UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

PROYECTO FIN DE CARRERA

EVALUACIÓN DE CARACTERÍSTICAS MUSICALES PARA DETECCIÓN DE TIPOS DE AUDIO

RICARDO LANDRIZ LARA

SEPTIEMBRE 2014

EVALUACIÓN DE CARACTERÍSTICAS MUSICALES PARA DETECCIÓN DE TIPOS DE AUDIO

AUTOR: Ricardo Landriz Lara
TUTOR: Daniel Ramos Castro

Área de Tratamiento de Voz y Señales
Dpto. de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Septiembre de 2014

Resumen

El objetivo de este proyecto es el de desarrollar un sistema capaz de identificar y segmentar audio radiofónico en distintas clases acústicas utilizando características musicales.

Se ha realizado un estudio sobre el estado del arte en el campo de la segmentación de audio, analizando los algoritmos y técnicas más utilizadas así como las bases de datos con más influencia de la literatura. El algoritmo desarrollado hace uso de modelos estadísticos basados en mezcla de gaussianas (GMM-UBM) a partir de características basadas en la entropía cromática espectral, extraída del audio de la base de datos proporcionada por la evaluación ALBAYZIN 2010 de segmentación de audio.

El sistema implementado se divide en siete sub-tareas, identificando en cada una de ellas un tipo de audio distinto. Entre estas sub-tareas se pueden encontrar sistemas como un discriminador de voz/música o un detector de actividad de voz, entre otros. Los resultados obtenidos se han comparado y fusionado con el sistema presentado por el grupo de investigación ATVS en la evaluación de segmentación de audio ALBAYZIN de 2010. Aun teniendo rendimientos inferiores, gracias a la fusión se llega a mejorar el rendimiento global de ambos sistemas.

Durante la ejecución de este proyecto fin de carrera se han realizado otras contribuciones en el campo de la Recuperación de Información Musical (MIR), desarrollando dos sistemas en las tareas de similitud de audio musical e identificación de versiones musicales. El sistema de identificación de versiones musicales ha servido de base para la generación del material utilizado en las prácticas de la asignatura Tecnologías de Audio, de 4º curso del Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación de la Universidad Autónoma de Madrid.

Palabras clave

Segmentación de audio, Entropía cromática, Recuperación de Información Musical, Gaussian Mixture Models (GMM), Universal Background Model (UBM), Adaptación MAP.

Abstract

The target of this project is to develop a system capable of identifying and segmenting audio radio at different acoustic classes using musical features.

There has been performed a study in the state of the art in the field of audio segmentation, analysing the algorithms and techniques most used as well as the databases most influential in the literature. The developed algorithm uses statistical models based on Gaussian Mixtures Models (GMM-UBM) using features based on spectral chromatic entropy, extracted from the audio database provided by the ALBAYZIN 2010 evaluation in audio segmentation.

The implemented system is divided into seven sub-tasks, identifying a different type of audio per task. Among these sub-tasks we can found a discriminator system between voice and music or a voice activity detector. The results have been compared and merged with the system presented by the research group ATVS in evaluation of audio segmentation ALBAYZIN 2010. Even with lower yields, thanks to the merger, we can improve the overall performance of both systems.

During the execution of this final project there has been made other contributions in the field of Music Information Retrieval (MIR), developing two systems in audio music similarity and audio cover song identification. The audio cover song identification system has been the basis for the generation of the material used in the practices of Tecnologías de Audio course, 4th year of the Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación of the Universidad Autónoma de Madrid.

Key words

Audio segmentation, Chromatic entropy, Music Information Retrieval, Gaussian Mixture Models (GMM), Universal Background Model (UBM), MAP adaptation.

Agradecimientos

Me gustaría agradecer en primer lugar a mi tutor Daniel Ramos la oportunidad que me dio de poder hacer el proyecto fin de carrera bajo su supervisión. Su pasión por la música es algo que me llamó desde el principio y supe que mi proyecto tenía que tratar sobre "el arte de las musas". Su ayuda ha sido inconmensurable, echándome una mano cuando más atascado estaba, o resolviendo mis dudas a la velocidad del rayo.

También tengo que agradecer todo el apoyo recibido por la gente del ATVS, ya que sin ellos este proyecto no habría salido adelante. Especial mención para Javier Franco, Iván Gómez, Alicia Lozano y Alfredo Serrano, a los que he atosigado a preguntas y siempre han tenido la paciencia de escucharme. Gracias también a los directores del ATVS, Javier Ortega y Joaquín González, por permitirme formar parte de uno de los grupos de investigación más punteros en el tratamiento de señales.

Del mismo modo no puedo olvidarme de mis compañeros de carrera, que ya forman parte de mi vida. Con ellos he compartido el estrés de estar días enteros en la escuela sin ver la luz del sol o de apurar al máximo para entregar una práctica. A mis compañeros de prácticas, Helia Relaño, Eva Morodo y Rodrigo López, tengo que darles las gracias por aguantarme durante todo este tiempo, y que no desesperasen por mis manías. Gracias también al resto de toligos: Jaime Mateo, Sara García-Mina, Berta Lorenzo, Sandra Uceda, María Lucena y Fátima García.

No puedo olvidarme de mis amigos de toda la vida, los que siempre han estado ahí, los que no hace falta ni nombrar, pues cada uno sabe que siempre me acuerdo de ellos. Darles las gracias se quedaría corto. Me siento muy orgulloso de tener amigos como ellos.

Mi familia lo es todo para mí. Gracias a mi hermano siempre tengo a alguien con el que es imposible aburrirse. Mi madre es alguien con la que siempre puedes hablar de lo que sea, siempre está cuando la necesitas. Y gracias a mi padre soy lo que soy. Sin él no hubiera estudiado esta carrera, sin él no me apasionaría la tecnología. Sin él no tendría el afán de descubrir algo nuevo cada día. Gracias a todos.

Ricardo Landriz Lara
Septiembre 2014

Este proyecto ha sido realizado en el Área de Tratamiento de Voz y Señales (ATVS) en la Escuela Politécnica Superior de la UAM. El proyecto ha sido financiado parcialmente por el Ministerio de Educación, Cultura y Deporte a través de una beca de colaboración.

A mi padre.

INDICE DE CONTENIDOS

INDICE DE FIGURAS

INDICE DE TABLAS

Glosario de términos

ANN	Artificial Neuronal Network
BIC	Bayesian Information Criterion
EER	Equal Error Rate
EM	Expectation Maximization
FAR	False Acceptance Rate
FFT	Fast Fourier Transform
FRR	False Rejection Rate
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
IR	Information Retrieval
LPC	Linear Prediction Coefficients
LPCC	Linear Prediction Cepstral Coefficients
LR	Likelihood Ratio
MAP	Maximun A Posteriori
MFCC	Mel-Frequency Cepstral Coefficients
MIR	Music Information Retrieval
ML	Maximun Likelihood
PLP	Perceptual Linear Prediction
SDC	Shifted Delta Coefficients
SER	Segmentation Error Rate
UBM	Universal Background Model
ZCR	Zero-Crossing Rate

1 Introducción

1.1 Motivación

La segmentación automática de contenido audiovisual en diferentes clases supone un gran valor añadido del propio contenido, lo que incrementa su utilidad enormemente. De hecho, hoy en día es un campo muy explotado por grandes empresas y multinacionales distribuidoras de contenidos de entretenimiento, como ocurre en la televisión o la radio.

Algunos ejemplos de estas técnicas son la detección automática de cuándo se está emitiendo una película por un determinado canal de televisión o, por ejemplo, cuándo está sonando música en un programa radiofónico. La utilidad de estos sistemas radica en el análisis de la propia señal recibida para luego determinar qué tipo de información se está recibiendo y actuar en función de ello. Algunas aplicaciones basadas en esta tecnología son, por ejemplo, sistemas de recomendación de música o sistemas que ayudan a los reconocedores de habla típicos, entre otros.

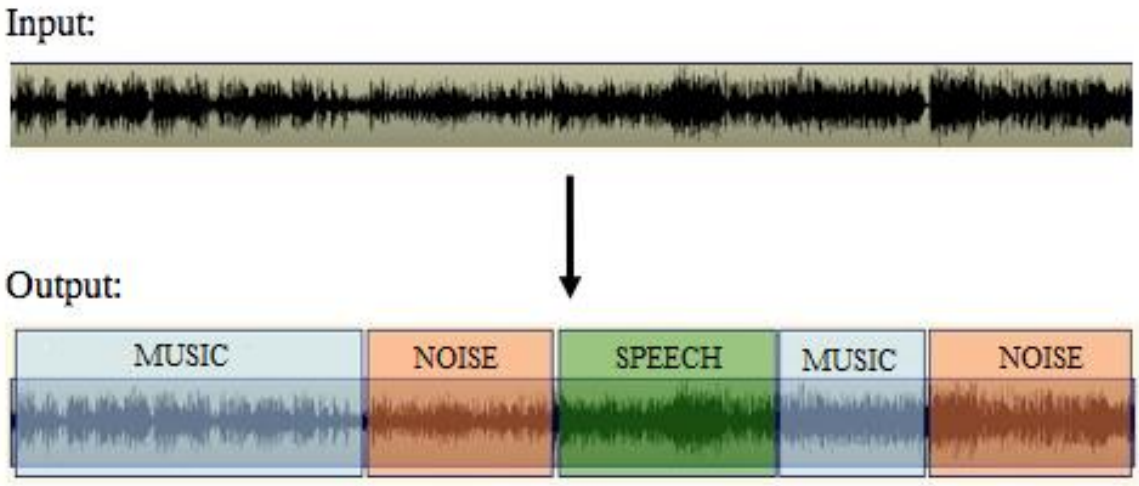


Figura 1-1: Ejemplo de segmentación de audio

Muchas de estas aplicaciones se pueden englobar dentro de los sistemas de Recuperación de Información (Information Retrieval, IR). Centrándose exclusivamente en los sistemas segmentadores de audio existen multitud de técnicas capaces de diferenciar con un gran nivel de acierto distintos tipos de audio. Estas técnicas van desde los análisis de características tímbricas, típicamente usados en reconocimiento de voz e identificación de locutor, hasta el uso de características menos conocidas, como son las características cromáticas, muy utilizadas en los sistemas de Recuperación de Información Musical (Music Information Retrieval, MIR). Todas estas técnicas derivan de en una de las ramas principales de la ingeniería moderna, la Teoría de la Señal.

No obstante, los sistemas segmentadores de audio actuales difieren poco de los tradicionales identificadores de locutor, ya que se siguen utilizando técnicas desarrolladas hace más de quince años, como son los Mel-Frequency Cepstral Coefficients (MFCC), el Zero-Crossing Rate (ZCR), los Gaussian Mixture Model con Universal Background Model (GMM-UBM) o los Hidden Markov Model (HMM). Ha sido en los últimos años, y gracias a multitud de congresos y evaluaciones internacionales que se llevan celebrando desde mediados de los 2000, cuando se han empezado a utilizar otro tipo de técnicas distintas y más innovadoras, pues centrándose en estudios tímbricos (MFCC) se pierde una gran cantidad de información [Aucouturier & Pachet, 2004]. Algunas de estas técnicas son las características cromáticas, basadas en el tempo y el ritmo del audio.

1.2 Objetivos

Partiendo del interés general de la comunidad científica en el análisis de señales de audio con el fin de extraer información adicional útil para el usuario y más aun basándose en la reciente predilección de los grupos de investigación de todo el mundo por el uso de técnicas alternativas en dicho análisis, este proyecto tiene como objetivo el poder segmentar audio radiofónico en distintas clases acústicas utilizando para ello características musicales.

De tal manera el proyecto consta de varios objetivos:

- Estudiar los sistemas actuales de segmentación de audio, así como las técnicas utilizadas en el estado del arte para tal propósito.
- Implementación de un sistema capaz de segmentar audio radiofónico en distintas clases acústicas, utilizando para ello características cromáticas.
- Fusión del sistema desarrollado con el presentado por el grupo ATVS de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid en la evaluación ALBAYZIN de Audio Segmentation de 2010, basado en características tímbricas.

Un objetivo secundario del proyecto es el desarrollar material para las prácticas de la asignatura de 4º curso del Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación de la Universidad Autónoma de Madrid, Tecnologías de Audio, lo que ha implicado un estudio exhaustivo de los sistemas de Recuperación de Información Musical (MIR), así como de las bases de datos más utilizadas en este campo.

1.3 Metodología y plan de trabajo

El desarrollo del proyecto se divide en las siguientes tareas:

- Documentación: En una primera fase, el estudiante ha llevado a cabo un amplio estudio de la literatura sobre el estado del arte actual en técnicas de segmentación de audio (extracción de características, modelado, scoring…), así como documentación sobre las bases de datos más utilizadas en esta tarea.
- Estudio del software: En una segunda fase, el estudiante se ha familiarizado con la estructura de servidores de la que dispone el grupo y los paquetes de funciones necesarios para el desarrollo de los experimentos.
- Experimentos y desarrollo del software: Posteriormente, se han realizado experimentos haciendo uso de las bases de audio anteriormente mencionadas. Todo el código desarrollado se ha organizado para su uso posterior.
- Evaluación de resultados y elaboración de la memoria: Se ha realizado un análisis de los resultados obtenidos a partir de las pruebas realizadas. Con los resultados obtenidos y los respectivos análisis realizados, se ha procedido a redactar la presente memoria.

El proyecto sigue el siguiente plan de trabajo, considerando 13 meses (sin contar el mes de agosto) de M1 a M13. El proyecto presentado defiende un sistema de segmentación de audio, llevado a cabo desde el M7 al M13, mientras que los primeros meses se emplearon en distintas técnicas MIR con el objetivo de profundizar en este campo de la Teoría de la Señal a la par de generar material para las prácticas de la asignatura de Tecnologías de Audio:

1. Familiarización con el entorno de experimentos del grupo de investigación (Área de Tratamientos de Voz y Señales, ATVS), las bases de datos con las que se trabajará y las herramientas de desarrollo (M1 a M3 y M7 a M8).
2. Investigación acerca del estado del arte en tratamiento de señales musicales y sus respectivos post-procesados para sistemas de MIR (M1 a M3).
3. Adaptación de las bases de datos y crear las herramientas de análisis oportunas que permitan comparar las mejoras de los distintos sistemas de MIR (M3 a M4).
4. Elaboración e implementación de los distintos sistemas MIR brevemente comentados (M4 a M5).
5. Desarrollo de material para las prácticas de la asignatura de 4º curso del Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación de la Universidad Autónoma de Madrid, Tecnologías de Audio (M6).

6. Realización de una batería de pruebas que avalen la adecuación de los sistemas MIR elegidos para un correcto sistema de recuperación musical (M6 a M7).
7. Investigación acerca del estado del arte en sistemas segmentadores de audio, tanto de las técnicas más utilizadas como las bases de datos más comúnmente usadas (M7 a M8).
8. Elaboración e implementación de un sistema capaz de segmentar audio radiofónico en distintas clases acústicas (M8 a M11).
9. Realización de una exhaustiva batería de pruebas que mida la bondad y el rendimiento del sistema implementado (M11 a M12).
10. Se analizarán los resultados a lo largo de todo el proyecto, extrayendo las conclusiones pertinentes (M12).
11. Se realizará una memoria detallada que resuma el trabajo realizado a lo largo del proyecto (M12 a M13).

1.4 Organización de la memoria

El presente trabajo se divide en seis capítulos:

- **Capítulo 1: Introducción.** En esta sección se presenta tanto la motivación para el desarrollo del proyecto como los objetivos que se persiguen.
- **Capítulo 2: Estado del arte.** En este apartado se recoge el estado del arte actual en el procesado de voz y en el tratamiento de señales de audio, así como diferentes métodos estadísticos que parametrizan las características extraídas del audio. También se estudian diferentes sistemas basados en segmentación de audio además de un breve repaso al mundo de la Recuperación de Información Musical. Adicionalmente, se presentan las herramientas que permiten la medida del rendimiento del sistema y hacer posible su comparación con otros trabajos.
- **Capítulo 3: Sistema propuesto.** En esta sección se presenta la descripción del sistema desarrollado junto con un análisis de la base de datos empleada. Además, se describe la estrategia de fusión del sistema implementado con el presentado por el grupo ATVS en la evaluación ALBAYZIN de Audio Segmentation de 2010.
- **Capítulo 4: Análisis de resultados.** En este capítulo se detalla el análisis de los parámetros utilizados por el sistema, elegidos a partir de una exhaustiva etapa de desarrollo destinada a optimizar el rendimiento del mismo. También se realiza un estudio de los resultados obtenidos, comparándose con otros sistemas similares, como es el presentado por el grupo ATVS en ALBAYZIN 2010.

- **Capítulo 5: Otras contribuciones realizadas.** En este apartado se estudian otros trabajos realizados durante la ejecución del proyecto, como son dos sistemas basados en Recuperación de Información Musical. El primer sistema trata sobre la similitud entre audios musicales, mientras que el segundo tiene como objetivo la identificación de versiones musicales. Adicionalmente, se describen distintas bases de datos utilizadas, además de los diferentes problemas que suscitan este tipo de sistemas. También se recoge y explica el material generado para las prácticas de la asignatura Tecnologías de Audio.
- **Capítulo 6: Conclusiones y Trabajo futuro.** En este capítulo se presentan las conclusiones extraídas del proyecto realizado, trabajo aportado y las futuras líneas a seguir en este ámbito.

2 Estado del arte

2.1 Introducción

En este capítulo se presenta el estado del arte en el procesado de voz y en el tratamiento de señales de audio, al igual que en los sistemas de Recuperación de Información Musical, prestando mayor atención a aquellos relacionados con la segmentación de audio.

El capítulo comienza con un repaso a las principales técnicas en tratamiento de señales de audio, muy utilizadas en el campo de la Teoría de la Señal. También se lleva a cabo un exhaustivo estudio sobre los sistemas de Recuperación de Información Musical, utilizados como base para el material generado para las prácticas de la asignatura Tecnologías de Audio.

En la parte final del capítulo se introducen los sistemas segmentadores de audio, basados en tecnologías del tratamiento de señales de audio. Estos sistemas serán la base del proyecto y del sistema implementado.

2.2 Procesado de voz y Tratamiento de señales de audio

Hoy en día existe una gran cantidad de aplicaciones capaces de interactuar con las personas mediante el reconocimiento y síntesis de voz. Estas van desde aplicaciones simples en el reconocimiento de comandos (palabras) aislados, hasta el reconocimiento de frases para ejecutar acciones de todo tipo. Estas tecnologías se utilizan en el día a día en multitud de lugares y situaciones: escritura dictada en teléfonos móviles, control por voz de sistemas mecánicos, acceso a servicios de compra por teléfono, reservas de viajes, etc.

La voz tiene las ventajas de su gran aceptabilidad y facilidad de adquisición. Esta señal lleva información consciente, inteligente y producida por los humanos para que las personas que la escuchan obtengan información directa, sin la necesidad de otra fuente adicional como imágenes o texto. Es la forma universal de comunicación entre las personas.

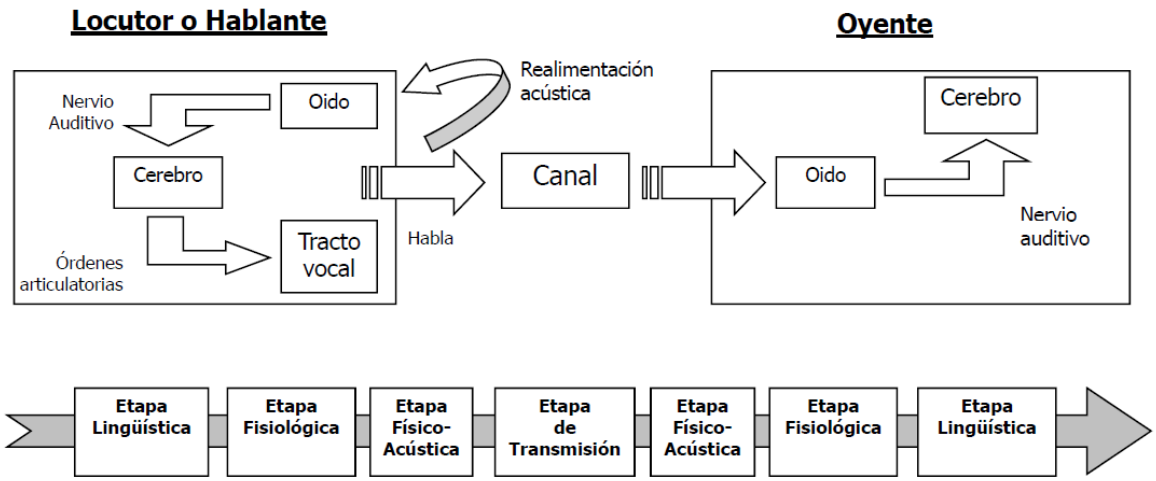


Figura 2-1: Etapas en la comunicación hablada [Ortega, 2012]

El procesado de voz pertenece a una rama de la ciencia más generalista, el tratamiento de señales de audio, donde se estudian fragmentos de audio sin importar el contenido, ya sea voz, música, sonidos ambientales o simplemente ruido. Algunas de las aplicaciones que más se repiten en la literatura son, por ejemplo, los detectores de actividad de voz o los segmentadores de audio en distintas clases acústicas, entre otras.

2.2.1 Procesado de audio a corto plazo

La señal de audio, en general, se compone de constantes fluctuaciones a lo largo de un fichero de audio, por lo que se define como una señal no estacionaria, es decir, sus propiedades estadísticas varían a lo largo del tiempo. Sin embargo, realizando un análisis de tramos de menor duración (decenas de ms) se aprecia un comportamiento cuasi-estacionario, lo que facilita enormemente el procesado de la información.

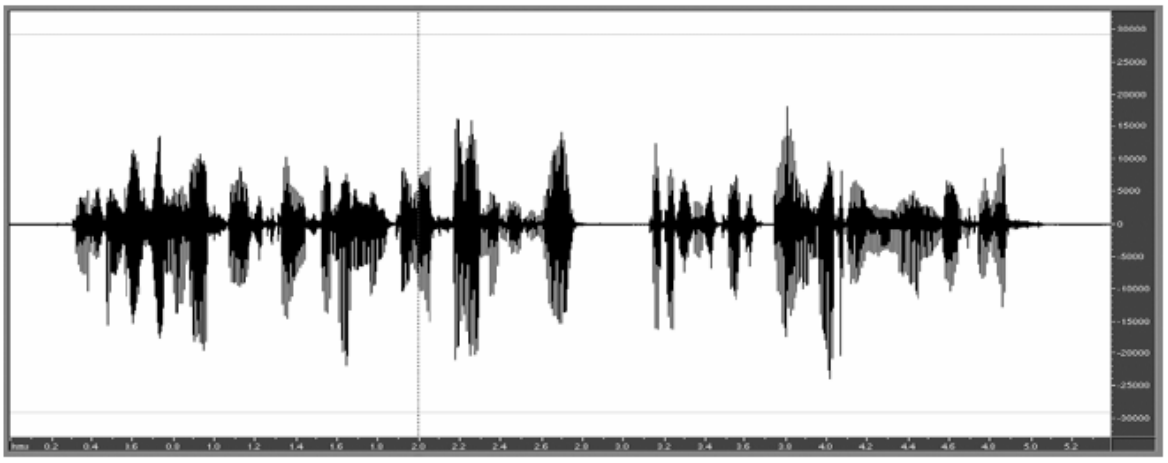


Figura 2-2: Locución de 5 s de duración [Ortega, 2012]

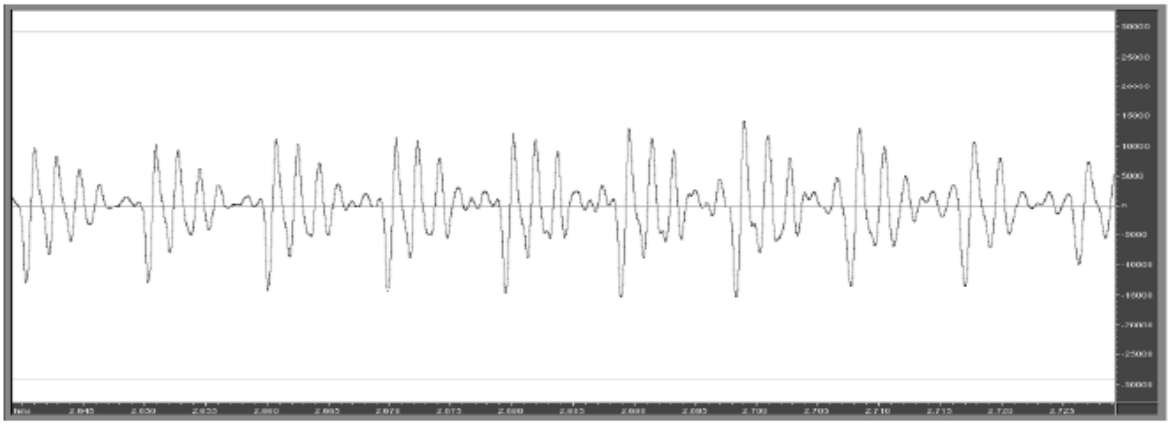


Figura 2-3: Forma de onda de una vocal con duración de 80 ms [Ortega, 2012]

Puesto que la voz en general y el audio en particular son señales pseudo-estacionarias sólo a corto plazo (decenas de ms), para poder aplicar técnicas de análisis y procesado se debe limitar el audio en segmentos de la duración reseñada. Típicamente se conoce a estos segmentos como tramas de voz, o más genéricamente, tramas de audio.

2.2.2 Algoritmos y técnicas en clasificación de audio

Los sistemas de clasificación de audio engloban multitud de sistemas con multitud de aplicaciones, desde el reconocimiento de locutores hasta segmentadores de audio. El reconocimiento de audio conlleva un procesado que permite extraer un conjunto de rasgos inherentes al propio audio y la posterior búsqueda de posibles similitudes mediante un proceso de reconocimiento de patrones.

Un sistema de reconocimiento de audio está formado por dos secciones: entrenamiento y test. Tienen una función bien diferenciada:

- La sección de entrenamiento tiene la finalidad de registrar uno o varios ficheros de audio para extraer sus características y guardarlas en la base de datos.
- La sección de test se centra en registrar el audio de entrada y extraer las características para poder compararlas con las que se encuentran almacenadas en la base de datos. Dependiendo del modo de funcionamiento se procederá de una u otra manera. En un modo de funcionamiento hipotético, después de obtener posibles coincidencias, el sistema podría presentar el audio susceptible de ser el buscado, de entre todos los audios de la base de datos. Algunos casos prácticos son la búsqueda de distintos tipos de audio (Shazam®) o clasificadores de audio, como discriminadores voz/música.

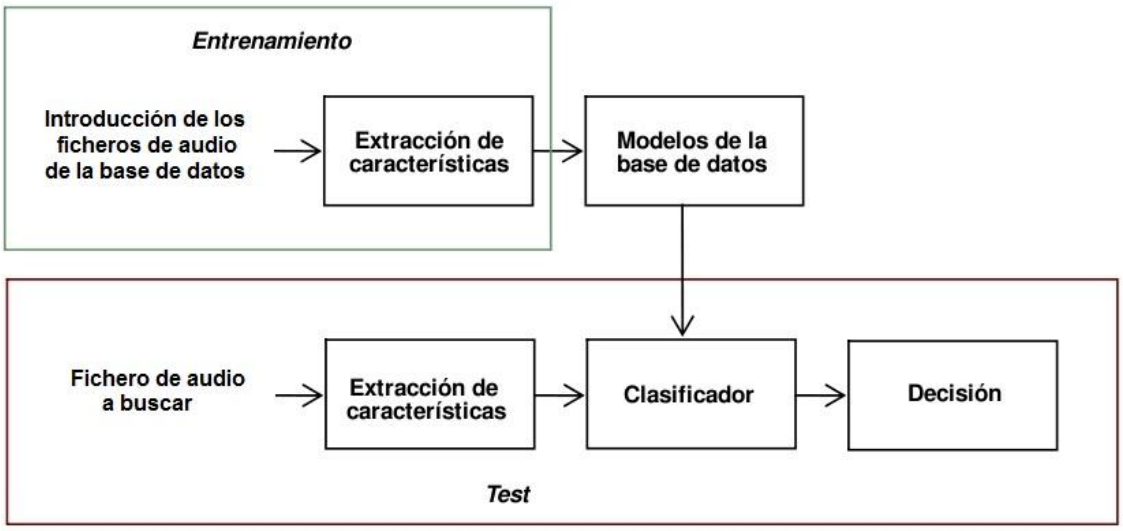


Figura 2-4: Arquitectura de un sistema de reconocimiento de audio [Ortega, 2012]

El mecanismo que permite, dada una señal de audio, realizar un análisis localizado mediante el uso de tramas consecutivas se denomina enventanado de la señal. Este enventanado se define como la aplicación (multiplicación) sobre la señal de voz completa de una función limitada en el tiempo (ventana), lo que produce una nueva señal de voz, cuyo valor fuera del intervalo definido por la ventana es nulo. Se puede expresar como:

$$x_m[n] = x[n] \cdot w[n-m] \tag{2.1}$$

siendo $x[n]$ la señal de audio original, $w[n]$ la ventana temporal aplicada y $x_m[n]$ la trama de señal enventanada, que valdrá cero fuera del intervalo $n \in [m, m+N-1]$, siendo N la duración en muestras de la ventana aplicada y m el desplazamiento temporal con el que se aplica el enventanado.

De entre todas las ventanas posibles, en el procesado de voz destacan dos tipos de ventanas:

- La ventana rectangular, que vale uno dentro del intervalo y cero fuera:

$$w[n] = \begin{cases} 1, & 0 \le n \le N-1 \\ 0, & en\ caso\ contrario \end{cases} \tag{2.2}$$

- La ventana tipo Hamming, cuya estructura temporal está definida de la siguiente forma (ponderación tipo coseno alzado):

$$w[n] = \begin{cases} 0.54 - 0.46 \cdot cos\left(2\pi n / (N-1)\right), & 0 \le n \le N-1 \\ 0, & en\ caso\ contrario \end{cases} \tag{2.3}$$

A partir del enventanado se extraen diferentes parámetros (características) que caracterizarán el procesado de audio mediante distintos métodos y modelos, algunos de los cuales se describirán más en detalle en las siguientes secciones del presente capítulo. Algunas de las características más utilizadas son las siguientes:

- LPC (Linear Prediction Coefficients), Coeficientes de Predicción Lineal [Atal & Hanauer, 1971]: En sistemas de procesado de voz, se usa partiendo de la idea de que la voz puede modelarse como una combinación lineal de p muestras anteriores más una señal de error. Modelan la envolvente espectral del sonido a corto plazo.
- LPCC (Linear Prediction Cepstral Coefficients), Predicción Lineal de los Coeficientes Cepstrales [Atal, 1974]: Se aplica una transformación cepstral a partir de los coeficientes LPC.
- MFCC (Mel-Frequency Cepstral Coefficients), Coeficientes Cepstrales en las Frecuencias de Mel [Davis & Mermelstein, 1980]: Son coeficientes para la representación del habla basados en la percepción auditiva humana. Cada vez más, se empiezan a utilizar en otras aplicaciones en el campo de la Recuperación de Información Musical, como por ejemplo la clasificación de géneros, medidas de similitud de audio, etc. También modelan la envolvente espectral del sonido a corto plazo, y se describirán más adelante.

Una vez parametrizada la señal de audio se modelan las características calculadas mediante diferentes esquemas. Entre los modelos más utilizados, algunos de los cuales se describirán en detalle más adelante, destacan:

- GMM (Gaussian Mixture Models), Modelos de Mezcla de Gaussianas [Reynolds & Rose, 1995] [Reynolds et al., 2000]: Permite modelar datos de forma estadística de forma flexible, ajustándose bien a datos que no sigan una distribución paramétrica clara.
- UBM (Universal Background Models) [Reynolds et al., 2000]: Cuando hay pocos datos (poca habla) el modelo GMM puede ser ineficaz, por lo que es necesario la introducción de un modelo universal que añada generalidad al sistema.
- HMM (Hidden Markov Models), Modelos Ocultos de Markov [Rabiner, 1989]: Modelan estadísticamente la acústica de la voz introduciendo contexto temporal. En la actualidad son la base tecnológica de los reconocedores de voz comerciales.
- ANN (Artificial Neuronal Networks), Redes Neuronales Artificiales [Pearson & Lipman, 1988]: Modelos matemáticos construidos basándose en el funcionamiento de las redes neuronales biológicas. Muy utilizadas en cualquier tarea de clasificación de patrones.

2.2.2.1 Extracción de características: Coeficientes MFCC (Mel-Frequency Cepstral Coefficients)

En 1963, Bogert, Healy y Tukey publicaron el artículo "The Quefrency Analysis of Time Series for Echoes: Cepstrum, Pseudoautocovariance, Cross-Cepstrum, and Saphe Cracking" en el que observaron que el logaritmo del espectro de potencia de una señal contenía un eco en forma de componente aditiva periódica debida a ese eco y, por tanto, el espectro de potencia del logaritmo de espectro de potencia debería mostrar un pico en el retardo correspondiente al eco. Denominaron cepstrum, a esta función, intercambiando las letras de la palabra spectrum.

Los métodos de análisis basados en el cepstrum han encontrado una amplia aplicación en problemas de tratamiento de voz, como la identificación de locutores [Atal, 1974], la verificación de locutores [Furui, 1981] y el reconocimiento de voz [Davis & Mermelstein, 1980].

Los coeficientes Mel-Frequency Cepstral Coefficients (MFCC) fueron introducidos por Davis y Mermelstein en 1980 originalmente para el reconocimiento de habla y posteriormente adaptados para otras tareas como reconocimiento de locutor, identificación de canciones, segmentación de audio, etc. El proceso de extracción de estos coeficientes consta de diferentes etapas.

En primer lugar y tal y como ya se adelantó en el apartado 2.2.1., es necesario dividir la señal de audio en tramas de duración del orden de las decenas de milisegundos (típicamente se utilizan tramas de 20 ms) para su posterior procesado individual. Para evitar la posible pérdida de información en la transición entre dos tramas contiguas se suele realizar un solapamiento, normalmente del 50 % (Figura 2-5).

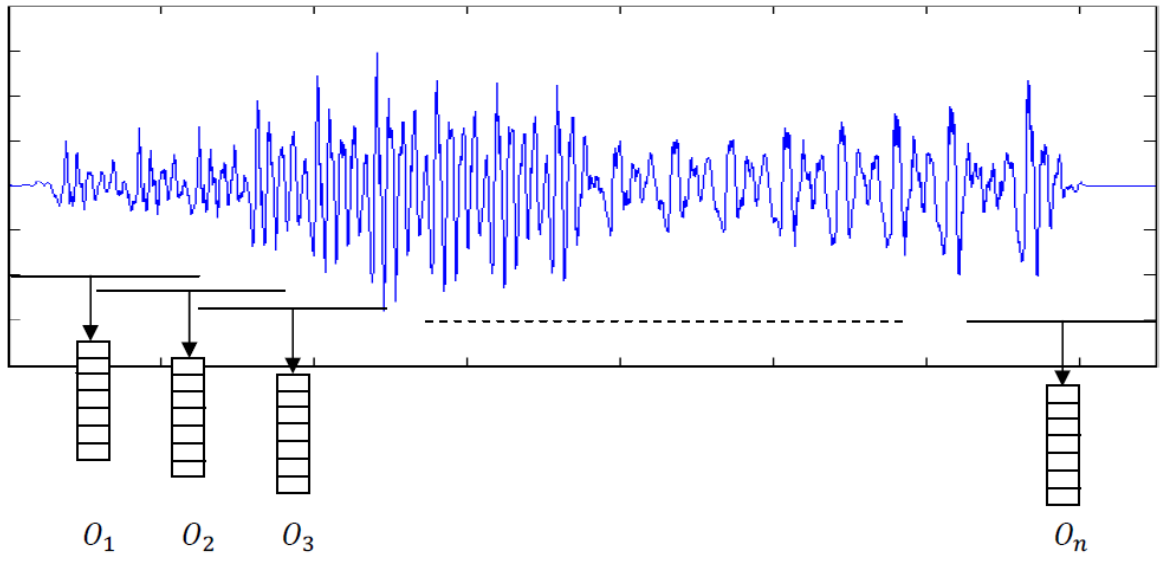


Figura 2-5: División de la señal de audio en tramas para la extracción de características

Posteriormente al enventanado aplicado a cada una de las tramas (normalmente de tipo Hamming) se procede al cálculo de la FFT (Fast Fourier Transform). Normalmente sólo se guarda la amplitud del espectro obtenido. La información de dicha envolvente se recoge mediante un banco de filtros perceptual en escala Mel. El objetivo de este filtrado es aproximar la resolución espectral a la respuesta del oído humano mediante la siguiente relación (donde f_l es la frecuencia de entrada en escala lineal):

$$f_{Mel} = 1127.01048 \cdot \log_e\left(1 + \frac{f_l}{700}\right) \tag{2.4}$$

A la salida de los filtros, que integran la energía existente de la señal de audio dentro de su ancho de banda, se le aplica el logaritmo natural y luego la transformada discreta del coseno (Discrete Cosine Transform, DCT) con el objetivo de comprimir la información en pocos coeficientes. De las salidas de los filtros, denotadas mediante $Y(m), m = 1, \dots, M$, los coeficientes MFCC se obtienen a partir de la siguiente transformación:

$$C_n = \sum_{m=1}^{M} [\ln Y(m)] cos\left[\frac{\pi \cdot n}{M}\left(m - \frac{1}{2}\right)\right] \tag{2.5}$$

donde n es el índice del coeficiente cepstral. El vector de características final se forma típicamente con los 12 o 20 primeros coeficientes C_n.

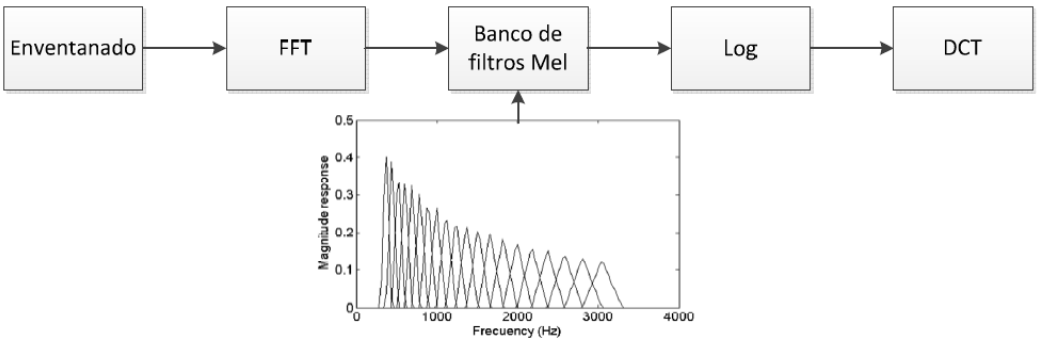


Figura 2-6: Proceso de obtención de los coeficientes MFCC [Gómez, 2014]

2.2.2.2 Modelado de audio mediante Modelos de Mezclas de Gaussianas (GMM)

Los sistemas basados en Modelos de Mezclas de Gaussianas (Gaussian Mixture Models, GMMs) [Reynolds & Rose, 1995] [Reynolds et al., 2000] han sido, durante muchos años, unas de las técnicas de referencia para los sistemas de clasificación de audio y voz [Aucouturier & Pachet, 2004] y una parte indispensable de los sistemas modernos de reconocimiento de locutores basados en supervectores y vectores de identidad (i-Vectors) [Dehak et al., 2011].

En GMM, dado un modelo estadístico λ de un fragmento de audio de identidad conocida o conjunto de fragmentos, la probabilidad de que un vector cepstral de evaluación $\overrightarrow{y_t}$ pertenezca a dicho modelo es representado mediante una combinación lineal de distribuciones de probabilidad gaussianas D-dimensionales (D es la dimensión del vector de características $\overrightarrow{y_t}$):

$$p(\overrightarrow{y_t}|\lambda) = \sum_{i=1}^{M} w_i p_i(\overrightarrow{y_t}) \tag{2.6}$$

donde M representa el número de componentes gaussianas, w_i sus correspondientes pesos sujetos a la restricción de que $\sum_{i=1}^{M} w_i = 1$ y $p_i(\overrightarrow{y_t})$ se expresa de la siguiente forma:

$$p_i(\overrightarrow{y_t}) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} e^{-\left(\frac{1}{2}\right)(\overrightarrow{y_t}-\overrightarrow{\mu_i})' \Sigma (\overrightarrow{y_t}-\overrightarrow{\mu_i})_i^{-1}} \quad i = 1, \dots, M \tag{2.7}$$

Es decir, cada componente gaussiana del modelo del fragmento de audio/voz o del modelo alternativo tendrá su correspondiente vector de medias y matriz de covarianza, de tal forma que se pueda identificar al modelo mediante la notación $\lambda = \{w_i, \vec{\mu}_i, \Sigma_i\},\ i = 1, \dots, M$. En cuanto a la matriz de covarianza, existen multitud de trabajos que demuestran de manera experimental que el usar la diagonal de dicha matriz da resultados prácticamente idénticos a si se usase la matriz completa, pero con el consiguiente considerable ahorro en coste computacional. El principal motivo que permite usar este tipo de matrices radica en la naturaleza ortogonal de los coeficientes cepstrales MFCC, que otorga una gran independencia entre las diferentes dimensiones. Además, con una gran cantidad de componentes gaussianas con matriz de covarianza diagonal se puede aproximar una función densidad de probabilidad gaussiana con matriz de covarianza completa.

Para entender con más claridad el concepto de GMM, se puede observar el ejemplo de la siguiente figura (Figura 2-7). En la parte de la izquierda se encuentra el histograma correspondiente a la primera componente de los vectores de características D-dimensionales de un fichero de entrenamiento de un modelo GMM. Se representa una única componente para facilitar la visualización. A continuación se puede verificar cómo con 4 o 32 componentes gaussianas y con sus respectivos pesos se reproduce dicha distribución, es decir, se genera la distribución estadística de la primera componente cepstral de dicho locutor.

