l'échelle en bark normalisée.

La fonction d'étalement permet de déterminer la quantité d'énergie avec laquelle influe une bande critique sur les bandes l'avoisinant. Elle permettra ainsi de calculer l'énergie de chaque bande critique en prenant en compte celles des bandes qui lui sont voisines. L'énergie étalée $Sm(z)$ (figure 6.5.b) est obtenue par l'équation suivante :

$$Sm(z) = Spz(z) * B(z) \qquad (6.4)$$



(a) **(b)**

Figure 6.5 : (a) *Fonction d'étalement B(z),* (b) *l'énergie étalée par bande critique Sm(z).*

Un calcul plus rigoureux de $Sm(z)$ devrait prendre en compte chaque composante de chaque bande critique mais nécessiterait cependant beaucoup plus de temps que l'utilisation de l'équation 6.4. Celle-ci est préférée car elle offre une bonne approximation (résultats presque identiques) sans prendre autant de temps.

6.2.3. Evaluation du seuil de masquage

a. Coefficients de masquage

Pour l'évaluation du seuil de masquage, deux coefficients seront utilisés :

- le premier, quand un son tonal masque un son non tonal (bruit); il est évalué à (14.5+*z*) dB au dessous du niveau de *Sm(z)*. Dans ce cas, z représente la bande critique à laquelle appartient le son masquant (ici tonal).
- le second, lorsqu'un son non tonal masque un son tonal. Sa valeur, indépendante de celle de *z*, est de 5.5 dB au dessous de *Sm(z)*.

Aussi, la détermination du caractère de chaque segment constitue-t-elle un préalable.

b. Mesure d'aplatissement spectral et facteur de tonalité

Pour déterminer le caractère de chaque segment du signal analysé (segment à comportement plus tonal que non tonal ou le contraire), l'algorithme doit au préalable mesurer l'aplatissement spectral (*SFM* : *Spectral Flatness Measure*). Ce dernier est déterminé par le rapport des moyennes géométrique et arithmétique de *Sp(z)* :

$$SFM_{dB} = 10\log_{10}\left\{\frac{\prod_{z=1}^{Zt} Spz(z)}{\frac{1}{Zt}\sum_{z=1}^{Zt} Spz(z)}\right\}^{\frac{1}{Zt}} \qquad (6.5)$$

où *Zt* représente le nombre total des bandes critiques occupées par le signal.

Le résultat obtenu à l'issue de ces calculs va permettre de déterminer le facteur de tonalité α qui renseignera sur le caractère du segment. Ce facteur est calculé comme suit :

$$\alpha = \min\left(\frac{SFM_{dB}}{SFM_{dB\max}}, 1\right) \qquad (6.6)$$

avec SFM_{dB} = -60 dB.

Si le facteur de tonalité α se rapproche de 1, le segment sera considéré comme tonal. S'il se rapproche de zéro, le segment sera considéré comme non tonal.

Cette valeur α va être utilisée par l'algorithme pour corriger la valeur de *Sm(z)* en réduisant sa valeur initiale de *O(z)*. Celui-ci se calcule comme suit :

$$O(z) = \alpha(14.5 + z) + (1 - \alpha)5.5 \qquad (6.7)$$

Ces différents calculs vont permettre de déterminer le seuil de masquage *Traw* (figure 6.6.a) par application de l'équation suivante :

$$Traw = 10^{\left(\log_{10}(Sm(z)) - \frac{O(z)}{10}\right)} \qquad (6.8)$$

c. Normalisation du seuil

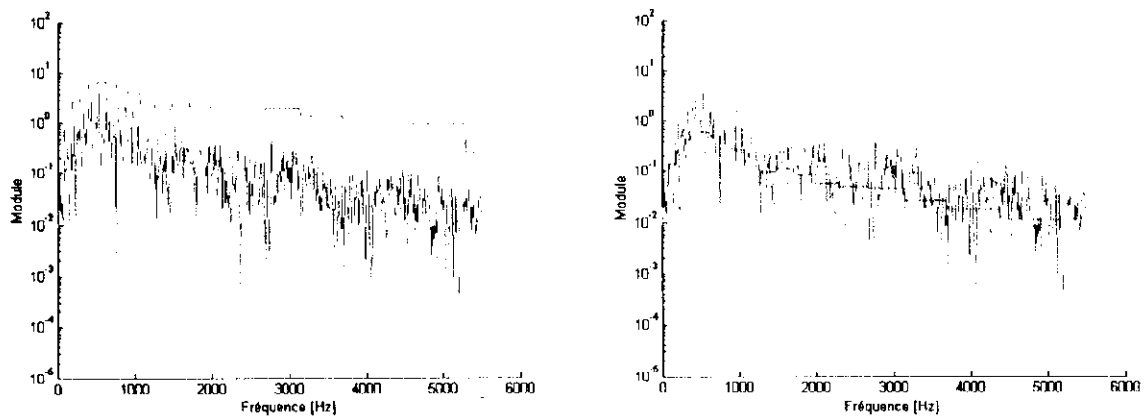
Comme la largeur des bandes critiques augmente avec l'augmentation de la fréquence (mesurée en Hz), une bande critique aux hautes fréquences pourra avoir plus de composantes

qu'une bande en basses fréquences. La comparaison entre les seuils des différentes bandes critiques ne sera, ainsi, pas rigoureuse; une normalisation de ces seuils est, en conséquence, indispensable (figure 6.6.b) :

$$Tnorm(z) = \frac{Traw}{P_z} \qquad (6.9)$$

avec : P_z : nombre de composantes dans chaque bande critique z.

z=1,...,nombre de bandes critiques Zt.



(a) **(b)**

Figure 6.6 : (a) *Seuil de masquage original Traw(z)*
(b) *Seuil de masquage normalisé Tnorm(z).*

d. Le seuil final de masquage

Après la normalisation, la dernière étape consiste à tenir compte du seuil absolu d'audition. Les recherches effectuées dans le domaine de l'acoustique physiologique révèlent d'une part que ce seuil varie en fonction de la fréquence, et d'autre part, que l'oreille humaine est la plus sensible lorsque cette fréquence se situe entre 2500 et 4500 Hz (cf. chapitre 2).

Le modèle psychoacoustique utilise une approximation consistant à considérer que le seuil absolu (Th) est constant et qu'il correspond au minimum du seuil absolu réel (aux environs de 4000Hz) :

$$Th = \max\left(\left|Pp(j\omega)\right|\right) \qquad (6.10)$$

avec : $Pp(j\omega)$ spectre d'énergie de p(t).

$p(t)=sin(2\pi 4000.t)$.

Le seuil de masquage final *(T)* recherché sera calculé par l'équation ci-après :

$$T(z) = \max(Tnorm(z), Th) \qquad (6.11)$$

Ce seuil de masquage permettra, après la génération du signal watermark, de modeler ce dernier de sorte à ce qu'il soit inaudible.

