Transcripción de melodías polifónicas mediante redes neuronales dinámicas

Antonio Pertusa Ibáñez

Director: Jose Manuel Iñesta Quereda

Memoria de suficiencia investigadora
Programa de doctorado: Aplicaciones de la informática
Trabajo de informática musical (60455)

Departamento de Lenguajes y Sistemas Informáticos

Universitat d'Alacant
Universidad de Alicante

curso 2002–2003

PREFACIO

La trascripción musical se define como la acción de escuchar una melodía y escribir la notación musical de las notas que la componen. El objetivo del trabajo que nos ocupa es la transcripción musical en un entorno monotímbrico y polifónico, es decir, la detección de notas en melodías en las que sólo suena un instrumento y en las que pueden estar sonando varias notas simultáneamente.

La transcripción automática de señales monofónicas (aquellas en la que sólo puede escucharse una nota simultáneamente) es un problema muy estudiado y algunos de los algoritmos que han sido propuestos operan en tiempo real y son fiables y comercialmente aplicables. En cambio, el problema de la extracción de notas de una señal musical polifónica en formato digital permanece abierto dentro del campo de la investigación. Se han probado distintas técnicas para intentar resolverlo sin haber logrado de momento resultados concluyentes.

La técnica que se propone en este trabajo está basada en redes neuronales dinámicas no recurrentes. Se pretende, más que desarrollar un sistema de detección de notas completo y fiable, poner a prueba la utilidad de las redes neuronales dinámicas no recurrentes para la transcripción musical polifónica.

Estructura de la memoria

Se ofrece a continuación un pequeño resumen de lo que se discutirá en cada uno de los capítulos.

Capítulo 1: Memoria del periodo de docencia en tercer ciclo, correspondiente al programa Aplicaciones de la Informática en el Departamento de Lenguajes y Sistemas Informáticos (DLSI) de la Universidad de Alicante.

Capítulo 2: Presentación preliminar de los problemas que plantea la transcripción musical y descripción de los objetivos del presente trabajo.

Capítulo 3: Repaso al estado actual de la transcripción musical automática.

Capítulo 4: Metodología y descripción del sistema propuesto.

Capítulo 5: Primeros experimentos (Configuración I). Descripción de las señales usadas en las pruebas, ajuste de parámetros y presentación de los resultados.

Capítulo 6: Configuración II: Cambio de parámetros, realización de nuevos experimentos y presentación de los resultados.

Capítulo 7: Discusión y conclusiones de los resultados obtenidos y trabajos futuros.

Los aspectos fundamentales de este trabajo están recogidos en un artículo aceptado para el congreso IAPR - TC3 *International Workshop on Artificial*

Neural Networks in Pattern Recognition (ANNPR), que se celebra en Florencia (Italia) en septiembre de 2003.

Agradecimientos

Vaya ahora mi agradecimiento a Jose Manuel Iñesta y Mikel Forcada por su contribución decisiva a mi formación como investigador, a Juan Antonio Pérez por su inestimable ayuda con el formato del documento y a Gema Ramírez por su asesoramiento lingüístico.

También quiero dar las gracias a mis compañeros de trabajo durante mi estancia como becario de investigación del Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Alicante, a toda la gente que me ha animado y apoyado durante el desarrollo de esta memoria de suficiencia investigadora y a Cristina, por su paciencia y apoyo moral.

ÍNDICE GENERAL

PARTE I

MEMORIA DEL PERIODO DE DOCENCIA (CURSO 2001–2002)

1. PERIODO DE DOCENCIA

1.1. Cómo se escriben y publican trabajos científicos

Profesorado: Mikel L. Forcada Zubizarreta
Créditos: 4
Op/Ob: Optativa
Tipo: Metodológica
Calificación obtenida: Notable

Una de las fases más difíciles del trabajo de un investigador es la de explicar lo que ha hecho en un documento que quiere publicar. Esta asignatura proporciona la oportunidad de reflexionar sobre el proceso de publicación científica y la formación necesaria para publicar trabajos con éxito.

Los objetivos de la asignatura son:

- Comprender la necesidad de métodos especializados de documentación y comunicación para el trabajo científico y tecnológico.
- Concienciarse de la importancia de los métodos de comunicación y documentación en el avance de la ciencia y la técnica.
- Conocer y asumir los mecanismos básicos de producción de documentos de naturaleza científica, así como los mecanismos de publicación de trabajos científicos.
- Aprender a preparar y realizar presentaciones orales de trabajos científicos.
- Conocer los mecanismos básicos de obtención de documentos científicos.

Para ello se ha hecho un recorrido sobre los distintos tipos de documentos científicos, sus diferencias y la organización de la información en cada uno de ellos, así como al tipo de publicación al que van asociados. También se proporcionó la estructura básica que se debe seguir para escribir los diversos tipos de documentos científicos.

Como trabajo práctico, a lo largo de la asignatura se fueron realizando simulacros de presentaciones y de posters.

Como bibliografía básica destaca el libro de R.A. Day [13].

Esta asignatura me parece imprescindible, ya que ofrece información sobre la labor investigadora que normalmente se desconoce, y no se obtiene fácilmente. También es muy importante porque ofrece una nueva perspectiva sobre el mundo de la publicación y sobre lo importante que puede llegar a ser el hecho de publicar en una determinada revista.

1.2. Traducció automàtica: fonaments i aplicacions

Profesorado: Mikel L. Forcada Zubizarreta
Créditos: 6
Op/Ob: Optativa
Tipo: Fundamental
Calificación obtenida: Sobresaliente

Esta asignatura es una introducción a la traducción automática. Por tanto ofrece una visión general de los problemas que se presentan al abordar esta tarea, así como sus posibles soluciones.

Los objetivos que de la asignatura son:

- Comprender los conceptos de traducción automática y semiautomática y ser conscientes de sus posibles aplicaciones y de los problemas que comporta esta tarea.
- Aprender los conceptos teóricos básicos sobre los que se basan las soluciones de los problemas planteados y sus aspectos técnicos: gramáticas, analizadores léxicos, sintácticos y semánticos, ambigüedad, etc.
- Conocer el estado de la cuestión y las soluciones más frecuentes en traducción automática, enfatizando aquellas dirigidas al uso en ordenadores personales: sistemas de traducción directa, de transferencia sintáctica, de transferencia semática superficial; sistemas basados en representaciones semánticas; sistemas de *interlingua*; sistemas estadísticos e híbridos.
- Ser conscientes de la complejidad intrínseca de la traducción automática y de las causas de esta complejidad.

Para conseguir alcanzar estos objetivos, en esta asignatura se realiza una introducción para saber en qué consiste realmente la traducción automática. A continuación se estudian sus posibles aplicaciones, el grado de dificultad de las tareas a realizar, y cuál es es su aplicabilidad real: asimilación y diseminación de información, preedición (adaptación) y postedición (revisión) de textos, utilización de lenguajes controlados, etc.

Posteriormente se describen las estrategias básicas utilizadas por los programas de traducción automática, como son los sistemas de interlingua o los sistemas de transferencia.

Por último, se presentan algunas nociones básicas sobre cómo evaluar un sistema de traducción automática, y se evalúan diversos traductores existentes del inglés al castellano.

La bibliografía básica de esta asignatura son las referencias [26] y [1].

Esta asignatura ha ampliado mis conocimientos sobre traducción automática, al mismo tiempo que me ha ayudado en mi trabajo como becario de investigación en el proyecto de traducción automática de Universia y en el proyecto interNOSTRUM.

1.3. Computación avanzada para música por ordenador

Profesorado: Jose Manuel Iñesta Quereda
Créditos: 3
Op/Ob: Optativa
Tipo: Fundamental
Calificación obtenida: Sobresaliente

Esta asignatura es la que me ha permitido obtener las bases musicales y matemáticas necesarias para desarrollar este trabajo de suficiencia investigadora.

Los métodos de reconocimiento de formas, aprendizaje computacional, etc. han invadido en los últimos tiempos el terreno de la informática musical. Hay tres grandes áreas dentro de este campo: el desarrollo de programas relacionados con la composición automática o la asistencia a la misma, la modelización de procesos cognitivos asociados con la escucha de la música (detección de notas, análisis melódicos y armónicos, etc.) y la simulación de la interpretación de partituras (expresión, armonización, acompañamiento, etc.)

Para comprender y conocer mejor estas tres áreas, la asignatura propone en su temario:

- Comprender el proceso de la percepción musical.
- Adquirir nociones sobre una señal el dominio de la frecuencia: espectros, armónicos, etc.
- Comprender las señales de audio en formato digital.
- Aprender a procesar señales musicales.
- Conocer el control digital de dispositivos y la secuenciación musical.
- Adquirir nociones básicas sobre la síntesis de sonido.
- Aprender el lenguaje de programación CSound.
- Aplicar conceptos de inteligencia artificial a la música.

A lo largo de la memoria del trabajo de investigación tutelado se hace una amplia referencia a la bibliografía relacionada con la informática musical.

En lo que se refiere a las prácticas de la asignatura, se ha hecho un estudio de la viabilidad de un secuenciador de libre distribución (Anvil Studio[1]) para la docencia de informática musical y se ha realizado una adaptación práctica de este programa para lograr este objetivo.

También se ha hecho un análisis y estudio de la aplicabilidad de FFTW (*The Fastest Fourier Transform in the West*[2]), que es una colección de rutinas escritas en C para calcular la transformada discreta de Fourier (DFT) en una o más dimensiones, trabajando con datos reales o complejos.

[1] `http://www.anvilstudio.com`
[2] `http://www.fftw.org`

1.4. Uso y diseño de bases de datos multidimensionales y datawarehouse

Profesorado: Juan Carlos Trujillo Mondéjar
Créditos: 3
Op/Ob: Optativa
Tipo: Fundamental
Calificación obtenida: Aprobado

En esta asignatura se tratan los almacenes de datos (Data Warehouses), bases de datos multidimensionales y herramientas OLAP (On-Line Analitycal Processing). Todos ellos constituyen un conjunto de técnicas, bases de datos, métodos y herramientas para facilitar al analista de la información el acceso a una gran cantidad de información histórica que le facilite la adopción de decisiones estratégicas.

El objetivo principal de esta asignatura es el de introducción al diseño y explotación de los almacenes de datos. En una primera parte se introducen los aspectos teóricos imprescindibles que se manejan en estos sistemas. A continuación se proponen casos de estudio para poder enfrentarse a problemas reales que surgen al implantar almacenes de datos. Una vez diseñado el almacén de datos, se intenta la familiarización con el análisis OLAP a través de algunas de las herramientas OLAP más extendidas del mercado. Con todo ello, se pretende adquirir una gran destreza en esta nueva tendencia de bases de datos.

Los principales objetivos de esta asignatura son:

- Introducir los principios y características fundamentales de los sistemas de apoyo a la decisión.
- Presentar una arquitectura general de almacenes de datos y conocer cada uno de sus componentes principales.
- Realizar el diseño conceptual y lógico de los almacenes de datos
- Aprender distintas técnicas de explotación de almacenes de datos a través de herramientas OLAP.
- Introducir una perspectiva amplia de la actualidad en cuestiones de investigación sobre los almacenes de datos y modelado multidimensional
- Introducir los principios básicos de una arquitectura de almacenes de datos en la web.

Todavía no se disponen de modelos o herramientas estándares (como sucede con las bases de datos relacionales) para afrontar todas las fases de diseño de un almacén de datos. En este sentido, en estos últimos años han surgido varias propuestas realizadas por la comunidad académica. Por ello, una parte del temario de la asignatura consiste en presentar estas propuestas, valorarlas y estudiar cómo utilizarlas conjuntamente con las herramientas comerciales para afrontar un proyecto real de almacenes de datos.

Como bibliografía básica cabe citar el libro de R. Kimball [32] y el de W. Inmon [28].

Como trabajo práctico de la asignatura se ha diseñado una base de datos multidimensional de un club de campo y, una vez poblada con datos, se ha llevado a cabo la explotación de dicha base de datos a través de herramientas de consulta OLAP.

1.5. Herramientas para el tratamiento de documentos estructurados

Profesorado: Rafael C. Carrasco Jiménez
Créditos: 4
Op/Ob: Optativa
Tipo: Fundamental
Calificación obtenida: Notable

El principal objetivo de esta asignatura es conocer las principales herramientas para tratar documentos estructurados. El desarrollo actual de las bibliotecas digitales (y de internet) permite consultar grandes volúmenes de información. Una forma de que esta información resulte más útil es su marcado, idea que ha originado la definición de técnicas y lenguajes de marcado (como el popular HTML o los estándares SGML y XML) y que constituye, hoy en día, uno de los campos de desarrollo más activos. En esta asignatura se explica cómo se utilizan y definen lenguajes de marcado así como nuevas técnicas de búsqueda, de transformación y de recuperación de la información.

Los objetivos de la asignatura son:

- Aprender la estructura de XML y usar `emacs` para editar ficheros en este formato.
- Conocer TEI (*Text Encoding Initiative*), un vocabulario de marcado para textos.
- Introducir una perspectiva de RELAX, Bibtex, XSLT, XML Schema, etc.
- Relacionar XML y bases de datos.
- Conocer la caracterización matemática de XML y los autómatas de árboles para XML.
- Aprender la simplificación y especialización de esquemas.
- Conocer distintos indexadores y buscadores para XML.

Como bibliografía básica, cabe citar el libro de Elliotte R. Harold [20].

Como trabajo práctico se desarrolló una base de datos musical en formato XML que contenía información sobre melodías (notas, cambios de tempo, número de pistas, información sobre polifonía, metadatos, etc.).

Los conceptos que he adquirido sobre documentos estructurados, además de haberlos aplicado en el extenso campo de la informática musical, me han ayudado en mi trabajo como becario de investigación.

PARTE II

MEMORIA DEL TRABAJO DE INVESTIGACIÓN TUTELADO

2. INTRODUCCIÓN

2.1. La transcripción automática en la informática musical

La trascripción musical se define como el acto de escuchar una melodía y escribir la notación musical de las notas que la componen [47]. En otras palabras, significa la transformación de una señal acústica en una representación simbólica que contiene las notas (con su altura y duración) y en la que se separan los distintos instrumentos empleados.

La altura de un sonido (en inglés, *pitch*) está principalmente relacionada con la frecuencia de los sonidos periódicos, e indica si un sonido es más o menos agudo. Según su definición ANSI de 1994, la altura es el atributo de la sensación auditiva en virtud de la cual los sonidos pueden ser ordenados en una escala de grave a agudo. La altura depende principalmente del contenido en frecuencias del estímulo sonoro, pero también depende de la presión del sonido y de la forma de onda del estímulo.

La percepción humana de la altura es un fenómeno complejo [22]. El oído humano puede detectar notas musicales incluso con la presencia de ruido. También podemos seguir varias notas simultáneamente y detectar desviaciones de altura leves pero expresivas (vibratos, intervalos microtonales,...). Los mecanismos involucrados en la percepción de la altura musical no están totalmente comprendidos y de ahí la falta de modelos computacionales para emular estos procesos.

Por lo general, una persona sin educación musical no es capaz de transcribir música. Incluso un músico tiene dificultades para transcribir música polifónica, es decir, aquella en la que pueden escucharse simultáneamente varias notas. Cuanto más rica es la complejidad polifónica de una composición musical, mayor es la experiencia requerida en estilo musical y teoría musical. Sin embargo, los músicos experimentados son capaces de reconocer melodías polifónicas con mayor precisión que los sistemas actuales de transcripción automática (computacional). La transcripción automática de señales monofónicas es un problema muy estudiado y algunos de los algoritmos que han sido propuestos operan en tiempo real y son fiables y comercialmente aplicables.

En cambio, el problema de la extracción de notas de una señal musical polifónica en formato digital permanece abierto dentro del campo de la investigación. Hasta el momento se han probado distintas técnicas para intentar resolverlo sin haber logrado de momento resultados concluyentes. Entre esas técnicas están las conexionistas. La tesis de Klapuri de 1998 [33] ofrece un excelente estudio sobre el estado de la cuestión de la transcripción musical polifónica.

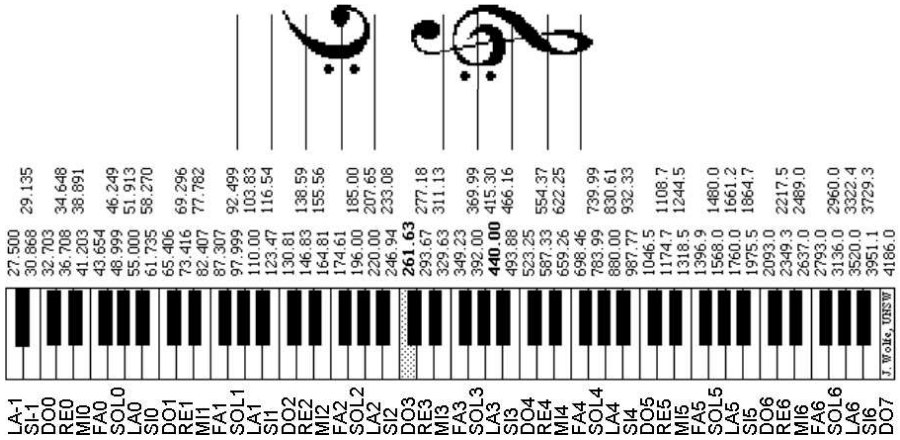

Figura 2.1: En la línea superior se indican las frecuencias de las notas en la escala bien temperada, con sus posiciones en las partituras y en el teclado del piano. Bajo éste se muestran las notas desde La_{-1} hasta Do_7.

El objetivo del trabajo que nos ocupa es la transcripción musical en un entorno monotímbrico y polifónico, es decir, la detección de notas en melodías[1] en las que sólo suena un instrumento y en las que pueden estar sonando acordes[2].

Toda nota musical afinada tiene una altura, que se corresponde con su frecuencia fundamental. La escala occidental moderna, introducida por J. S. Bach en su estudio "El clave bien temperado", usa unas 100 alturas con frecuencias que van aproximadamente desde 30 Hz hasta 4.5 kHz. La relación entre las frecuencias de las notas vecinas (entre Do-Do♯ o Mi-Fa) es de $2^{1/12}$, así que las notas que hacen un intervalo de una octava (12 semitonos[3]) tienen una relación de frecuencia de $(2^{1/12})^{12} = 2$, como se puede observar en la figura 2.1. También hay otras escalas, pero se usan muy raramente.

Lo que hace que éste sea un problema tan difícil de resolver es la ambigüedad presente. La condición de que un armónico[4] p de una nota R coincida con un armónico q de otra nota S es $pf_0^R = qf_0^S$, donde $(p, q) \geq 1$ son números enteros y f_0^R y f_0^S son las frecuencias fundamentales de las notas R y S respectivamente. Ambos lados de la ecuación representan las frecuencias de los armónicos parciales. A partir de esta ecuación, podemos deducir la relación entre dos notas para que coincidan sus armónicos:

$$f_0^R = \frac{q}{p} f_0^S \tag{2.1}$$

[1] Sólo vamos a considerar fuentes de sonido afinadas, es decir, aquellas que producen una altura musical, dejando aparte las producidas a partir de ruido aleatorio y las fuentes de sonido que sean altamente inarmónicas.