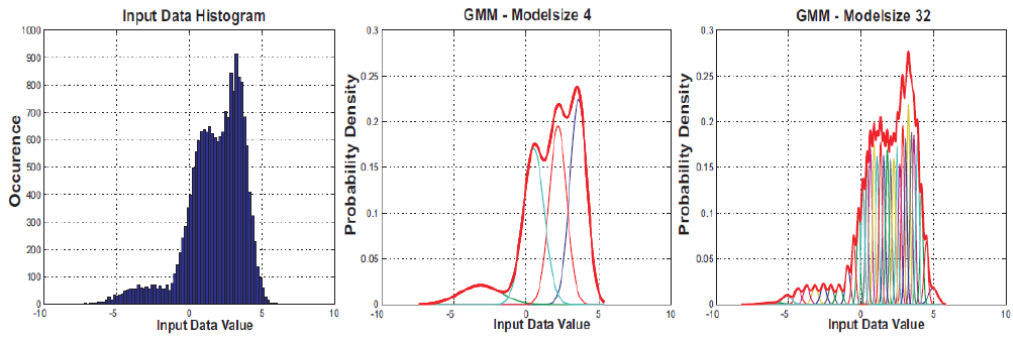


Figura 2-7: GMM para un conjunto de entrenamiento con datos univariados. Histograma (izquierda) junto con dos modelos de diferente número de componenetes gaussianas (4 y 32, respectivamente) [Robin Harald Priewald, 2009]

Los Modelos de Mezcla de Gaussianas pueden tener tantas dimensiones como el sistema requiera. A continuación se ilustra el mismo ejemplo anterior pero con las dos primeras componentes de los vectores de características. De tal forma, en este caso se está trabajando con un vector bidimensional:

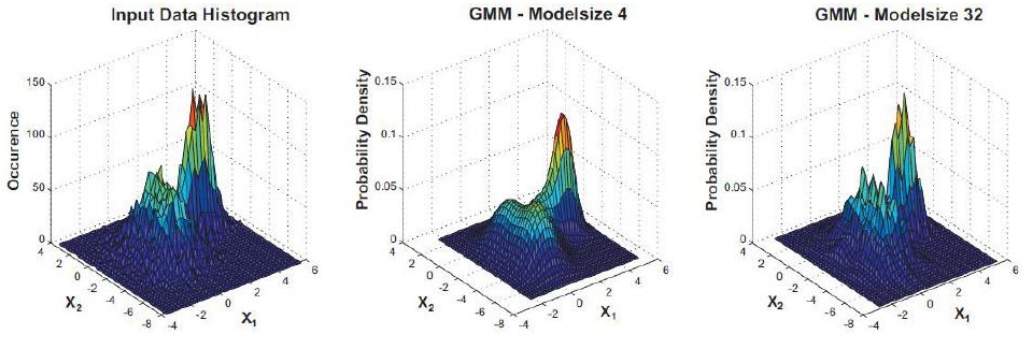


Figura 2-8: GMM bidimensional para datos de dos dimensiones. Histograma (izquierda) junto con dos modelos de diferente números de componentes gaussianas (4 y 32, respectivamente) [Robin Harald Priewald, 2009]

El entrenamiento de un GMM consiste en asignar los parámetros de un modelo $\lambda = \{w_i, \vec{\mu}_i, \Sigma_i\},\ i = 1, \dots, M$ a partir de datos de entrenamiento $X = \{\vec{x}_1, \dots, \vec{x}_T\}$ y, así, intentar ajustar la distribución del modelo a la de los vectores de características de entrenamiento. Para ajustar la distribución del GMM a los vectores de características se hace uso del criterio de máxima verosimilitud (Maximun Likelihood, ML) que pretende asignar los parámetros que maximicen la verosimilitud del GMM dado los datos de entrenamiento.

Para una secuencia de datos $Y = \{y_1, \dots, y_T\}$ y asumiendo independencia entre los vectores de características:

$$p(Y|\lambda) = \prod_{t=1}^{T} p(\overrightarrow{y_t}|\lambda) \tag{2.8}$$

Sin embargo, se puede realizar una aproximación mediante el algoritmo de Expectation Maximization (EM) [Duda et al., 2000], un algoritmo que va cambiando iterativamente los parámetros del GMM, con un procedimiento muy similar al algoritmo Baum-Welch utilizado en los Modelos Ocultos de Markov (HMM). De esa forma, el algoritmo comienza con un modelo inicial λ y estima un nuevo modelo $\bar{\lambda} = \{\overline{w_i}, \overrightarrow{\bar{\mu}_i}, \overline{\Sigma_i}\}$, $i = 1, \dots, M$, de modo que $p(Y|\bar{\lambda}) \geq p(Y|\lambda)$.

El nuevo modelo se convierte en el modelo inicial en la siguiente iteración y el proceso continúa hasta que el valor de la verosimilitud converge o hasta que se alcance un determinado número de iteraciones.

Por otra parte, puede usarse el método K-means para estimar el modelo inicial λ, de forma que se necesiten menos iteraciones del algoritmo EM. De este modo, los centroides calculados determinarían los vectores de medias del GMM, la covarianza de los vectores del conjunto Y asignados a cada centroide determinarían las matrices de covarianza de cada gaussiana y los pesos estarían determinados por el porcentaje de vectores del conjunto Y asignados a cada centroide. Una vez obtenido el modelo final λ, mediante el mismo procedimiento, se calcula la probabilidad del conjunto de vectores de un fragmento de audio frente al modelo.

2.2.2.2.1 GMM-UBM

La técnica GMM-UBM [Reynolds et al., 2000] propone que el modelo alternativo (identificado como λ), definido como aquel modelo que teniendo un segmento de audio a evaluar es necesario definir con qué clase acústica de un grupo de test tiene mayor nivel de semejanza, es generado a través de un conjunto de segmentos pertenecientes a muchas clases acústicas de tal forma que seamos capaces de generar un modelo GMM universal (Universal Background Model, UBM). Este modelo representa la distribución independiente de una determinada clase acústica de todos los vectores de características, es decir, modela las características comunes a todas las clases acústicas. Sería interesante que los segmentos de audio presenten alto grado de variabilidad entre ellos, de tal forma que el modelo represente de manera fiel dicha variabilidad, y por lo tanto pueda ser general ante la llegada de nuevos datos. Cuando se registra una nueva clase en el sistema, los parámetros del UBM se adaptan a la distribución de características de la clase, de forma que el modelo universal o UBM adaptado es el modelo de la clase.

La motivación del uso de esta técnica frente a los GMM clásicos es que busca hacer frente a dos problemas típicos de los sistemas GMM:

- Ser robusto en la escasez de datos que existe en muchas ocasiones a la hora de entrenar un modelo de clases.
- Proporcionar un mecanismo que permita ponderar la puntuación de distintos fragmentos de audio de test y poder compararlos entre sí.

2.2.2.2.2 Adaptación MAP

En la adaptación MAP, el UBM es utilizado como un modelo a priori, y los nuevos parámetros GMM son adaptados de éste de tal manera que el nuevo modelo se ajuste más a los datos de entrenamiento $\boldsymbol{X}$. Los parámetros del UBM que se pueden adaptar son los pesos, los vectores de medias y las matrices de covarianza. Generalmente se adaptan únicamente los vectores de medias [Reynolds et al., 2000], ya que en la mayoría de los casos da el mismo resultado que utilizar todos los parámetros que definen el UBM. No obstante, la adaptación seguida en este proyecto es completa (pesos, medias y covarianzas), pues la adaptación básica infería en graves errores de segmentación (apartado 4.1.1.). Las ecuaciones, muy similares a las que se utilizan en el algoritmo EM son las siguientes:

$$\gamma_i(t) = \frac{w_i p_i(\overrightarrow{x_t})}{\sum_{j=1}^{M} w_j p_j(\overrightarrow{x_t})} \tag{2.9}$$

donde $\gamma_i(t)$ representa la probabilidad de ocupación de la mezcla i-ésima en la trama $\overrightarrow{x_t}$, mientras que w_i y p_i son los parámetros del UBM para la mezcla i-ésima. Luego se necesita calcular los estadísticos de orden cero (para el peso), de orden uno (para la media) y de segundo orden (para la covarianza) de cada mezcla o componente gaussiana:

$$n_i = \sum_{t=1}^{N_x} \gamma_i(t) \quad i = 1, \dots, M \tag{2.10}$$

$$E_i(\vec{x}) = \frac{1}{n_i} \sum_{t=1}^{N_x} \gamma_i(t) \vec{x}_t \quad i = 1, \dots, M \tag{2.11}$$

$$E_i(\vec{x}^2) = \frac{1}{n_i} \sum_{t=1}^{N_x} \gamma_i(t) \vec{x}_t^2 \quad i = 1, \dots, M \tag{2.12}$$

Con los nuevos estadísticos calculados a partir del segmento de entrenamiento, se procede a actualizar cada una de las mezclas gaussianas del UBM mediante adaptación MAP:

$$\hat{w}_i = \left[\frac{\alpha_i n_i}{N_x} + (1-\alpha_i)w_i\right]\eta \tag{2.13}$$

$$\hat{\vec{\mu}}_i = \alpha_i E_i(\vec{x}) + (1-\alpha_i)\vec{\mu}_i \tag{2.14}$$

$$\hat{\Sigma}_i = \alpha_i E_i(\vec{x}^2) + (1-\alpha_i)(\Sigma_i + \vec{\mu}_i^2) - \hat{\vec{\mu}}_i^2 \tag{2.15}$$

donde $\alpha_i = \frac{n_i}{n_i+r}$ es el cociente de adaptación, r es el factor de relevancia (relevance factor) [Huai-You et al., 2012] que debe ser ajustado en función de la cantidad de audio disponible para adaptar el UBM y η es el factor de normalización para el vector de pesos $\hat{w}_i$ de cada clase acústica. En las ecuaciones anteriores, $\{w_i, \vec{\mu}_i, \Sigma_i\}$ y $\{\hat{w}_i, \hat{\vec{\mu}}_i, \hat{\Sigma}_i\}$ representan los parámetros de la i-ésima mezcla gaussiana para el UBM y para la clase que está siendo entrenada mediante MAP, respectivamente. En la siguiente figura (Figura 2-9) se puede apreciar de manera clara el proceso del entrenamiento MAP.

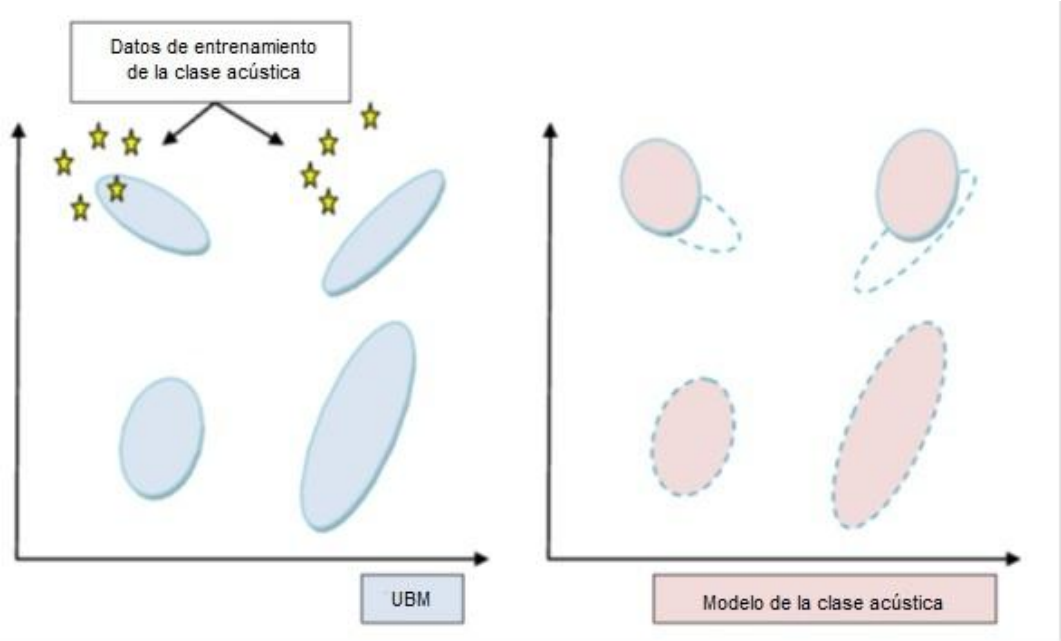


Figura 2-9: Ejemplo de adaptación MAP de una clase acústica [Reynolds et al., 2000]

En la parte izquierda está representado esquemáticamente el UBM (suponiendo que se trabaja solamente con 4 mezclas gaussianas y asumiendo que los vectores cepstrales tienen únicamente 2 dimensiones) además de los vectores cepstrales entrenamiento de una clase acústica de la que se quiere calcular su modelo adaptado del UBM mediante MAP. El proceso a seguir en la adaptación MAP consiste en que las mezclas gaussianas del

UBM que guarden mayor similitud con los vectores de entrenamiento, es decir, aquellas cuyos estadísticos n_i sean relativamente altos (en la Figura 2-9, las dos mezclas gaussianas superiores) sufrirán grandes cambios, en otras palabras, sus parámetros se adaptarán a los datos de entrenamiento. Por otra parte, aquellas mezclas gaussianas con $n_i \approx 0$ no sufrirán cambio alguno con respecto al UBM. En la parte de la derecha de la Figura 2-9 podemos ver el modelo, donde las dos componentes superiores se han visto modificadas con respecto al UBM, mientras que las dos inferiores permanecen inalteradas.

2.2.2.3 Cálculo de puntuaciones en sistemas de clasificación de audio

En esta sección se hará un breve repaso de los principales métodos y herramientas utilizados en la evaluación de sistemas de clasificación de audio, como son los segmentadores de audio. Asimismo se llevará a cabo una descripción de distintas técnicas para mejorar el rendimiento de dichos sistemas

2.2.2.3.1 Razón de Verosimilitudes (Likelihood Ratio, LR)

Un funcionamiento típico en clasificación binaria es el llamado modo de detección, donde dado un conjunto de datos de entrenamiento de entrenamiento ($\boldsymbol{X} = \{\overrightarrow{x_t}\}_{t=1,\dots,N_X}$) y uno de evaluación ($\boldsymbol{Y} = \{\overrightarrow{y_t}\}_{t=1,\dots,N_Y}$), el objetivo es determinar si $\boldsymbol{Y}$ pertenece a la misma clase que $\boldsymbol{X}$. Existen dos posibles hipótesis:

- H_0: los segmentos $\boldsymbol{X}$ e $\boldsymbol{Y}$ pertenecen a la misma clase.
- H_1: los segmentos $\boldsymbol{X}$ e $\boldsymbol{Y}$ no pertenecen a la misma clase.

En segmentación de audio, H_0 suele querer decir que los conjuntos de datos de entrenamiento y evaluación pertenecen a la misma clase acústica, es decir, si el audio de entrada se debería etiquetar como aquel con el que se está comparando (música, voz, ruido…), y H_1 tiene el significado de que la clase acústica del fragmento de audio de evaluación no coincide con la del audio de entrenamiento, como cuando se compara una trama de música con otra de voz.

La toma de decisión entre las dos hipótesis anteriores (H_0, H_1) se podría basar, en un marco bayesiano de decisión, en el test de la relación de verosimilitud (Likelihood Ratio, LR). Este test será idealmente óptimo si óptimo no sólo si se conocen las funciones de verosimilitud que contempla cada hipótesis, sino si además se deben conocer las probabilidades a priori de cada hipótesis y los costes de las decisiones.

$$\frac{p(\boldsymbol{Y}|\boldsymbol{X}, H_0)}{p(\boldsymbol{Y}|\boldsymbol{X}, H_1)} \begin{cases} \geq \theta, & se\ acepta\ H_0 \\ < \theta, & se\ acepta\ H_1 \end{cases} \tag{2.16}$$

donde $p(\boldsymbol{Y}|\boldsymbol{X}, H_i), i = 0, 1$ es la función de densidad de probabilidad de los datos de test si se supone una hipótesis cierta y se conocen los datos de train $\boldsymbol{X}$. En función de la aplicación en la que se trabaje, se necesitará un umbral (θ) más bajo (sistema más permisivo) o más alto (sistema más restrictivo), que dependerá del ratio entre las probabilidades a priori de las hipótesis y del de los costes de las decisiones incorrectas. En la Figura 2-10 se muestra el esquema de funcionamiento de un sistema de verificación de clases acústicas basado en la razón de verosimilitud.

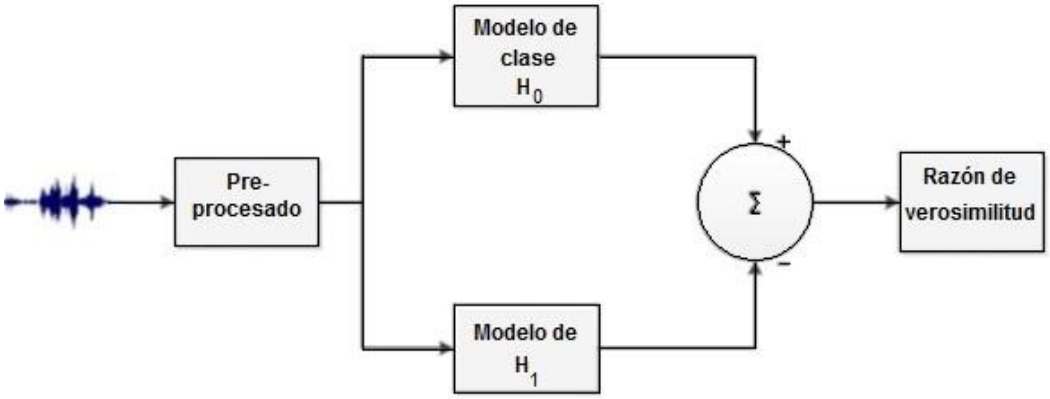


Figura 2-10: Sistema de verificación de clases acústicas basado en LR

donde la etapa de pre-procesamiento extrae la características $\boldsymbol{Y} = \{\overrightarrow{y_t}\}_{t=1,\ldots,N_Y}$, para posteriormente calcular las funciones de verosimilitud de las hipótesis H_0 y H_1, representándose cada hipótesis mediante los modelos estadísticos X y $\bar{X}$, respectivamente. La nueva razón de verosimilitud, aplicando posteriormente el logaritmo, es:

$$\Lambda = \log\big(p(\boldsymbol{Y}|X)\big) - \log\big(p(\boldsymbol{Y}|\bar{X})\big) \tag{2.17}$$

donde se utiliza la notación más común, condicionada a los modelos, que condicionada a los datos y a las hipótesis.

Dentro del contexto de los sistemas de reconocimiento basados en GMM, el modelo X compara los datos de evaluación $\boldsymbol{Y}$ con el modelo entrenado con X bajo H_0, es decir, suponiendo que ambos provienen de la misma clase, mientras que $\bar{X}$ caracteriza al archivo de audio de evaluación $\boldsymbol{Y}$ dentro del Universal Background Model (UBM). Es decir, se compara la probabilidad de que las características extraídas provengan del mismo tipo de audio ($X = target$) entre la probabilidad que provengan del modelo UBM ($\bar{X} = UBM$).

Si el conjunto de clases del problema de detección es muy alto, esta es una buena aproximación a la razón de verosimilitudes, ya que el modelo UBM simboliza la hipótesis de que el archivo de audio de test pertenece a "otra fuente". Cuando el número de clases es restringido, como en los problemas de clasificación de audio en este PFC, el modelo UBM no representa bien la hipótesis H_1. Sin embargo, el UBM constituye en estos

contextos un elemento de robustez a la falta de datos, y mediante el ratio de verosimilitudes se obtiene una normalización de las puntuaciones de los datos de evaluación con respecto a las distintas clases del problema, que resulta muy útil de cara a la clasificación final.

2.2.2.4 Evaluación del rendimiento

El diseño y la implementación de un sistema de reconocimiento de audio/voz conlleva también una etapa de evaluación. El objetivo de la evaluación es comprobar las capacidades y la bondad del sistema desarrollado. Para ello se evalúan las diferentes técnicas empleadas para dicho reconocimiento. Estas pruebas deben realizarse en condiciones lo más parecidas posibles a aquel entorno para el que se desarrolla el sistema, lo que permitirá evaluar el rendimiento de forma más fiable.

Los sistemas de detección funcionan normalmente en dos pasos. En primer lugar se calcula un valor de similitud (también llamado puntuación o score) entre las características capturadas por el sistema, en este caso extraídas del audio, y el patrón de referencia de la clase reclamada. Idealmente, cuanto mayor sea la puntuación o score, mayor será el apoyo a la hipótesis evaluada, como cuando un detector de música (música/no música) puntúa con un gran score un determinado fragmento de audio quiere decir que es muy probable que se corresponda con un fragmento de música. En segundo lugar, mediante el proceso de calibración se obtiene una relación de verosimilitud que puede ser comparada con un umbral θ, pudiéndose calcular como ratio de las probabilidades a priori y los cortes de decisión. De esta manera pueden darse dos tipos de errores en las decisiones tomadas por el sistema:

- Error de falso rechazo: se produce cuando el sistema no identifica un segmento de audio como perteneciente a la clase acústica analizada, cuando sí debería hacerlo.
- Error de falsa aceptación: se produce cuando el sistema acepta erróneamente un segmento de audio como perteneciente a la clase acústica estudiada.

En segmentación de audio, cuando por ejemplo se identifica un segmento de audio como habla cuando en realidad es música, supone un error de falso rechazo para la clase acústica de música y un error de falsa aceptación para la clase acústica de habla.

Como ya se adelantó antes, existe una relación directa entre las tasas de error y el valor del umbral escogido. Para un umbral muy bajo un mayor número de segmentos de audio pertenecientes a otras clases acústicas podrían ser clasificados como pertenecientes a la clase acústica evaluada, pero a la vez disminuiría el número de segmentos genuinos rechazados. Por otra parte, para un valor de umbral muy alto muchos segmentos válidos serían rechazados, pero el número de aceptación de segmentos de audio impostores

disminuiría. Por lo tanto hay que buscar un valor de umbral equilibrado acorde a las especificaciones del sistema a desarrollar y de la seguridad que se le quiera dar.

Este tipo de tasas de error, aunque no se usan expresamente durante este proyecto, sí constituyen la base fundamental de otras tasas de error empleadas para medir los rendimientos del sistema implementado y poder compararlo con otros con similares propósitos. La principal tasa de error utilizada a lo largo de todo el proyecto es la Tasa de Error de Segmentación (ver apartado 2.3.2.1.), derivada de la Tasa de Error de Diarización utilizada en las evaluaciones National Institute of Standards and Technology (NIST) [Butko, 2010].

En segmentación de audio se evalúa cada clase acústica por separado, midiendo dos tipos de error análogos a los estudiados en este apartado: el número de tramas de audio etiquetado erróneamente como perteneciente a una determinada clase (AC_i) además del audio que sí debería haberse etiquetado como AC_i pero se reconoce como otra clase acústica. A estos errores se les llama error de inserción y error de omisión, respectivamente.

2.2.2.5 Fusión de sistemas

La fusión de sistemas de reconocimiento consiste en la combinación de los resultados de dos o más sistemas de reconocimiento con el objetivo de conseguir un sistema más robusto y con mejores prestaciones que los sistemas individuales trabajando por separado [Brümmer et al., 2007]. De esta forma, se puede aprovechar distinta información del audio para que, combinándose, dé como resultado un sistema con mayor rendimiento [Reynolds et al., 2003]. La combinación de resultados puede realizarse desde dos perspectivas, fusiones basadas en puntuaciones o scores y fusiones basadas en decisiones categóricas:

Fusiones basadas en reglas

- **Fusión basada en reglas fijas [Kittler et al., 1998]**
  Combina directamente las puntuaciones obtenidas por los sistemas individuales mediante un operador simple, como la suma, el producto, el máximo o el mínimo. Como requisito es necesario que las puntuaciones se encuentren en un rango de valores homogéneo, dado que puede que el rango de los valores de las puntuaciones obtenidas por los dos sistemas a fusionar sea muy diferente.
- **Fusión basada en reglas entrenadas**
  Hace uso de los scores de los sistemas individuales como patrones de entrada a un nuevo sistema de reconocimiento, tratando la fusión como un problema de clasificación de patrones entrenado. Para ello, existen técnicas tales como las redes neuronales, las Support Vector Machines (SVMs) [Fierrez-Aguilar et al., 2003] o la regresión logística [Brümmer et al., 2007].

Fusiones basadas en decisiones categóricas

Este tipo de fusiones utilizan exclusivamente las etiquetas calculadas a partir de las puntuaciones de los sistemas a fusionar, perdiendo la noción del peso que aporta cada score, ya que las decisiones sobre scores son una transformación unidireccional sin posibilidad de retorno. Por ejemplo, dos scores, por muy diferentes que sean, pueden resultar en la misma decisión, como ocurre en los detectores de presencia de música. Algunas de las técnicas más utilizadas son el máximo voto (mayority voting), el criterio de unanimidad o la suficiencia de un solo voto. En tareas en las que se disponga únicamente de dos clases (como un detector de voz/no voz o música/no música) el criterio de unanimidad se consigue operando directamente un AND, mientras que la técnica de un voto es suficiente se convierte en la operación lógica OR. El método del máximo voto para fusiones de únicamente dos sistemas (como el presentado en este proyecto) sería lo mismo que aplicar el criterio de unanimidad o la operación lógica AND.

La fusión propuesta en este proyecto del sistema implementado con el presentado por el grupo ATVS en la evaluación ALBAYZIN 2010 sigue un patrón de mezcla de decisiones o etiquetas (fusión basada en decisiones categóricas) debido a la ausencia de las puntuaciones generadas por el sistema ATVS para ALBAYZIN 2010. De esta forma se combinan las decisiones de los sistemas por separado, es decir, las segmentaciones de audio, mediante operaciones lógicas sencillas como AND u OR, buscando un mejor rendimiento final. Una de las mejoras propuestas durante este PFC es la de simular de nuevo el sistema ATVS para ALBAYZIN 2010 y fusionar los scores de ambos sistemas.

2.2.3 Características musicales

Antes de definir algunas de las características musicales más utilizadas en la literatura es necesario destacar el timbre, muy utilizado en sistemas clásicos de tratamiento de audio. El timbre es la característica que diferencia a dos sonidos que tienen la misma intensidad percibida y el mismo tono o conjunto de tonos. Los MFCC y características basadas en envolvente espectral se suelen llamar tímbricas en aplicaciones MIR [Aucouturier & Pachet, 2004]. Esto es porque el timbre está íntimamente relacionado con la envolvente espectral. Cuando los armónicos tienen relación de amplitud diferente entre ellos, cambia el timbre. No obstante, según el estudio llevado a cabo por Aucouturier y Pachet en 2004 "*Improving Timbre Similarity: How high is the sky?*", de gran impacto en la comunidad científica, se ha llegado a la conclusión que con características tímbricas se llega a un techo de rendimiento, llamado "techo de cristal".

Como alternativa y complemento a las características tímbricas se hace uso de todo el resto de información en la música, que es mucho y muy variado, como la utilización de la información de tono, las llamadas características cromáticas, siendo las que se utilizan en este PFC, entre otras. Apartándose de las técnicas clásicas de tratamiento de audio, se empezaron a utilizar nuevas características del audio, como el análisis de picos en el espectrograma, la variabilidad de armónicos o el ritmo. En este apartado se estudiarán

algunas características y técnicas alternativas en el tratamiento de señales de audio, que dan un valor añadido a los sistemas ampliamente estudiados durante los últimos veinte años, ya que aunque el uso de características musicales es bueno para la detección de música, no lo es para la diferenciación entre habla y música. Así pues, de querer tener un sistema mucho más completo y robusto en todos los sentidos habría que combinar diferentes técnicas.

2.2.3.1 Cromagramas

Una de las técnicas que se lleva utilizando asiduamente durante los últimos años es la representación del audio mediante cromagramas (chroma features) [Ellis & Poliner, 2007]. Gracias a estas representaciones se consigue identificar el contenido melódico y armónico de una pieza de audio. Este tipo de sistemas se suelen emplear en la identificación de versiones musicales (se estudiará más detenidamente en las secciones 2.4.1.2. y 5.2.), ya que no pueden hacer un uso adecuado de las características tímbricas (MFCCs). A la hora de comparar dos canciones tendría más peso la componente cantada del artista que la propia pieza en sí, dando lugar a un identificador de cantantes y no de versiones.

Los cromagramas basan su tecnología en la representación de la energía en cada nota sincronizada con el ritmo. El motivo principal detrás de estar restringidos por el tempo es que dos canciones que sean versiones de una sola pueden estar interpretadas a velocidades distintas, por lo que hay que hacer un seguimiento exhaustivo de esta característica.

Un cromagrama es la representación en el tiempo de los coeficientes de energía de cada una de las bandas de un banco de filtros adaptado a las octavas de cada una de las doce notas (croma). De modo que para la nota Do se empleará un banco de filtros donde exista un filtro en la nota Do de cada octava del espectro de audio que se maneje centrado en la frecuencia de la nota para cada octava (rango de frecuencias entre dos sonidos cuyas frecuencias fundamentales tienen una relación de dos a uno).

El proceso para calcular el cromagrama a partir de un fichero de audio es el de calcular la energía por cada nota de la escala musical. Los cromagramas utilizados en este proyecto [Ellis & Poliner, 2007] resultan de dividir el espectrograma de dicho fichero con un detector de ritmo o pulsos, es decir, donde la canción muestra cambios notables, por ejemplo, en el tempo, en la potencia… Los divisores obtenidos se conocen como onsets. También se puede optar por dividir el audio en fragmentos de duración fija, pero se ha demostrado experimentalmente que la división en onsets da mejores resultados (emulando a cómo lo haría el oído humano).

Una vez dividido el audio en pequeños fragmentos, se extrae la información de los doce semitonos (chromas) distintos (Do, Do #, Re, Re #, Mi, Fa, Fa #, Sol, Sol #, La, La #y Si, siendo # una nota sostenida) de los que se compone la escala musical (Anexo A).

El objetivo de dividir el audio en los doce semitonos es el de poder comparar audio en diferentes octavas.

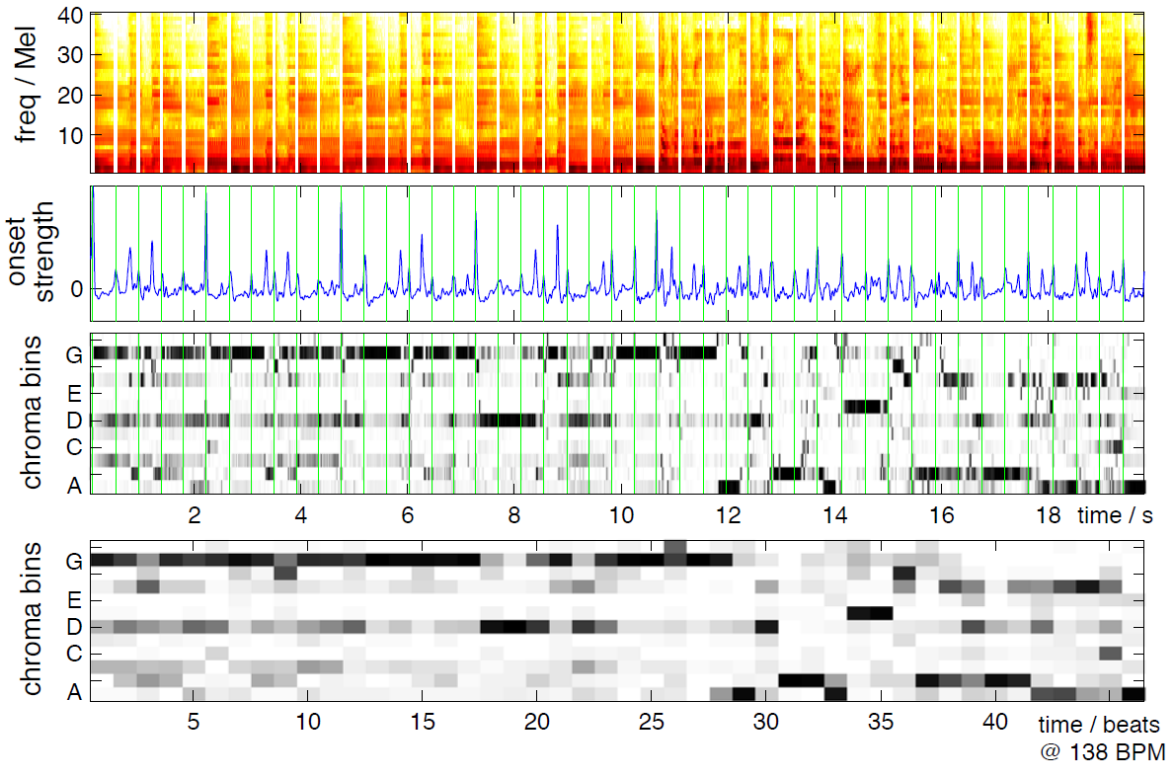


Figura 2-11: Espectrograma en la escala Mel (primera gráfica). Detección de onset a partir de la potencia del audio (segunda gráfica). Cromagrama dividido por tramas (tercera gráfica). Cromagrama dividido por pulsos (cuarta gráfica). [Ellis & Poliner, 2007]

2.2.3.2 Entropía cromática

La entropía mide la incertidumbre o la imprevisibilidad de una función de probabilidad de masa (Probability Mass Function, PMF). Si consideramos el módulo de la FFT normalizado (es decir, dividido por la suma de todas las componentes en módulo) como una PMF, podemos también calcular su entropía, llamada entropía cromática [Pikrakis et al., 2006]. Dicha entropía espectral puede utilizarse como discriminante entre voz y música, puesto que, en general, la entropía espectral es mayor para los fragmentos de habla que para los musicales. Asimismo, entropías basadas en MFCC son típicas en discriminadores de voz y no voz [Kinnunen & Li, 2010].

La entropía cromática [Pikrakis et al., 2006] es una variante de la entropía espectral. En lugar de calcular la entropía directamente a partir del módulo de la FFT normalizado, se hace primero una asignación del espectro de potencia en la escala de frecuencias de Mel y se divide en doce sub-bandas, donde la frecuencia central f_k de cada banda coincide con cada uno de los doce semitonos de la escala musical mayor (Anexo A). Para

una frecuencia central fija f_0 (apartado 3.4.1.), siendo esta la banda más baja a evaluar, las frecuencias centrales de las demás L sub-bandas, de manera análoga a la ecuación (2.4), son:

$$f_k = 1127.01048 \cdot \log_e\left(1 + \frac{f_0 \cdot 2^{\frac{k}{12}}}{700}\right) \quad k = 0, \dots, L-1 \tag{2.18}$$

donde L hace referencia al número de sub-bandas desde f_0 hasta la mitad de la frecuencia de muestreo $f_{muestreo} = 16000\ Hz$, calculado según (3.2).

Al igual que para la entropía espectral, la energía X_i de la sub-banda i-ésima se normaliza dividiendo entre la energía total de todas las sub-bandas (ecuación (2.19)), mientras que la entropía cromática de la energía espectral normalizada se calcula mediante la ecuación (2.20):

$$n_i = \frac{X_i}{\sum_{i=0}^{N} X_i} \quad i = 0, \dots, L-1 \tag{2.19}$$

$$H = -\sum_{i=0}^{L-1} n_i \cdot \log_2(n_i) \tag{2.20}$$

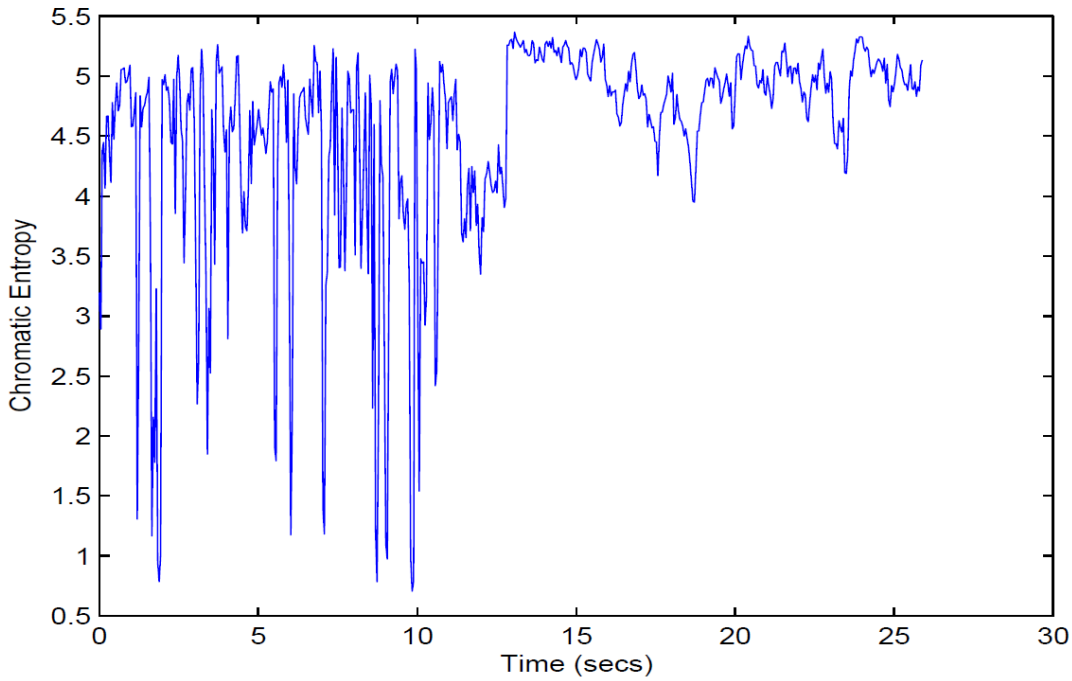


Figura 2-12: Entropía cromática para un fragmento de radio de la BBC. [Pikrakis et al., 2006]

En la Figura 2-12 se aprecia la entropía cromática extraída a partir de un fragmento radiofónico de la BBC de 26 segundos, en el que la primera parte corresponde exclusivamente a voz mientras que la segunda parte se trata únicamente de música.

Una de las principales utilidades de la entropía cromática vuelve a ser la capacidad de discriminar entre voz y música. El método más sencillo para alcanzar este objetivo es calcular las medias sobre la entropía cromática y, mediante la definición de un umbral decidir qué tipo de audio está sonando en cada momento. El cálculo de medias a partir de la entropía cromática se realiza mediante una ventana temporal deslizante de tamaño ajustable. La elección del rango en el que se calculan dichas medias es vital para evitar posibles fluctuaciones o errores de oscilación (si se elige un tamaño de ventana excesivamente pequeño), pero a la par hay que tener especial cuidado para no descartar pequeños fragmentos de voz/música escogiendo una ventana demasiado grande, operando como un filtro paso bajo muy restrictivo.

Una posible mejora de este sistema sería la de añadir más medidas estadísticas sobre la propia entropía cromática, como son la varianza o la skewness, entre otras. De esta forma el sistema ganaría en robustez y se podrían ampliar los campos de aplicación, como por ejemplo la segmentación de audio (sistemas capaces de etiquetar el audio entrante con distintas clases acústicas: voz, música, ruido…).

2.2.4 Análisis de señales mediante métodos estadísticos

En esta sección se presentan algunas de las características que poseen las series temporales como las vistas en el apartado 2.2.3.2. (la entropía cromática). Las principales características, o momentos, que definen las series temporales son: la media, la varianza, el skewness y la kurtosis.

2.2.4.1 Media

La primera característica que se muestra es la media aritmética, o también conocida como momento de primer orden. La media aritmética o *media* de un conjunto de N números $X_1, X_2, X_3, \dots, X_N$ se representa por μ y se define como:

$$\mu = \frac{X_1 + X_2 + X_3 + \cdots + X_N}{N} = \frac{1}{N}\sum_{j=1}^{N} X_j \tag{2.21}$$

2.2.4.2 Varianza

La varianza, o momento de segundo orden, de un conjunto de datos se define como la esperanza del cuadrado de la desviación del conjunto de datos (o variable aleatoria) respecto a su media:

$$Var(X) = \sigma^2 = E[(X - \mu)^2] = E[X^2] - \mu^2 \tag{2.22}$$

Cuando es necesario distinguir la desviación típica de una población de la desviación típica de una muestra sacada de esta población, se emplea el símbolo s para la última y σ para la primera. Así, s^2 y σ^2 representarían la varianza muestral y la varianza poblacional, respectivamente.

2.2.4.3 Skewness

La siguiente característica a definir es el skewness, o momento de tercer orden. El skewness es el grado de asimetría, o falta de simetría, de una distribución. Si la función de densidad de probabilidad de una distribución tiene una "cola" más larga a la derecha del máximo central que a la izquierda, se dice de la distribución que está sesgada a la derecha o que tiene skewness positivo. Si es al contrario, se dice que está sesgada a la izquierda o que tiene skewness negativo.

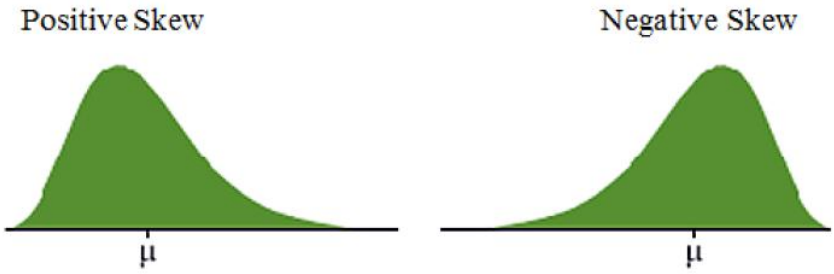


Figura 2-13: Representación de una supuesta distribución con skewness positivo y otra con skewness negativo

En una distribución simétrica, como la normal, los valores por encima y por debajo de la media se cancelan, y cuando se calcula el skewness se obtiene un valor igual a cero.

Siguiendo la definición genérica de momento de orden r, el skewness se puede expresar en función de sus momentos anteriores (media y varianza):

$$Skew(X) = E\left[\left(\frac{X - \mu_x}{\sigma_x}\right)^3\right] = \frac{E[X^3] - 3\mu E[X^2] + 3\mu^2 E[X] - \mu^3}{\sigma^3} = \\ = \frac{E[X^3] - 3\mu\sigma^2 + 2\mu^3}{\sigma^3} \tag{2.23}$$

2.2.4.4 Kurtosis

La kurtosis, o momento de cuarto orden, es el grado de apuntamiento de una distribución, normalmente se toma en relación a la distribución normal. Una distribución que presenta un apuntamiento relativo alto, tal como la de la curva roja de la Figura 2-14, se llama leptocúrtica, mientras que la curva azul de la Fig. 2-14, que es más achatada, se llama platicúrtica. La distribución normal, Fig. 2-14, que ni es muy apuntada ni achatada, se llama mesocúrtica.

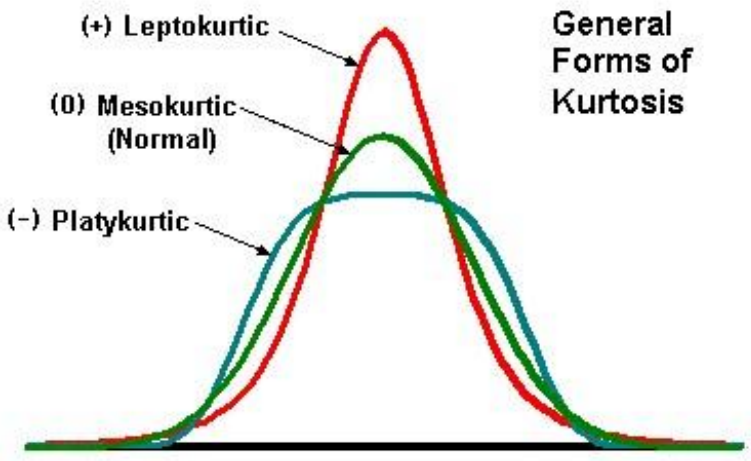


Figura 2-14: Distintos tipos de kurtosis en distribuciones gaussianas

De manera análoga a (2.23), la kurtosis viene definida por la relación:

$$Kurt(X) = E\left[\left(\frac{X-\mu_x}{\sigma_x}\right)^4\right] \tag{2.24}$$

La distribución normal posee una kurtosis igual a 3, por lo que siguiendo la definición de los distintos tipos de kurtosis existentes (leptocúrtica, platicúrtica y mesocúrtica), se distinguen tres rangos de valores posibles de kurtosis:

- Valores mayores que 3: si la distribución tiene valores alejados de la media, ya sean por la izquierda o por la derecha, producirán grandes valores de kurtosis. Las distribuciones con estos valores de kurtosis presentan unas colas más levadas o altas que las de la normal. Nos referimos a distribuciones leptocúrticas o de kurtosis positiva.
- Valores iguales a 3: este valor es característico de la distribución normal, y esta distribución implica que los eventos extremos no ocurren muy a menudo, son poco probables. Es el caso de las distribuciones mesocúrticas o de kurtosis nula.