6.3. GÉNÉRATION ET INSERTION DU WATERMARK

6.3.1. Génération du watermark

La génération du filigrane consiste à construire une signal watermark à partir d'une séquence initiale qui contient des informations sur le distributeur ou le propriétaire du média audio à tatouer. Une fois dans le signal hôte, ce watermark devra satisfaire deux conditions :

- résister non seulement aux différentes tentatives de dégradations dont il pourrait être l'objet (tentatives illicites visant à l'altérer ou le supprimer), mais aussi aux bruits et aux interférences dans les canaux de transmission.
- rester inaudible. Pour cela, le watermark devra être inséré dans les régions spectrales fixées par le modèle psychoacoustique. Il faudra donc que le signal watermark occupe la même bande de fréquences que celle du signal original.

Une technique utilisée dans le domaine des communications permet de satisfaire ces deux conditions : *l'étalement de spectre*. Elle permettra de générer le signal watermark à partir de la séquence watermark, et ceci en l'étalant avec une séquence pseudo-aléatoire. Pour augmenter les performances de détection, des codes correcteurs seront utilisés (code répéteur et interleaving).

La génération du watermark se fait de la manière suivante :

Soit $\{w\}$ une séquence binaire bipolaire (1 ou –1) de 16 bits choisie par le propriétaire :

$$\{w\} = \{1\ \ 1\ -1\ \ 1\ -1\ -1\ \ 1\ -1\ \ 1\ \ 1\ -1\ \ 1\ \ 1\ \ 1\ -1\ -1\ -1\}$$

Chaque symbole de $\{w\}$ sera répété m fois (ici m=3) pour obtenir $\{w_R\}$:

$$\{w_R\} = \begin{matrix} \{1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1\} \end{matrix}$$

Pour prévenir contre d'éventuelles erreurs, qui généralement frappent non pas un mais des blocs de bits successifs, la technique d'entrelacement (Interleaving) sera utilisée (figure 6.7). Pour son application, une matrice de dimension (LxC) sera utilisée. La séquence à coder $\{w_R\}$ sera écrite dans cette matrice suivant les C colonnes, puis retranscrite suivant les L lignes pour donner $\{w_T\}$. Pour poursuivre l'exemple entamé, on prendra L=10 et C=5.

X_1	X_{11}	X_{21}	X_{31}	X_{41}
X_2	X_{12}	X_{22}	X_{32}	X_{42}
X_3	X_{13}	X_{23}	X_{33}	X_{43}
X_4	X_{14}	X_{24}	X_{34}	X_{44}
X_5	X_{15}	X_{25}	X_{35}	X_{45}
X_6	X_{16}	X_{26}	X_{36}	X_{46}
X_7	X_{17}	X_{27}	X_{37}	X_{47}
X_8	X_{18}	X_{28}	X_{38}	X_{48}
X_9	X_{19}	X_{29}	X_{39}	X_{49}
X_{10}	X_{20}	X_{30}	X_{40}	X_{50}

Figure 6.7 : *Matrice d'interleaving (10×5).*

La séquence résultant de ce code est $\{w_T\}$:

$\{w_T\}=$ {1 1 1 -1 1 1 1 -1 -1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1 1 -1 1 1 -1
1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 1 1 1 1 1 1 1 1 1}

Dans le cas où il reste des cases vides dans la matrices, celles-ci seront complétées par des 1.

La dernière étape consiste à étaler le watermark en utilisant une *PN sequence* (séquence pseudo-aléatoire) qui représente la *clé* (notée *Label 1*) du watermarking, sans laquelle l'extraction est impossible. La séquence aléatoire doit être dé-corrélée avec $\{w_T\}$.

Si $c(t)$ représente la *PN sequence* alors le watermark étalé sera obtenu pas la formule suivante :

$$w(t)=w_T(t) \times c(t) \qquad (6.12)$$

C'est ce watermark qui sera inséré dans chaque segment du signal à tatouer.

6.3.2. Insertion du watermark

Une fois les seuils de masquage déterminés, et le signal watermark généré, l'algorithme procède à l'insertion de celui-ci. Cette insertion est basée sur un principe simple : la qualité sonore d'un signal ne sera en rien altérée si les composantes dont la puissance est inférieur au seuil de masquage sont remplacées par d'autres composantes, elles aussi de puissance inférieure au seuil de masquage.

Le signal hôte est divisé en plusieurs segments, dans chacun d'eux on injectera le même watermark ce qui renforcera la robustesse du filigrane. La méthode d'insertion est la suivante :

1) Supprimer les composantes inaudibles du signal original en utilisant la relation suivante :

$$Swnew_i(j\omega) = \begin{cases} Sw_i(j\omega) & Sp_i(j\omega) \geq T(z) \\ 0 & Sp_i(j\omega) < T(z) \end{cases} \qquad (6.13)$$

avec : i = 1,2,...,nombre de composantes,

z et ω dépendent de la i^e composante.

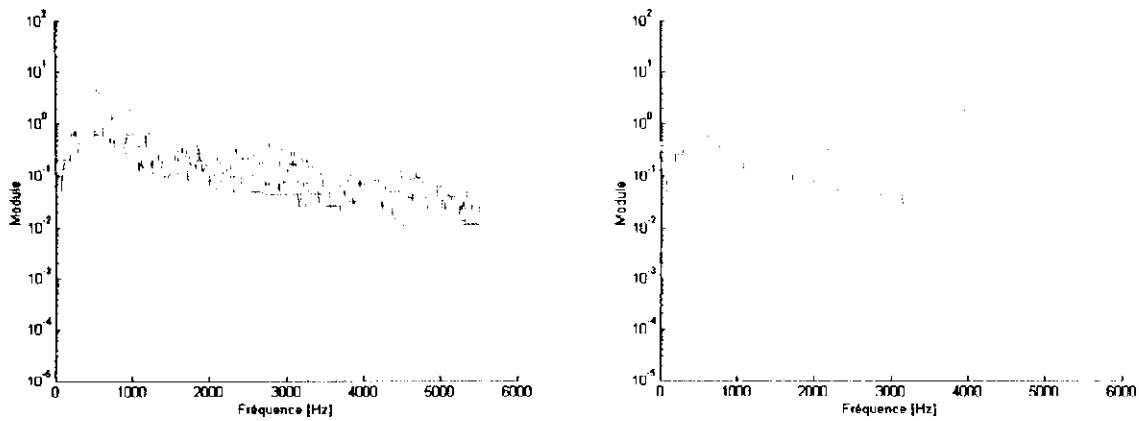
2) Supprimer les composantes inutiles du watermark en utilisant l'équation suivante :

$$Wnew_i(j\omega) = \begin{cases} 0 & Sp_i(j\omega) \geq T(z) \\ W_i & Sp_i(j\omega) < T(z) \end{cases} \qquad (6.14)$$

avec : i = 1,2,...,nombre de composantes,

z et ω dépendent de la i^e composante.

La figure 6.8 qui suit montre les composantes du signal original dont la puissance est au dessus du seuil de masquage (figure 6.8.a), et les composantes du signal watermark qui seront insérées (figure 6.8.b).



