

**PROPUESTA DEL USO DE MARCAS DE AGUA EN
AUDIO COMO SECUENCIAS DE CONTROL**

TESIS

Que para obtener el grado de
MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA

presenta

Antonio Ulises Sáenz Trujillo

Dr. Felix Calderón Solorio

Director de Tesis

Universidad Michoacana de San Nicolás de Hidalgo
División de Estudios de Posgrado de la Facultad de Ingeniería Eléctrica

Agosto 2008

Resumen

En este documento se propone el uso de marcas de agua en audio como secuencias de control, para manipular equipos de radio y/o televisión que realizan la conmutación entre las programaciones local, y nacional. Lo anterior, implica el embebido de información en señales de audio (que son utilizados como inicio y fin de tanda de comerciales) en forma de una marca de agua digital, y donde dicho audio tendrá que ser transmitido por medios analógicos sufriendo diversos ataques como corrupción por ruido, filtrado pasabajas, expansión en tiempo y amplitud, translaciones, etc.. Por las condiciones de transmisión de la señal de audio, las marcas de agua deberán ser diseñadas para ser robustas a ataques como los anteriormente mencionados, que generalmente aparecen tras una conversión D/A - A/D.

En el estado del arte se pueden encontrar un gran número de trabajos enfocados en el embebido y extracción de información en señales de audio, sin embargo no se encontró alguno que reportara robustez ante todos los ataques mencionados anteriormente. Por lo que se eligió un par de algoritmos en el estado del arte, que pos si solos no cumplen con todos los requerimientos necesarios. Sin embargo, uno de estos algoritmos cumple la mayoría de los requerimientos, mientras que el otro cumple los requerimientos que el primero no.

Por lo anterior, en este documento se propone un algoritmo que toma como base los principios básicos de funcionamiento de dos algoritmos en el estado del arte, logrando un algoritmo que se basa en la modificación simultanea de dos coeficientes de baja frecuencia, obtenidos del cálculo de la Transformada Wavelet Discreta (TWD), de la señal de audio en cuestión, la cual ha sido muestreada a 44100 Hz, y que ha sido dividida en bloques de 256 muestras. Este algoritmo tiene como principal cualidad, el no afectar de manera perceptible la calidad del audio anfitrión. Por otra parte, la marca de agua tiene como cualidades el ser *autosincronizada*, robusta contra ataques de ruido gaussiano, compresión mp3, recuantización, remuestreo, así como escalamientos en amplitud y tiempo, los cuales aparecen durante la transmisión analógica de la señal de audio anfitrión.

Contenido

Resumen	III
Contenido	V
Lista de Figuras	IX
Lista de Tablas	XI
Lista de Símbolos	XIII
1. Introducción	1
1.1. Planteamiento del Problema	1
1.2. Antecedentes	4
1.2.1. Marcas de Agua en Audio	4
1.2.2. Aplicaciones de las Marcas de Agua en Audio	4
1.2.3. Requerimientos que una Marca de Agua en Audio Debe Cumplir	6
1.3. Objetivos de la Tesis	8
1.3.1. Objetivo general	8
1.3.2. Objetivos particulares	8
1.4. Descripción de Capítulos	8
2. Esteganografía y Marcas de Agua en Audio	11
2.1. Audio Digital	11
2.1.1. Notación	12
2.1.2. Potencia de un Segmento de Audio	12
2.2. Principios del Insertado de Marcas de Agua en Audio	13
2.2.1. Modelo General del Proceso de Inserción de Marcas de Agua en Audio	14
2.2.2. Concepto del Término Ataque	15
2.3. Efectos de la Conversión A/D y D/A en Señales Digitales	15
2.3.1. Medios de Transmisión	16
2.3.2. Distorsión de la Magnitud de la Señal	17
2.3.3. Distorsión por Escalamiento en Tiempo	17
2.3.4. Corrupción por Ruido	17
2.3.5. Modelo de Distorsión	18
2.4. Algoritmos de Inserción de Marcas de Agua	18
2.4.1. Embebido de Información Mediante la Inserción de Tonos Imperceptibles	19
2.4.2. Códigos de Sincronización	21

2.4.3. Marcas de Agua en el Dominio de la Transformada Wavelet Discreta	23
2.5. Conclusiones del Capítulo	26
3. Marcas de Agua Robustas, Estrategias de Inserción y Detección de Marcas	29
3.1. Caracterización del Dispositivo Capturador para Contrarrestar el Escalamiento en Tiempo	30
3.1.1. Remuestreo Digital	31
3.1.2. Búsqueda del Factor de Escalamiento en Tiempo Causado por el Dispositivo Capturador	33
3.2. Algoritmo de Wu con Normalización de Potencia	36
3.2.1. Procedimiento de Insertado	36
3.2.2. División del Audio en Marcos	36
3.2.3. Códigos de Sincronización a Insertar	38
3.2.4. Datos a Embeber	38
3.2.5. Normalización de la Potencia de un Marco	39
3.2.6. Embebido de Datos.	40
3.2.7. Extracción de la Información.	41
3.2.8. Algoritmo de Búsqueda de Códigos de Sincronización	42
3.3. Algoritmo Por Modificación Simultánea de Coeficientes	44
3.3.1. Algoritmo de Embebido de Datos	44
3.3.2. Extracción de Datos	47
3.4. Conclusiones del Capítulo	48
4. Experimentos y Resultados	51
4.1. Condiciones de los Experimentos	51
4.2. Observaciones a los Algoritmos Usados como Base	52
4.2.1. Observaciones al Algoritmo de Inserción de Tonos Imperceptibles	52
4.2.2. Observaciones al Algoritmo Basado en la Transformada Wavelet Discreta	53
4.3. Medición de la Distorsión por Embebido de Información	53
4.4. Mediciones de Tiempos de Ejecución	55
4.5. Experimentos con el Algoritmo de Modificación Simultánea de Coeficientes	56
4.5.1. Compresión mp3	57
4.5.2. Ataque por Ruido Blanco	57
4.5.3. Ataque por Conversión D/A - A/D	61
4.6. Conclusiones del Capítulo	61
5. Conclusiones	63
5.1. Conclusiones Generales	63
5.2. Trabajos Futuros	65
A. Relación Señal-Ruido	67
B. Circuito de Muestreo y Retención	69

C. Transformada Wavelet	71
C.1. Análisis Multiresolución	71
C.2. Características de la Transformada Wavelet	72
C.3. Cálculo de la Transformada Wavelet Discreta	73
C.3.1. Esquema Piramidal	74
Referencias	77
Glosario	81

Lista de Figuras

1.1. Esquema típico de una <i>carta de continuidad</i> de transmisión de radio.	3
2.1. Modelo general del proceso de inserción y extracción de marcas de agua en audio	14
2.2. Posibles ambientes de transmisión del audio	16
2.3. Algoritmo de Inserción de Tonos Propuesto por Gopalan [Gopalan04]	20
2.4. Esquema de marcado del algoritmo de Wu basado en transformada wavelet.	24
2.5. Esquema de extracción de marca del algoritmo de Wu basado en transformada wavelet.	25
3.1. Una señal muestreada a dos diferentes frecuencias.	33
3.2. Esquema de Embebido del algoritmo con normalización de potencia.	37
3.3. Estructura de la secuencia a insertar	39
3.4. Esquema de extracción de información del algoritmo por normalizado de potencia.	41
3.5. Diagrama de flujo que representa el algoritmo de marcado usando modificación simultanea de coeficientes	45
B.1. Circuito de muestreo y retención.	69
C.1. Escala Diádica	73
C.2. Esquema de codificación piramidal para el cálculo de la TWD	75

Lista de Tablas

4.1. Comparativa de distorsión causada por embebido de datos en una señal de voz.	54
4.2. Comparativa de distorsión causada por embebido de datos en una melodía del género bolero.	54
4.3. Comparativa de distorsión causada por embebido de datos en sonido de cuerdas.	54
4.4. Comparativa de distorsión causada por embebido de datos en sonido de piano.	55
4.5. Comparativa de distorsión causada por embebido de datos en una melodía de mariachi.	55
4.6. Comparativa del promedio de tiempo de ejecución entre los tres algoritmos basados en transformada wavelets.	56
4.7. Porcentajes de error después de ataque por compresión mp3 en señal de voz, usando un valor de fuerza de embebido de $S = 0.25$	57
4.8. Porcentajes de error después de ataque por compresión mp3 en melodía de bolero, usando un valor de fuerza de embebido de $S = 0.3$	58
4.9. Porcentajes de error después de ataque por compresión mp3 en sonido de cuerdas, usando un valor de fuerza de embebido de $S = 0.09$	58
4.10. Porcentajes de error después de ataque por compresión mp3 en sonido de piano, usando un valor de fuerza de embebido de $S = 0.09$	58
4.11. Porcentajes de error después de ataque por compresión mp3 en melodía de mariachi, usando un valor de fuerza de embebido de $S = 0.25$	59
4.12. Porcentajes de error en la detección de una marca en señal de voz contaminada a distintos niveles de ruido	59
4.13. Porcentajes de error en la detección de una marca en señal de voz contaminada a distintos niveles de ruido	60
4.14. Porcentajes de error en la detección de una marca en sonido de cuerdas contaminado a distintos niveles de ruido	60
4.15. Porcentajes de error en la detección de una marca en sonido de piano contaminado a distintos niveles de ruido	60
4.16. Porcentajes de error en la detección de una marca en una melodía de mariachi contaminada a distintos niveles de ruido	61
4.17. Porcentajes de error después de haber sometido la señal a una conversión D/A - A/D	61

Lista de Símbolos

$\psi(t)$	Wavelet Haar
$\Re(Z)$	Parte real de un número Z .
$\Im(Z)$	Parte imaginaria de un número Z .
c_i	Valor del i -ésimo coeficiente de la TWD.
\hat{c}_i	Valor del i -ésimo coeficiente de la TWD modificado por regla de embebido.
x	Señal de audio.
\hat{x}	Señal de audio con marca de agua embebida.
N	Tamaño de un marco de audio.
$x_{n,m}$	Valor del n -ésima muestra de una señal discreta en el m -ésimo marco.
$X_{k,m}$	Valor del k -ésimo coeficiente en el dominio de la frecuencia, en el m -ésimo marco de una señal de audio.
ε_m	Valor de la energía del m -ésimo marco de una señal de audio.
P_m	Valor de la potencia promedio del m -ésimo marco de una señal de audio.
a	Secuencia binaria a ser embebida.
a_i	i -ésimo bit de la secuencia binaria a ser embebida.
d_{atq}	Distorsión en un audio marcado causado por un ataque.
d_{emb}	Distorsión en un audio causado por el embebido de información.
e	Umbral de error.
f_s	Frecuencia de muestreo.
S	Factor de intensidad de embebido de marca de agua.
L	Tamaño de la secuencia binaria que es usada como etiqueta de sincronización (Secuencia PN).
M	Número de marcos en los que ha sido dividido un audio.
N	Tamaño de un marco de audio.

Capítulo 1

Introducción

1.1. Planteamiento del Problema

Actualmente en México, como en el resto del mundo, existen una gran cantidad de empresas de radio y televisión que, con el fin de mantenerse vigentes y en el gusto del público, se afilian a otras empresas productoras conformando grandes cadenas de cobertura nacional. En general, la afiliación de una estación ya sea de radio o televisión, con una cadena productora, consiste en el permiso que otorga la productora a la estación de retransmitir los contenidos generados por dicha productora, incluyendo publicidad que ha sido contratada para ser transmitida a nivel nacional, dejando algún margen de tiempo para que la estación local lo pueda comercializar a su conveniencia, obteniendo ganancias por el contenido que la cadena generó. Esta dinámica de trabajo, requiere de un mecanismo en todas y cada una de las estaciones repetidoras (que generalmente se encuentran en provincia) para conmutar entre la comercialización local, y la señal que debe ser transmitida a nivel nacional. Este trabajo, en la gran mayoría de los casos lo realiza un operador humano.

Basta con visitar unas cuantas estaciones de radio o televisión para darse cuenta de que éstas se van equipando con sistemas que automatizan casi por completo las transmisiones de su contenido. Sin embargo, uno de los últimos aspectos que no todas las estaciones han automatizado, es precisamente el trabajo que realiza un operador humano conmutando entre el contenido foráneo y el contenido local.

Aquellas estaciones que han automatizado la conmutación anteriormente mencionada, utilizan un sistema de control por tonos, el cual consiste en transmitir tonos audibles de cierta duración y frecuencia¹. Éstos tonos son detectados por una computadora, la cual, los usa como comandos de control para hacer la conmutación entre la comercialización local y la señal nacional. La Figura 1.1 muestra un esquema típico de una *carta de continuidad* de transmisión de radio que utiliza tonos DTMF como comandos de control. En dicho esquema se puede ver como un bloque de comercialización local, precede un pequeño bloque compuesto por una identificación de estación, un tono DTMF (que sirve de comando de control) que manipula el equipo que realiza la conmutación de la señal local a la señal nacional, y un audio de introducción de programa comúnmente conocido como *cortinilla*. El tiempo de transmisión subsecuente pertenece a la programación nacional, que está compuesto por un primer bloque de programa, un bloque de comercialización nacional, un segundo bloque de programa, y un medio corte de comercialización nacional. Este último medio corte, termina con un audio de ambiente e identificación de estación conocido comúnmente como *jingle*, y un tono DTMF que indica el inicio de tiempo para comercialización local y que sirve como comando de control para conmutar entre la señal nacional, y la señal local, que consiste principalmente por anuncios de patrocinadores locales.

Este método, aunque efectivo, no es muy utilizado por radiodifusoras y televisoras, pues resulta ser invasivo a los contenidos, y/o invasivo a los valiosos tiempos de transmisión (tal y como se puede ver en la Figura 1.1). Este inconveniente, es precisamente una de las razones que motivó la investigación del tema aquí expuesto, pues se desea superar el problema de la invasión en tiempo de transmisión o contenidos, que como se mencionó, es precisamente una de las razones por las que una gran cantidad de televisoras y radiodifusoras no han automatizado por completo sus procesos.

Otra de las necesidades que motivaron este trabajo, es que desde hace bastante tiempo (tal vez desde que se comenzó la explotación comercial de los medios electrónicos de comunicación), se ha tenido la necesidad de hacer *Auditorías de Transmisión*, las cuales consisten en verificar que los anuncios contratados, sean transmitidos en tiempo y forma. La

¹La mayoría de los programas disponibles comercialmente, y que soportan este método, utilizan tonos duales multifrecuencia (DTMF por sus siglas en Inglés y que son utilizados en telefonía).

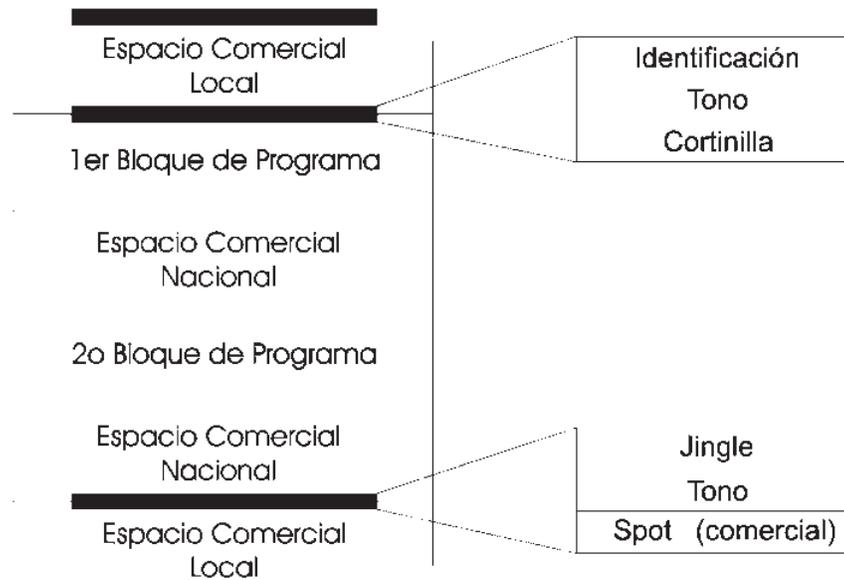


Figura 1.1: Esquema típico de una *carta de continuidad* de transmisión de radio.

manera que es resuelto en casi la totalidad del país es mediante la grabación de una *cinta testigo*, la cual como su nombre lo indica, es la grabación de la transmisión que se ha hecho durante todo el día, por al menos un mes seguido. En caso de ser necesaria una auditoría de transmisión, una persona se encargará de rastrear los anuncios en las grabaciones de los horarios en que se supone deberían haber sido transmitidos, pudiéndose convertir en una tarea bastante lenta y tediosa. Aunque ya existen formas alternativas y completamente automáticas de realizar dichas auditorías, éstas no están al alcance de todas las empresas que pudieran necesitarlas.

Después de una lectura sobre el uso de técnicas esteganográficas [Ruanaidh98] para ocultar y transmitir información en imágenes, se intuyó que esta misma tecnología podría servir para resolver los problemas anteriormente descritos. Con esta idea en mente, se hizo una revisión del estado del arte encontrando varios trabajos que presentan una serie de técnicas para embeber información en secuencias de audio.

De la lectura de dichos artículos en los que se proponen diversos algoritmos de

marcado y embebido de información, de las características de la información embebida que reportan los autores de dichos artículos, y de la necesidad de evitar la invasión al contenido a transmitir, o al tiempo del mismo, se concluyó que esta tecnología era adecuada para ser utilizada y presentar una alternativa a las soluciones descritas anteriormente y que son utilizadas actualmente.

1.2. Antecedentes

1.2.1. Marcas de Agua en Audio

El proceso de insertar y extraer marcas de agua en audio, inicialmente se trató como una subdisciplina del procesamiento digital de señales, la cual se enfocaba principalmente en las técnicas adecuadas para embeber información a secuencias de audio. Los trabajos en esta subdisciplina se limitaban a la investigación y búsqueda de alguna transformación de dominio adecuada para el embebido de la marca, además de la creación de esquemas para la modificación del audio anfitrión procurando que dicha modificación fuese imperceptible al oído humano [Cvejic04].

Fue hasta hace unos pocos años que el *Embebido de Marcas de Agua en Audio*, pasó a ser una disciplina madura con fuertes fundamentos teóricos y bases tanto en modelos de comunicación, como en teoría de la información [Moulin01]. Tal crecimiento es debido a que actualmente el “Embebido de Información” y la inclusión de “Marcas de Agua”, están jugando un papel preponderante en el área de “Protección de Derechos de Autor” [Swanson98], [Arnold00].

1.2.2. Aplicaciones de las Marcas de Agua en Audio

El embebido de información en una señal de audio puede tener varias aplicaciones prácticas. Arnold [Arnold00] presenta un listado de las aplicaciones más comunes que se le ha dado a las marcas de agua en audio. A continuación se presenta un resumen de dicho listado.

