Tararira: Sistema de búsqueda de música por melodía cantada

Ernesto López , Martín Rocamora*

Instituto de Ingeniería Eléctrica – Facultad de Ingeniería de la Universidad de la República
Julio Herrera y Reissig 565 – (598) (2) 711 09 74, Montevideo, Uruguay

elopez@fing.edu.uy, rocamora@fing.edu.uy

Abstract. *The problem of music retrieval by sung query consists of building a machine capable of simulating the cognitive process of identifying a musical piece from a few sung notes of its melody. In this paper, the algorithms of pitch tracking, onset detection and melody matching used in the system Tararira are described. Much effort has been put on automatic transcription of singing voice as it is a key factor in the overall performance. A novel way of combining note by note matching with a recent approach based on pitch time series matching is introduced.*

Resumen. *El problema de búsqueda de música por tarareo consiste en construir un sistema capaz de simular el proceso cognitivo de identificar una pieza musical a partir de unas pocas notas cantadas de su melodía. En este artículo se describen los algoritmos de detección de altura, segmentación de audio en notas y comparación de melodías utilizados en el sistema Tararira. Se concentran esfuerzos en la transcripción automática de la voz cantada ya que es determinante en el desempeño del sistema. Para la comparación de melodías se propone una forma de combinar los enfoques basados en notas y series temporales, considerados antagónicos hasta el momento.*

1. Introducción

El desarrollo tecnológico de los sistemas de almacenamiento y reproducción de audio permiten disponer de grandes colecciones de música, incluso en dispositivos muy pequeños. En este escenario se plantean nuevas dificultades. Al aumentar la cantidad de material disponible, crece la dificultad de organizar y buscar esta información. Asimismo, el tamaño de algunos dispositivos (por ejemplo, los reproductores portátiles de audio comprimido) impone una interfaz con el usuario muy reducida que resulta limitada para acceder a la gran cantidad de datos que almacenan. Se hace necesario entonces, el desarrollo de nuevas formas de interacción hombre-máquina para el acceso práctico y eficiente a bases de datos de música. En los últimos años, la búsqueda y recuperación de música se ha convertido en un campo muy activo de investigación.

En la música occidental, una de las características mas representativas y recordables es la melodía. La mayoría de las personas son capaces de identificar con facilidad una pieza musical a partir de un fragmento de melodía. Surge así el interés por desarrollar sistemas de búsqueda de música por contenido que permitan el acceso por melodía usando la voz cantada para formular la consulta. Estos sistemas reciben el nombre de sistemas de búsqueda de música por tarareo (QBH, Query By Humming). Sin embargo, simular de forma automática la habilidad de los humanos de reconocer melodías (así como otros procesos cognitivos) es una tarea desafiante.

*Ambos financiados por la Comisión Sectorial de Investigación Científica (CSIC).

Los sistemas de búsqueda de música por tarareo buscan identificar la melodía cantada por el usuario en una base de datos de melodías. Desde el sistema propuesto en [Ghias et al., 1995], a lo largo de la última década se han considerado distintos enfoques para enfrentar este problema. En todas las propuestas, tanto por razones prácticas como técnicas, la base de datos se compone de música codificada en alguna notación simbólica, generalmente MIDI, en lugar de audio crudo (wav, aiff) o comprimido (ogg, mp3). La razón técnica más importante es que no existen formas automáticas de extraer la melodía de una grabación para compararla con la melodía cantada por el usuario. La principal razón práctica es la de reducir la información para un menor costo de procesamiento.

Los sistemas propuestos pueden dividirse respecto a la representación y técnica de búsqueda básicamente en dos enfoques. El enfoque tradicional es el basado en comparación de notas (Event Based Search) [Ghias et al., 1995] [McNab et al., 1996], mientras que un enfoque más reciente utiliza la comparación de series temporales de frecuencia fundamental (Frame Based Search) [Mazzoni y Dannenberg, 2001] [Dannenberg y Hu, 2002] [Shasha y Zhu, 2003]. El primer enfoque se fundamenta en que una forma natural de comparar melodías es a través de las notas que las componen y consiste en transcribir la señal de voz a una secuencia de notas y buscar las mejores ocurrencias de ese patrón en una base de datos de melodías. Debido a que los errores en la transcripción automática deterioran el desempeño del sistema, el otro enfoque busca evitarla comparando melodías a través de series temporales de frecuencia fundamental (F0). Desafortunadamente, al trabajar con secuencias largas (mucho más que las secuencias de notas) el tiempo de procesamiento requerido se torna intolerable. Adicionalmente, es necesario imponer que el usuario cante un fragmento de la melodía previamente definido [Dannenberg y Hu, 2002] [Shasha y Zhu, 2003]. En este trabajo se propone una forma novedosa de combinar ambos enfoques aprovechando las ventajas de cada uno.