(a) **(b)**

Figure 6.8 : (a) *Composantes audibles du signal original Swnew(jω)*

(b) *Composantes du signal watermark à insérer Wnew(jω).*

Afin que le watermark à insérer soit inaudible, il est nécessaire de faire baisser la puissance de ses composantes sélectionnées *Wnew(jω)* au dessous du seuil *T(jω)*. Elles seront pour cela multipliées par un facteur F_z. Ce facteur est calculé en utilisant la formule suivante :

$$F_z = A\frac{\sqrt{T(j\omega)}}{\max\left(\left|Wnew(j\omega)\right|\right)} \qquad (6.15)$$

où : $z = 1,2,\ldots,Zt$,

ω allant de *LBZ* jusqu'à *HBZ* pour chaque bande critique z,

A : Facteur d'amplification compris entre 0 et 1. Ce facteur a pour rôle de contrôler le module du watermark, et d'assurer que ce dernier soit bien inaudible.

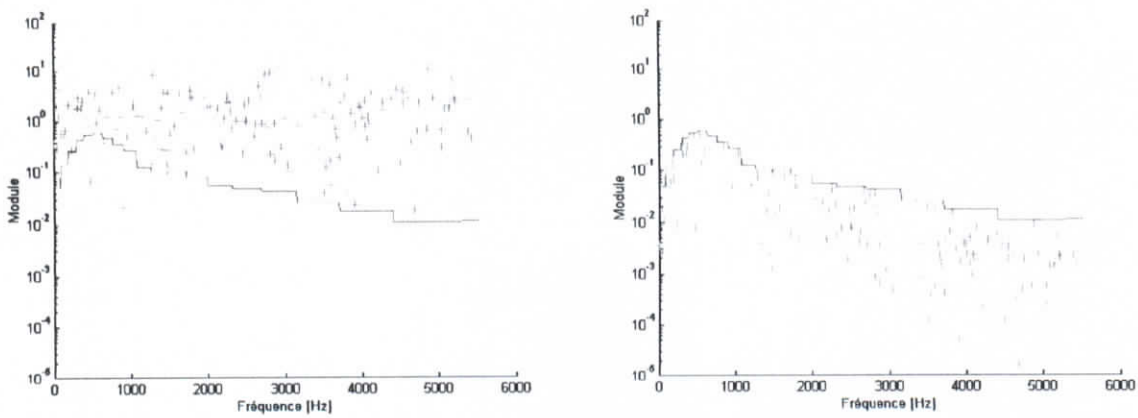
Chaque composante du watermark sera multiplié par le facteur F_z lui correspondant. L'expression final du watermark à insérer est donnée par :

$$Wfinal(j\omega) = F_z \times Wnew(j\omega) \qquad (6.16)$$

avec : $z = 1,2,\ldots,Zt$,

ω allant de *LBZ* jusqu'à *HBZ* pour chaque bande critique z.

La figure 6.9 qui suit montre le spectre du watermak avant et après son modelage (respectivement figure 6.9.a et figure 6.9.b). Il est clair qu'après cette opération, le watermark se trouve bien au dessous du seuil de masquage, il est par conséquent inaudible.



(a) **(b)**

Figure 6.9 : (a) *Signal watermark non modelé Wnew(jω)*

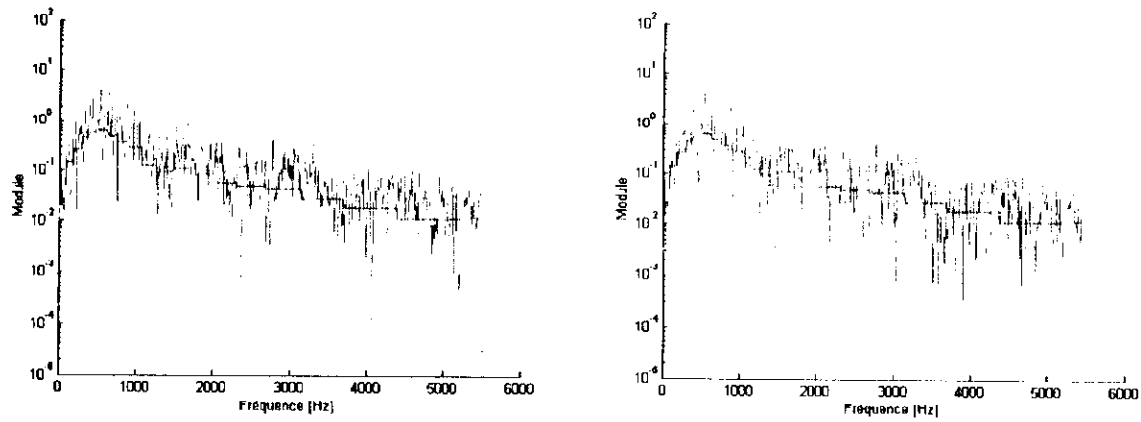
(b) *Signal watermark modelé Wfinal(jω) avec A =0.9.*

6.3.3 Génération du signal tatoué

Le spectre du segment tatoué $OUT(j\omega)$ sera obtenu en additionnant $Swnew(j\omega)$ et $Wfinal(j\omega)$:

$$OUT(j\omega) = Swnew(j\omega) + Wfinal(j\omega) \qquad (6.17)$$

La figure 6.10 qui suit montre le spectre du segment original (figure 6.10.a), et celui du segment tatoué (figure 6.10.b). Les composantes dont la puissance est supérieure au seuil de masquage sont restées inchangé. Seules les composantes inaudibles ont été remplacées.



(a) **(b)**

Figure 6.10 : (a) *Spectre de l'énergie du segment original Sp(jω)*
(b) *Spectre de l'énergie du segment tatoué OUT(jω).*

Il suffit alors d'appliquer à $OUT(j\omega)$ une IFFT (FFT Inverse) pour obtenir la frame tatouée dans le domaine temporel :

$$r(t) = \text{IFFT}\{OUT(j\omega)\} \qquad (6.18)$$

Les frames tatouées seront ensuite concaténées pour constituer le signal tatoué *output(t)*.

6.4. DÉTECTION DU WATERMARK

La particularité dans cette algorithme est que le fichier original (non tatoué) n'est pas nécessaire à la détection du watermark. La procédure de détection a recours au modèle psychoacoustique de l'audition pour ôter les composantes qui ont peu de chance de faire partie du watermark. Pour cela, le seuil est calculé pour chaque segment $s_2(t)$ du signal tatoué et les composantes qui se trouvent au dessus de celui ci sont supprimées. Les composantes restantes se trouvant au dessous du seuil sont ensuite traitées afin d'en extraire la séquence du watermark.

Les étapes du mécanisme d'extraction son explicitées ci dessous.

6.4.1. Synchronisation

La procédure qui suit exige une synchronisation du signal tatoué *output(t)* avec la séquence pseudo aléatoire *c(t)*. Pour cela, l'algorithme d'insertion fournira, en plus de la clé pseudo-aléatoire *c(t)*, une séquence contenant les indices des maximums d'amplitude de chaque segment du signal tatoué *r(t)*. Cette deuxième clé (notée *Label 2*) est nécessaire pour pouvoir détecter le watermark.

L'algorithme d'extraction commence par repérer les maximums d'amplitude de chaque segment du signal tatoué (sur lequel s'opère l'extraction), puis procédera à leur synchronisation avec ceux fournis par le *Label 2*.