La principal aplicación de las marcas de agua en audio es en el ámbito de la *Protección de Derechos de Autor*. Esta afirmación se puede comprobar al revisar el estado

del arte donde la mayoría de los trabajos están enfocados en este objetivo. En general, se asume que una marca de agua embebida en audio puede servir como una prueba de la propiedad de dicho audio, al confirmar la existencia de la clave secreta sólo conocida por su creador.

Otra de las aplicaciones, que de hecho está muy relacionada con la anterior, es el *Monitoreo*². Esta aplicación consiste en la búsqueda de marcas de agua secretas en contenidos de audio con el fin de rastrear copias ilegales de contenido multimedia.

Una marca de agua también puede ser usada como medio de *Identificación Segura* (Fingerprinting). Arnold [Arnold00] afirma que en ambientes de distribución de contenido punto a punto, la información de los clientes autenticados puede ser embebida como una marca de agua secreta, justo antes del envío seguro de datos, y servir como llave de acceso a otros contenidos.

El análisis de una señal que se sabe está marcada, y la extracción de su marca de agua, puede servir como una “Indicación de Manipulación de Contenido”. Es decir, una forma de saber si un contenido ha sido modificado de su estado original, es mediante la búsqueda de una marca de agua pública y de naturaleza bastante frágil. Si la marca de agua es distinta a la esperada, o incluso imposible de recuperar, es obvio que el audio analizado ha sido modificado.

Otra aplicación bastante interesante, y que posiblemente tomará mayor relevancia en un futuro, es que una marca de agua puede ser usada como una *Portadora de Información*. Arnold [Arnold00] afirma que una marca de agua pública embebida en un flujo de datos, puede actuar como un subcanal de transmisión dirigido a una base de datos, la cual puede recopilar información tal como derechos de autor y/o las condiciones de la licencia establecida.

De la observación de las necesidades de la industria de la radio y la televisión en nuestro país, surge otro enfoque de la aplicación anteriormente mencionada. La idea es que un audio marcado que es transmitido ya sea por radio o televisión, puede servir como un canal de transmisión de datos, donde la marca de agua contenga información, que puede ser

²La descripción de esta aplicación es la tradicional contenida en la literatura, la cual no cubre el tipo de monitoreo propuesto en este trabajo.

interpretada ya sea como una secuencia de control de uso interno para dichas estaciones, o como información pública donde se le indique al auditorio el nombre, duración, autor, etc. de la melodía o programa en curso. La tecnología actual de radio en *Frecuencia Modulada* (F.M.) (que es la misma que se utiliza para el audio en televisión), es capaz de enviar este tipo de información al auditorio. Esta tecnología es el *Sistema de Radiodifusión de Datos* (RDS, acrónimo del inglés Radio Data System), que es una técnica que permite añadir, de forma inaudible, información relacionada con los programas de radio en F.M.. La señal digital que contiene dicha información, se transmite con una velocidad de 1187.5 bps y modula una subportadora de 57 KHz, utilizando el método de modulación de amplitud con portadora suprimida [Tomasi00], que se suma a la señal múltiplex estereofónica a la entrada del transmisor de frecuencia modulada. Sin embargo la tecnología usada en la transmisión de radio en *Amplitud Modulada* (A.M.) no lo permite por tener un menor ancho de banda, y debido a que la modulación del RDS es precisamente de A.M..

Esta aplicación es la que inspira la investigación presentada en este trabajo.

1.2.3. Requerimientos que una Marca de Agua en Audio Debe Cumplir

De acuerdo a la “Federación Internacional de la Industria Fonográfica” (International Federation of the Phonographic Industry, IFPI) [Katzenbeisser00], las marcas de agua en audio deben cumplir con los siguientes requisitos:

- a) La marca de agua no debería degradar la percepción del audio.
- b) El algoritmo debería ofrecer mas de 20 dB de tasa ruido - señal (SNR por sus siglas en Inglés) para el audio marcado en comparación con el audio original.
- c) El algoritmo debería permitir al menos una tasa de transmisión de 20 bps.
- d) La marca de agua debe resistir el procesamiento y ataques más comunes tales como: conversiones D/A y A/D, escalamiento temporal ($\pm 10\%$), corrupción por ruido tanto de naturaleza aditiva como multiplicativa, y compresión mp3.
- e) La marca de agua debería estar diseñada para evitar ser detectada, removida o embebida sin autorización, a menos que la calidad del audio anfitrión se vuelva inaceptable.

Cabe mencionar que cumplir con todos esos requerimientos presenta un gran reto. Sin embargo para el objetivo de este trabajo (que se describirá más adelante), no es necesario cumplir con el último de los requisitos anteriormente mencionados; es decir, evitar que la marca sea detectada sin autorización es completamente irrelevante para la aplicación propuesta. De hecho, para el objetivo de este trabajo es necesario agregar un nuevo requisito a la lista anterior que será: *La marca de agua debe poder ser detectada lo más fácilmente posible.*

1.3. Objetivos de la Tesis

1.3.1. Objetivo general

El objetivo que se persigue con este trabajo es poder enviar información (marca) embebida en una señal de audio que será transmitida por un medio analógico tal como las transmisiones de radio A.M. y F.M.. Esta información deberá servir como un comando de control para manipular equipos en estaciones de radio o televisión.

1.3.2. Objetivos particulares

- Implementar al menos un algoritmo para embeber información en una señal de audio.
- Implementar al menos un algoritmo capaz de extraer la información embebida por un algoritmo de marcado propuesto en este trabajo.
- Lograr que el algoritmo de marcado no degrade de manera perceptible la calidad del audio anfitrión, por lo que deberá ofrecer más de 20 dB de SNR para el audio marcado en comparación con el audio original, de acuerdo con las exigencias de la IFPI.
- Insertar al menos 20 bits, en una señal digital de longitud equivalente a un segundo con los algoritmos propuestos.
- Proponer un método para solucionar el problema de vulnerabilidad al ataque por modificación de amplitud en la señal, así como encontrar un mecanismo para contrarrestar los efectos del escalamiento en tiempo que aparecen tras las conversiones D/A - A/D, que son escasamente tratados en el estado del arte.

1.4. Descripción de Capítulos

En el capítulo 1 se plantea el problema y se describen las necesidades en la industria que inspiraron este trabajo. Se hace también una breve descripción de como se resuelven dichos problemas actualmente, y se justifica el desarrollo propuesto. Se plantea el objetivo de la tesis.

En el capítulo 2, *Esteganografía y Marcas de Agua en Audio*, se establecen los principios en que se sustenta la tesis tales como la naturaleza de las señales de audio digital, la nomenclatura utilizada, y una de las definiciones más ampliamente utilizadas en este trabajo que es la potencia de una señal (secuencia) de audio. Se introducen también en los principios de esteganografía e inserción de marcas de agua aplicados a señales de audio. Se presenta una síntesis del estado del arte, resumiendo un par de algoritmos en los que se basa este trabajo.

El capítulo 3 titulado *Marcas de Agua Robustas, Estrategias de Inserción y Detección de Marcas*, trata sobre un par de estrategias que se idearon debido a los problemas encontrados en los algoritmos mencionados en el capítulo anterior. Se describen a detalle un par de algoritmos con las estrategias propuestas, en los que se toman en cuenta los ataques que puede sufrir una señal marcada.

El capítulo 4 titulado *Experimentos y Resultados*, contiene los experimentos realizados, y las condiciones en que fueron efectuados. Se reportan los resultados de someter la señal marcada a distintos tipos de ataques, tanto de ataques sintéticos, como de ataques en condiciones reales.

Finalmente, en el capítulo 5 se exponen las conclusiones y se enumeran los aspectos en los que se puede mejorar el algoritmo presentado.

Se incluyen 3 apéndices que contienen un breve resumen sobre ciertos tópicos que se consideran relevantes para el tema tratado en esta tesis. En el apéndice A, *Relación Señal - Ruido*, se deduce la fórmula de cálculo adecuada para las aplicaciones aquí tratadas a partir de su definición. En el apéndice B, *Circuito de Muestreo y Retención*, se da un vistazo a la teoría de un circuito muestreador típico, con el fin de aclarar uno de los fenómenos (ataque) que sufre una señal digital marcada al ser transmitida por un canal analógico. Finalmente, en el apéndice C, *Transformada Wavelet* [Daubechies92][Polikar01], se presentan los principios básicos de la transformada wavelet, de su interpretación, y el algoritmo implementado en este trabajo.

Capítulo 2

Esteganografía y Marcas de Agua en Audio

En este capítulo se hace un breve resumen sobre el marco teórico necesario, así como también se describe brevemente el concepto del embebido de información oculta en secuencias de audio. También se presentan de forma sintetizada una selección de algoritmos de inserción y extracción de información en audio, que se consideró que son relevantes y lo suficientemente robustos para los propósitos aquí perseguidos.

2.1. Audio Digital

Las señales pueden describir una amplia variedad de fenómenos físicos y pueden representarse de muchas formas. En todos los casos, la información en una señal está contenida en un patrón de variaciones que presenta alguna forma determinada. Un ejemplo de esto es el sonido.

Desde el punto de vista de la física, el sonido es una onda de presión que viaja por el aire, es posible entonces medir las variaciones de presión durante un lapso de tiempo con algún dispositivo, y almacenar las mediciones para posteriormente reproducir la onda original.

La tecnología actual permite convertir las fluctuaciones de presión acústica en una

señal eléctrica que puede ser transferida a una gran variedad de medios de almacenamiento. En la actualidad los medios de almacenamiento más comunes son los medios digitales, por lo que se hace necesario convertir la señal de audio continua, en una forma adecuada para ser almacenada en una forma digital. Para lograr lo anterior, la forma más comúnmente utilizada es convertir la señal de audio continua, en una señal discreta. Al proceso de convertir una señal continua en función del tiempo a una señal discreta se le llama “Muestreo”. Más información sobre el muestreo y sus implicaciones puede encontrarse en [Oppenheim98].

2.1.1. Notación

En este trabajo, una señal de audio digital (ya sea una melodía completa, un anuncio, etc.) será representada con el símbolo x . En audio digital, a menudo conviene dividir un audio en partes más pequeñas llamadas “cuadros” o “marcos”, que son conjuntos de valores o muestras, donde el tamaño de estos conjuntos será típicamente representado por el símbolo N , y la cantidad de marcos resultantes de la división se representará mediante la letra M . Así mismo, y cuando sea conveniente dividir un audio x , para representar la n -ésima muestra contenida en el m -ésimo marco, se usará la notación: $x_{n,m}$ donde n representa la posición de un elemento (muestra) dentro de un marco, y m representa el número de marco de un total de M en los que fué dividido el audio.

2.1.2. Potencia de un Segmento de Audio

Una de las características de una señal de audio discreta que más se utiliza en este trabajo, es la potencia promedio de dicho audio digital. Las ecuaciones que definen como se calcula esta propiedad se muestran a continuación.

La energía de un segmento de señal discreta (en este caso un marco) se define con la Ecuación (2.1) [Oppenheim98]

$$\varepsilon_m = \sum_{n=0}^{N-1} |x_{n,m}|^2 \quad (2.1)$$

y la potencia promedio de un segmento (marco) de señal discreta se define por la Ecuación (2.2) [Smith07]¹

¹http://ccrma.stanford.edu/~jos/mdft/Signal_Metrics.html

$$P_m = \frac{\varepsilon_m}{N} \quad (2.2)$$

así, sustituyendo la Ecuación (2.1) en (2.2) se obtiene la expresión para el cálculo de la potencia media de un marco.

$$P_m = \frac{1}{N} \sum_{n=0}^{N-1} |x_{n,m}|^2 \quad (2.3)$$

También es posible hacer el cálculo de la potencia media en el dominio de la frecuencia. Para lograrlo basta recordar la relación de Parseval mostrada en la Ecuación(2.4).

$$\sum_{n=0}^{N-1} |x_{n,m}|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X_{k,m}|^2 \quad (2.4)$$

donde $X_{k,m}$ es el valor del k -ésimo coeficiente resultante del cálculo de la *Transformada de Fourier de Tiempo Discreto* (TDF) [Oppenheim98] en el m -ésimo marco del audio analizado. Se puede ver fácilmente que el lado derecho de la Ecuación (2.1), que representa la energía de un ciclo completo de una señal periódica, corresponde perfectamente con el lado izquierdo de la ecuación (2.4). Sustituyendo el lado derecho de de la Ecuación (2.4) en la Ecuación (2.3) se obtiene:

$$P_m = \frac{1}{N^2} \sum_{k=0}^{N-1} |X_{k,m}|^2 \quad (2.5)$$

Así, usando la Ecuación (2.5) se está en posibilidad de calcular la potencia media de un segmento de señal de N muestras, en el dominio de la frecuencia.

2.2. Principios del Insertado de Marcas de Agua en Audio

Ya que se han repasado algunos aspectos básicos de las señales discretas, se puede entrar de lleno al área del embebido de marcas de agua en audio. Éste, como cualquier otro proceso, requiere de un modelo a seguir, y el cual se presenta a continuación.

2.2.1. Modelo General del Proceso de Inserción de Marcas de Agua en Audio

Cvejic en [Cvejic04] presenta un modelo general para la inserción de marcas de agua en una señal de audio digital, el cual se reproduce en la Figura 2.1. En dicha figura se puede ver que un mensaje “*msg*” (que será la marca de agua, y que generalmente será una secuencia binaria que puede ser interpretada a conveniencia) es embebido en el audio anfitrión x , produciendo la señal marcada \hat{x} . El proceso de embebido es dependiente de una clave K , la cual debe ser adecuada para que pueda ser satisfecho el requerimiento de transparencia perceptual; es decir, que la diferencia subjetiva de calidad entre las señales x y \hat{x} definida como *Distorsión de Embebido* y que será denotada como d_{emb} , debe estar apenas debajo del umbral en el que la diferencia se hace sensible. Se puede observar que antes de la detección y decodificación de la marca, la señal \hat{x} generalmente sufre modificaciones las cuales pueden ser intencionales o fortuitas.

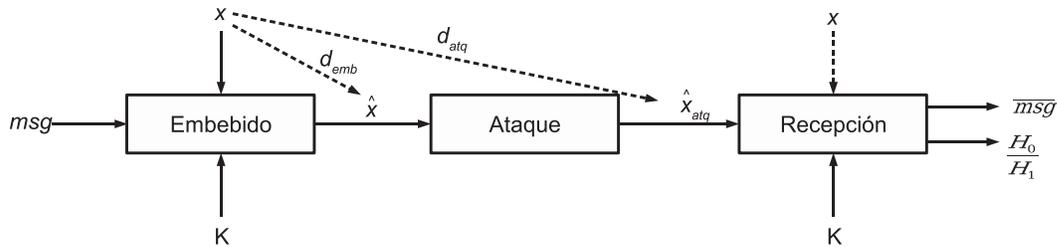


Figura 2.1: Modelo general del proceso de inserción y extracción de marcas de agua en audio

Las modificaciones intencionales son usualmente conocidas como *Ataques*; los cuales, producirán una señal modificada que puede ser representada como \hat{x}_{atq} , la cual ha sufrido una distorsión que será denotada como d_{atq} . En cualquier ataque, siempre se procurará que la distorsión generada, sea perceptualmente aceptable; por lo que en general, el extractor de marca recibirá una señal que ha sufrido algún ataque.

El proceso de extracción de la marca de agua consiste de dos subprocesos, uno es la decodificación de la marca de agua recibida, mediante el que se obtiene un mensaje \overline{msg} por medio de la clave K ; y el otro es la detección de la marca de agua

2.2.2. Concepto del Término Ataque

El término *ataque* requiere ser aclarado un poco más a fondo. Considere una señal de audio marcada \hat{x} , la cual puede ser modificada sin la intención de impactar su contenido oculto. Por ejemplo, modificar el volumen del audio antes de ser transmitido por radio. ¿Por qué a un procesamiento de este tipo se le llama ataque?. La primera razón por la que recibe este nombre, es para simplificar la notación en el modelo general del proceso de inserción de marcas de agua en audio digital. La otra razón, y la más significativa es que cualquier procesamiento de señal puede dañar drásticamente una marca embebida, y puede convertirse en un método para que un atacante, de manera intencional pueda eliminar la información embebida. Es por lo anterior que los algoritmos de inserción de marcas de agua, deben ser diseñados para soportar los peores casos posibles de distorsión d_{atq} , los cuales podrían ser causado por algún procesamiento de señal simple y común, tal como compresión de amplitud (modificación del volumen) o tiempo, filtrado pasabajas, etc.

2.3. Efectos de la Conversión A/D y D/A en Señales Digitales

Como se mencionó anteriormente, para diseñar un algoritmo que soporte los peores casos posibles, hay que conocer cuales son estos casos. Del objetivo principal establecido en este documento, se sabe que la señal marcada digitalmente tendrá que ser transmitida por un medio analógico, por lo que se hace necesario investigar cuales efectos adversos aparecerán durante este proceso, y estar en condiciones de proponer alguna posible solución. Xiang [Xiang06] hace un estudio bastante extenso sobre los efectos de las conversiones A/D y D/A sufridas por una señal marcada con un algoritmo como el propuesto por Wu [Wu04],[Wu05]. En esta sección se presenta un breve resumen sobre los posibles escenarios de transmisión de una señal digital marcada, los efectos sufridos por la señal después de una conversión A/D - D/A, así como un modelo propuesto por Xiang [Xiang06] para explicar este fenómeno.

2.3.1. Medios de Transmisión

En aplicaciones prácticas como la que se pretende desarrollar en este trabajo, la señal marcada puede ser transmitida por varios ambientes. Algunos posibles escenarios son descritos en [Popa98], y se pueden resumir con los casos que se muestran en la figura 2.2.