[2] Se produce un acorde cuando hay más de una nota sonando simultáneamente.

[3] Un semitono corresponde con una nota. En una octava hay doce notas.

[4] Los armónicos son frecuencias del espectro múltiplos de una frecuencia base a la que llamamos frecuencia fundamental, que define la altura de una nota.

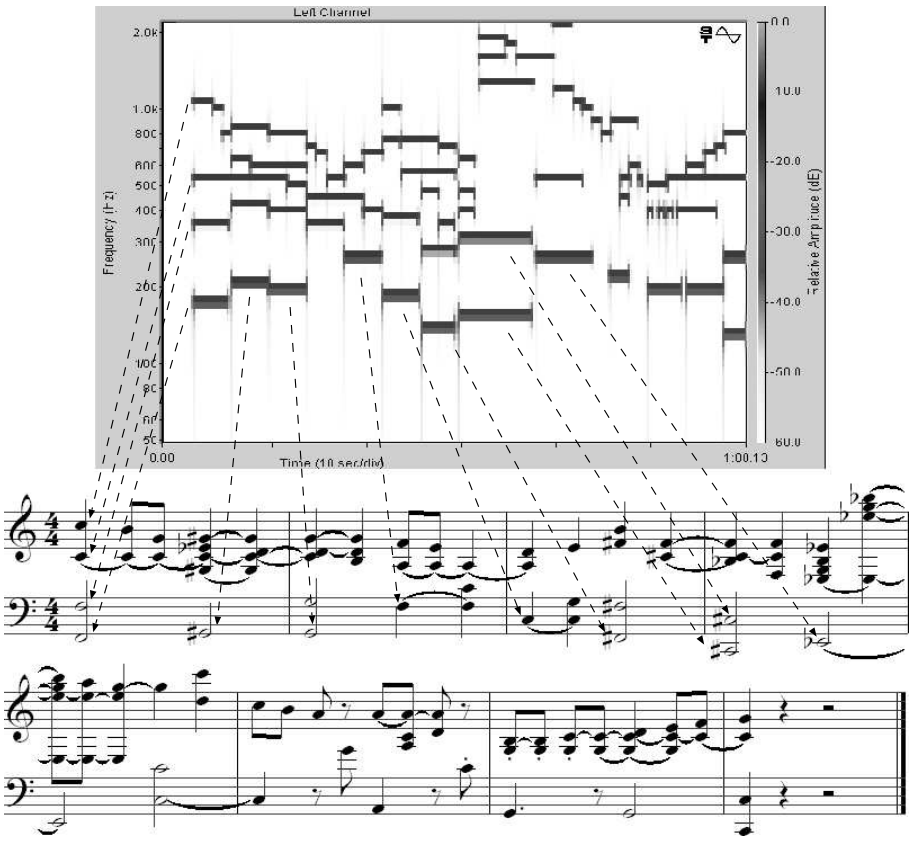

Figura 2.2: Espectrograma de una melodía generada con una onda sinusoidal y su partitura correspondiente. Se muestran las correspondencias entre algunas bandas espectrales y las notas que producen. Nótese que las bandas están en una escala logarítmica, y centradas en las frecuencias de las notas musicales.

Esto implica que los armónicos de dos notas coinciden con determinadas relaciones de distancia entre dichas notas. El hecho de que en música estas coincidencias entre armónicos sean especialmente comunes (pues en ello se basa la armonía) hace más complicado nuestro objetivo de separación de las notas.

La figura 2.2 muestra las correspondencias entre una partitura y su espectro generado con una onda sinusoidal cuyo patrón espectral[5] sólo contiene un armónico, que coincide con su frecuencia fundamental. En este caso la transcripción es bastante sencilla, pero cuanto más armónicos tiene el patrón espectral del timbre utilizado más complicada es la transcripción. Esto se debe a que en algunos casos coinciden armónicos pertenecientes a distintas frecuencias fundamentales.

[5]El patrón espectral de una señal $s(t)$ es la distribución de energía que se encuentra cada una de las frecuencias parciales[6] que componen su espectro, $S(f)$. Este patrón es el parámetro más importante que caracteriza un instrumento musical; es decir, todo instrumento musical tiene asociado un patrón espectral característico, según lo que en lenguaje musical se denomina "timbre".

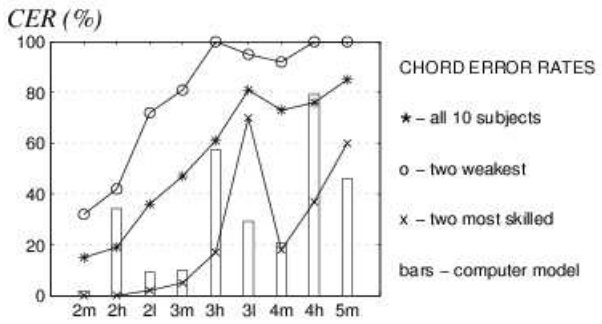

Figura 2.3: Figura extraída de [36]. Porcentaje de error cometido en acordes comparando transcripción manual con el sistema de transcripción automática de Klapuri. El número representa la polifonía, y la letra el rango de altura al que pertenece el acorde (l: bajo, m: medio, h: alto)

En el año 2000, Klapuri y Holm hicieron un estudio sobre la calidad de la transcripción polifónica humana, comparándola con los resultados de su sistema de transcripción automática [39]. Las pruebas consistieron en un total de 200 acordes de distinta polifonía generados con un piano sintetizado[7]. Los resultados se clasificaron en tres rangos de altura: acordes bajos (33 Hz – 130 Hz), medios (130 Hz – 520 Hz) y altos (520 Hz – 2100 Hz). La tarea de las personas que participaron en el experimento era escribir los intervalos musicales, es decir, las relaciones entre alturas que presentaba cada acorde. No se preguntaba por las notas en concreto, sólo por su relación de altura. En las pruebas participaron diez músicos experimentados. Dos de ellos destacaban especialmente y otros dos eran principiantes. La figura 2.3 muestra el porcentaje de error cometido. Se contabilizaron como errores los acordes en los que se producía algún fallo en la detección de cualquiera de sus notas. Como se puede observar, los principales problemas se producen con acordes de notas muy altas o muy bajas. También se puede ver que la detección polifónica no es una tarea trivial para un músico.

2.2. Objetivos

El principal objetivo del presente trabajo es la transcripción musical polifónica automática, es decir, la detección por medio de un sistema informático de notas en melodías en las que sólo suena un instrumento y en las que pueden estar sonando acordes.

Como se verá posteriormente en el análisis sobre el estado de la transcripción, se han usado una gran variedad de técnicas para afrontar este problema, pero de momento ninguna lo ha resuelto completamente.

La mayoría de técnicas empleadas que han ofrecido resultados prometedores se mueven en el dominio de la frecuencia, analizando los parciales armónicos para obtener a partir de ellos las frecuencias fundamentales. Muchas de estas técnicas usan modelos auditivos, que intentan comprender la percepción humana de la altura y construir un modelo computacional de la misma. También se han

[7] Las señales sintetizadas son las generadas artificialmente, de manera que tienen un número de armónicos controlado.

propuesto unas pocas técnicas basadas en redes neuronales, aplicándolas a la salida de modelos basados en la percepción humana [45].

No está totalmente claro que el espectro de un sonido contenga toda la información relativa a la alturas que lo forman, ya que éste tiene limitaciones para capturar todos los datos que contiene su correspondiente forma de onda. En cualquier caso, como se verá más adelante, las técnicas que han obtenido mejores resultados en transcripción polifónica se han basado en el análisis espectral de la señal, mientras que las técnicas de análisis en el dominio del tiempo han quedado prácticamente descartadas.

En este trabajo usaremos espectrogramas para llevar a cabo la transcripción. Un espectrograma es una representación de la evolución del espectro de frecuencias de una señal sonora en función del tiempo. Nos permite observar la componente de amplitud, es decir, la componente real, del resultado de aplicar a una señal la transformada de Fourier a corto plazo (*Short-Time Fourier Transform*, STFT). Para generar un espectrograma, la señal se divide en pequeñas ventanas temporales, cada una de las cuales es analizada mediante la transformada de Fourier.

No vamos a modificar el espectro de una señal para intentar adaptarlo al modelo de percepción humana, sino que lo vamos a introducir directamente en una red neuronal para ver si es capaz de detectar las notas sin ningún preproceso. Aplicar cambios a un espectrograma puede desvirtuar o destacar la información contenida en él. Por lo tanto en la técnica que proponemos en este trabajo se van a introducir espectrogramas directamente en la red para que ésta intente encontrar de manera automática las relaciones entre alturas presentes en el espectro de una señal.

Se pretende abordar el problema mediante la identificación del patrón espectral de un instrumento, usando para ello redes neuronales con retardos temporales (*Time-Delay Neural Networks*, TDNN), entrenadas con espectrogramas de melodías polifónicas generadas con el instrumento objetivo. Las redes TDNN se han usado anteriormente en problemas de reconocimiento de voz [23, 42] y muy recientemente en problemas de transcripción musical [45].

La dificultad del problema depende directamente del número de notas simultáneas y del número de armónicos que posee el espectro del timbre a tratar. Cuanto mayor riqueza armónica, mayor solapamiento de armónicos. Una onda sinusoidal es un sonido completamente simple, ya que su espectro tiene toda la energía concentrada en la amplitud del parcial correspondiente a la frecuencia fundamental f_0. Una onda cuadrada es un sonido que contiene sólo los armónicos impares, con amplitudes que decaen con una tasa de $1/p$, donde p es el número del parcial (es decir, fp). Comenzaremos por la detección de notas en melodías generadas con sonidos de poca riqueza armónica, y posteriormente emplearemos timbres más complejos.

2.3. Red neuronal de retardos temporales

Generalmente se considera a las redes TDNN como redes dinámicas no recurrentes [25], aunque en esencia son redes estáticas que recorren series temporales. Este tipo de redes son capaces de modelar sistemas donde la salida $y[t]$ tiene

una dependencia no lineal de un intervalo de tiempo limitado de la entrada $u[t]$:

$$y(t) = F\left[u(t-m), ..., u(t-1), u(t), u(t+1), ..., u(t+n)\right] \tag{2.2}$$

Con este tipo de red podemos procesar datos de series temporales como una colección de patrones estáticos de entrada/salida relacionados en función del tamaño de la ventana de entrada. Debido a la ausencia de realimentación, su arquitectura se corresponde con la de un perceptrón multicapa y podemos entrenarla usando un algoritmo estándar de retropropagación del error (*back-propagation* [59]).

Construiremos pares de entrada/salida para la red formados por el espectro del sonido producido por una fuente para muestras temporales distintas alrededor de un instante dado t_i. La entrada es $\{S(f, t_{i+j})\}$ para $j \in [-m, +n]$, siendo m y n el número de ventanas consideradas antes y después del instante central t_i. La salida consiste en una codificación del conjunto de notas posibles $\nu(t_i)$ que están activas en ese instante t_i y que producen esos espectros usados para la entrada. En otras palabras, la entrada de la red consiste en espectrogramas y la salida en la representación simbólica de la melodía que producen esos espectrogramas.

Para obtener estos datos usaremos ficheros de audio digital y ficheros en formato MIDI [2] estándar. Ambos necesitan ser transformados previamente. A partir de un fichero de audio digital obtendremos su espectrograma mediante la transformada de Fourier a corto plazo (STFT). Un programa, Spectralab[8], es el encargado de realizar la STFT. A partir del fichero MIDI de la melodía correspondiente obtendremos los patrones de notas en notación simbólica.

Tras la fase de aprendizaje del patrón espectral, se espera que la red sea capaz de detectar notas a partir de la digitalización de una melodía generada por la misma fuente de sonido usada en el entrenamiento. Se desea que el método sea robusto ante patrones solapados (polifonía) e, idealmente, ante patrones producidos por instrumentos de timbres similares.

2.4. Fuentes bibliográficas relacionadas

Los artículos más relevantes relacionados con el amplio campo de la informática musical se publican en *Journal of the Acoustical Society of America*, *Computer Music Journal*, *IEEE Transactions on Acoustics Speech and Signal Processing*, *Journal of New Music Research*, *Journal of the Audio Engineering Society* y *Music Perception*.

También se publican artículos relacionados con la informática musical en los libros de actas de congresos como *International Computer Music Conference*, *International Conference of Music and Artificial Intelligence* e *International Symposium on Music Information Retrieval* entre otros.

A nivel internacional, los centros de investigación más destacados en este campo son el CCRMA (*Center for Computer Research in Musics and Acoustics*) de la Universidad de Stanford, el IRCAM (*Institut Recherche Coordination Acoustique Musique*), y el *Medialab* del MIT (*Massachussets Institute of Technology*). En España, destaca el IUA (*Institut Universitari de l'Audiovisual*) de

[8] `http://www.soundtechnology.com`

la *Universitat Pompeu Fabra* y el IIIA (Instituto de Investigación en Inteligencia Artificial) del CSIC.

3. ESTADO DE LA CUESTIÓN

La aplicaciones potenciales de un sistema de transcripción automática son numerosas. Músicos y compositores podrían utilizar estas herramientas para analizar de manera eficiente composiciones que se encuentran únicamente en formato de grabación acústica. La representación simbólica (en una partitura) de una señal acústica permite mezclarla de manera flexible, editarla y codificarla selectivamente. La transcripción facilitaría desde la perspectiva musical un análisis psicológico de la composición, por no hablar de su utilidad para conservar archivos acústicos, para su análisis estadístico, para la detección de plagios y para aplicaciones pedagógicas y educativas.

Como queda dicho, las aplicaciones de un sistema de transcripción automática se pueden comparar con las de un sistema de reconocimiento de voz. Ambas tareas son complicadas pero el masivo interés comercial por codificar y reconocer la voz ha sido mucho mayor que el suscitado por la investigación en reconocimiento de música. Y es por esto que los autores de sistemas de transcripción han estado, por lo general, aislados dentro de un grupo de investigación dedicado a varios aspectos de la música pero desde una perspectiva de aplicación más comercial.

La transcripción musical automática está relacionada con diversas ramas de la ciencia:

- Una de sus fuentes principales de conocimiento es la psicoacústica, que estudia la percepción humana del sonido, incluyendo la música y la voz.
- El análisis computacional de la escena auditiva es un área de investigación más amplia que la nuestra. Estudia el análisis de la información acústica proveniente del entorno físico y la interpretación de los distintos fenómenos que en él ocurren.
- El análisis de sonidos de instrumentos musicales ayuda a la transcripción musical dado que la música está compuesta por varios tipos de sonidos provenientes de instrumentos y de voces humanas.
- El procesamiento digital de señales es una rama de las ciencias de la información que, junto con la psicoacústica, tiene un papel destacado en este trabajo. Está relacionada con la representación digital de las señales y el empleo de los ordenadores para analizar, modificar o extraer información de las mismas [27].

3.1. Primeros sistemas de transcripción

Los primeros intentos de transcripción polifónica se remontan a los setenta, cuando Moorer construyó un sistema para transcribir duos, es decir, composiciones a dos voces [53]. Su sistema sufría fuertes limitaciones en cuanto a la

relación de frecuencias permitidas entre dos notas simultáneas y en cuanto al rango de notas utilizadas. A principios de los 80, un grupo de investigadores de Stanford continuó el trabajo de Moorer [12, 11]. Maher hizo a continuación un estudio más pormenorizado [43, 44]. Sin embargo, en aquel momento la polifonía se limitaba a dos voces y el rango de frecuencias básicas para cada voz estaba a su vez restringido.

A finales de los ochenta, la Universidad de Osaka en Japón inició un proyecto que tenía como objetivo la extracción de sentimientos de las señales musicales y la construcción de un robot que pudiera responder a la música de la misma manera en que lo hace un humano [31]. Durante el proyecto se diseñaron dos sistemas de transcripción. Uno de ellos transcribía canciones populares japonesas. El segundo transcribió composiciones polifónicas para piano, guitarra y *shamisen*[1]. La polifonía de este sistema abarcó hasta cinco voces simultáneas pero a costa de permitir más errores en la salida.

En 1993 Hawley publicó una tesis sobre el análisis computacional de la escena auditiva y se enfrentó también con el problema de la transcripción polifónica de composiciones para piano [21]. No nos ha sido posible obtener su tesis pero, según Martin [46], el sistema de Hawley llegó a tener mucho éxito.

Douglas Nunn trabaja en transcripción en la Universidad de Durham en el Reino Unido. Su sistema de transcripción se caracteriza principalmente por un acercamiento heurístico al procesamiento de la señal y ha sido aplicado a señales sintetizadas que abarcan incluso hasta ocho voces de órgano simultáneas [56]. Sin embargo, el objetivo de Nunn era la similitud perceptual entre el original y las piezas transcritas, de manera que su sistema era bastante permisivo con los errores [33].

Uno de los avances significativos en la historia de la trascripción automática se hizo cuando un grupo de investigadores de la Universidad de Tokio publicó su sistema de transcripción que incluía nuevas técnicas [30]. Fueron los primeros en detallar y utilizar de manera precisa reglas de separación auditiva, es decir, indicaciones auditivas que intentan tanto la fusión como la segregación de componentes de frecuencia simultánea en una señal. Posteriormente introdujeron el procesamiento basado en modelos tonales (utilizando información sobre sonidos de los instrumentos en el procesamiento) y propusieron un algoritmo para modelar tonos automáticamente. Este algoritmo consiste en la extracción automática de modelos tonales a partir de la señal analizada. En 1995, mejoraron de nuevo su sistema [29] al emplear la llamada arquitectura de pizarra (*blackboard architecture*), que parece favorecer el proceso de transcripción. Una excelente introducción a los sistemas de pizarra puede encontrarse en [54]. Dicha arquitectura usaba una red bayesiana de Pearl [57] para propagar la información a través del sistema.

Otro sistema de transcripción más reciente, el de Martin (del MIT) usa también una arquitectura de pizarra [48]. Se puso mucho esfuerzo en la implementación de la estructura pero el sistema no trabajaba con conocimiento musical de alto nivel en la magnitud en que lo hacía el de Kashino (de la Universidad de Tokio), ni tampoco con redes probabilísticas de propagación de la información. Tampoco extrae modelos tonales automáticamente. Aún así su

[1] Un shamisen es un instrumento de cuerda de Japón, de aspecto parecido al banjo.

sistema, junto con el de Kashino, representa la situación actual reciente de la transcripción musical. Posteriormente Martin mejoró su sistema añadiendo un *front-end* más orientado a la percepción y que empleaba correlogramas (introducidos por Slaney y Lyon en 1993 [60]) en el análisis de la señal [46].

3.2. Trabajos relacionados

Hasta ahora hemos hablado de sistemas de transcripción que se han utilizado para transcribir música polifónica compuesta por sonidos armónicos (sin instrumentos de percusión no afinados). Como se ha mencionado anteriormente, hay otros campos de la ciencia relacionados con la transcripción musical. Exponemos a continuación algunos de los trabajos de investigación más relevantes.