- Valores menores que 3: si la distribución no tiene valores alejados de la media, el valor de kurtosis para estas distribuciones no será muy elevado. Las distribuciones con estos valores de kurtosis presentan unas colas más delgadas o bajas que las de la normal. Estas distribuciones son las conocidas como platicúrticas o de kurtosis negativa.

2.2.5 Segmentadores de señales basados en criterios estadísticos

Una vez se han extraído las características del audio, entrenado modelos y enfrentado distintos "targets" contra dichos modelos se obtienen las puntuaciones finales o scores (ver apartado 2.2.2.3.). De igual manera que la media nos servía para evitar posibles fluctuaciones u oscilaciones a la hora de discriminar entre voz y música usando la entropía cromática (ver apartado 2.2.3.2.), se suelen filtrar los scores resultantes de los sistemas de procesado de audio haciendo uso de criterios estadísticos, desde los más básicos como la media o la mediana, hasta algunos un poco más avanzados como el Bayesian Information Criterion (BIC).

En esta sección se estudiarán estas técnicas aplicadas a sistemas de segmentación de audio, las cuales serán de mucha utilidad en la parte final del proyecto.

2.2.5.1 Filtrado por media, mediana y moda

Supongamos que disponemos de un sistema detector de música, activando una etiqueta o flag cada vez que se identifique el audio de entrada como música. Para conseguir un decisor entre dos clases acústicas (música y no-música) se evaluará y puntuará el audio para las dos clases, y aquella que de mayor resultado, es decir, la que más probabilidades tenga de definir el audio de entrada, será la elegida como la etiqueta de salida. Se ha supuesto un sistema análogo al presentado en este proyecto, que calcula las puntuaciones o scores trama a trama, lo que podría implicar la generación de muchas decisiones espurias. La Figura 2-15 explica de manera visual este concepto:

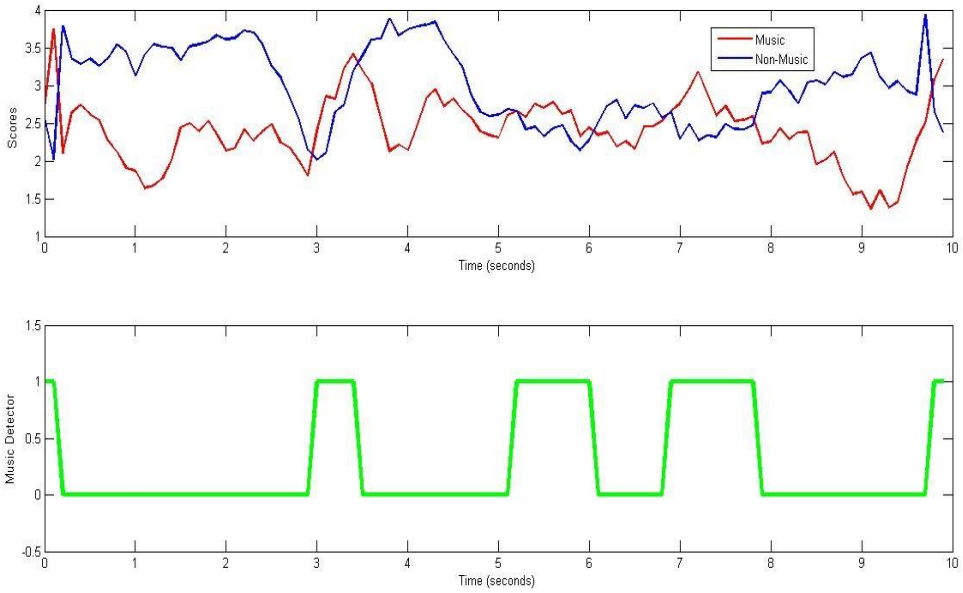


Figura 2-15: Detector de música basado en comparación directa de scores (música frente a no-música)

A partir de la Figura 2-15 se proponen dos tipos de filtrado de resultados: filtrados sobre los scores o filtrados sobre las etiquetas. Ambas técnicas proporcionan un suavizado a la salida que protege al sistema de cambios espurios en el decisor.

Dos de los filtrados típicos de scores son la media y la mediana, ya que pueden operar perfectamente con conjuntos de números no enteros:

- El filtrado por medias es equivalente a lo explicado en el apartado 2.2.4.1. mediante la ecuación (2.21).
- La mediana es el percentil 50 de un conjunto de datos, o la interpolación si no hubiese un percentil 50 exacto.

Geométricamente, la mediana es el valor de X (abscisa) que corresponde a la vertical que divide un histograma en dos partes de igual área.

Si se opta por filtrar las etiquetas, o en el caso estudiado, las clases acústicas, conviene recurrir a los filtrados por moda ya que tratamos únicamente con variables discretas (un determinado fragmento de audio puede pertenecer a la clase A, B o C, si tratamos con un sistema de tres clases). La moda de una serie de números es aquel valor que se presenta con la mayor frecuencia, es decir, es el valor de etiqueta acústica más común.

En el caso de histogramas, la moda será el valor (o valores) de X correspondientes al máximo (o máximos) del histograma.

En la Figura 2-16 se muestran las posiciones relativas de la media, mediana y moda para una función de densidad de probabilidad asimétrica (skewness positivo). Para curvas simétricas, la media, moda y mediana coinciden.

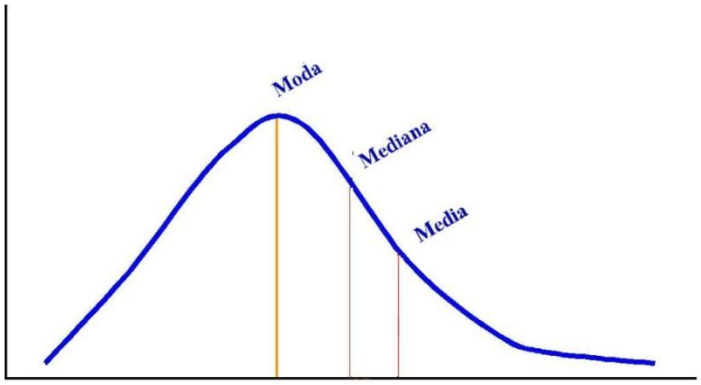


Figura 2-16: Distribución con skewness positivo

2.3 Segmentación de audio

La segmentación de audio consiste en dividir una grabación en regiones homogéneas de acuerdo a su contenido, asignando a cada segmento la etiqueta de la clase a la que pertenece [Butko, 2010] [Cheng et al., 2008] [Franco-Pedroso et al., 2010]. En función de la aplicación para la que se realice, el objetivo de la segmentación de audio puede ser muy diferente: separar la voz de la música y el ruido, separar las voces masculinas de las femeninas, separar los segmentos que corresponden a distintos locutores, etc. Tiene muchas aplicaciones y comúnmente se utiliza como primer paso de pre-procesado para mejorar los resultados de otros sistemas como los de reconocimiento automático de habla, identificación de locutores, recuperación de información e indexado de audio basada en su contenido, etc.

2.3.1 Técnicas en segmentación de audio

Los sistemas segmentadores de audio aglutinan gran cantidad de técnicas y tecnologías en sus algoritmos, pero hay algunas que tienden a usarse con mayor recurrencia, como son las segmentaciones mediante BIC o la extracción de características PLP, MFCC, SDC (Shifted Delta Coefficients) junto a modelados basados en GMM-UBM o clasificadores HMM.

El Bayesian Information Criterion, o BIC [Chen & Gopalakrishnan, 1998], es un algoritmo muy útil para tareas de segmentación, pues evalúa la propia señal a tratar (de audio en este caso) y define los instantes temporales que con mayor probabilidad separan fragmentos pertenecientes a distintas clases acústicas. Esos fragmentos podrían entrenarse

a continuación mediante modelados típicos (como GMM-UBM) y en la etapa de test ser evaluados como un solo bloque.

Otra de las técnicas más usadas es la extracción de características acústicas, como los MFCC con CMN-Rasta-Warping, y hacer una segmentación trama a trama (frame-by-frame) haciendo uso de Modelos de Mezcla de Gaussianas, es decir, evaluando cada trama resultante del proceso de enventanado y decidiendo a que clase acústica corresponde. Sin embargo, los mejores sistemas de segmentación de audio se aprovechan de las ventajas intrínsecas de tal tarea, como es el número limitado de clases acústicas, ya que en las especificaciones de todas las competiciones y evaluaciones se piden sistemas que puedan segmentar el audio de entrada en un determinado número de tipos de audio (por ejemplo, en ALBAYZIN 2010 son las siguientes clases: voz, música, voz sobre música, voz sobre ruido y otros).

De esta manera se desarrollan segmentadores basados en Modelos Ocultos de Markov (HMM), definiendo cada estado como una de las distintas clases acústicas a identificar [Franco-Pedroso et al., 2010].

Un dato curioso es que teniendo en cuenta la gran carga musical que tiene este tipo de sistemas sorprende el poco uso de características musicales que se hace, con la salvedad de algunos equipos que incorporan a sus algoritmos características cromáticas [Butko et al., 2010].

2.3.2 Evaluaciones tecnológicas

En el ámbito nacional, la Red Temática en Tecnologías del Habla[1] organiza las campañas competitivas de evaluación ALBAYZIN que se celebran cada dos años y evalúan distintos aspectos relacionados con las tecnologías del habla. La segmentación de audio se ha incluido en las dos últimas campañas realizadas, ALBAYZIN 2010 y 2012.

2.3.2.1 Evaluación ALBAYZIN 2010 en segmentación de audio

Como ya se ha ido adelantando a lo largo de toda la memoria, este proyecto se basa y busca poder compararse con el sistema presentado por el grupo ATVS de la Escuela Politécnica Superior de la UAM en la evaluación ALBAYZIN 2010 de segmentación de audio [Franco-Pedroso et al., 2010], debido en mayor medida al fácil acceso a todo el material presentado por el grupo ATVS a dicha evaluación.

[1] http://www.rthabla.es/

La evaluación ALBAYZIN 2010 fue la primera organizada por la Red Temática en Tecnologías del Habla en recoger la tarea de segmentación de audio como tal, aunque en evaluaciones pasadas ya se estudiaban otras tareas como la segmentación de locutores e identificación de idioma (ALBAYZIN 2006). En cada tarea de esta evaluación de distribuía entre los participantes una gran cantidad de bases de datos, en función de la tarea a desarrollar. En cuanto a la segmentación de audio se disponía de grabaciones del canal catalán de televisión 3/24. Puesto que esta base de datos es la utilizada durante todo el proyecto se estudiará más detenidamente en la sección 3.3.

La tarea de segmentación de audio en la evaluación ALBAYZIN 2010 tenía por objetivo segmentar el audio de entrada en cinco clases acústicas, a saber: voz (speech, *sp*), voz sobre ruido (speech over noise, *sn*), voz sobre música (speech over music, *sm*), música (music, *mu*) y otros (other, *ot*). Todos los algoritmos presentados se comparaban siguiendo una métrica similar a la utilizada en evaluaciones NIST en diarización de locutor (National Institute of Standards and Technology), la Tasa de Error de Segmentación (Segmentation Error Rate, SER), que se corresponde con la fracción de tiempo que no ha sido correctamente atribuida a la clase correspondiente. En las zonas de solapamiento entre clases la duración del segmento se atribuye a todas las clases presentes en el mismo, por lo que un mismo segmento temporal puede ser considerado más de una vez en los cálculos.

El SER se calcula como la suma de dos tipos de errores: el porcentaje de tiempo en el que una clase está presente pero no ha sido etiquetada (Error de Omisión o Missed Class Time) y el porcentaje de tiempo en que se ha etiquetado una clase cuando realmente no estaba presente (Error de Inserción o False Alarm Time). Para poder comparar distintos errores entre sí, se normaliza el SER por la duración total de cada una de las clases acústicas definidas en la base de datos. De tal forma, la ecuación que calcula el error de los sistemas es:

$$Error = average_i\left(\frac{dur(miss_i) + dur(fa_i)}{dur(ref_i)}\right) \tag{2.25}$$

donde:
$dur(miss_i) =$ duración total de todos los errores de omisión (misses) de la i-ésima clase acústica
$dur(fa_i) =$ duración total de todos los errores de inserción (false alarms) de la i-ésima clase acústica
$dur(ref_i) =$ duración total de la i-ésima clase acústica según el Ground-Truth

2.4 Sistemas de Recuperación de Información Musical

La Recuperación de Información Musical (Music Information Retrieval, MIR) [Downie, 2008] es un área de creciente interés en la música por computador. La disponibilidad en línea de millones de objetos musicales ha impulsado la investigación en algoritmos capaces de extraer y procesar esta información para indexación, clasificación, recuperación, etc.

En este apartado se estudiarán algunas de las principales tareas en MIR, así como las principales evaluaciones y congresos de este ámbito.

2.4.1 Tareas en Recuperación de Información Musical

Desde hace unos años la Recuperación de Información Musical ha cobrado cada vez más fuerza, alentada por grupos de investigación de todo el mundo y por grandes multinacionales que ven en este campo un negocio por explotar. Aplicaciones como reconocedores de canciones (Shazam®) o sistemas de recomendación musical son sólo algunas de las muchas presentes en nuestros ordenadores y dispositivos móviles [Downie et al., 2010].

2.4.1.1 Similitud de audio musical

Los sistemas de similitud de audio musical tienen por objetivo proporcionar al usuario canciones "musicalmente" similares a la canción que se está escuchando. La similitud de audio musical es una medida de similitud o parentesco entre dos archivos musicales [Pohle et al., 2009].

Esta clase de sistemas tienen un gran valor, pero a la vez su complejidad es máxima, pues esa medida de similitud entre dos archivos musicales tiene un carácter completamente subjetivo, ya que no hay un protocolo definido que dicte cuando dos canciones se parecen o no, sino que es una persona la que debe decidir tal supuesto, pudiendo opinar de manera opuesta a otra persona. Como reto añadido a estos sistemas, las medidas de similitud no son binarias, pues entre dos archivos musicales puede haber un parentesco del 70 %, por ejemplo. Así pues no basta con decidir si dos audios son similares o no, sino que además hay que concretar el nivel de similitud entre ambos.

Por lo tanto, es vital disponer de una buena base de datos, elaborada por expertos (con formación de musicología, psicología, estudios académicos de música, procesamiento de señales, aprendizaje automático, etc.) que sean capaces de determinar con toda la exactitud posible las similitudes entre distintas canciones. Desgraciadamente estas bases de datos suelen estar en manos de grandes empresas privadas con un gran capital invertido (Sony, Shazam®, Spotify…), que no suelen distribuirse al público en general.

Otro inconveniente de estos sistemas es el tratar continuamente con archivos musicales, pues la mayoría están sujetos a derechos de autor que son prohibitivos para la casi totalidad de los grupos de investigación.

Este tipo de sistemas, al ser de carácter tan general, suelen utilizar las técnicas más usadas en reconocimiento de audio, como son los MFCC junto con modelas GMM-UBM.

2.4.1.2 Identificación de versiones musicales

Otro de los sistemas con más proyección durante los últimos años son los identificadores de versiones musicales (covers). La principal diferencia con la similitud de audio musical es que aquí se busca exclusivamente la probabilidad de que dos archivos musicales sean versiones de una misma pieza musical [Serrà et al., 2009].

Las dos principales ventajas que presentan estos sistemas frente a los estudiados en el apartado anterior es la facilidad para crear bases de datos pues, quitando algunos casos especiales[2], es trivial definir si una canción es una versión musical de otra composición interpretada por otro artista. La otra ventaja se basa en la etapa de evaluación, puesto que dos piezas musicales pueden o no ser versiones de una misma canción, teniendo sistemas binarios, facilitando la decisión.

Sin embargo sigue tratándose de una tarea muy compleja, tanto desde un punto de vista de implementación como de disposición de recursos, ya que comparte la problemática de los derechos de autor de los sistemas de similitud musical.

Las características tímbricas (MFCC) no funcionan especialmente bien en esta tarea, debido a que si se hiciera un comparador utilizando estas características se estaría relacionando componentes tímbricas. Existen versiones de una misma canción que presentan orquestaciones radicalmente diferentes, y por lo tanto timbres muy diferentes. Por el contrario, dos canciones que no son versiones pueden presentar timbres muy similares. De este modo los sistemas actuales tienden a centrarse en el contenido propio de la canción y utilizar características musicales, basadas en tono, armonía, estructura, ritmo, etc. Una de las técnicas que más se ha utilizado en los últimos trabajos ha sido los "chroma features" (ver apartado 2.2.3.1.) con sincronismo de ritmo, es decir, cromagramas calculados a partir del audio detectando el ritmo de la música.

Una de las metodologías más empleadas [Ellis & Poliner, 2007] para identificar dos versiones musicales parte de los dos cromagramas sincronizados (ver apartado 2.2.3.1.) de cada una de las piezas musicales a comparar. A partir de esos cromagramas se procede

[2] La canción "My Sweet Lord" de George Harrison y la canción "He's so Fine" de Ronnie Mack, que fueron canciones diferentes en su origen pero debido a un caso de plagio el juez dictaminó que efectivamente Harrison plagió la canción, aunque de forma subconsciente.

a calcular la correlación cruzada rotando cíclicamente uno de los cromagramas, tanto de forma tonal como temporal. De esta manera se evitan posibles diferencias debido a distintas tonalidades a la vez que se evitan errores producidos por diferencias de tempo o velocidad entre determinados instantes de dos versiones de una misma canción. La puntuación, probabilidad o score de que dos canciones sean versiones de una misma pieza musical se obtiene a partir del máximo de todas las correlaciones cruzadas, determinando así tanto el desplazamiento tonal como el temporal.

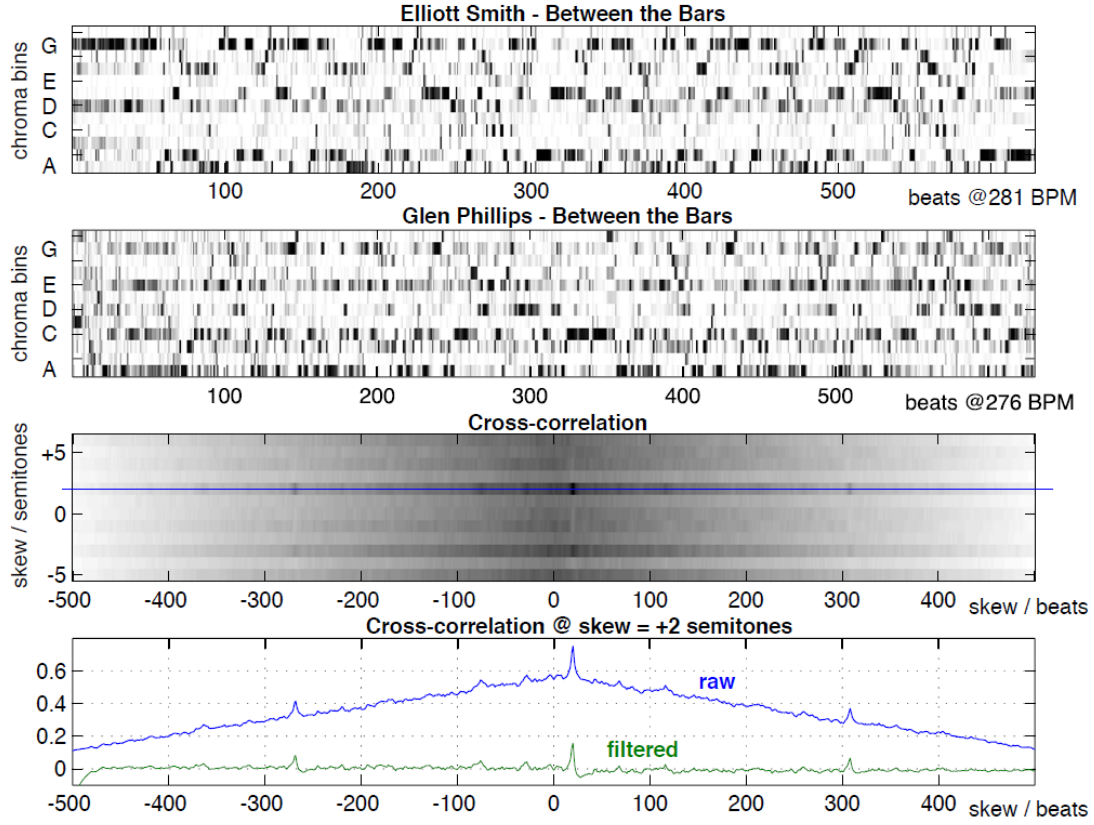


Figura 2-17: Sistema de identificación de versiones musicales [Ellis & Poliner, 2007]

En la Figura 2-17 vemos el esquema básico de un sistema de identificación de versiones musicales [Ellis & Poliner, 2007]. Las dos primeras imágenes muestran los cromagramas de dos versiones de la canción "Between the Bars". La tercera representa la correlación cruzada bidimensional para todas las posibles rotaciones del cromagrama con un desplazamiento temporal de ± 500 pulsos (en azul viene marcada la correlación que tiene el máximo). La última imagen muestra la correlación cruzada que mayor valor alcanza (señal azul), además de la misma correlación filtrada paso alto (señal verde). El score resultante es el máximo de la señal filtrada.

2.4.2 Evaluaciones tecnológicas

Desde principios de los años 2000 se han sucedido multitud de congresos y evaluaciones donde los mejores grupos de investigación en MIR de todo el mundo se reúnen y comparten sus avances en pos de dar un paso más en el avance de la tecnología.

De entre todas las evaluaciones, quizás la más importante sea la organizada por el International Music Information Retrieval Systems Evaluation Laboratory (IMIRSEL), fundada a mediados de los 90. El objetivo del IMIRSEL es el establecimiento de los medios necesarios tanto para el desarrollo como para la evaluación de las nuevas técnicas y tecnologías empleadas en MIR.

Uno de los proyectos con más transcendencia del IMIRSEL es la MIREX (Music Information Retrieval Evaluation eXchange[3]). La MIREX es una evaluación anual basada en la comunidad MIR, donde en función de la tarea elegida, los laboratorios de investigación de todo el mundo envían sus técnicas y algoritmos diseñados a la IMIRSEL. Estos algoritmos corren con las bases de datos estandarizadas y son evaluados utilizando métricas de evaluación definidos por la comunidad. Los resultados de la evaluación se publican al acabar las pruebas y se presentan en la International Conference on Music Information Retrieval (ISMIR[4]). Una de las tareas de MIREX es la creación de colecciones de datos a gran escala, seguras, pero accesibles, de materiales de musicales con una gran variedad de audio y meta-datos.

La historia de la MIREX se remonta a una evaluación llevada a cabo por el IMIRSEL en Barcelona en el año 2004 y organizada por la Universitat Pompeu Fabra (UPF) de Barcelona. Al año siguiente se estableció la primera MIR Evaluation eXchange (MIREX 2005), repitiéndose hasta nuestro días. Entre las principales tareas en la que se puede participar destacan algunas como la clasificación de audio en función del género musical, la detección de versiones musicales, la similitud musical, la detección de onsets (apartado 2.2.3.1.) y la detección de ritmo.

[3] http://www.music-ir.org/mirex/wiki/MIREX_HOME

[4] http://www.ismir.net/

3 Sistema propuesto

3.1 Introducción

En este capítulo se presenta una descripción del sistema segmentador de audio implementado, detallando las técnicas utilizadas desde la extracción de características hasta la obtención de los distintos modelos, así como la métrica utilizada para poder comparar el sistema con otros similares. También se hace una breve descripción de la base de datos utilizada. En la última sección se detalla la fusión llevada a cabo con el sistema de segmentación de audio presentado por el grupo ATVS en ALBAYZIN 2010.

3.2 Sistema propuesto: objetivos y definición

Como ya se adelantó en los primeros capítulos, este proyecto tiene como objetivo el estudio de características musicales para su uso en segmentación de audio y poder detectar distintos tipos de audio. Puesto que el proyecto se basa en las componentes musicales del audio, se ha buscado orientar los objetivos hacia la identificación de música, aunque se pretende evaluar el uso de características musicales para detectar otros tipos de audio.

De cara a evaluar diferentes escenarios, se han propuesto siete tareas derivadas de la evaluación ALBAYZIN, con siete sistemas correspondientes que emplean modelos diferentes, en los que cada uno es capaz de detectar un tipo de audio diferente. La nomenclatura elegida para referirse a cada sistema de manera rápida, unívoca y concisa deriva de las clases acústicas definidas en la base de datos utilizada (ver apartado 3.3.): voz = *sp*, voz sobre ruido = *sn*, voz sobre música = *sm*, música = *mu* y otros = *ot*.

Se definen tres sistemas básicos a desarrollar, que son:

- Detector exclusivamente de música (**MU-ALL**): este detector de música tiene por objetivo identificar los fragmentos de audio que se correspondan únicamente con música, es decir, siempre y cuando no haya otro tipo de audio sonando al mismo tiempo (como locutores o ruidos). Las clases acústicas se dividen en dos grupos: *mu* frente a *sp*, *sn*, *sm* y *ot*.
- Detector de música (**MUSM-ALL**): la principal diferencia de este sistema con el anterior es que busca la identificación de música en cualquier ambiente, haya o no sonidos de fondo o que sea la propia música la que suene en segundo plano. Las clases acústicas se dividen en dos grupos: *mu* y *sm* frente a *sp*, *sn* y *ot*.

- Detector de voz (**SP-NSP**): este sistema persigue la segmentación del audio de entrada en dos clases acústicas, voz y no-voz. De esta forma se podría desechar fácilmente todo el audio donde no hable ninguna persona, muy útil para sistemas de reconocimiento de locutor, habla o idioma. Las clases acústicas se dividen en dos grupos: *sp*, *sn* y *sm* frente a *mu* y *ot*.

Debido a que la base de datos utilizada (ver apartado 3.3.) distingue una clase acústica que engloba ruidos, silencios, y demás sonidos ambiguos (*ot*), y que no se evalúa como tal en las evaluaciones ALBAYZIN (la métrica no calcula el error de la clase acústica other, pero sí contabiliza los errores para otras clases si se identifica el audio como other erróneamente), se ha realizado un estudio especial para comprobar la influencia de esta clase acústica. De esta manera se han implementado otros cuatro sistemas más:

- Diferenciador entre música exclusivamente, voz y "otros" (**MU-SP-OT**): sistema similar a MU-ALL, pero haciendo distinción entre la voz y la clase acústica "otros", modelando cada clase por separado. Las clases acústicas se dividen en tres grupos: *mu* frente a *sp*, *sn* y *sm* frente a *ot*.
- Diferenciador entre música exclusivamente y voz, sin tener en cuenta la clase acústica "otros" (**MU-SP**): sistema idéntico al anterior en el que se obvia la clase acústica *ot*. Las clases acústicas se dividen en dos grupos: *mu* frente a *sp*, *sn* y *sm*.
- Diferenciados entre música, voz y "otros" (**MUSM-SP-OT**): sistema similar a MUSM-ALL, pero haciendo distinción entre la voz y la clase acústica "otros", modelando cada clase por separado. Las clases acústicas se dividen en tres grupos: *mu* y *sm* frente a *sp* y *sn* frente a *ot*.
- Diferenciador entre música y voz, sin tener en cuenta la clase acústica "otros" (**MUSM-SP**): sistema idéntico al anterior en el que se obvia la clase acústica *ot*. Las clases acústicas se dividen en dos grupos: *mu* y *sm* frente a *sp* y *sn*.

Uno de los objetivos generales a todos los sistemas desarrollados es el de poder compararse y a su vez fusionarse con el sistema presentado por el grupo ATVS en la evaluación ALBAYZIN 2010 de segmentación de audio. Puesto que ese sistema segmentaba el audio en cada una de las cinco clases acústicas disponibles, se ha tenido que llevar a cabo un ajuste en el etiquetado final para hacerlo coincidir con las tareas presentadas en este proyecto.

A lo largo de todo este capítulo se explicará el proceso llevado a cabo de manera genérica para alcanzar los objetivos de todos los sistemas implementados, mientras que en la sección 4. se estudiarán por separado los resultados de cada uno de los sistemas.

3.3 Base de datos

Para la realización de este proyecto se ha utilizado la base de datos proporcionada en la evaluación ALBAYZIN 2010 de segmentación de audio. Está formada por unas 87 horas de grabaciones de programas emitidos por el canal catalán de televisión 3/24. La distribución de las clases de audio contenidas en esta base de datos es la siguiente: 37 % de voz limpia (*sp*), 5 % de música (*mu*), 15 % de voz con música de fondo (*sm*), 40 % de voz con ruido de fondo (*sn*) y 3 % de otros (*ot*). En esta última clase se engloba todo el material que no pertenece a las cuatro clases anteriores, incluyendo el ruido.

La base de datos ha sido confeccionada y creada por el Centro de Investigación TALP de la Universitat Politècnica de Catalunya (UPC) y comentada por Verbio Technologies. La base de datos está compuesta por 24 ficheros WAV con una duración aproximada de 4 horas cada uno, y dividida en dos partes: los 16 primeros ficheros para entrenamiento/desarrollo (training/development) y los 8 restantes para evaluación (testing). Para el desarrollo de este proyecto se han utilizado los 12 primeros ficheros para el entrenamiento de modelos y los ficheros 13 a 16 para el ajuste de parámetros diferentes a los de los modelos. Las señales de audio siguen un formato PCM, mono, con resolución de 16 bits y muestreadas a 16 kHz.

El formato que siguen las etiquetas de Ground-Truth es el siguiente:

Campo 1	Campo 2	Campo 3	Campo 4
Nombre Archivo Audio	Tiempo Inicial	Tiempo Final	Clase Acústica

donde:
- Campo 1 = nombre del archivo que contiene el audio
- Campo 2 = instante temporal (segundos) en el que empieza una nueva clase acústica
- Campo 3 = instante temporal (segundos) en el que acaba la clase acústica
- Campo 4 = clase acústica detectada (en formato de dos letras: sp, mu…)

Ejemplo:

Audiofile1	13.011	14.462	sp
Audiofile1	15.959	19.882	ot
Audiofile2	19.972	25.482	mu

3.4 Diseño

Cualquiera de los sistemas desarrollados se compone de las mismas etapas antes de llegar a un resultado definitivo, a saber:

- Extracción de la entropía cromática a partir del audio (ver sección 2.2.3.2.).
- Cálculo mediante métodos estadísticos de las características musicales utilizadas en el modelado (ver apartado 2.2.4.).
- Uso del Modelo de Mezclas de Gaussianas y del UBM para el entrenamiento de modelos de las distintas clases acústicas (ver capítulo 2.2.2.2.).
- Tratamiento de las características en la etapa de reconocimiento, como la puntuación o scoring, los filtrados de scores o el tamaño de las ventanas (ver sección 2.2.2.3.).

A lo largo de todos los sistemas hay tres procesos principales: entrenamiento, desarrollo y evaluación. El entrenamiento hace uso de doce sesiones de la base de datos para calcular los modelos de las distintas clases acústicas a reconocer, mientras que la etapa de desarrollo sirve para estudiar y modificar los distintos ajustes de post-procesado, como el filtrado de scores para evitar fluctuaciones en el etiquetado de salida. El proceso de desarrollo hace uso de cuatro sesiones de la base de datos. Por último, el proceso de evaluación es muy similar al de evaluación, con la salvedad que sólo se utilizan los mejores parámetros calculados precisamente durante el desarrollo. De esta forma se obtiene el rendimiento final de cada sistema.

En las siguientes figuras (Figura 3-1, 3-2 y 3-3) se pueden apreciar los diagramas de flujo por cada uno de estos procesos:

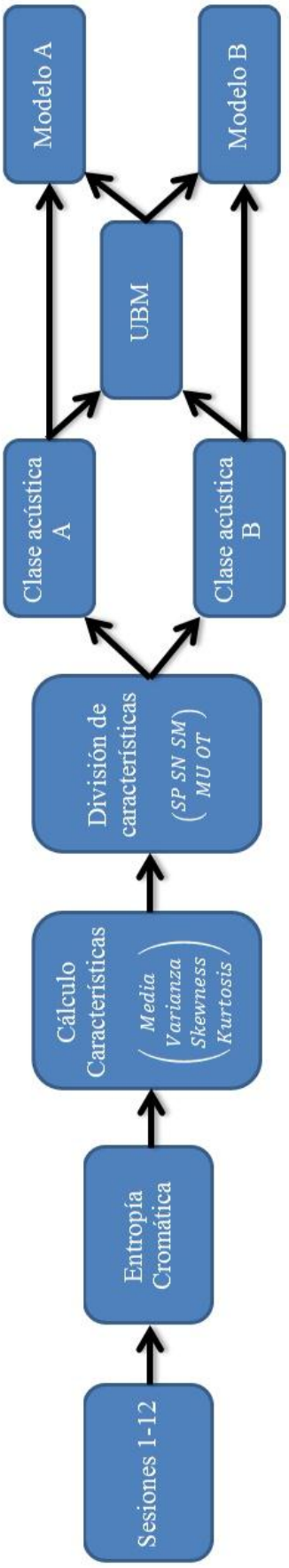


Figura 3-1: Diagrama de flujo de la etapa de entrenamiento de modelos del sistema propuesto

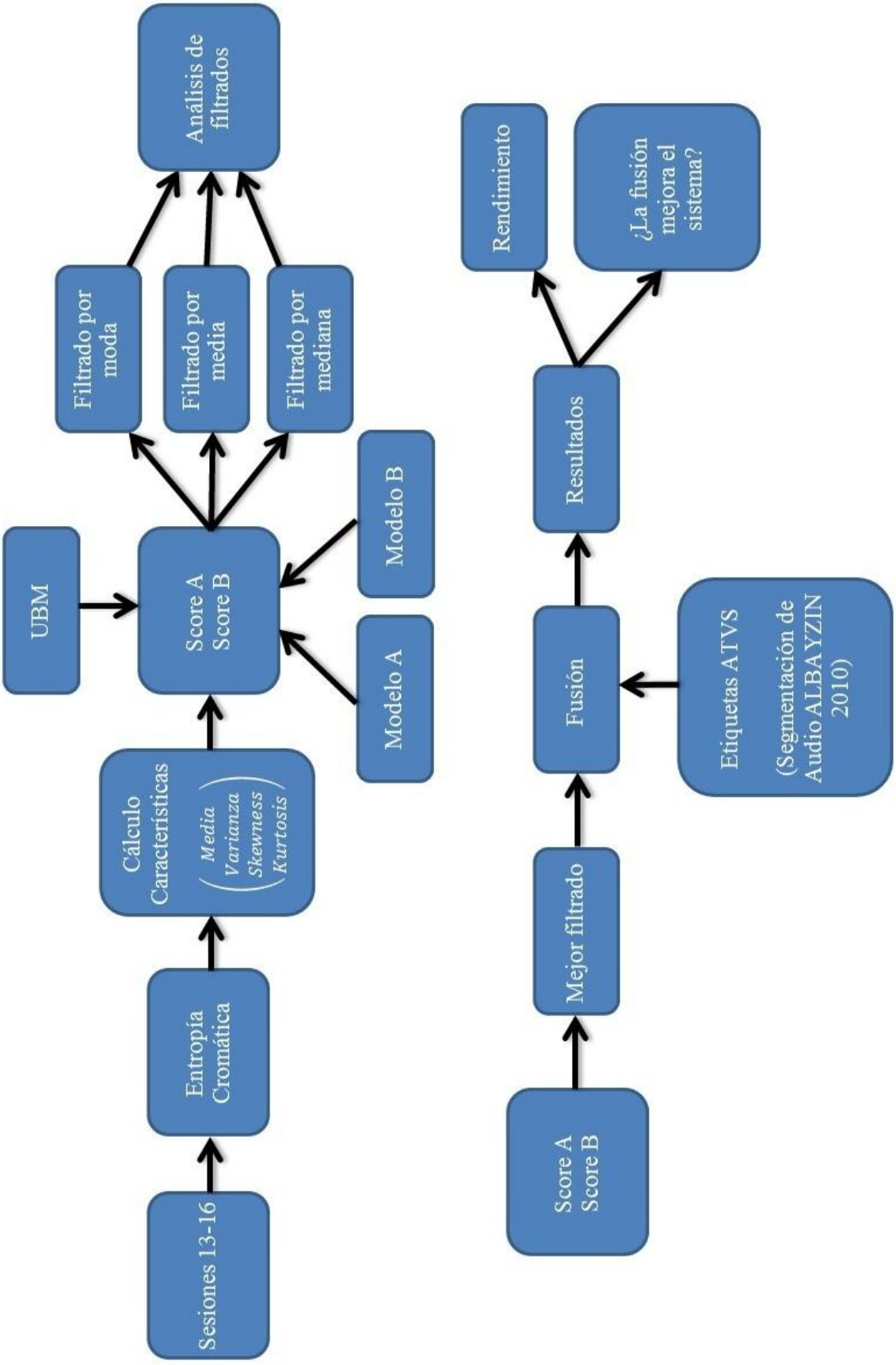


Figura 3-2: Diagrama de flujo de la etapa de desarrollo del sistema propuesto

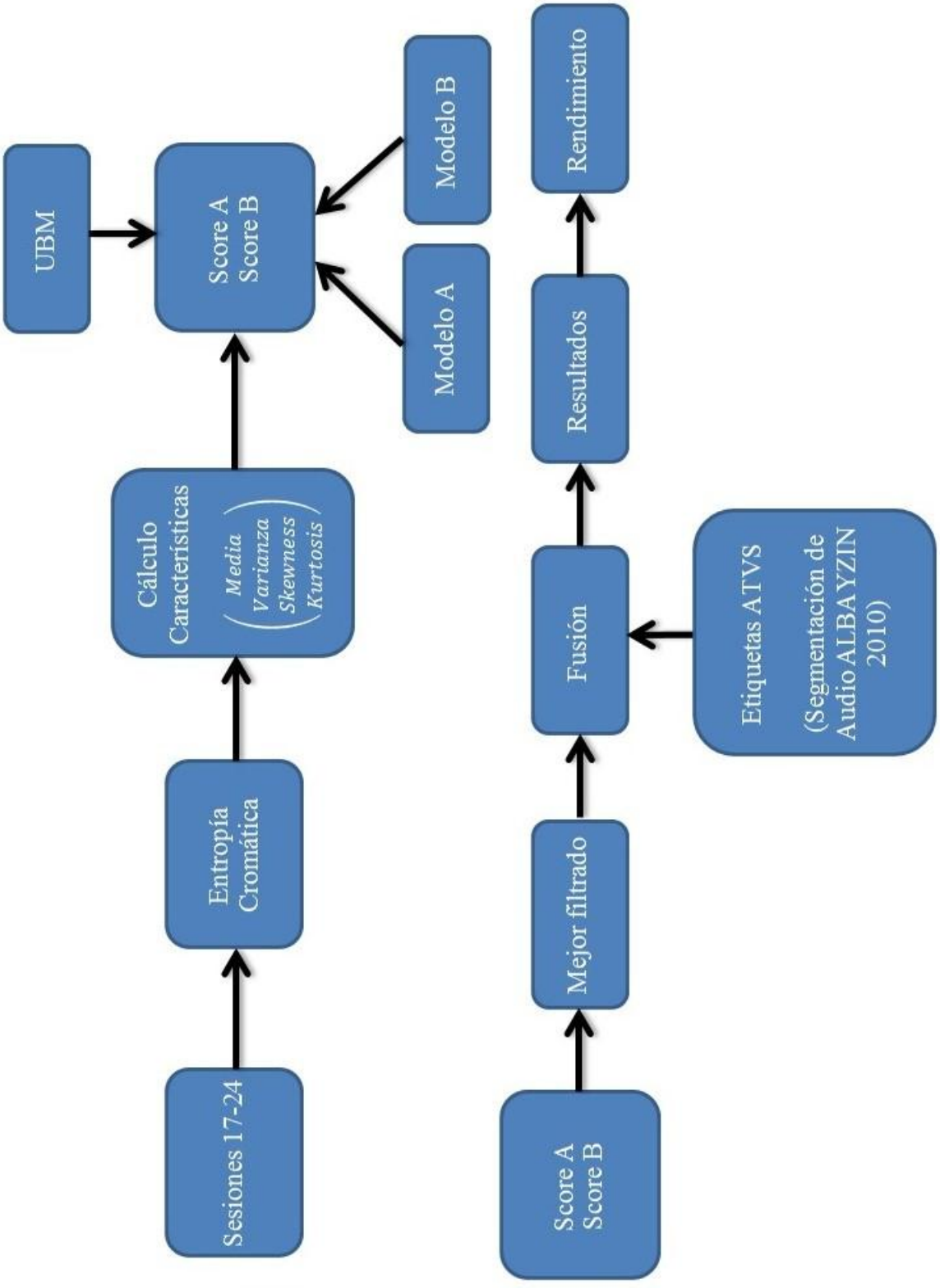


Figura 3-3: Diagrama de flujo de la etapa de evaluaión de modelos del sistema propuesto

3.4.1 Extracción de la entropía cromática a partir del audio

El cálculo de la entropía cromática es el primer paso del sistema implementado (es común a todos los subsistemas desarrollados). El proceso a seguir es el siguiente:

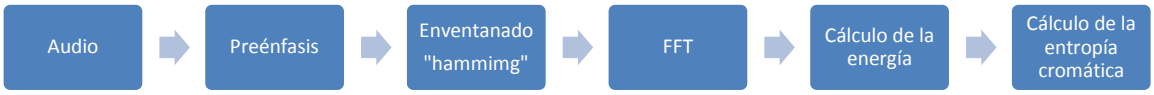


Figura 3-4: Proceso de extracción de la entropía cromática a partir del audio de entrada

Antes de cualquier cálculo se realiza un preénfasis a la señal de audio con el fin de acentuar las frecuencias más elevadas, y por consiguiente distinguir con mayor facilidad audios musicales de locuciones. El rango frecuencial de una locución es mucho menor que el de la música en general. Típicamente la voz humana oscila entre alrededor de 60 y 7000 Hz, mientras que las composiciones musicales son capaces de llenar todo el espectro auditivo (20 Hz - 20 kHz). La ecuación de un filtro de preénfasis sencillo es la siguiente:

$$y(n) = x(n) - ax(n-1) \tag{3.1}$$

donde $x(n)$ es el audio de entrada y a es el factor de filtrado, que en este caso toma el valor de 0.97.