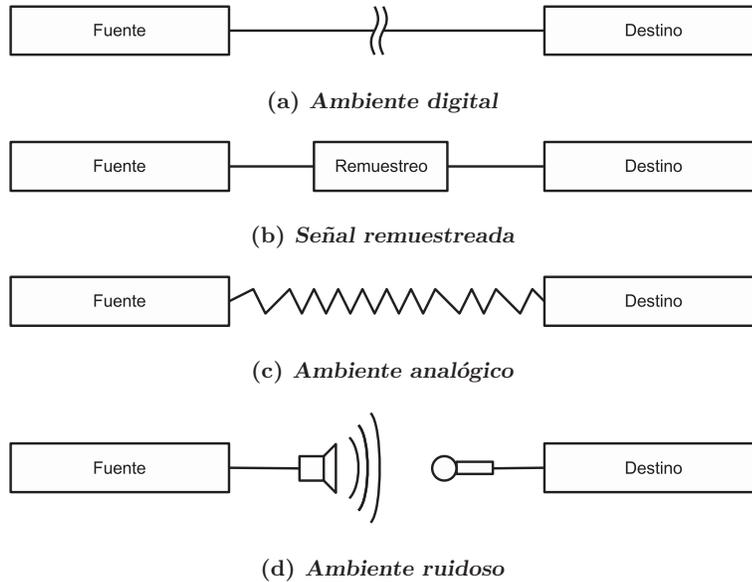


Figura 2.2: Posibles ambientes de transmisión del audio

El primer caso es cuando la señal es transmitida por un ambiente digital, en el cual no sufre ninguna modificación, éste corresponde a la Figura 2.2(a). En la Figura 2.2(b) se puede ver el caso donde la señal es remuestreada con una frecuencia de muestreo inferior o superior a la que fue muestreada originalmente. La amplitud y la fase no sufren cambios, pero sus características en tiempo cambian. El tercer caso mostrado en la Figura 2.2(c) consiste en la conversión y transmisión de la señal en forma analógica. En este caso, aun si la línea es considerada limpia, la amplitud, la fase, y la tasa de muestreo sufren alguna modificación. Por último se considera el caso donde la señal atraviesa por un medio ruidoso, sujeto a transformaciones no lineales los cuales resultan en cambios de fase y amplitud, ecos, etc. Este caso se ve en la Figura 2.2(d). Son estos dos últimos casos los que representan las condiciones que se presentan y se pretenden superar en este trabajo, y que deben ser

considerados más a fondo. Los efectos que aparecen en estos dos últimos casos se describen a continuación.

2.3.2. Distorsión de la Magnitud de la Señal

Esta distorsión, como su nombre lo indica, consiste en la modificación de la amplitud de la señal transmitida, y aparece por el ajuste de volumen tanto en el dispositivo emisor, como en el ajuste de la ganancia de captura ajustada en el dispositivo receptor. El efecto sufrido por esta distorsión, es el cambio de la magnitud de la energía total de la señal de audio, pero sin afectar de manera significativa el contenido de frecuencias original.

2.3.3. Distorsión por Escalamiento en Tiempo

Esta distorsión consiste en la adición o decremento del número de muestras que aparecen después de que la señal digital original ha sido reconstruida y enviada en forma analógica por un canal de transmisión, y es vuelta a muestrear. El dispositivo capturador (en este caso una tarjeta de sonido), difícilmente tendrá exactamente la misma frecuencia de muestreo f_s a la que ha sido muestreada originalmente la señal en análisis, pudiendo tener pequeñas desviaciones del orden de 1 a 10 Hz las cuales implican al menos ± 1 muestras por segundo, de diferencia con la señal original. Esta es la distorsión más nociva para las marcas de agua, pues implica ya sea la pérdida de información, o la adición extra de la misma; y tiene serios efectos adversos en la búsqueda de los códigos de sincronización. En el apéndice B se trata la teoría de un circuito de muestreo y retención² básico que aclara un poco más la causa de esta distorsión.

2.3.4. Corrupción por Ruido

La corrupción por ruido es un problema omnipresente en cualquier sistema de transmisión analógico, ocasionado por una gran cantidad de factores que van desde la naturaleza misma de los componentes electrónicos, radiación electromagnética, temperatura, etc., su principal efecto indeseado en una señal marcada es la adición de energía extra a la misma.

²Un circuito de esta naturaleza siempre estará incluido en cualquier dispositivo que capture señales.

Todo lo anterior se puede resumir en el modelo que se presenta a continuación:

2.3.5. Modelo de Distorsión

Desde el punto de vista del procesamiento digital de señales, las marcas de agua son señales muy débiles embebidas en un medio de mucho mayor potencia. Por lo tanto, cualquier cambio pequeño en la señal marcada, afectará directamente en la supervivencia de la marca. Xiang [Xiang06] propone que el efecto de las conversiones A/D - D/A sea modelado matemáticamente mediante la Ecuación (2.6).

$$x'[i] = \lambda \cdot x \left[\frac{i}{\alpha} \right] + \eta_i \quad (2.6)$$

Donde x es la señal, α denota un factor de escalamiento temporal, y λ es un factor de escalamiento de amplitud, η_i representa ruido gaussiano aditivo en la i -ésima muestra después de las conversiones D/A - A/D. Si $\frac{i}{\alpha}$ no es entero, $\left[\frac{i}{\alpha} \right]$ es interpolado con las muestras más cercanas de la señal de audio.

2.4. Algoritmos de Inserción de Marcas de Agua

Los algoritmos de embebido de Marcas de Agua en audio más recientes, pueden ser agrupados en dos categorías:

1. Algoritmos en el dominio del tiempo como el propuesto por Gruhl en [Gruhl05].
2. Algoritmos en el dominio de la frecuencia.

También existen algunos otros algoritmos como los mostrados en [Hsieh02] y [Mansour05], en los que se proponen métodos que se sustentan en características en el dominio del tiempo, pero aprovechan algunas propiedades en el dominio de la frecuencia.

Actualmente la mayoría de los algoritmos son diseñados para explotar y modificar una o más características en el dominio de la frecuencia, debido principalmente a que resultan ser más robustos que los algoritmos en el dominio del tiempo. Sin embargo, los algoritmos existentes aun presentan debilidades tales como: la baja carga de datos que son capaces de embeber, y poca robustez ante algunos ataques específicos.

De la revisión del estado del arte, de la implementación y experimentación con algunos algoritmos, se eligió un par de éstos para ser usados como base para el trabajo presentado en este documento, debido principalmente a las características de las marcas de agua embebidas por ellos, así como de sus principios de funcionamiento, también se da un vistazo a uno de los problemas más graves a los que se enfrenta la disciplina del embebido de marcas de agua en audio y que se presentaron durante el proceso de investigación.

El primero de dichos algoritmos se presenta a continuación.

2.4.1. Embebido de Información Mediante la Inserción de Tonos Imperceptibles

Gopalan [Gopalan04] propone un método que depende de la inaudibilidad de tonos de muy baja potencia en presencia de otros componentes espectrales de potencia significativamente más grande, por lo que explota de forma indirecta el fenómeno de enmascaramiento psicoacústico en el dominio de la frecuencia [Bosi03].

El algoritmo consiste en la generación y adición de dos tonos a frecuencias f_0 y f_1 y de muy baja potencia a cada uno de los cuadros de la señal de audio anfitrión para embeber un bit de información en cada uno de dichos cuadros. Como se muestra en la Figura 2.3, el audio anfitrión es dividido en segmentos sin traslape (marcos que pueden ser de 512 muestras). Las frecuencias f_0 y f_1 se pueden seleccionar de manera arbitraria. Posteriormente, a cada cuadro de audio se le calcula su potencia media P_m mediante la Ecuación (2.3) en el dominio del tiempo, o bien usando la Ecuación (2.5) en el dominio de la frecuencia. Aunque es posible incluir más de un bit por marco, en este caso se considerará el embebido de únicamente un bit; así, si el bit a embeber es un 0 lógico, al marco se le suma un tono de frecuencia f_0 cuya potencia P_0 deberá ser igual a un 0.25 % de P_m , al mismo tiempo se le suma otro tono de frecuencia f_1 con potencia P_1 igual al 0.001 de la potencia del tono de frecuencia f_0 (P_0). Para embeber un 1 lógico, el procedimiento es similar al anterior, y se puede ver resumido en la Figura 2.3.

En este caso, el procedimiento de extracción del mensajes es bastante simple. Para lograr la recuperación de información contenida, a cada cuadro de audio le es calculada la transformada discreta de Fourier, se calcula su potencia promedio usando la Ecuación 2.5, al

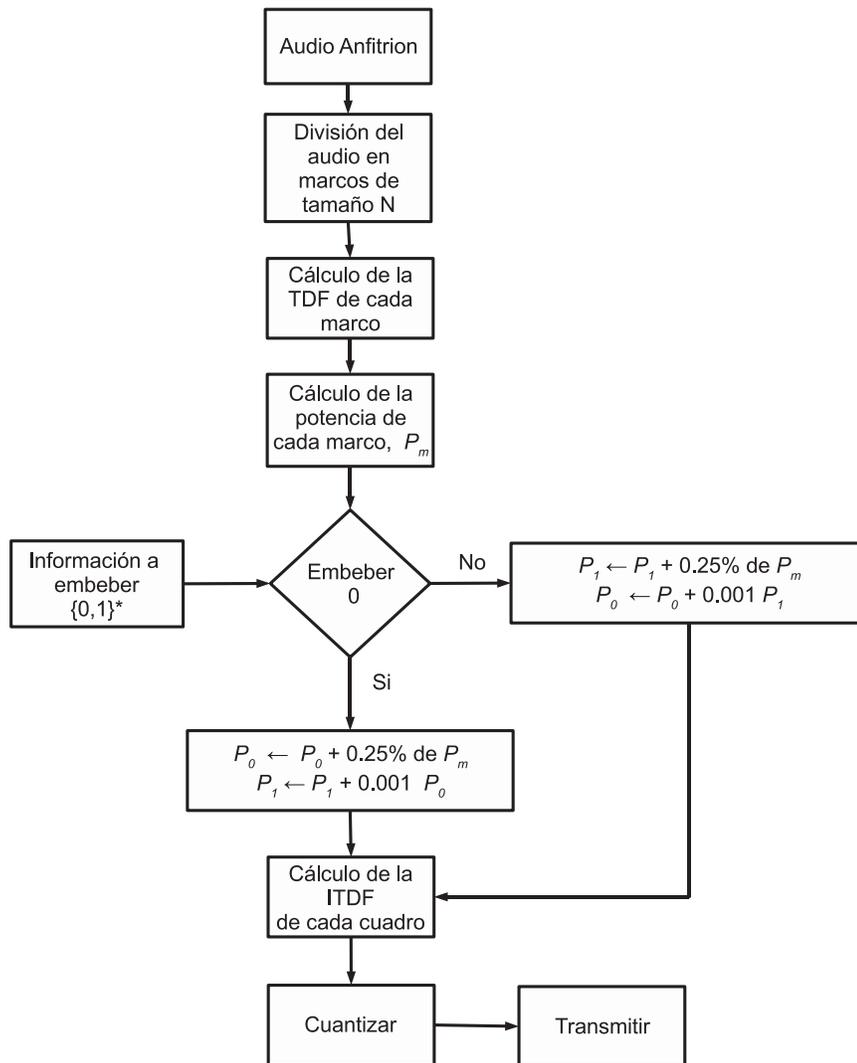


Figura 2.3: Algoritmo de Inserción de Tonos Propuesto por Gopalan [Gopalan04]

mismo tiempo que es determinada P_0 y P_1 en f_0 y f_1 respectivamente. Si la razón mostrada en la Ecuación (2.7) se cumple, entonces el bit es declarado como 0; de lo contrario el bit embebido en el cuadro es considerado 1.

$$\frac{P_m}{P_0} > \frac{P_m}{P_1} \quad (2.7)$$

En el proceso de búsqueda de algoritmos adecuados para embeber información en audio, este método se implementó y se realizaron varios experimentos con él, observando que una de las cualidades más importantes de la marca de agua insertada, es ser *robusta ante ataques de escalamiento de amplitud de la señal marcada*. También se observó que por la naturaleza del proceso de detección, también resulta ser robusto antes ataques de ruido, y recuantización.

Sin embargo, en este algoritmo, tal como lo presenta su autor, no se hace uso del concepto de una clave K , vista en la Sección 2.2.1, y asume que la ubicación³ de la marca es conocida. Es obvio que con simple ataque de desplazamiento en tiempo de la señal, la ubicación se pierde provocando que sea casi imposible recuperar la información.

Esto evidencia la necesidad de usar alguna estrategia para superar el problema de la pérdida de la ubicación de la información embebida. La respuesta más lógica a lo anterior, es hacer uso de algún tipo de marca, que pueda ser encontrada a pesar de de ser desplazada de su lugar original. Un resumen sobre algunas posibilidades para implementar estas marcas se presenta a continuación.

2.4.2. Códigos de Sincronización

En varios de los algoritmos existentes en la literatura [Lee00][Gopalan04][Gopalan05], se asume que la ubicación de la marca es conocida, por lo que se limitan a proporcionar la metodología de embebido; sin embargo, sin importar cuan robusta sea la marca ante diversos tipos de ataques, basta con hacer un simple corrimiento en tiempo, o un recorte de unas cuantas muestras de la señal marcada, para que se vuelva casi imposible extraer la información oculta.

³Número de muestra en una señal de audio digital a partir de donde se comienzan el proceso de extracción de la información.

Una estrategia comúnmente utilizada para superar este problema, es la inclusión de códigos de sincronización, los cuales indicarán la ubicación del inicio de la marca. Tales códigos tienen que ser diseñados para ser robustos a ataques como el desplazamiento, o el recorte (eliminación) de segmentos de audio.

Por ejemplo, Chung P. Wu et al. [Wu99] propone una definición de *Puntos Salientes*, los cuales son un tipo características de energía, y consiste en la búsqueda de segmentos de audio de “Rápido Crecimiento”. Esta definición satisface los requerimientos de una *Etiqueta de Sincronización*, y se realiza en el dominio del tiempo.

Este método fue implementado, y de la realización de varios experimentos se encontró que aunque la etiqueta de sincronización es robusta en algunos sentidos, también resultó ser tremendamente vulnerable a los ataques de adición de un componente de CD, lo cual es muy común en canales de transmisión analógica y que inevitablemente sufrirá la señal marcada. Esta es la principal razón por la que se abandonó el uso de esta técnica. Cabe mencionar que con este método, ante la presencia de algún ataque, cabe la posibilidad de que la etiqueta de sincronización sobreviva pero la marca no, o bien, la sincronización sea perdida aunque la información no sufra ninguna alteración. Este último caso implica la imposibilidad de decodificar la información, por lo que el total de la marca es perdida, concluyéndose también, que este método no es adecuado para el objetivo que se pretende alcanzar.

Otro método más comúnmente usado, consiste en la inclusión de una secuencia binaria conocida, en la misma marca a insertar, y que servirá de código (o etiqueta) de sincronización. La principal ventaja será que la etiqueta de sincronización tendrá la misma naturaleza que la marca, lo cual obviamente incrementa las posibilidades de detección de la información. Aunque por otro lado, este tipo de enfoques resulta ser más costoso computacionalmente hablando. Sin embargo, debido a su principal ventaja, este método será utilizado en este trabajo.

En este enfoque se vuelve necesario analizar absolutamente todo el audio en el que se pretende encontrar una marca. El audio deberá ser dividido en M partes, y de cada uno de dichos segmentos se determinará un bit⁴. Con lo anterior se obtendrá una secuencia

⁴El valor de dicho bit se determina según la regla de extracción específica del algoritmo usado.

binaria $A = \{1, 0\}^*$ de tamaño M . Una vez hecho ésto, se procederá a la búsqueda de una secuencia de bits K de tamaño L (donde $K \subset A$) que funge como etiqueta de sincronización dentro de la secuencia A . Una vez localizado el código de sincronización K , se determina que los bits subsecuentes corresponden a la información embebida.

Es precisamente este método el que se usa en el algoritmo que se presenta a continuación.

2.4.3. Marcas de Agua en el Dominio de la Transformada Wavelet Discreta

Una herramienta que está tomando cada vez más relevancia en el área del marcaje de agua es la *Transformada Wavelet Discreta* (TWD) [Daubechies92][Polikar01]; pues gracias a su característica de localización tiempo-frecuencia, resulta de gran ayuda en la reducción de carga computacional en la búsqueda de los códigos de sincronización, además de que permite crear la marca en el dominio de la frecuencia.

Shaoquan Wu et al. [Wu04][Wu05] propone un algoritmo basado en la modificación de coeficientes de baja frecuencia obtenidos a través de la transformada wavelet discreta. Este algoritmo utiliza códigos de sincronización incluidos dentro de la misma marca en forma de una secuencia PN⁵ que es una secuencia binaria aparentemente aleatoria, pero que es perfectamente determinística. Tal como se mencionó anteriormente, al hacer uso de las características de localización tiempo-frecuencia de la transformada wavelet, se logra un gran ahorro computacional en la búsqueda de la etiqueta de sincronización.

La Figura 2.4 muestra el esquema general de marcado propuesto por Wu [Wu04][Wu05]. En dicha figura se puede ver que el primer paso consiste en la división del audio a ser marcado en pequeños segmentos (marcos). Posteriormente, a cada uno de los marcos a modificar se le calcula la transformada wavelet discreta. El siguiente paso consiste en el proceso de embebido de la marca, la cual fue generada de manera paralela mediante la concatenación de dos secuencias binarias; una que hará las veces de código de sincronización, y la otra secuencia será la información que se desea transmitir. El embebido consiste en la

⁵Para el enfoque manejado en este documento, la secuencia PN, hace las veces de la clave K mencionada en la Sección 2.2.1

modificación de un coeficiente de una sub-banda de baja frecuencia mediante la regla de embebido de la expresión (2.8)

$$\hat{c}_{n,m} = \begin{cases} \lfloor \frac{c_{n,m}}{S} \rfloor S + 3\frac{S}{4} & \text{si } a_m = 1 \\ \lfloor \frac{c_{n,m}}{S} \rfloor S + \frac{S}{4} & \text{si } a_m = 0 \end{cases} \quad (2.8)$$

donde $c_{n,m}$ y $\hat{c}_{n,m}$ son los n -ésimos coeficientes correspondientes a una sub-banda de baja frecuencia obtenidos a partir del cálculo de la TWD de los m -ésimos marcos del audio original, y del audio marcado respectivamente. S es el *Índice de Intensidad de Marcado*, cuyo valor debe ser lo más grande posible, pero sin que se llegue a afectar la calidad perceptible del audio; y finalmente, a_m es el m -ésimo bit de la secuencia binaria a embeber. El último paso consiste en calcular la *Transformada Wavelet Discreta Inversa* (ITWD) para regresar la señal del dominio de la transformada wavelet, al dominio del tiempo. El resultado será el audio con la marca de agua insertada.

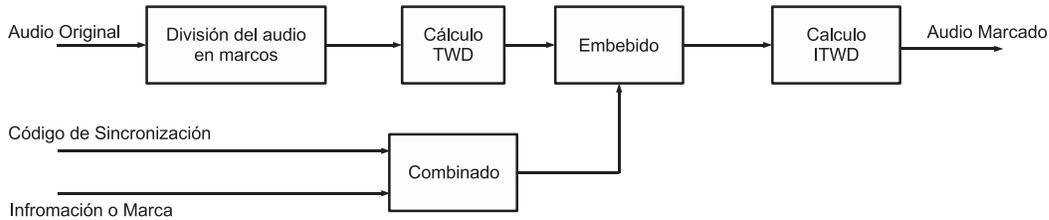


Figura 2.4: Esquema de marcado del algoritmo de Wu basado en transformada wavelet.