Dos excelentes fuentes de información en el campo de la psicoacústica son [6] y [52]. El libro de Albert Bregman, *Auditory Scene Analysis: the Perceptual Organization of Sound* [6] contiene los resultados de tres décadas dedicadas a la investigación de este psicólogo experimental, y ha recibido muchas citas en las ramas de la informática relacionadas con la percepción auditiva. La percepción musical también es mencionada en dicho libro. Otro libro en el que se trata la percepción auditiva, algo más reciente y menos conocido, es *Hearing: Handbook of perception and Cognition* [52], editado por Brian Moore. Ambos son magníficas fuentes de información psicoacústica para el diseño de un sistema de transcripción automática.

El análisis computacional de la escena auditiva (ACEA) se refiere al análisis computacional de la información acústica proveniente de un ambiente físico y a la interpretación de las numerosas y distintas situaciones que en él se dan. En 1991, Mellinger [49] preparó una revisión de los estudios sobre psicoacústica y neuropsicología relacionados con el análisis de la escena auditiva humana. No implementó ningún modelo informático completo del sistema auditivo pero examinó computacionalmente los modelos existentes utilizando señales musicales como material de prueba. El trabajo de Ellis [16] se ha constituido como un referente en la investigación en el ACEA. Ellis evaluó su modelo computacional y obtuvo importantes similitudes entre las situaciones detectadas por la máquina y por los oyentes humanos.

3.3. Métodos de transcripción automática

3.3.1. Métodos en el dominio del tiempo

Método basado en la autocorrelación. La función de autocorrelación es particularmente útil para identificar periodos escondidos en la señal. Se han propuesto varios detectores de frecuencia fundamental basados en la autocorrelación. Por ejemplo, en 1991 Brown publicó los resultados de un estudio en el que la altura de sonidos instrumentales se determinaba mediante la autocorrelación [9]. Probó con autocorrelación convencional y con autocorrelación restringida (*narrowed autocorrelation*) que da como resultado una mejor resolución del periodo a cambio de un tiempo de resolución mayor [8].

El objetivo del estudio era determinar si la correlación se adapta bien a la detección de la frecuencia fundamental de señales musicales y bajo qué condiciones es más ventajosa la autocorrelación restringida. Los detalles relacionados

con el cálculo de la función de autocorrelación y la selección del mejor periodo obtenido de ésta se encuentran en el artículo de Brown [9]. Los resultados eran buenos para señales monofónicas, pero la autocorrelación no es muy recomendable para señales de música polifónica. Así lo confirman Doval et al. [14] y Rabiner [58] cuando apuntan que la autocorrelación no es apropiada cuando se trata con un rango tonal amplio y con una variedad de espectros. Por todo esto, consideramos que los métodos de autocorrelación son más adecuados para el discurso oral que para las señales musicales.

Métodos basados en la forma de la onda. Algunos detectores de altura se basan en mediciones de picos y valles de la onda en el dominio del tiempo, o en mediciones basadas en los cruces por cero (*zero–crossing*), es decir, detectando lugares donde la forma de onda tiene amplitud cero. Por ejemplo, el algoritmo de Miller detecta primero ciclos de excursión (*excursion cycles*) en la onda observando los intervalos entre los principales cruces por cero [50]. Un ciclo de excursión es la sección de la forma de onda que hay entre dos cruces por cero consecutivos. El resto del algoritmo trata de identificar los principales ciclos de excursión, es decir, aquellos que corresponden a auténticos periodos de tonos. El algoritmo parece tener buenos resultados aunque no tanto como otros métodos. Según Klapuri, en la actualidad este trabajo tiene un valor meramente histórico [33].

3.3.2. Métodos en el dominio de la frecuencia

Detección de patrones tonales armónicos. La detección de patrones armónicos en el dominio de frecuencia se define como la búsqueda de la frecuencia fundamental cuyos armónicos expliquen con exactitud los parciales de una señal [24].

Un método propuesto por Doval y Rodet es de este tipo [14]. Se basa en un modelo probabilístico para calcular la probabilidad de una posible frecuencia fundamental cuando se conoce la frecuencia de los parciales en una señal. El método es lo suficientemente flexible como para tener en cuenta un conocimiento previo disponible del sonido en cuestión. Los investigadores ponen su atención en el amplio rango de aplicación de la frecuencia fundamental (50 Hz–4000 Hz) y en el corto tiempo de respuesta (20 ms). Probaron el algoritmo con señales de voz y con señales musicales y propusieron una implementación en tiempo real del mismo. El método utiliza un tratamiento probabilístico separado para las frecuencias y para las amplitudes de cada parcial en la señal. Este hecho es importante si lo que se busca son los objetivos mencionados anteriormente: amplio rango de frecuencia fundamental y flexibilidad.

Doval y Rodet publicaron posteriormente una estructura de mayor nivel con la que se pueden imponer obligaciones a la evolución de la señal y suprimir errores de valores aislados en la detección de altura haciendo un seguimiento de su contorno a través de bandas temporales sucesivas [15]. La lógica de más alto nivel estaba basada en modelos ocultos de Markov y en el algoritmo de Viterbi. En las simulaciones, la tasa de error disminuyó significativamente al emplear la estructura de mayor nivel.

Otros métodos en el dominio de la frecuencia. Existen otros métodos mucho más directos, más de "fuerza bruta" que el método probabilístico de Doval y Rodet. La estructura del espectro de sonidos armónicos puede revelar la

frecuencia fundamental, estimada como el intervalo entre máximos de frecuencia equidistantes en el espectro, mediante el uso de autocorrelación en el espectro de amplitudes. Exponemos dos ejemplos. Lahat describe un algoritmo en el que el espectro de la señal se suaviza mediante un banco de filtros pasabanda y el tono se extrae de las funciones de autocorrelación a la salida de los filtros [41]. Esto es algo más sofisticado que la autocorrelación directa del espectro, pero parte del mismo principio. Kunieda utiliza también la función de autocorrelación del espectro de la señal [40]. Su método se llama ACLOS (*AutoCorrelation of LOg Spectrum*). En este método, se estima la frecuencia fundamental a partir del pico máximo de la función de autocorrelación del espectro logarítmico (*log spectrum*). Este último se define como el logaritmo neperiano del espectro original.

También se ha utilizado ampliamente un método de detección de alturas cepstral [55, 58], sobre todo para extraer la altura de señales de voz en tiempo breve (STFT). El cepstrum se calcula tomando la transformada de Fourier a corto plazo para un segmento de la señal y el logaritmo de su magnitud y finalmente la transformada inversa de Fourier [4]. El valor de pico cepstral determina la altura de la señal. El problema que tiene este método es que fue diseñado para el reconocimiento del habla, donde las frecuencias fundamentales están en el rango de 200 a 500 Hz y, por tanto, no sirve para un rango amplio de tonos y esto es muy importante en el procesamiento de señales musicales.

3.4. Modelos de detección polifónica

Determinar la frecuencia fundamental de sonidos musicales es un problema poco estudiado si se compara con los esfuerzos dedicados a la estimación de alturas de la voz para codificadores de voz y fines comunicativos [9]. Sin embargo, la transcripción polifónica es básica para el análisis de señales musicales y para el análisis computacional de la escena auditiva.

Hay un consenso generalizado en que los métodos para la detección de alturas en una señal monofónica antes mencionados no son los más adecuados para la transcripción de una señal polifónica [33]. Y esto es aún más cierto cuando se trata de señales musicales, puesto que a menudo, en música, la relación de frecuencias de sonidos simultáneos puede hacer que muchos sonidos parezcan uno solo, o que un sonido inexistente destaque con fuerza por el efecto derivado de la interferencia con otros.

El problema no puede ser resuelto por ningún algoritmo que reconozca el valor de pico derivado de un único detector tonal. Pongamos como ejemplo dos sonidos tal que uno sea la octava del otro: todos los armónicos del sonido más alto coinciden perfectamente con posiciones de los parciales del más bajo por lo que ambos sonidos suenan como uno solo y hacen que la separación sea un problema teóricamente ambiguo [33].

En 1986, Chafe señaló: “los patrones de detección forzados son menos propicios a tener éxito en el análisis de la polifonía musical... El balance entre una infradetección y una sobredetección es difícil de alcanzar” [10]. Sin embargo, hasta hace muy poco los problemas de detección de alturas múltiples han sido controlados por técnicas heurísticas *ad hoc* tomando principios algorítmicos de la detección de una única altura. Sólo en los últimos años se han propuesto métodos diseñados específicamente para la detección polifónica.

3.4.1. Modelos tonales

El ejemplo anterior de separar dos sonidos que están a una octava de distancia es un problema teóricamente ambiguo sin un conocimiento a priori de los sonidos. Sin embargo, puede resolverse si esa información se recoge en modelos tonales y se usa en la fase de análisis de la señal. Los modelos tonales son estructuras de datos que representan las características espectrales de los sonidos.

Por lo general, el proceso de modelos tonales recibe como entrada un conjunto de señales de entrenamiento monofónicas, donde cada instrumento se toca con un número suficiente de alturas y estilos para representar el rango completo de colores tonales (*tone colours*) que puede producir. La información almacenada para cada altura suele ser su frecuencia fundamental y las amplitudes de sus armónicos.

Como se ha mencionado anteriormente, Kashino y Tanaka fueron los primeros en describir un sistema de transcripción automática basado en modelos tonales [30]. También propusieron un algoritmo para obtener modelos tonales automáticamente a partir de la señal analizada. El sistema sufría serias limitaciones, pero era bastante genérico en lo que respecta a los tipos de sonidos instrumentales empleados. En simulaciones, un almacenamiento avanzado de modelos tonales tuvo un efecto significativo en la transcripción con polifonía de tres voces, mientras que el modelado tonal automático sólo funcionaba con polifonía de dos voces.

Posteriormente, Klapuri [33] hizo uso de los modelos tonales para transcribir música polifónica principalmente de piano. El sistema de Klapuri se comentará con más detalle más adelante.

El procesamiento basado en modelos tonales también se ha usado en la transcripción de música que contiene sólo instrumentos de percusión [3], es decir, instrumentos no afinados (por tanto, carentes de altura).

3.4.2. Modelos auditivos

El objetivo de técnicas como la de correlogramas [60] no es sólo explicar un amplio rango de fenómenos psicoacústicos auditivos, sino también intentar organizar sonidos con respecto a sus fuentes de generación. Esto, en el caso de la música, significa distinguir sonidos separados.

En 1996, Martin propuso el uso de correlogramas para facilitar la detección de dos sonidos que están separados por una octava de distancia sin introducir modelos tonales [48]. No obstante, hizo muy pocas pruebas y simulaciones para demostrar esto y Klapuri insiste en que los correlogramas son apropiados para el análisis computacional de la escena auditiva, pero demuestra que no lo son para la transcripción de música polifónica [33].

3.4.3. Aplicación explícita de reglas perceptuales

El grupo de investigación de la Universidad de Osaka mencionado anteriormente propone una serie de características auditivas para fusionar o separar características espectrales simultáneas, con el objetivo de encontrar las componentes de frecuencia que pertenecen a la misma fuente [30].

Bregman enumera una serie de características espectrales simultáneas [6] que considera como las más significativas. Entre ellas se encuentran, por ejemplo, la proximidad espectral, la concordancia armónica y la proximidad espacial.

En su tesis de 1998, Klapuri opina que la extracción de características espectrales es la menos prometedora y genérica en el campo de la detección polifónica [33].

3.5. Avances recientes

De las técnicas comentadas anteriormente en la sección 3.4, sólo la de modelos tonales ha tenido verdadero impacto en el campo de la transcripción polifónica. En los últimos años se han producido grandes avances en este campo, llegando incluso a intentos de transcripción de melodías que provienen de fuentes politímbricas, con ruido, con sonidos inarmónicos o con desviaciones de altura leves.

En los últimos años se han propuesto muchas líneas de investigación nuevas para la transcripción polifónica. Un ejemplo es la aplicación de segmentación de imágenes al campo de la transcripción musical [61].

Klapuri es, posiblemente, el investigador que más aportaciones ha realizado recientemente dentro del campo de la transcripción polifónica. Explicaremos su sistema de manera superficial y, posteriormente, terminaremos este capítulo describiendo el papel de las redes neuronales en la transcripción polifónica.

3.5.1. Detección polifónica basada en la escena auditiva

El trabajo reciente de Klapuri se centra en la estrecha relación que hay entre el análisis de la escena auditiva y la detección polifónica. Si se puede determinar la altura de un sonido sin que éste se confunda con otros sonidos que ocurren al mismo tiempo, la información sobre la altura puede usarse para organizar componentes espectrales en sus fuentes de producción [35]; es decir, si tenemos un espectro y tenemos información de las alturas presentes, podemos extraer patrones espectrales. De manera análoga, si pueden separarse de la muestra los componentes espectrales de una fuente, la detección polifónica se reduce a detección monofónica. Esta es la razón por la que la mayoría de sistemas de detección polifónica recientes hacen uso de los principios del análisis humano de la escena auditiva. Se sabe que, en el caso del oído humano, la organización perceptual de componentes espectrales depende de ciertas características acústicas. Dos componentes pueden asociarse a la misma fuente en función de su cercanía en tiempo o frecuencia, concordancia armónica, cambios sincronizados en la frecuencia o amplitud de esos componentes o proximidad espacial en el caso de una entrada multisensorial [6].

El campo de la escena auditiva dió el primer salto cuando en 1994 Brown y Cooke [7] construyeron con éxito modelos computacionales del proceso de análisis de la escena auditiva. Agruparon sonidos musicales de acuerdo a propiedades acústicas comunes.

Posteriormente, en 1999 Godsmark y Brown propusieron una arquitectura de pizarra para facilitar la integración entre distintas características de organización auditiva y la competición entre ellas [18]. El modelo se evaluó mostrando

que puede usarse para obtener líneas melódicas de música polifónica y resolver melodías entrecruzadas.

El sistema de Goto [19] introdujo el primer método de análisis tonal que funcionaba bien para señales musicales complejas y reales. Propuso un algoritmo que puede estimar la frecuencia fundamental en presencia tanto de otros sonidos armónicos como de ruido. El algoritmo de Goto tiene como objetivo encontrar la línea melódica y la línea de bajo en señales de audio complejas. Esto es muy útil para detección de plagios en música, comparando similitudes entre líneas melódicas o de bajo de varias melodías.

En 1998, Klapuri [33] hizo uno de los primeros métodos robustos de transcripción polifónica. Su sistema hacía uso de los modelos tonales, y tenía como objetivo transcribir música de piano. Tomaba el instrumento de la melodía a transcribir y grababa sus notas una a una, hasta representar todos los colores tonales distintos que podía producir ese instrumento. Cuando su programa había estudiado este material, intentaba descomponer señales en un subconjunto de tonos de entrenamiento y determinar su intensidad y la frecuencia fundamental de la señal.

Posteriormente, Klapuri mejoró su sistema y propuso un algoritmo para la detección polifónica sin conocimiento a priori del instrumento empleado [39]. El sistema también es capaz de trabajar con música politímbrica mezclada con fuentes no armónicas, aunque no distingue entre instrumentos (actualmente ningún sistema lo hace), así que la salida es monotímbrica y sólo muestra las notas de los instrumentos afinados, no de los de percusión. Para ello no utiliza conocimiento ni reglas musicales.

Básicamente, el sistema de Klapuri trabaja en dos líneas paralelas; la detección de la métrica de la melodía [34] a partir de los cambios notables en intensidad, altura o timbre de un sonido [51] y la detección de notas, que es la parte que nos interesa y que consta a su vez de dos partes: la de detección polifónica y la de detección de activaciones y desactivaciones de notas.

La parte del sistema de detección de notas dedicada a la detección polifónica se compone fundamentalmente de dos fases que se aplican de manera iterativa. En la primera fase, la de estimación de la altura predominante, se estima la altura del sonido más destacado de la mezcla (es decir, el de frecuencia fundamental de mayor amplitud). Esto se hace mediante la concordancia armónica de componentes espectrales simultáneos [39]. Para ello se suaviza el espectro de la señal (*spectral smoothing*) aplicando conocimiento psicoacústico de acuerdo con los algoritmos expuestos en [35].

En la segunda fase, se estima el espectro del sonido detectado y se sustrae linealmente de la mezcla. Esta fase se basa en el hecho de que las envolventes espectrales de las fuentes de sonido reales tienden a ser continuas [35]. Los pasos de estimación y sustracción se repiten para la señal residual.

Para determinar la parada del sistema de detección debe estimarse el número de voces concurrentes. Este paso es complicado, ya que la dificultad de estimar el número de voces es comparable a la de encontrar los valores de las alturas [36]. Para resolver este problema, el sistema hace uso de una aproximación estadística. Se midieron una serie de características de parada del algoritmo, reflejando la polifonía del sonido en cuestión, y se obtuvo un modelo de estimación polifónica

que se describe con detalle en [37]. Finalmente, hay una fase para suprimir el ruido, entendiéndolo como los componentes de la señal que no pertenecen a sonidos armónicos. Los detalles de este proceso se muestran en [37].

La parte de detección de activaciones y desactivaciones de notas se basa en la separación de sonidos [38]. Los parámetros se estiman en dos recorridos. En el primero, se avanza hacia adelante y se sigue el sonido hasta que su envolvente en amplitud indica que ha dejado de sonar. El segundo avanza hacia atrás en el tiempo y estima el instante de activación de la nota de manera similar [36].

Los resultados del sistema de Klapuri se han mostrado anteriormente en la figura 2.3, junto con los de la capacidad de transcripción que posee un músico.

3.5.2. Redes neuronales en la transcripción polifónica

Las redes neuronales se han explotado muy poco en el campo de la transcripción polifónica, a pesar de ser éste un problema típico de reconocimiento de formas.

Uno de los pocos investigadores que las ha usado es Matija Marolt. Su sistema (SONIC [45]) detecta notas sólo en melodías polifónicas de piano. Soporta bien la presencia de ruido, pero no los instrumentos de percusión, a diferencia del sistema de Klapuri. Los resultados que obtiene en la transcripción de piano son bastante satisfactorios. Para ello usa una técnica de seguimiento de parciales y redes neuronales de tipo TDNN.

La técnica de seguimiento de parciales de Marolt se basa en una combinación entre un modelo auditivo y redes de osciladores adaptativos. Después hace uso de redes TDNN para reconocer notas a la salida del modelo de seguimiento de parciales. Cada red se entrena para reconocer una sola nota, así que el sistema consta de 76 redes. Por lo tanto, cada red tiene una sola salida; un valor alto indica la presencia de una nota en la señal de entrada, y un valor bajo indica que la nota no está presente.

A diferencia del sistema de Marolt, en nuestro trabajo se usa una sola red con tantas neuronas de salida como notas que se quieren detectar. Además, no realizamos un preproceso del espectrograma, sino que lo introducimos directamente en la red, sin considerar previamente ningún modelo auditivo.

SONIC también incluye un algoritmo de detección de activaciones de notas, otro para detectar notas repetidas, uno de afinación y, además, procedimientos simples para calcular la longitud e intensidad de cada nota.