6.4.2. Seuils de masquage et signal résiduel

Le modèle psychoacoustique est appliqué au signal tatoué $s_2(t)$, qui a éventuellement subit des transformations (modifications). Le signal résiduel R est le résultat de la suppression des composantes inutiles à la détection.

$$R_i(j\omega) = \begin{cases} Sw_{2i}(j\omega) & Sp_{2i}(j\omega) \le T_2(z) \\ 0 & Sp_{2i}(j\omega) > T_2(j\omega) \end{cases} \qquad (6.19)$$

avec : $i = 1,2,\ldots,Zt$,

z et ω dépendent de la i^e composante.

6.4.3. Normalisation du signal résiduel

Les composantes spectrales dans chaque segment du signal résiduel sont normalisées pour que les maxima de chaque segment soient à un même niveau. Les facteurs de normalisation sont données par :

$$F_z = \frac{1}{\max\left(\left|R(j\omega)\right|\right)} \qquad (6.20)$$

où : $z = 1,2,\ldots,Zt$,

ω allant de LBZ jusqu'à HBZ pour chaque bande critique z.

Les composantes du signal résiduel R sont modifiées en utilisant un facteur approprié pour chaque sous bande critique z.

$$R_{final}(j\omega) = R(j\omega).F_z \qquad (6.21)$$

où : $z = 1,2,\ldots,Zt$,

ω allant de LBZ jusqu'à HBZ pour chaque bande critique z.

Le signal résiduel *r(t)* sera constitué par toutes les composantes résultant d'une IFFT appliquée à $R_{final}(j\omega)$.

6.4.4. Extraction du watermark

Pour retrouver le watermark inséré dans le signal audio on doit effectuer les opérations inverses à celles qui ont été effectuées lors de la génération du watermark. Ces opérations visent à retrouver la séquence initiale du watermark.

Pour trouver la séquence du watermark avant l'étalement de spectre on doit multiplier le signal résiduel *r(t)* par la même séquence pseudo aléatoire *c(t)* qui a été utilisée dans la génération du watermark.

$$g(t) = r(t)c(t) \qquad (6.22)$$

Pour évaluer la séquence du watermark on doit intégrer *g(t)* sur chaque période *Tb* (période de chaque symbole de la séquence).

$$a_i = \int_{(i-1)Tb}^{iTb} g(t)dt \qquad (6.23)$$

où : $i = 1,2,\ldots$,nombre de bits constituant le watermark avant son l'étalement.

La règle de décision permettant de reconstituer le watermark $\{\hat{d}\}$ avant l'étalement est la suivante :

$$\hat{d} = \begin{cases} 1 & a_i > 0 \\ -1 & a_i \le 0 \end{cases} \qquad (6.24)$$

avec : $i = 1,2,\ldots$,nombre de bits constituant le watermark avant son l'étalement.

L'étape suivante consiste à utiliser la matrice d'Interleaving pour décoder le watermark pour cela la séquence est écrite suivant les lignes et est retranscrite suivant les colonnes, la séquence ainsi obtenu est appelée $\hat{w}_R$.

Le décodage de la séquence codée par le code répéteur dont le nombre de répétitions est m est donné par :

$$\hat{w}_k = \begin{cases} 1 & \sum_{i=1}^{m} \hat{w}_{ri} > 0 \\ -1 & \sum_{i=1}^{m} \hat{w}_{ri} \le 0 \end{cases} \qquad (6.25)$$

avec : $i = 1,2,\ldots$,nombre de bits constituant le watermark avant son l'étalement.

La séquence retrouvée $\{\hat{w}\}$ représente le watermark détecté.

6.5. MISE EN ŒUVRE DE LA MÉTHODE

L'application a été développée sous l'environnement MATLAB version 6 en raison de l'existence de fonctions intégrées nécessaires à notre traitement (transformée de Fourier, génération de nombre aléatoires...) au niveau de la librairie SIGNAL PROCESSING TOOLBOX. Nous avons cependant minimisé l'appel à ces fonctions intégrées pour optimiser le temps de calcul nécessaire aux traitements effectués.

Afin de simplifier l'entrée des paramètres de tatouage et d'extraction, nous avons développé un interface graphique plus convivial qui se présente comme suit :

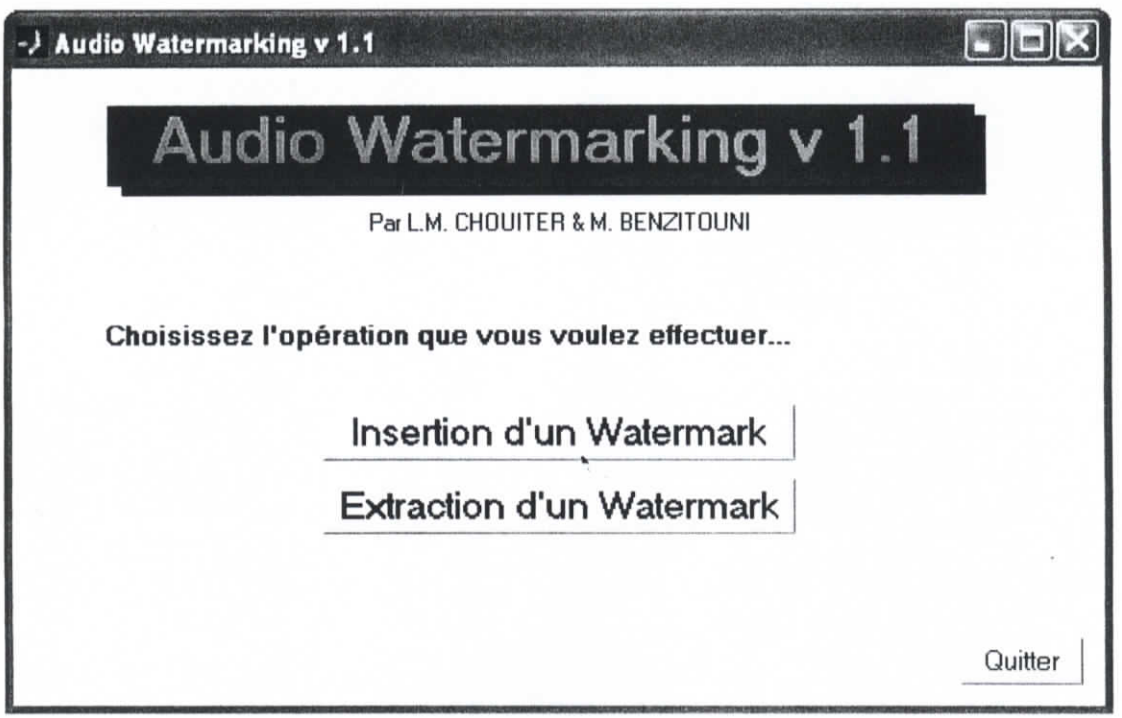


Figure 6.11 : *GUI (Graphical User Interface) du choix entre l'insertion ou l'extraction du watermark.*