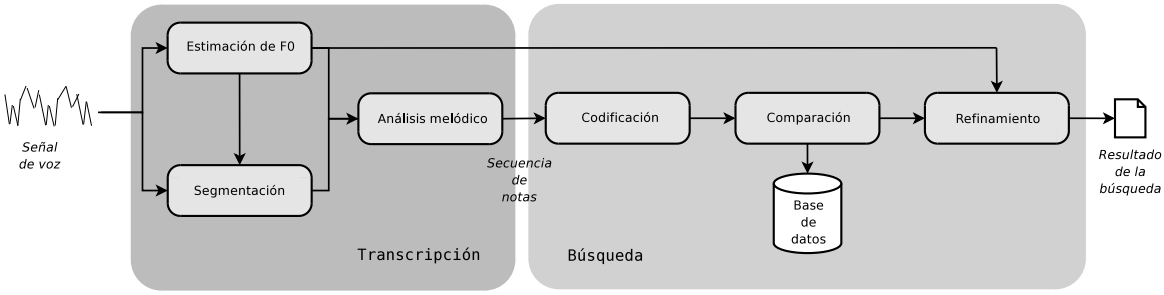

Figura 1: Diagrama de bloques del sistema.

El sistema selecciona inicialmente un grupo reducido de elementos de la base de datos a través de la comparación de secuencias de notas. Luego se refina la selección mediante la comparación de series temporales de F0. La arquitectura del sistema entonces consta básicamente de dos etapas, como se muestra en la figura 1. La primera consiste en la transcripción de la consulta a una secuencia de notas. En la siguiente etapa se compara esta secuencia con las melodías almacenadas en la base de datos y se devuelve una lista de piezas musicales ordenadas según su similitud con la consulta. La etapa de transcripción involucra las siguientes tareas:

- Estimar el contorno de F0 de la voz para determinar la altura de las notas.
- Segmentar la señal para establecer el tiempo de comienzo y fin de cada nota.
- Realizar un análisis melódico para ajustar la altura de las notas a la escala temperada.

Las tareas que constituyen la etapa de búsqueda son:

- Codificar la secuencia de notas para independizar la comparación de melodías del tempo y altura.
- Establecer criterios de similitud flexibles en la comparación para contemplar adornos y errores en la consulta, así como errores en la transcripción automática.
- Refinar la selección de candidatos comparando series temporales de frecuencia fundamental, de forma de eludir los errores en la transcripción automática.

2. Transcripción de voz cantada

Dada la forma de onda digitalizada de una señal de audio producida por la voz cantada, el objetivo de la transcripción automática es extraer la secuencia de notas que mejor representa la melodía cantada. Para ello, se identifican en la señal de audio los eventos con mayor probabilidad de corresponder a notas. Cada evento se caracteriza por tres valores: altura, tiempo de inicio y duración. Otro tipo de información, como características expresivas (intensidad, vibrato), no es de interés debido a que no forma parte de la notación simbólica tradicional.

En este trabajo se hace hincapié en la transcripción de la voz cantada ya que es un problema para el cual no existe aún una solución completamente satisfactoria. La voz cantada es uno de los instrumentos musicales más difíciles de tratar. Las grandes variaciones tímbricas, los recursos expresivos, la microentonación, la entonación inexacta y errores de altura y duración son algunas de las características que dificultan su análisis.

2.1. Detección de altura

La altura de un sonido es un concepto subjetivo determinado a partir de la percepción, si bien se relaciona en general con la frecuencia fundamental de la onda sonora. Esto tiene excepciones, como señales no periódicas que producen una sensación de altura (ruido de banda limitada, sonidos percutivos). Por esta razón, para establecer la altura de las notas cantadas se debe estimar la evolución de la frecuencia fundamental (F0) de la señal de voz, para lo cual existen técnicas bien conocidas. La frecuencia fundamental de la voz cantada usualmente varía entre 60 y 1000 Hz[1].