3.6. Productos comerciales

Todavía no ha salido al mercado ningún sistema de transcripción capaz de resolver completamente el problema de la transcripción musical polifónica, pero en los últimos años se han realizado importantes esfuerzos a nivel comercial.

3.6.1. Sistemas de transcripción monofónica

La mayoría de sistemas monofónicos proporcionan la conversión entre formatos WAV[2] (audio digital) y MIDI (representación simbólica de la secuencia de notas), e incluso algunos también realizan la transcripción en tiempo real (por ejemplo, a través de un micrófono). La sigla MIDI proviene de *Musical Instrument Digital Interface*, y es un estándar para representar y comunicar notas musicales y parámetros entre dos dispositivos digitales [2].

SOLO Explorer[3] es el único sistema de entre los que hemos encontrado que publica datos sobre los errores que comete. Es uno de los mejores sistemas comerciales de transcripción monofónica, y no necesita información previa del instrumento que se ha usado para generar la melodía. Funciona muy bien con voz y es bastante robusto ante el ruido, además de detectar desviaciones leves de la altura (vibratos). Para hacernos una idea de la calidad de uno de los mejores sistemas monofónicos que existen hasta la fecha, vamos a mostrar el error cometido por SOLO (en su versión actual) sobre una muestra de unas 100 canciones de música folk:

- Error de octava: 0.33 %
- Notas "partidas": 4.60 %
- Notas no detectadas: 18.50 %
- Error de reconocimiento de altura: 6.60 %

Lamentablemente no conocemos el sistema empleado para medir el error, pero mencionan las condiciones en las que se han realizado las pruebas. Algunas de ellas tenían mucho ruido de fondo y notas con desviaciones leves de altura. La mayoría de ejemplos están realizados con voz humana, que es más difícil de detectar que muchos instrumentos musicales debido a su riqueza y variabilidad armónicas.

El problema de la transcripción monofónica sigue abierto, pero se han producido grandes avances. Hay muchos otros sistemas de detección monofónica. Entre ellos destacan Digital Ear[4], Inst2Midi[5] y Autoscore[6].

3.6.2. Sistemas de transcripción polifónica

Todos los sistemas que se mencionan a continuación son sistemas comerciales de transcripción polifónica que convierten ficheros de formato audio a formato MIDI, o que también funcionan en tiempo real a través de un micrófono.

WIDI[7] es posiblemente el sistema de transcripción polifónica más completo hasta la fecha. En la actualidad, ningún programa de transcripción es capaz de separar los instrumentos de una melodía. Pero éste, a partir de una fuente de audio polifónica y con varios instrumentos (multitímbrica), genera un MIDI

[2] El formato WAV, desarrollado por Microsoft, es uno de los formatos de fichero más estándar para el almacenamiento del sonido digital. Es un caso particular del más general *Resource Interchange File Format* (RIFF).

[3] `http://www.recognisoft.com`

[4] `http://www.digital-ear.com/`

[5] `http://www.nerds.de/english/inst2midi.html`

[6] `http://www.wildcat.com/`

[7] `http://www.widisoft.com/`

polifónico monotímbrico. Es decir, a la entrada puede tener varios instrumentos sonando de manera polifónica, y la salida es una partitura con las notas de todos los instrumentos mezcladas como si se tratase de uno solo. *WIDI* acepta fuentes de sonido no armónicas (ruido, percusiones, ...) que no se corresponden con notas afinadas, y las omite a la salida de la transcripción. Además, es sensible a la amplitud sonora de las notas.

AmazingMIDI[8] puede reconocer música polifónica monotímbrica. Necesita una muestra (una nota) de la señal original, y asume que todas las notas de la melodía a transcribir tienen el mismo patrón espectral que la señal de ejemplo.

AKoff Music Composer[9] usa modelos armónicos para mejorar el reconocimiento de los instrumentos adecuados. El programa transcribe música polifónica monotímbrica. Filtra los armónicos altos y tiene restringido el rango de notas que puede reconocer. Además, se puede filtrar el ruido de la señal original de manera manual o automática.

Intelliscore[10] de IMSysInc viene en dos versiones: monofónica y polifónica. También puede usarse en tiempo real, y detecta vibratos. Está en constante desarrollo desde 1997, y su versión actual es la 5.1. Emplea tres algoritmos de reconocimiento diferentes (de los que no dan detalles) basados en física psicoacústica. Usa plantillas de instrumentos, es decir, se elige la configuración del instrumento a transcribir antes de ejecutar el programa. Intelliscore incluye 95 configuraciones de instrumentos obtenidos a partir del análisis de 10 000 grabaciones musicales.

Es difícil evaluar cuantitativamente sistemas comerciales de transcripción automática, ya que estos no proporcionan datos sobre la calidad de la transcripción lograda ni sobre los algoritmos empleados.

[8] `http://www.pluto.dti.ne.jp/~araki/amazingmidi/`
[9] `http://www.akoff.com/`
[10] `http://www.intelliscore.net`

4. METODOLOGÍA DE LA TRANSCRIPCIÓN

En este trabajo se va a abordar el problema de la transcripción polifónica sin considerar para ello el análisis de la escena auditiva y sin separar el problema en detección de activación/desactivación de notas y detección de alturas. No somos expertos en percepción auditiva, así que no vamos a tratar de resolver el problema desde esta perspectiva, sino que usaremos una red neuronal que esperamos que sea capaz de resolverlo por nosotros y deducir alturas a partir de un espectrograma. Por lo tanto, no usaremos modelos de ningún tipo ni arquitecturas especializadas tales como la de pizarra, sino que alimentaremos directamente a la red con espectrogramas.

Los datos que se necesitan para el entrenamiento son espectrogramas de melodías junto con su correspondiente notación simbólica. Se describe el proceso con más detalle en la sección 4.1. Posteriormente pasaremos a describir la red neuronal empleada y los parámetros necesarios para su correcto funcionamiento. Explicaremos la medida del cálculo del error que se ha empleado y, por último, el programa que se ha desarrollado para representar gráficamente este error.

4.1. Construcción de los pares de entrada/salida de la red

Para entrenar la red queremos presentar a la entrada las amplitudes de una serie de bandas del espectrograma para un instante dado y a la salida las notas que suenan en ese momento y que dan lugar a ese espectrograma.

Necesitamos construir pares de entrada/salida formados por el espectro del sonido producido por una fuente para ventanas temporales distintas alrededor de un instante dado t_i. La entrada es $\{S(f, t_{i+j})\}$ para $j \in [-m, +n]$, siendo f la frecuencia y m y n el número de ventanas consideradas antes y después del instante central t_i. La salida consiste en una codificación del conjunto de notas posibles $\nu(t_i)$ que están activas en ese instante t_i y que producen esos espectros usados para la entrada.

Como muestras de entrenamiento, se ha preparado un conjunto de melodías polifónicas en formato MIDI. Estas melodías se transforman, mediante un programa (`midi2cs`[1]) en ficheros de texto con formato de partitura CSound.

CSound es un programa de síntesis de sonido [62, 5]. El programa usa como entrada una serie de especificaciones de sintetizadores (instrumentos) en formato texto localizadas en el fichero *orquesta* y ejecuta esos instrumentos interpretando una lista de eventos que contiene notas y parámetros de control de síntesis localizada en el fichero *partitura*. Con todo esto genera un fichero de sonido en formato WAV. Como hemos comentado anteriormente, el fichero de partitura puede obtenerse a partir de un fichero MIDI.

[1] `http://www.midi2cs.de`

Para nuestro propósito, tenemos que realizar una transformación del formato de partitura de CSound para conseguir una matriz de activaciones de notas. Llamaremos a esta matriz Pianola Binaria Digital (*Binary Digital Piano-roll*, BDP), y su estructura se detalla en el apartado 4.1.2. Dentro de este proyecto, se ha desarrollado un programa que obtiene esta matriz a partir de un fichero de partitura de CSound. En resumen, hemos transformado un fichero MIDI en una partitura CSound para el control preciso de la síntesis de los sonidos que van a servir de base al entrenamiento de la red y, por otro lado, en una BDP en la que tenemos codificadas, para cada instante de tiempo, las notas que se activan. Esta salida será el objetivo de la red durante la fase de entrenamiento.

4.1.1. Datos de entrada

Utilizaremos CSound para sintetizar las señales de audio, que almacenaremos en formato WAV. Se ha optado por usar ficheros de audio monoaurales, muestreados a 44100 Hz y con una resolución de 16 bits por muestra (calidad correspondiente a un CD de música). Como datos de entrada a la red se usan las bandas del espectrograma obtenido a partir de estas muestras de audio.

Para obtener los espectrogramas se ha usado Spectralab, un programa de análisis de sonido que permite volcar en un fichero los datos del espectrograma correspondientes a una muestra de sonido. Pasaremos, por tanto, las melodías que han sido generadas con CSound en formato digital (WAV) a ficheros de texto que contienen espectrogramas, y que consisten en una matriz con los datos de la amplitud de cada banda de frecuencia para cada instante de tiempo.

Para llevar a cabo la STFT de una señal hace falta tener en cuenta una serie de parámetros, que se detallan a continuación.

La frecuencia de muestreo es la frecuencia a la cual está digitalizada la señal. En este trabajo, la frecuencia de muestreo es $f_s = 44100$ Hz. Para realizar el análisis espectral, se pueden usar divisores de la frecuencia de muestreo, lo cual reduce la cantidad de datos aumentando así la eficiencia pero reduciendo también el rango de frecuencias del espectro.

En el espectrograma, la representación tanto en el eje de frecuencias como en el de amplitudes se puede hacer usando una escala lineal o logarítmica. En este trabajo se ha optado por escalas logarítmicas tanto para los valores de frecuencia como para los de amplitud, lo que equivale a una escala lineal medida en alturas de notas musicales para la frecuencia y lineal en decibelios para la amplitud, más cercana en ambos casos a la percepción auditiva de las amplitudes sonoras.

La resolución espectral de la STFT indica cuál es la división mínima representable en el eje de frecuencias. Para las entradas de la red vamos a usar una escala logarítmica de un doceavo de octava (una octava contiene 12 notas). Esto equivale a decir que cada banda de frecuencia en la entrada de la red se corresponderá con un un semitono de la escala musical.

El rango de frecuencias que se obtiene en la STFT depende de la frecuencia de muestreo. Como máximo este rango puede ser igual a la frecuencia de Nyquist que, por definición, tiene un valor de $f_s/2$.

A la hora de realizar la STFT se puede optar por distintos tipos de ventanas para establecer qué parte del fichero de sonido se analiza en cada instante. La transformada de Fourier opera sobre una señal de longitud infinita, de forma que

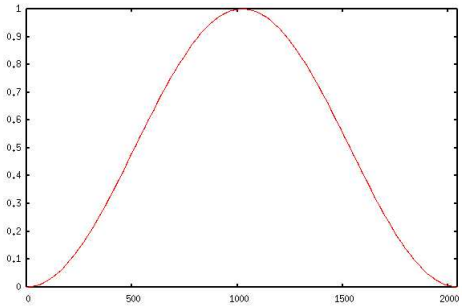

Figura 4.1: Ventana Hanning de 2048 muestras.

la STFT, aplicada sólo a una parte de la señal, requiere que ésta sea, en cierta forma, expandida hasta una longitud infinita teórica. Esto se consigue repitiendo la parte de la señal correspondiente a la ventana utilizada en la STFT un número infinito de veces para producir la señal que es luego transformada. Como consecuencia, se produce una discontinuidad en los extremos de la ventana. En este trabajo se ha usado una ventana Hanning, ya que con ella no se distorsionan en exceso estos extremos. Dicho tipo de ventana corresponde a un ciclo de coseno escalado y desplazado en su valor, y su expresión en el instante τ es la siguiente:

$$w(\tau) = \frac{1}{2}\left(1 - \cos\frac{2\pi\tau}{N}\right) \tag{4.1}$$

donde N es el número total de muestras que abarca la ventana. En los experimentos se ha usado $N = 2048$. Se puede ver una representación gráfica de la ventana Hanning en la figura 4.1.

Con estos datos vamos a calcular la resolución temporal $\Delta t = t_{i+1} - t_i$, que viene dada por la ecuación

$$\Delta t = \frac{N}{f_s} \tag{4.2}$$

siendo N el tamaño de la ventana en muestras y f_s la frecuencia de muestreo.

Al aplicar este tipo de ventanas a una señal se pierde cierta información en los límites de la misma. La solución es solapar las ventanas, de manera que la información que se pierde en cada una de ellas sea tenida en cuenta por la siguiente. Por ejemplo, si se empleara una ventana Hanning con una anchura de 100 ms, se podrían solapar las ventanas al 50 % (lo típico en reconocimiento automático del habla), de forma que a la señal se le aplica la ventana cada 50 ms. En este trabajo, se ha optado por un solapamiento $S = 50\,\%$. La nueva resolución temporal $\Delta t = t_{i+1} - t_i$, vendrá dada por la ecuación

$$\Delta t = \frac{SN}{100 f_s} \tag{4.3}$$

siendo N el tamaño de la ventana en muestras, f_s la frecuencia de muestreo y S el porcentaje de solapamiento.

La amplitud de las bandas del espectrograma se ha obtenido en dB como atenuaciones de la amplitud máxima encontrada. Se ha optado por usar 16 bits

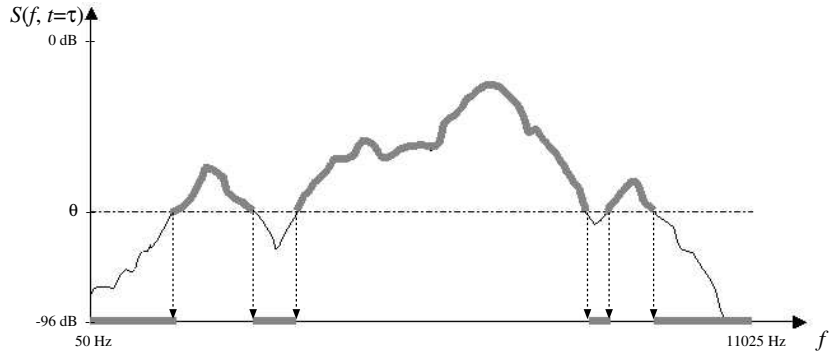

Figura 4.2: Se aplica a los espectrogramas un umbral pasa-alta para eliminar valores de amplitud muy pequeños que pudieran confundir a la red.

de resolución (la misma que la de un CD de audio), lo que nos da un rango dinámico[2] de ($20 \log_{10} 2^{16} = 96$ dB). Antes de introducir las atenuaciones en la entrada de la red, se normalizan en el intervalo $[-1, +1]$, asignando el valor -1 a la atenuación máxima (-96 dB) y $+1$ a la atenuación de 0 dB.

Para quitar el ruido y destacar las componentes importantes del espectrograma, se aplica un umbral de bajo nivel θ, de tal manera que si $S(f_k, t_i) < \theta$ entonces $S(f_k, t_i) = -1$. La fig. 4.2 muestra el efecto de este umbral. Experimentalmente se ha comprobado que el umbral no condiciona mucho los resultados dentro de un rango grande de valores. Para ello, se ha establecido un valor fijo para este parámetro en los experimentos que es $\theta = -45$ dB.

Normalmente, la activación o desactivación de una nota no está centrada temporalmente en la ventana correspondiente de la STFT, así que la posición concreta de la campana de la ventana puede afectar a la amplitud, perdiéndose así una cantidad importante de energía. A la hora de resolver este problema es cuando una red dinámica como la TDNN se revela útil. Las posiciones adyacentes solapadas del espectrograma dan a la red esta información dinámica. Consideremos b como el número de bandas del espectrograma. Para cada ventana adicional, se añadirán b neuronas más a la capa de entrada. El número total de neuronas de entrada será igual a $(b\,(n + m + 1))$. En la figura 4.3 se muestra un esquema de la arquitectura de la red.

4.1.2. Datos de salida

La capa de salida de la red se compone de b neuronas (ver figura 4.3), una por cada posible nota que puede detectarse, valor que coincide con el número de bandas del espectro para cada entrada. Por tanto, tendremos una salida simbólica con tantas neuronas como notas en el rango válido.

Hemos codificado la salida de manera que un valor de activación $\nu(k, t_i) = 1$ para una unidad particular k significa que la nota k-ésima de la escala está activa en ese instante, y un valor 0 representa una nota no activa. Por lo tanto, el vector

[2]El rango dinámico es la diferencia entre los niveles sonoros máximo y mínimo que se pueden registrar.

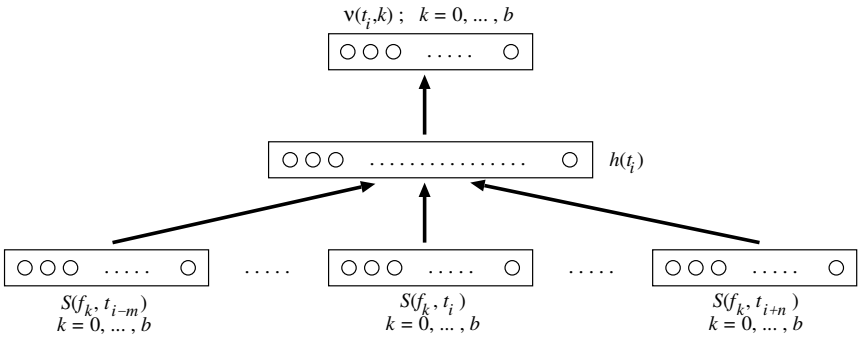

Figura 4.3: Estructura de la red e información que manipula durante el entrenamiento. Arriba se muestra la capa de salida, en el centro la capa oculta y abajo la capa de entrada.

```
G#0                    C5          C6          C7          C8    F8
 |                      |          |           |           |     |
 00000000 ... 00000000000000000001000000000000000010000000000000
 00000000 ... 00000000000000000001000000000000000010000000000000
 00000000 ... 00000000000000000001000000000000000010000000000000
 00000000 ... 00000000000000000001000000000000000010000000000000
 00000000 ... 00000000000000000001000000000000001000000000000000
```

Figura 4.4: Subconjunto de una BDP, que codifica las activaciones de las notas (unos). Las columnas representan las activaciones de las notas en cada instante. Se han omitido columnas para que la figura quepa en la página. El tiempo avanza de arriba a abajo.

$\nu(t_i)$ tendrá b componentes. Normalmente el número de ceros es mucho mayor que el de unos, porque sólo un pequeño subconjunto de notas está activo en cada instante.

Representaremos una pianola binaria digital (*Binary Digital Piano-roll*, BDP) como una serie de vectores $\nu(t), t = 1, 2, ...$ Podemos observar un ejemplo de una BDP en la figura 4.4. Existen dos notaciones para representar las notas musicales; la latina y la sajona (ver tabla 4.1), que coexisten en la presente memoria. La figura 4.4 se muestra en notación sajona.

Los vectores en cada instante son las salidas que se muestran a la red durante la fase de entrenamiento, mientras a la entrada se le introducen los espectros correspondientes. La BDP se obtiene a partir del fichero MIDI y marca las notas que están activas en los instantes donde esán centradas las ventanas de la STFT.