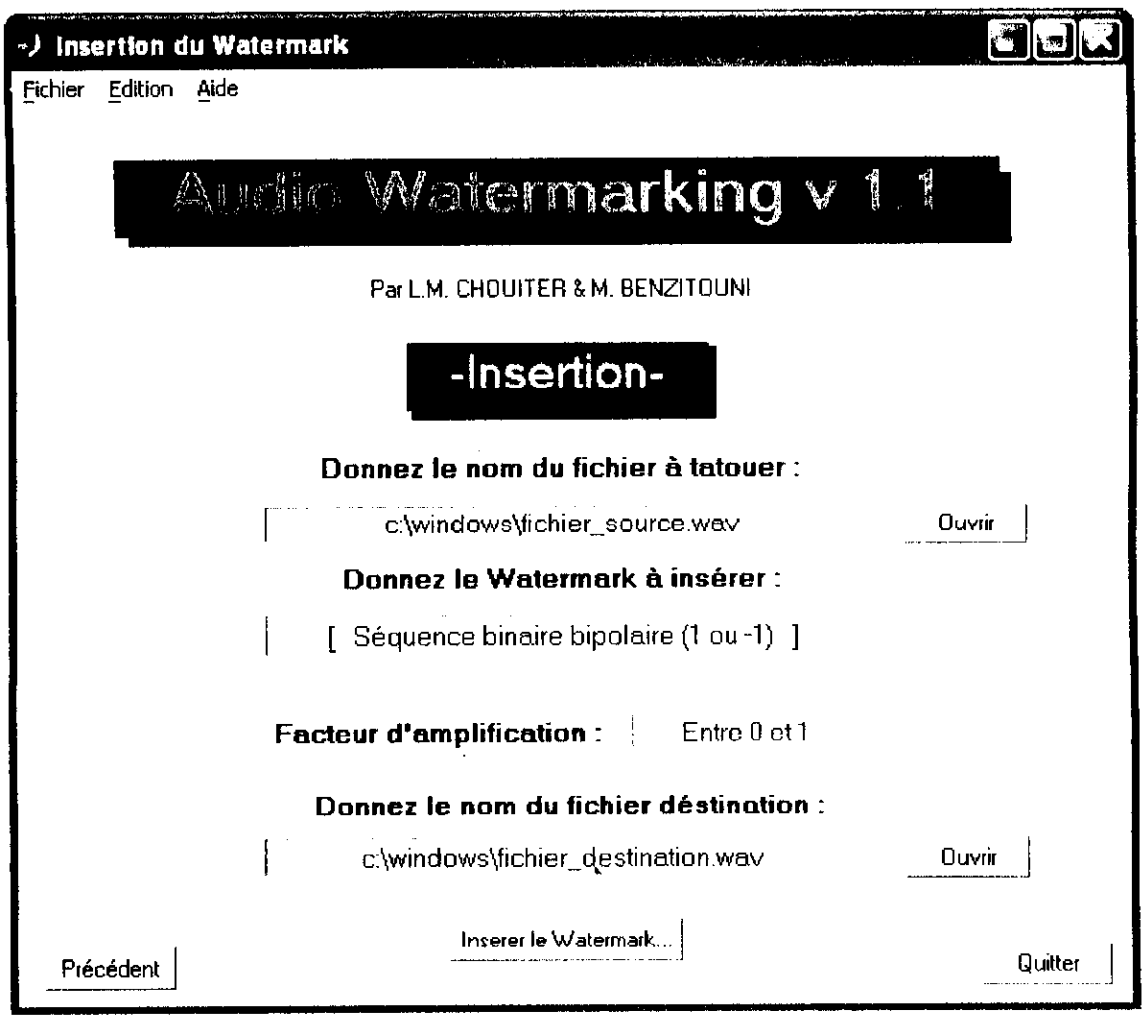


Figure 6.12 : *GUI (Graphical User Interface) de l'insertion du watermark.*

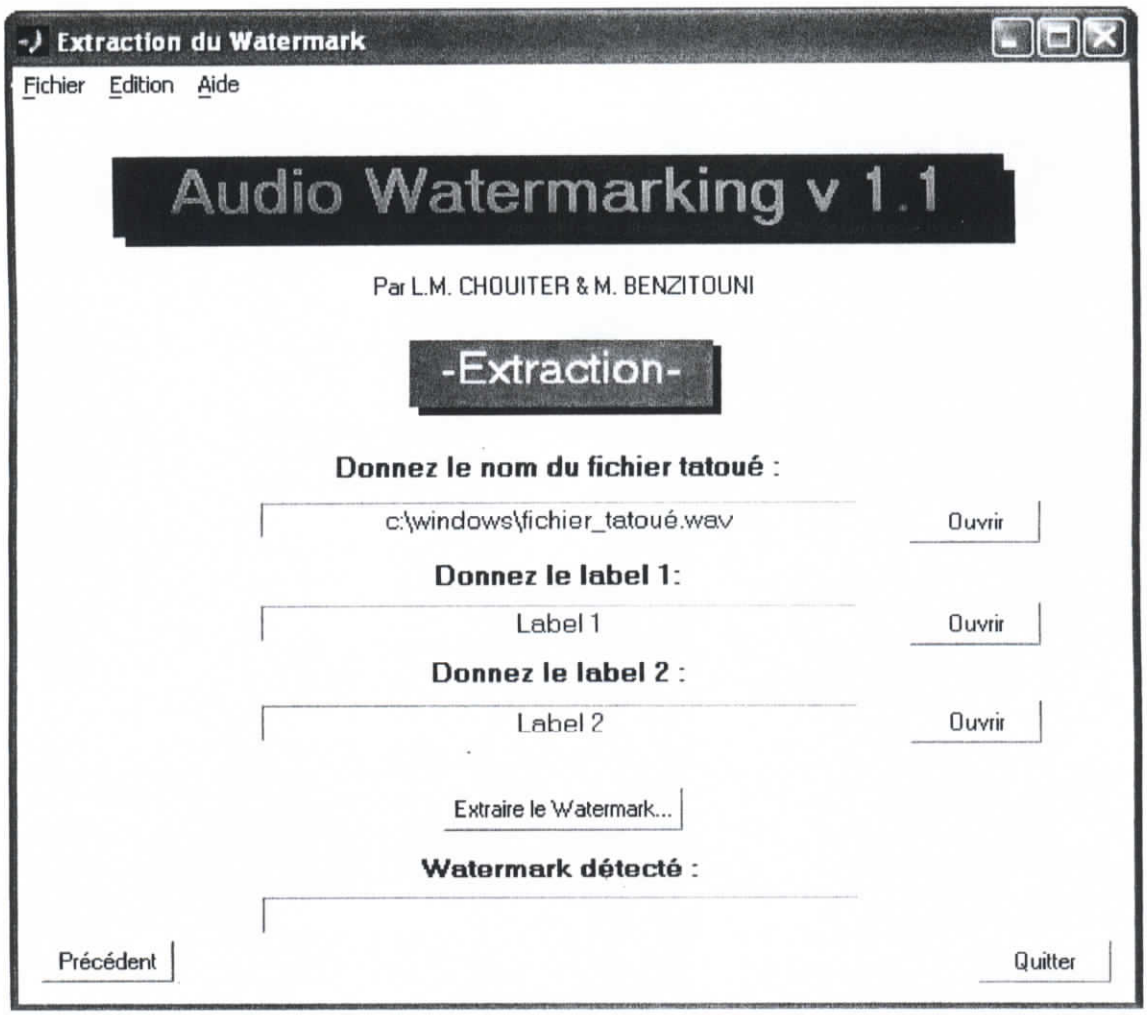


Figure 6.13 : *GUI (Graphical User Interface) de l'extraction du watermark.*

6.6. ÉXPERIMENTATIONS

Afin d'évaluer les performances et la qualité de l'algorithme de tatouage implémenté, une batterie de tests a été utilisée. Ces tests consistent en l'évaluation des watermarks insérés dans les fichiers, après les avoir attaqués selon différentes techniques.