En la Figura 2.5 se puede ver el esquema de extracción de marca. Nótese la naturaleza iterativa del algoritmo para la búsqueda del código de sincronización. En dicha figura se puede ver que nuevamente el primer paso consiste en la división del audio que se piensa contiene información en marcos de tamaño N , posteriormente para todos y cada uno de los marcos resultantes les es calculada la TWD. De cada uno de los marcos transformados, es determinado un bit en base a una regla de extracción (que será vista más a fondo en el siguiente capítulo) y del análisis del mismo coeficiente que “pudo” haber sido modificado en el proceso de embebido, obteniendo una gran secuencia binaria de tamaño M . El siguiente paso consiste en la búsqueda de un patrón de bits (que es precisamente el código de sin-

cronización) en toda la secuencia binaria obtenida del paso anterior. De la búsqueda son posibles dos resultados.

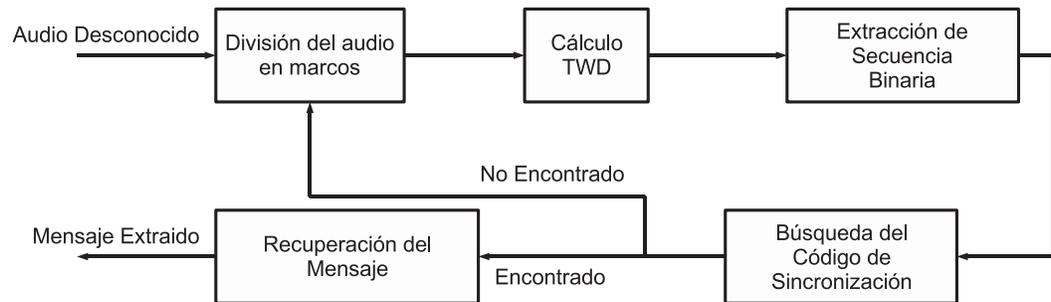


Figura 2.5: Esquema de extracción de marca del algoritmo de Wu basado en transformada wavelet.

El primer caso es cuando no se encontró el código de sincronización en la secuencia binaria, en tal situación se regresa al paso de la división del audio, pero con una gran diferencia. La primer muestra del audio completo deberá ser eliminada (quedando una señal de audio distinta a la original) y se realiza la división del audio tal y como se hizo la primera vez. Por cada iteración en la que el código de sincronización no es encontrado, la muestra que está al inicio del audio analizado deberá ser eliminada hasta completar un número de iteraciones igual al tamaño del marco (N). En caso de realizar estas N iteraciones sin haber encontrado el código de sincronización, se declara que en el audio analizado no existe la marca buscada.

El segundo caso posible de la búsqueda del código de sincronización, es que éste haya sido localizado, en cuyo caso se asume que los bits subsecuentes al final de la secuencia binaria que conforma el código de sincronización, son la información deseada, y deberá ser interpretada como tal.

De la implementación y la realización de varios experimentos, se observó que las marcas agregadas con este algoritmo, demostraron ser bastante robustas ante ataques y procesamientos de señal común como: adición de ruido blanco, recuantización, remuestreo⁶, compresión mp3; y gracias a la inclusión de códigos de sincronización y redundancia de la

⁶La marca sobrevive a pesar del remuestreo, pero sólo es posible recuperarla, si la señal es regresada a su frecuencia de muestreo original.

marca, la información sobrevive ante ataques de corrimientos de señal en tiempo y recortes de la misma.

Sin embargo, también se pudo comprobar que la información embebida por este algoritmo se vuelve prácticamente imposible de localizar si dicha señal es transmitida a través de un canal analógico.

Ya que las marcas embebidas mediante este algoritmo cumplen con la mayoría de los requerimientos establecidos en los objetivos de este trabajo, y que el algoritmo es adecuado para ser usado en una aplicación en tiempo real, se eligió trabajar sobre él como base, tomando en cuenta el conocimiento adquirido sobre los efectos de las conversiones D/A - A/D.

2.5. Conclusiones del Capítulo

Por lo expuesto en este capítulo, y los experimentos realizados se pueden rescatar algunas ideas y concluir varios aspectos importantes.

Una idea importante a tener en cuenta es que cualquier procesamiento o condición (de transmisión, almacenamiento, etc.) que pudiese afectar la calidad de una marca de agua será denotado como “*ataque*”, aun y cuando el daño que pudiese sufrir la marca no sea infringido intencionalmente.

Aunque el algoritmo propuesto por Gopalan [Gopalan04] presentó muchas deficiencias, y se concluyó que no era adecuado para los fines que se persiguen con este trabajo, se demostró por experimentación que la marca embebida por este método, supera uno de los ataques que Xiang en [Xiang06] considera más difíciles de superar, que es el ataque de modificación de amplitud.

De los experimentos realizados con los dos algoritmos descritos en este capítulo, se puede concluir que los primeros dos casos de transmisión presentados en la sección 2.3.1 (que son el caso de transmisión de la señal por un medio digital, y el caso donde la señal sufre un remuestreo), no representan ninguna amenaza a la integridad de la marca, así que no serán analizados más allá de lo que ahí se menciona. Sin embargo los últimos dos casos descritos en esa sección, son los que representan el reto a superar para cumplir con el objetivo fijado.

Una de las ideas más importantes vertida en este capítulo es la necesidad de la inclusión de un mecanismo para localizar la información embebida en la señal de audio, sobre todo para la aplicación que se propone en este documento, donde el audio fluye constantemente y por obvias razones, el audio sufre múltiples desplazamientos.

Por la naturaleza de la aplicación aquí propuesta, en definitiva, la señal marcada tendrá que sufrir múltiples ataques entre ellos las conversiones D/A - A/D, por lo que dichos ataques deberán ser tomados en cuenta para proponer estrategias que superen estos problemas.

Capítulo 3

Marcas de Agua Robustas, Estrategias de Inserción y Detección de Marcas

En el capítulo anterior se mencionaron las causas por las que una marca de agua embebida en una señal de audio sufre una degradación cuando dicha señal de audio es transmitida por un medio analógico. Tales causas se pueden resumir en el modelo de la ecuación (2.6) el cual indica que la señal sufre de un escalamiento temporal, un escalamiento en amplitud, y ruido gaussiano aditivo; además habría que agregarle un desplazamiento en tiempo a dicho modelo, el cual es causado por la diferencia de tiempo al momento de iniciar la captura y la reproducción del sonido.

El algoritmo que Wu presenta en [Wu04][Wu05], ya resuelve el problema del desplazamiento haciendo uso de una búsqueda exhaustiva por todo el audio, y resuelve también el problema del ruido gaussiano aditivo mediante la modificación de los coeficientes de baja frecuencia resultantes del cálculo de la TWD del audio anfitrión; esto es porque se apostará a que la mayor parte de la energía del ruido no se agregará en las bandas de baja frecuencia.

Sin embargo los problemas causados por los escalamientos en amplitud y tiempo, quedan sin resolver, y es por eso que en este capítulo se describe a detalle una estrategia

para mejorar el algoritmo de Wu y lograr que las marcas embebidas mediante este algoritmo sean robustas al escalamiento en amplitud.

Se presenta también una estrategia para resolver el problema del escalamiento en tiempo, la cual consiste en caracterizar el dispositivo capturador para conocer el efecto de escalamiento de tiempo inflingido por dicho dispositivo, y poder revertirlo.

También se presenta una segunda propuesta que consiste en la combinación de la lógica de embebido propuesta por Gopalan [Gopalan04], y la estructura del algoritmo propuesto por Wu [Wu04] [Wu05]. Con dicha combinación se pretende superar también los problemas anteriormente descritos.

3.1. Caracterización del Dispositivo Capturador para Contrarrestar el Escalamiento en Tiempo

Una de las ventajas más importantes presentada por el algoritmo basado en la modificación de coeficientes de baja frecuencia resultantes del cálculo de la TWD, es que la marca embebida es robusta al remuestreo.

Al decir “robusta” se entiende que la marca sobrevive al remuestreo, pero no necesariamente quiere decir que pueda ser localizada y decodificada correctamente. Para que dicha información pueda ser extraída, se hace necesario remuestrear nuevamente la señal escalada en tiempo de tal manera que sus características en el tiempo sean iguales que en el momento de haber sido marcada.

De la comparación entre la grabación de varias secuencias de audio con sus versiones originales, se comprobó que la tarjeta de sonido utilizada, no muestreaba la señal de audio analógico a la frecuencia ajustada por software, lo que explicó el porqué no era posible extraer la información de un audio marcado que sufrió una conversión D/A - A/D. Una respuesta lógica para resolver este problema, sería conocer el nivel de distorsión provocado por el dispositivo capturador y revertirlo. En otras palabras, se trata de caracterizar el comportamiento de una tarjeta de sonido para saber que frecuencia de muestreo puede llevar a cabo, para que las señales capturadas sean procesadas para que tengan la misma frecuencia de muestreo que en el momento de ser marcadas.

En esta sección se propone un método para caracterizar el dispositivo capturador¹ en la cual se determinará la desviación existente entre la frecuencia de muestreo a la cual se ajusta la tarjeta de sonido (que para los experimentos realizados será de $f_s = 44100Hz$) y la frecuencia de muestreo real que la tarjeta de sonido puede generar, así como el procedimiento para corregir esta desviación.

3.1.1. Remuestreo Digital

El remuestreo digital puede ser definido de forma simple como el procesamiento digital que cambia una señal muestreada a cierto periodo de muestreo, en otra señal muestreada con un periodo de muestreo distinto al anterior.

Dicho en otras palabras; considérese el proceso donde una señal continua y_c ha sido muestreada a una frecuencia $f_{old} = 1/T_{old}$, y las muestras discretas son $x_{old}[n] = y_c(nT_{old})$. El remuestreo digital es necesario cuando se desee una señal $x_{new}[n] = y_c(nT_{new})$, y el muestreo directo de la señal continua y_c a la frecuencia $f_{new} = 1/T_{new}$ no es posible [Lyons97].

Existen dos formas de remuestreo principales, que son el sub-muestreo y el sobremuestreo. El sub-muestreo consiste en incrementar el periodo de muestreo (bajar la frecuencia de muestreo) de la señal original, mientras que el sobremuestreo significa incrementar la frecuencia de muestreo, obviamente decrementando el periodo de muestreo.

De manera práctica, el submuestreo se realiza con un factor D reteniendo cada D -ava muestra de una señal discreta, y eliminando el resto. A este proceso se le llama decimación. El cálculo de la nueva frecuencia de muestreo con relación a la frecuencia anterior se hace con la ecuación (3.1).

$$f_{new} = \frac{f_{old}}{D} \quad (3.1)$$

Por otra parte, el sobremuestreo es un poco más complicado que el caso anterior, pues implica interpolación e inserción de datos. De forma general, el sobremuestreo se hace con un factor U , donde entre cada muestra de la señal a sobremuestrear se insertan u ceros

¹Aunque en el desarrollo de este trabajo, se utilizó como dispositivo capturador una tarjeta de sonido común, otra posibilidad es el uso de un DSP.

(proceso comunmente llamado zero padding)[Lyons97], para posteriormente hacer pasar dicha señal por un filtro pasabajas que hará el proceso de interpolación.

Sin embargo, como puede verse, parecería que tanto el submuestreo como el submuestreo solo pueden llevarse a cabo mediante factores enteros, pero afortunadamente esto no es así.

De forma general, el proceso de remuestreo digital para factores no enteros se puede representar por un factor I mediante la ecuación (3.2):

$$I = \frac{U}{D} \quad (3.2)$$

donde U representa un factor de interpolación, D un factor de decimación, donde $(D, U) \in \mathbb{N}$.

Por ejemplo, si se desea incrementar la frecuencia de muestreo de una señal con factor de 7.125, se puede realizar con una interpolación con un factor $U = 57$, seguida de una decimación con factor de $D = 8$; es decir que $7.125 = 57/8$.

El efecto de muestrear una señal a dos frecuencias ligeramente distintas puede verse en la Figura 3.1. En dicha Figura se puede observar que si ambas señales muestreadas a frecuencias distintas son graficadas juntas, se verá que una es más angosta que la otra.

La señal representada por la línea continua ha sido muestreada con una frecuencia $f_s = 1/T_s$, mientras que la otra curva representa la misma señal pero cuya frecuencia de muestreo $f_u = 1/T_u$ es ligeramente distinta y donde $f_s > f_u$. Si se desea que la señal representada por la curva b) sea aproximada a la curva a), entonces es necesario remuestrear dicha curva b) con algún factor de escalamiento.

Encontrar ese factor de escalamiento es exactamente el problema que se plantea en este trabajo.

Planteamiento

Supóngase que se genera una señal senoidal discreta x_s de frecuencia f_A con un periodo de muestreo de $T_s = 1/f_s$. Dicha señal x_s es sometida a una conversión D/A y transmitida por un canal analógico. Posteriormente la señal es capturada con una tarjeta de sonido convirtiéndose en x_u . Como se mencionó anteriormente, se sabe que casi cualquier

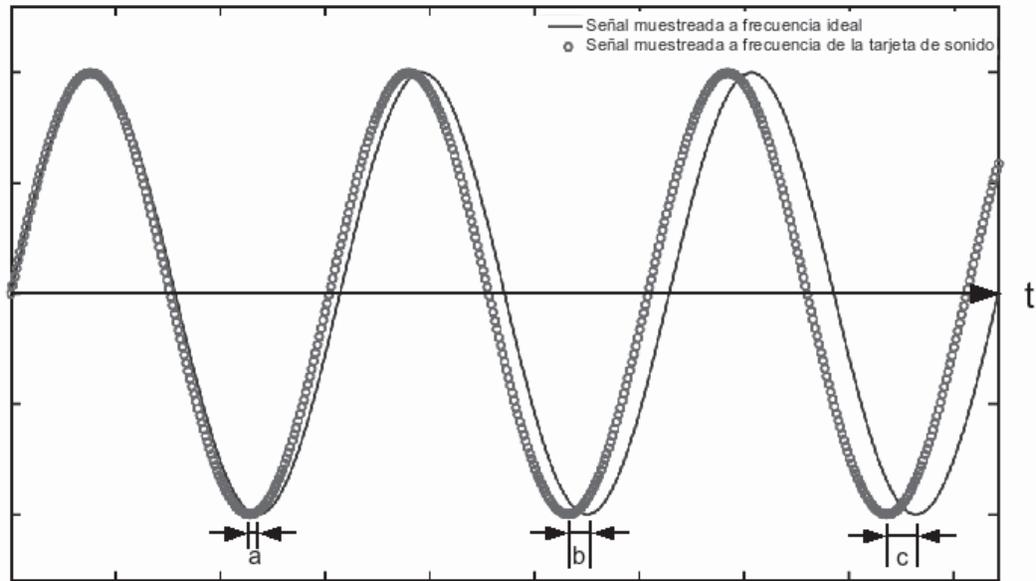


Figura 3.1: Una señal muestreada a dos diferentes frecuencias.

tarjeta de sonido tiene una frecuencia de muestreo $f_u \approx f_s$; es decir, que su frecuencia de muestreo real, es ligeramente distinta a la frecuencia de muestreo elegida por el usuario, y típicamente tendrá una desviación de algunos Hertz hacia arriba o abajo de lo deseado²; y se sabe también que esto depende casi completamente de la construcción física de dicho dispositivo capturador.

Si se asume que dicha desviación es constante y perdura a través del tiempo, entonces debe existir un factor de escalamiento θ que aplicado a x_u , minimice la diferencia (que se puede ver gráficamente en la Figura 3.1) entre las señales x_s y x_u .

3.1.2. Búsqueda del Factor de Escalamiento en Tiempo Causado por el Dispositivo Capturador

Del modelo de distorsión por conversiones A/D-D/A de la ecuación (2.6) se entiende que existe un factor $\beta = 1/\alpha$ que cambia las propiedades de la señal marcada, y un factor $\theta = 1/\beta$ que debería corregir la distorsión por escalamiento de tiempo. Para encontrar ese

²Para fines de esta investigación, se usan sólo audios muestreados teóricamente a 44100 Hz.

factor θ , se propone una minimización de la suma de diferencias cuadradas E , entre una señal ideal y_s de referencia y una señal distorsionada capturada y_u . Para llevar a cabo la minimización se utiliza el método de Levenberg-Marquard [Nocedal99].

Como se mencionó anteriormente, el remuestreo digital se realiza mediante un filtrado (para interpolar), y una decimación. Matemáticamente hablando, la decimación se define con la Ecuación (3.3), donde x' es la señal decimada, x es la señal original, y $\theta \in \mathbb{N}$ es el factor de decimación. Pero si θ es un número real positivo, habrá la necesidad de interpolar con las muestras más cercanas. En el desarrollo del presente trabajo se utiliza interpolación lineal, la cual actúa como un filtro pasabajas de banda limitada [Smith02].

$$x'_n = x_{\theta n} \quad (3.3)$$

Para el fin que se pretende alcanzar, se propone la siguiente transformación de coordenadas:

$$n = \theta_0 i + \theta_1 \quad (3.4)$$

Donde n es el índice de la señal transformada, θ_0 es el factor de escalamiento, i es el índice de la señal original (en este caso de la señal capturada), y θ_1 es un desplazamiento.

E puede ser expresado como se muestra en la ecuación (3.5)

$$E(y(\theta), y_s) = \sum_{\forall \langle i \rangle \in y_s} (y(n(\theta, i)) - y_s(i))^2 \quad (3.5)$$

donde el error entre muestra y muestra se define como

$$e_i = y(n(\theta, i)) - y_s(i) \quad (3.6)$$

Tal como el método de minimización lo exige, es necesario calcular la derivada de E

$$\frac{\partial E}{\partial \theta_i} = 2 \sum_{\forall \langle i \rangle \in y_s} e_i \frac{\partial e_i}{\partial \theta_i} \quad (3.7)$$

y donde

$$\frac{\partial e_i}{\partial \theta_i} = \frac{\partial y(n(\theta, i))}{\partial \theta_i} \quad (3.8)$$

y por regla de la cadena

$$\frac{\partial e_i}{\partial \theta_i} = \frac{\partial y(n(\theta, i))}{\partial n(\theta, i)} \frac{\partial n(\theta, i)}{\partial \theta_i} \quad (3.9)$$

Por lo que de la ecuación (3.4) es posible calcular

$$\frac{\partial n(\theta, i)}{\partial \theta_0} = i \quad \text{y} \quad \frac{\partial n(\theta, i)}{\partial \theta_1} = 1 \quad (3.10)$$

Así, el jacobiano queda definido como:

$$J = \begin{pmatrix} i \\ 1 \end{pmatrix}^T \quad (3.11)$$

Haciendo un cambio de nomenclatura, donde $n(\theta, i) = \bar{n}$, y para calcular la otra parte de la ecuación (3.9)

$$y(\bar{n}) = y_n(i) \quad (3.12)$$

por lo que:

$$\frac{\partial y(n)}{\partial n} = \frac{\partial y_n(i)}{\partial n} \quad (3.13)$$

y nuevamente por regla de la cadena

$$\frac{\partial y_n(i)}{\partial n} = \frac{\partial y_n(i)}{\partial i} \frac{\partial i}{\partial n} \quad (3.14)$$

De la ecuación (3.4) se despeja i quedando

$$i = \frac{1}{\theta_0} n - \frac{\theta_1}{\theta_0} \quad (3.15)$$

lo cual deja en posibilidad de calcular directamente:

$$\frac{\partial i}{\partial n} = \frac{1}{\theta_0} \quad (3.16)$$

Con todos los elementos anteriores, es posible generar el gradiente el cual se ve en la ecuación (3.17)

$$\nabla E(\theta) = 2 \sum_{\langle i \rangle} e_i \binom{i}{1} \frac{1}{\theta_0} \frac{\partial y_n(i)}{\partial i} \quad (3.17)$$

donde el término $\partial y_n(i)/\partial i$ se calcula directamente de la señal usando el método llamado “derivada de filtros gaussianos” usando un factor $\sigma = 1$.