El algoritmo implementado utiliza la función diferencia normalizada [de Cheveignè y Kawahara, 2002], una variante de la función de autocorrelación que presenta algunas ventajas. La función diferencia consiste en la diferencia entre la señal y una versión retardada de la misma, cuya variable es el retardo,

$$d(\tau) = \sum_{j=1}^{W} (x_j - x_{j+\tau})^2 \tag{1}$$

donde W corresponde al largo del bloque de análisis. Si la señal es periódica, esta función se anula para valores del retardo múltiplos del período, mientras que para señales que no son perfectamente periódicas (como los tramos sonoros de la voz), no se anula pero presenta mínimos cercanos a cero. El valor del retardo correspondiente al primer mínimo coincide con el período de la señal. Por otro lado, los mínimos de la función diferencia de una señal no periódica (tramo de voz sordo), no son cercanos a cero. Es posible discriminar entre tramos sonoros y tramos sordos a partir del valor del mínimo de la función diferencia. Esto puede hacerse estableciendo un umbral de decisión, para lo cual es necesario normalizar la función diferencia. La normalización adoptada consiste en dividir los valores de la función entre la media acumulada,

$$d_{norm}(\tau) = \begin{cases} 1, & \text{si } \tau = 0 \\ \frac{d(\tau)}{\frac{1}{\tau}\sum_{j=1}^{\tau} d(j)}, & \text{en otro caso} \end{cases}. \tag{2}$$

[1] Sin embargo una cantante soprano puede cantar frecuencias fundamentales de hasta 1400 Hz.

La discriminación entre tramos sonoros y sordos colabora a la segmentación de la señal en notas, además de evitar la estimación de F0 para tramos de señal que no presentan periodicidad (en los tramos sordos y silencios, se establece F0 = 0).

El algoritmo calcula la función diferencia normalizada cada 10 ms para bloques solapados de señal y estima el valor de F0 detectando el primer mínimo. Debido a que se trabaja con señales muestreadas a 8 kHz, el error en la estimación es mayor al semitono para cierto rango de frecuencias. Para aumentar la resolución de la estimación se aproxima por una parábola cada valle de la función diferencia donde se presenta un mínimo. Los errores típicos de un algoritmo de estimación de F0 en el dominio del tiempo corresponden a la detección de un subarmónico de la frecuencia fundamental. Para evitar estos errores se suma una recta de pendiente positiva a la función diferencia, favoreciendo la elección del mínimo de menor orden. La aproximación por parábolas también colabora a la elección del mínimo correcto.

2.2. Segmentación en notas

Una vez obtenido el contorno de F0 de la señal de voz para establecer la altura de las notas, debe determinarse su tiempo de inicio y duración. Este problema se denomina segmentación automática de audio en notas y constituye la etapa más difícil de la transcripción automática. La voz cantada contiene un conjunto de rasgos que hacen que los límites entre notas sean a veces difusos y por lo tanto difíciles de resolver. Es importante entonces poder discriminar comienzos genuinos de cambios graduales y modulaciones que tienen lugar durante el transcurso de una nota. Es posible distinguir distintos tipos de comienzo de nota en una señal de voz. En el caso de notas cantadas con sílabas que comienzan con fonemas oclusivos (por ejemplo, /ta/), la repentina liberación de energía posterior a la restricción del pasaje de aire produce inicios marcados. El inicio de estas notas resulta sencillo de determinar dado el gran cambio de energía presente en la señal. Cuando la nota comienza con un aumento gradual de energía (por ejemplo, una consonante nasal), el inicio es más suave y por lo tanto más difícil de establecer. En señales de voz cantada existen inicios suaves e inicios marcados por lo que un algoritmo de segmentación automática debe manejar adecuadamente ambos casos. Desafortunadamente, no ha sido desarrollado hasta el momento un sistema capaz de detectar el amplio espectro de clases de inicio de nota existentes en las distintas formas de cantar. El peor escenario lo constituye la voz cantada con letra. En este caso el contorno de frecuencia fundamental y la forma de onda de la señal de voz presentan una infinidad de particularidades difíciles de caracterizar que entorpecen el análisis [Pollastri y Haus, 2001]. Esto es especialmente notorio en cantantes no experimentados.