A continuación se realiza un enventanado Hamming, tal y como se introdujo en la sección 2.2.2. mediante la ecuación (2.3). Se ha utilizado un tamaño de ventana de 20 milisegundos con un solapamiento del 50 %, es decir, las tramas comienzan cada 10 milisegundos.

Para calcular el espectro de potencia se calcula primero la FFT (Fast Fourier Transform) de cada una de las ventanas en las que se ha dividido el audio operando a continuación la potencia. La FFT empleada utiliza $2^{13} = 8192$ puntos de resolución, siendo un valor lo suficientemente grande como para que todas las L sub-bandas en las que se divide la FFT para el cálculo de la entropía cromática tengan alguna componente frecuencial (apartado 2.2.3.2.). El número de puntos es esencial para la etapa inmediatamente posterior, pues para calcular la energía en diferentes sub-bandas se necesita buena resolución a frecuencias bajas (ya que el método a seguir emula al sistema auditivo humano).

Tal y como se introdujo en la sección 2.2.3.2., la entropía cromática se calcula a partir de las energías de las distintas sub-bandas localizadas según la escala musical (ecuación (2.19)). Este filtrado por bandas se realiza de manera análoga al cálculo de MFCCs, cambiando las frecuencias de la escala Mel por las notas musicales (ecuación (2.18)).

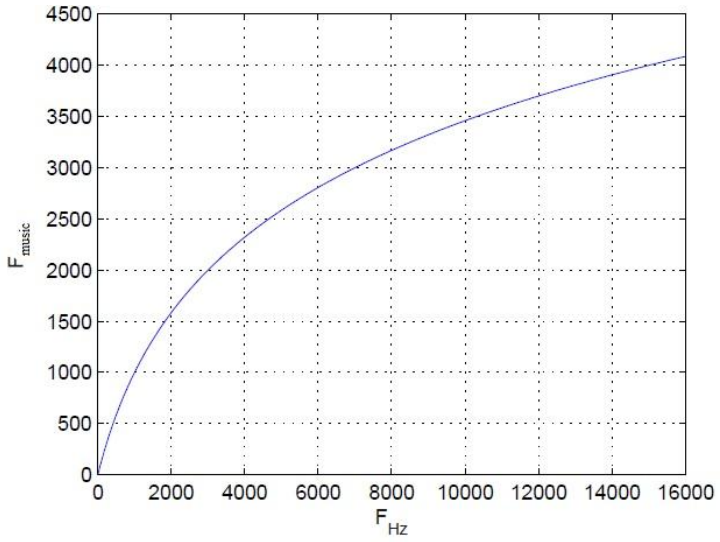


Figura 3-5: Relación entre las frecuencias en escala lineal y la escala musical

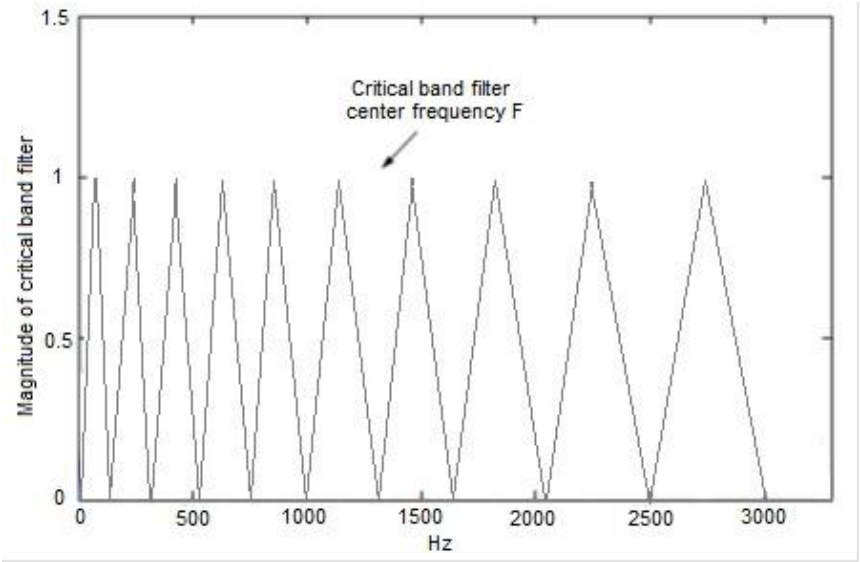


Figura 3-6: Banco de filtros para la escala musical. Cada pico corresponde con una nota musical

Teniendo en cuenta que la frecuencia de muestreo (sr) del audio de la base de datos es de 16 kHz, tomando como primera nota audible el Do de 65,406 Hz y que las distintas notas musicales siguen un patrón exponencial, tendremos un total de sub-bandas de:

$$N_{sub-bandas} = \left\lceil 12 \left(\log_2 \left(\frac{sr/2}{65{,}406\ Hz} \right) \right) \right\rceil = \lceil 83{,}21 \rceil = 84 \qquad (3.2)$$

Aunque el espectro audible empieza en 20 Hz se ha tomado como primera frecuencia de análisis el Do de 65,406 Hz, ya que de escoger un frecuencia menor habría que aumentar la finura de la FFT, es decir, aumentar el número de puntos, para que todas las sub-bandas donde se calcula la energía (apartado 2.2.3.2.) tengan componentes frecuenciales, lo que implicaría un incremento excesivo en la carga computacional.

Por lo tanto, las frecuencias centrales de los filtros comenzarán en 65,406 Hz (Do 1) y llegarán hasta la nota inmediatamente inferior a la tasa de muestreo dividida por 2 ($^{sr}/_{2} = 8000\ Hz$), siendo en este caso 7902,133 Hz (Si 7). Se pueden consultar el resto de frecuencias centrales en el Anexo A.

El rango de frecuencias del que se compone la sub-banda SB_i, siendo f_i la frecuencia central de dicha sub-banda, va desde $\frac{f_i+f_{i-1}}{2}$ hasta $\frac{f_i+f_{i+1}}{2}$ (la frecuencia máxima de la sub-banda SB_i coincide con la frecuencia mínima de la sub-banda SB_{i+1}), definiéndose los filtros de manera asimétrica ya que las frecuencias centrales crecen de manera exponencial, es decir, el rango de frecuencias desde la frecuencia mínima de la sub-banda SB_i hasta la frecuencia central f_i es menor que el rango frecuencial desde la frecuencia central f_i hasta la frecuencia máxima de la sub-banda SB_i.

Finalmente la entropía cromática se calcula según (2.20) a partir de las energías.

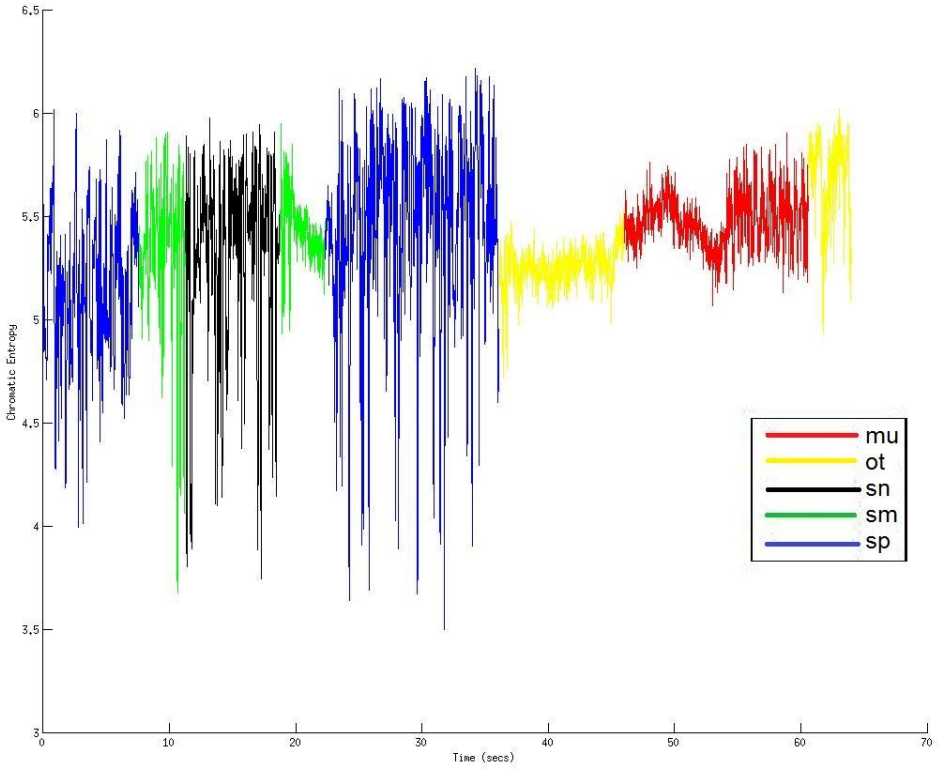


Figura 3-7: Entropía cromática dividida en cinco clase acústicas (*sp*, *sn*, *sm*, *mu* y *ot*)

3.4.2 Cálculo de las características musicales

A partir de la entropía cromática se calculan los cuatro estadísticos introducidos en el apartado 2.2.4. utilizando una ventana deslizante de tamaño 1 segundo con un paso de 10 milisegundos, es decir, por cada trama a corto plazo sobre la que se calcula la entropía cromática.

En la siguiente figura se muestra la entropía cromática antes calculada junto con sus cuatro estadísticos que discriminarán el audio para su posterior segmentación:

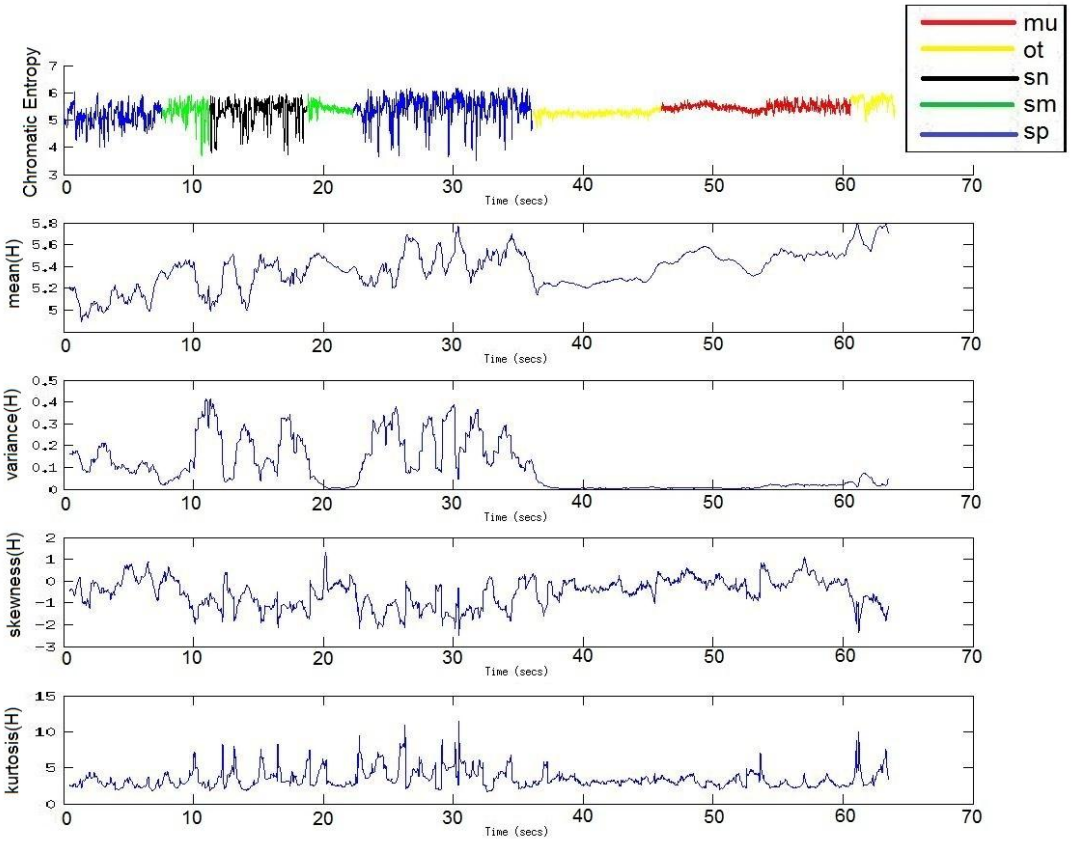


Figura 3-8: Entropía cromática y sus 4 características: media, varianza, skewness y kurtosis

3.4.3 Entrenamiento GMM-UBM

A continuación se juntan las características de las 12 primeras sesiones de la base de datos y se dividen entre las cinco clases acústicas, ayudándose de las etiquetas de Ground-Truth. De esta manera se pueden entrenar los modelos específicos para cada una de las clases.

Esta etapa es la primera que presenta diferencias dependiendo del sistema a desarrollar, puesto que cada uno aglomera las clases acústicas de manera distinta para entrenar sus modelos: el sistema **MU-ALL** agrupa en un modelo las clase acústica *mu* y en otro modelo las restantes (*sp*, *sn*, *sm* y *ot*); el sistema **SP-NSP** agrupa en un modelo las clases acústicas *sp, sn* y *sm* mientras que entrena otro modelo con *mu* y *ot*; etc.

El proceso a seguir es tomar los vectores de las cuatro características divididos según las clases acústicas y formar tantos grupos como modelos se vayan a entrenar. Para evitar la descompensación de datos en unos y otros modelos (ya que la base de datos divide de manera no igualitaria la cantidad de audio, teniendo por ejemplo un 40 % etiquetado como *sn* y sólo un 5 % como *mu*) se hace un diezmado de aquellas clases que tengan más muestras, equiparándose al modelo que de menos datos disponga. De esta forma, el UBM representará por igual todas las clases acústicas.

Puesto que cada sistema desechará una cantidad distinta de datos, no se puede hacer uso de un mismo UBM, aunque todos los UBMs entrenados representan la población mundial de las cinco clases acústicas. Para el cálculo de los UBMs se ha entrenado un sistema GMM de 128 mezclas gaussianas con 10 iteraciones del algoritmo Maximun Likelihood y 5 del algoritmo K-means.

Inmediatamente después se procede a adaptar los distintos modelos a partir del UBM entrenado mediante una adaptación MAP completa (adaptación del vector de pesos, del vector de medias y de la matriz de covarianzas) de 10 iteraciones con un factor de relevancia (relevance factor) [Huai-You et al., 2012] igual a 8. Se ha comprobado de manera experimental el bajo nivel de influencia del factor de relevancia escogido, variando los rendimientos finales en apenas varias centésimas porcentuales. El principal motivo de utilizar una adaptación MAP completa es que el vector de medias no es capaz de discriminar las clases acústicas de voz (*sp*, *sn* y *sm*) y música (*mu*). Este razonamiento se estudiará con mayor detenimiento en la sección 4.1.

3.4.4 Segmentador de características musicales y scoring

Con esta etapa comienza el proceso de desarrollo (development) y de evaluación (testing). Aquí se repite todo el cálculo llevado a cabo en los apartados 3.4.1. y 3.4.2. para las sesiones 13 a 16 (development) o 17 a 24 (testing), a la vez que se hace uso de los resultados obtenidos en el apartado 3.4.3.

De esta manera se enfrentan las cuatro características extraídas del audio a analizar contra los distintos modelos de cada uno de los sistemas (normalizando las puntuaciones mediante el UBM). Previamente al cálculo de scores (ver sección 2.2.2.3.) se procede a identificar las 5 mezclas más pesadas (mezclas que aportan mayor densidad de probabilidad) del UBM para cada vector de características. De esta manera se consiguen

resultados prácticamente idénticos a si se usasen las 128 mezclas, pero con el consiguiente ahorro computacional.

Para la puntuación de cada uno de los vectores de características se hace uso de la función de densidad de probabilidad (2.7) del GMM de cada uno de los modelos, teniendo así tantos scores como modelos tenga el sistema. Gracias al uso del UBM en la normalización de puntuaciones mediante el algoritmo LR (apartado 2.2.2.3.1.) se pueden comparar directamente y tomar la más alta como el modelo que identifica con mayor probabilidad al vector de características analizado.

El proceso posterior al cálculo de puntuaciones es el filtrado de tales scores y la segmentación en distintas clases acústicas. Para encontrar los valores óptimos que maximicen el rendimiento de todos los sistemas se realizan numerosas pruebas en el proceso de desarrollo, en el que se varían parámetros como el tamaño de los filtros de scores como el tipo de filtrado (ver apartado 2.2.5.1.).

Una vez analizados los resultados y determinado el tipo de filtrado que mejor combina con cada uno de los siete sistemas implementados se pasa al proceso de evaluación (testing), en el que únicamente se emplean los parámetros seleccionados en la etapa de desarrollo y se obtienen los resultados o rendimientos finales de los sistemas. Para determinar este rendimiento final, se hace uso de (2.25), en el que se enfrenta el fichero resultante que segmenta el audio frente a las etiquetas de Ground-Truth proporcionadas por la evaluación ALBAYZIN 2010.

3.5 Fusión con el sistema de segmentación de audio ATVS para la evaluación ALBAYZIN 2010 a nivel de etiqueta

Se ha implementado una fusión con el sistema de segmentación de audio ATVS para la evaluación ALBAYZIN 2010 a nivel de etiqueta o basadas en decisiones. Es decir, una fusión de las etiquetas de salida de cada uno de los sistemas en el que aparece el audio segmentado en las distintas clases acústicas.

Tal y como se adelantaba en el apartado 3.2., el sistema ATVS para ALBAYZIN 2010 divide el audio en cinco clases acústicas (*sp*, *sn*, *sm*, *mu* y *ot*). De tal forma se han agrupado los resultados para hacerlos coincidir con las nuevas clases acústicas definidas en cada uno de los siete sistemas implementados (*sp+sn+sm*, *sp+sn*, *mu+ot*, *mu+sm*…).

El sistema ATVS de segmentación de audio para ALBAYZIN 2010 está basado en el alineamiento de Viterbi de cadenas de MFCC extraídos del audio usando un HMM de cinco estados. Cada uno de los cinco estados se corresponde con una clase acústica (*sp*, *sn*, *sm*, *mu* y *ot*).

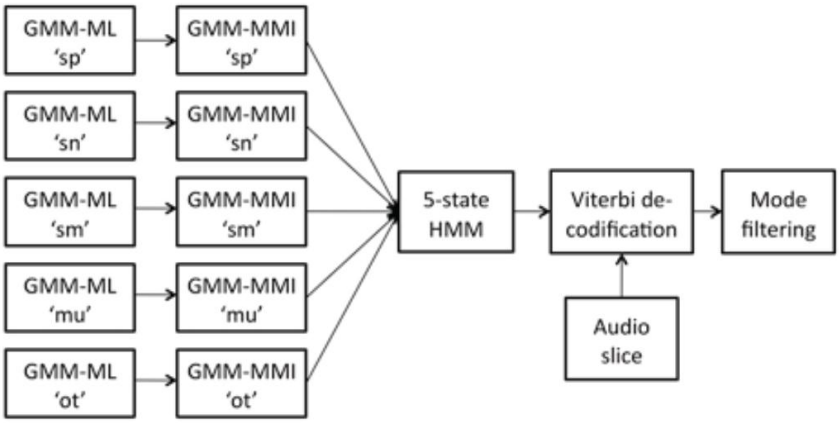


Figura 3-9: Diagrama esquemático del sistema de segmentación de audio ATVS para ALBAYZIN 2010 [Franco-Pedroso et al., 2010]

Cada estado del HMM consiste en un GMM de 1024 mezclas, entrenado previamente mediante 5 iteraciones del algoritmo Maximun Likelihood (ML) y mejorado a posteriori con 18 iteraciones del algoritmo Maximun Mutual Information (MMI).

A partir del HMM se realiza un alineamiento de Viterbi (HMM Toolbox de Matlab™ de Kevin Murphy). Tras la decodificación de Viterbi, se aplica un filtrado de modas con una ventana deslizante de 700 milisegundos, con el fin de evitar cambios espurios entre estados.

Al no disponer de las puntuaciones obtenidas por el sistema ATVS para ALBAYZIN 2010, se ha optado por una fusión exclusivamente de etiquetas. Esta fusión intenta definir las bondades de la mezcla de ambos sistemas, para en un futuro implementar una fusión más avanzada como las vistas en el apartado 2.2.2.5.

La fusión a realizar se hará mediante ventanas de 10 milisegundos sin solapamiento. Se han llevado a cabo dos tipos de fusiones:

- Fusión OR: siempre que la clase acústica a estudiar (AC_i) se identifique en alguno de los dos sistemas, el sistema fusión tendrá como salida que la clase acústica correspondiente a ese intervalo temporal sea AC_i.
- Fusión AND: el sistema fusión sólo determinará que la clase acústica perteneciente a un determinado intervalo temporal es AC_i si para el mismo intervalo temporal tanto en el sistema implementado como en el sistema ATVS para ALBAYZIN 2010 la clase acústica es AC_i.

Se ha seguido el convenio de aplicar la fusión sobre la primera clase definida en cada sistema. Así, en el sistema **MU-SP** se fusionará únicamente la clase *mu*, ya que de hacerlo sobre *sp* sólo invertiría el rendimiento, es decir, el resultado de fusionar la clase *mu* mediante un AND es el mismo que fusionar la clase *sp* mediante un OR.

4 Análisis de resultados

En este capítulo se explican y estudian los resultados obtenidos a partir de los siete sistemas desarrollados, así como los parámetros elegidos tanto para la adaptación MAP como en la etapa de desarrollo. También se hace una comparación de los sistemas presentados frente al sistema ATVS para ALBAYZIN 2010.

4.1 Parámetros utilizados

4.1.1 Adaptación MAP

Uno de los principales problemas que se han tenido que abordar a lo largo del proyecto ha sido la adaptación MAP.

En un principio se optó por adaptar los distintos modelos desde el UBM como se ha venido haciendo en la mayoría de los sistemas GMM-UBM (en tareas como el reconocimiento de locutores), es decir, adaptando únicamente la matriz de medias. Los resultados obtenidos distaban mucho de los rendimientos que presentaban otros sistemas en la evaluación ALBAYZIN 2010, puesto que por algún motivo el sistema implementado no conseguía diferenciar correctamente entre las distintas clases acústicas (*sp*, *sn*, *sm*, *mu* y *ot*).

Para entender este concepto hace falta recurrir a la distribución de las cuatro características (media, varianza, skewness y kurtosis). Puesto que tal estudio sería imposible de visualizar gráficamente al tratar con espacios tetradimensionales, se ha llevado a cabo un estudio conocido generalmente como "búsqueda exhaustiva" para ver qué características son las que más discriminan el audio, probando todas las combinaciones posibles de los cuatro estadísticos y volviendo a simular el sistema completo. Se constata que los mejores resultados se obtienen utilizando los cuatro primeros momentos centrales (media, varianza, skewness y kurtosis). Sin embargo, se obtienen resultados muy similares desechando la skewness o la kurtosis, pero no ambas a la vez. La mejora relativa de usar los cuatro estadísticos en lugar de desechar la skewness o la kurtosis es de apenas el 5 %, mientras que no utilizar la media, la varianza o la skewness y kurtosis supone un descenso en el rendimiento de más del 280 %.

Haciendo uso del estudio de "búsqueda exhaustiva", se pueden estudiar con más detalle las características más determinantes del sistema: la media y la varianza. Así pues, y simulando el sistema teniendo en cuenta únicamente estas características se pueden apreciar mejor las diferencias entre distintas clases acústicas

Una forma sencilla de ver dichos estadísticos es recurrir a los GMMs obtenidos a partir del UBM mediante una adaptación MAP. Sin embargo, estas representaciones tienen el peligro de perder cualquier noción de temporalidad y puede haber un sobreajuste de los datos, no viendo exactamente las características que se tienen. Así, podemos ver un gran lóbulo para las 5 clases acústicas en torno a media = 5,5 y varianza = 0,2. También se puede apreciar un segundo lóbulo más pequeño para medias en torno a 6 (se pueden consultar el resto de representaciones gaussianas para las demás clases acústicas, así como las estimaciones de densidad de kernels bivariados en el Anexo C):

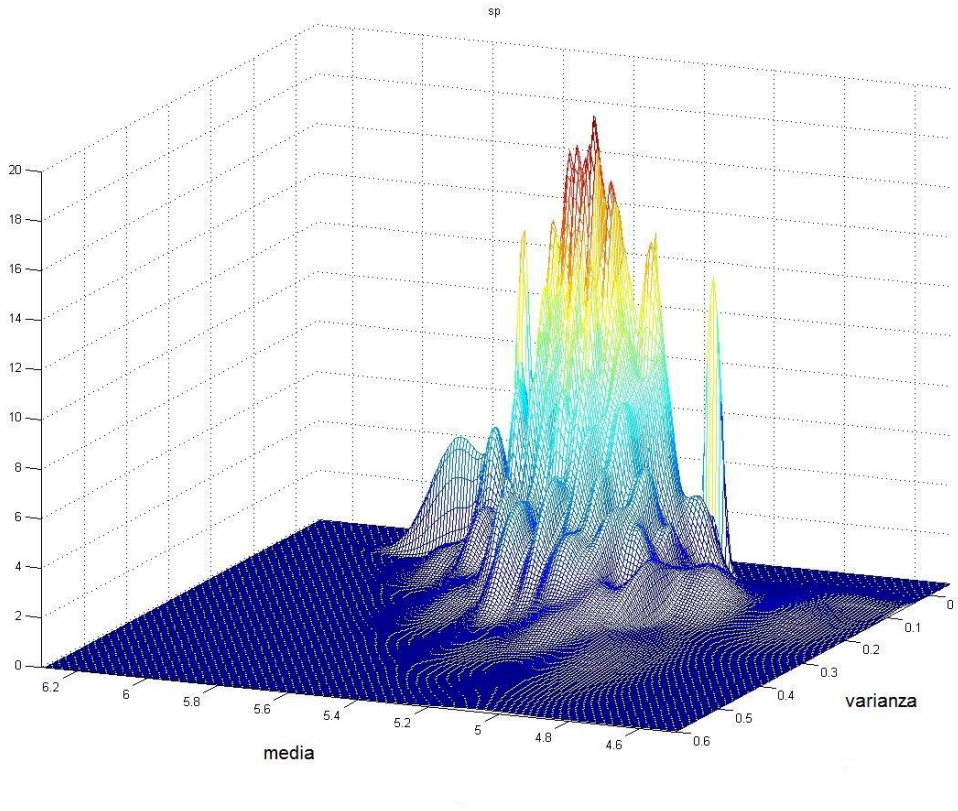


Figura 4-1: GMM adaptando sólo medias para *sp*

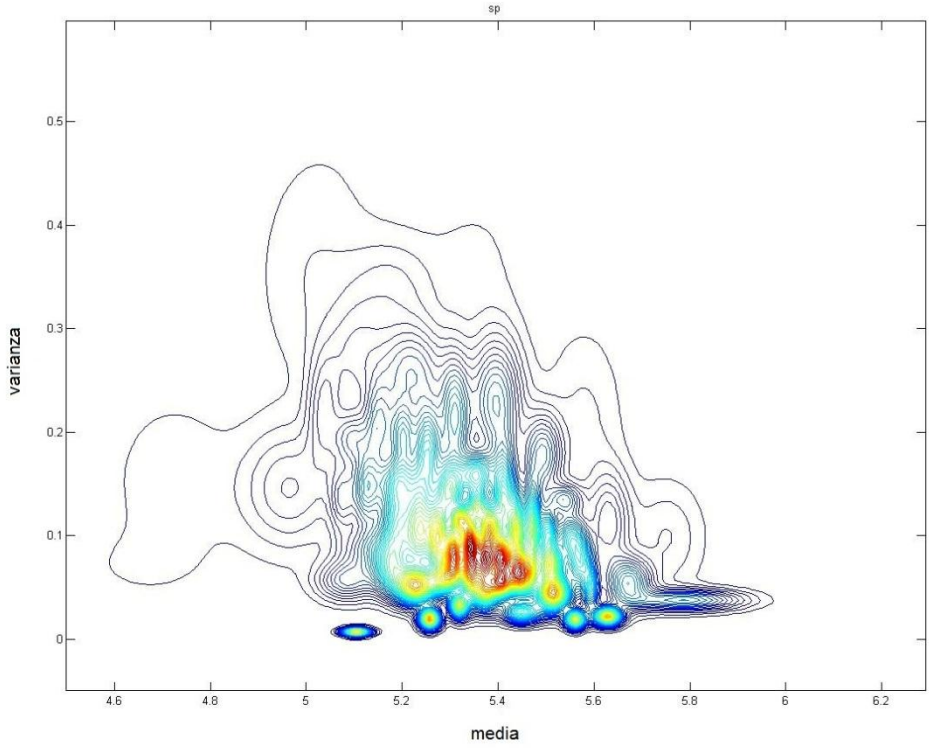


Figura 4-2: Curvas de nivel del GMM adaptando sólo medias para *sp*

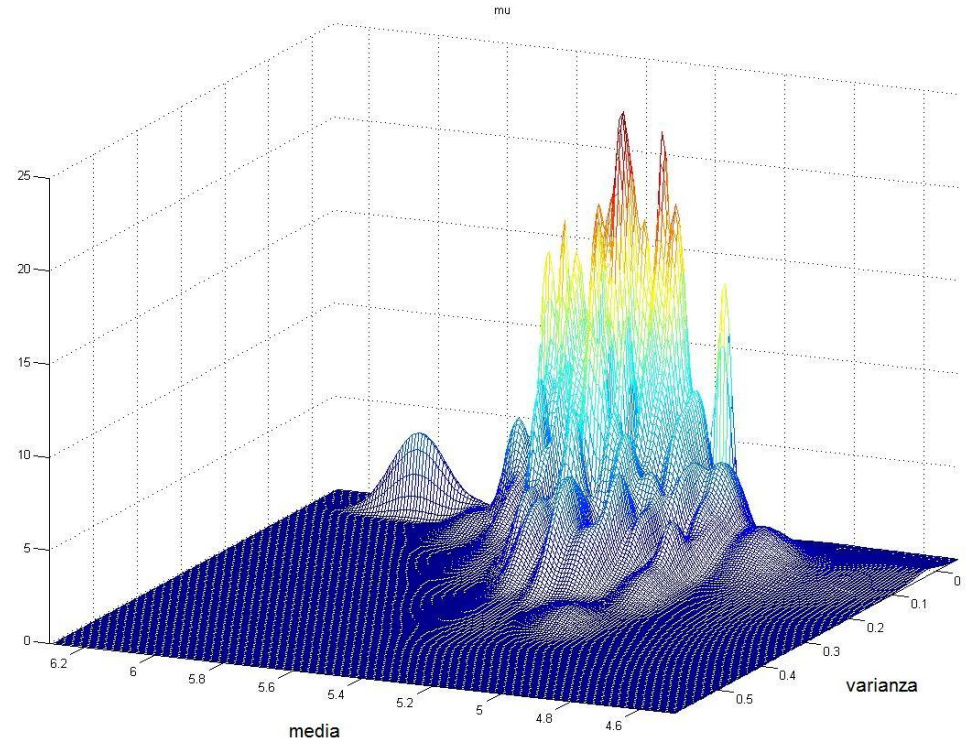


Figura 4-3: GMM adaptando sólo medias para *mu*

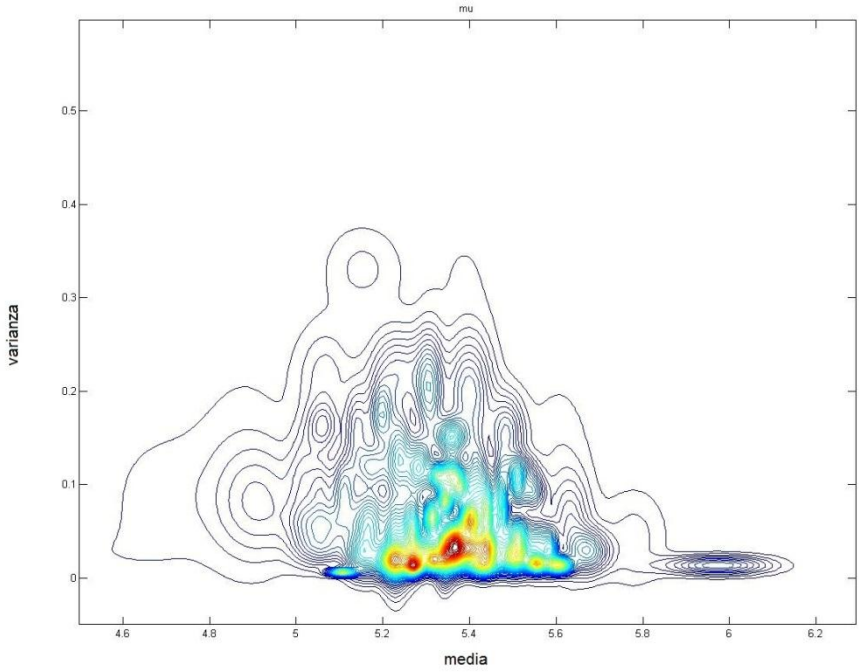


Figura 4-4: Curvas de nivel del GMM adaptando sólo medias para *mu*

Sin embargo, recurriendo a estimaciones de densidad de kernels bivariados (KDF) se puede aprovechar la visualización tridimensional de los modelos pero sin perder la información aportada por la posición espacial de las características.

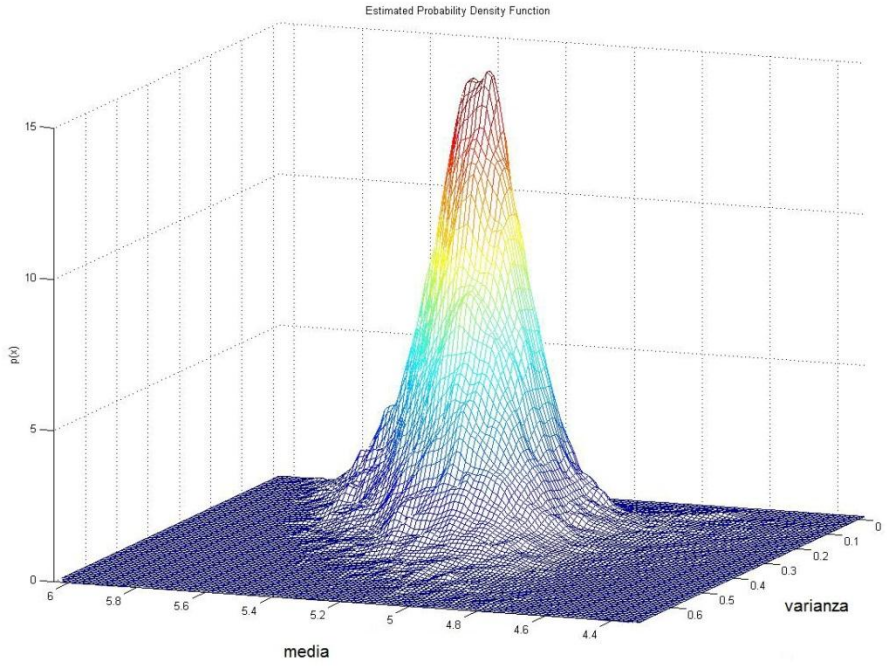


Figura 4-5: Densidad de probabilidad con KDF para *sp*

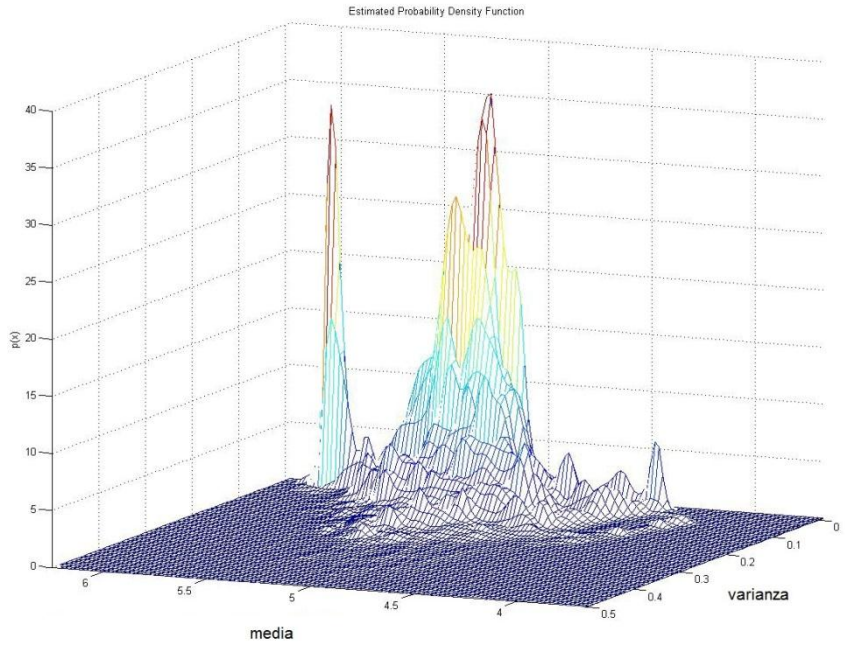


Figura 4-6: Densidad de probabilidad con KDF para *mu*

La primera información relevante que se puede extraer de estos gráficos es la ausencia de ese segundo lóbulo situado en medias cercanas a 6 para las clases acústicas contenedoras de habla (speech, speech-noise, speech-music), mientras que si se aprecia dicho lóbulo para la clase de music. De hecho, este lóbulo es tan importante para music como su "lóbulo principal", localizado en medias cercanas a 5,3 y varianzas en torno a 0,2.

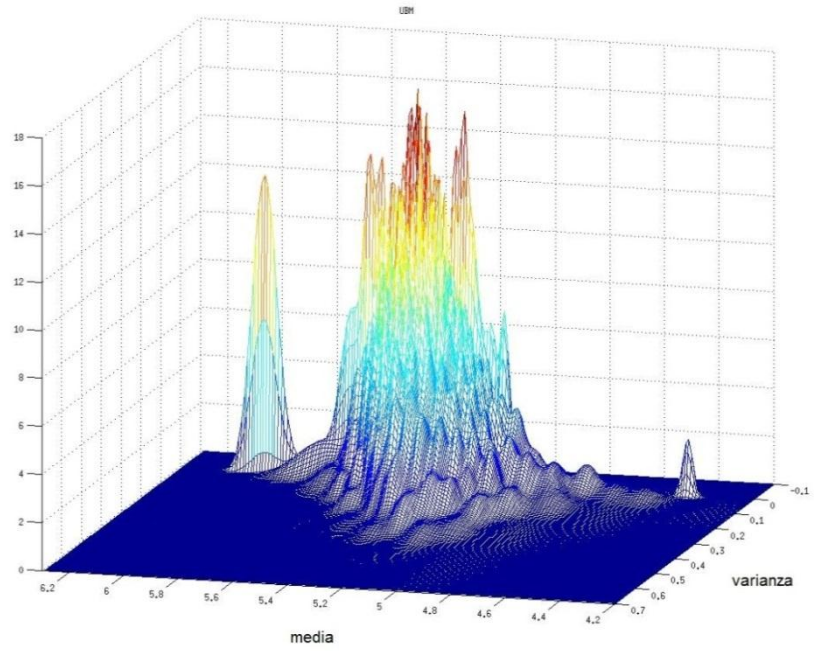


Figura 4-7: UBM para las cinco clases: *sp*, *sn*, *sm*, *mu* y *ot*

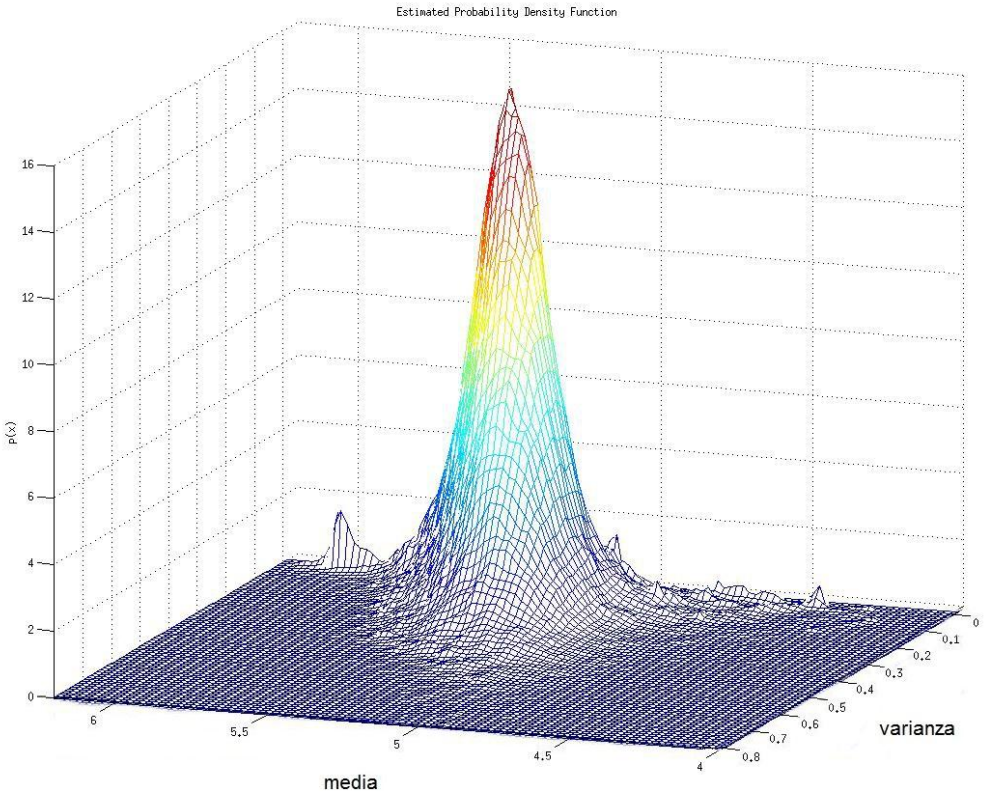


Figura 4-8: Densidad de probabilidad con KDF del UBM

Esta notable diferencia con los modelos adaptados puede radicar en la adaptación MAP llevada a cabo a partir del UBM, que adapta sólo las medias de las componentes gaussianas, pues este modelo universal contiene características de todas las clases acústicas, y el caso particular de la música se extrapola a otras clases. De tal manera se podría decir que este tipo de adaptación MAP es ineficiente y escasa. Por ejemplo, el lóbulo que sólo aparece en la densidad KDF de *mu* se manifiesta en el UBM, y por lo tanto, al sólo adaptar las medias de las componentes, aparecerá con mayor o menor desplazamiento en los modelos GMM adaptados de *sp*, *sm*, *sn*. Sin embargo, estos últimos no presentan dicho lóbulo en sus distribuciones KDF.