Les fichiers sur lesquels les tests ont été effectués sont des fichiers musicaux de différents genres (blues, pop, metal, rock et classique)[15], tous d'une durée de 5 secondes et au format

[15] Les fichiers musicaux sur lesquels les tests ont été effectués sont:
Blues : *Groovin with the king* (King Curtis),
Classique : *Night on bald mountain* (Mussorgsky),
Metal : *Nothing else matters* (Metallica),
Pop : *Treat her like a lady* (Celine Dion),
Rock : Another brik in the wall, part II (Pink Floyd).

WAV où chaque échantillon est quantifié sur 16 bits avec une fréquence d'échantillonnage de 22050 Hz. Ces fichiers ont été, par ailleurs, tous tatoués en utilisant un même facteur d'amplification du watermark : $A = 0.5$.

Les résultats de ces tests, exprimés en pourcentages de bits identifiés, permettent d'apprécier la robustesse des watermarks insérés.

Les différentes techniques d'attaque utilisées pour tester la robustesse des watermarks insérés sont :

6.6.1. Le réechantillonnage et la requantification

Cette attaque consiste en la conversion d'un signal digital en un signal analogique (D/A) puis l'inverse (A/D). Durant ces deux opérations, le signal subit un réechantillonnage et une requantification.

Si le réechantillonnage n'a pas d'effet significatif sur la qualité du son pour des fréquences supérieures à 22050 Hz, les distorsions qu'il provoque sont cependant audibles[16] pour des fréquences inférieures à 22050 Hz.

Quant à la requantification, qui consiste à coder les composantes du signal audio sur 8 bits seulement alors qu'elles étaient initialement codées sur 16 bits, l'effet des distorsions qu'elle engendre est plus ou moins audible, pour la plupart des fichiers attaqués.

Les résultats de ces tests sont restitués par le tableau 6.1 ci-après.

Attaques / Musiques	Réechantillonnage Fréquence d'échantillonnage (Hz)					Requantification
	44100	32000	16000	12000	11025	Sur 8 bits
Blues	100 %	100 %	81.25 %	81.25 %	100 %	100 %
Classique	100 %	100 %	87.5 %	93.75 %	81.25 %	100 %
Metal	100 %	100 %	87.5 %	75 %	93.75 %	100 %
Pop	100 %	100 %	56.25 %	50 %	87.5 %	100 %
Rock	100 %	100 %	100 %	93.75 %	93.75 %	100 %

Tableau 6.1 : *Pourcentages de bits identifiés dans les fichiers tatoués après attaque par réechantillonnage et requantification.*

[16] Un bruit de fond est nettement audible pour une fréquence d'échantillonnage de 16 kHz.

6.6.2. La compression MP3

Cette attaque consiste à compresser les fichiers tatoués au format MP3 avec différents débits. Elle a été réalisée en utilisant les deux logiciels suivants :

- le premier, ***MP3EncoderX***, pour la compression (du WAV au MP3),
- et le second, *Easy MP3 Converter,* pour la décompression (du MP3 au WAV).

Bien que la qualité sonore des fichiers tatoués se soit très légèrement dégradée après la compression (particulièrement pour les faibles débits), les résultats des tests se sont avérés satisfaisants comme le montre le tableau 6.2 ci-après.

Débit (kbits/s) / Musique	320	256	224	192	160	128
Blues	100 %	100 %	100 %	100 %	100 %	100 %
Classique	100 %	100 %	100 %	100 %	100 %	100 %
Metal	100 %	100 %	100 %	100 %	100 %	93.75 %
Pop	100 %	100 %	100 %	100 %	100 %	100 %
Rock	100 %	100 %	100 %	100 %	100 %	100 %

Débits (kbits/s) / Musique	112	96	80	64	56	48
Blues	93.75 %	93.75 %	87.5 %	93.75 %	87.5 %	87.5 %
Classique	100 %	100 %	100 %	100 %	87.5 %	93.75 %
Metal	100 %	93.75 %	81.25 %	87.5 %	87.5 %	81.25 %
Pop	81.25 %	87.5 %	81.25 %	81.25 %	81.25 %	87.5 %
Rock	100 %	100 %	100 %	93.75 %	93.75 %	93.75 %

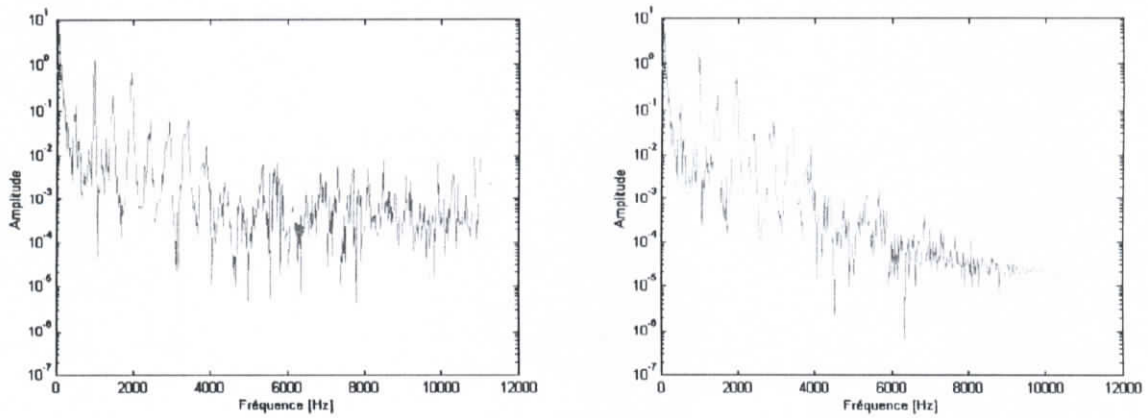
Tableau 6.2 : *Pourcentages de bits identifiés dans les fichiers tatoués après attaque par compression MP3.*

L'attaque opérée peut être considérée comme une attaque multiple. *MP3EncoderX* ne compresse que des fichiers WAV échantillonnés avec 44 kHz minimum et quantifiés sur 16 bits. Aussi, a-t-il été nécessaire de réechantillonner les fichiers avant de les compresser.

L'extraction du watermark, quant à elle, n'a pu s'opérer qu'après décompression puis réechantillonnage des fichiers avec la fréquence initiale f_c=22050 Hz.

6.6.3. Le filtrage passe-bas

Le filtrage passe-bas consiste à atténuer très fortement une partie du spectre du signal audio à partir d'une certaine fréquence de coupure f_c comme le montre les figures ci-après.



(a) **(b)**

Figure 6.14 : *Spectre d'un signal audio avant filtrage (a) et après filtrage (b) (Pour une fréquence de coupure f_c = 4000 Hz).*

Pour ce test, c'est le logiciel de traitement du son ***Diamond Cut Audio*** qui a été utilisé. Alors que le spectre des signaux audio sur lesquels les tests ont été effectués s'étendait jusqu'à 11025 Hz, ce logiciel a permis de filtrer ces signaux avec différentes fréquences de coupure allant de 5000 Hz à 3250 Hz.