El hecho de que las ventanas STFT estén centradas implica que los ficheros de espectrograma y de la BDP difieren ligeramente en los instantes de tiempo.

C	D	E	F	G	A	B
Do	Re	Mi	Fa	Sol	La	Si

Tabla 4.1: Notación sajona para notas y su correspondencia latina.

Mientras que en el fichero de espectrograma el instante de cada muestra es un múltiplo de la resolución temporal (Δt), en la BDP se codifican las notas que están sonando en el instante $\Delta t + (\Delta t/2)$, ya que en esta posición es donde se encuentra el centro de la ventana STFT. Por ejemplo, el valor dado en el espectrograma en el instante $t = 2.55$ s para una banda siendo $\Delta t = 23$ ms representa la amplitud media de esa banda en el intervalo de tiempo que va desde $t = 2.32$ s hasta t =2.55 s. Al tratarse de la amplitud medida entre dos instantes de tiempo, capturamos en la BDP las notas que suenan justo en mitad de esa ventana, es decir, en t =2.43 s.

4.2. Descripción de la red

La red neuronal construida para resolver el problema es básicamente una red TDNN entrenada mediante el algoritmo de retropropagación del error (*back-propagation* [59]). Como hemos visto, la red está compuesta por tres capas de neuronas; una capa de entrada, que recibe los datos del espectrograma, una capa oculta y una capa de salida, que codifica las activaciones de las notas en cada instante (ver figura 4.3).

Como se ha comentado con anterioridad, es frecuente que una nota no empiece o deje de sonar justo en el instante en el que se toma la muestra, debido a la naturaleza del problema. Puede que en la STFT para determinada posición de la ventana se detecte parte de la energía de una nota en el espectrograma (eso sí, con menor amplitud de lo habitual) y que según la BDP ésta deje ya de haber sonado o no esté sonando todavía.

Existe además un problema añadido. Hay determinadas señales con un patrón espectral dinámico, es decir, que varía con el tiempo. En este tipo de señales, los armónicos de una ventana pueden tener distintas amplitudes que los de la ventana siguiente pese a pertenecer todos a la misma nota. Para estas señales que no tienen un espectro "estable", la red debe tener información sobre las ventanas adyacentes a la ventana considerada.

Para intentar paliar estos dos problemas, adquiere sentido el uso de una red dinámica TDNN, que toma como entrada más de una ventana (más de una muestra de tiempo) en cada instante.

4.3. Parámetros de la red

Hay una serie de parámetros de la red que tienen un efecto directo sobre su comportamiento:

- **Límite de reconocimiento (Θ)**. Es el umbral usado para transformar la salida continua de la red ($\tilde{y}(k) \in [-1, 1]$) en discreta, es decir, en codificaciones de activación de notas ($y(k) \in \{0, 1\}$). Según este umbral, si $\tilde{y}(k) < \Theta$ entonces $y(k) = 0$ y en otro caso $y(k) = 1$. Disminuir este parámetro provoca que la red sea menos restrictiva con las notas que detecta. Cuanto menor sea el valor de Θ, más posibilidades hay de que una nota dudosa se codifique con un 1, es decir, que aparezca como activa en un determinado instante.
- **Número de ventanas anteriores en la entrada (m)**. Como se ha comentado anteriormente, para ondas con un espectro que varía con el

paso del tiempo convendría aumentar m para que la red TDNN capte a su entrada más ventanas temporales del espectrograma. En otros casos es innecesario, ya que incrementa notablemente el coste computacional del entrenamiento y no mejora los resultados.

- **Número de ventanas posteriores en la entrada (n).** Es el número de ventanas que se usan después (temporalmente hablando) del instante t_i correspondiente a la ventana actual.
- **Número de neuronas en la capa oculta (h).** Si el numero de ventanas anteriores o posteriores es alto, conviene que este parámetro también lo sea. Aumentar el número de neuronas de la capa oculta puede mejorar el entrenamiento, pero con un valor demasiado alto se corre el riesgo de que la red aprenda los casos (se produzca sobreaprendizaje) en lugar de aprender la relación buscada.
- **Parámetro de parada (T_S).** Marca el número de épocas que transcurren desde que se alcanza un valor mínimo de error de validación hasta que se da por finalizado el entrenamiento. Cuanto mayor sea este parámetro, más tardará en converger el algoritmo.
- **Número de épocas del entrenamiento (T_E).** Indica el número de épocas transcurridas desde el inicio del aprendizaje hasta su finalización.
- **Coeficiente de aprendizaje (μ).** El coeficiente de aprendizaje μ puede ser constante durante el proceso de entrenamiento o bien decrecer con el número de época. Se ha probado con un decrecimiento logarítmico, de manera que en las primeras épocas el coeficiente es más alto (la red aprenderá más rápido pero con menor precisión) que en los últimos, es decir, conforme va aumentando el número de épocas, μ va decreciendo para que el aprendizaje sea más lento pero más preciso. Tras realizar pruebas exhaustivas, se ha comprobado que un coeficiente μ variable empeora los resultados, así que para los experimentos mostrados en los siguientes capítulos, μ siempre es fijo.

4.4. Muestras utilizadas

Como hemos comentado anteriormente, para realizar las melodías de muestra se ha usado CSound, un sintetizador por software. No se han usado grabaciones reales (por ejemplo, la obtenidas directamente desde un micrófono) debido principalmente a la estricta sincronización entre la partitura y la grabación que necesitan las muestras a la entrada de la red. Además, no poseemos instalaciones insonorizadas con las características acústicas necesarias para la grabación. El uso de señales sintetizadas hace más fácil el problema, ya que no existe ruido en la señal generada y el espectro es fielmente el del instrumento sintético empleado.

Para realizar el entrenamiento se ha intentado usar melodías con la mayor variedad posible de acordes y de ritmos, para intentar que la red aprenda a relacionar el espectrograma con las notas en vez de aprender casos. Hay melodías que incluyen acordes, silencios y escalas.

Todas las notas de las melodías usadas han sido generadas usando los mismos parámetros de síntesis para cada una de las formas de onda consideradas. El sistema sólo detecta notas sin tener en cuenta su volumen.

Ninguna de las melodías contiene notas desafinadas ni modulaciones en altura (vibratos).

Los experimentos se han realizado con dos configuraciones distintas, que se detallan más adelante. En un principio, las pruebas se realizaron con 7 melodías en el conjunto de entrenamiento y 3 en el conjunto de validación, todas ellas de un minuto de duración aproximadamente. Cada minuto contiene unos 260 espectros en el caso de la Configuración I y unos 1300 en el caso de la Configuración II. Con las mismas melodías, la Configuración II tiene más muestras que la Configuración I debido a que su Δt es menor, tal como se verá más adelante.

Se observó que era insuficiente tener 7 melodías en el conjunto de entrenamiento y 3 en el de validación, ya que se produjeron mejoras significativas en el entrenamiento aumentando el tamaño de estos conjuntos. Esto se debe a que, aunque en principio el número de muestras parecía suficiente para el entrenamiento, en éstas también había espectros muy parecidos, al repetirse acordes y notas. También se debe a que es habitual que una nota esté sonando durante más de Δt segundos, que es la duración de una ventana temporal, lo cual resta variabilidad a las muestras.

Por lo tanto, los entrenamientos se repitieron con 18 melodías en el conjunto de entrenamiento y 6 en el de validación. Esto equivale a unas 4 680 muestras de entrenamiento (ventanas) para el caso de la Configuración I y 32 500 para la Configuración II.

Todos los resultados expuestos en esta memoria se corresponden con estos conjuntos de entrenamiento (18 melodías) y de validación (6 melodías).

4.5. Cálculo del error

No existe un consenso claro entre los sistemas de transcripción existentes a la hora de medir la calidad del sistema. La mayoría (incluido Klapuri [36]) considera el porcentaje de error como la suma de las notas detectadas incorrectamente dividido entre el número total de notas de la transcripción original.

A nuestro juicio, este no es el sistema más adecuado para medir la calidad de la transcripción. Puede haber un porcentaje de error mayor del 100 %, y no se tienen en cuenta las notas no detectadas. Es decir, si se detectan sólo dos notas siendo una correcta y la otra incorrecta en una melodía de cien notas, el porcentaje de error sería tan sólo de $1/100 = 1\,\%$.

Otra medida de error de investigadores como Marolt [45] se basa en lo contrario y favorece la sobredetección; el porcentaje de aciertos es el número de notas correctamente detectadas dividido entre el número de notas de la partitura original. Marolt da un porcentaje de acierto del 94.39 %, cuando el porcentaje de notas detectadas que no aparecía en la partitura original es del 11.15 % y el porcentaje de notas con errores de octava es del 77.90 % [45]. Marolt afirma en este artículo que los errores de octava no son muy relevantes a la hora de escuchar la transcripción resintetizada.

En este trabajo se va a considerar un porcentaje de acierto más restrictivo, y se medirá el error cometido en dos niveles diferentes:

- Considerando las detecciones en cada posición t_i de la ventana del espectrograma para saber qué ocurre con la detección en cada instante. Llamaremos a esto "detección de eventos".
- Considerando notas como series de detecciones de eventos consecutivas en el tiempo. Llamaremos a esto "detección de notas".

Para cada instante t_i las activaciones de las neuronas de la capa de salida $y(t_i)$ se comparan con el vector $\nu(t_i)$. Se considera una detección correcta de un evento cuando $y(t_i, k) = \nu(t_i, k) = 1$ para una neurona de salida k. Se produce un falso positivo cuando $y(t_i, k) = 1$ y $\nu(t_i, k) = 0$ (se detecta un evento correspondiente a una nota que realmente no estaba activa en el instante t_i), y un falso negativo cuando $y(t_i, k) = 0$ y $\nu(t_i, k) = 1$ (no se ha detectado un evento pero la nota sí estaba activa en t_i).

En los experimentos se contabilizan el número de estos eventos, y la suma de eventos correctos Σ_{OK}, de falsos positivos Σ_+, y de falsos negativos Σ_-. Usando estos datos, el porcentaje de aciertos en la detección se define como:

$$\sigma = \frac{100\Sigma_{OK}}{\Sigma_{OK} + \Sigma_- + \Sigma_+} \tag{4.4}$$

Con respecto a las notas, hemos estudiado la salida obtenida de acuerdo a los criterios descritos anteriormente y las secuencias de eventos detectados se han analizado de tal manera que una nota es un falso positivo cuando hay una serie contigua de eventos que son falsos positivos y que están rodeados de silencios. Se dice que una nota es un falso negativo si hay una serie de eventos consecutivos que son falsos negativos y que están rodeados de silencios. Cualquier otra secuencia de eventos consecutivos (sin silencios en su interior) se considera como una nota detectada correctamente. Para calcular el porcentaje de acierto en la detección de notas se usa la misma ecuación anterior.

Representación gráfica del error

Se ha desarrollado un programa para comparar gráficamente dos BDP. La entrada del programa es la BDP obtenida a la salida de la red y la BDP original del fichero. La salida es una gráfica en la que se representan las similitudes y diferencias entre las dos BDPs.

Las marcas 'o' corresponden a eventos detectados correctamente, el símbolo '-' corresponde a falsos negativos y los eventos etiquetados como '+' son falsos positivos. En lo sucesivo, los resultados se representarán con esa notación.

5. CONFIGURACIÓN I

Hemos definido anteriormente la metodología común a todas las pruebas realizadas, y ahora vamos a concretar los parámetros individuales de los experimentos preliminares. El objetivo de estos experimentos es comprobar la viabilidad del método.

En este capítulo, veremos los parámetros de espectrograma y de red usados en las primeras pruebas, así como los resultados de los experimentos para distintos tipos de timbres. Asimismo, iremos variando los parámetros de la red con el fin de obtener los mejores resultados posibles.

Los experimentos se van a realizar con melodías polifónicas generadas con timbres de onda sinusoidal, triangular, cuadrada, diente de sierra y con dos clases de clarinete.

Para estas primeras pruebas, sólo se muestra el porcentaje de acierto de eventos, no de notas. Se hará una crítica de las limitaciones que tiene el método para introducir mejoras que se verán reflejadas posteriormente en la Configuración II.

5.1. Parámetros del espectrograma

Para realizar los primeros experimentos, se ha probado con una configuración de espectrograma con una división (*decimation ratio*) de 10, con el objetivo de reducir la carga de la red en estos experimentos preliminares. La frecuencia de muestreo original es de $f_s = 44100$ Hz, pero al dividir entre 10 el número de muestras, la frecuencia de muestreo operativa queda en $f_s = 4410$ Hz. Debido al teorema de Nyquist, la frecuencia máxima representable es $f_s/2 = 2120$ Hz.

Por lo tanto, usando una escala logarítmica para las bandas de entrada a la red y teniendo la primera banda de frecuencia centrada en 50 Hz, el número de bandas es $b = 66$, con un rango de que va desde 50 Hz hasta 2120 Hz, correspondiente al rango de notas desde Sol♯ (octava 0) hasta Do♯ (octava 6)[1]. Nos quedamos, pues, con unas seis octavas, un rango cercano al que puede emitir un piano. Cuanto mayor sea ese rango, mayor será el coste computacional que deba soportar la red neuronal durante el entrenamiento.

Como se ha mencionado anteriormente, la resolución temporal Δt viene dada por la ecuación 4.3. Para el caso que nos ocupa, $N = 2048$, $S = 50\,\%$ y $f_s = 4410$ Hz, por lo que la resolución temporal es $\Delta t = 232.2$ ms. En la implementación de la STFT, Δt está definido con mucha más precisión en sus decimales para no ir perdiendo exactitud temporal de las sucesivas muestras cuando se va avanzando en el tiempo. Con estos datos, la STFT proporciona un conjunto de amplitudes

[1] En lo sucesivo utilizaremos una notación de subíndices para representar la octava. En este caso sería Sol♯$_0$ y Do♯$_6$.

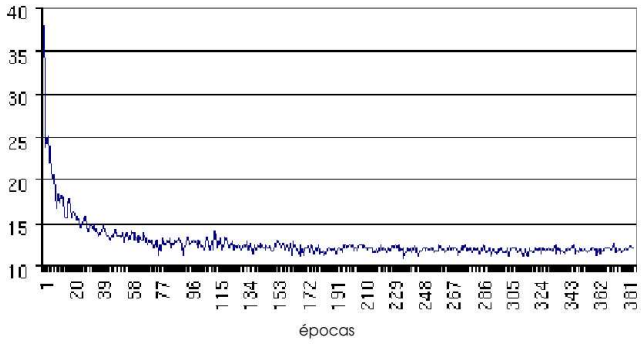

Figura 5.1: Evolución del error de validación durante el entrenamiento

para 1024 frecuencias con una resolución espectral (separación entre frecuencias) $f_R = f_s/N = 2.15$ Hz.

Una resolución temporal $\Delta t = 232.2$ ms provoca que se pierdan algunas notas cortas. Hay que tener en cuenta que, con un tempo de $T = 120$ bpm[2], una nota corchea dura teóricamente 250 ms, con lo que el sistema puede tener dificultades para detectarla.

5.2. Sobre la convergencia

El algoritmo de entrenamiento converge en relativamente pocas épocas. En la figura 5.1 se puede observar una curva típica de la evolución del porcentaje de error de validación durante la fase de entrenamiento.

Como hemos mencionado anteriormente, conviene que el coeficiente de aprendizaje (μ) sea fijo, ya que si decrece logarítmicamente se dan muchos "saltos" en las primeras épocas del entrenamiento y hay muchas posibilidades de que se acabe en un mínimo local del espacio de pesos con un error muy superior al que se busca. Si para evitar este problema se reduce este coeficiente, éste se hace muy pequeño en las siguientes épocas y a la red le cuesta mucho aprender. Experimentalmente, los mejores resultados se han conseguido con un coeficiente de aprendizaje fijo perteneciente al intervalo [0.002,0.02].

La modificación del número de neuronas de la capa oculta no afecta en exceso al comportamiento de la red. Experimentalmente se ha observado que el número adecuado de neuronas en la capa oculta es, aproximadamente, 100. Aumentando este valor, la red puede sufrir sobreaprendizaje y dar errores mayores para el conjunto de validación. Reduciendo este parámetro, la red tiene dificultades para encontrar la relación buscada.

En los experimentos se ha mostrado adecuado un límite en decibelios en torno a $\theta = 45$ dB. Con valores menores de 40 dB, perdemos información importante del espectrograma, y a partir de los 50 dB aproximadamente, a la red le cuesta más el aprendizaje al encontrarse con atenuaciones irrelevantes en su entrada.

[2]bpm (*beats per minute*) es una medida de tiempo de la melodía, y su valor representa los tiempos de compás por minuto.

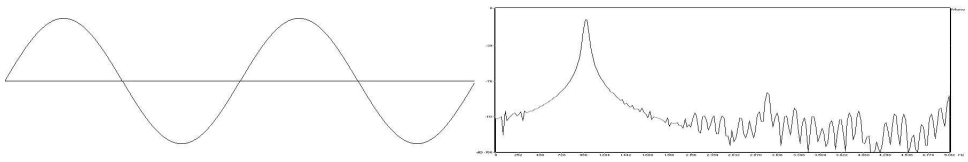

Figura 5.2: Dos ciclos de la onda sinusoidal usada para realizar los experimentos y su correspondiente espectro utilizando una ventana Hanning.

El entrenamiento de la figura 5.1 se ha realizado en un PC con una frecuencia de reloj de 800 MHz, y ha tardado aproximadamente 10 minutos.

5.3. Onda sinusoidal

Las primeras pruebas se han realizado con una onda sinusoidal, que tiene la característica de concentrar toda la energía de su espectro en la amplitud del parcial correspondiente a la frecuencia fundamental. Esto equivale a decir que, en su espectrograma, la banda de frecuencia que se corresponde con la amplitud máxima coincidirá con la banda de frecuencia de la nota deseada. El entrenamiento con una onda simple sinusoidal es el más básico que se le puede plantear a la red. La ecuación de la forma de onda de un seno de amplitud unidad es:

$$s(t) = \text{sen}\left(\frac{2\pi f t}{N}\right) \tag{5.1}$$

En la figura 5.2 puede verse gráficamente la forma de onda de un seno definido por $N = 2048$ puntos, junto con su espectro calculado con los mismos parámetros utilizados en el desarrollo de nuestros experimentos. Éste es el timbre que se ha usado para realizar las pruebas iniciales.

La tabla 5.1 muestra los resultados obtenidos tras el entrenamiento. En esta tabla destaca la influencia del número de ventanas anteriores (m) y del número de ventanas posteriores (n). El resto de parámetros tienen un efecto de ajuste fino. También hay que tener en cuenta que pequeñas diferencias entre los resultados pueden achacarse a la aleatoriedad de los pesos iniciales.

Como se puede observar en las filas sombreadas de la tabla 5.1, el hecho de incluir ventanas adyacentes a la ventana actual mejora el entrenamiento. Se obtienen los mejores resultados entrenando la red con una ventana anterior y ninguna posterior, y con una anterior y una posterior. No parece que añadir más ventanas mejore los resultados.