Para suplir estos problemas se propone una adaptación MAP completa en la que no sólo se adapte los vectores de medias, ya que ese segundo lóbulo, presente únicamente para la clase *mu*, se divide a partes iguales entre todas las clases a la hora de adaptar. Haciendo uso de una adaptación completa (pesos, medias y covarianzas) se consigue discriminar mejor ese lóbulo y darle mayor peso únicamente para la clase *mu*. En las siguientes figuras se pueden apreciar las representaciones gaussianas de los modelos *sp* y *mu* utilizando una adaptación MAP completa:

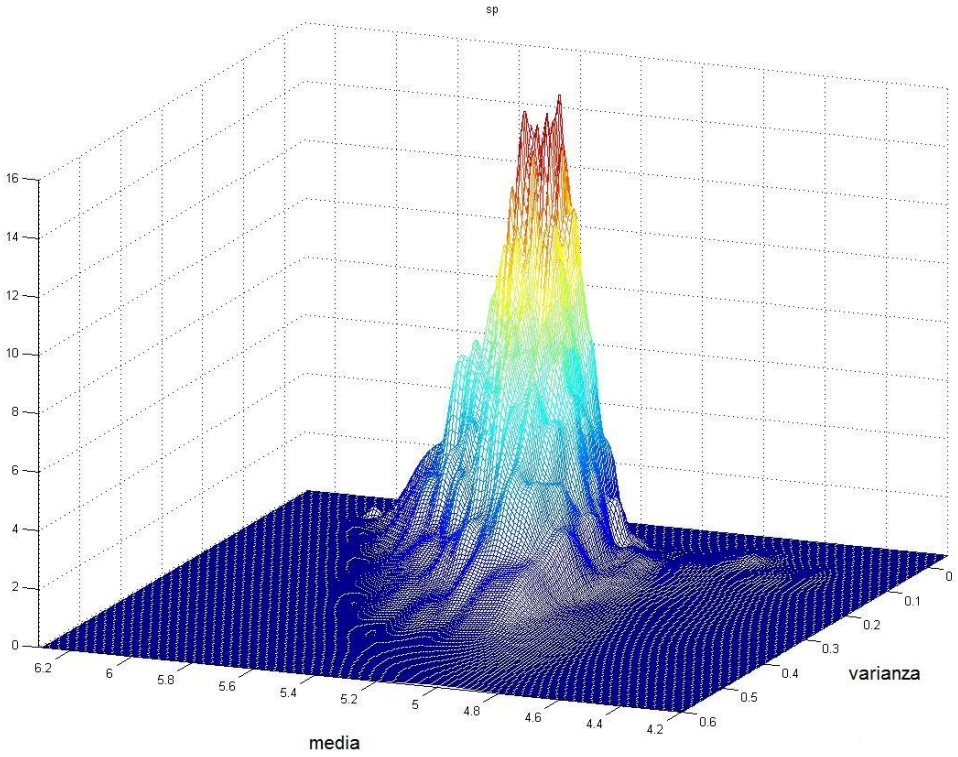


Figura 4-9: GMM adaptando todos los parámetros para *sp*

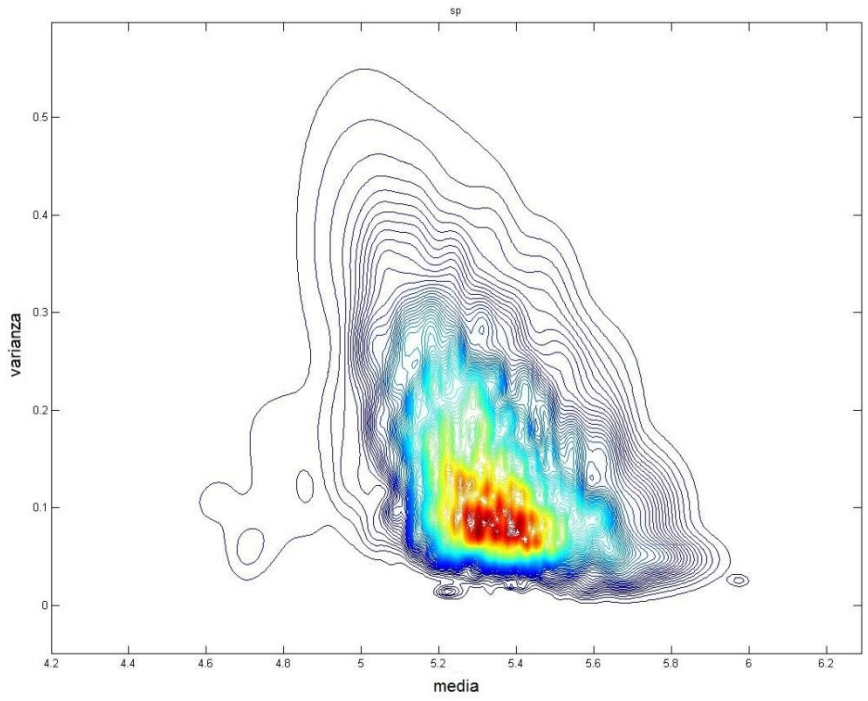


Figura 4-10: Curvas de nivel del GMM adaptando todos los parámetros para *sp*

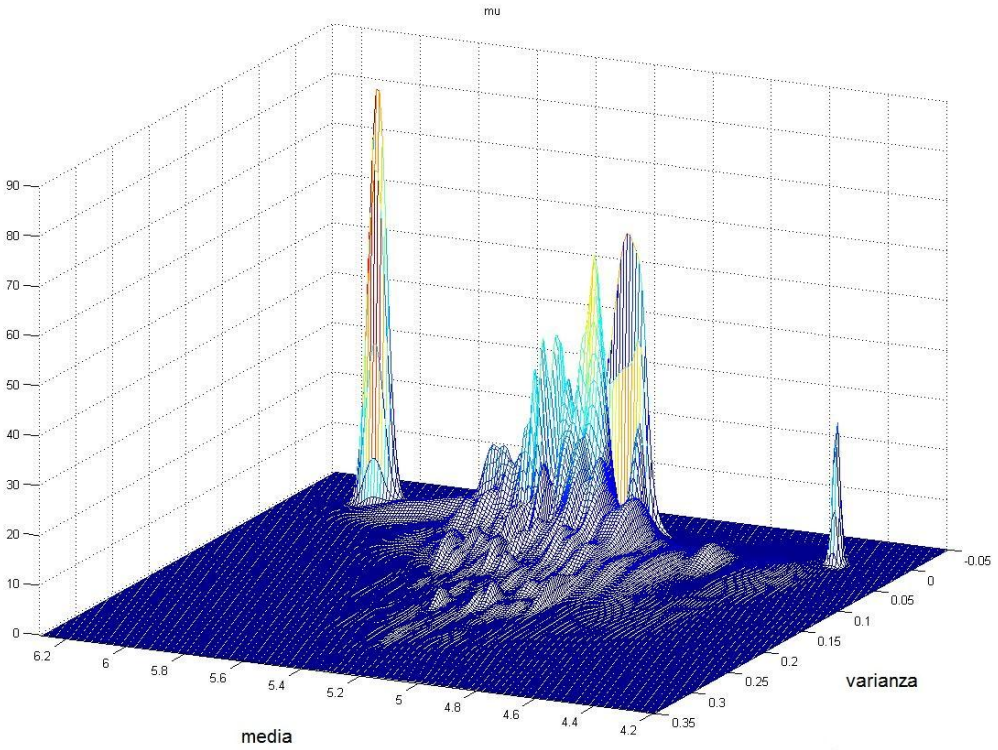


Figura 4-11: GMM adaptando todos los parámetros para *mu*

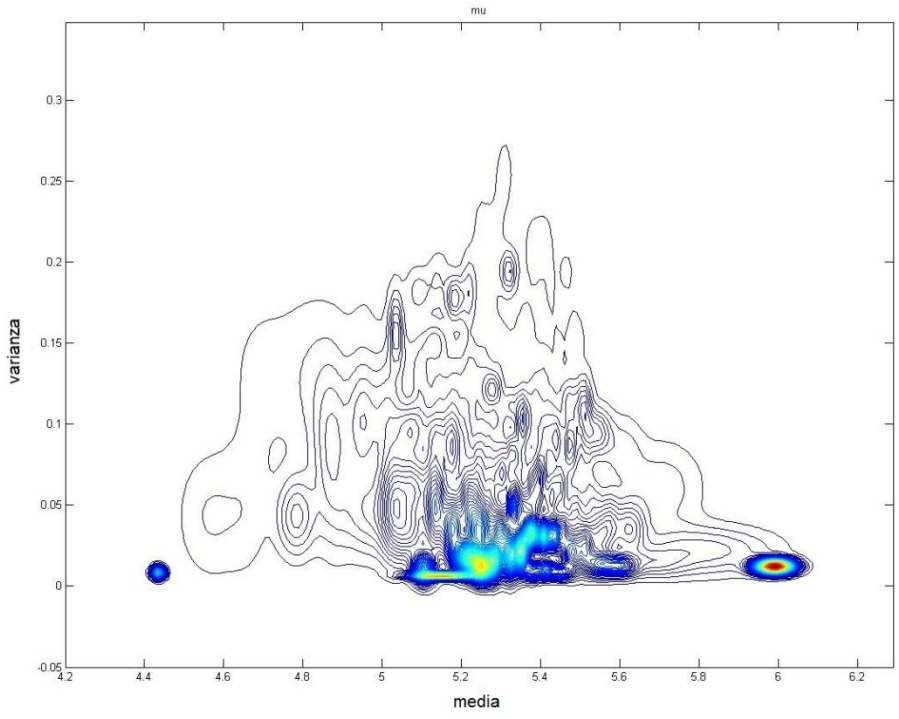


Figura 4-12: Curvas de nivel del GMM adaptando todos los parámetros para *mu*

Se comprueba la notable diferencia en las representaciones gaussianas de los modelos de las distintas clases acústicas, lo que repercute directamente en el rendimiento final de los sistemas, acercándose a otros como el presentado por el grupo ATVS en la evaluación ALBAYZIN 2010 de segmentación de audio.

4.1.2 Etapa de desarrollo (development)

Durante la etapa de desarrollo se realiza un estudio exhaustivo del proceso de filtrado de scores, ya sea el tipo de filtro a utilizar (moda, media o mediana) o el tamaño de la ventana deslizante de dicho filtro. Se pueden ver ejemplos prácticos de la etapa de desarrollo en el Anexo B. De esta forma, los resultados obtenidos para cada uno de los siete sistemas desarrollados son (siendo los ejes de abscisas el tamaño de la ventana deslizante y el eje de ordenadas el error de segmentación):

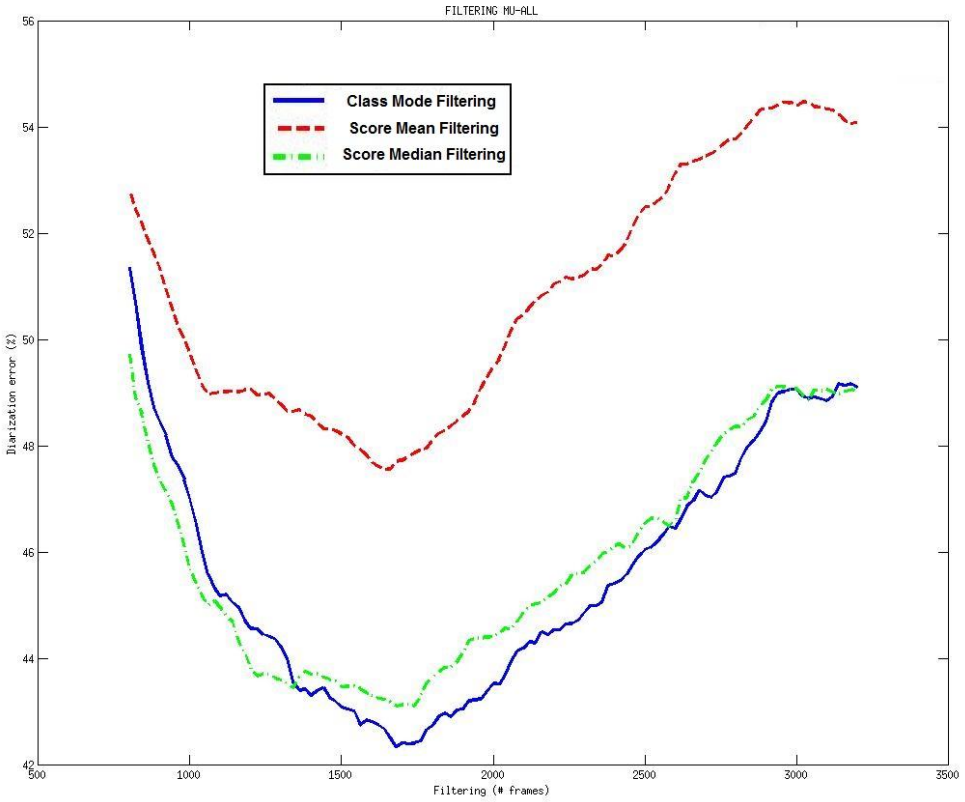


Figura 4-13: Análisis del filtrado de scores para el sistema MU-ALL

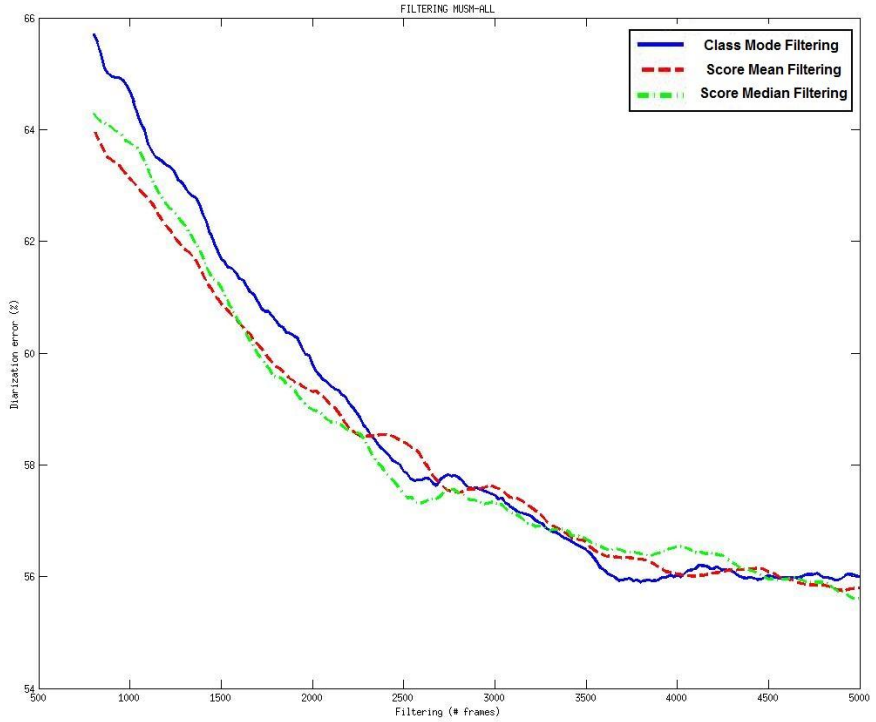


Figura 4-14: Análisis del filtrado de scores para el sistema MUSM-ALL

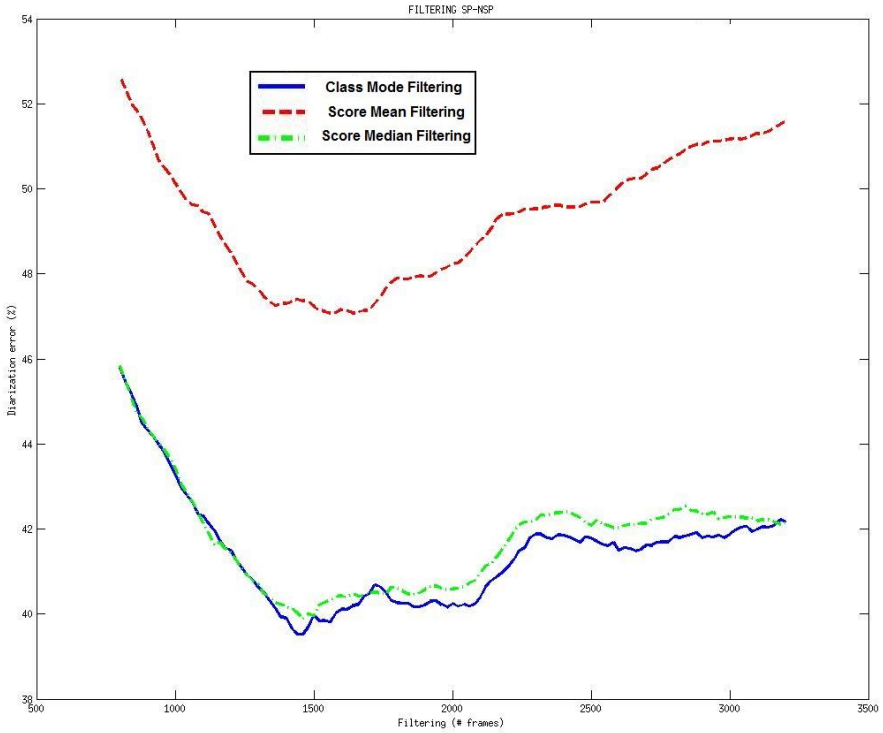


Figura 4-15: Análisis del filtrado de scores para el sistema SP-NSP

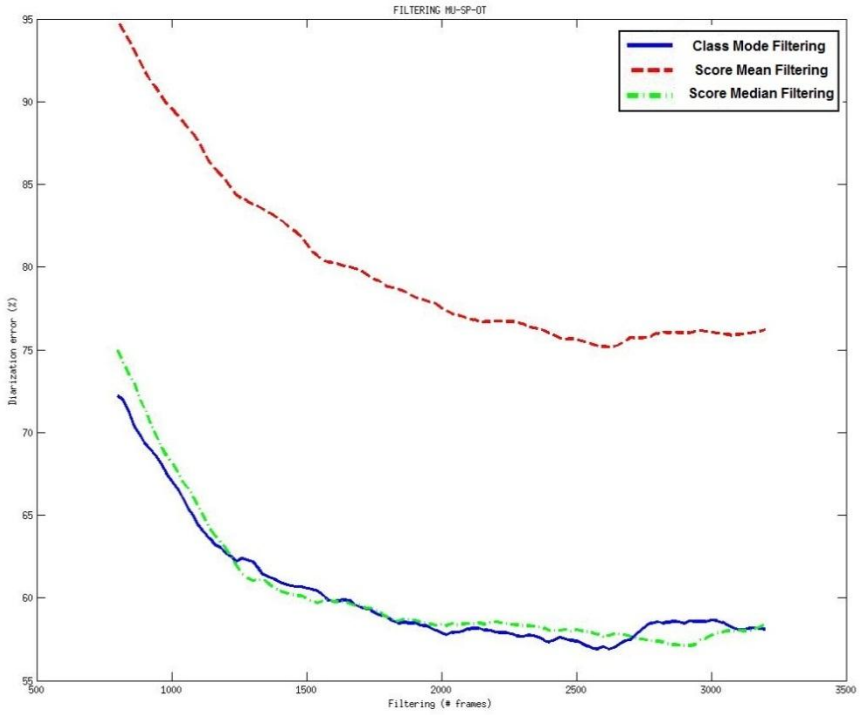


Figura 4-16: Análisis del filtrado de scores para el sistema MU-SP-OT

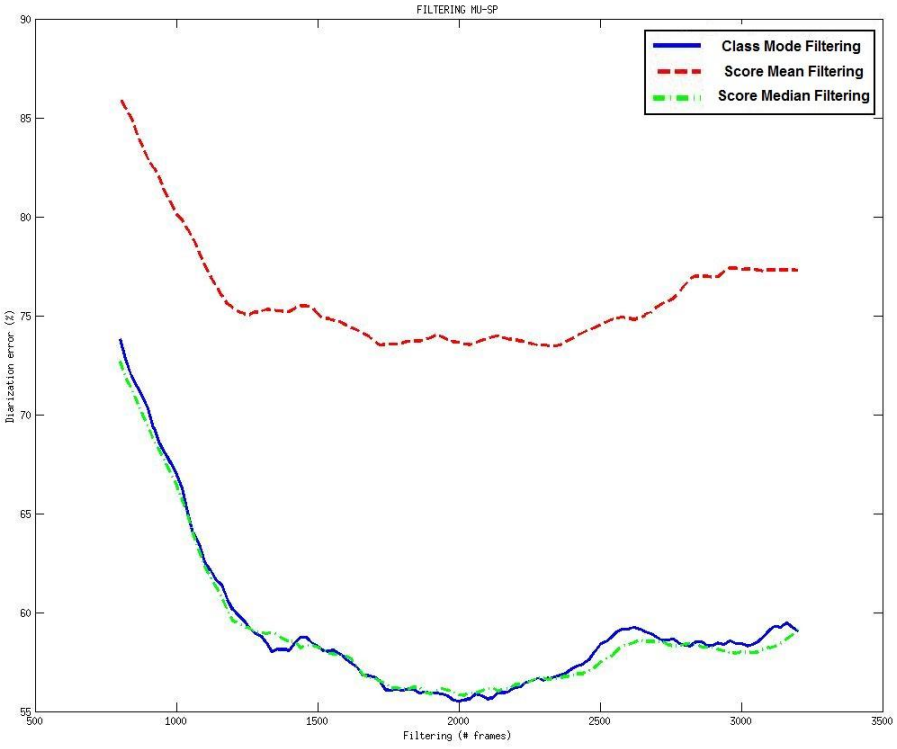


Figura 4-17: Análisis del filtrado de scores para el sistema MU-SP

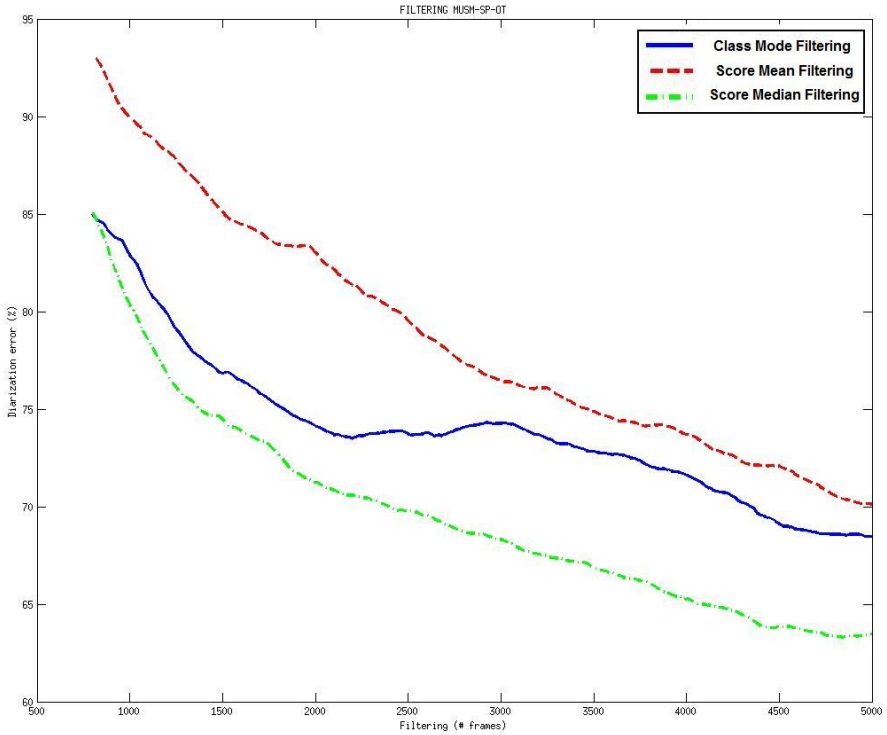


Figura 4-18: Análisis del filtrado de scores para el sistema MUSM-SP-OT

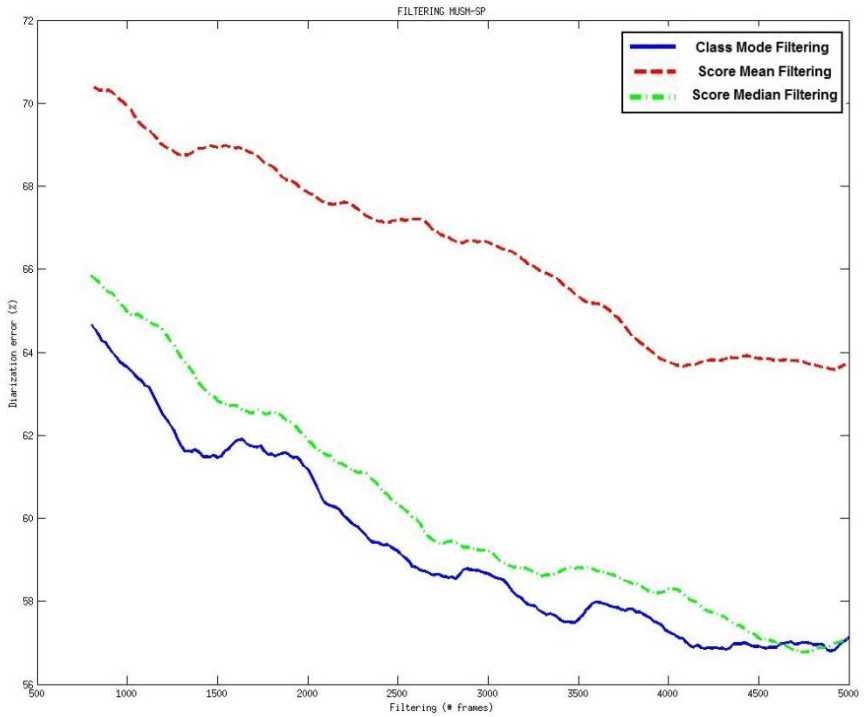


Figura 4-19: Análisis del filtrado de scores para el sistema MUSM-SP

La tabla que resume todos estos datos de manera conjunta es:

Tabla 1: Parámetros resultantes de la etapa de desarrollo (development)

Sistema	Tipo de filtrado	Tamaño del filtro
MU-ALL	Por moda	16,8 segundos
MUSM-ALL	Por mediana	50 segundos
SP-NSP	Por moda	14,4 segundos
MU-SP-OT	Por moda	26,2 segundos
MU-SP	Por moda	20 segundos
MUSM-SP-OT	Por mediana	48,4 segundos
MUSM-SP	Por mediana	47,4 segundos

El primer concepto clave que se puede extraer de estas gráficas es que en todos los sistemas (salvo en **MUSM-ALL**) el filtrado mediante medias es el que da peores resultados con diferencia, por lo que este tipo de post-procesado de puntuaciones queda descartado para cualquier sistema. El sistema **MUSM-ALL** presenta los mismos resultados independientemente del tipo de filtrado escogido, pues no hay un método claramente mejor al resto. En este caso, y en todos los demás que presenten más de un tipo de filtrado con resultados parecidos, se ha optado por el filtro que presenta menor error de diarización o segmentación, aunque también se podría haber optado por el método con menor carga computacional (apartado 4.3.).

En cuanto al resto de sistemas se pueden agrupar en dos grandes grupos: los que presentan mejores resultados utilizando un filtrado por modas a partir de las puntuaciones segmentadas (grupo A) y los que consiguen mejor rendimiento con un filtrado por medianas de las puntuaciones (grupo B). Así pues, en el primer grupo, o grupo A, se encuentran los sistemas **MU-ALL**, **MU-SP-OT**, **MU-SP** y **SP-NSP**, mientras que en el otro grupo, o grupo B, quedan los sistemas **MUSM-ALL**, **MUSM-SP-OT** y **MUSM-SP**.

Otra de las diferencias notables entre estos dos grupos de sistemas es el tamaño del filtro utilizado, pues mientras que los sistemas del grupo A tienen tamaños de ventana entre 15 y 25 segundos, los sistemas del grupo B se acercan al minuto (45-50 segundos). El uso de filtros de tamaños tan grandes viene motivado por las etiquetas de Ground-Truth de la evaluación ALBAYZIN, pues se tratan de etiquetas con un bajo nivel de finura que obvian los fragmentos de menor tamaño, motivando el uso de filtros largos.

4.2 Comparativa con el sistema de segmentación de audio ATVS para la evaluación ALBAYZIN 2010

Durante la etapa de desarrollo se ha realizado también un estudio sobre si la fusión propuesta mejora el rendimiento del sistema implementado y del sistema ATVS para ALBAYZIN 2010 por separado con el objetivo de comprobar la viabilidad de la combinación de información en un trabajo futuro. En este caso se han realizado dos tipos de fusión a nivel de etiqueta (basada en decisiones) mediante operaciones lógicas: AND y OR (apartado 3.5.). En la siguiente tabla se pueden apreciar los resultados obtenidos, correspondiéndose con el error de segmentación (sección 2.3.2.1.):

Tabla 2: Errores de segmentación para los sistemas ATVS para ALBAYZIN 2010, para el sistema implementado y para la fusión de ambos sistemas (AND y OR) durante la etapa de desarrollo (development)

	MFCC-2010	CromaEnt	AND	OR
MU-ALL	**10,32 %**	42,32 %	12,45 %	41,19 %
MUSM-ALL	**10,38 %**	55,61 %	37,70 %	27,59 %
SP-NSP	17,26 %	39,52 %	40,74 %	**15,65 %**
MU-SP-OT	**27,47 %**	56,90 %	41,48 %	48,75 %
MU-SP	**11,37 %**	55,51 %	41,81 %	105,11 %
MUSM-SP-OT	**27,33 %**	63,31 %	58,97 %	32,19 %
MUSM-SP	**11,16 %**	56,77 %	61,20 %	74,37 %

donde “MFCC-2010” hace referencia al sistema presentado por el grupo ATVS en la evaluación ALBAYZIN 2010 de segmentación de audio, “CromaEnt” al sistema implementado y “AND” y “OR” a las distintas fusiones realizadas.

La única fusión que mejora expresamente a ambos sistemas por separado es la de hacer un OR sobre la clase *sp* en el sistema **SP-NSP**, teniendo una mejora sobre el sistema ATVS para ALBAYZIN 2010 del 9,33 % relativo (de 17,26 % a 15,65 %).

El objetivo de esta fusión es el de ayudar al sistema ATVS de ALBAYZIN 2010 en los fragmentos que etiquete erróneamente mediante el sistema presentado en este PFC. Se puede ver un claro ejemplo en la siguiente figura, donde el sistema ATVS de ALBAYZIN 2010 identifica mal un fragmento de audio de la sesión 14 de la base de datos para el sistema **MU-ALL**, mientras que el algoritmo implementado sí lo etiqueta correctamente.

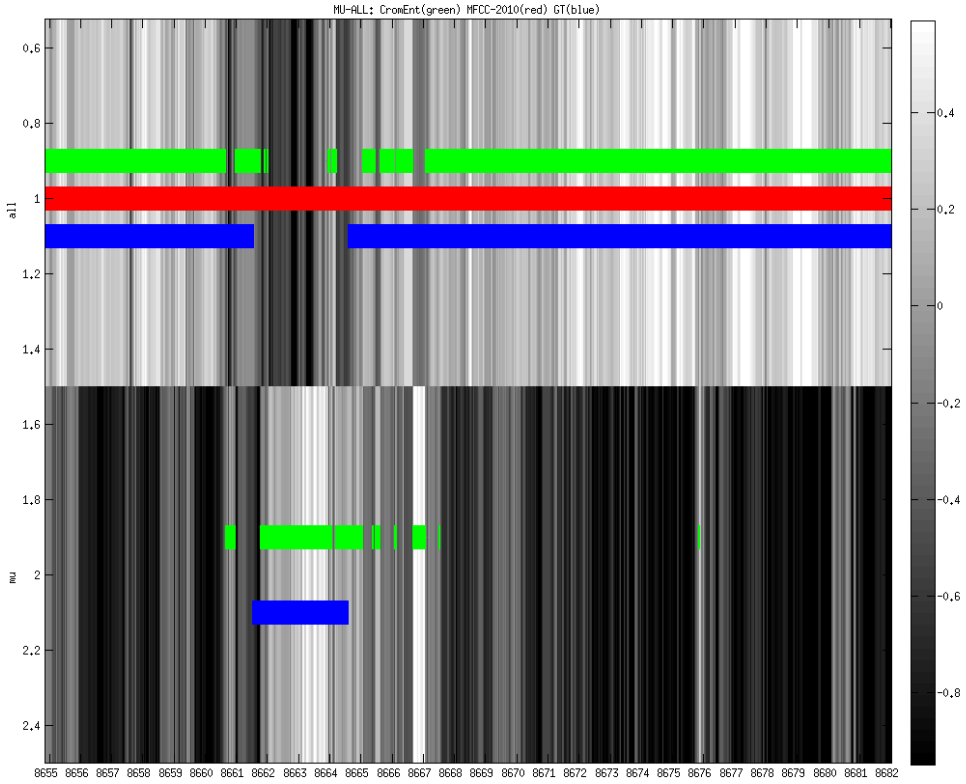


Figura 4-20: Comparativa de etiquetado entre sistemas CromaEnt y MFCC-2010

donde la línea verde hace referencia al sistema presentado en este PFC (CromaEnt), la línea roja al sistema ATVS de ALBAYZIN 2010 (MFCC-2010) y la línea azul identifica las etiquetas de Ground-Truth. Estas etiquetas están impresas sobre la matriz de puntuaciones obtenida por el sistema CromaEnt en escala de grises, donde una puntuación mayor tomará una tonalidad más blanca. La banda superior representa las puntuaciones de la clase *all*, mientras que la inferior representa las de la clase *mu*.

En el caso anterior, el sistema MFCC-2010 identifica todo el fragmento como perteneciente a la clase acústica *all*, mientras que el sistema CromaEnt etiqueta de manera correcta la porción de audio que según las etiquetas de Ground-Truth se trata de música.

Teniendo en cuenta los estudios llevados a cabo durante toda la etapa de desarrollo (tipo de filtrado de scores y fusión con el sistema ATVS para ALBAYZIN 2010) se han extraído los rendimientos finales haciendo uso de las sesiones 17 a 24 de la base de datos (correspondientes a la etapa de evaluación). De tal forma, los parámetros utilizados en cada uno de los sistemas implementados son:

Tabla 3: Recapitulación de parámetros comunes a todos los sistemas implementados

Preénfasis	Tamaño de enventanado	Solapamiento en el enventanado	Tipo de enventanado
0,97	20 ms	50 %	Hamming
Nº puntos en la FFT	Ventana para extracción de características	Nº de componentes GMM	Factor de relevancia (MAP)
$2^{13} = 8192$	1 segundo	128	8

Tabla 4: Parámetros utilizados en cada uno de los sistemas implementados

Sistema	Tipo de filtrado	Tamaño del filtro	Fusión con MFCC-2010	Rendimiento
MU-ALL	Moda	16,8 segundos	-	**42,33 %**
MUSM-ALL	Mediana	50 segundos	-	**52,80 %**
SP-NSP	Moda	14,4 segundos	OR	**14,91 %**
MU-SP-OT	Moda	26,2 segundos	-	**63,90 %**
MU-SP	Moda	20 segundos	-	**60,02 %**
MUSM-SP-OT	Mediana	48,4 segundos	-	**68,57 %**
MUSM-SP	Mediana	47,4 segundos	-	**57,44 %**

Puesto que únicamente la fusión del sistema **SP-NSP** con el presentado por el grupo ATVS en ALBAYZIN 2010 mejora ambos sistemas, se ha utilizado la fusión propuesta exclusivamente para este caso, llegando a conseguir una mejora relativa del 5 % (de 15,70 % a 14,91 %). En el resto de sistemas se ha optado por utilizar exclusivamente la implementación desarrollada. Comparando estos resultados con los obtenidos por el grupo ATVS para ALBAYZIN 2010 se comprueba un peor rendimiento, debido a la simplicidad de las características utilizadas en este proyecto frente a la complejidad y sofisticación del presentado por ATVS en ALBAYZIN 2010 (apartado 3.5.).

Tabla 5: Errores de segmentación para los sistemas ATVS para ALBAYZIN 2010, para el sistema implementado y para la fusión de ambos sistemas (AND y OR) durante la etapa de evaluación (testing)

	ATVS	SYSTEM	AND	OR
MU-ALL	**16,43 %**	42,33 %	19,73 %	39,76 %
MUSM-ALL	**17,14 %**	52,80 %	36,14 %	33,44 %
SP-NSP	15,70 %	41,29 %	43,15 %	**14,91 %**
MU-SP-OT	**29,25 %**	63,90 %	55,46 %	55,75 %
MU-SP	**17,10 %**	60,02 %	42,52 %	110,34 %
MUSM-SP-OT	**29,77 %**	69,57 %	64,35 %	45,98 %
MUSM-SP	**17,88 %**	57,44 %	59,86 %	78,44 %

4.3 Tiempos de ejecución

En este apartado se estudia la carga computacional de los sistemas desarrollados y se compara con el tiempo empleado por el sistema ATVS para ALBAYZIN 2010.

El proyecto desarrollado se ha llevado sobre el rack perteneciente al grupo ATVS de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid, disponiendo de un servidor con 24 procesadores y con una memoria RAM de 16 GB.

La extracción de características en el sistema ATVS para ALBAYZIN 2010 para una sola sesión (~ 4 horas de audio) supone una dedicación exclusiva de un procesador de 14 minutos, mientras que la decodificación de Viterbi junto con el filtrado de modas aplicado a la salida consume un total de 20 horas en un solo procesador, también por cada sesión de la base de datos (~ 4 horas de audio).

El sistema presentado en este proyecto, aun usando distintas características, también tiene una carga computacional de 14 minutos para la extracción de características por cada sesión de la base de datos. Aquí se incluye el cálculo de la entropía cromática a partir del audio y el cómputo de las cuatro características a partir de dicha entropía.

El entrenamiento y modelado de los distintos sistemas conlleva una carga de 41 minutos en un procesador a pleno rendimiento. Puesto que el entrenamiento hace uso de las doce primeras sesiones de la base de datos, se estima en algo más de 3 minutos la carga computacional por cada sesión de la base de datos.

La etapa de evaluación se compone de 3 posibles procesos, dependiendo del tipo de filtrado escogido. Así pues se necesitan 11 minutos para segmentar el audio de una sesión si se utiliza un filtrado por modas, 8 minutos para el filtrado por medias y 18 minutos si se emplea el filtrado por medianas. La fusión de una sesión con su respectiva del sistema ATVS para ALBAYZIN 2010 supone una carga computacional de 40 segundos.

De manera resumida, el sistema implementado necesita un procesador a pleno rendimiento entre 26 y 36 minutos desde que se analiza una sesión de entrenamiento hasta que la correspondiente sesión de evaluación es correctamente segmentada. El cómputo global del sistema (teniendo en cuenta todas las sesiones utilizadas) es de entre 4 horas y media hasta casi 6 horas, dependiendo del filtrado elegido.

Se concluye una mejora considerable frente a la carga computacional que implica el sistema ATVS para ALBAYZIN 2010.

Tabla 6: Comparación de la carga computacional del sistema frente al presentado por ATVS en ALBAYZIN 2010 para una sesión de test (~ 4 horas)

	MFCC-2010	CromaEnt
Extracción de características	14 minutos	14 minutos
Decodificación Viterbi + filtrado de modas	20 horas	-
Segmentación de audio + filtrado	-	18 minutos
TOTAL	~20 horas	~30 minutos

5 Otras contribuciones realizadas

En este capítulo se exponen otros estudios realizados en el ámbito de los sistemas de Recuperación de Información Musical (Music Information Retrieval, MIR). Estos estudios han sido de gran utilidad para poder tener una visión global de los sistemas de tratamiento de audio y poder desarrollar el sistema implementado de segmentación de audio más eficientemente.

Se han estudiado esencialmente dos tipos de tareas de MIR: la similitud de audio musical y la identificación de versiones musicales. Ésta última ha servido como base para la generación de material empleado en las prácticas de la asignatura Tecnologías de Audio, de 4º curso del Grado de Ingeniería de Tecnologías y Servicios de Telecomunicación de la Universidad Autónoma de Madrid (Anexo D).

5.1 Similitud de audio musical

La primera toma de contacto con los sistemas de tratamiento de audio musical se realizó con la tarea de similitud de audio musical definida por la MIREX (ver apartado 2.4.1.1.). En esta sección se propone un sistema de similitud musical, explicando su diseño. También se hace un breve estudio sobre las bases de datos más utilizadas en este campo.

5.1.1 Sistema propuesto

Uno de los objetivos más ambiciosos de los sistemas MIR es el poder generar de manera automática una medida de similitud entre dos grabaciones musicales basándose exclusivamente en análisis del propio audio. De esta manera se propone el desarrollo de un sistema capaz de medir el nivel de similitud musical entre dos piezas de música.

En la literatura de la similitud de audio musical se usan gran cantidad de técnicas para alcanzar esa medida buscada, la mayoría derivadas de los sistemas de tratamiento de señales de audio, como son los MFCCs, los cromagramas, las redes neuronales, las entropías espectrales, etc. El sistema propuesto no hace uso de técnicas de clasificación de audio, como los GMM o el UBM, puesto que no se pueden hacer conjuntos de canciones que formen una misma clase acústica, sino que cada pieza musical tiene un nivel de similitud distinto con respecto a todas las demás canciones. Como contraparte existen otras muchas tareas MIR que sí pueden hacer uso de las principales técnicas de agrupamiento y modelado, como reconocedores de género musical o identificación de cantante, ya que sí se pueden formar clases acústicas altamente definidas, como por ejemplo todas las canciones pertenecientes al género musical de rock. No obstante se

pueden emplear modelos de gaussianas en la comparación de dos piezas de música, evaluando las características de cada audio mediante sus representaciones gaussianas.

De tal forma se propone un sistema comparador de audios musicales 1-by-1, en el que se analizarán dos audios y se presentará una medida de similitud entre ambos.

Un modelo alternativo al sistema propuesto es aquel que teniendo un audio musical a evaluar es necesario definir con qué canción de un grupo de test tiene mayor nivel de similitud. En este caso sí son utilizables técnicas como los GMM-UBM a partir de los MFCC.

5.1.2 Bases de datos

Para la implementación del sistema de similitud de audio musical se han utilizado dos subconjuntos de la base de datos The Million Song Dataset (MSD) [Bertin-Mahieux et al., 2011]: la USPOP2002 y la Million Song Subset.

La MSD está compuesta por casi un millón de canciones extraídas de la API de The Echo Nest[5] (plataforma proveedora de servicios musicales para desarrolladores y grandes compañías), incluyendo tanto metadatos como características extraídas del propio audio. Cada archivo se corresponde con una canción interpretada por un determinado artista (por si hubiera distintas versiones de una misma canción). La MSD reúne todos los datos de cada canción en archivos HDF5. Este tipo de formato comprime la información y la organiza de manera jerárquica. Estos archivos HDF5 no contienen el audio en sí, sino que directamente proporcionan distintas características extraídas de él: coeficientes MFCC, características cromáticas e intensidad máxima. Todas estas características se han extraído mediante la definición de onsets (ver apartado 2.2.3.1.) en lugar de enventanados de tamaño fijo, también disponibles en la MSD.

La USPOP2002 consta de 8752 canciones de 400 artistas diferentes convertidas del formato MP3 (monofónicas, codificadas a 128 Kbps y muestreadas a 22050 Hz) al HDF5 de la MSD, mientras que la Million Song Subset se compone de 10000 canciones extraídas directamente de la MSD.