Fréquence (Hz) / Musique	5000	4500	4000	3750	3500	3250
Blues	100 %	93.75 %	87.5 %	87.5 %	87.5 %	87.5 %
Classique	100 %	100 %	87.5 %	87.5 %	87.5 %	87.5 %
Metal	100 %	100 %	100 %	93.75 %	100 %	93.75 %
Pop	100 %	93.75 %	93.75 %	87.5 %	87.5 %	87.5 %
Rock	100 %	100 %	100 %	100 %	100 %	100 %

Tableau 6.3 : *Pourcentages de bits identifiés dans les fichiers tatoués après attaque par filtrage passe-bas.*

Les filtrages effectués ont introduit des distorsions audibles, particulièrement à partir de la fréquence de coupure 4000 Hz.

6.6.4. L'ajout et la suppression d'échantillons

L'ajout comme la suppression d'échantillons dans le signal audio constituent des techniques d'attaque visant simplement à désynchroniser le processus d'extraction. Elles n'ont en conséquence aucune incidence sur la qualité sonore des fichiers attaqués, sauf dans les cas où l'ajout ou la suppression portent sur un nombre élevé d'échantillons.

Les pourcentages de bits identifiés dans les fichiers attaqués selon ces techniques sont portés sur le tableau 6.4.

6.6.5. La restauration

Cette attaque qui consiste à "reconstruire" le signal tatoué, bloc par bloc[17], a été exécutée avec le logiciel ***Diamond Cut Audio***. Contre cette attaque également, l'algorithme s'est montré robuste comme le montrent les résultats portés sur tableau 6.4.

6.6.6. Ajout d'un bruit

Pour cette attaque qui consiste à ajouter au signal tatoué un bruit blanc gaussien, le test a été effectué avec des SNR allant de 40 dB jusqu'à 5 dB.

Même si la qualité sonore des fichiers a été très affectée par cette attaque (particulièrement pour des SNR inférieurs à 10 dB), le watermark a été parfaitement identifié comme le montre le tableau 6.4.

Attaques / Musique	Décalages En temps	Suppression d'échantillons	Restauration	Ajout de bruits
Blues	100 %	100 %	100 %	100 %
Classique	100 %	100 %	100 %	100 %
Metal	100 %	100 %	100 %	100 %
Pop	100 %	100 %	93.75 %	100 %
Rock	100 %	100 %	100 %	100 %

Tableau 6.4 : *Pourcentages de bits identifiés dans les fichiers tatoués après attaque par ajout d'échantillons, suppression d'échantillons, restauration et ajout de bruits.*

[17] Technique décrite au chapitre 4 (paragraphe 4.2.6).

6.7. RÉSULTATS ET PERSPECTIVES

Les fichiers obtenus après tatouage se sont révélés, en utilisant un facteur d'amplification A=0.5, de qualité pratiquement semblable à celle des originaux même si de très légères distorsions ont pu être perçues pour le fichier musical de type « *Classique* ». Une réduction du facteur d'amplification du watermark permettrait la suppression des ces distorsions mais entraînerait une perte de robustesse du tatouage face aux attaques, particulièrement à la compression MP3.

Les tests effectués ont permis d'identifier le watermark inséré, dans tous les fichiers attaqués dont la qualité sonore n'a pas été altérée, et ce, avec un maximum de 3 bits erronés sur 16. Ce résultat autorise à considérer que le watermark identifié est bien le watermark inséré, dès lors que le pourcentage de bits identifiés est supérieur à 80%.

Même dans les cas où l'altération de la qualité sonore des fichiers attaqués était perceptible, le watermark a été identifié dans les mêmes conditions (maximum de 3 bits erronés sur 16), sauf dans un cas.

Le fichier « *Pop* » s'est avéré le moins résistant dans la mesure où sa qualité sonore se détériorait à la moindre attaque. Ceci n'a cependant pas empêché l'identification du watermark qui y était inséré, sauf dans le cas de l'attaque par réechantillonnage. En effet, le watermark n'a été identifié dans ce cas qu'à 50 %. En raison cependant de la médiocrité de la qualité sonore du fichier engendrée par cette attaque, celle-ci ne peut être considérée comme réussie.

Pour les travaux futurs, une amélioration des performances de l'algorithme de tatouage semble pouvoir être obtenue par l'utilisation d'un code correcteur qui permettrait de réduire le nombre de bits erronés révélés lors de l'extraction (un code de HAMMING par exemple).

Pour ce qui est des expérimentations, plusieurs axes mériteraient d'être explorés tels que la recherche des paramètres optimaux concernant le facteur d'amplification, la longueur des segments sur lesquels s'effectuent les traitements, etc. Il serait également intéressant d'appliquer les tests à d'autres genres de fichiers audio.

CONCLUSION

Le travail réalisé consistait à implémenter une méthode permettant l'insertion d'informations sur le copyright dans les signaux audio. La méthode implémentée porte sur l'insertion d'un watermark binaire bipolaire d'une longueur de 16 bits, au moyen d'une clé pseudo-aléatoire. Pour que l'insertion n'introduise pas des distorsions audibles susceptibles d'endommager la qualité sonore du signal à tatouer, cette méthode effectue ses traitements dans le domaine fréquentiel et se base sur un modèle psychoacoustique. De même, pour garantir la robustesse du tatouage inséré, elle utilise la technique de l'étalement de spectre.

Le procédé d'extraction ne requiert pas la connaissance du signal audio original, mais celle de la clé pseudo-aléatoire (label 1) utilisée lors de l'insertion et d'une seconde clé de synchronisation (label 2).

Les fichiers obtenus après tatouage ont conservé une qualité sonore proche de celle des originaux. Quant aux watermarks insérés dans ces fichiers, ils se sont avérés résistants aux nombreuses attaques qui les visaient (compression MP3, réechantillonnage, requantification, filtrage passe-bas, ajout d'un bruit blanc, ajout et suppression de composantes et restauration).

Des améliorations peuvent encore être apportées à la méthode implémentée, notamment en vue de la réduction du temps d'insertion et d'extraction des watermarks. La reprogrammation de l'algorithme en C++ permettrait d'atteindre cet objectif. De même, d'autres recherches pourraient être consacrées à l'amélioration des paramètres utilisés par l'algorithme pour que l'insertion des filigranes puisse être encore plus transparente et plus robuste. Enfin, l'utilisation d'un code correcteur d'erreurs permettrait d'augmenter les performances de cet algorithme lors du processus d'extraction du watermark.

BIBLIOGRAPHIE

[1] F.A.P. Petitcolas, R.J.Andeson and M.G.Kuhn, "Information Hiding-A Survey," in *Proceeding of the IEEE*, vol. 87, No. 7, July 1999, pp.1062-1077.

[2] L. Boney, A.H.Tewfik, K.N.Hamdy, "Digital watermarks for audio signals," in *Proc. 1996 IEEE Int. Conf. Multimedia Computing and Systems*, Hiroshima, Japan, June 17-23, 1996, pp. 473-480.