El algoritmo de segmentación implementado, para lograr una detección robusta, busca indicios de eventos tanto en la envolvente de amplitud de la señal como en el contorno de F0. En una primera etapa se detectan eventos asociados a cambios de energía. Los eventos de mayor intensidad se asumen como comienzos de nota genuinos. En la segunda etapa se validan los eventos de menor intensidad si están acompañados de un cambio de altura. Finalmente se identifican los inicios de nota asociados a cambios evidentes de altura que no presentan un incremento de energía (por ejemplo, ligados).

Detección de eventos por cambios de energía El incremento de energía en la señal de audio cuando se produce un nuevo evento se manifiesta en un aumento de amplitud de la envolvente de la forma de onda. Tomando la derivada de la envolvente es posible construir una función de detección que presente picos donde hay cambios bruscos de amplitud [Dixon, 2001] [Schloss, 1985]. Esto funciona adecuadamente solo para casos particulares, como sonidos percutivos. Para segmentar señales más complejas, se propone trabajar con envolventes de amplitud de distintas bandas de frecuencia [Klapuri, 1999].

En [Scheirer, 1998] se señala que es posible mantener la información rítmica de una señal de audio solo a través de envolventes de amplitud de distintas bandas de frecuencia. Este mismo enfoque puede aplicarse a la segmentación de una señal de audio, dividiéndola en bandas de frecuencia y detectando eventos en las envolventes de amplitud de cada banda. En general, los eventos aparecen más claramente en alguna de las bandas que en la envolvente de la señal (ver figura 2, izquierda). Por ejemplo, cuando un nuevo evento está asociado a cambios de altura o cambios tímbricos, puede producir la aparición repentina de información en alguna banda de frecuencia.

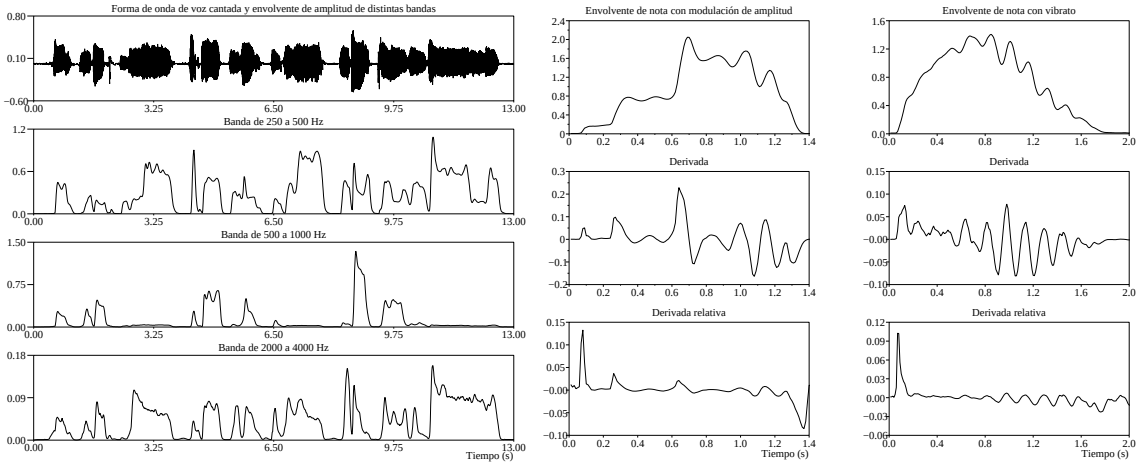

Figura 2: Envolventes de amplitud de distintas bandas de frecuencia (izquierda). Los eventos son más evidentes en alguna banda que en la señal original. Envolvente de amplitud, derivada y derivada relativa (derecha). El máximo de la derivada relativa coincide con el inicio físico de las notas.

En el algoritmo implementado la señal de audio se divide en bandas de frecuencia, se extrae la envolvente de cada banda y se obtiene su derivada y derivada relativa. Estas señales son utilizadas para determinar candidatos de comienzo de nota, a los cuales se les asigna un valor de intensidad. Finalmente se combina la información de las distintas bandas y se seleccionan los candidatos más prominentes. A continuación se describen cada una de las etapas.