Puesto que esta base de datos no proporciona etiquetas de Ground-Truth que definan qué canciones se parecen a otras, es necesario el uso de una base de datos auxiliar: la Last.fm Dataset. Esta base de datos sigue el mismo esquema que la MSD, pero sólo proporciona dos tipos de información de cada canción: el género musical y un listado con sus canciones más similares (disponiendo de la medida de similitud musical). Estas etiquetas han sido creadas a partir de la API de Last.fm.

[5] http://the.echonest.com/

5.1.3 Diseño

El sistema propuesto se divide en dos grandes tareas: etapa de entrenamiento y de etapa de evaluación. Durante el entrenamiento se hace uso de las bases de datos y de las etiquetas de Ground-Truth para entrenar el sistema y los algoritmos de comparación de características para acercarse lo más posible a dichas etiquetas. Es en la etapa de evaluación donde se comprueba la bondad del sistema implementado.

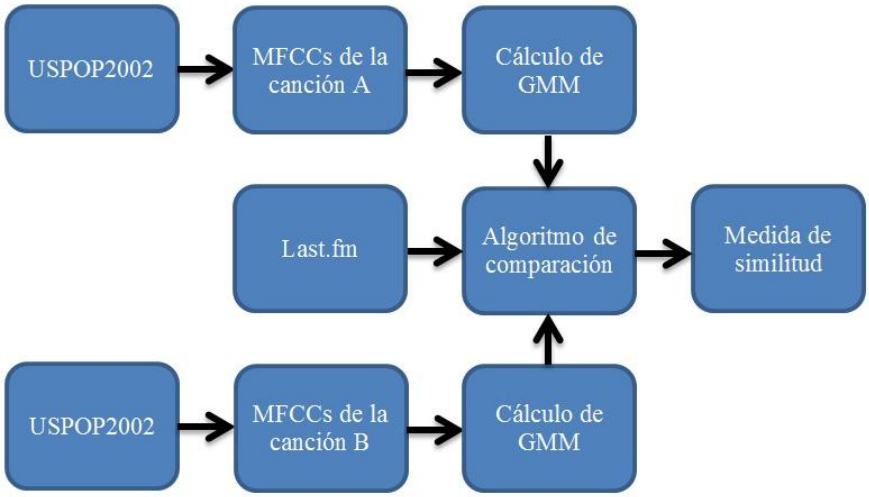


Figura 5-1: Diagrama de flujo del sistema de similitud de audio musical propuesto

El algoritmo de comparación propuesto se basa en cuán parecida es una distribución gaussiana a otra, siendo ese nivel de semejanza la medida de similitud de salida que dará el sistema.

El modelo alternativo propuesto en el apartado 5.1.1. (aquel que teniendo un audio musical a evaluar es necesario definir con qué canción de un grupo de test tiene mayor nivel de similitud) emplea una gran cantidad de archivos musicales para formar un UBM lo más "amplio" posible, es decir, un UBM donde se recojan la mayoría de las características presentes en un audio musical genérico. El modo de proceder sería el de extraer las características del audio (en este caso la base de datos ya proporciona esta información) y se enfrentaría siguiendo el algoritmo LR (ver apartado 2.2.2.3.1.) a los modelos de las distintas canciones objetivo, o target, normalizadas por el UBM previamente calculado (Figura 2-10).

5.1.4 Problemática de la tarea

Si bien esta tarea tiene un gran valor tanto académico como comercial, su complejidad es máxima debido a varios factores. El primer problema que presenta este sistema es el uso exclusivo de características tímbricas (MFCC), cuando se ha demostrado que

cualquier audio en general, y aquellos con componentes musicales en especial, proporcionan multitud de información altamente valiosa no registrada mediante este tipo de características [Aucouturier & Pachet, 2004]. La solución más plausible es la incorporación de nuevas características que suplan la carencia de los MFCC, como las características cromáticas, disponibles en las bases de datos utilizadas.

Sin embargo, los otros dos problemas que presentan este tipo de sistemas en general suponen un gran inconveniente. Esto es la problemática de disponer de audio musical libremente y la difícil adquisición de etiquetas de Ground-Truth [Ellis et al., 2002].

Como ya se adelantó en la sección 2.4.1.1., el contenido musical supone una pieza clave en nuestra sociedad de consumo, habiendo grandes intereses comerciales detrás de cada artista. Es por esa razón, entre otras, que las bases de datos públicas no distribuyen el audio original, pues estarían incurriendo en delitos de derechos de autor. De esta manera sólo pueden proporcionar los análisis llevados a cabo sobre el propio audio. Sin embargo, este tipo de análisis suelen hacerse de manera opaca al usuario mediante APIs o algoritmos privativos, perdiendo el control de lo que se está haciendo.

La otra gran desventaja de este tipo de sistemas es inherente a la propia tarea, pues la definición de similitud entre dos audios musicales no deja de ser un concepto subjetivo y sujeto a la percepción humana. Por ejemplo: una persona puede encontrar dos canciones muy parecidas mientras que otra persona no vea similitud alguna. Las bases de datos que proporcionan este tipo de información pueden complicar la evaluación del rendimiento del sistema implementado. Por ejemplo, si usamos las etiquetas de similitud de Last.fm, no sabremos qué criterios se han seguido para las mismas, con lo que nuestro sistema de referencia serán dichas etiquetas. Al no haber un criterio objetivo, la evaluación del rendimiento del sistema queda comprometida.

5.2 Identificación de versiones musicales

Salvando el inconveniente de las etiquetas de Ground-Truth, en este capítulo se propone un sistema de identificación de versiones musicales. También se explican algunas de las bases de datos más utilizadas en esta tarea, así como un breve resumen del material generado para las prácticas de la asignatura Tecnologías de Audio.

5.2.1 Sistema propuesto

Tal y como se adelantó en el capítulo 2.4.1.2., los sistemas de identificación de versiones musicales son aquellos capaces de definir si dos piezas musicales son versiones de una misma canción, interpretadas generalmente por dos artistas diferentes.

Se ha seguido la metodología propuesta por Daniel Ellis [Ellis & Poliner, 2007] como base para la implementación de un sistema identificador de versiones musicales. El sistema propuesto extrae y analiza las componentes cromáticas de dos audios musicales y evalúa su nivel de semejanza con el fin de determinar sin son versiones de una misma canción o no.

5.2.2 Bases de datos

Para el desarrollo del sistema de identificación de versiones musicales se han utilizado mayoritariamente dos bases de datos: la COVERS80 y la Second Hand Songs (SHS).

La COVERS80 se compone de 80 canciones interpretadas cada una por dos artistas diferentes, teniendo un total de 160 canciones. Fue creada a partir de la USPOP2002 por el Laboratory for the Recognition and Organization of Speech and Audio (LabROSA[6]) de la Universidad de Columbia. Cada archivo de audio está convertido a formato MP3 a partir de WAV a una tasa de 32 Kbps, siendo audio monofónico con 16 KHz de tasa de muestreo y un ancho de banda limitado a 7 KHz. Es por tanto una base de datos con audio de calidad limitada, lo que permite su distribución pública.

La SHS funciona de igual modo que la Last.fm, ya que únicamente relaciona canciones de la MSD entre sí, no disponiendo del audio musical. Consta de 18196 canciones, agrupadas en 5854 sets de versiones, entendiendo como set aquel conjunto de canciones que son versiones de una misma pieza musical. La SHS llega a tener sets de únicamente dos artistas hasta sets de más de quince. La SHS viene dividida en dos sub-bases de datos: una para entrenamiento (de 12960 canciones en 4128 sets de versiones) y otra para evaluación (de 5236 canciones en 1726 sets de versiones).

5.2.3 Diseño

El proceso de identificación de versiones musicales comienza con la extracción de los cromagramas (apartado 2.2.3.1.) a partir del audio. Estos cromagramas se consiguen delimitando el audio mediante fronteras notables u onsets, midiendo el tempo o la potencia, entre otras medidas. A partir de estos onsets se calcula la energía para cada una de las 12 notas musicales de la escala cromática, teniendo así el cromagrama deseado.

Una vez extraídos los cromagramas se procede a la comparación 1-by-1 de los distintos audios a evaluar. Esta evaluación se consigue mediante el cálculo de la correlación cruzada de los dos cromagramas bajo estudio (sección 2.4.1.2.), haciendo rotar uno de ellos sobre su eje tonal para suplir las posibles diferencias causadas por el

[6] http://labrosa.ee.columbia.edu/

uso de distintas notas tónicas. Esta correlación cruzada también incluye un pequeño desplazamiento temporal que corrige diferencias en el tempo de dos canciones.

La medida final que decidirá, tras la determinación de un cierto umbral (θ), si dos canciones son versiones de una misma pieza musical es el valor máximo de todas las correlaciones cruzadas filtradas paso alto (Figura 2-17).

Para la base de datos Second Hand Songs, al no disponer del audio original, es necesario operar directamente con las características cromáticas calculadas mediante la API de The Echo Nest. El uso de esta base de datos pierde mucho valor debido a la incapacidad de crear un sistema completo de principio a fin, en el que se tenga audio a la entrada del sistema y un valor que relacione el nivel de semejanza entre dos canciones en cuanto a versiones musicales se refiere. No pasa lo mismo con la base de datos COVERS80, que aunque ofrece una cantidad de datos muy restringida, sí se dispone del audio original.

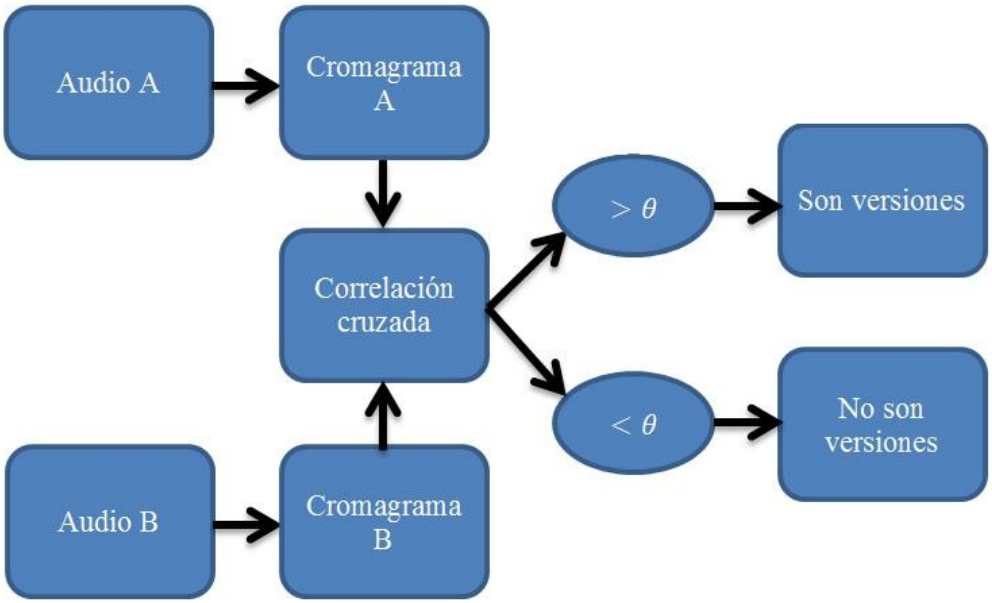


Figura 5-2: Diagrama de flujo del sistema de identificación de versiones musicales propuesto

5.2.4 Material generado para las prácticas de "Tecnologías de Audio"

Uno de los primeros objetivos del proyecto fue el de generar material para las prácticas de la asignatura Tecnologías de Audio, de 4º curso del Grado de Ingeniería de Tecnologías y Servicios de Telecomunicación de la Universidad Autónoma de Madrid.

La práctica para la que se generó todo el material en adelante descrito tenía por objetivo el estudio de diferentes sistemas que tratasen audio musical utilizando Matlab™. Se utilizó para tal fin los algoritmos propuestos por Daniel Ellis [Ellis & Poliner, 2007] en

el tratamiento e identificación de versiones musicales. Para la realización de esta práctica, se utilizó software desarrollado por LabROSA para la identificación de versiones. Dicho software está implementado en Matlab™, estando disponible[7] bajo licencia Gnu GPL. La base de datos a utilizar, denominada COVERS80, es también de dominio público.

Junto a la base de datos se proporcionan 2 listados con todas las canciones que contienen la base de datos (lista "A" y lista "B"). Esto servirá para poder comparar las 80 canciones con sus respectivas versiones.

Debido al coste computacional del sistema de Ellis [Ellis & Poliner, 2007] se confeccionó una sub-base de datos a partir de la COVERS80, con únicamente 10 canciones interpretadas por dos artistas diferentes, para agilizar la práctica.

El sistema utilizado en la práctica parte del audio disponible en la base de datos "COVERS10", extrayendo los cromagramas de cada una de las piezas musicales.

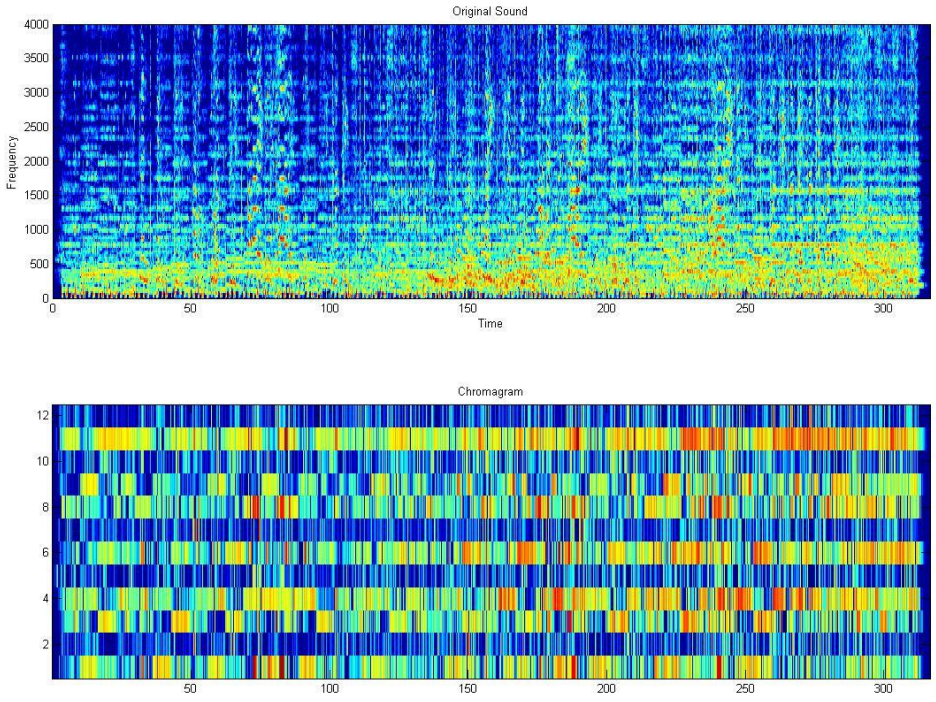


Figura 5-3: Cromagrama extraído a partir del espectrograma de un audio musical [Práctica de la asignatura Tecnologías de Audio, 2013]

Una vez calculados los cromagramas de todas las canciones, se enfrentan todas las de la lista A contra todas las de la lista B tal y como se hacía en el sistema propuesto en el apartado 5.2.3.

[7] http://labrosa.ee.columbia.edu/projects/coversongs/

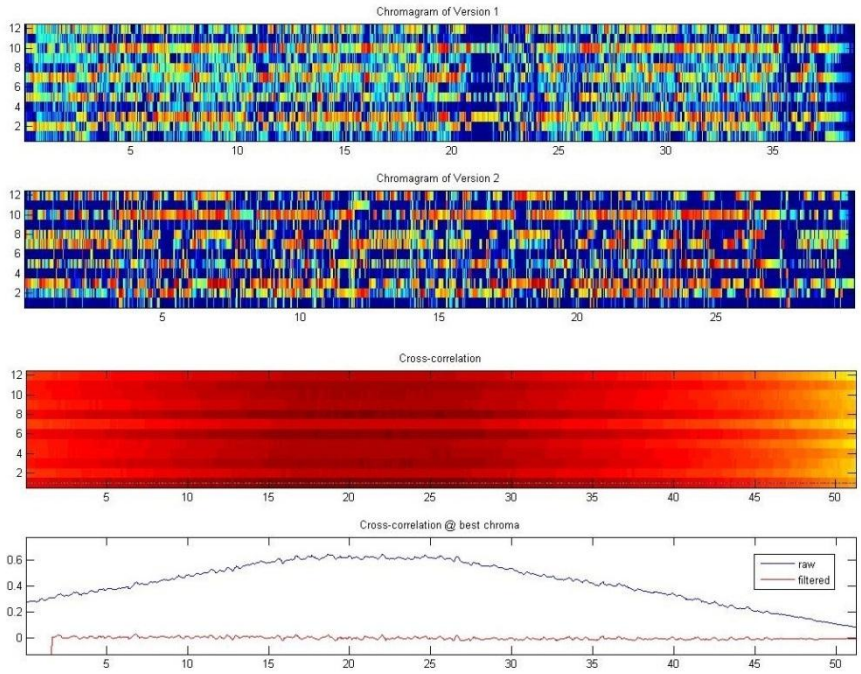


Figura 5-4: Cálculo de la relación entre dos cromagramas para determinar si son versiones de una misma composición [Práctica de la asignatura Tecnologías de Audio, 2013]

El rendimiento de este sistema sigue el mismo patrón que los sistemas propuestos por D. Ellis. Esto es, acordar que la canción i de la lista A es versión musical de aquella de la lista B que mayor puntuación haya dado. De esta forma, enfrentando las canciones de ambas listas, y marcando las puntuaciones más altas, se observa de manera gráfica la bondad del sistema mediante una matriz de confusión:

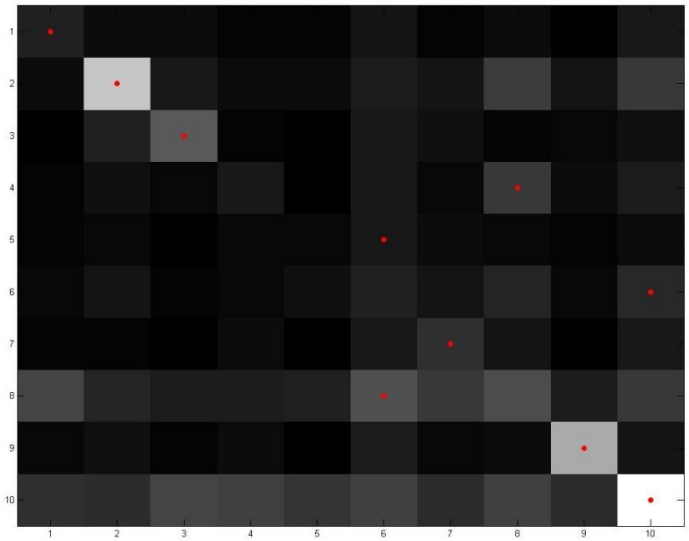


Figura 5-5: Matriz de confusión para la COVERS10 [Práctica de la asignatura Tecnologías de Audio, 2013]

donde el eje de abscisas representa las 10 canciones de la lista B, ordenadas de la 1 a la 10, mientras que el eje de ordenadas hace referencia a las canciones de la lista A. Se toma como versión de una determinada canción de la lista A aquella canción de la lista B que de mayor nivel de similitud, tomando así en la matriz de confusión los máximos horizontales.

Las matrices de confusión son una herramienta de visualización muy útil para comprobar si un determinado sistema de clasificación confunde dos clases. En general, cada columna de la matriz representa el número de predicciones de cada clase (número total de elementos clasificados como pertenecientes a una determinada clase), mientras que cada fila representa a las instancias en la clase real (número total de elementos pertenecientes a una determinada clase). En el sistema propuesto, las clases son las versiones a identificar, y los elementos son las canciones versionadas de esa composición genérica, es decir, por cada clase (versión) se tienen dos elementos (canciones).

La tasa de acierto será el número de versiones identificadas correctamente frente al número total de canciones disponibles en la base de datos. En el ejemplo de la Figura 5-5, la tasa de acierto es del 60 % (6 aciertos de 10 posibles). Probando el mismo algoritmo con la COVERS80 se obtiene un rendimiento del 37,5 % (teniendo 30 aciertos de 80 posibles).

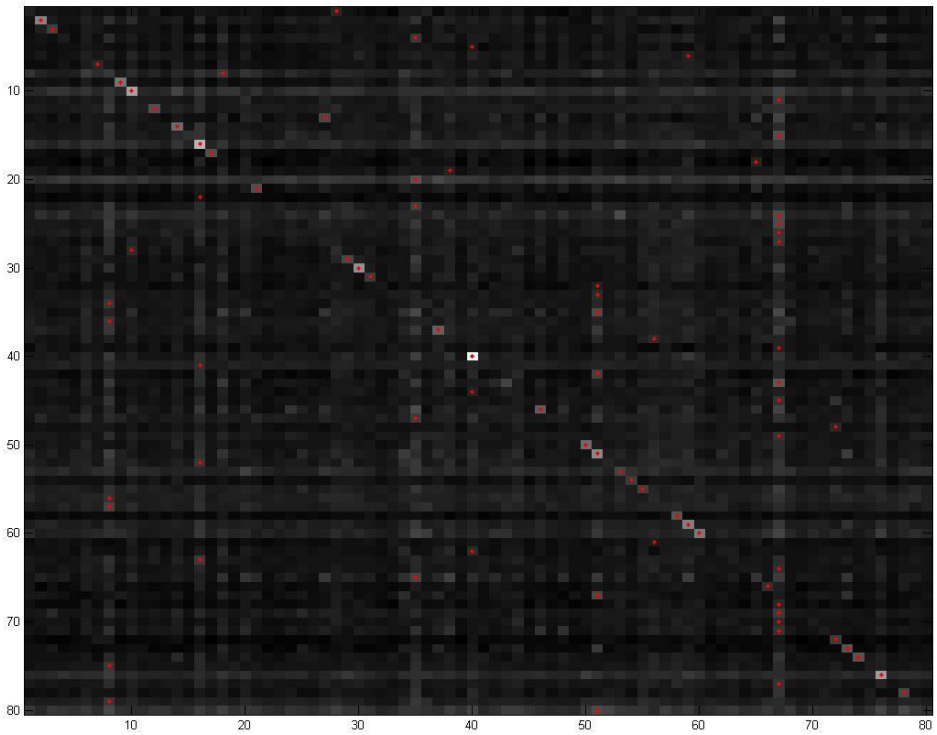


Figura 5-6: Matriz de confusión para la COVERS80 [Práctica de la asignatura Tecnologías de Audio, 2013]

Como apartado de ampliación, se propuso la mejora del sistema mediante una denormalización [Ellis & Cotton, 2007]. La normalización aseguraba que los resultados estuviesen comprendidos entre 0 y 1, pero a su vez se introducía un escalado variable en los valores de similitud de versión en determinadas canciones, lo que impedía determinar qué canción era la versión de otra canción.

Realizados los cambios oportunos en el código, obtenemos la siguiente matriz de confusión:

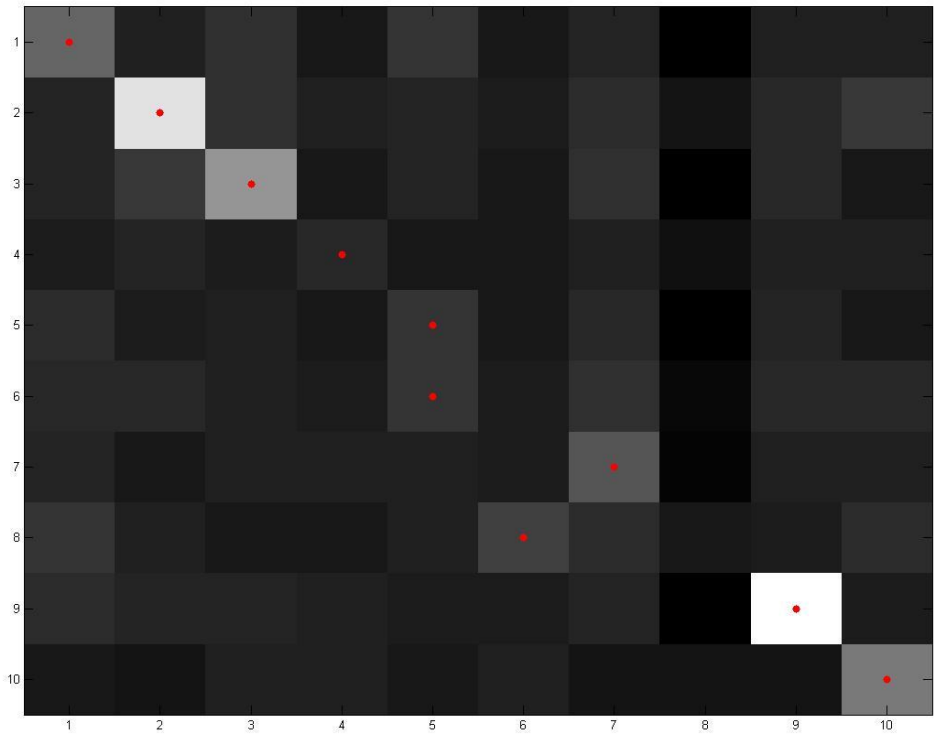


Figura 5-7: Matriz de confusión para la COVERS10 haciendo uso de la denormalización [Práctica de la asignatura Tecnologías de Audio, 2013]

En este caso la tasa de acierto se eleva hasta el 80%, dejando más que patente la mejora aplicada.

Aplicando esta mejora a la COVERS80 se incrementa el rendimiento en un 13,33 % relativo, pasando de una tasa de acierto del 37,5 % a una del 42,5 % (teniendo 34 aciertos de 80 posibles).

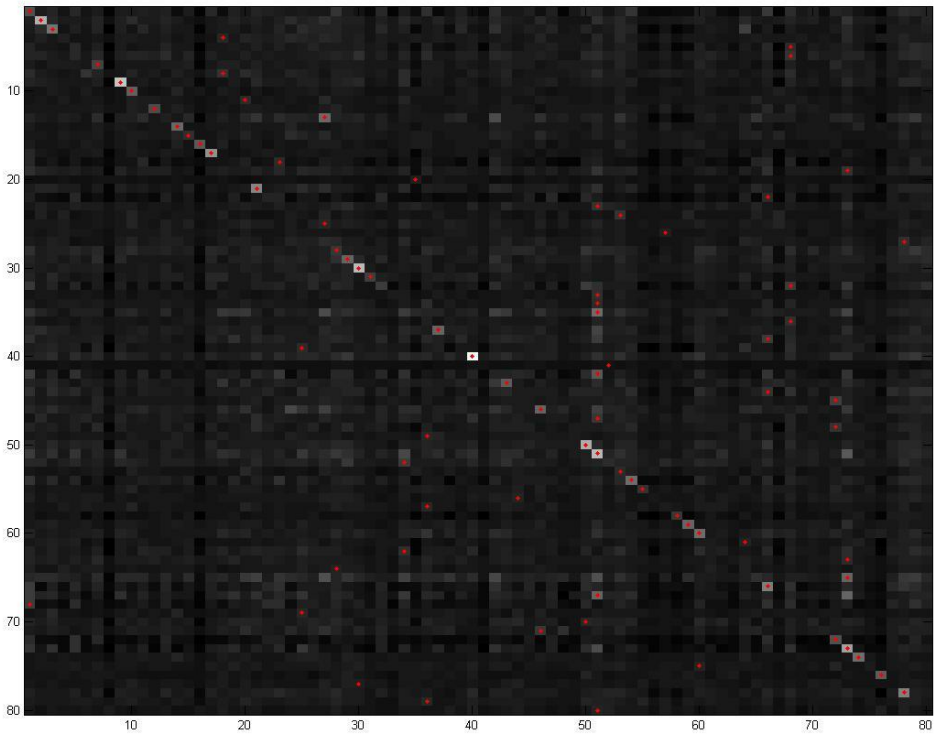


Figura 5-8: Matriz de confusión para la COVERS80 Haciendo uso de la denormalización [Práctica de la asignatura Tecnologías de Audio, 2013]

6 Conclusiones y Trabajo futuro

Con este capítulo se concluye el proyecto fin de carrera presentado, haciendo una breve valoración final del trabajo realizado, así como un estudio de las conclusiones a las que se ha llegado. También se detallan todas las herramientas que se han implementado y que quedan para el uso del grupo de investigación ATVS de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid.

Este capítulo concluye con una breve perspectiva de futuro sobre las posibles ampliaciones que aceptan los sistemas desarrollados.

6.1 Conclusiones

A la vista de los resultados obtenidos se pueden extraer varios conceptos claves de la segmentación de audio, tales como que esta tarea es una de amplia aplicación en el campo del tratamiento de señales de audio y Recuperación de Información, y que presenta una línea de investigación muy prometedora de aquí a varios años para el grupo ATVS.

En cuanto al sistema desarrollado se constata un rendimiento inferior al alcanzado por el sistema ATVS para ALBAYZIN 2010. Las principales razones de esta diferencia radican en la complejidad de las características, así como en un buen ajuste de la base de datos de entrenamiento y test, que permite un buen rendimiento de los MFCC en esta tarea. No obstante, mediante el estudio de "búsqueda exhaustiva" realizado, se ha demostrado el alto poder de discriminación de las características utilizadas, extraídas a partir de la entropía cromática. De hecho, aunque el sistema implementado presente peores resultados, realizando una fusión con el sistema ATVS para ALBAYZIN 2010 se consigue mejorar el rendimiento global del detector de habla (sistema **SP-NSP**), dejando constancia de que la aproximación es complementaria en algunos casos, incluso en esquemas de fusión muy simplistas. Esto motiva el seguir investigando en este campo.

Uno de los estudios que se querían llevar a cabo mediante la implementación de los siete sub-sistemas fue el de evaluar la relevancia de la clase acústica other (*ot*). Partiendo de los sistemas base, se ha jugado con esta clase estudiándola por separado, acoplándola a otras clases o directamente omitiéndola del estudio, para terminar concluyendo que presenta un comportamiento similar al característico del habla y diferenciándose del audio musical. La clase *mu* tiene una organización armónica clara, que se refleja en unos valores muy organizados de los estadísticos de la entropía cromática. Tanto la clase *ot* como las clases de voz no tienen esta organización tan estricta, y por lo tanto tiene sentido que la clase *ot* se parezca más a las clases de voz que a la clase de música para estas características. Se puede comprobar mediante las Tablas 2 y 5 que los mejores resultados se obtienen cuando *ot* se agrupa con el resto de clases de habla (*sp, sn* o *sm*), mientras que los peores se alcanzan al modelar dicha clase por separado, incurriendo en errores de

segmentación entre el habla y *ot*. Se obtienen rendimientos intermedios cuando se omite la clase *ot*, debido a que este tipo de audio se asigna en la etapa de evaluación a una de las clases entrenadas. Como las etiquetas de Ground-Truth delimitan la duración temporal de cada clase, si alguna ocupa más de lo debido resulta en un mayor error de segmentación.

Otro de los factores importantes a destacar es la diferencia de rendimientos (Tablas 2 y 5) en función de la fusión utilizada con el sistema ATVS para ALBAYZIN 2010. Para entender estas diferencias hay que hacer referencia a la distribución de clases acústicas que hace la base de datos ALBAYZIN 2010, pues no todas las clases presentan la misma cantidad de datos (37 % de voz limpia (*sp*), 5 % de música (*mu*), 15 % de voz con música de fondo (*sm*), 40 % de voz con ruido de fondo (*sn*) y 3 % de otros (*ot*)). Cuando la presencia en la base de datos de las clases enfrentadas en un determinado sistema es notablemente distinta, aplicar una fusión de tipo AND dará un rendimiento diferente a si se aplica una fusión de tipo OR. De esta manera se alcanzan mejores resultados aplicando una fusión de tipo AND para aquellas clases que tengan menos presencia en la base de datos, como ocurre en los sistemas **MU-ALL** y **MU-SP**. Ocurre exactamente lo contrario para los sistemas **SP-NSP** y **MUSM-SP-OT**, donde aplicar una fusión OR implica un mejor resultado final. El resto de sistemas, al tener una distribución más igualada, presentan rendimientos más similares. Esto viene a motivar el uso de estrategias de fusión más complejas que las utilizadas.

6.1.1 Trabajo aportado

Este proyecto fin de carrera se ha realizado en los laboratorios del grupo de investigación Área de Tratamiento de Voz y Señales (ATVS) de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid, disponiendo de un ordenador personal con acceso al rack del grupo. Para el desarrollo de este proyecto se ha utilizado uno de los servidores del rack del ATVS, disponiendo de 24 procesadores con 16 GB de RAM.

Este proyecto comenzó a finales del curso 2012/2013 bajo el título "Medidas de similitud de audio para recuperación de información musical", siendo modificado a finales del año 2013 por el título definitivo "Evaluación de características musicales para detección de tipos de audio". Este cambio de orientación vino motivado por la dificultad que implicaban los sistemas de Recuperación de Información Musical, tanto para la tarea de similitud de audio musical como para la de identificación de versiones musicales, al no disponer de bases de datos basadas en audio, como la Million Song Dataset o la Second Hand Songs, respectivamente, basadas exclusivamente en características extraídas a partir del audio. La duración de este proyecto ha sido de 13 meses con una dedicación de media jornada.

Todo el software utilizado a lo largo del proyecto ha sido generado durante el mismo, haciendo uso exclusivamente de los entrenadores GMM del grupo ATVS, así como del paquete NETLAB (Matlab™) para el cálculo de las adaptaciones MAP.

6.1.2 Resultados para el grupo de investigación ATVS

Al finalizar este proyecto, el grupo de investigación ATVS de la EPS de la UAM dispone de un sistema clasificador de audio, capaz de identificar en audio radiofónico distintas clases acústicas, que se complementa con el que ya disponía presentado en la evaluación de segmentación de audio ALBAYZIN 2010. Entre los sistemas desarrollados destacan algunos como un discriminador voz/música o un detector de habla.

También se han llevado a cabo dos sistemas en el campo de la Recuperación de Información Musical, abriendo una nueva línea de investigación para el grupo.

6.2 Trabajo futuro

Como primera mejora se propone ampliar el rango de trabajo del sistema desarrollado para que sea capaz de segmentar el audio de entrada en cinco clases distintas de audio, tal y como hacía en su origen el sistema presentado por ATVS en ALBAYZIN 2010. Esta tarea no llegó a realizarse debido a la complejidad del objetivo, teniendo en cuenta las limitaciones del sistema implementado en comparación con el sistema ATVS para ALBAYZIN 2010.

Otra posible continuación del trabajo realizado es la de incorporar otro tipo de características cromáticas al análisis realizado, tales como los cromagramas. Se ha comprobado que estas representaciones del audio son capaces de discriminar con bastante fiabilidad distintos tipos de audio.

A lo largo de toda la literatura se han estudiado numerosos casos [Pikrakis et al., 2006] en los que los Modelos Ocultos de Markov (Hidden Markov Model, HMM [Rabiner, 1989]) ayudan significativamente en la tarea de segmentación de audio. Puesto que las características calculadas en este proyecto (media, varianza, skewness y kurtosis a partir de la entropía cromática) presentan un alto grado de discriminación en cuanto a segmentación de audio se refiere, una de las propuestas de mejora con mayor futuro es la de fusionar a nivel de característica las extraídas en este sistema con las que utiliza el sistema ATVS de ALBAYZIN 2010 (MFCC + SDC), y hacer uso de los HMMs y del algoritmo de Viterbi que tan buenos resultados dieron.

Referencias

[Atal, 1974]
B. S. Atal, "*Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification*". J. Acoust. Soc. Amer. 55(6), 1304-1312, 1974.

[Atal & Hanauer, 1971]
B. Atal, S. Hanauer, "*Speech anaysis and synthesis by linear prediction of the speech wave*". J. Acoust. Soc. Amer. 50(2), 637-655, 1971.

[Aucouturier & Pachet, 2004]
Jean-Julien Aucouturier, Francois Pachet, "*Improving Timbre Similarity: How high is the sky?*". Journal of Negative Results in Speech and Audio Sciences, 1(1), 2004.

[B. Fauve, 2009]
Benoîit Fauve, "*Tackling Variabilities in Speaker Verification with a Focus on Short Durations*". PHD, School of Engineering, Swansea University, 2009.

[Berenzweig et al., 2004]
Adam Berenzweig, Beth Logan, Daniel P.W. Ellis, Brian Whitman, "*A Large-Scale Evaluation of Acoustic and Subjective Music- Similarity Measures*". Computer Music Journal, 28(2), 63–76, Summer 2004.

[Bertin-Mahieux et al., 2011]
Thierry Bertin-Mahieux, Daniel P. W. Ellis, Brian Whitman, Paul Lamere, "*The Million Song Dataset*". Proceedings of the 2011 International Conference on Music Information Retrieval and Related Activities (ISMIR), 2011.

[Brümmer et al., 2007]
N. Brümmer, L. Burget, J. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D. A. van Leeuwen, P. Matejka, P. Scwartz, and A. Strasheim, "*Fusion of Heterogeneous Speaker Recognition Systems in the STBU Submission for the NIST Speaker Recognition Evaluation 2006*". IEEE Transactions on Audio, Speech and Signal Processing, 15(7), 2072–2084, 2007.

[Butko, 2010]
Taras Butko, "*Albayzin evaluations 2010: Audio Segmentation*". Universidad de Vigo, FALA 2010, Albayzin 2010 Evaluation Campaign, 2010.

[Butko et al, 2010]
Taras Butko, Climent Nadeu, Henrik Schulz, "*Albayzin-2010 Audio Segmentation Evaluation: Evaluation Setup and Results*". FALA 2010, 2010.

[Chen & Gopalakrishnan, 1998]
S. S. Chen, P. Gopalakrishnan, "*Clustering via the bayesian information criterion with applications in speech recognition*". Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, Vol. 2, Seattle, USA, 645-648, 1998.

[Cheng et al., 2008]
Shih-Sian Cheng, Hsin-Min Wang, Hsin-Chia Fu, "*BIC-based audio segmentation by divide-and-conquer*". Acoustics, Speech and Signal Processing (ICASSP). IEEE International Conference, Las Vegas, NV, Abril, 4841-4844, 2008.

[Davis & Mermelstein, 1980]
S. Davis, P. Mermelstein, "*Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences*". IEEE Trans. Acoustics, Speech, Signal Process. 28(4), 357-366, 1980.

[Dehak et al., 2011]
Najim Dehak, Patrick J. Kenny, Réda Dehak, Pierre Dumouchel, Pierre Ouellet, "*Front-End Factor Analysis for Speaker Verification*". IEEE Transactions on Audio, Speech, and Language Processing, 19(4), 2011.

[Downie, 2008]
J. Stephen Downie, "*The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research*". Acoust. Sci. & Tech. 29, 4, 2008.

[Downie et al., 2010]
J. Stephen Downie, Andreas F. Ehmann, Mert Bay, M. Cameron Jones, "*The Music Information Retrieval Evaluation eXchange: Some Observations and Insights*". Advances in Music Information Retrieval, 274, 93-115, 2010.

[Duda et al., 2000]
Richard O. Duda, Peter E. Hart, David G. Stork, "*Pattern Classification*". Segunda edición. Wiley-Interscience, Cap. 1-3, 2000.

[Ellis et al., 2002]
Daniel P. W. Ellis, Brian Whitman, Adam Berenzweig, Steve Lawrence, "*The Quest for Ground Truth in Musical Artist Similarity*". Proceedings of the 2002 International Conference on Music Information Retrieval and Related Activities (ISMIR), 2002.

[Ellis & Poliner, 2007]
D. P. Ellis, Graham E. Poliner, "*Identifying `Cover Songs' with Chroma Features and Dynamic Programming Beat Tracking*". In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP, Vol. 4, April 2007.

[Ellis & Cotton, 2007]
D. P. W. Ellis, C. Cotton, "*The 2007 LabROSA Cover Song Detection System*". Music Information Retrieval Evaluation eXchange (MIREX), 2007.

[Fierrez-Aguilar et al., 2005]
J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez, "*Target dependent score normalization techniques and their application to signature verification*". IEEE Trans. On Systems, Man and Cybernetics, part C, 35(3), 418-425, 2005.

[Franco-Pedroso et al., 2010]
J. Franco-Pedroso, I. Lopez-Moreno, D. T. Toledano, J. Gonzalez-Rodriguez, "*ATVS-UAM System Description for the Audio Segmentation and Speaker Diarization Albayzin 2010 Evaluation*". in Proc. FALA 2010, Vigo, Spain, 2010.

[Furui, 1981]
S. Furui, "*Cepstral analysis technique for automatic speaker verification*". IEEE Trans. Acoustics, Speech Signal Process. 29(2), 254-272, 1981.

[Gómez, 2014]
Iván Gómez Piris, "*Extracción de información en señales de voz para el agrupamiento por locutores de locuciones anónimas*". Proyecto Fin de Carrera, Escuela Politécnica Superior, Universidad Autónoma de Madrid, 2014.

[Huai-You et al., 2012]
Chang Huai You, Haizhou Li, Bin Ma, Kong Aik Lee, "*Effect of Relevance Factor of Maximum a posteriori Adaptation for GMM-SVM in Speaker and Language Recognition*". Interspeech 2012, 1(1), 2012.

[Kinnunen & Li, 2010]
Tomi Kinnunen, Haizhou Li, "*An overview of text-independent speaker recognition: From features to supervectors*". Speech Communication 52, 1-22, 2010.

[Kittler et al., 1998]
Josef Kittler, Mohamad Hatef, Robert P.W. Duin, Jiri Matas "On Combining Classifiers". IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(3), march 1998.

[Oppenheim et al., 2011]
A. Oppenheim, R. Schafer, J. Buck, "*Tratamiento de señales en tiempo discreto*". Tercera edición. Prentice-Hall, 2011.

[Ortega, 2012]
J. Ortega, "*Ampliación de señales aleatorias*". Ingeniería de Telecomunicación de la UAM, 2012.

[Pearson & Lipman, 1988]
W. R. Pearson, D. J. Lipman, "*Improved tools for biological sequence comparison*". Proc Natl Acad Sci USA, Abril 85(8), 2444–2448, 1988.

[Pikrakis et al., 2006]
A. Pikrakis, T. Giannakopoulos, S. Theodoridis, "*A computationally efficient speech/music discriminator for radio recordings*". Proceedings of the 2006 International Conference on Music Information Retrieval and Related Activities (ISMIR), 8-12 October 2006, Victoria BC, Canada, 2006.

[Pikrakis et al., 2006]
Aggelos Pikrakis, Theodoros Giannakopoulos, Sergios Theodoridis, "*Speech/Music Discrimination for Radio Broadcasts Using a Hybrid HMM-Bayesian Network Architecture*". 14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006.

[Pohle et al., 2009]
Tim Pohle, Dominik Schnitzer, Markus Schedl, Peter Knees, Gerhard Widmer, "*On rhythm and general music similarity*". 10th International Society for Music Information Retrieval Conference (ISMIR'09), 2009.

[Rabiner, 1989]
L. R. Rabiner, "*A tutorial on Hidden Markov Models and selected applications in speech recognition*". Proceedings of the IEEE 77, 257–286, 1989.