El límite de reconocimiento Θ se ha ido bajando deliberadamente tras comprobar que en los resultados debían aparecer más notas activas de las que estaban apareciendo. Bajando este umbral se consigue ser menos riguroso a la hora de clasificar una nota como activada y mejoran los resultados.

El mejor resultado de las pruebas se ha obtenido en la época $T_E - T_S = 109$, dando como resultado una tasa de acierto de eventos del 89.01 %. Podemos ver, por tanto, que no es necesario que el parámetro de tiempo de parada sea superior a 200 épocas.

Θ	m	n	h	θ	T_S	μ	T_E	$\sigma(\%)$
0	1	1	100	45	200	0.01	292	87.57
0	1	1	100	45	100	0.005	242	88.59
0	1	1	70	45	100	0.005	228	85.06
0	1	1	120	45	100	0.005	207	86.92
-0.5	1	1	100	45	100	0.005	154	88.27
0	1	1	100	30	100	0.005	138	83.27
-0.7	1	1	100	60	100	0.01	489	85.45
0.5	1	1	100	45	100	0.005	203	86.22
-0.8	1	1	100	45	100	0.01	252	88.27
-0.8	1	1	100	45	200	0.005	572	87.89
-0.7	1	1	100	45	200	0.01	299	87.57
-0.7	1	1	100	45	1000	0.01	1206	88.72
0	1	1	100	45	1000	0.008	1436	88.21
-0.7	0	0	100	45	150	0.01	202	71.37
-0.7	0	1	100	45	150	0.01	318	79.93
-0.7	1	0	100	45	150	0.01	259	**89.01**
-0.7	1	1	100	45	100	0.01	406	87.63
-0.7	2	2	100	45	150	0.01	785	87.08
-0.7	1	0	100	80	150	0.01	195	84.54
-0.7	1	0	100	45	150	0.1	316	73.17
-0.7	1	0	100	45	200	0.005	244	88.88
-0.7	1	0	140	40	200	0.01	347	88.31
-0.7	1	0	100	45	150	0.01	179	88.50
-0.7	1	0	100	45	300	0.01	415	88.95

Tabla 5.1: Resultados del entrenamiento con una onda sinusoidal.

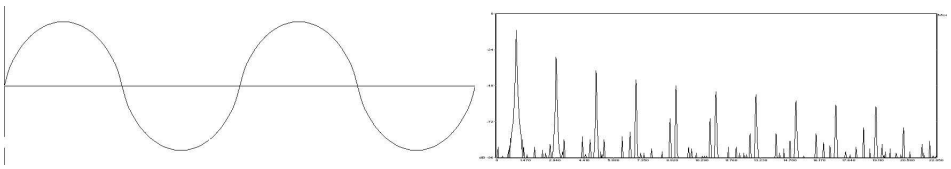

Figura 5.3: Onda triangular usada para realizar los experimentos y su correspondiente espectro utilizando una ventana Hanning.

Θ	m	n	h	θ	T_S	μ	T_E	$\sigma(\%)$
0	1	1	100	45	150	0.01	362	85.07
0	1	1	100	40	150	0.01	328	86.67
0	1	1	100	60	150	0.01	273	83.22
-0.7	1	1	100	40	150	0.01	251	87.70
-0.8	1	1	100	35	150	0.01	295	87.25
-0.7	1	0	100	45	200	0.01	272	87.29
0	1	0	100	45	200	0.01	332	87.41
-0.7	1	0	100	45	150	0.01	236	87.10
-0.7	1	0	100	45	150	0.005	318	**87.86**
-0.8	1	0	100	45	250	0.005	395	87.48

Tabla 5.2: Resultados del entrenamiento con una onda triangular.

La mayoría de errores cometidos se producen en el instante en que una nota comienza o termina de sonar, debido a las limitaciones que impone la resolución temporal.

5.4. Onda triangular

La onda triangular se puede representar matemáticamente mediante la ecuación 5.2. Contiene, por tanto, sólo los armónicos impares con amplitudes que decaen con una tasa de $1/p^2$, siendo p el número de armónico. Es bastante parecida a la sinusoidal, ya que la amplitud del primer armónico es mucho mayor que la del resto y para obtener un aspecto más triangular harían falta muchos más armónicos de los que hemos considerado a la hora de generarla mediante la síntesis con CSound. La figura 5.3 muestra la forma de onda que se ha usado para realizar las pruebas junto con su correspondiente espectro. Se trata de una onda triangular de amplitud unidad definida por sus 10 primeros armónicos, y con una resolución de 1024 puntos.

$$s(t) = \sum_{p=1}^{P} \left\{ \left(\frac{1}{2p-1} \right)^2 \text{sen} \left(\frac{2\pi(2p-1)ft}{N} \right) \right\} \tag{5.2}$$

Con este tipo de onda se han realizado menos pruebas que con la sinusoidal, ya que la calibración de los parámetros obtenidos con el timbre sinusoidal es extrapolable a este caso. Como se puede observar en la tabla 5.2, la variación de los parámetros (dentro de un rango pequeño) produce variaciones mínimas

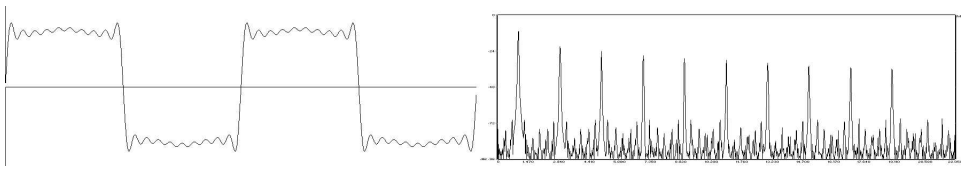

Figura 5.4: Onda cuadrada usada para realizar los experimentos y su correspondiente espectro utilizando una ventana Hanning.

Θ	m	n	h	θ	T_S	μ	T_E	$\sigma(\%)$
-0.7	1	1	100	45	150	0.01	336	85.13
-0.7	1	1	100	60	150	0.01	258	82.69
0	1	1	100	40	150	0.01	393	84.62
-0.7	1	1	100	45	150	0.005	419	85.71
-0.7	1	0	100	45	150	0.01	215	85.11
0	1	0	100	45	150	0.01	196	84.41
-0.7	1	0	100	45	150	0.005	202	85.69
-0.7	1	0	100	45	350	0.005	402	85.88
-0.7	1	0	100	45	350	0.005	427	**87.09**
-0.7	1	0	100	45	250	0.002	420	85.43

Tabla 5.3: Resultados del entrenamiento con una onda cuadrada

en los resultados finales. El mejor resultado da una tasa de aciertos sobre el conjunto de validación del 87.86 %.

5.5. Onda cuadrada

La onda cuadrada tiene la expresión de la ecuación 5.3. Por tanto, contiene sólo armónicos impares con amplitudes que decaen con una tasa de $1/p$. Este tipo de onda es algo más compleja que la triangular, ya que las amplitudes de los armónicos decrecen con una tasa menor. La figura 5.4 muestra la forma de onda que se ha usado para realizar las pruebas junto con su espectro correspondiente. Se trata de una onda cuadrada de amplitud unidad definida por sus 10 primeros armónicos, y con una resolución de 1024 puntos.

$$s(t) = \sum_{p=1}^{P} \left\{ \left(\frac{1}{2p-1} \right) \text{sen} \left(\frac{2\pi(2p-1)ft}{N} \right) \right\} \tag{5.3}$$

En la tabla 5.3 se puede observar el resultado de los entrenamientos. Los porcentajes de aciertos son algo inferiores a los de la onda triangular pero aun así, en la prueba con mejores resultados se ha conseguido una tasa de acierto del 87.10 %.

5.6. Onda diente de sierra

Este tipo de onda viene definida por la ecuación 5.4.

$$s(t) = \sum_{p=1}^{P} \left\{ \left(\frac{1}{p} \right) \text{sen} \left(\frac{2\pi pft}{N} \right) \right\} \tag{5.4}$$

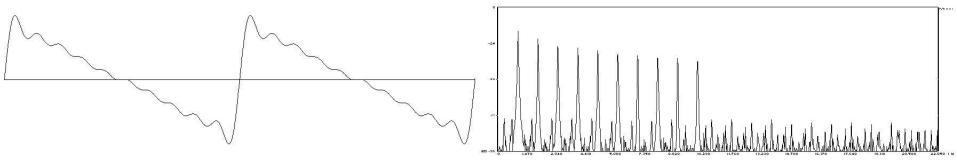

Figura 5.5: Onda diente de sierra usada para los experimentos y su correspondiente espectro utilizando una ventana Hanning.

Θ	m	n	h	θ	T_S	μ	T_E	$\sigma(\%)$
-0.7	1	1	100	45	150	0.01	242	78.91
0	1	1	100	45	150	0.01	326	80.07
0	1	1	100	45	150	0.005	244	81.22
0.2	1	1	100	40	150	0.005	244	80.07
0	1	0	100	45	150	0.005	224	82.43
0	1	0	100	45	150	0.005	226	81.86
0	1	0	120	45	150	0.005	277	82.43
0	1	0	100	55	150	0.005	182	81.09
0	1	0	100	40	150	0.005	224	82.24
0	1	0	100	45	150	0.01	402	81.92
0	1	0	100	45	250	0.003	374	**82.62**

Tabla 5.4: Resultados del entrenamiento con una onda diente de sierra

Por tanto, contiene todos los parciales armónicos con una amplitud que decae con una tasa de 1/p. Se trata de un tipo de onda rica en armónicos, y por lo tanto, con mayor dificultad para la red. La figura 5.5 muestra la forma de onda usada para las pruebas, junto con su correspondiente espectro. Se trata de una onda diente de sierra de amplitud unidad definida por sus 10 primeros armónicos, y con una resolución de 1024 puntos.

Los porcentajes de acierto que muestra la tabla 5.4 son, como cabía esperar debido a su riqueza armónica, bastante inferiores que los obtenidos anteriormente con el resto de tipos de onda. Los mejores resultados dan una tasa de acierto del 82.62 %.

Nótese que los mejores resultados se han obtenido con un umbral Θ mayor que con el resto de ondas empleadas hasta el momento. Esto se debe a que la onda diente de sierra es la de mayor riqueza armónica de los experimentos, y esto puede inducir a la sobredetección. Aumentando Θ conseguimos ser más restrictivos con las notas detectadas, mejorando los resultados obtenidos.

5.7. Onda de instrumento de viento

El último experimento se ha realizado sintetizando el sonido de un clarinete con CSound utilizando técnicas de modelado físico. Se trata de probar con una forma de onda menos sencilla que las anteriores, que provenga de la síntesis que se puede encontrar en un sintetizador comercial para lograr un sonido creíble de un instrumento real.

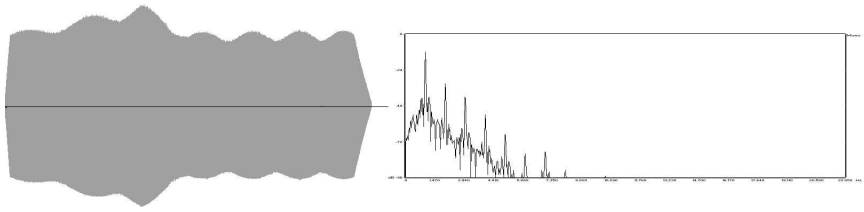

Figura 5.6: Timbre clarinete: En este caso se muestra la envolvente de amplitud de una nota para ilustrar el hecho de que es un timbre que posee dinámica. A la derecha se muestra su espectro obtenido con una ventana Hanning.

Θ	m	n	h	θ	T_S	μ	T_E	$\sigma(\%)$
-0.7	1	0	100	45	150	0.01	327	83.98
-0.7	1	0	100	45	150	0.005	210	84.63
-0.7	1	1	100	45	150	0.005	210	83.68
-0.7	1	0	100	45	150	0.005	230	84.06
-0.7	1	0	100	45	150	0.002	285	84.67
-0.7	1	0	100	45	150	0.001	275	85.15
-0.7	1	0	100	45	150	0.0007	750	**85.92**
-0.7	1	0	100	45	150	0.0001	1508	83.49
-0.7	1	0	100	45	150	0.0005	717	83.90
0	1	0	100	45	150	0.0007	497	83.70
0	1	0	100	45	150	0.001	354	84.38

Tabla 5.5: Resultados del entrenamiento con el timbre clarinete.

Se ha escogido el clarinete porque sus tiempos de ataque y liberación son cortos, porque hemos querido evitar percusivos y porque, dentro de los instrumentos reales, no presenta grandes estridencias y modulaciones que pudieran complicar la detección en esta fase temprana del estudio.

Para la síntesis de este instrumento se ha utilizado CSound con dos técnicas: una utilizando el *opcode*[3] wgclar, que es un modelo virtual de clarinete, y la otra mediante la simulación de la acústica de un clarinete partiendo de la generación de ondas más sencillas.

La figura 5.7 muestra la forma de onda y el espectro del instrumento wgclar. Para el caso del timbre clarinete (figura 5.6), se muestra la envolvente en amplitud[4] de una nota y su espectro en el instante central de esta misma nota. Este timbre tiene dinámica aunque, como se puede ver, su envolvente es bastante uniforme.

En las tablas 5.5 y 5.6 se muestran los resultados obtenidos tras las pruebas, tanto para el clarinete sintetizado como para el clarinete generado por el *opcode* wgclar. Como se puede observar en las tablas 5.5 y 5.6, los resultados son com-

[3]Denominación que usa el lenguaje CSound para denominar a sus operadores: *operator code* (opcode).

[4]Desde el punto de vista acústico se puede definir la envolvente como la representación de la intensidad sonora percibida en función del tiempo. Gráficamente corresponde a la línea imaginaria que une los puntos de máxima amplitud de las oscilaciones de la onda sonora.

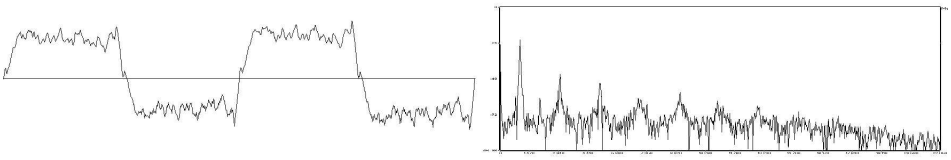

Figura 5.7: Timbre wgclar: Forma de onda y su correspondiente espectro utilizando una ventana Hanning.

Θ	m	n	h	θ	T_S	μ	T_E	$\sigma(\%)$
-0.7	1	0	100	45	150	0.005	270	86.25
-0.7	1	0	100	45	150	0.001	566	86.49
-0.7	1	0	100	45	150	0.0007	733	86.36
-0.7	1	1	100	45	150	0.005	315	85.63
0	1	0	100	45	150	0.001	279	**87.46**
0	1	0	100	45	150	0.0007	336	85.80
0	1	0	100	45	150	0.005	199	86.08
0	1	0	100	45	150	0.001	302	86.85

Tabla 5.6: Resultados del entrenamiento con el timbre wgclar.

parables a los de las ondas sintéticas, por lo que en principio se puede pensar que el método podría extenderse fácilmente a la detección polifónica sobre timbres reales que tengan unas características uniformes de ataque, sostenimiento y liberación, es decir, sobre instrumentos cuyo patrón espectral sea aproximadamente uniforme en el tiempo.

5.8. Limitaciones conocidas y posibles mejoras

Hay una serie de factores que inciden directamente sobre el entrenamiento y que limitan en parte las posibilidades de la técnica empleada. La figura 5.8 muestra un ejemplo de un espectrograma, que se corresponde con una escala que va desde Sol$\sharp_0$ hasta Do$\sharp_6$, recorriendo todo el rango de bandas usado en esta primera configuración. La señal se ha generado a partir de una onda cuadrada.

Como se puede observar en la figura 5.8, las componentes de amplitud de las bandas de frecuencia que quedan fuera de rango, en vez de perderse, se reflejan como si se tratase de un espejo. Este efecto se conoce como *aliasing* y se produce al considerar un rango de frecuencias finito, lo que produce armónicos artificiales menores que los que corresponderían.

En este caso concreto no parece un problema grave, pero cuando hay acordes en la melodía, este factor influye directamente sobre el resultado. Existe además un problema añadido; aunque estas amplitudes no se reflejaran, para las notas de frecuencias altas perderíamos información sobre los armónicos, pues la frecuencia fundamental puede estar muy cercana al límite superior de frecuencias.

Se ha comprobado que la mayoría de errores se corresponde a las notas muy agudas y a las transiciones, es decir, cuando comienzan o acaban las notas. Esto

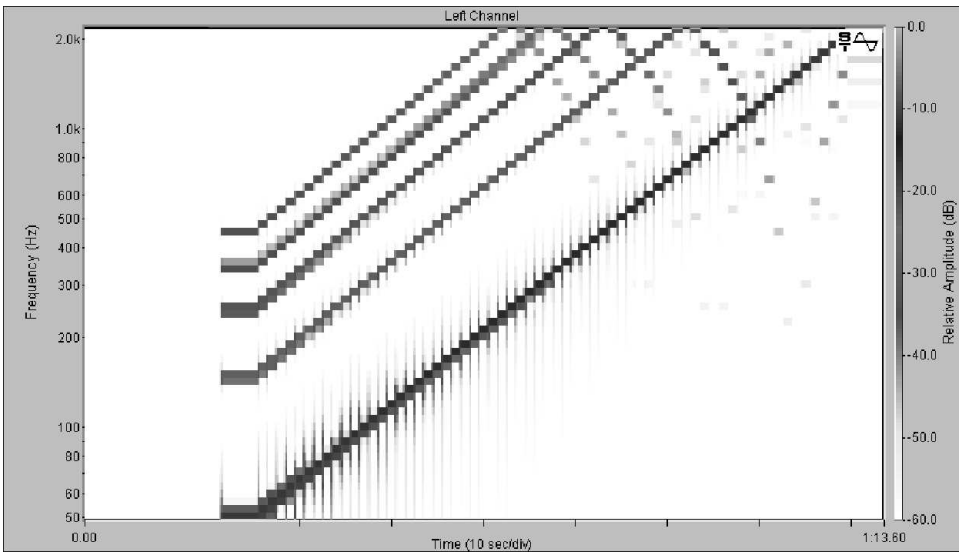

Figura 5.8: Espectrograma resultante de una escala con el timbre de onda cuadrada.

se debe, en parte, a que una resolución temporal $\Delta t = 232.2$ ms provoca que se pierdan algunas notas cortas.

Hemos visto que los parámetros de espectrograma usados en esta configuración permiten mostrar la viabilidad de esta propuesta, pero plantean una serie de limitaciones a la hora de conseguir resultados óptimos. Intuitivamente, podemos pensar que es posible mejorar la precisión del sistema si:

- Se reduce el *aliasing*.
- Se aumenta el número de bandas de frecuencia, para poder permitir notas más agudas y para que las amplitudes de las frecuencias altas se conserven inalteradas.
- Se reduce el valor de Δt.

Vamos a intentar resolver estos problemas en la Configuración II cambiando los parámetros usados para obtener el espectrograma y, como consecuencia, la parte de configuración de la red dependiente de estos parámetros.