La señal de audio es normalizada en amplitud y filtrada en 6 bandas de octava por un banco de filtros de frecuencias de corte 125, 250, 500, 1000 y 2000 Hz. Las señales de cada banda se rectifican onda completa y se deciman a una frecuencia de muestreo de 100 Hz. Las envolventes se obtienen convolucionando cada señal con una ventana mitad Hanning (coseno elevado) de 100 ms, que produce una integración de potencia que mantiene cambios repentinos pero enmascara modulaciones rápidas.

Tradicionalmente los algoritmos que trabajan con envolventes de amplitud calculan su derivada de primer orden y asocian los puntos de máxima pendiente al comienzo de notas. Sin embargo, la derivada de primer orden es apropiada para reflejar la sonoridad de las notas, pero sus máximos no indican adecuadamente el instante de su inicio. Esto se debe principalmente a que la envolvente correspondiente al inicio de un sonido no es monótonamente creciente en la mayoría de los casos, dando lugar a varios máximos locales en la derivada de primer orden. Asimismo, muchos sonidos (en particular los graves) tardan cierto tiempo en alcanzar el punto donde su amplitud crece con pendiente máxima, y éste es posterior al inicio físico del sonido. Por esta razón se propone utilizar la derivada relativa $D_r(t) = \frac{\frac{d}{dt}A(t)}{A(t)}$, donde $A(t)$ es la envolvente de amplitud [Klapuri, 1999]. De esta forma, las oscilaciones de amplitud que ocurran durante el inicio de una nota serán atenuadas. Por otra parte el máximo de la derivada relativa se ubica antes que el máximo de la derivada, coincidiendo con el inicio físico de la nota (ver figura 2, derecha). El uso de

la derivada relativa tiene sentido desde el punto de vista psicoacústico, ya que el incremento de intensidad percibido por el sistema auditivo está relacionado con la amplitud del sonido, siendo un mismo incremento de amplitud más relevante para sonidos de menor amplitud[2]. En el algoritmo implementado la derivada relativa de la envolvente se calcula como la derivada de la envolvente comprimida según la ley μ, con un valor de $\mu = 100$.

La selección de candidatos de comienzo de nota se realiza detectando los picos de la derivada relativa de cada banda que superan cierto umbral. El valor de intensidad se establece como el primer máximo de la derivada de amplitud a partir del instante del pico de la derivada relativa. Para evitar la detección de inicios duplicados debido a variaciones de la pendiente durante el ataque se eliminan candidatos de baja intensidad cercanos a otro más prominente. De esta forma se toma en cuenta por ejemplo, que no es posible un cambio instantáneo de fortissimo a pianissimo.

La información de las distintas bandas se combina asumiendo que candidatos ubicados a menos de 50 ms corresponden al mismo evento. La elección definitiva de los candidatos se realiza usando un umbral fijo y un umbral dinámico similar al anterior. Debido a que los candidatos de menor intensidad podrían corresponder a modulaciones de amplitud provenientes de recursos expresivos o respiración, el algoritmo clasifica los candidatos según su intensidad y devuelve un grupo de eventos fuertes y un grupo de eventos débiles. Los primeros se consideran inicios de nota genuinos, mientras que los más débiles se validan en una etapa posterior.

Validación de eventos débiles por cambios de altura La mayoría de los eventos débiles provienen de modulaciones de amplitud que tienen lugar durante el transcurso de una nota, mientras que otros se deben a transiciones suaves entre notas. Un evento débil se valida como inicio de nota si va acompañado de un cambio de altura significativo (ver figura 3, izquierda).

Para validar el evento débil se obtiene la mediana del contorno de frecuencia durante cierto intervalo de tiempo antes y después del mismo. Luego se comparan estos valores determinando si su diferencia supera un umbral establecido a partir del intervalo de semitono (6 %). El umbral utilizado es menor al semitono ya que el contorno de F0 presenta transiciones suaves entre notas sucesivas.