[Reynolds et al., 2000]
D. A. Reynolds, T. F. Queatieri, R. B. Dunn, "*Speaker Verification Using Adapted Gaussian Mixture Models*". Digital Signal Processing 10, 19-41, 2000.

[Reynolds et al., 2003]
Douglas Reynolds, Walter Andrews, Joseph Campell, Jiri Navratil, Barbara Peskin, Andre Adam, Qin Jin, David Klusacek, Joy Abramson, Radu Mihaescu, Jack Godfrey, Doug Jones, Bing Xiang, "*SuperSID Project: "Exploiting high-level information for high accuracy speaker recognition"*". Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vol. 4, 784-787, Abril 2003.

[Reynolds & Campbell, 2008]
D. A. Reynolds, W. M. Campbell, "*Springer Handbook of Speech Processing. Text-Independent Speaker Recognition*". Springer, 38, 763-771, 2008.

[Reynolds & Rose, 1995]
D. Reynolds, R. Rose, "*Robust text-independent speaker identification using Gaussian mixture speaker models*". IEEE Trans. Speech Audio Process. 3, 72-83, 1995.

[Robin Harald Priewald, 2009]
Robin Harald Priewald, "*Classification of Acoustic Plastic Pipe Water Leak Signals with Gaussian Mixture Models*", Graz University of Technology, Diploma Tesis, 2009.

[Serrà et al., 2009]
Joan Serrà, Emilia Gómez, Perfecto Herrera, "*Audio cover song identification and similarity: background, approaches, evaluation, and beyond*". Music Technology Group, Department of Information and Communication Technologies, Universitat Pompeu Fabra, 2009.

[Seyerlehner et al., 2007]
Klaus Seyerlehner, Tim Pohle, Markus Schedl, Gerhard Widmer, "*Automatic Music Detection in Television Productions*". Proc. of the 10th Int. Conference on Digital Audio Effects (DAFx-07), Bordeaux, France, September 10-15, 2007.

[Spiegel, 1970]
Murray R. Spiegel, "*Estadística*". Primera edición. Serie Schaum, Libros McGraw-Hill, Cap. 3 y 5, 45-68, 89-98, 1973.

Anexos

A Frecuencias centrales de la escala musical

Tomando f_0 como la primera nota musical (se ha tomado el Do de 65,406 Hz), el resto de notas musicales se pueden calcular según la fórmula:

$$f(Hz) = f_0 \cdot 2^{\frac{k}{12}} \quad k = 0,1,\dots,L-1 \tag{B.1}$$

Puesto que $sr = 16000\ Hz$ (siendo sr la frecuencia de muestreo del audio presente en la base de datos utilizada), y según (3.2), las 84 frecuencias centrales del banco de filtros son:

Tabla 7: Frecuencias centrales de la escala musical (A)

Do 1	65,406 Hz	Do 2	130,812 Hz	Do 3	261,625 Hz
Do# 1	69,295 Hz	Do# 2	138,591 Hz	Do# 3	277,182 Hz
Re 1	73,416 Hz	Re 2	146,832 Hz	Re 3	293,664 Hz
Re# 1	77,781 Hz	Re# 2	155,563 Hz	Re# 3	311,127 Hz
Mi 1	82,406 Hz	Mi 2	164,813 Hz	Mi 3	329,627 Hz
Fa 1	87,307 Hz	Fa 2	174,614 Hz	Fa 3	349,228 Hz
Fa# 1	92,498 Hz	Fa# 2	184,997 Hz	Fa# 3	369,994 Hz
Sol 1	97,998 Hz	Sol 2	195,997 Hz	Sol 3	391,995 Hz
Sol# 1	103,826 Hz	Sol# 2	207,652 Hz	Sol# 3	415,304 Hz
La 1	110 Hz	La 2	220 Hz	La 3	440 Hz
La# 1	116,540 Hz	La# 2	233,081 Hz	La# 3	466,163 Hz
Si 1	123,470 Hz	Si 2	246,941 Hz	Si 3	493,883 Hz

Tabla 8: Frecuencias centrales de la escala musical (B)

Do 4	523,251 Hz	Do 5	1046,502 Hz
Do# 4	554,365 Hz	Do# 5	1108,730 Hz
Re 4	587,329 Hz	Re 5	1174,659 Hz
Re# 4	622,254 Hz	Re# 5	1244,508 Hz
Mi 4	659,255 Hz	Mi 5	1318,510 Hz
Fa 4	698,456 Hz	Fa 5	1396,913 Hz
Fa# 4	739,988 Hz	Fa# 5	1479,977 Hz
Sol 4	783,990 Hz	Sol 5	1567,981 Hz
Sol# 4	830,609 Hz	Sol# 5	1661,219 Hz
La 4	880 Hz	La 5	1760 Hz
La# 4	932,327 Hz	La# 5	1864,655 Hz
Si 4	987,766 Hz	Si 5	1975,533 Hz

Tabla 9: Frecuencias centrales de la escala musical (C)

Do 6	2093,004 Hz	Do 7	4186,009 Hz
Do# 6	2217,461 Hz	Do# 7	4434,922 Hz
Re 6	2349,318 Hz	Re 7	4698,636 Hz
Re# 6	2489,016 Hz	Re# 7	4978,032 Hz
Mi 6	2637,020 Hz	Mi 7	5274,041 Hz
Fa 6	2793,826 Hz	Fa 7	5587,652 Hz
Fa# 6	2959,955 Hz	Fa# 7	5919,911 Hz
Sol 6	3135,963 Hz	Sol 7	6271,927 Hz
Sol# 6	3322,438 Hz	Sol# 7	6644,876 Hz
La 6	3520 Hz	La 7	7040 Hz
La# 6	3729,310 Hz	La# 7	7458,621 Hz
Si 6	3951,066 Hz	Si 7	7902,133 Hz

B Ejemplos prácticos de segmentación de características y scoring

En esta sección se estudia un caso práctico de segmentación de características y cálculo de puntuaciones.

Supongamos que disponemos de las puntuaciones para el sistema **MU-ALL** de una secuencia de audio de 150 milisegundos. Esas puntuaciones se pueden organizar en lo que llamamos una "matriz de scores". Dicha matriz de scores se ha calculado a partir del vector de características extraído a partir del audio, de tamaño 4x15 (puesto que el sistema trabaja con 4 características [media, varianza, skewness y kurtosis] y el tamaño de enventanado al ser de 20 milisegundos con un solapamiento del 50 %: $\frac{150\ ms}{20\ ms \cdot 50\ \%} = 15$). Ya que el sistema sólo hace distinción entre dos clases acústicas (*mu* y *all=sp+sn+sm+ot*), la matriz de scores tendrá un tamaño de 2x15. De esta manera, la matriz de scores con la que se segmentará el audio es:

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
mu	1	3	2	1	3	4	4	5	2	2	1	1	4	2	2
all	3	4	3	2	1	2	1	2	2	3	4	4	3	3	4

siendo la primera fila los scores correspondientes a la clase acústica *mu*, mientras que la segunda fila corresponden a *all*.

A partir de la matriz de scores se han desarrollado dos procesos a seguir: segmentar el audio a partir de dicha matriz tomando la clase correspondiente a la puntuación máxima para cada trama y filtrando posteriormente las clases acústicas mediante un filtrado por moda, o filtrar los scores mediante medias o medianas y luego segmentar los scores modificados.

B.1 Segmentación frame-by-frame y filtrado por moda

Para llevar a cabo este proceso se empieza determinando los máximos de la matriz de scores. Esto se puede realizar gracias al uso del UBM en el cálculo de los scores, pudiendo así comparar las puntuaciones de una misma columna sin problema. En caso de empate se decidirá en favor de la clase principal del sistema (en este caso *mu*). No obstante es prácticamente imposible que se dé este caso, pues las matrices de scores contienen puntuaciones extraídas directamente de los GMMs, lo que implica números con más de 4 decimales:

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
mu	1	3	2	1	3	4	4	5	2	2	1	1	4	2	2
all	3	4	3	2	1	2	1	2	2	3	4	4	3	3	4

La segmentación resultante sería:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
all	all	all	all	mu	mu	mu	mu	mu	all	all	all	mu	all	all

Es a partir de la segmentación que se hace un filtrado por moda. Suponiendo un tamaño de la ventana de filtrado de 30 milisegundos (3 tramas), el resultado final quedaría:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
all	all	all	all	mu	mu	mu	mu	mu	all	all	all	all	all	all

Cabe destacar que las tramas de los extremos que no pudieran ser filtrados permanecen iguales.

El fichero resultante (escrito de manera análoga a como se indicó en el apartado 3.3.) es:

SecuenciaAudio	0.00	0.04	all
SecuenciaAudio	0.04	0.09	mu
SecuenciaAudio	0.09	0.15	all

B.2 Filtrado por media y segmentación

A diferencia del proceso anterior, aquí se filtra primero los scores y luego se segmenta el audio. Suponiendo un tamaño de ventana de 30 milisegundos (3 tramas), la matriz de scores resultante es:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
mu	1	2	2	2	2,6	3,6	4,3	3,6	3	1,6	1,3	2	2,3	2,6	2
all	3	3,3	3	2	1,6	1,3	1,6	1,6	2,3	3	3,6	3,6	3,3	3,3	4

Seleccionando los scores máximos se procede a la segmentación del audio:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
mu	1	2	2	2	2,6	3,6	4,3	3,6	3	1,6	1,3	2	2,3	2,6	2
all	3	3,3	3	2	1,6	1,3	1,6	1,6	2,3	3	3,6	3,6	3,3	3,3	4

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
all	all	all	mu	mu	mu	mu	mu	mu	all	all	all	all	all	all

Se puede apreciar una ligera diferencia frente a los resultados obtenidos mediante el filtrado por moda, pues la trama 4 fue identificada como *all*, mientras que ahora se etiqueta como *mu*.

El fichero resultante (escrito de manera análoga a como se indicó en el apartado 3.3.) es:

SecuenciaAudio	0.00	0.03	all
SecuenciaAudio	0.03	0.09	mu
SecuenciaAudio	0.09	0.15	all

B.3 Filtrado por mediana y segmentación

De manera similar al proceso anterior, aquí se filtra primero los scores y luego se segmenta el audio. Suponiendo un tamaño de ventana de 30 milisegundos (3 tramas), la matriz de scores resultante es:

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
mu	1	2	2	2	3	4	4	4	2	2	1	1	2	2	2
all	3	3	3	2	2	1	2	2	2	3	4	4	3	3	4

Seleccionando los scores máximos se procede a la segmentación del audio:

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
mu	1	2	2	2	3	4	4	4	2	2	1	1	2	2	2
all	3	3	3	2	2	1	2	2	2	3	4	4	3	3	4

1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
all	all	all	mu	mu	mu	mu	mu	mu	all	all	all	all	all	all

En este caso no hay diferencia alguna frente a los resultados obtenidos mediante el filtrado por media. Esto se debe a la simplicidad de la matriz de scores utilizada. Si se usase una matriz de scores real, las diferencias entre filtrar por medias y filtrar por medianas serían notables.

El fichero resultante sería idéntico al de la sección anterior (C.2)

C Representaciones gaussianas para las 5 clases acústicas definidas en ALBAYZIN 2010 utilizando adaptación MAP sólo de medias y completa

En el apartado 4.1.1. se ha realizado un estudio de la repercusión en el rendimiento final del sistema en cuanto al tipo de adaptación MAP utilizada, ya sea la adaptación exclusiva del vector de medias como la adaptación completa en la que también se modifican los vectores de pesos y las matrices de covarianzas.

En este anexo se pueden observar el resto de representaciones gaussianas de las clases acústicas definidas en la evaluación ALBAYZIN 2010 de segmentación de audio, habiendo estudiado las de *sp* y *mu* en la sección 4.1.1.