El Hessiano queda entonces definido como:

$$H(\theta) = 2 \sum_{\langle i \rangle} \begin{pmatrix} i^2 & i \\ i & 1 \end{pmatrix} \frac{1}{\theta_0^2} \left(\frac{\partial y_n(i)}{\partial i} \right)^2 \quad (3.18)$$

3.2. Algoritmo de Wu con Normalización de Potencia

En esta sección se presenta una modificación del algoritmo propuesto por Wu en [Wu04][Wu05], al cual se le agrega una estrategia para reforzar la marca contra cambios en la energía de la señal marcada.

3.2.1. Procedimiento de Insertado

La idea principal del algoritmo de marcado, es el de segmentar la secuencia de audio anfitrión x , en pequeñas partes (cuadros), para insertar un bit de la información a embeber en cada uno de éstos. Esto se logrará mediante la modificación de los coeficientes de baja frecuencia, obtenidos a partir del cálculo de la TWD de cada cuadro, obteniendo así una nueva secuencia de audio \hat{x} . Lo anterior se puede observar en la figura 3.2, y la explicación detallada de cada paso se presenta a continuación.

3.2.2. División del Audio en Marcos

Como ya se mencionó anteriormente, el audio anfitrión completo es denotando con el símbolo y y puede ser representado mediante la ecuación (3.19)

$$y = \{x_0, x_1, \dots, x_M, x_{final}\} \quad (3.19)$$

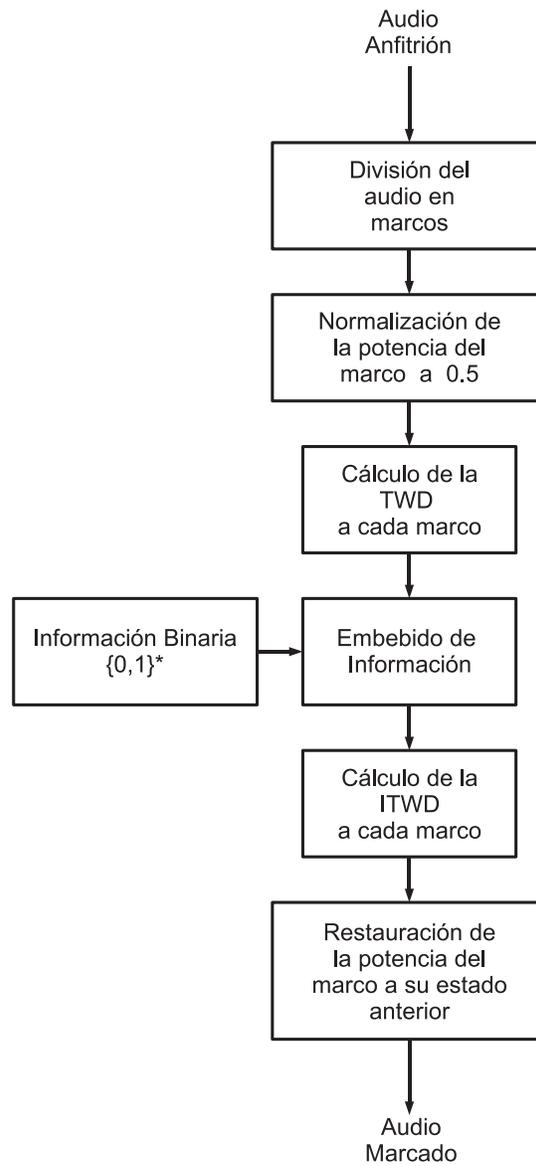


Figura 3.2: Esquema de Embebido del algoritmo con normalización de potencia.

donde cada elemento x_i del conjunto es un *cuadro* o *marco* (subconjunto de datos) de tamaño N y que será representado por la ecuación (3.20)

$$x_i = \{x_{0,i}, x_{1,i}, \dots, x_{N-1,i}\} \quad (3.20)$$

mientras que el subconjunto x_{final} tendrá un tamaño variable L donde $0 \leq L < N$. Dicho subconjunto generalmente será ignorado, y pasará intacto del audio original y al audio marcado \hat{y} .

Para los fines prácticos del algoritmo propuesto, el audio será segmentado en marcos de tamaño $N = 256$.

3.2.3. Códigos de Sincronización a Insertar

Como se mencionó en la sección 2.4.2, es necesario el uso de un mecanismo para poder localizar la posición de los bits que componen la información oculta. Este algoritmo hace uso de esta estrategia usando una secuencia PN.

Supóngase que $\{a_i\}$ es la secuencia binaria original, que se utiliza como código de sincronización, y $\{b_i\}$ es una secuencia binaria desconocida del mismo tamaño que $\{a_i\}$. Si el número de bits diferentes entre $\{a_i\}$ y $\{b_i\}$ es menor o igual a un umbral predefinido e , entonces se determinará que $\{b_i\}$ es el código de sincronización buscado. Un análisis de la probabilidad de error en la búsqueda de los códigos de sincronización se da en [Wu05].

Para la implementación del algoritmo, en este trabajo se utilizó la secuencia binaria que corresponde a los códigos ASCII de la cadena “Msg:Msg:” que es un código de 64 bits.

3.2.4. Datos a Embeber

Para embeber la información, antes que nada será necesario concatenar la secuencia correspondiente al código de sincronización, seguida de la secuencia binaria correspondiente a la marca o información a insertar. La secuencia resultante de dicha concatenación será denotada como m . Con el fin de ganar redundancia en la marca, y como medida para contrarrestar un posible ataque de eliminación de segmentos, es conveniente insertar varias veces m en el audio anfitrión, armando una secuencia como la que se muestra en la figura 3.3, es decir, $\{m\}^*$.



Figura 3.3: Estructura de la secuencia a insertar

3.2.5. Normalización de la Potencia de un Marco

Se ha hablado bastante sobre el hecho de que una señal esta supeditada a sufrir cambios en su amplitud, lo cual obviamente afecta la energía se dicha señal. Xiang [Xiang06] analiza más a fondo el efecto de la modificación de amplitud, y afirma que si se cumple la relación de la ecuación (3.21), entonces no será posible la extracción correcta del bit.

$$(\delta - 1) \cdot c'_i > kS + \frac{S}{4} \quad (3.21)$$

Donde δ es un factor de cambio de amplitud y $k = \lfloor (\delta - 1) \cdot c'_i / S + 0.25 \rfloor$. Como puede verse por simple inspección, la tolerancia al cambio de amplitud es prácticamente nula.

La estrategia propuesta, consiste en normalizar la potencia de cada uno de los marcos a un valor predefinido, previo al embebido de la información. Una vez modificados los coeficientes adecuados, el marco es regresado a su potencia original.

Esto nos dará la información necesaria para aproximar la señal que ha sufrido una modificación a la señal original marcada, y estar en condiciones de extraer el mensaje libre de errores.

Sean x y \bar{x} dos secuencias de audio similares, cuya única diferencia es el nivel de volumen, y cuyas potencias son P_x y $P_{\bar{x}}$ respectivamente; además dichas potencias están relacionadas mediante la ecuación (3.22)

$$P_{\bar{x}} = \lambda P_x \quad (3.22)$$

donde λ es una constante y representa un factor de amplitud. Dado que la potencia del audio x se puede calcular usando la ecuación (2.3), ésta se puede combinar con la ecuación (3.22) obteniendo:

$$\begin{aligned}
P_{\bar{x}} &= \frac{\lambda}{N} \sum_{n=0}^{N-1} |x_{n,m}|^2 \\
&= \frac{\lambda}{N} |x_{0,m}|^2 + \frac{\lambda}{N} |x_{1,m}|^2 + \cdots + \frac{\lambda}{N} |x_{N-1,m}|^2
\end{aligned} \tag{3.23}$$

por otro lado $P_{\bar{x}}$ también puede ser expresado como:

$$\begin{aligned}
P_{\bar{x}} &= \frac{1}{N} \sum_{n=0}^{N-1} |\bar{x}_n|^2 \\
&= \frac{1}{N} |\bar{x}_0|^2 + \frac{1}{N} |\bar{x}_1|^2 + \cdots + \frac{1}{N} |\bar{x}_{N-1}|^2
\end{aligned} \tag{3.24}$$

de esta manera, si se iguala la parte derecha de (3.23) y la parte derecha de (3.24)

$$\frac{1}{N} |\bar{x}_0|^2 + \frac{1}{N} |\bar{x}_1|^2 + \cdots + \frac{1}{N} |\bar{x}_{N-1}|^2 = \frac{\lambda}{N} |x_0|^2 + \frac{\lambda}{N} |x_1|^2 + \cdots + \frac{\lambda}{N} |x_{N-1}|^2 \tag{3.25}$$

y dado que X y \bar{X} son el mismo audio, pero con distinto volumen, para que dicha igualdad sea cierta, se puede decir que:

$$\frac{1}{N} |\bar{x}_n|^2 = \frac{\lambda}{N} |x_n|^2 \quad \text{para } 0 \leq n < N - 1 \tag{3.26}$$

Así pues \bar{x} puede ser calculado en función de X usando la siguiente expresión

$$\bar{x}_n = \pm \sqrt{\lambda |x_n|^2} \tag{3.27}$$

3.2.6. Embebido de Datos.

Una vez que se tiene la secuencia $\{m\}^*$, y se ha segmentado el audio anfitrión x en marcos sin traslape tal como se describe en la Sección 3.2.2, se procede a normalizar cada marco para que su potencia media sea de $P_i = 0.5$. Hecho esto, se procede a calcular la TWD de cada marco x_i usando una wavelet Haar, pudiendo embeber un bit de $\{m\}^*$ en un coeficiente de una sub-banda de baja frecuencia c_i del marco, conforme a la regla descrita en la ecuación (2.8). Ésto se repite para cada elemento de $\{m\}^*$ en cada marco x_i de \mathcal{A} respectivamente. Obviamente la longitud del audio deberá ser lo suficientemente grande para contener la cantidad de información deseada.

Una vez que el marco ha sido modificado adecuadamente, se sigue el procedimiento inverso al ya mencionado. Ésto es, calcular la transformada wavelet inversa del marco en cuestión regresándolo al dominio del tiempo, para posteriormente regresarlo a su potencia original.

Con este último paso la señal queda marcada y está lista para ser transmitida.

3.2.7. Extracción de la Información.

El proceso de extracción propuesto es muy parecido al del algoritmo en el que se basa esta propuesta. La idea principal puede observarse en el esquema de la figura 3.4 donde se puede observar que la principal diferencia entre el esquema de la figura 2.5 es la adición de dos bloques, uno donde la señal sufre una corrección de los efectos del escalamiento en tiempo, y otro donde se realiza una normalización de potencia. Nótese que el primer bloque se representa con línea punteada, que significa que este paso sólo será efectuado si la señal que se desea analizar ha sufrido los efectos de un remuestreo.

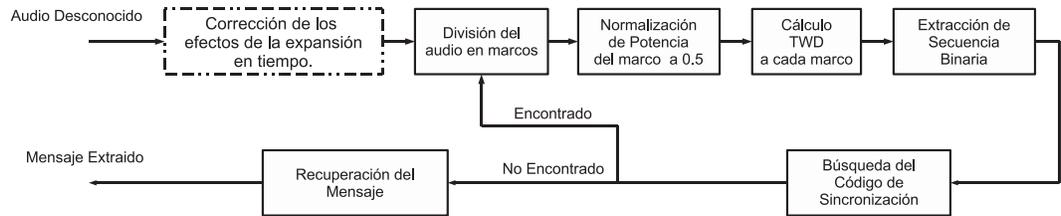


Figura 3.4: Esquema de extracción de información del algoritmo por normalizado de potencia.

Si el audio tiene que sufrir una conversión A/D -D/A, se caracteriza el comportamiento de la tarjeta de sonido para obtener el factor de escalamiento que corrija los efectos indeseados del escalamiento en tiempo.

- 1.- Si el audio ha sufrido un escalamiento en tiempo, se remuestrea con el factor de escalamiento obtenido de la caracterización de la tarjeta de sonido para corregir los efectos indeseados.
- 2.- El audio a analizar \hat{y} es dividido en cuadros de 256 muestras $\{x_1, x_2, \dots, x_M\}$, tal como se mencionó en la sección 3.2.2.

- 3.- Cada cuadro es normalizado a la potencia en que debió haber sido marcado. Dicho valor es de 0.5.
- 4.- Se le calcula la transformada wavelet a cada marco \mathcal{X}_i usando una wavelet madre tipo “Haar”.
- 5.- Sea $\{c'_i\}$ que denota los coeficientes de una sub-banda de baja frecuencia de cada segmento; se puede extraer una secuencia binaria de $\{c'_i\}$ usando la regla de la ecuación (3.28).

$$m'_i = \begin{cases} 1 & \text{si } c'_i - \lfloor \frac{c'_i}{S} \rfloor \cdot S \geq \frac{S}{2} \\ 0 & \text{si } c'_i - \lfloor \frac{c'_i}{S} \rfloor \cdot S \leq \frac{S}{2} \end{cases} \quad (3.28)$$

- 6.- Una vez obtenida la secuencia binaria m' , la cual tendrá un tamaño de M bits se procederá a buscar el código de sincronización dentro de ella, fijando un umbral de error “ e ” de 6 bits erróneos de un total de 64, lo cual significa un poco más del 9% de error.
- 7.- Si el código de sincronización no es encontrado, el audio será nuevamente segmentado, pero haciendo un desplazamiento de una muestra del audio, ignorando dicha muestra (y todas las que estén a la izquierda). El proceso anterior se repite nuevamente desde el paso 3 hasta encontrar la etiqueta de sincronización, o repetir todo el ciclo un máximo de 256 veces, con una muestra menos en cada ocasión.

En caso de que en las 256 búsquedas, no haya sido encontrado ninguna etiqueta de sincronización, se declarará que dicho audio no contiene ninguna marca.

3.2.8. Algoritmo de Búsqueda de Códigos de Sincronización

La búsqueda de los códigos de sincronización, se realiza con un algoritmo de fuerza bruta que barre toda la secuencia m' obtenida, comparándola bit a bit con la secuencia PN (código de sincronización) y contando el número de bits distintos entre ellas. Si el número de bits distintos es igual o mayor al umbral más uno ($e + 1$) de error, se asume que esa parte

de la secuencia m' no es el código buscado. Si esto ocurrió, la secuencia PN se recorre un bit a la derecha de la secuencia m' y se repite el proceso de comparación bit a bit.

Todo lo anterior se repite hasta que el último bit de la secuencia PN que funge como código de sincronización, queda alineado con el último bit de la secuencia m' . Si el numero de bits distintos entre ambos iguala o supera al umbral más uno, se declara que no existe un código de sincronización en la secuencia m' .

A continuación se presenta en pseudocódigo el algoritmo utilizado para la búsqueda del código de sincronización mencionado anteriormente.

BUSCA SINCRONIZACION(m', M, PN, N_{PN})

```

1  posicion ← -1
2  j ← 0
3  umbral ← 7
4  errores ← 0
5   $N_{busqueda} \leftarrow M - N_{PN}$ 
6  repetir
7      errores ← 0
8      desde  $i = 0$  hasta  $i = N_{PN}$ 
9          si  $m'[i + j] = PN[i]$ 
10             entonces  $errores \leftarrow errores + 1$ 
11                 si  $errores \geq umbral$ 
12                     entonces Rompe Ciclo
13             si  $i = N_{PN}$ 
14                 entonces  $posicion \leftarrow j$ 
15                     Rompe Ciclo
16              $j \leftarrow j + 1$ 
17         hasta  $j < N_{busqueda}$ 
18 regresar posicion
19
```

En el algoritmo anterior m' es una secuencia binaria de tamaño M , y PN es el código de sincronización que esta compuesto por N_{PN} elementos.

3.3. Algoritmo Por Modificación Simultánea de Coeficientes

El segundo algoritmo se basa en la lógica de embebido propuesta por Gopalan en [Gopalan04] donde se modifican dos coeficientes simultáneamente, pero se aprovecha la estructura del algoritmo anterior y el uso de la transformada wavelets como herramienta principal.

Para proponer la regla de marcado, se tomó en cuenta la idea de que en una señal real, los componentes de baja frecuencia son de larga duración[Polikar01]; si esta afirmación se cumple, entonces es lógico pensar que los dos coeficientes de más baja frecuencia de un segmento corto de audio que ocurren en tiempos distintos, serán muy parecidos. Suponiendo que se desea hacer una modificación que haga que dichos coeficientes se diferencien más, entonces la modificación (de ser necesario hacerla) de ambos coeficientes será muy pequeña, minimizando la distorsión de embebido.

3.3.1. Algoritmo de Embebido de Datos

En este algoritmo se pretende aprovechar la naturaleza misma del audio para evitar modificaciones innecesarias, y al mismo tiempo que se evitan modificaciones, la diferencia entre el audio marcado y el original expresado en función del SNR mejora, y por lo tanto la imperceptibilidad de dicha marca también mejora.

El esquema de marcado es exactamente el mismo que el de la figura 2.4, pero con una regla de marcado completamente distinta.

Dicha regla consiste en analizar dos coeficientes c_i y c_{i+1} , y establecer que si

$$|c_i| \geq |c_{i+1}| \quad (3.29)$$

se estará hablando de un “1” lógico, mientras que si

$$|c_i| < |c_{i+1}| \quad (3.30)$$

se considerará como un “0” lógico.