[3] S.Czewinski, R.Fromm, T.Hodes, "Digital music distribution and audio watermarking," *Computer Science Division*, University of California, Berkeley (2000).

[4] M.Arnold, "Audio watermarking : Features, applictions and algorithms," Institute of Electrical and Electronics Engineers (IEEE): *IEEE International Conference on Multimedia and Expo 2000. Proceedings CD-ROM 2000*, S. 1013-1016.

[5] C. Amar et E. Michon, "Le Mpeg Audio 1," *Tipe 2001.*

[6] R. Boite, H. Pourlard, T. Dutoit, J. Hancq, H. Leich, "Traitement de la parole", Presse Polytechnique Universitaire Romandes, 2000.

[7] C.A. Lanciani, "Auditory Perception and the MPEG Audio Standard," Ph.D. thesis. Georgia Institute of Technology. August 11,1995.

[8] D.Y. Pan, "Digital audio ompression," in *Digital Technical Journal* Vol. 5 No. 2, Spring 1993.

[9] Calliope, "La parole et son traitement automatique " Editions Masson, 1989.

[10] N. Jayant, J. Johnston, and R. Safranek, "Signal compression based on models of perception," *Proceedings of the IEEE*, Vol. 81, pp. 1385-1422, Oct. 1993.

[11] E. Zwicker and H. Fastl, "Psychoacoustics : Facts and Models," Ch 1-4, 6-8. New York Springer-Verlag, 1990.

[12] D. Robinson, "Audio coding : 3-dimentional stereo and presence," *CIN/TIS/M.Eng*, University of Essex, Jan. 2002.

[13] D. Pan, "A Tutorial on MPEG/Audio Compression," *Proceeding of the IEEE Multimedia Journal*, Summer 1995.

[15] F.A.P. Petitcolas, Markus G. Kuhn and Ross J. Anderson, "Attacks on Copyright Marking Systems," David Aucsmith, Ed., *Second workshop on information hiding*, in vol. 1525 of Lecture Notes in Computer Science, Portland, Oregon, USA, 14-17 April, 1998, pp. 218-238.

[16] S. Voloshynovskiy, S. Pereira, T. Pun, J.J. Eggers and J.K. Su, "Attacks on Digital Watermarking : Classification, Estimation-Based Attacks, and Benchmarks" *Proceedings in the IEEE Communication Magazine*, August 2001.

[17] Ingemar J. Cox and Jean-Paul M.G. Linnartz, "Some general methods for tampering with watermarks," appeared in *IEEE International conference on image processing*, 1997.

[18] S. V. Vaseghi, "Algorithms for restoration of archived gramo-phone recording". Ph.D. thesis, Emmanuel College, University of Cambridge, England, Feb. 1988.

[19] "Giovanni audio marking software" *Blue Spike company*, http://www.bluespike.com, May 1998.

[20] M. Kutter, S. Voloshynovskiy and A. Herrigel, "The Watermark Copy Attack" In *Ping Wah Wong and Edward J. Delp eds., IS&T/SPIE's 12th Annual Symposium, Electronic Imaging 2000 : Security and Watermarking of Multimedia Content II*,Vol.3971 of SPIE Proceedings, San Jose, California USA, 23-28 January 2000.

[21] S. Katzenbeisser and H. Veith, "Securing Symmetric Watermarking Schemes Against Protocol Attacks" Proceed*ings of the SPIE vol. 4675, Security and Watermarking of Multimedia Contents IV*, 2002.

[22] M. Steinebach, A. Lang, J. Dittmann ane F.A.P. Petitcolas, "StirMark Benchmark : audio watermarking attacks based on lossy compression" *Electronic Imaging* 2002.

[23] F. Hartung and M.Kutter, "Multimedia watermarking techniques", *Proc.IEEE*, vol. 87, no. 7, pp. 1079–1107, July 1999.

[24] M. D. Swanson, M. Kobayashi, and A. H. Tewfik, "Multimedia data embedding and watermarking technologies", *Proc. IEEE*, vol. 86, no. 6, pp. 1064–1087, June 1998.

[25] M. Swanson, B. Zhu, and A. Tewfik, "Current state of the art, challenges and future directions for audio watermarking," in *Proc. IEEE ICMCS*, vol. 1, Florence, Italy, June 7–11, 1999, pp. 19–24.

[26] P. Bassia, Pitas, "Robust Audio Watermarking in the Time Domain", *Proceedings of ICASSP* 1999.

[27] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding", *IBM Syst.J.*, vol. 35, no. 3&4, pp. 313–335, 1996.

[28] L. Boney, A. H. Tewfik, and K. N. Hamdy, "Digital watermarks for audio signals," in *Proc. EUSIPCO*, vol. III, Sept. 1996, pp. 1697–1700.

[29] M. D. Swanson, B. Zhu, A. H. Tewfik, and L. Boney, "Robust audio watermarking using perceptual masking", *Elsevier Signal Processing, Special Issue on Copyright Protection and Access Control*, vol. 66, no. 3, pp. 337–355, 1998.

[30] W. Kim, J. Lee, and W. Lee, "An audio watermarking scheme robust to mpeg audio compression," in *Proc. NSIP*, vol. I, Antalya, Turkey, Juin 20–23, 1999, pp. 326–330.

[31] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[32] D. Gruhl, A. Lu, andW. Bender, "Echo hiding," in *Proc. 1st Information Hiding Workshop*, Cambridge, U.K., Mai 1996, pp. 295–316.

[33] Sandford, S. et.al.,"compression Embedding",*US Patent 5,778,102* (1997).

[34] Petitcolas, F. (1999), Cambridge Univ. of UK.
http://www.cl.cam.ac.uk/~fapp2/steganography/mp3stego/.

[35] K. Nahrstedt, "Non-invertible watermarking methods for mpeg video and audio," in *Multimedia and SecurityWorkshop, ACMMultimedia'98*, Bristol, U.K., Sept. 1998.

[36] Wen-Nung Lie, Li-Chun Chang, "Robust and high-quality time-domain audio watermarking subject to psychoacoustic masking", *Proc. Of IEEE International Symposium on Circuits and Systems, ISCAS-2001*, Sydney, Australia, pp.II-45~48.

[37] Khalid A. Kaabneh, Abdou Youssef, "Muteness-Based Watermarking Technique", *The 3rd International Workshop on Multimedia* Network Systems, 2001.

[38] P. Bassia, I. Pitas, and N. Nikolaidis, "Robust Audio Watermarking in the Time Domain", *IEEE Transactions on multimedia*, Vol. 3, No. 2, June 2001

[39] C. Xu , J. Wu and D.D. Feng, "Content-based digital Watermarking for compressed audio", *in Proc. IEEE Int. Conf. on multimedia computing and systems*, Sydney, Australia 2000.

[40] ISO/CEI, "Codage de l'image animée et du son associe pour les supports de stockage numerique jusqu'a environ 1,5 Mbit/s", Tech. Rep. 11172, ISO/CEI, 1993. pp.303–311.

[41] R.A. Garcia, "Digital watermarking of audio signals using a psychoacoustique auditory model and spread spectrum theory", *107th Convention, Audio Engineering Society*, New York, NY, September 24-27, 1999.