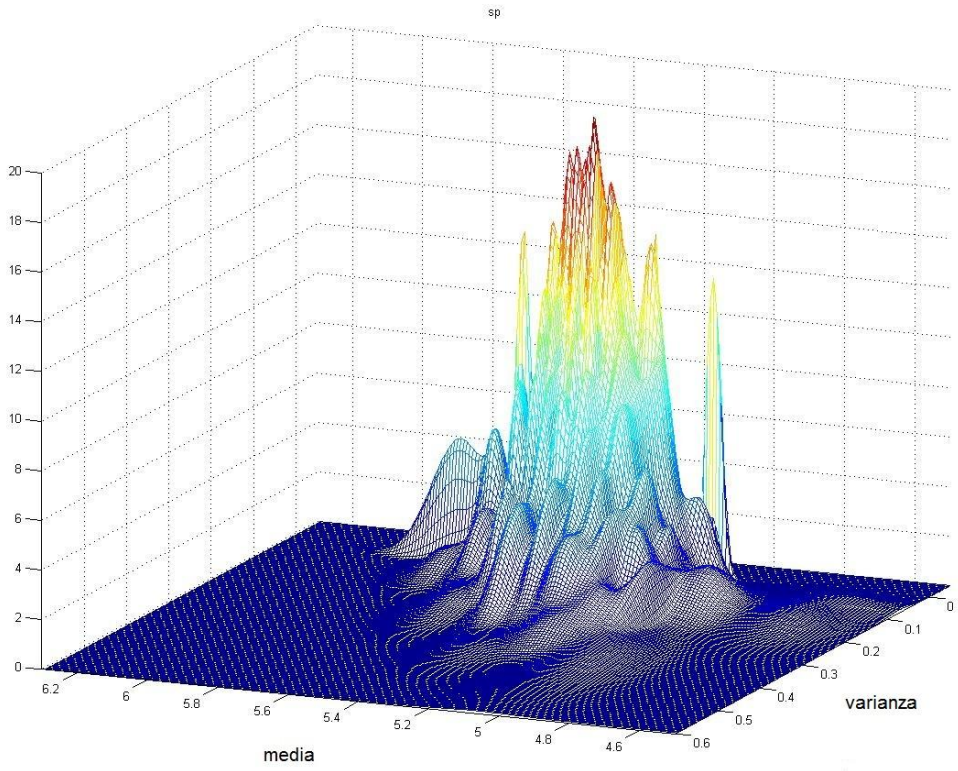


Figura 0-1: GMM adaptando sólo medias para *sp*

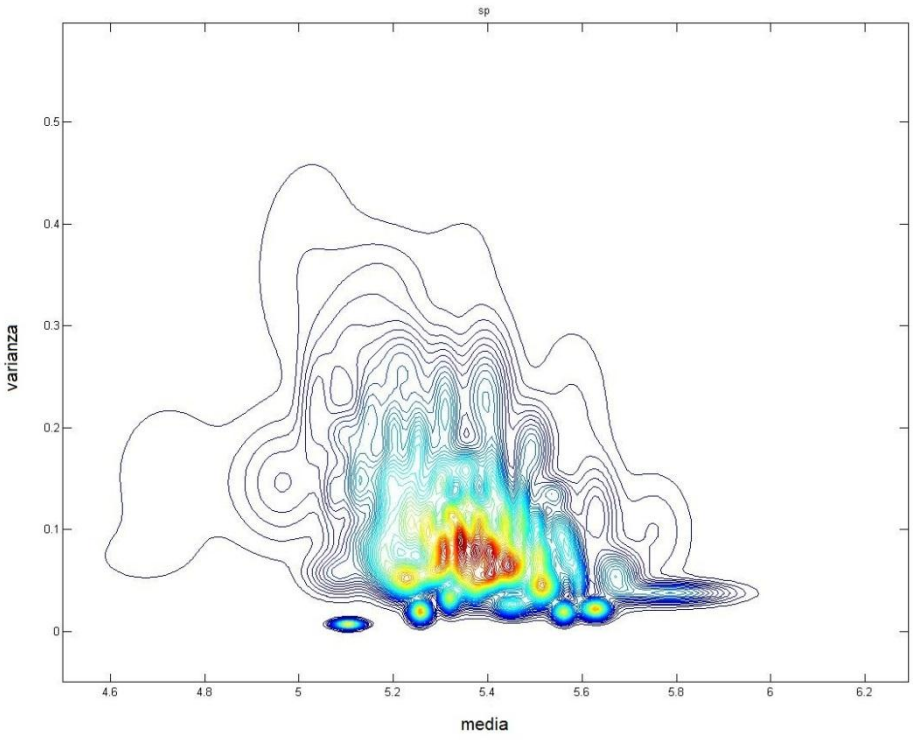


Figura 0-2: Curvas de nivel del GMM adaptando sólo medias para *sp*

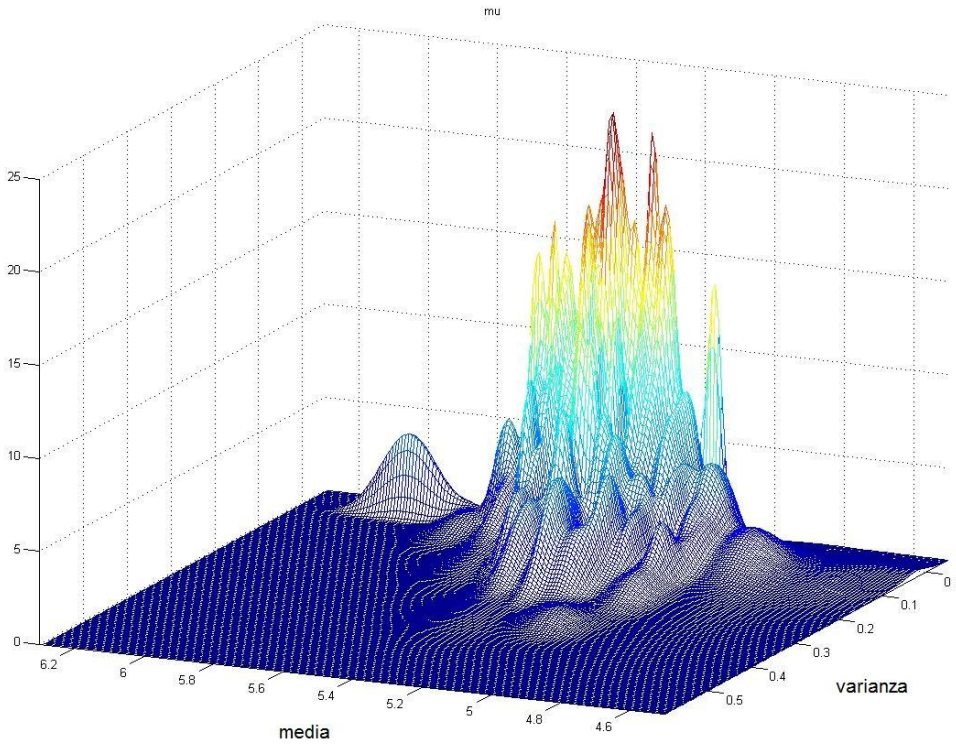


Figura 0-3: GMM adaptando sólo medias para *mu*

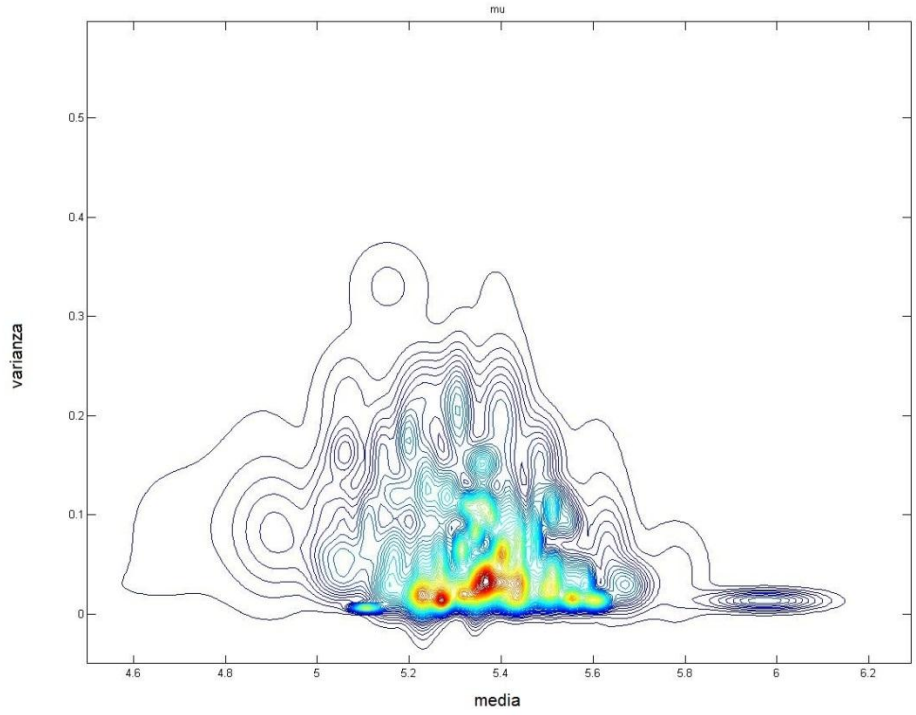


Figura 0-4: Curvas de nivel del GMM adaptando sólo medias para *mu*

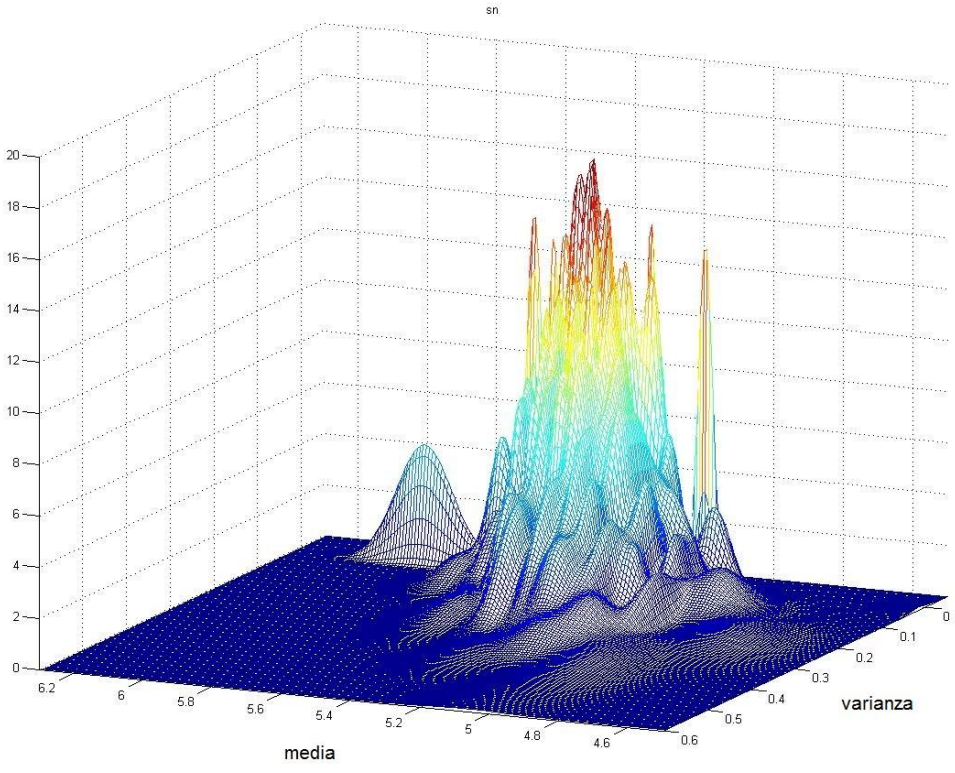


Figura 0-5: GMM adaptando sólo medias para *sn*

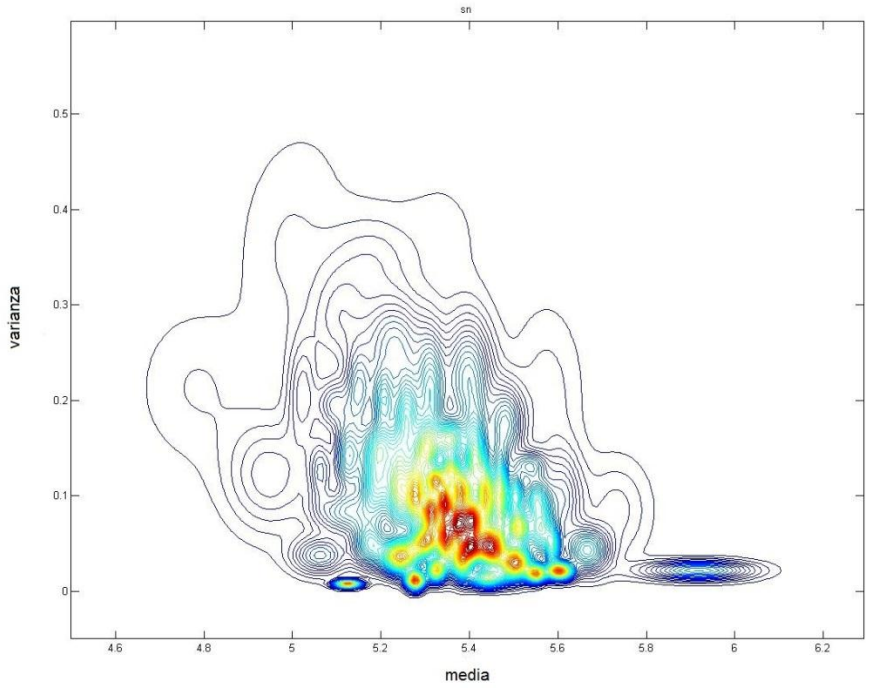


Figura 0-6: Curvas de nivel del GMM adaptando sólo medias para *sn*

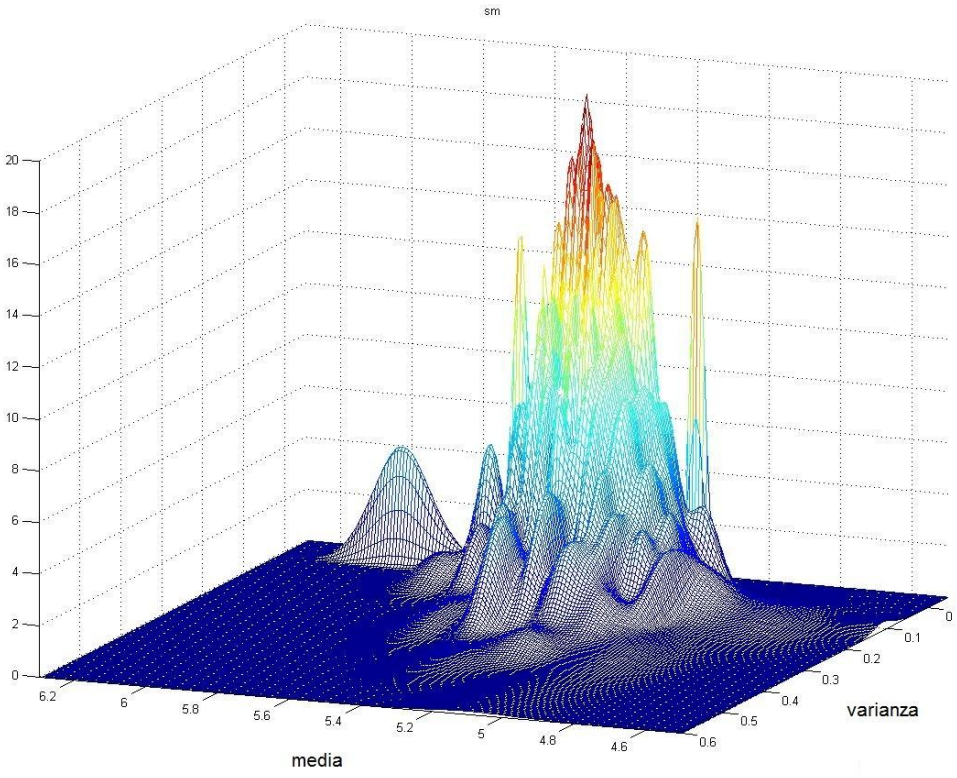


Figura 0-7: GMM adaptando sólo medias para *sm*

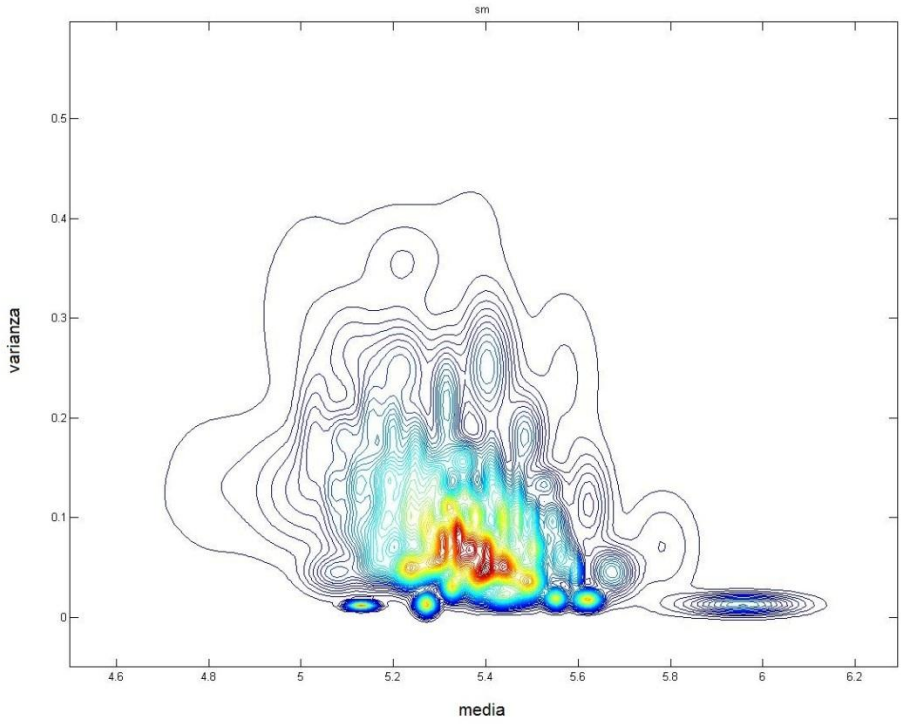


Figura 0-8: Curvas de nivel del GMM adaptando sólo medias para *sm*

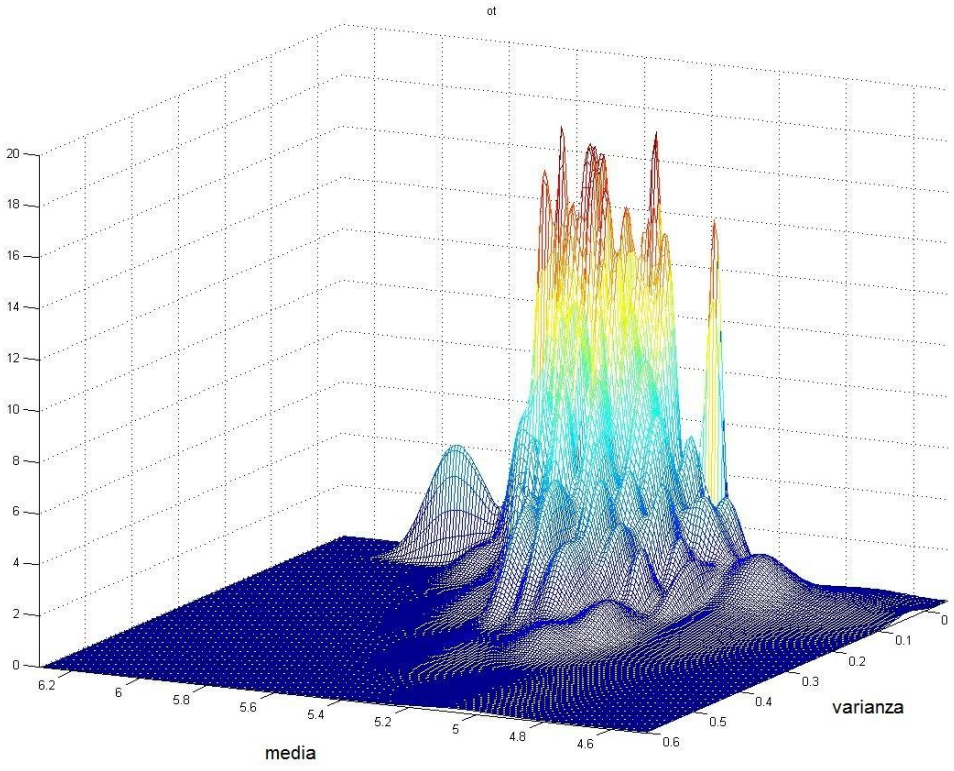


Figura 0-9: GMM adaptando sólo medias para *ot*

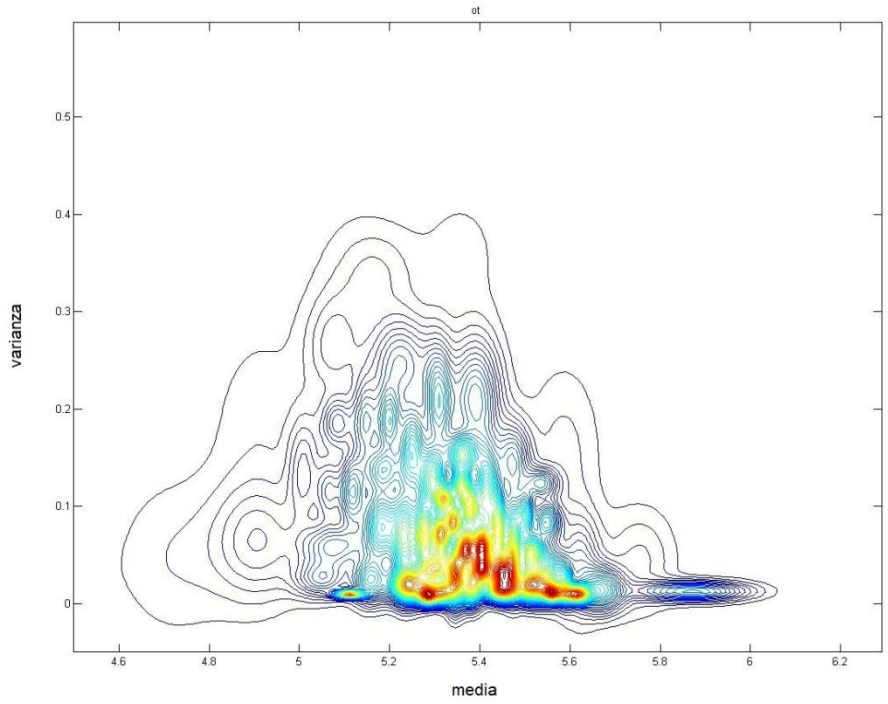


Figura 0-10: Curvas de nivel del GMM adaptando sólo medias para *ot*

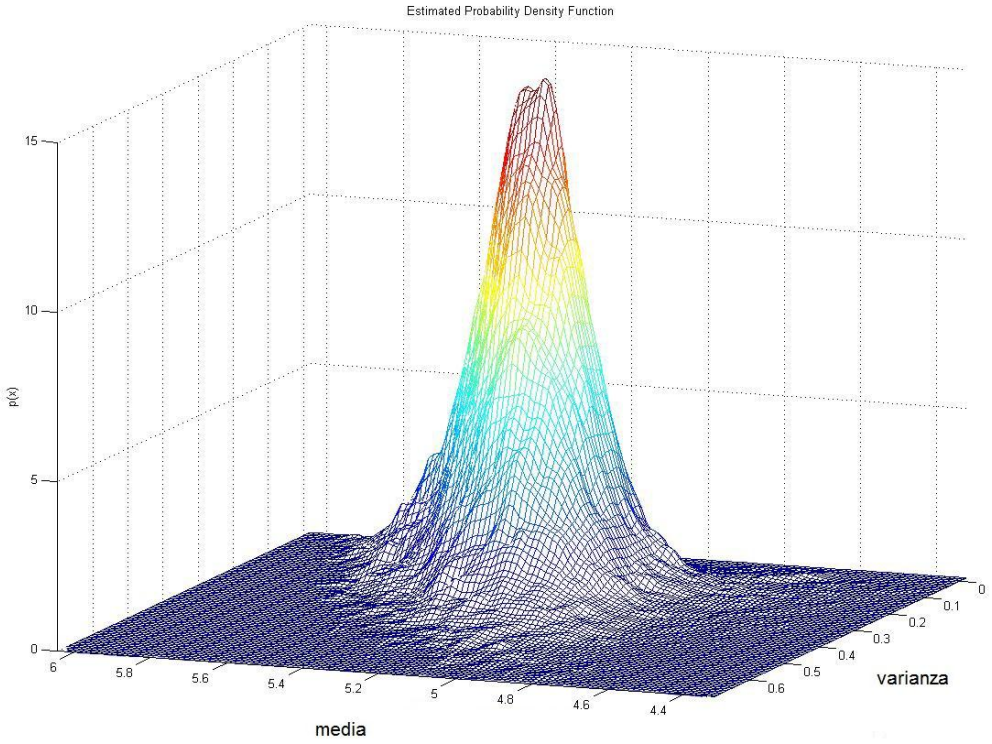


Figura 0-11: Densidad de probabilidad con KDF para *sp*

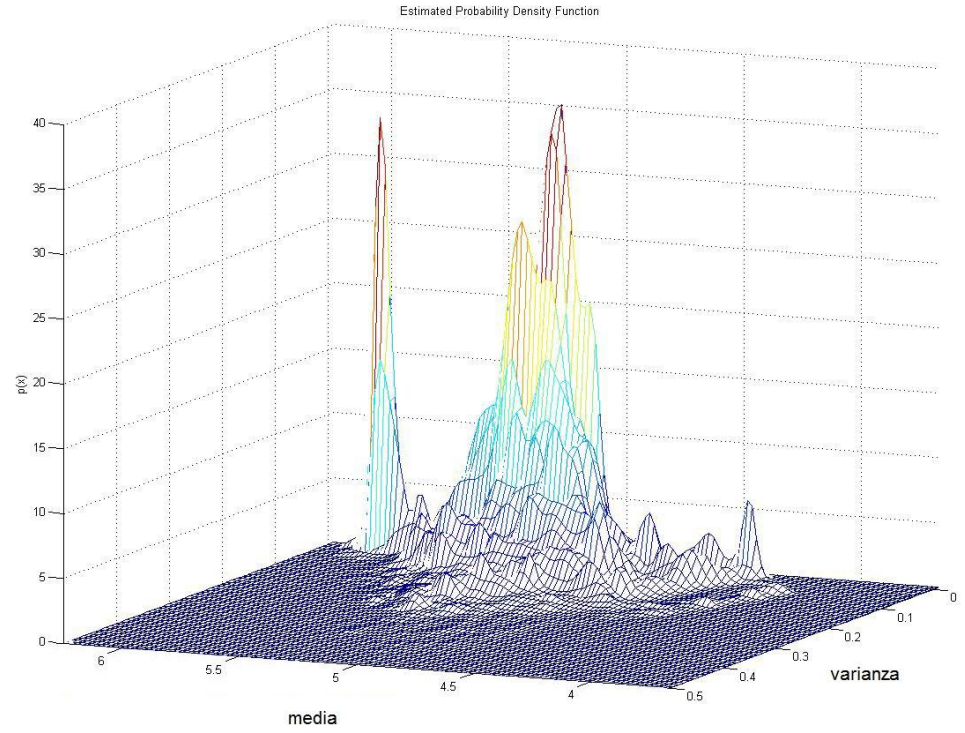


Figura 0-12: Densidad de probabilidad con KDF para *mu*

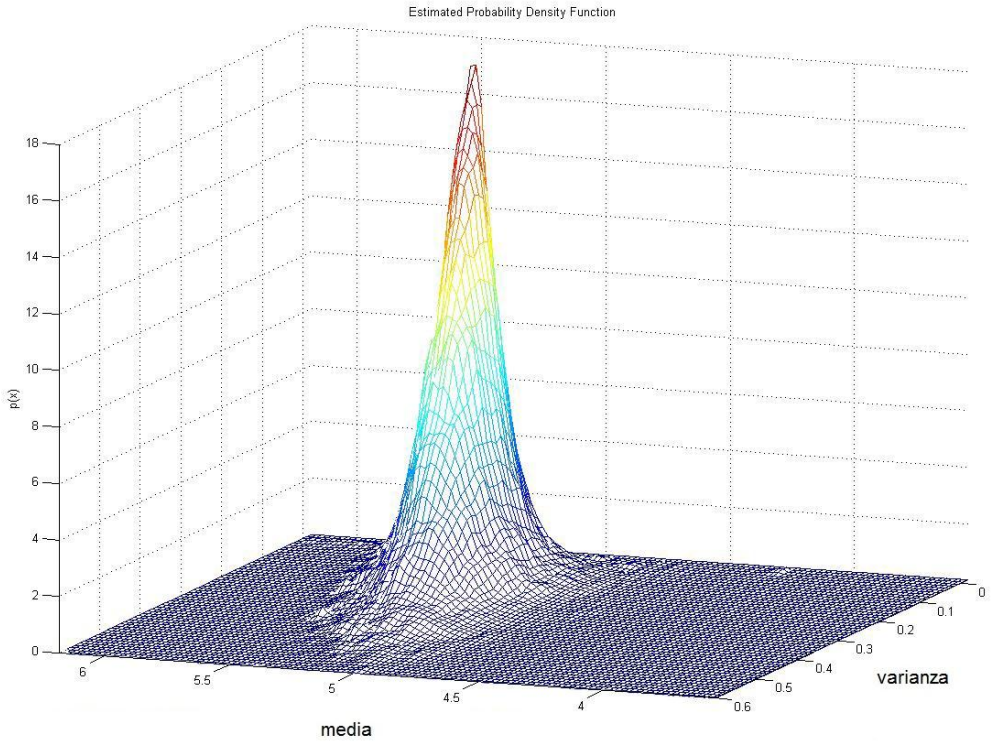


Figura 0-13: Densidad de probabilidad con KDF para *sn*

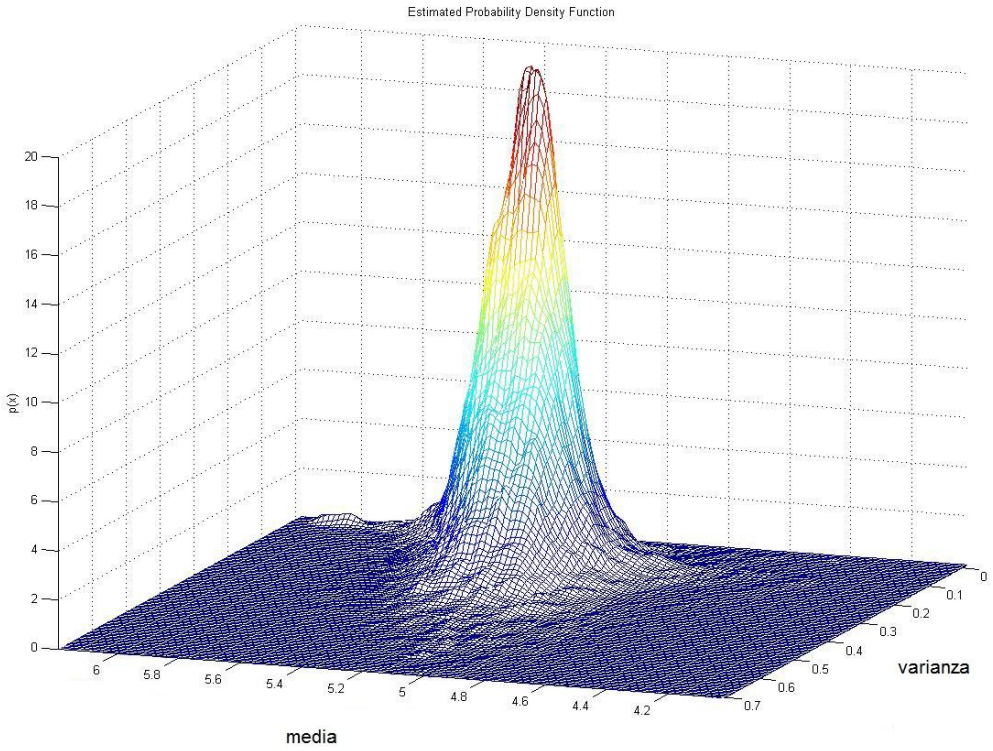


Figura 0-14: Densidad de probabilidad con KDF para *sm*

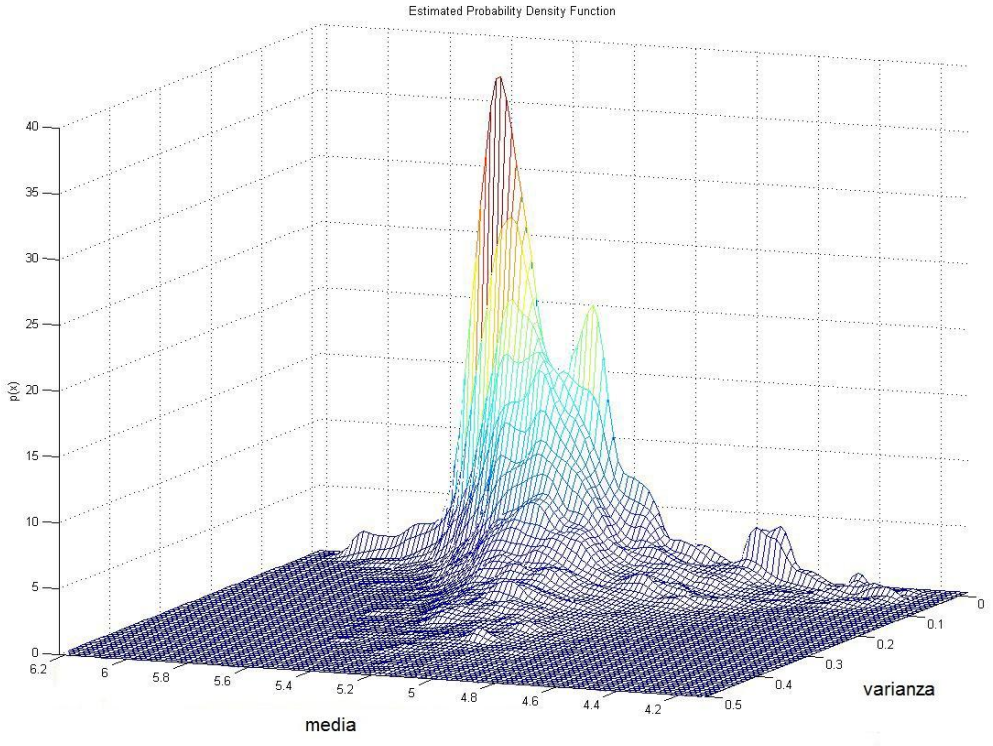


Figura 0-15: Densidad de probabilidad con KDF para *ot*

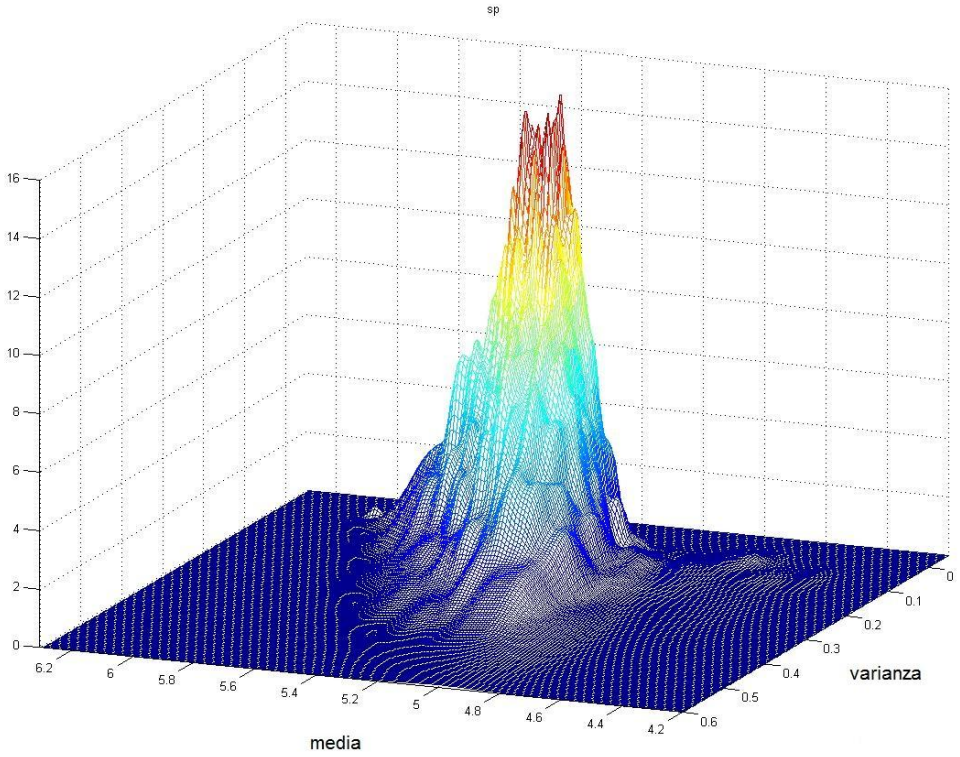


Figura 0-16: GMM adaptando todos los parámetros para *sp*

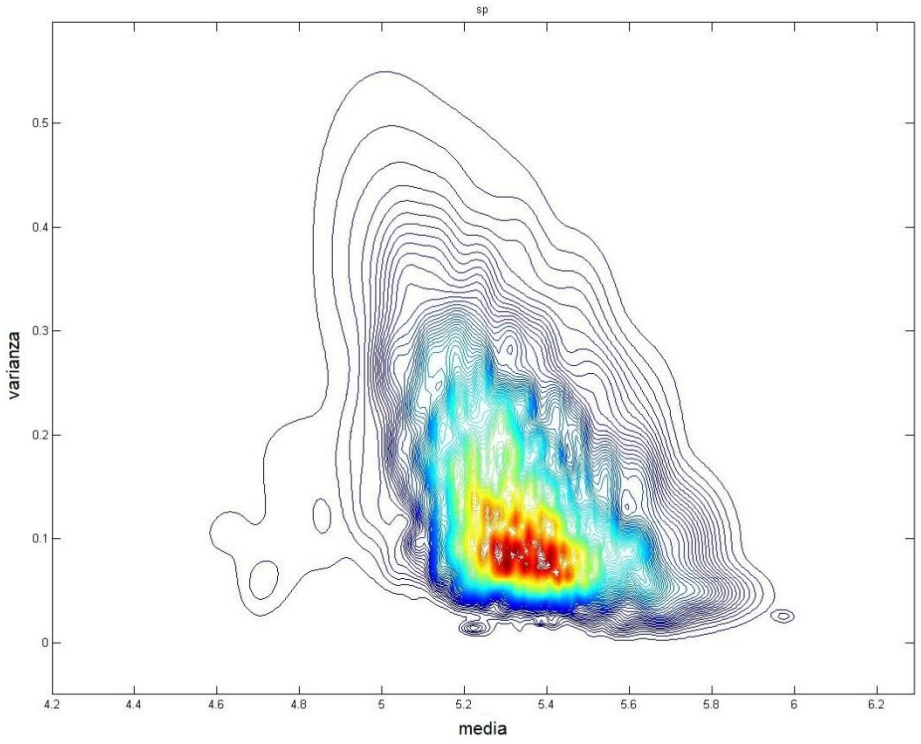


Figura 0-17: Curvas de nivel del GMM adaptando todos los parámetros para *sp*

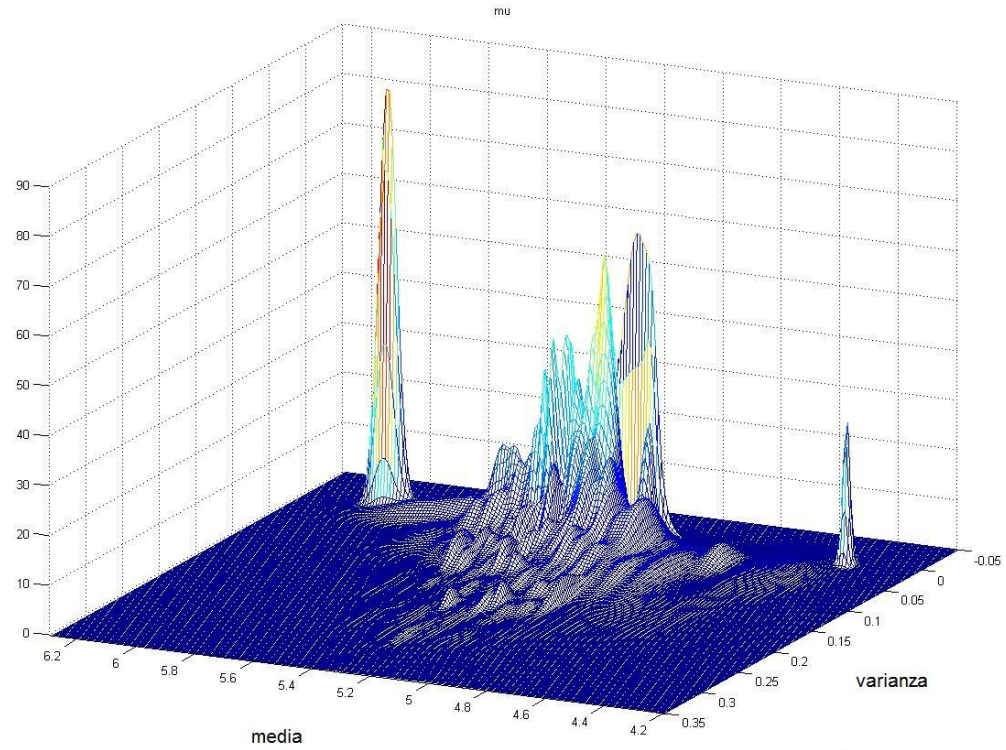


Figura 0-18: GMM adaptando todos los parámetros para *mu*

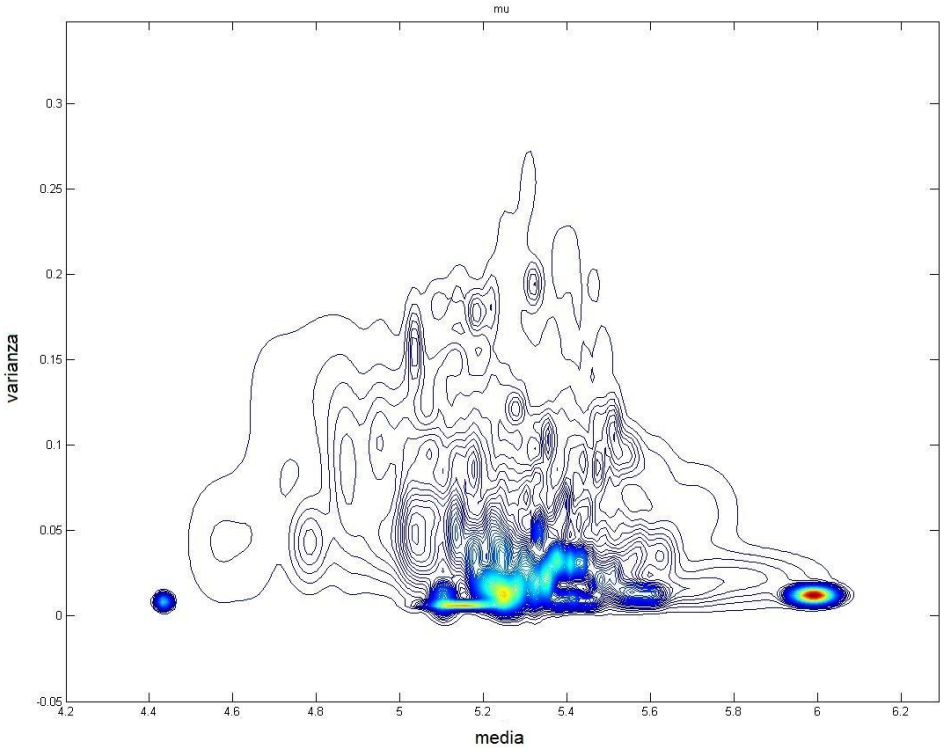


Figura 0-19: Curvas de nivel del GMM adaptando todos los parámetros para *mu*

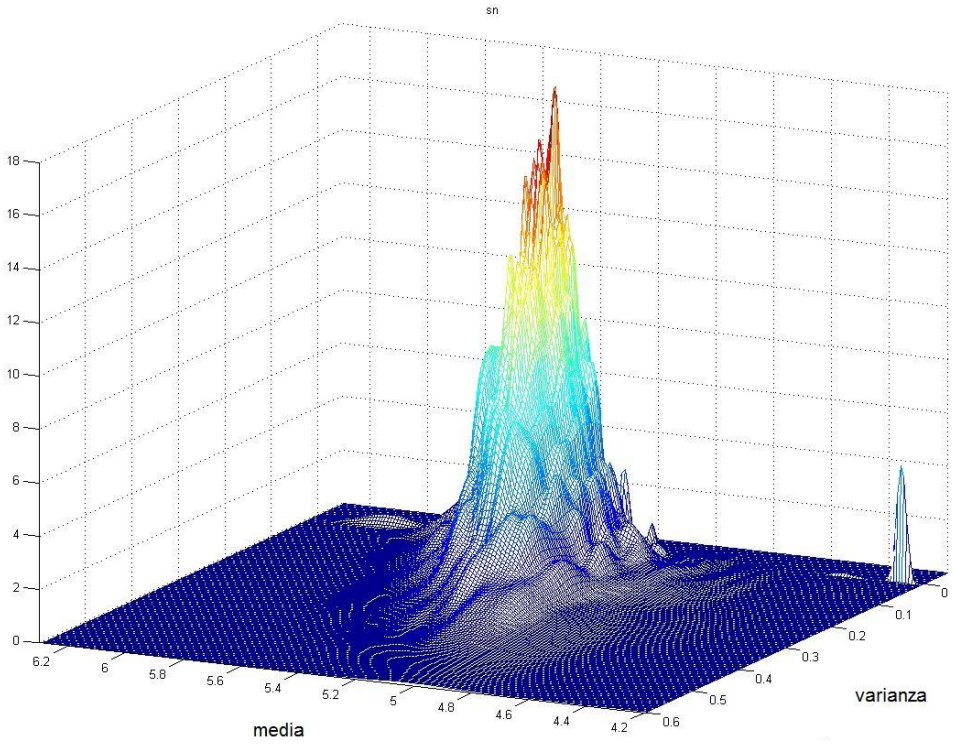


Figura 0-20: GMM adaptando todos los parámetros para *sn*

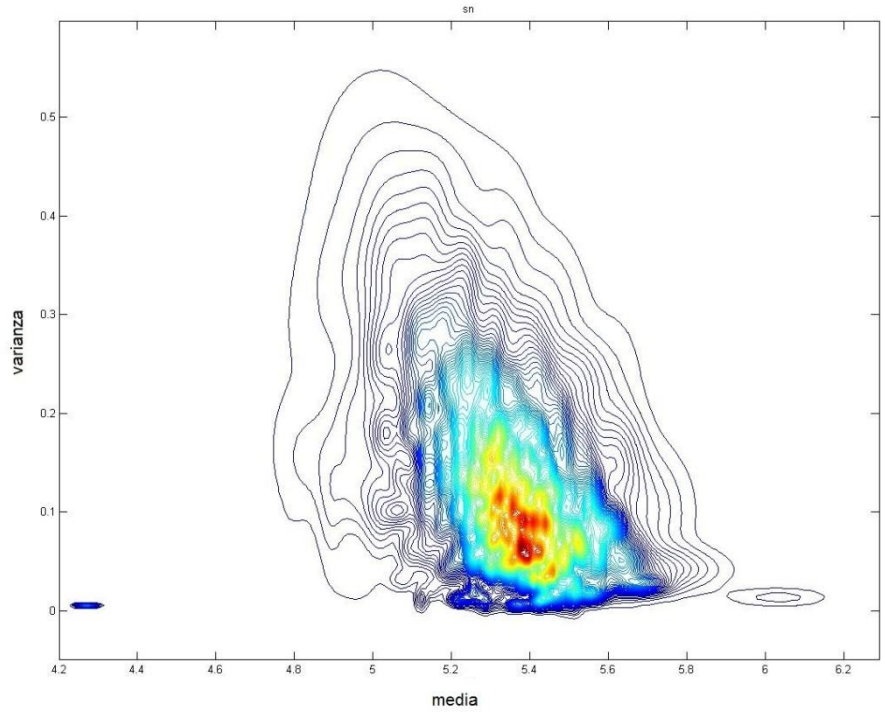


Figura 0-21: Curvas de nivel del GMM adaptando todos los parámetros para *sn*

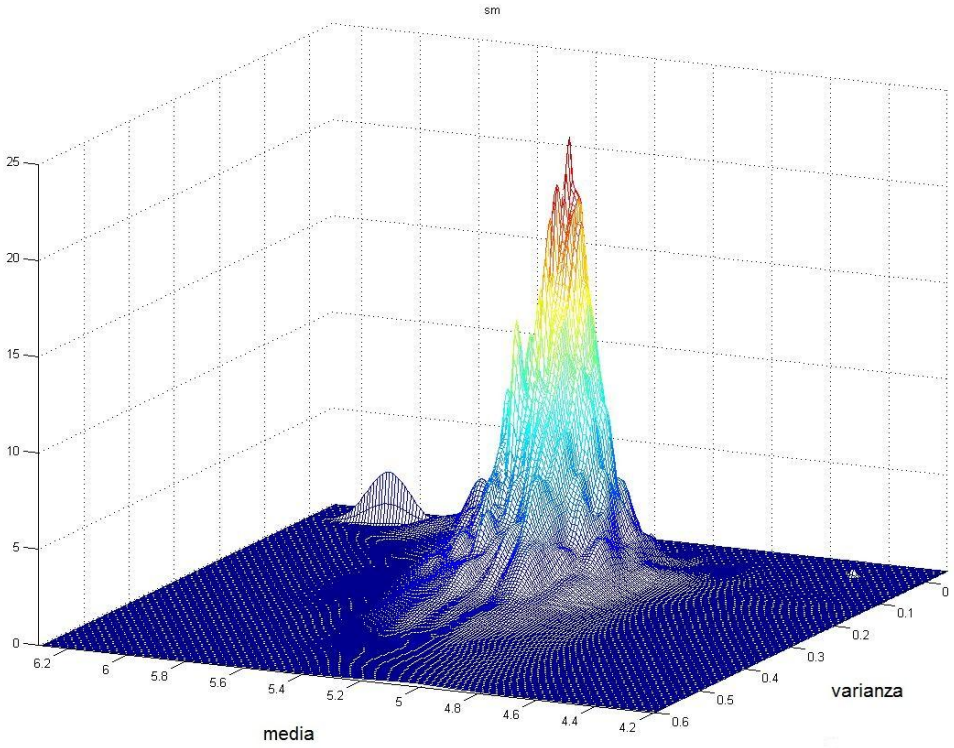


Figura 0-22: GMM adaptando todos los parámetros para *sm*

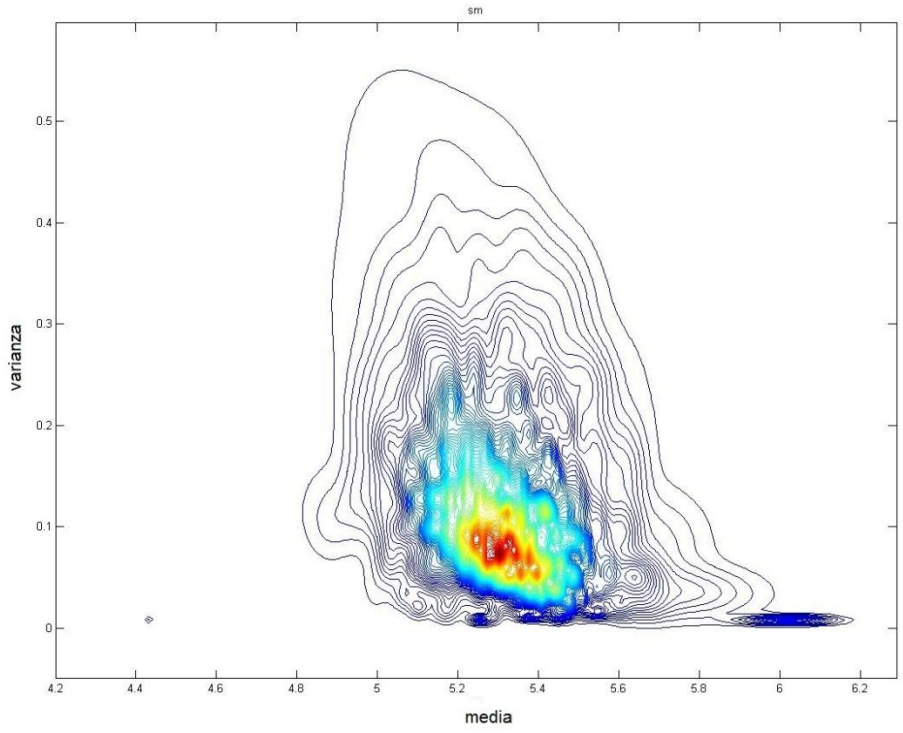


Figura 0-23: Curvas de nivel del GMM adaptando todos los parámetros para *sm*

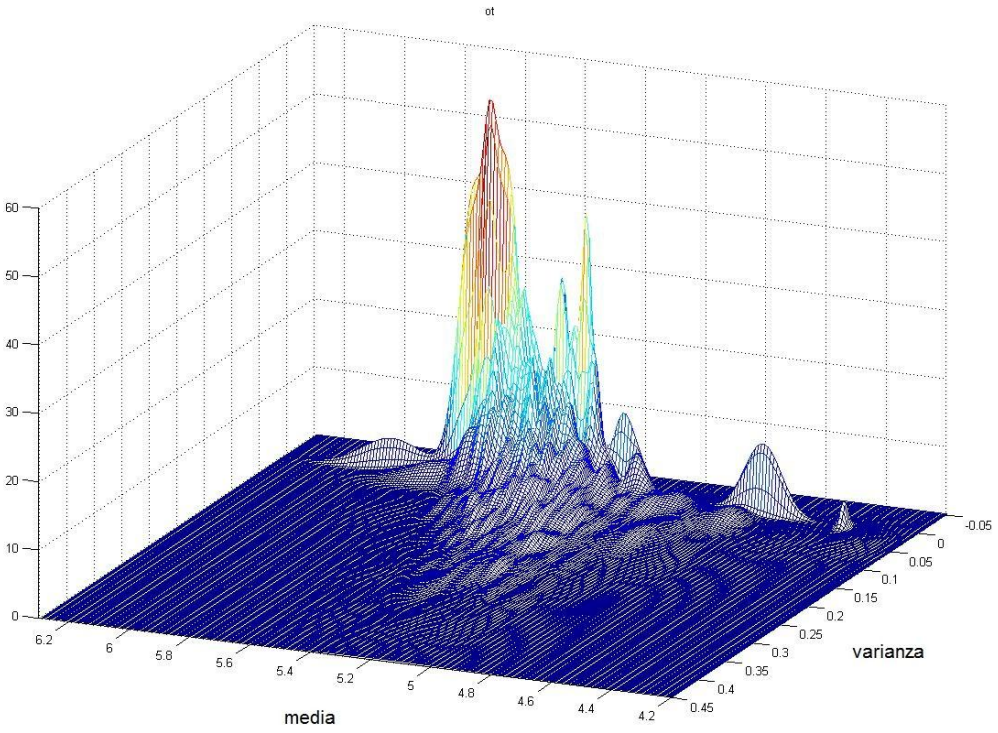


Figura 0-24: GMM adaptando todos los parámetros para *ot*

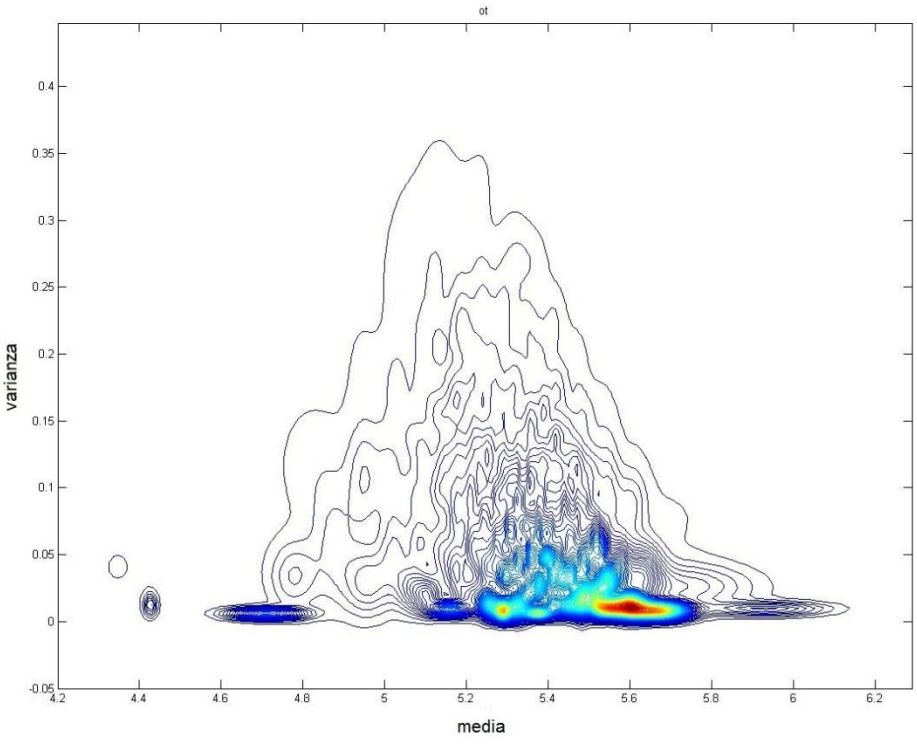


Figura 0-25: Curvas de nivel del GMM adaptando todos los parámetros para *ot*

Similitud de Audio Musical Para Identificación de Versiones (*Covers*)

1. INTRODUCCIÓN

El objetivo de esta práctica es comprender las técnicas utilizadas en la tarea de comparación de versiones de canciones, mediante la extracción de cromagramas a partir de archivos musicales.

Este guión se facilita junto con un paquete de software disponible en la página web del laboratorio LabROSA de la Universidad de Columbia (http://labrosa.ee.columbia.edu/), que puede ser utilizado con fines académicos. Todo ello, junto con los artículos del sistema presentado por dicho laboratorio en distintas ediciones de la evaluación MIREX (disponibles también en Moodle), es suficiente para la realización de la práctica.

Para la realización de esta práctica, se utilizará software desarrollado por LabROSA para la identificación de versiones. Dicho software está implementado en Matlab™, estando disponible bajo licencia Gnu GPL. La base de datos a utilizar, que es un subconjunto de la base de datos denominada COVERS80, es también de dominio público. Esta base de datos está compuesta por 80 canciones, estando cada canción versionada por 2 artistas diferentes (teniendo un total de 160 canciones).

Las canciones de COVERS80 están en formato MP3 (aunque el código de LabROSA también acepta el formato WAV), codificadas a una tasa de 32 Kbps a partir de ficheros WAV mono con 16 KHz de muestreo y ancho de banda limitado a 7 KHz.

El subconjunto utilizado en esta práctica se denomina COVERS10, y consiste en 10 canciones con sus correspondientes versiones extraídas de la COVERS80. Junto a la base de datos COVERS10 se proporcionan 2 listados con todas las canciones que contiene la base de datos (lista “A” y lista “B”). Esto servirá para poder comparar las 10 canciones con sus respectivas versiones. La nomenclatura que se sigue es la siguiente:

Nombre_Canción/nombre_artista+Nombre_Album+01-Nombre_Canción

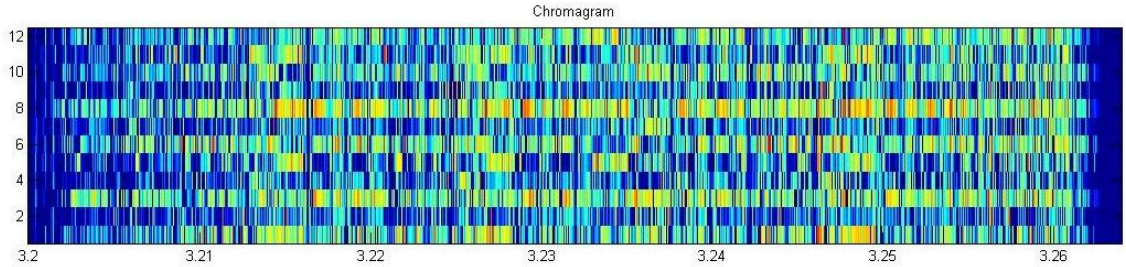
El código a utilizar (disponible en Moodle) se agrupa en el directorio ‘coversongs’, disponiendo de todo lo necesario para el desarrollo de la práctica (incluida la lectura de archivos MP3). La base de datos se encuentra a su vez en el directorio ‘COVERS10’, teniendo dentro de él el directorio ‘covers32k’ (base de datos y listados de canciones). Se recomienda disponer de dichos directorios en el mismo nivel de trabajo. A la hora de ejecutar el software de LabROSA se puede hacer uso de la función `addpath` de Matlab™ para añadir los directorios a la ruta de búsqueda. Para más información acceder a la ayuda de Matlab™.

La evaluación de la práctica consistirá en una prueba individual escrita de 20 minutos de duración en la siguiente sesión de prácticas.

2. DESARROLLO DE LA PRÁCTICA

2.1 CROMAGRAMAS

Un cromagrama es la representación en el tiempo de los coeficientes de energía de cada una de las bandas de un banco de filtros adaptado a las octavas de cada una de las 12 notas (croma). De modo que para la nota Do se empleará un banco de filtros donde exista un filtro en la nota Do de cada octava del espectro de audio que se maneje centrado en la frecuencia de la nota para cada octava.



En el software de LabROSA se proponen 3 métodos diferentes para calcular dichos cromagramas:

- La rutina principal, `chromagram_IF`, funciona de manera similar a un espectrograma, tomando un archivo de audio como entrada y generando una secuencia de tramas cromáticas en tiempos cortos. `chromagram_IF` calcula el espectrograma del archivo de audio, haciendo un seguimiento tonal basado en frecuencias instantáneas. Obtiene el cromagrama final mediante una cuantificación cromática de las frecuencias instantáneas.
- Las otras 2 rutinas, `chromagram_E` y `chromagram_P`, son implementaciones alternativas que asignan la salida de cada filtro cromático en la STFT (Transformada de Fourier a corto plazo) directamente al croma. La función `chromagram_P` sólo utiliza los picos espectrales (peaks).

Dentro del directorio 'coversongs/Auxiliary_data' se dispone de una pieza de piano tocando una escala cromática ('piano-chrom.wav'), y una flauta recorriendo su tesitura (flauta_tesitura.wav). Se pide lanzar el código de LabROSA para convertir dicho archivo musical (en formato WAV) en cromagramas. Para llevar a cabo la lectura de archivos WAV puede hacer uso de la función `wavread` de Matlab™. Para la representación de cromagramas se recomienda la utilización de la función `imagesc`:

```
>> tt = [1:size(C,2)]*cfftlen/4/sr;

>> imagesc(tt,[1:12],20*log10(C+eps));

>> axis xy;

>> caxis(max(caxis)+[-60 0]);
```

Siendo `C` la matriz devuelta (cromagrama) por `chromagram_IF`, `sr` la frecuencia de muestreo del audio y `cfftlen` el valor por defecto de la longitud de la FFT a utilizar (`cfftlen=2048`).

- PREGUNTA 2.1.1. ¿Cuáles son los parámetros de entrada por defecto para las distintas funciones (`chromagram_IF`, `chromagram_E` y `chromagram_P`)?

- PREGUNTA 2.1.2. Comparar los cromagramas obtenidos entre sí y con el audio en sí. Señalar semejanzas y diferencias apreciables.

- PREGUNTA 2.1.3. Evaluar los tiempos de proceso (coste computacional) de cada una de las funciones y hacer una comparación de complejidad. Se recomienda el uso de las funciones `tic` y `toc` para el cálculo de tiempos de ejecución. Para más información consultar la ayuda de Matlab™.

Pruebe también a visualizar el cromagrama de un par de canciones de la base datos. Recuerde que están en formato MP3, por lo que tendrá que utilizar la función `mp3read` (proporcionada en el directorio 'coversongs').

- PREGUNTA 2.1.4. Evaluar los resultados obtenidos y compararlos tanto con su propio audio como con los resultados obtenidos para la escala cromática del piano.

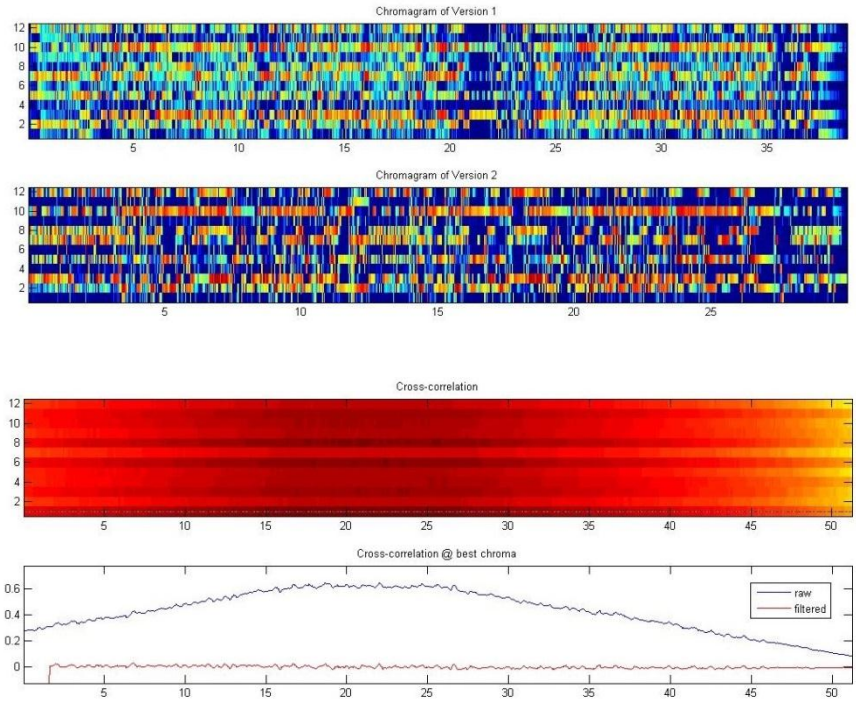
Para poder calcular los cromagramas de grandes cantidades de audio y almacenar toda esa información de forma correcta se adjunta la función `calclistftrs`. Se pide realizar dicho cálculo para la base de datos COVERS10. Los listados de las canciones de esta base de datos se pueden encontrar en la dirección 'COVERS10/covers32k' con los nombres `list1.list` y `list2.list`. Por limitaciones de los equipos a utilizar (laboratorio 6b de la Escuela Politécnica Superior de la UAM), y para acelerar el proceso de lectura de los ficheros, se recomienda utilizar de aquí en adelante la tercera parte de las lecturas de los archivos .mp3, es decir, utilizar 1/3 del vector `d` devuelto por la función `mp3read` (ver función `calclistftrs`). Las conclusiones a extraer en la práctica no se verán afectadas por dicha reducción.

- PREGUNTA 2.1.5. Utilice la función `calclistftrs` sobre la base de datos "COVERS10". Se pide analizar los resultados obtenidos en el directorio destino (matrices de cromagramas) y ver si hay alguna diferencia con los cromagramas obtenidos directamente de las funciones `chromagram_IF`, `chromagram_E` y `chromagram_P`.

- PREGUNTA 2.1.6. Estudiando los diferentes parámetros de entrada de la función `calclistftrs`, ¿cómo se puede alternar entre el uso de las funciones `chromagram_IF`, `chromagram_E` y `chromagram_P`? Evaluar los tiempos de ejecución en función de este parámetro y comprobar si concuerda con los resultados obtenidos en la pregunta 2.1.3.

2.2 COMPARACIÓN ENTRE CROMAGRAMAS

Una vez se tienen todos los cromagramas calculados se procede a la comparación entre ellos para así obtener una medida de similitud de versión, que deberá ser más alta si dos ficheros de audio son versiones de la misma canción, y más baja si lo son de canciones diferentes. Esta comparación se hace mediante una correlación cruzada, aparte de las respectivas normalizaciones y filtrados. Para más información consultar el artículo `Identifying 'Cover Songs' With Chroma Features And Dynamic Programming Beat Tracking` (disponible en Moodle).



La función apropiada para hacer comparaciones entre cromagramas se llama `coverTestLists`, a la que simplemente se tiene que pasar como argumentos las direcciones de los resultados obtenidos con la función `calclistftrs` (esta función ya devuelve dicho listado). La salida principal de la función es una matriz cuadrada `R` donde se enfrentan las 2 listas de canciones con cada una de las puntuaciones obtenidas para cada cruce. Si la puntuación más elevada de una línea coincide en número con la columna (ej.: la puntuación más elevada de la fila 3 corresponde con la columna 3) habremos acertado en nuestra comparación. El caso ideal es que los aciertos se ubiquen en la diagonal principal de la matriz R.

Se pide analizar en detalle la función `coverTestLists` y calcular la correlación y medida de similitud de 2 canciones cualesquiera de la base de datos de manera similar a como se calcula en el software de LabROSA.

- PREGUNTA 2.2.1. Explicar los pasos seguidos hasta llegar al valor numérico de similitud, así como las funciones empleadas. Se pide también la interpretación de cada una de las partes del proceso de comparación (correlación, filtrado paso-alto de la correlación, normalizado, etc.).

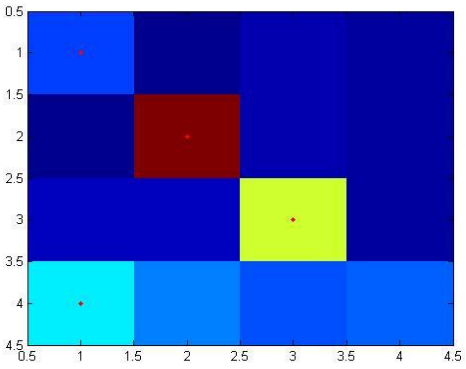
Se pide ejecutar el software de LabROSA para la "COVERS10", y analizar los datos de entrada y salida de la función `coverTestLists`.

- PREGUNTA 2.2.2. ¿Qué información contienen los datos de salida de la función `coverTestLists`?

Para una visualización más gráfica de la matriz `R` se puede recurrir a una matriz de confusión. Se recomienda usar el siguiente código para tal caso:

```
>> [vv,xx] = max(R');

>> ncovers = length(xx);

>> figure; imagesc(R);

>> hold on;

>> plot(xx,[1:ncovers],'.r');

>> hold off;
```

Este código convierte la matriz (`R` en nuestro caso) en una imagen escalada (asignando cada valor de la matriz a los diferentes índices del mapa de colores activo en Matlab™, pasando a ser cada uno de ellos un pixel de la imagen), marcando sobre ella con puntos rojos las puntuaciones más elevadas de cada fila. Ejemplo de resultados para 4 canciones:



Para obtener de forma sencilla el número de aciertos se puede hacer uso del siguiente código:

```
>> ncorr = sum(xx==1:ncovers);
```

- PREGUNTA 2.2.3. Obtenga la matriz de confusión para la base de datos "COVERS10", así como el número de aciertos obtenidos. Interprete los datos obtenidos y razone si son aceptables.

2.3 MEJORAS: DENORMALIZACIÓN

En la implementación que se ha suministrado, el porcentaje de acierto en la tarea de reconocimiento de versiones para la base de datos COVERS80 es bastante bajo, por lo que se propone buscar diferentes mejoras. Las principales mejoras se centran en mejorar el rendimiento de la tarea de comparación de cromagramas, es decir, disponiendo de los mismos cromagramas y cambiando únicamente la manera en que se comparan.

El valor pico de la correlación cruzada entre dos cromagramas tiende a crecer con la longitud de las matrices que implementan dichos cromagramas. En el software antes utilizado se normalizaba cada correlación tal y como se explica en los apartados 4 y 5 del artículo `Identifying 'Cover Songs' With Chroma Features And Dynamic Programming Beat Tracking`[i].

- PREGUNTA 2.3.1. Se pide implementar esta mejora y evaluar los resultados para la base de datos "COVERS10" de la misma forma que se hacía en apartados anteriores (cromagramas obtenidos, matriz de confusión, número de aciertos, etc.).

- PREGUNTA 2.3.2. Comparar estos resultados con los obtenidos en el apartado 2.2 y constatar si hay mejora alguna, y en tal caso determinar el por qué.

Nótese que al ser una modificación exclusiva de una subrutina de la función `coverTestLists` no es necesario el cálculo nuevamente de los cromagramas. Para más información se puede consultar el artículo `The 2007 Labrosa Cover Song Detection System`, en especial el apartado 4.1 (disponible en Moodle).

REFERENCIAS

Daniel P.W. Ellis and Graham E. Poliner: Identifying 'Cover Songs' With Chroma Features And Dynamic Programming Beat Tracking. MIREX Evaluations (2006). Disponible en

Daniel P.W. Ellis and Courtenay V. Cotton: The 2007 Labrosa Cover Song Detection System. MIREX Evaluations (2007).

PRESUPUESTO

1) Ejecución Material

- Compra de ordenador personal (Software incluido)..... ..... 2.000 €
- Material de oficina ..... 200 €
- Total de ejecución material ..... 2.200 €

2) Gastos generales

- 16 % sobre Ejecución Material ..... 352 €

3) Beneficio Industrial

- 6 % sobre Ejecución Material ..... 132 €

4) Honorarios Proyecto

- 1400 horas a 15 € / hora ..... 21.000 €

5) Material fungible

- Gastos de impresión ..... 120 €
- Encuadernación ..... 200 €

6) Subtotal del presupuesto

- Subtotal Presupuesto ..... 24.004 €

7) I.V.A. aplicable

- 21 % Subtotal Presupuesto ..... 5.040,84 €

8) Total presupuesto

- Total Presupuesto ..... 29.044,84 €

Madrid, Septiembre de 2014

El Ingeniero Jefe de Proyecto

Fdo.: Ricardo Landriz Lara
Ingeniero de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un SISTEMA DETECTOR DE TIPOS DE AUDIO BASADO EN CARACTERÍSTICAS MUSICALES. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4 % del presupuesto y la provisional del 2 %.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.