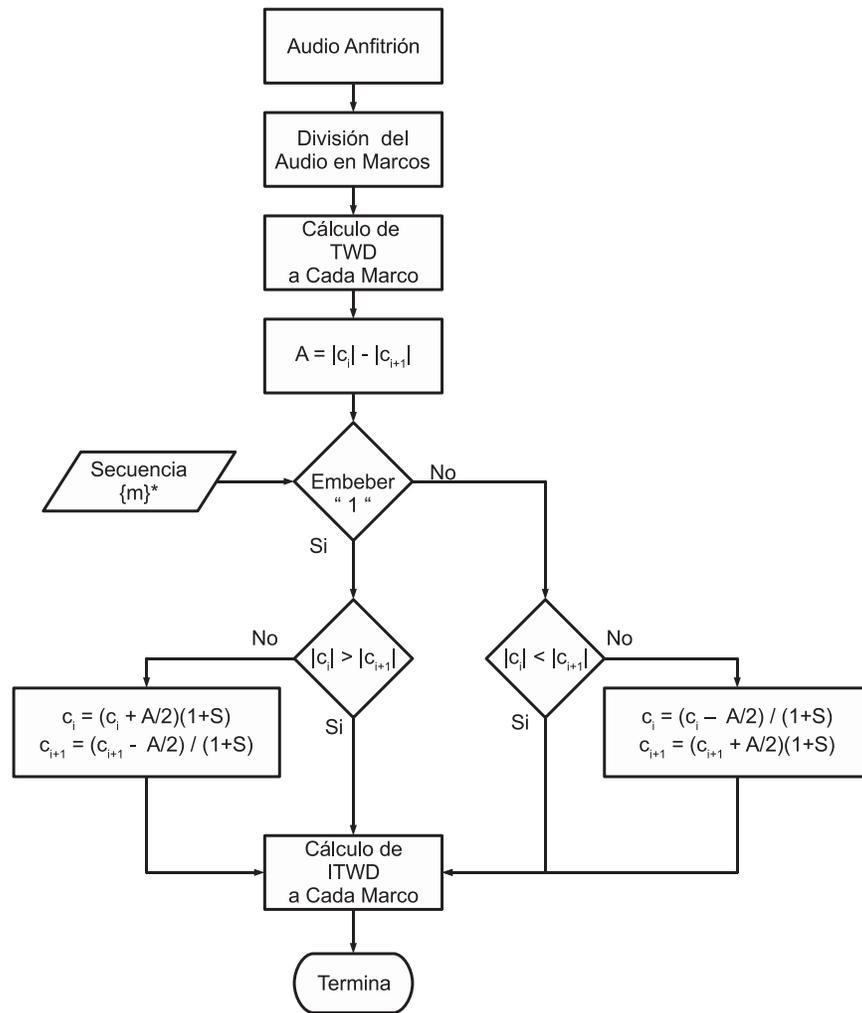


Figura 3.5: Diagrama de flujo que representa el algoritmo de marcado usando modificación simultánea de coeficientes

Si se debe embeber un “1”, primero se verifica si los coeficientes mencionado cumplen con lo establecido en la ecuación (3.29), si es así el marco se deja intacto, pero de lo contrario, dichos coeficientes serán modificados para que cumplan con dicha relación. Si se debe insertar un “0” lógico, se prueba si los coeficientes c_i y c_{i+1} cumplen con la relación de la ecuación (3.30). En caso de que la relación se cumpla el marco no sufre ninguna modificación y pasa en su estado original; si no se cumple, entonces los coeficientes son modificados para que cumplan con la relación.

La figura 3.5 muestra el algoritmo de marcado, y que es descrito a continuación:

Estrategia de Marcado

- 1.- La información a embeber es preparada según se indica en la sección 3.2.4.
- 2.- El audio es segmentado como se indica en la sección 3.2.2.
- 3.- Por cada bit de la secuencia de información $\{m\}^*$ se calcula la TWD a uno de los marcos obtenidos del paso anterior.
- 4.- Sea c_i y c_{i+1} dos coeficientes de una sub-banda de baja frecuencia que representan información de la misma frecuencia, pero en “distinto tiempo”. Se calcula la diferencia entre los valores absolutos de ambos coeficientes. $A = |c_i| - |c_{i+1}|$.
- 5.- Si el bit a embeber es “1” se verifican c_i y c_{i+1}

- a) Si cumplen la relación de la ecuación (3.29), se pasa el marco sin cambio alguno.
- b) Si no cumplen, ambos coeficientes se modifican según las reglas de las siguientes ecuaciones:

$$|\hat{c}_i| = \left(|c_i| + \frac{|A|}{2} \right) (1 + S) \quad (3.31)$$

y

$$|\hat{c}_{i+1}| = \left(|c_i| - \frac{|A|}{2} \right) \frac{1}{1 + S} \quad (3.32)$$

Donde $|\hat{c}_i|$ y $|\hat{c}_{i+1}|$ representan las magnitudes de los coeficientes de baja frecuencia del marco marcado, $|A|$ es la magnitud de la diferencia de magnitudes entre ambos coeficientes, y S es el coeficiente de “fuerza de embebido”.

6.- Si el bit a embeber es “0” se verifican c_i y c_{i+1}

a) Si cumplen la relación de la ecuación (3.30), se pasa el marco sin cambio alguno.

b) Si no cumplen, ambos coeficientes se modifican según las reglas de las siguientes ecuaciones:

$$|\hat{c}_i| = \left(|c_i| - \frac{|A|}{2} \right) \frac{1}{1+S} \quad (3.33)$$

y

$$|\hat{c}_{i+1}| = \left(|c_i| + \frac{|A|}{2} \right) (1+S) \quad (3.34)$$

Donde $|\hat{c}_i|$ y $|\hat{c}_{i+1}|$ representan las magnitudes de los coeficientes de baja frecuencia del marco marcado, $|A|$ es la magnitud de la diferencia de magnitudes entre ambos coeficientes, y S es el coeficiente de “fuerza de embebido”.

7.- Se calcula la transformada wavelet inversa al marco en análisis para regresar la señal al dominio del tiempo.

8.- Se repite todo el proceso para cada uno de los bits de la secuencia binaria m desde el paso 4.

3.3.2. Extracción de Datos

Nuevamente, el esquema usado es el que se muestra en la figura 2.5, pero con una regla de extracción distinta.

En este algoritmo, también se aprovecha la caracterización de la tarjeta de sonido, si es que el audio tiene que sufrir una conversión A/D -D/A, con lo que se obtendrá el factor de escalamiento que corrija los efectos indeseados del escalamiento en tiempo, y mejorar la detección.

1.- Si el audio ha sufrido un escalamiento en tiempo, se remuestrea con el factor de escalamiento obtenido de la caracterización de la tarjeta de sonido para corregir los efectos indeseados.

2.- El audio a analizar \hat{y} es segmentado en cuadros de 256 muestras $\{x_i, x_2, \dots, x_M\}$, tal como se mencionó en la sección 3.2.2.

- 3.- Se le calcula la transformada wavelet a cada marco \mathcal{X}_i usando una wavelet madre tipo “Haar”.
- 4.- Sean c_i y c_{i+1} dos coeficientes de una sub-banda de baja frecuencia referentes a la misma frecuencia, pero en distintos tiempos. De la comparación de ambos coeficientes se puede determinar que el dígito binario embebido es un “1” si se cumple con la condición de la ecuación (3.29) y se determinará que el dígito binario embebido en el marco es “0” si la comparación cumple con la ecuación (3.30).
- 5.- Una vez obtenida la secuencia binaria, la cual tendrá un tamaño de M bits se procederá a buscar el código de sincronización dentro de ella, fijando un umbral de error de 6 bits erróneos de un total de 64, lo cual significa un poco más del 9% de error. La búsqueda se realiza con el algoritmo descrito en la sección 3.2.8.
- 6.- Si el código de sincronización no es encontrado, el audio será nuevamente segmentado, pero haciendo un desplazamiento de una muestra hacia la derecha del audio, e ignorando dicha muestra (y todas las que estén a la izquierda). El proceso anterior se repite nuevamente desde el paso 3 hasta encontrar la etiqueta de sincronización, o repetir todo el ciclo un máximo de 256 veces, con una muestra menos en cada ocasión.

3.4. Conclusiones del Capítulo

Una vez que se conoce la naturaleza y efectos indeseados de los distintos tipos de ataques, se está en posibilidad de proponer alguna estrategia o método que permita que la información embebida sobreviva antes condiciones adversas de procesamiento de la señal.

Hasta ahora, la gran mayoría de los algoritmos del estado del arte, resuelven los problemas causados por la mayoría de los procesamientos que puede sufrir una señal marcada incluyendo la compresión. Sin embargo la mayoría de ellos aun no resuelve el problema de las conversiones D/A - A/D dejándolo como un reto a resolver.

Del modelo de distorsión por conversiones A/D - D/A presentado en el capítulo anterior, se propone una estrategia para contrarrestar los efectos del escalamiento temporal que sufre una señal después de haber sido remuestreada a una frecuencia ligeramente distinta

de la frecuencia ajustada. Dicho método consiste en conocer la distorsión que provoca cada tarjeta de sonido en particular, para poder reducirla lo máximo posible por medio de un remuestreo digital.

Cabe señalar que esta estrategia sería aplicable prácticamente a cualquier algoritmo existente, pues es independiente del método de embebido utilizado, es decir, se puede aplicar una corrección a un audio capturado, independientemente si se analiza o no dicho audio en busca de una marca.

Se presenta también una estrategia para hacer que la marca sea más robusta ante las modificaciones de amplitud de la señal anfitrión. Dicha estrategia consiste en la normalización de la potencia del marco analizado previo al embebido de la información. Esto hace posible conocer las condiciones originales de marcado de la señal de audio, y permite aproximar esas condiciones para la extracción de información.

Se propone también otro algoritmo el cual aprovecha características de los dos algoritmos descritos en el capítulo anterior, pero que supera en robustez a ambos. Este algoritmo aprovecha la naturaleza de la señal de audio original, permitiendo que sea modificada en menor medida que en los otros casos, mejorando significativamente su relación ruido - señal. La marca embebida soporta todos los ataques mencionados, e incluso puede ser extraída con un nivel de error aceptable, sin haber realizado la corrección del escalamiento en tiempo sufrido por la señal después de haber pasado por conversiones D/A -A/D.

De la descripción de los algoritmos así como de la regla de extracción en el algoritmo de Wu, y en el algoritmo con normalización de potencia, se puede ver que para que la marca pueda ser extraída, es necesario que en el proceso de análisis se conozca tanto el parámetro de intensidad de embebido S , como el contenido del código de sincronización. Esto refuerza la noción de seguridad para evitar la detección no autorizada de la marca. Por otro lado en el algoritmo de modificación simultánea de coeficientes, la única medida de seguridad es el hecho de tener que conocer la secuencia del código de sincronización para poder extraer la información embebida, sin embargo para los fines del objetivo planteado en este documento, la seguridad para evitar la detección de la marca no tiene ninguna relevancia. En otras palabras, es deseable que la marca pueda ser detectada por la computadora fácilmente, y con la menor cantidad de bits erróneos posibles.

Capítulo 4

Experimentos y Resultados

En esta sección se presentan los experimentos realizados, y se comparan los resultados obtenidos de la implementación del algoritmo de Wu en [Wu04][Wu05] y los resultados arrojados por los algoritmos propuestos en este trabajo.

4.1. Condiciones de los Experimentos

Para realizar los experimentos se utilizaron audios en formato WAV muestreados a 44.1 KHz con longitudes variables en tiempo. Las señales de audio utilizadas tienen propiedades distintas (potencia promedio, componentes frecuenciales, silencios, etc.).

Como códigos de sincronización se utiliza una secuencia binaria de 64 bits que es obtenida de la cadena “Msg:Msg:”. Es decir, cada carácter es convertido en una secuencia binaria de 8 bits según su código ASCII, la cual es concatenada en el mismo orden que sus caracteres correspondientes.

El mensaje a embeber es la secuencia binaria obtenida de la cadena “Hola Mundo\0”¹ que es equivalente a una secuencia binaria de 88 bits. En total se insertarán 152 bits.

Las transformada wavelets necesarias en todos los algoritmos son calculadas usando el esquema piramidal aplicando una Wavelet Haar, con 7 niveles de descomposición². El

¹Caracter nulo.

²El cálculo de únicamente 7 niveles proporciona los coeficientes necesarios, y ahorra un poco de costo computacional en el cálculo de la transformada.

tercer³ coeficiente de baja frecuencia de la TWD es utilizado en el algoritmo base, y el algoritmo por normalización de potencia, mientras que en el algoritmo de modificación simultánea se utilizan el tercer y cuarto coeficientes⁴.

El umbral de error tolerado en la búsqueda del código de sincronización para todos los experimentos es de 6 bits. Los experimentos han sido realizados en una computadora equipada con un procesador Pentium Core Duo modelo T2300 a 1.66 GHz con 1.2 GB de RAM. Los programas han sido escritos en lenguaje C++ usando el compilador y las librerías estándar GNU libG 6.

4.2. Observaciones a los Algoritmos Usados como Base

Aunque los los autores de los algoritmos mencionados en las secciones 2.4.1, y 2.4.3 reportan una serie de resultados en [Gopalan04] y [Wu04][Wu05] respectivamente, en esta sección se comentarán algunos aspectos considerados interesantes y que no son mencionados en los artículos citados.

4.2.1. Observaciones al Algoritmo de Inserción de Tonos Imperceptibles

Despues de la implementación de este algoritmo, y de la realización de varios experimentos se pudieron observar algunos detalles no comentados por su autor original.

En la sección 2.4.1 se mencionó que la elección de las frecuencias de los tonos a insertar es arbitraria. Se realizaron varios experimentos en los que las frecuencias de los tonos cayeron en la banda de baja frecuencia, en la banda “media” y finalmente en la banda de alta frecuencia encontrando que los mejores resultados en cuanto a robustez de la marca fueron cuando los tonos insertados cayeron en la banda de baja frecuencia, soportando incluso la compresión mp3. Sin embargo la calidad perceptiva de la señal marcada se degradó bastante. Por otro lado cuando los tonos cayeron en la banda media de frecuencia, la marca ya no fue tan robusta, y no soportó la compresión mp3. Finalmente, cuando los tonos fueron insertados en la banda de alta frecuencia, la calidad perceptiva de la señal no se vio afectada; es decir,

³(Que es $c[2]$)

⁴Que son los coeficientes $c[2]$ y el $c[3]$

que fue casi imposible escuchar los tonos insertados; sin embargo, y como era de esperarse, la marca fue completamente destruida en un filtrado pasabajas y en la compresión mp3.

Pero la observación más importante, fue el observar que la forma del espectro de potencia en el dominio de la frecuencia se mantenía sin cambios a pesar de los cambios de volumen de la señal marcada. Esto fue lo que inspiró la regla de marcado del algoritmo por modificación simultánea de coeficientes.

4.2.2. Observaciones al Algoritmo Basado en la Transformada Wavelet Discreta

En general, el autor de este algoritmo hace un estudio bastante extenso sobre su algoritmo y reporta los resultados obtenidos de un gran número de experimentos, sin embargo no menciona nada sobre lo poco robusto que resulta ser la marca ante ataques por conversiones D/A - A/D. Asegura también que la marca es robusta al remuestreo, sin embargo no especifica en que sentido.

Fue con la realización de varios experimento que se llegó a la conclusión que aunque la marca sobrevivía en una señal remuestreada, no era posible extraerla, sino hasta que dicha señal fuese remuestreada nuevamente para aproximarla a su estado original.

4.3. Medición de la Distorsión por Embebido de Información

Uno de los requerimientos de la IFPI para una señal marcada, es que la relación ruido - señal entre el audio original y el audio marcado sea mayor de 20 dB. Para verificar que dicho requerimiento se cumpla se analizaron varios tipos de audio marcados por los tres métodos basados en TWD, y se midieron sus SNR usando la ecuación (A.4). Cabe mencionar que esta prueba arroja datos un tanto subjetivos, pues el valor del SNR varia de forma inversamente proporcional a la fuerza de embebido de la señal; es decir, entre más se incremente la fuerza de embebido, la distorsión audible se hace más notoria, por lo tanto, el valor del SNR disminuye. Otro punto a considerar es que el “significado” del factor de intensidad de embebido S es distinto en todos los algoritmos presentados. Sin embargo el parametro de comparación será la imperceptibilidad de la marca. Es decir que se usará el

valor de S más grande posible antes de que la marca sea perceptible.

Los resultados obtenidos con una señal de voz se muestran en la tabla 4.1.

Algoritmo	S	SNR (dB)
Algoritmo de Wu	0.9	32.3020
Algoritmo de Wu con Normalización de potencia	1.2	37.3098
Algoritmo con modificación simultánea de coeficientes	0.25	36.9246

Tabla 4.1: Comparativa de distorsión causada por embebido de datos en una señal de voz.

Los resultados del mismo experimento, pero con un fragmento de una melodía del genero bolero se presentan en la tabla 4.2.

Algoritmo	S	SNR (dB)
Algoritmo de Wu	0.3	36.4169
Algoritmo de Wu con Normalización de potencia	0.96	48.1813
Algoritmo con modificación simultánea de coeficientes	0.3	35.4424

Tabla 4.2: Comparativa de distorsión causada por embebido de datos en una melodía del género bolero.

Con sonido de cuerdas se obtienen los resultados mostrados en la tabla 4.3.

Algoritmo	S	SNR (dB)
Algoritmo de Wu	0.5	37.4787
Algoritmo de Wu con Normalización de potencia	-	-
Algoritmo con modificación simultánea de coeficientes	0.09	28.6871

Tabla 4.3: Comparativa de distorsión causada por embebido de datos en sonido de cuerdas.

En este experimento en particular, ningún algoritmo fue capaz de insertar una marca imperceptible aun y cuando la fuerza de embebido s fuese muy pequeña. El reducir más el valor de S de lo que esta marcado en la tabla 4.3, implicaba errores en la detección de la marca. En la misma tabla se puede ver que no se reporta resultado en el algoritmo con normalización de potencia; ésto debido a que no se encontró ningún valor de S con el que la marca pudiese ser extraída sin error.

Con sonido de piano se obtienen los resultados mostrados en la tabla 4.4.

Algoritmo	S	SNR (dB)
Algoritmo de Wu	0.1	43.8673
Algoritmo de Wu con Normalización de potencia	1.3	40.2431
Algoritmo con modificación simultánea de coeficientes	0.001	27.5429

Tabla 4.4: Comparativa de distorsión causada por embebido de datos en sonido de piano.

En este caso, únicamente el algoritmo con modificación simultánea de coeficientes fue incapaz de insertar una marca imperceptible.

En una melodía de mariachi donde se conjuntan varios tipos de instrumentos, entre los cuales se cuentan instrumentos de cuerdas e instrumentos de viento así como voz, todos los algoritmos tuvieron un buen desempeño en cuanto a la imperceptibilidad de la marca, pues aún y cuando la marca puede ser escuchada, es difícil de distinguirla entre todos los elementos audibles de la melodía. Los resultados obtenidos se muestran en la tabla 4.5.

Algoritmo	S	SNR (dB)
Algoritmo de Wu	0.3	35.1951
Algoritmo de Wu con Normalización de potencia	2	25.4514
Algoritmo con modificación simultánea de coeficientes	0.25	37.9546

Tabla 4.5: Comparativa de distorsión causada por embebido de datos en una melodía de mariachi.