6. CONFIGURACIÓN II

Hemos visto que los principales problemas detectados en la Configuración I se podrían resolver aumentando el rango de bandas de frecuencia en varias octavas (para perder la menor cantidad posible de información sobre los armónicos superiores y para reducir el *aliasing*) y reduciendo el valor de Δt.

La única diferencia en la estructura de la red con respecto a la Configuración I es el número de neuronas de entrada y de salida, debido a la distinta frecuencia de muestreo empleada, como veremos más adelante. Los resultados de los experimentos con la Configuración II mostrarán el porcentaje de acierto tanto de eventos como de notas.

Para esta configuración sólo probaremos con las señales sinusoidal, diente de sierra y wgclar, como señales más representativas de la Configuración I. La onda sinusoidal es la más sencilla y la triangular es la rica en armónicos. Vamos a tomar también uno de los clarinetes (wgclar) como ejemplo de señal cercana a un instrumento real.

Además, vamos a incluir en los experimentos un sonido de cuerda percutida, que es un timbre con envolventes en amplitud y frecuencia bastante acusadas. Los instrumentos de la Configuración I carecen de esta característica, a excepción del timbre clarinete que posee una pequeña envolvente.

6.1. Parámetros del espectrograma

Disminuyendo el divisor usado para obtener la frecuencia de muestreo operativa (es decir, disminuyendo el tamaño de la ventana) se puede aumentar el límite máximo de frecuencia. Esto también influye directamente sobre Δt y sobre la distribución de energía de los armónicos.

Para esta configuración se ha probado con una división (*decimation ratio*) de 2. La frecuencia de muestreo original es de $f_s = 44100$ Hz, pero al dividir entre 2 el número de muestras, la frecuencia de muestreo operativa queda en $f_s = 22050$ Hz. Debido al teorema de Nyquist, la frecuencia máxima representable es $f_s/2 = 11025$ Hz.

Por lo tanto, usando una escala logarítmica para las bandas de entrada a la red y teniendo la primera banda de frecuencia centrada en 50 Hz, el número de bandas es $b = 94$, con un rango de que va desde 50 Hz hasta 11025 Hz, correspondiente al rango de notas que va desde $\text{Sol}\sharp_0$ hasta Fa_8 (banda centrada en 10 600 Hz). Este rango abarca unas ocho octavas. Conviene recordar que, a mayor rango, mayor coste computacional en la fase de entrenamiento y en la de reconocimiento.

	Configuración II		Configuración I
	notas	eventos	eventos
seno	94.98 %	94.81 %	89.01 %
sierra	91.79 %	92.13 %	82.62 %
wgclar	91.44 %	92.41 %	87.46 %

Tabla 6.1: La tabla muestra los porcentajes de acierto de detección de notas y de eventos con la Configuración II y el porcentaje de acierto de eventos obtenido con la Configuración I. respectivamente.

Para el caso que nos ocupa, $N = 2048$, $S = 50\,\%$ y $f_s = 22050$ Hz, por lo que la resolución temporal es $\Delta t = 46.4$ ms. Con estos datos, la STFT proporciona un conjunto de amplitudes para 1024 frecuencias con una resolución espectral (separación entre frecuencias) de $f_R = f_s/N = 10.77$ Hz.

6.2. Sobre la convergencia

Con estos parámetros de espectrograma, el entrenamiento converge mucho antes que con los de la Configuración I. En los experimentos de la Configuración II se ha observado que, en promedio, el algoritmo converge en la época 40, mientras que en la Configuración I lo hace aproximadamente en la 150.

Esto se explica porque al tener un menor Δt, con el mismo número de melodías tendremos realmente más muestras de entrenamiento. También se puede deducir que para la red es más fácil aprender al paliarse en parte los problemas provocados por el *aliasing* y ampliarse el límite superior de frecuencias que tenía la Configuración I.

6.3. Resultados de los experimentos

En la Configuración I hemos ajustado los parámetros de la red para obtener los mejores resultados posibles. En la Configuración II, nuestro propósito es ajustar los parámetros del espectrograma para reducir las carencias de la Configuración I en cuanto a Δt y límite de bandas de frecuencia, por lo que sólo se han realizado las pruebas con los mejores parámetros de la red obtenidos en los experimentos de la Configuración I.

En los entrenamientos se han ido alternando melodías en el conjunto de entrenamiento y validación, de manera que estos conjuntos han ido variando. Los porcentajes de acierto que muestra la tabla de resultados 6.1 corresponden a la media de los porcentajes de acierto obtenidos tras realizar las pruebas rotando melodías entre estos conjuntos.

Como puede observarse, el porcentaje de acierto ha aumentado de media un 7 % con respecto a los resultados de la Configuración I.

La figura 6.1 muestra un ejemplo de los resultados obtenidos al transcribir una melodía generada con el timbre wgclar. Arriba se muestra la partitura de la melodía original. En el centro, esta partitura mostrada como en una pianola (*piano-roll*) de un secuenciador. Abajo se compara la BDP original y la BDP obtenida tras reconocer el espectrograma de la melodía.

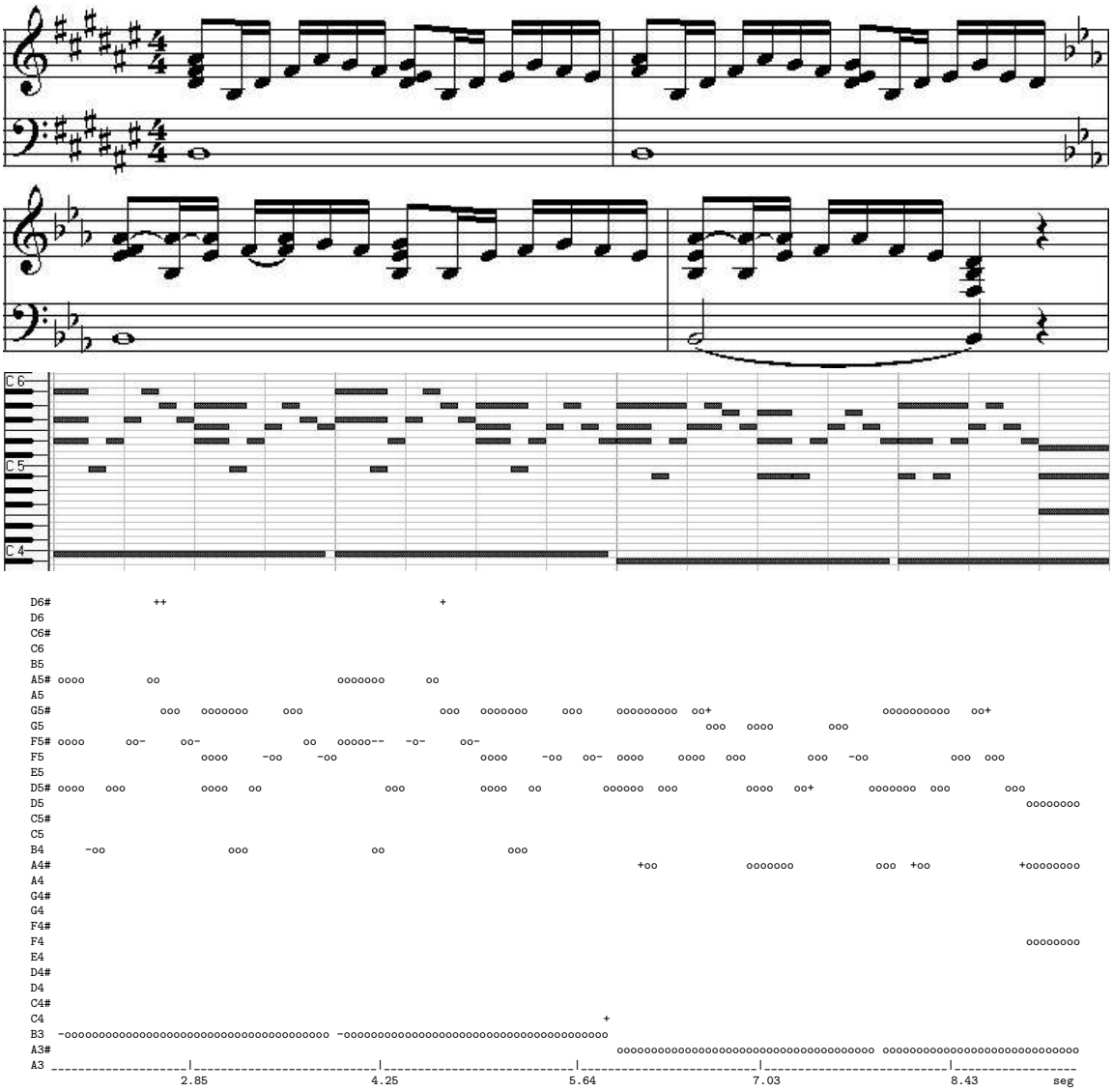

Figura 6.1: Evolución temporal de la detección de notas para una determinada melodía usando el timbre wgclar. Arriba: la partitura original; centro: la melodía original mostrada como una pianola (*piano-roll*) de un secuenciador; abajo: la BDP obtenida tras la salida de la red, comparada con la BDP deseada (es decir, con la salida del secuenciador). El acierto en la detección es del 94.3 %, la proporción de falsos positivos con respecto al número de eventos detectados es del 2.3 %, y para falsos negativos, del 3.4 %. Para notas, todas ellas han sido detectadas y sólo 3 notas cortas de las detectadas corresponden a falsos positivos. Notación: '`o`': Eventos detectados correctamente, '`+`': falsos positivos, y '`-`': falsos negativos. La melodía empleada no se encuentra ni en el conjunto de entrenamiento ni en el de validación.

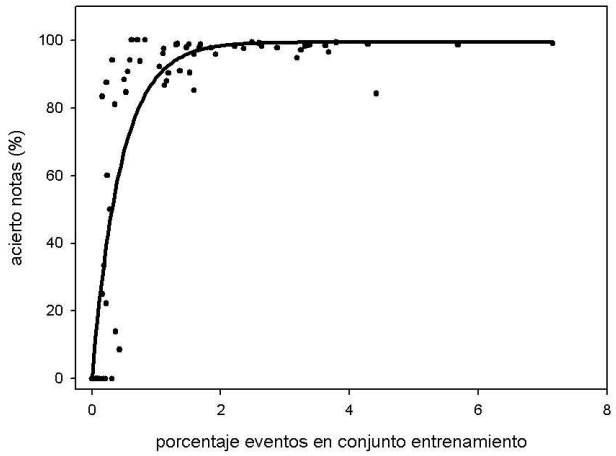

Figura 6.2: Porcentaje de acierto en notas frente al porcentaje del número de eventos en el conjunto de entrenamiento.

Se puede observar que la mayor parte de los errores cometidos se produce en los transitorios. También se producen algunos errores en notas muy altas.

En la tabla 6.2 se muestran las estadísticas de las pruebas realizadas con el timbre wgclar. En total se usaron 78 048 eventos en el conjunto de entrenamiento. El porcentaje de acierto de eventos es del 92.41 % y el de notas, del 91.44 % (como muestra la tabla 6.1). Se puede observar el número de eventos que hay para cada nota en los espectros del conjunto de entrenamiento. Tras la fase de reconocimiento, se muestra el número de aciertos y de errores de cada altura, junto con el porcentaje de acierto de notas.

Podemos ver en la tabla 6.2 que los principales errores del sistema se producen en las notas muy altas o muy bajas. También se puede observar que el porcentaje de acierto suele depender del número de eventos en el conjunto de entrenamiento. Esto se ve gráficamente en la figura 6.2, que muestra la correlación entre el número de eventos de cada altura en el entrenamiento y la capacidad de reconocimiento de esas alturas. Los puntos han sido ajustados a una exponencial para mostrar su correlación ($r = 0.94$).

Esto datos nos hacen pensar que aumentando el número de muestras de entrenamiento pueden conseguirse mejores resultados. Nótese que las notas no tienen una distribución equiprobable en el conjunto de entrenamiento ya que en las melodías de entrenamiento (y en música en general) las notas no tienen la misma probabilidad (por ejemplo, un Do suele ser más probable que un Do♯).

El sistema es robusto en el intervalo [Si_2,Si_6]. Un piano usa desde la nota La_{-1} hasta la nota Do_7, por lo que los errores fuera de este rango no son muy relevantes en la práctica. Además, las notas correspondientes a la octava -1 prácticamente no se usan.

Se ha calculado también el error por longitud de nota. Las notas muy cortas (de longitud igual a uno o dos eventos) tienen más probabilidades de sufrir

	Entrenamiento		Reconocimiento		
Nota	Eventos	Eventos(%)	Notas correctas	Notas error	Acierto(%)
FA_8	23	0.03	0	2	0.00
MI_8	11	0.01	0	1	0.00
$RE\sharp_8$	0	0.00	0	0	
RE_8	11	0.01	0	1	0.00
$DO\sharp_8$	0	0.00	0	0	
DO_8	11	0.01	0	1	0.00
SI_7	11	0.01	0	1	0.00
$LA\sharp_7$	11	0.01	0	1	0.00
LA_7	22	0.03	0	2	0.00
$SOL\sharp_7$	0	0.00	0	0	
SOL_7	63	0.08	0	4	0.00
$FA\sharp_7$	44	0.06	0	2	0.00
FA_7	144	0.18	0	10	0.00
MI_7	46	0.06	0	4	0.00
$RE\sharp_7$	57	0.07	0	6	0.00
RE_7	167	0.21	0	16	0.00
$DO\sharp_7$	67	0.09	0	9	0.00
DO_7	337	0.43	3	32	8.57
SI_6	181	0.23	14	2	87.50
$LA\sharp_6$	285	0.37	5	31	13.89
LA_6	589	0.75	30	2	93.75
$SOL\sharp_6$	390	0.50	38	5	88.37
SOL_6	1318	1.69	75	1	98.68
$FA\sharp_6$	438	0.56	58	6	90.62
FA_6	1181	1.51	83	1	98.81
MI_6	1501	1.92	91	4	95.79
$RE\sharp_6$	932	1.19	74	8	90.24
RE_6	2027	2.60	113	1	99.12
$DO\sharp_6$	871	1.12	77	2	97.47
DO_6	2966	3.80	146	1	99.32
SI_5	1240	1.59	86	15	85.15
$LA\sharp_5$	1844	2.36	116	3	97.48
LA_5	2059	2.64	105	2	98.13
$SOL\sharp_5$	2618	3.35	121	2	98.37
SOL_5	4441	5.69	199	3	98.51
$FA\sharp_5$	905	1.16	65	9	87.84
FA_5	3448	4.42	144	27	84.21
MI_5	1440	1.85	126	3	97.67
$RE\sharp_5$	3351	4.29	166	2	98.81
RE_5	2645	3.39	152	2	98.70
$DO\sharp_5$	2489	3.19	125	7	94.70
DO_5	5591	7.16	195	2	98.98
SI_4	1039	1.33	87	1	98.86
$LA\sharp_4$	2534	3.25	129	4	96.99
LA_4	1945	2.49	139	1	99.29
$SOL\sharp_4$	2582	3.31	113	2	98.26
SOL_4	2876	3.68	135	5	96.43
$FA\sharp_4$	1067	1.37	60	6	90.91
FA_4	2834	3.63	114	2	98.28
MI_4	1243	1.59	70	3	95.89

	Entrenamiento		Reconocimiento		
Nota	Eventos	Eventos(%)	Notas correctas	Notas error	Acierto (%)
$RE\sharp_4$	1733	2.22	55	1	98.21
RE_4	1026	1.31	72	1	98.63
$DO\sharp_4$	1300	1.67	45	1	97.83
DO_4	2245	2.88	88	2	97.78
SI_3	283	0.36	17	4	80.95
$LA\sharp_3$	821	1.05	35	3	92.11
LA_3	556	0.71	29	0	100.00
$SOL\sharp_3$	879	1.13	26	4	86.67
SOL_3	1185	1.52	47	5	90.38
$FA\sharp_3$	461	0.59	16	1	94.12
FA_3	1151	1.47	45	1	97.83
MI_3	486	0.62	22	0	100.00
$RE\sharp_3$	645	0.83	17	0	100.00
RE_3	253	0.32	16	1	94.12
$DO\sharp_3$	417	0.53	11	2	84.62
DO_3	866	1.11	24	1	96.00
SI_2	128	0.16	5	1	83.33
$LA\sharp_2$	174	0.22	2	7	22.22
LA_2	77	0.10	0	6	0.00
$SOL\sharp_2$	151	0.19	2	4	33.33
SOL_2	190	0.24	15	10	60.00
$FA\sharp_2$	80	0.10	0	4	0.00
FA_2	218	0.28	3	3	50.00
MI_2	92	0.12	0	2	0.00
$RE\sharp_2$	123	0.16	1	3	25.00
RE_2	30	0.04	0	2	0.00
$DO\sharp_2$	93	0.12	0	2	0.00
DO_2	239	0.31	0	7	0.00
SI_1	10	0.01	0	1	0.00
$LA\sharp_1$	27	0.03	0	2	0.00
LA_1	11	0.01	0	1	0.00
$SOL\sharp_1$	28	0.04	0	2	0.00
SOL_1	45	0.06	0	7	0.00
$FA\sharp_1$	10	0.01	0	1	0.00
FA_1	12	0.02	0	1	0.00
MI_1	12	0.02	0	1	0.00
$RE\sharp_1$	10	0.01	0	1	0.00
RE_1	12	0.02	0	1	0.00
$DO\sharp_1$	10	0.01	0	1	0.00
DO_1	11	0.01	0	1	0.00
SI_0	11	0.01	0	1	0.00
$LA\sharp_0$	10	0.01	0	1	0.00
LA_0	11	0.01	0	1	0.00
$SOL\sharp_0$	21	0.03	0	2	0.00

Tabla 6.2: Estadísticas de entrenamiento y reconocimiento de todas las melodías con el timbre wgclar. En la primera columna de la tabla se muestra la nota. La segunda y la tercera hacen referencia al número de eventos en el conjunto de entrenamiento y al porcentaje que constituyen en ese conjunto. La cuarta, quinta y sexta columna muestran los resultados de reconocimiento. Representan el número de notas correctas, el número de notas erróneas (falsos positivos y falsos negativos) y el porcentaje de acierto.

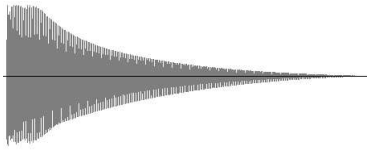

Figura 6.3: Envolvente temporal de la cuerda percutida usada en los experimentos.