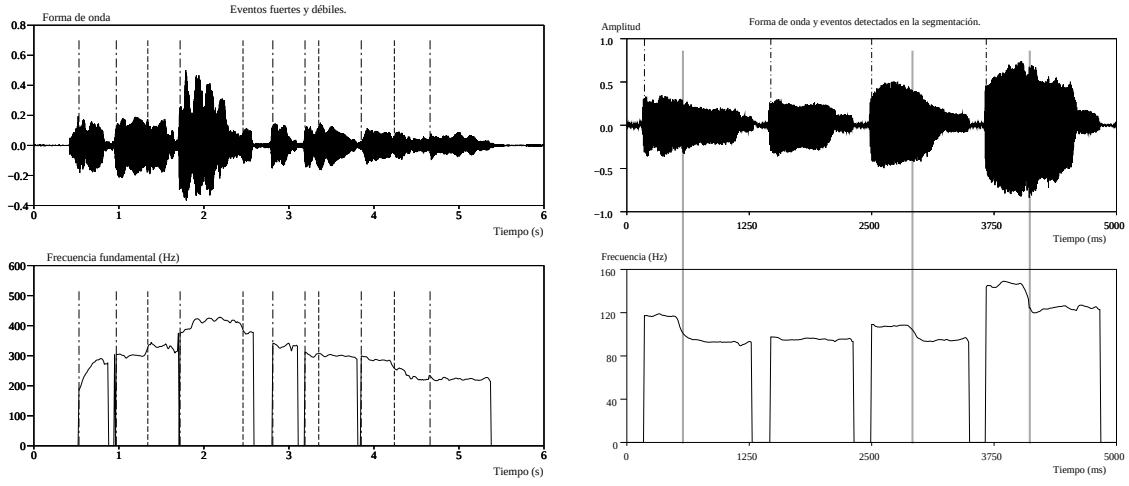

Figura 3: Se indican los eventos fuertes (raya y punto) y los eventos débiles (línea punteada) de la segmentación (izquierda). Los eventos débiles asociados a un cambio de altura se confirman. Las notas ligadas no se pueden deducir a partir de la forma de onda (derecha).

[2]Si ΔI es el incremento de intensidad mí nimo perceptible, la relacón $\Delta I/I$ (fracción de Weber) es constante.

Detección de eventos por cambios de altura La segmentación basada en cambios de energía no es capaz de detectar el inicio de algunas notas. Un ejemplo son las notas ligadas, en donde existe un cambio de altura que no va acompañado de un aumento de amplitud (ver figura 3, derecha). Otro caso son las notas de ataque muy suave, por ejemplo con fonemas sonoros nasales.

Las notas omitidas, en muchos casos, pueden ser agregadas observando los cambios de altura en el contorno de F0. Esto sin embargo no es una tarea sencilla debido a que la expresividad en la interpretación y la falta de entrenamiento en cantantes inexpertos introducen un conjunto de rasgos en el contorno de F0 [Pollastri, 2003] que pueden ser considerados por error como notas adicionales (transiciones suaves, picos, inestabilidades, vibrato).

A partir del contorno de F0 y de los eventos detectados hasta el momento se construye un conjunto de alturas y tiempos de inicio y fin de nota. El tiempo de fin está dado por el inicio de una nueva nota o cuando se anula el contorno de F0. La altura se calcula como la mediana del contorno de F0 dentro del intervalo. Se procesa cada intervalo identificando tramos del contorno que se desvían más de un semitono del valor de altura asignado. Si el tramo cumple ciertas condiciones de estabilidad y duración mínima es considerado una nota.

2.3. Ajuste a la escala temperada

Para asignar una altura a cada nota es necesario en primer lugar aproximar el contorno de F0 de la nota a un único valor de frecuencia y luego asociar este valor a una altura de un sistema de afinación (por ejemplo, escala temperada con A_4 = 440 Hz).

Como ya se mencionó, el contorno de F0 de una nota puede ser muy variable y resulta difícil establecer un criterio para asignarle un único valor de frecuencia. El valor de frecuencia que mejor representa la altura de la nota es aquel que toma el contorno cuando alcanza la estabilidad, lo que implica ignorar las transiciones suaves, inestabilidades, picos espurios. En este trabajo se adopta un criterio sencillo y efectivo: el valor de frecuencia de la nota es la mediana del contorno. Típicamente la región de transición y los picos espurios son de corta duración respecto a la zona de estabilidad, por lo que la mediana generalmente se aproxima al valor de estabilidad.

La música de diversas culturas utiliza distintos sistemas de afinación. Una hipótesis necesaria para la transcripción de una melodía es el sistema de afinación a usar. Este trabajo se concentra en música occidental, por lo que la transcripción se hará a la escala temperada. Sin embargo, las personas al cantar no tienen la capacidad de ajustarse a un sistema de afinación sin escuchar una referencia, salvo raras excepciones (oído absoluto). La interpretación entonces no respeta exactamente los intervalos ni la referencia de la escala temperada. Se hace necesario corregir el desajuste natural entre la melodía cantada y el sistema de afinación. Una alternativa es asignar la nota mas cercana de la escala temperada, pero de esta forma las notas pueden ajustarse en diferentes sentidos distorsionando los intervalos cantados.