4.4. Mediciones de Tiempos de Ejecución

Este experimento se realizó para determinar la factibilidad del uso de los algoritmos presentados, en un ambiente en tiempo real, en una computadora común. El experimento consiste simplemente en medir el tiempo de ejecución del programa de búsqueda y extracción de marca en un audio que no contiene información embebida. Esto quiere decir que el programa tiene que barrer toda la secuencia de audio, o en otras palabras, el peor caso para el análisis y la búsqueda.

Para la realización de este experimento se utilizó una señal de audio de 1 minuto de duración, y el experimento se repitió 10 veces para cada algoritmo para obtener un tiempo

promedio de búsqueda.

Algoritmo	Tiempo (seg)
Algoritmo de Wu	13.12
Algoritmo de Wu con Normalización de potencia	82.28
Algoritmo con modificación simultánea de coeficientes	13.16

Tabla 4.6: Comparativa del promedio de tiempo de ejecución entre los tres algoritmos basados en transformada wavelets.

Se debe aclarar que la diferencia en tiempo de ejecución de un mismo experimento se debe a que se está trabajando en un ambiente multitarea y que los tiempos varían por la carga de trabajo que el procesador tiene en el momento de realizar la búsqueda.

De la observación de los resultados anteriores, es fácil deducir que el algoritmo con normalización de potencia no es adecuado para la aplicación que se propone en este documento, pues para analizar un audio de 60 segundos de duración, requiere más de 80 segundos de procesamiento, lo que lo deja fuera de la posibilidad de ser usado en tiempo real. Por otro lado se comprueba que el algoritmo con modificación simultánea de coeficientes es lo suficientemente rápido para ser usado en una aplicación de tiempo real.

De los resultados arrojados y reportados en las secciones anteriores, se decidió que el algoritmo con normalización de potencia no se seguiría estudiando, por lo que los experimentos subsecuentes se enfocan únicamente en el algoritmo con modificación simultánea de coeficientes.

4.5. Experimentos con el Algoritmo de Modificación Simultánea de Coeficientes

Como se ha mencionado a lo largo de este documento, del objetivo planteado en esta tesis se desprende que la cualidad más deseada de la marca a embeber, es que ésta sobreviva una serie de ataques y sea lo “más detectable” posible. Es por esto que los experimentos realizados están enfocados en medir “la tasa de bits erróneos” (BER por sus siglas en inglés) después de los ataques esperados que se dan en las condiciones de almacenamiento y transmisión de la señal marcada. El BER se calcula usando la ecuación (4.1), donde el

número de bits errneos se obtiene de la comparación bit a bit entre la secuencia binaria correspondiente al mensaje original, y la secuencia binaria extraída después del ataque.

$$BER = \frac{\text{Número de bits errneos}}{\text{Número total de bits}} 100\% \quad (4.1)$$

A continuación se presentan los resultados de hacer varios experimentos, en los que se somete a la señal marcada a los ataques que sufrirá en un ambiente real; estos ataques se realizarán por separado y de forma sintética. Al final se presentan los resultados de someter la señal marcada en un ambiente real con todos los ataques anteriormente mencionados, afectando simultáneamente la señal marcada.

4.5.1. Compresión mp3

Este experimento consiste en someter la señal marcada a distintos niveles de compresión mp3, descomprimirla, extraer el mensaje y revisar la tasa de bits errneos (BER).

En las tablas 4.7, 4.8, 4.9, 4.10 y 4.11 se presentan los resultados de comprimir señales de audio marcadas de voz, bolero, cuerdas, piano y mariachi, respectivamente.

Tasa de compresion MP3 (kbps)	BER
192	0 %
160	0 %
128	0 %
112	1.31 %
96	1.97 %
80	3.94 %
64	7.23 %
56	51.31 %
48	53.94 %

Tabla 4.7: Porcentajes de error después de ataque por compresión mp3 en señal de voz, usando un valor de fuerza de embebido de $S = 0.25$

4.5.2. Ataque por Ruido Blanco

Este experimento consiste en contaminar con diferentes intensidades de ruido blanco una señal de audio marcada. Para contaminar la señal marcada se utilizó la herramienta

Tasa de compresion MP3 (kbps)	BER
192	0 %
160	0 %
128	0.65 %
112	0 %
96	2.63 %
80	2.63 %
64	7.89 %
56	50.65 %
48	50 %

Tabla 4.8: Porcentajes de error después de ataque por compresión mp3 en melodía de bolero, usando un valor de fuerza de embebido de $S = 0.3$

Tasa de compresion MP3 (kbps)	BER
192	0 %
160	0 %
128	0 %
112	0 %
96	0 %
80	0.66 %
64	6.58 %
56	55.26 %
48	56.58 %

Tabla 4.9: Porcentajes de error después de ataque por compresión mp3 en sonido de cuerdas, usando un valor de fuerza de embebido de $S = 0.09$

Tasa de compresion MP3 (kbps)	BER
192	0 %
160	0 %
128	0 %
112	0 %
96	0 %
80	0.66 %
64	12.5 %
56	46.05 %
48	49.34 %

Tabla 4.10: Porcentajes de error después de ataque por compresión mp3 en sonido de piano, usando un valor de fuerza de embebido de $S = 0.09$

Tasa de compresion MP3 (kbps)	BER
192	0 %
160	0 %
128	0 %
112	0.66 %
96	1.32 %
80	1.97 %
64	9.87 %
56	45.39 %
48	44.74 %

Tabla 4.11: Porcentajes de error después de ataque por compresión mp3 en melodía de mariachi, usando un valor de fuerza de embebido de $S = 0.25$

“Generar ruido” del programa “Audacity” v.1.3.4-beta. Pero dada la naturaleza aleatoria del ruido, es difícil cuantificar y controlar la relación señal - ruido a la hora de contaminar la señal marcada, sin embargo, la herramienta mencionada permite elegir la amplitud máxima de la onda de ruido generada, y que puede tomar cualquier valor en el rango de 0 a 1. De manera arbitraria se escogieron 6 valores para la amplitud máxima de la señal de ruido que son 0.001, 0.009, 0.01, 0.03, 0.05, 0.08, y 0.1, donde los últimos tres valores para la amplitud de señal de ruido provocan que escuchar la señal de audio marcada y contaminada sea bastante molesto para el oído humano.

Las tablas 4.12, 4.13, 4.14, 4.15 y 4.16, muestran los resultados de contaminar señales de voz, música de bolero, sonido de cuerdas, sonido de piano, y música de mariachi respectivamente.

SNR (dB)	BER
44.3405	0 %
25.2378	4.6052 %
24.32	5.9210 %
14.7837	15.1316 %
10.35	21.7105 %
6.25916	30.9211 %
4.31603	4.31603 %

Tabla 4.12: Porcentajes de error en la detección de una marca en señal de voz contaminada a distintos niveles de ruido

SNR (dB)	BER
45.8774	0 %
26.7931	0.6578 %
24.32	5.9210 %
16.3436	9.21053 %
7.82134	17.7632 %
5.8703	18.4211 %

Tabla 4.13: Porcentajes de error en la detección de una marca en señal de voz contaminada a distintos niveles de ruido

SNR (dB)	BER
54.5514	0 %
37.0791	1.3157 %
36.1693	1.3157 %
26.643	13.1579 %
22.2155	18.4211 %
18.1205	19.7368 %
16.1997	27.6316 %

Tabla 4.14: Porcentajes de error en la detección de una marca en sonido de cuerdas contaminado a distintos niveles de ruido

SNR (dB)	BER
46.9180	0 %
27.8415	0.6578 %
26.9252	0 %
17.3870	1.3158 %
12.9324	4.6052 %
8.8493	9.2105 %
6.9212	11.8421 %

Tabla 4.15: Porcentajes de error en la detección de una marca en sonido de piano contaminado a distintos niveles de ruido

SNR (dB)	BER
49.3835	0 %
30.3028	0.6578 %
29.3837	1.3158 %
19.8331	4.6052 %
15.3976	10.5263 %
11.3293	23.0263 %
9.3779	20.3947 %

Tabla 4.16: Porcentajes de error en la detección de una marca en una melodía de mariachi contaminada a distintos niveles de ruido

4.5.3. Ataque por Conversión D/A - A/D

Este experimento consiste en someter una señal de audio marcada a una compresión mp3 con una tasa de 128 kbps para ser reproducida por una tarjeta de sonido típica sufriendo así una conversión D/A; posteriormente dicha señal de audio es transmitida por un canal analógico (cable de audio típico conectado de la salida de audio de una tarjeta de sonido a la entrada de captura de otra tarjeta de sonido) para ser capturada nuevamente efectuándose así una conversión A/D. El audio capturado es analizado en busca de la marca, y en caso de encontrarla se mide la cantidad de bits erróneos. La tabla 4.17 muestra los resultados obtenidos de someter varias señales de audio al proceso antes mencionado.

Tipo de Audio	BER
Voz	9.2105 %
Bolero	11.8421 %
Cuerdas	17.7632 %
Piano	5.2613 %
Mariachi	5.2632 %

Tabla 4.17: Porcentajes de error después de haber sometido la señal a una conversión D/A - A/D

4.6. Conclusiones del Capítulo

En este capítulo se presentaron algunos comentarios sobre los algoritmos que han servido como base para este trabajo, así como los resultados obtenidos de una gran cantidad

de experimentos realizados con los algoritmos propuestos en este trabajo.

Una de los aspectos a tener en cuenta es que el parámetro S tiene efectos distintos en cada uno de los algoritmos presentados, por lo que la única manera de comparar la efectividad de los algoritmos respecto a este parámetro, es con una prueba subjetiva de percepción audible de la marca.

En los experimentos realizados con el algoritmo con normalización de potencia se observó que el SNR mejoró con respecto al algoritmo de Wu. Esto se puede explicar de manera sencilla si se toma en cuenta el hecho de que al normalizar la potencia de un marco de audio a un valor mayor que el que tenía originalmente, equivaldría a hacer un zoom sobre dicho marco; así al hacer un cambio muy pequeño en él, dicho cambio se vuelve más pequeño aun cuando dicho marco regresa a su potencia original.

También se pudo comprobar que el algoritmo de modificación simultánea de coeficientes superó al algoritmo de Wu en cuanto a la distorsión de embebido medida por el SNR. Esto era una cualidad esperada, ya que al reducir la cantidad de marcos modificados, es lógico que la distorsión disminuirá.

De los resultados obtenidos, se concluye que el único algoritmo que cumple con todos los requerimientos necesarios para la aplicación propuesta, es el algoritmo por modificación simultánea de coeficientes.

La imperceptibilidad de la marca embebida por cualquiera de los tres algoritmos es mejor cuando se utiliza un audio anfitrión con un contenido frecuencial muy rico, y sobre todo que la potencia de dicho audio sea lo más grande posible sin que ello implique saturación de dicha señal. En otras palabras, entre más volumen, y el espectro de frecuencias sea más rico en el audio original, la marca es menos perceptible.

Capítulo 5

Conclusiones

5.1. Conclusiones Generales

- En el algoritmo propuesto por Wu[Wu04][Wu05] la información es embebida modificando la energía de un coeficiente en particular, y tal modificación es dependiente de un parámetro de “fuerza de embebido” S . La extracción por consiguiente, consiste en el análisis de ese mismo coeficiente, y determina un valor binario en base a un rango predefinido dependiente de S en el que debe caer el valor de la magnitud de energía de dicho coeficiente. Es obvio que al cambiar la magnitud de la señal marcada, seguramente cambiará el rango en el que debe caer el coeficiente; es obvio también que existe otro valor S' que determina un rango correcto de detección. Pero para determinar el valor correcto, sería necesario conocer la energía del marco al momento de ser marcado.

Es precisamente por esta razón que en el algoritmo con normalización de potencia se modifica la magnitud de la potencia promedio de la señal a marcar a un valor conocido, permitiendo así aproximar el marco marcado a su estado original y aumentar las probabilidades de una extracción correcta.

- Aunque la marca embebida por el algoritmo con normalización de potencia superó casi todas las pruebas en los experimentos sintéticos, resulta ser inadecuado para implementarse en una aplicación como la propuesta en este trabajo.

La razón principal de la afirmación anterior, es que el proceso de normalización al momento de la búsqueda de los códigos de sincronización, hace que el procesamiento dure más que el tiempo de reproducción de la señal marcada, haciendo que este algoritmo sea inadecuado para uso en “tiempo real”.

Otra de las razones es la poca robustez mostrada ante ataques por ruido, pues éste agrega energía a la señal marcada, y obviamente afecta en la aproximación a las condiciones de marcado haciendo muy difícil obtener los rangos adecuados.

- Una señal de audio no pierde su “forma” tras una amplificación o atenuación de su volumen, simplemente cambia la magnitud de su energía promedio. Es lógico pensar entonces que el resultado de alguna transformación de dominio entre una misma señal amplificada y sin modificación, mostrará que su “forma” es similar, y cuya única y principal diferencia es el contenido de su energía. Un ejemplo claro de esto es la relación de Parseval aplicada en la transformada de Fourier.
- El algoritmo de modificación simultánea de coeficientes aprovecha la propiedad descrita en el punto anterior, pues asume que aunque la energía total de la señal de audio, y la magnitud de los coeficientes resultantes del cálculo de la TWD cambien radicalmente, la “forma” de la gráfica del resultado de la TWD no sufrirá ningún cambio, para cualquier volumen de la señal analizada en que no se pierda información correspondiente a algún componente de frecuencia.

Por lo expuesto anteriormente, se concluye que mientras la distorsión de la magnitud de la señal no afecte los componentes de baja frecuencia que han sido modificados para la inserción de la información, la relación entre ellos no variará y por lo tanto el mensaje puede ser extraído sin errores.

- De los experimentos realizados, se puede concluir que el algoritmo por modificación simultánea de coeficientes pierde un poco de robustez ante la compresión mp3, pues comienzan a aparecer errores con compresiones a bitrates menores a 112 kbps. Sin embargo y a pesar de dichos errores, este algoritmo es la mejor opción de las que se exponen en este trabajo. En cualquier estación de radio, el mínimo bitrate de

compresión mp3 considerado adecuado para transmisión es de 128 kbps, que es una tasa de compresión en la que el algoritmo demuestra robustez en cualquier tipo de audio, lo que lo convierte en adecuado para los fines que se pretenden alcanzar con esta investigación.

- Basándose en en los resultados obtenidos de una gran cantidad de experimentos, tanto en condiciones sintéticas, como en condiciones reales, se concluye que el algoritmo de modificación simultánea cumple con la mayoría de los requerimientos de la IFPI. El único requerimiento que no se cumple del todo es que el único elemento de seguridad para evitar la detección de la marca es la secuencia binaria de sincronización. Conociendo ésta, la extracción del mensaje oculto es sencilla. Sin embargo, dada la naturaleza de la aplicación que se propone, la seguridad en este sentido no es una cuestión primaria.
- Gracias a la robustez de la marca embebida por el algoritmo de modificación simultánea de coeficientes, es posible recuperar la marca con un porcentaje de error aceptable, incluso si la señal no ha pasado por el proceso de remuestreo que corrige el efecto de la expansión en tiempo. Ésto reduce considerablemente el tiempo de cálculo; y aprovechando las características de localización tiempo-frecuencia de la TWD, hace que el algoritmo sea adecuado para su implementación en aplicaciones con requerimientos de análisis en tiempo real.

5.2. Trabajos Futuros

1. Dado que es posible recuperar una marca de una señal de audio que ha sufrido distintos tipos de ataques incluyendo conversiones D/A - A/D sin corregir el efecto del escalamiento temporal, con un porcentaje de error aceptable, la implementación de una estrategia y códigos de corrección de errores, agregará aun más robustez a la marca.
2. Mejorar la robustez de la marca de manera que ésta se convierta en un canal efectivo de transmisión de información.

3. Agregar tecnología de reconocimiento de formas en caso de que la marca de agua se interprete como una imagen binaria.
4. Investigar y proponer una estrategia para resolver el problema de remuestreo dinámico.
5. Investigar el efecto de la reconstrucción realizada por la tarjeta de sonido para explicar los errores por cambio de magnitudes de los coeficientes en análisis.
6. Investigar si el uso de alguna wavelet discreta distinta a la Haar proporciona mejor robustez.

Apéndice A

Relación Señal-Ruido

La relación señal/ruido (en inglés Signal to noise ratio SNR o S/N) se define como el margen que hay entre la potencia de la señal que se transmite y la potencia del ruido que la corrompe. Se define mediante la ecuación (A.1)

$$SNR = \frac{P_{signal}}{P_{noise}} \quad (\text{A.1})$$

Debido a que muchas señales tienen un rango dinámico muy grande, el SNR es expresado en términos de decibeles. Así, la relación señal-ruido, en términos de decibeles, se define como 10 veces el logaritmo de la relación de potencia, tal como se muestra en la ecuación (A.2)

$$SNR_{dB} = 10 \log_{10} \left(\frac{P_{signal}}{P_{noise}} \right) \quad (\text{A.2})$$

En este documento, el SNR se utiliza como una medida de la distorsión entre el audio original F , y el audio marcado F' . Si se asume que la distorsión causada por el embebido de información es ruido, entonces se puede obtener la señal de ruido restando la señal marcada F' de la señal original F .

$$F_{noise} = F - F' \quad (\text{A.3})$$

Con la ecuación (A.3) es posible obtener el denominador de la ecuación (A.2), que combinándola con la ecuación (2.3) se obtiene la ecuación (A.4).

$$SNR_{dB} = 10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} [f_n]^2}{\sum_{n=0}^{N-1} [f_n - f'_n]^2} \right) \quad (\text{A.4})$$

Apéndice B

Circuito de Muestreo y Retención

El propósito de un circuito de muestreo y retención es probar periódicamente la señal de entrada analógica, continuamente cambiante, y convertir las muestras en una serie de niveles PAM de ancho constante [Tomasi00].

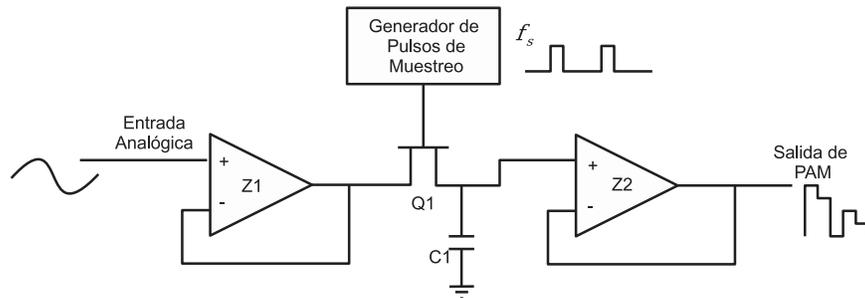


Figura B.1: Circuito de muestreo y retención.