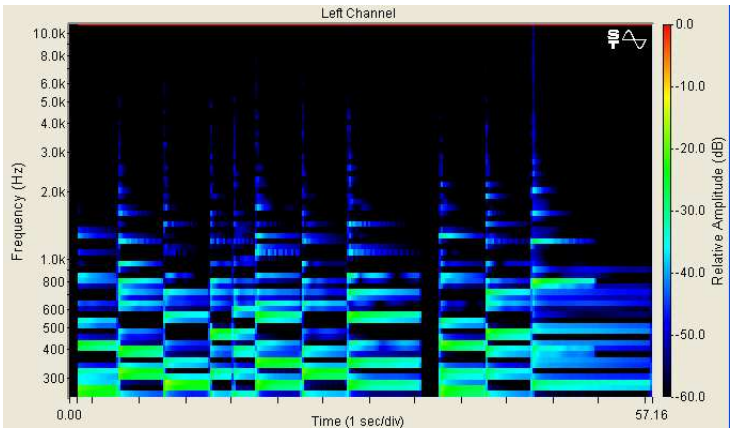

Figura 6.4: Ejemplo de espectrograma de una melodía generada con el timbre de cuerda percutida.

errores. El porcentaje de notas erróneas cuya longitud es igual a un evento constituye el 18.6 % del total de errores por longitud, mientras que aquellas cuya longitud es igual a dos eventos, el 12.4 %.

Adicionalmente, se han realizado experimentos usando una forma de onda proveniente de un instrumento de cuerda percutida. Este tipo de instrumentos tiene la característica de poseer una envolvente muy dinámica (ver figura 6.3).

Los resultados no son muy prometedores. Con este timbre, hemos obtenido alrededor de un 35 % de porcentaje de acierto en eventos. Esto se explica por la envolvente que posee el instrumento. En la figura 6.4 podemos ver un ejemplo del espectrograma de una melodía generada con este timbre. Vemos que los armónicos no son estables desde el inicio hasta el final de las notas, sino que varían en amplitud conforme avanza el tiempo y algunos se hacen imperceptibles antes de finalizar la duración de la nota. Esto provoca que la red no tenga suficiente información temporal (suficientes ventanas) para captar todo el patrón espectral de este timbre.

6.4. Validación cruzada

El objetivo de este experimento es ver lo similares que son los pesos de la red para los diferentes timbres considerados. Para esto, se ha intentado que la red reconozca espectrogramas de una forma de onda tras haberla entrenado con espectrogramas de otra forma de onda distinta. Las pruebas se han realizado sobre tres tipos de formas de onda diferentes y se pueden ver los resultados en la tabla 6.3.

	seno	sierra	wgclar
seno	95 ; 95 %	56 ; 48 %	65 ; 55 %
sierra	56 ; 60 %	92 ; 92 %	72 ; 65 %
wgclar	56 ; 60 %	50 ; 49 %	92 ; 91 %

Tabla 6.3: Resultados de la detección cruzada. Las filas corresponden a los timbres usados en el entrenamiento, y las columnas a los timbres usados en el reconocimiento. El contenido de la tabla muestra los porcentajes de acierto de detección de eventos (antes del punto y coma) y de notas (después del punto y coma).

Los porcentajes de acierto sobre timbres distintos de los que fueron usados para entrenar la red van desde el 48 % hasta el 72 %. Son claramente peores, pero aun así puede decirse que se han detectado dos o tres de cada cuatro notas. Estos resultados muestran que la red aprende específicamente a identificar patrones espectrales mostrados durante el entrenamiento. Aunque algo común hay en los distintos timbres que hace que la red sí que sea capaz de identificar que algo está pasando en el espectrograma y que ese algo son notas.

No parece que puedan establecerse unos pesos "universales" para la red, pero sí podrían crearse familias de instrumentos con patrones espectrales parecidos y reconocerlos todos con los mismos pesos.

6.5. Limitaciones conocidas y posibles mejoras

Como hemos podido comprobar, este tipo de red funciona bien con los timbres cuyo patrón espectral es aproximadamente estático, pero no con los que tienen envolventes espectrales y de amplitud que evolucionan en periodos mayores de lo que pueden abarcar las ventanas de la red.

El sistema considera de la misma manera los datos del espectrograma en el ataque de la nota y los de la liberación de la misma, cuando, como podemos ver en la figura 6.4, estos datos son variables en el caso de ondas cuyo patrón espectral varía en función del tiempo.

Por lo tanto, para este tipo de timbres parece necesario probar con otros tipos de red que no tengan en cuenta sólo un pequeño intervalo temporal del espectrograma definido por un número de ventanas adyacentes, sino que conserven información más a medio plazo, tales como las redes neuronales recurrentes [17].

7. DISCUSIÓN Y CONCLUSIONES

El objetivo de este trabajo ha sido explorar una nueva vía de investigación dentro del campo de la extracción de notas de una señal musical polifónica en formato digital. Lo que ha pretendido, más que desarrollar un sistema de detección de alturas completo y fiable, es poner a prueba la utilidad de las redes neuronales dinámicas no recurrentes para la transcripción musical polifónica.

No somos expertos en modelos de percepción auditiva, así que hemos escogido un modelo conexionista para intentar que una red de tipo TDNN sea capaz de resolver el problema a partir de espectrogramas sin usar ningún modelo psicoacústico de referencia. Tampoco se ha efectuado ningún preproceso previo, ya que aplicar cambios a un espectrograma puede desvirtuar o destacar la información contenida en él. Por lo tanto, a diferencia de otras técnicas [45], hemos introducido el espectrograma directamente en la red para que ésta intente encontrar de manera automática las relaciones entre alturas presentes en el espectro de una señal.

Las entradas a la red de tipo TDNN han sido una serie de amplitudes de determinadas bandas de espectrogramas obtenidos a partir de melodías polifónicas monotímbricas. Estas melodías han sido generadas mediante el sintetizador virtual CSound utilizando formas de onda sencillas y partituras provenientes de ficheros en formato MIDI.

La salida de la red es la codificación de las notas musicales en el rango cubierto por las bandas del espectrograma para cada uno de los instantes en los que éste se ha obtenido a partir del fichero de sonido digital que contenía la melodía.

7.1. Análisis de los resultados

Hemos hecho los primeros experimentos con una configuración de parámetros de espectrograma que ha demostrado la validez del enfoque y se han identificado los posibles aspectos a mejorar, así que se han repetido los experimentos que mostraban los mejores resultados en las primeras pruebas con otra configuración distinta. Con esta nueva configuración, ha mejorado el porcentaje de acierto en un 7 % aproximadamente.

Para timbres correspondientes a ondas sencillas se han obtenido buenos resultados en la detección, con porcentajes de acierto superiores al 92 % llegando incluso al 95 % tanto para detección de eventos como para detección de notas. La mayor parte de los errores de eventos se producen al comienzo y al final de las notas. La mayor parte de errores de notas se corresponde a notas muy cortas (de longitud igual a 1 o 2 eventos), muy altas (mayores que Si_6) o muy bajas (menores que Si_2).

En lo que respecta a la carga computacional, ésta no se ha revelado como crítica, al menos con la tecnología disponible, debido a que el algoritmo de entrenamiento converge en pocas épocas (unas 40 en el caso de la Configuración II).

7.2. Trabajo futuro

Para incrementar la tasa de aciertos se pueden añadir más muestras de entrenamiento de manera que el número de eventos por altura sea equiprobable, ya que el sistema parece que flaquea en las notas de las que dispone de pocas muestras. La evolución de los resultados sugiere que es posible que ampliando el conjunto de aprendizaje se logre una mayor precisión.

Una vez logrados los objetivos iniciales, el trabajo debería continuar en la dirección de experimentar con formas de onda más complejas, modulaciones temporales y timbres de instrumentos naturales, para explorar las fronteras de aplicación de esta metodología.

Los experimentos sugieren la posibilidad de estudiar grupos tímbricos como maderas, cuerdas, metales, voz, etc., de forma que se pueda adiestrar a la red con distintos timbres de un mismo grupo para ver si el sistema mantiene después su eficiencia en la detección de cualquier timbre del mismo grupo.

Es difícil pensar que el sistema soporte bien el intercambio entre grupos pero podría incluirse una primera fase de reconocimiento del timbre para ordenar a la red que cargara los pesos adecuados en función de dicho timbre para proceder a la detección.

Otra línea de trabajo es la de hacer un post-procesamiento de la señal resultante para que, sometida a una cuantización[1] temporal, se eliminen los efectos de los transitorios, que son los que más confunden a la red. También se puede aplicar conocimiento musical a la salida, descartando las notas que son poco probables en teoría musical o mediante el uso de inferencia probabilística.

En cuanto a los timbres con envolventes espectrales muy dinámicas, se hace necesario estudiar otros tipos de redes que tengan en cuenta la historia del espectrograma a medio plazo, tales como las redes neuronales recurrentes [17].

En cualquier caso, los resultados de este trabajo abren infinidad de lineas prometedoras para seguir investigando la transcripción musical polifónica mediante redes neuronales.

[1] La cuantización es la acción, en una representación simbólica, de ajustar las notas a los tiempos que le corresponden.

ÍNDICE DE SÍMBOLOS IMPORTANTES

BIBLIOGRAFÍA

[1] D. Arnold, L. Balkan, S. Meijer, R.L. Humphreys, y L. Sadle. *Machine Translation: an Introductory Guide*. NCC Blackwell, Oxford, 1994.

[2] MIDI Manufacturers Association y the Japan MIDI Standards Committee. General MIDI System Level 1 specification, Septiembre 1991.

[3] J.A. Bilmes. Timing is of the essence: Perceptual and computational techniques for representing, learning, and reproducing expressive timing in percussive rhythm. Tesis de maestría, MIT, 1993.

[4] B.P. Bogert, M.J.R. Healy, y J.W. Tukey. The quefrency alanysis of time series for echoes: cepstrum, pseudo-autocovariance, cross-cepstrum and saphe cracking. En Murray Rosenblatt, editor, *Proceedings of the Symposium on Time Series Analysis*, páginas 209–243. John Wiley and Sons, 1963.

[5] R. Boulanger. *The CSound Book*. MIT Press, Cambridge, Massachusetts, 1999.

[6] A.S. Bregman. *Auditory scene analysis*. MIT Press, 1990.

[7] G.J. Brown y M. Cooke. Perceptual grouping of musical sounds: A computational model. *J. of New Music Research*, 23:107–132, 1994.

[8] J. Brown. Calculation of a "narrowed" autocorrelation function. *J. Acoust. Soc. Am.*, páginas 1595–1601, Abril 1989.

[9] J. Brown y B. Zhang. Musical frequency tracking using the methods of conventional and 'narrowed' autocorrelation. *J. Acous. Soc. Am.*, 89:2346–2354, Mayo 1991.

[10] C. Chafe y D. Jaffe. Source separation and note identification in polyphonic music. En *Proceedings of the IEEE International Conference on Acoustics Speech and signal processing*, volumen 2, página 25.6.125.6.4, Tokio, Abril 1986.

[11] C. Chafe, D. Jaffe, K. Kashima, B. Mont-Reynaud, y J. Smith. Techniques for note identification in polyphonic music. En *Proceedings of the IEEE International Conference of Computer Music*, páginas 399–405, Vancouver, 1985.

[12] C. Chafe, B. Mont-Reynaud, y L. Rush. Toward an intelligent editor of digital audio: Recognition of musical constructs. *Computer Music Journal*, 6(1), 1982.

[13] R.A. Day. *Cómo escribir y publicar trabajos científicos*. Prensas Universitarias de Zaragoza, Zaragoza, 1996.

[14] B. Doval y X. Rodet. Estimation of fundamental frequency of musical signals. En *Proceedings of the International Conference on Acoustics Speech and signal processing*, 1991.

[15] B. Doval y X. Rodet. Fundamental frequency estimation and tracking using maximum likelihood harmonic matching and HMMs. En *Proceedings of the ICASSP*, páginas 27–30, Abril 1993.

[16] D.P.W. Ellis. *Prediction-driven computational auditory scene analysis*. Tesis doctoral, MIT, 1996.

[17] J.L. Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.

[18] D. Godsmark y G.J. Brown. A blackboard architecture for computational auditory scene analysis, 1999. Speech communication.

[19] M. Goto. A robust predominant-f0 estimation method for real-time detection of melody and bass lines in CD recordings. En *IEEE International Conf. on Acoust., Speech, and Signal Processing*, Estambul, Turquía, Junio 2000.

[20] E.R. Harold. *XML Bible*. IDG Books, 1999.

[21] M. Hawley. *Structure of sound*. Tesis doctoral, MIT, Septiembre 1993.

[22] D. Hermes. *Pitch Analysis*, capítulo Visual Representations of Speech Analysis. John Wiley and Sons, New York, 1992.

[23] J. Hertz, A. Krogh, y R.G. Palmer. *Introduction to the theory of Neural Computation*. Addison-Wesley, Redwood (CA), 1991.

[24] W.J. Hess. *Algorithms and Devices for Pitch Determination of Speech-Signals*. Springer-Verlag, Berlin, 1983.

[25] D.R. Hush y B.G. Horne. Progress in supervised neural networks. *IEEE Signal Processing Magazine*, 1(10):8–39, 1993.

[26] W.J. Hutchins y H.L. Somers. *An Introduction to Machine Translation*. Academic Press, London, 1992.

[27] E.C. Ifeachor y B.W. Jervis. *DSP–practical approach*. Addison-Wesley Publishing Co., 1993.

[28] W.H. Inmon. *Building the Data Warehouse*. Ed. John Wiley and Sons, 1992.

[29] K. Kashino, K. Nakadai, T. Kinoshita, y H. Tanaka. Application of Bayesian probability network to music scene analysis. En *Proceedings of the International Joint Conference on AI, CASA workshop*, 1995.

[30] K. Kashino y H. Tanaka. A sound source separation system with the ability of automatic tone modeling. En *Proceedings of the International Computer Music Conference*, 1993.

[31] H. Katayose y S. Inokuchi. The Kansei music system. *Computer Music Journal*, 13(4):72–77, 1989.

[32] R. Kimball. *The Data Warehouse Lifecycle Toolkit: Tools and Techniques for Designing, Developing, and Deploying Data Warehouses*. Ed. John-Wiley and Sons, 1998.

[33] A. Klapuri. Automatic transcription of music. Tesis de maestría, Tampere University of Technology, Department of Information Technology, 1998.

[34] A. Klapuri. Sound onset detection by applying psychoacoustic knowledge. En *IEEE International Conf. on Acoust., Speech, and Signal Processing*, Phoenix, Arizona, 1999.

[35] A. Klapuri. Multipitch estimation and sound separation by the spectral smoothness principle. En *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2001.

[36] A. Klapuri, A. Eronen, J. Sëppanen, y T. Virtanen. Automatic transcription of music. En *Symposium on Stochastic Modeling of Music*, Ghent, Bélgica, Octubre 2001.

[37] A. Klapuri, A. Eronen, J. Sëppanen, y T. Virtanen. Automatic transcription of musical recordings. En *Consistent and Reilable Acoustic Cues Workshop (CRAC)*, Aalborg, Dinamarca, Septiembre 2001.

[38] A. Klapuri y T. Virtanen. Separation of harmonic sounds using multipitch analysis and iterative parameter estimation. En *IWW Workshop on Applications of Signal Processing to Audio and Acoustics*, 2001.

[39] A. Klapuri, T. Virtanen, y J.M. Holm. Robust multipitch estimation for the analysis and manipularion of polyphonic musical signals. En *COST-G6 Conference on Digital Audio Effects, DAFx-00*, Verona, Italia, 2000.

[40] N. Kunieda, T. Shimamura, y J. Suzuki. Robust method of measurement of fundamental frequency by ACLOS– AutoCorrelation of LOg Spectrum. *IEEE Trans. on acoustic, Speech and Signal Processing*, 1996.

[41] M. Lahat, R. Niederjohn, y D. Krubsack. Spectral autocorrelation method for measurement of the fundamental frequency of noise-corrupted speech. *IEEE Trans. on acoustic, Speech and Signal Processing*, 35(6), Junio 1987.

[42] K.J. Lang, A.H. Waibel, y G.E. Hinton. A time-delay neural network architecture for isolated word recognition. En J.W. Shavlik y T.G. Dietterich, editores, *Readings in Machine Learning*, páginas 150–170. Kaufmann, San Mateo (CA), 1990.

[43] R.C. Maher. *An approach for the Separation of Voices in Composite Musical Signals*. Tesis doctoral, University of Illinois, 1989.

[44] R.C. Maher. Evaluation of a method for separating digitized duet signals. *J. Audio Eng. Soc.*, 38(12):956, Diciembre 1990.

[45] M. Marolt. Sonic: transcription of polyphonic piano music with neural networks. En *Proceedings of Workshop on Current Research Directions in Computer Music*, Barcelona, Noviembre 2001.

[46] K. Martin. Automatic transcription of simple polyphonic music: Robust front end processing. Informe técnico 385, MIT Media Laboratory Perceptual Computing Section Technical Report, 1996.

[47] K. Martin. A blackboard system for automatic transcription of simple polyphonic music. Informe técnico 385, MIT Media Lab, Julio 1996.

[48] K. Martin. A blackboard system for automatic transcription of simple polyphonic music. Informe técnico, MIT Media Laboratory Perceptual Computing Section Technical Report No. 399, 1996.

[49] D.K. Mellinger. *Event formation and separation in musical sounds.* Tesis doctoral, Stanford University. Dep. of Music Report STAN-M-77, 1991.

[50] N.J. Miller. Pitch detection by data reduction. *IWWW Trans. on Acoustics, Speech and Signal Processing*, páginas 72–79, Febrero 1975.

[51] D. Moelants y C. Rampazzo. A computer system for the automatic detection of perceptual onsets in a musical signal. En A. Camurri, editor, *KANSEI, The Technology of Emotion*, páginas 140–146, Génova, Italia, 1997.

[52] B.C.J. Moore, editor. *Hearing: Handbook of perception and cognition (2nd Edition).* Academic Press Inc., 1995.

[53] J.A. Moorer. *On the segmentation and Analysis of Continuous Musical Sound by Digital Computer.* Tesis doctoral, Stanford University: Department of Music; Report STAN-M-3, 1975.

[54] H.P. Nii. Blackboard systems: the blackboard model of problem solving and the evolution of blackboard architectures. *The AI Magazine*, páginas 38–53, Verano 1986.

[55] A. Noll. Cepstrum pitch detection. *J. Acoust. Soc. Am.*, 41:293–309, 1966.

[56] D. Nunn. Source separation and transcription of polyphonic music. En *International Colloquium on New Music Research*, Gent, Bélgica, 1994.

[57] J. Pearl. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29(3):241–288, 1986.

[58] L.R. Rabiner, M.J. Cheng, A.E. Rosenberg, y C.A. McGonegal. A comparative performance study of several pitch detection algorithms. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 24(5):369–377, Octubre 1976.

[59] D.E. Rumelhart, G.E. Hinton, y R.J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[60] M. Slaney y R.F. Lyon. On the importance of time - a temporal representation of sound. En M. Cooke, S. Beet, y M. Crawford, editores, *Visual Representation of Speech Signals*, páginas 95–116, New York, 1993. John Wiley & Sons.

[61] A.D. Sterian. *Model-based Segmentation of Time-Frequency Images for Musical Transcription*. Tesis doctoral, University of Michigan, 1999.

[62] B. Vercoe. *The CSound Reference Manual*. MIT Press, Cambridge, Massachusetts, 1991.