La figura B.1 muestra el diagrama esquemático de un circuito de muestreo y retención. El FET Q1 actúa como un conmutador sencillo. Cuando se “enciende”, proporciona un camino de baja impedancia para depositar el voltaje de la muestra analógica en el capacitor “C1”. El tiempo en el que Q1 está “encendido” se llama *apertura* o *tiempo de adquisición* y corresponde al tiempo “activo” de la señal proporcionada por el generador de pulsos de muestreo. Esencialmente C1 es el circuito de retención. Cuando Q1 está “apagado” (tiempo “pasivo” del pulso de muestreo), el capacitor no tiene un camino completo por el cual descargarse y, por lo tanto almacena el voltaje muestreado. El tiempo de almacenamiento del

capacitor, también se llama tiempo de conversión A/D debido a que, durante este tiempo, el ADC convierte al voltaje de la muestra en un código digital. Si la entrada al ADC está cambiando mientras se realiza la conversión, resultará en una distorsión llamada “*distorsión de apertura*”, por lo que es extremadamente importante que el pulso de muestreo, el tiempo de retención, y el ADC estén muy bien sincronizados. Por lo tanto, teniendo un tiempo de apertura corto y manteniendo la entrada de información al ADC relativamente constante, el circuito de muestreo y retención reduce la distorsión de apertura.

En circuitos reales, el generador de pulsos de muestreo tendrá como componente principal un cristal oscilador, cuya frecuencia de oscilación depende de su composición química, forma y tamaño físico (características determinadas por el fabricante), y en menor medida de la magnitud de la corriente aplicada. Esta última depende de toda una circuitería de polarización y estabilizamiento.

Es bastante sencillo calcular los valores de los componentes de dichas circuiterías basándose en las especificaciones de los fabricantes para lograr una frecuencia f_s de muestreo. Sin embargo, todos y cada uno de los componentes electrónicos tendrán una desviación de su valor marcado llamado tolerancia, que de manera típica será de $\pm 5\%$. Estas diferencias entre los valores marcados, y los valores reales, conducen siempre a pequeñas variaciones en los resultados esperados.

Esta es la razón por la que cuando se remuestrea una señal de audio, se obtienen efectos indeseados como la expansión en tiempo, distorsión de apertura y ruidos.

Apéndice C

Transformada Wavelet

La “Transformada Wavelet” es una herramienta que separa datos, funciones u operadores en diferentes componentes de frecuencia, y estudia cada componente con una resolución adecuada a su propia escala [Daubechies92].

C.1. Análisis Multiresolución

Aunque los problemas de resolución en tiempo y frecuencia son resultado de un fenómeno físico (el principio de incertidumbre de Heisenberg) y existan a pesar de la transformación usada, es posible analizar cualquier señal usando una aproximación alternativa llamada *Análisis Multiresolución* (MRA), la cual tal como su nombre lo indica, analiza la señal a diferentes frecuencias con diferentes resoluciones, es decir, que cada componente espectral no es resuelto de la misma manera.

Este tipos de análisis está diseñado para dar una resolución buena en tiempo pero pobre en frecuencia en las altas frecuencias; y una resolución buena en frecuencia y pobre en tiempo a bajas frecuencias. Esta aproximación tiene sentido especialmente cuando la señal analizada tiene componentes de alta frecuencia durante pequeños periodos de tiempo, y componentes de baja frecuencia que se mantienen por periodos relativamente largos. Afortunadamente, la gran mayoría de las señales que se encuentran en aplicaciones prácticas son de este tipo [Polikar01].

C.2. Características de la Transformada Wavelet

En análisis de señales, la transformada wavelet nos permite ver un historial en el transcurso del tiempo pero en términos de componentes frecuenciales. En este sentido, la transformada wavelet se realiza en la misma forma que la transformada de Fourier [Williams94]. Tanto la transformada de Fourier como la transformada wavelet representan una señal a través de una combinación lineal de sus funciones base. Para la transformada de Fourier, las funciones base son dilataciones de señales senos y cosenos (cada una abarcando el intervalo de tiempo completo). Para la transformada wavelet, existirán diferentes translaciones y dilataciones de una función llamada *Wavelet Madre* junto con una función de escalamiento (cada una abarcando un subintervalo logarítmicamente reducido). Las dilataciones de ambos conjuntos de funciones base son posibles gracias a sus localizaciones en frecuencia, permitiendo así obtener la información de frecuencia de la señal analizada. Esto lleva a la diferencia más importante entre los dos conjuntos de funciones base: “La Localización en Tiempo”.

Las funciones base de la transformada wavelet son “compactas” o “de duración finita”, mientras que en Fourier las funciones seno y coseno no lo son. Esta característica le permite a la transformada wavelet obtener información de tiempo de una señal, adicional a la información de frecuencia.

La Transformada wavelet (TW) usa una “ventana de análisis” que varía la escala de frecuencia lo cual resulta ventajoso para el análisis de señales que contienen tanto “discontinuidades”, como “regiones suaves”. Funciones base cortas en tiempo (o de alta frecuencia), son necesarias para analizar las “discontinuidades”, mientras que al mismo tiempo, muy bajas frecuencias son necesarias para los “componentes suaves”. Este es exactamente el tipo de arreglo escala-frecuencia que se obtiene de la transformada wavelet discreta.

La figura C.1 representa esta relación, y es comúnmente usada para explicar como la resolución tiempo-frecuencia debe ser interpretada. Cada caja en la figura C.1 corresponde a un valor de la transformada wavelet en el plano tiempo-frecuencia. Nótese que las cajas tienen un área de cierto valor distinto de cero, lo cual implica que el valor de un punto en particular en dicho plano, no puede ser conocido; por lo que todos los puntos en el

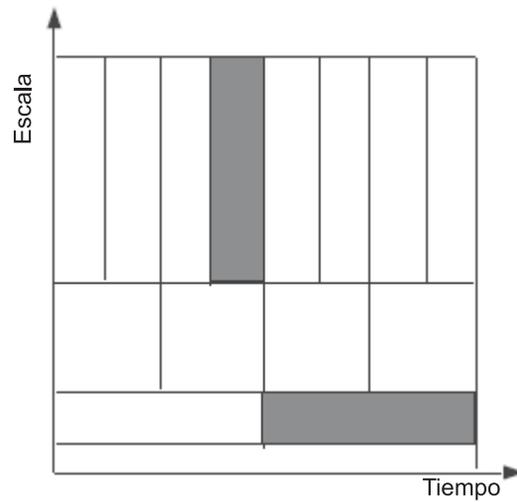


Figura C.1: Escala Diádica

plano tiempo-frecuencia que caen en una caja, están representados por un único valor de la transformada wavelets.

Se debe ver, que aunque las alturas y los anchos de las cajas varían, sus áreas son constantes. Esto es, que cada caja representa una igual porción del plano tiempo-frecuencia, pero dando diferentes proporciones al tiempo y a la frecuencia. Se debe notar que también que a bajas frecuencias, la altura de las cajas son más pequeñas (lo cual corresponde a una mejor resolución de frecuencia, puesto que hay menos ambigüedad con respecto al valor de la frecuencia exacta), pero sus anchos son más grandes (lo cual corresponde a una resolución pobre en tiempo, ya que hay más ambigüedad respecto al valor del tiempo exacto). A más altas frecuencias, el ancho de las cajas disminuye, y por lo tanto la resolución en tiempo mejora, y las alturas de las cajas incrementa, por lo que la resolución en frecuencia se vuelve más pobre.

C.3. Cálculo de la Transformada Wavelet Discreta

La idea principal consiste en obtener una representación escala-tiempo de una señal digital, la cual es obtenida usando técnicas de filtrado digital. La transformada wavelet

continua es una correlación entre una wavelet madre a diferentes escalas y la señal; usando la escala (o frecuencia) como una medida de similaridad. La transformada wavelet continua se puede calcular mediante el cambio de escala de la ventana de análisis, recorriendo dicha ventana en tiempo, multiplicando por la señal, e integrando sobre todos los tiempos. En el caso discreto, filtros a diferentes frecuencias de corte son usados para analizar la señal a diferentes escalas. La señal es pasada a través de series de filtros pasa altas para analizar las altas frecuencias, y también es pasada por series de filtros pasabajas para analizar las bajas frecuencias.

La resolución de la señal, la cual es una medida de la cantidad de información de detalle en la señal, es cambiada por las operaciones de filtrado, y la escala es cambiada ya sea por operaciones de “sobremuestreo” o más comúnmente, de “submuestreo”.

C.3.1. Esquema Piramidal

El procedimiento comienza con el paso de la señal (secuencia) a través de un filtro digital pasabajas cuya frecuencia de corte se encuentra a media banda y con respuesta al impulso $h[n]$. Este filtro pasabajas remueve todas las frecuencias que están por arriba de la mitad del espectro de frecuencia de dicha señal. Por ejemplo, si una señal tiene un componente de 1000Hz como máximo, entonces el filtro pasabajas removerá todos aquellos componentes de más de 500Hz.

Después de pasar la señal a través del filtro pasabajas, la mitad de las muestras pueden ser eliminadas de acuerdo a la regla de Nyquist, ya que la señal ahora tiene como máxima frecuencia $\pi/2$ radianes en lugar de π radianes. Con el simple hecho de un submuestreo por 2, la señal tendrá la mitad de sus muestras; y con esto, la escala de la señal ahora será del doble. Hay que notar que el filtrado pasabajas remueve la información de alta frecuencia, pero deja la escala sin cambios. Sólo el proceso de submuestreo cambia la escala. Por otro lado, la resolución está relacionada a la cantidad de información en la señal, y por lo tanto, es afectada por las operaciones de filtrado. En resumen, el filtrado pasabajas parte por la mitad la resolución, pero mantiene intacta la escala; la señal es entonces submuestreada por 2 ya que la mitad de el número de muestras es redundante, y esto dobla la escala.

Este procedimiento se puede expresar matemáticamente con la ecuación (C.1)

$$c_n = \sum_{k=0}^{L-1} h_k x_{2n-k} \tag{C.1}$$

La figura C.2 ilustra el procedimiento completo.

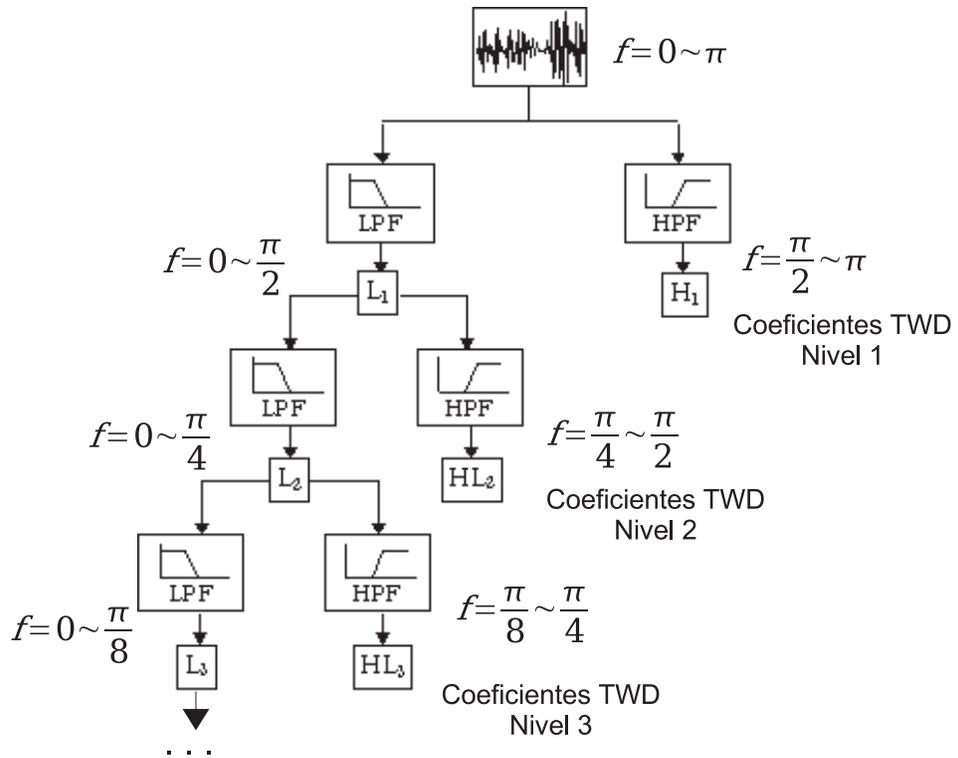


Figura C.2: Esquema de codificación piramidal para el cálculo de la TWD

Referencias

- [Arnold00] Arnold, M. Audio watermarking: Features, applications, and algorithms. *En IEEE International Conference on Multimedia and Expo (II)*, págs. 1013–1016. 2000.
- [Bosi03] Bosi, M. y Goldberg, R. E. *Introduction to Digital Audio Coding and Standards*. Engineering and Computer Science. Kluwer Academic Publishers, 1^a ed^{ón}., 2003. ISBN 1-4020-7357-7.
- [Cvejic04] Cvejic, N. *Algorithms for audio watermarking and steganography*. University of Oulu, Finland, <http://herkules.oulu.fi/isbn9514273842/>, 2004. ISBN 951-42-7384-2.
- [Daubechies92] Daubechies, I. *Ten Lectures on Wavelets*. CBMS-NSF Regional Conference Series, City Press, 1992.
- [Gopalan04] Gopalan, K. y Wenndt, S. Audio steganography for cover data transmission by imperceptible tone insertion. *En The IASTED International Conference on Communication Systems And Applications (CSA 2004)*. 2004.
- [Gopalan05] Gopalan, K. Audio steganography by cepstrum modification. *En IEEE International Conference on Acoustics, Speech, and Signal Processing*, tomo 5, págs. 481–484. 2005.
- [Gruhl05] Gruhl, D., Lu, A., y Bender, W. Hecho hiding. *En 1st Information Hiding Workshop*, tomo 1174, págs. 295–315. Springer Berlin / Heidelberg, 2005.

- [Hsieh02] Hsieh, C. T. y Tsou, P. Y. Blind cepstrum domain audio watermarking based on time energy features. *En IEEE International Conference on Digital Signal Processing*, tomo 2, págs. 705–708. 2002.
- [Katzenbeisser00] Katzenbeisser, S. y Petitcolas, F. A., eds. *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House, Inc., Norwood, MA, USA, 2000. ISBN 1580530354.
- [Lee00] Lee, S.-K. y Ho, Y.-S. Digital watermarking in cepstrum domain. *En Digest of technical papers, International Conference on Consumer Electronics*, págs. 334–335. 2000.
- [Lyons97] Lyons, R. G. *Understanding Digital Signal Processing*. Addison Wesley, 1ª ed^{ón}., 1997. ISBN 0-201-63467-8.
- [Mansour05] Mansour, M. F. y Tewfik, A. H. Data embebing in audio using time-scale modification. *En IEEE Speech and Audio Processing*, tomo 13, págs. 432–440. 2005.
- [Moulin01] Moulin, P. The role of information theory in watermarking and its application to image watermarking. *Signal Process.*, 81(6):1121–1139, 2001. ISSN 0165-1684. doi:[http://dx.doi.org/10.1016/S0165-1684\(01\)00037-8](http://dx.doi.org/10.1016/S0165-1684(01)00037-8).
- [Nocedal99] Nocedal, J. y Wright, S. J. *Numerical Optimization*. Springer, August 1999. ISBN 0387987932.
- [Oppenheim98] Oppenheim, A. V., Willsky, A. S., y Nawab, S. H. *Señales y Sistemas*. Prentice Hall Hispanoamericana, 2ª ed^{ón}., 1998. ISBN 970-17-0116-X.
- [Polikar01] Polikar, R. The engineer’s ultimate guide to wavelet analysis, the wavelet tutorial. Rowan University, College of Engineering, 2001. URL <http://users.rowan.edu/~polikar/WAVELETS/WTtutorial.html>
- [Popa98] Popa, R. *An Analisis of Steganographic Techniques*. Tesis Doctoral, The “Politehnica” University of Timisoara, 1998.

- [Ruanaidh98] Ruanaidh, J. J. Ó. y Pun, T. Rotation, scale and translation invariant spread spectrum digital image watermarking. *Signal Processing*, 66:303–317, mayo 1998.
- [Smith02] Smith, J. O. *Digital Audio Resampling Home Page*. <http://www.ccrma.stanford.edu/~jos/resample/>, January 28 2002.
- [Smith07] Smith, J. O. *Mathematics of the Discrete Fourier Transform (DFT): with Audio Applications*. W3K Publishing, <http://www.w3k.org/books/>, 2007. ISBN 978-0-9745607-4-8.
- [Swanson98] Swanson, M. D., Zhu, B., y Tewfik, A. H. Audio watermarking and embedding current state of the art, challenges and future directions. *En GDM Report 41, Multimedia and Security Workshop at ACM Multimedia*, págs. 63–69. 1998.
- [Tomasi00] Tomasi, W. *Sistemas de comunicaciones electrónicas*. Prentice Hall, 2^a ed^{ón}., 2000. ISBN 968-880-674-9.
- [Williams94] Williams, J. R. y Amaratunga, K. Introduction to wavelets in engineering. *En International Journal for Numerical Methods in Engineering*, tomo 37, págs. 2365–2388. Wiley, Chichester, ROYAUME-UNI (1969) (Revue), 1994.
- [Wu99] Wu, C.-P., Su, P.-C., y Kuo, C. Robust audio watermarking for copyright protection, 1999.
URL <http://citeseer.ist.psu.edu/article/wu99robust.html>
- [Wu04] Wu, S., Huang, J., Huang, D., y Shi, Y.-Q. Self-synchronized audio watermarking in dwt domain. *IEEE International Symposium on Circuits and Systems (ISCAS)*, 5:23–26, 2004.
- [Wu05] Wu, S., Huang, J., Huang, D., y Shi, Y. Q. Efficiently self-synchronized audio watermarking for asured audio data transmission. *IEEE Transactions on Broadcasting*, 51:69–76, 2005.

- [Xiang06] Xiang, S. y Huang, J. Analysis of d/a and a/d conversions in quantization-based audio watermarking. *International Journal of Network Security*, 3:230–238, 2006.

Glosario

Diádica *Perteneciente o relativo a la diáda* - Pareja de dos seres o cosas, estrecha y especialmente vinculados entre sí.

FET *Transistor de efecto de campo* - El FET de juntura o jFET, es un transistor monopolar que sólo necesita de portadores mayoritarios para trabajar. Sus principales ventajas frente al transistor bipolar son: su menor consumo de corriente y mayor impedancia de entrada.

Filtro FIR uno-cero *one-zero FIR Filter* - Es un filtro de respuesta al impulso finita que llega a cero en un número finito de muestras.

IFPI *International Federation Of The Phonographic Industry* - Federación Internacional De La Industria Fonográfica.

PAM *Modulación de amplitud de pulsos* - La amplitud de un pulso de posición constante y de ancho constante varía de acuerdo a la amplitud de la señal analógica.

PN *Pseudo-random Noise* - Secuencia de numeros binarios que parece ser aleatoria, pero que sin embargo es perfectamente determinística.

SPL *Sound Pressure Level* - Nivel de Presión de Sonido.