Se han propuesto métodos mas apropiados de ajuste a la escala temperada [McNab et al., 1996] [Pollastri y Haus, 2001] [Viitaniemi et al., 2003]. Los resultados de evaluar las distintas técnicas indican que el método más apropiado es el propuesto por Pollastri. Este se basa en la hipótesis de que al cantar se mantiene un tono de referencia en mente y las notas cantadas pertenecen a una escala temperada referida a este tono. El método consiste en determinar la desviación más frecuente para estimar el tono de referencia y así ajustar las notas cantadas a la escala temperada absoluta. Para ello se divide el semitono en 10 intervalos iguales de 0.2 semitonos solapados 0.1 semitono, se calcula

la desviación de cada nota como la parte fraccionaria de la nota MIDI equivalente[3] y se asocia al intervalo correspondiente. La desviación más frecuente corresponde a la media de las desviaciones del intervalo con más notas. A cada nota se le resta esta desviación y se redondea a la nota MIDI más cercana. Al estimar el tono de referencia a partir de la desviación mas frecuente se considera que además de la desviación respecto a la escala absoluta, en algunos intervalos puede existir una desviación adicional pequeña que se relaciona con la dificultad de cantarlos.

3. Comparación de melodí as

Los principales desafíos que presenta la etapa de búsqueda son la codificación de la información y los criterios de similitud. La melodía es información de más alto nivel que la secuencia de notas específica que la compone. Una melodía puede identificarse a pesar de que se interprete en diferentes alturas, a distinto tempo (dentro de límites razonables) y con adornos o rasgos expresivos. Estas consideraciones son esenciales en el diseño de la etapa de búsqueda. La independencia respecto a la altura y al tempo puede lograrse en la codificación de las notas (codificación invariante a la transposición y al tempo). Mediante criterios de similitud flexibles durante la búsqueda es posible establecer tolerancia a las alteraciones provenientes de adornos, así como a errores en la interpretación y en la transcripción automática.

Como ya se mencionó, existen básicamente dos enfoques para la comparación de melodías: la comparación de notas y la comparación de series temporales de frecuencia. Ambos enfoques presentan sus desventajas. El primero requiere la transcripción automática de la consulta, lo que inevitablemente introduce errores que deterioran el desempeño del sistema. El segundo, si bien evita la transcripción, involucra un costo computacional alto y permite buscar solo fragmentos de melodía definidos previamente.

Usualmente los sistemas de comparación de melodías retornan una larga lista como resultado de la búsqueda. En el sistema desarrollado se busca retornar únicamente la pieza musical buscada. Por esta razón, se aumenta la eficacia del sistema agregando a la búsqueda por comparación de notas una etapa de refinamiento basada en comparación de series temporales de F0.

3.1. Comparación basada en secuencias de notas

Al trabajar con secuencias de notas, la búsqueda de melodías es básicamente un problema de búsqueda aproximada de cadena de caracteres (Approximate String Matching). Es necesario derivar de la secuencia de notas una codificación que sea invariante tanto a la transposición de altura como al tempo. La cuantización de los intervalos de altura y duración permite ajustar la tolerancia a los errores en la consulta. Asimismo, cuantizar los intervalos a un alfabeto discreto es necesario para utilizar técnicas de búsqueda de cadena de caracteres.

Codificación La secuencia de notas que devuelve la transcripción automática se representa por una secuencia de alturas y una secuencia de tiempos de inicio y duración. La invarianza a la transposición de altura se obtiene codificando la secuencia de alturas $A = (a_1, a_2, \ldots, a_n)$ en la secuencia de intervalos $\overline{A} = (a_2 - a_1, a_3 - a_2, \ldots, a_n - a_{n-1})$. Resulta evidente que una secuencia A' transposición de A, es decir $a'_i = a_i + c$, tiene la misma representación en intervalos. Una forma de lograr invarianza al tempo es normalizar las duraciones respecto a una duración de referencia d_{ref} invariante al tempo.

[3] La nota MIDI equivalente se obtiene a partir de la frecuencia como $n_{MIDI} = 69 + 12\frac{log(\frac{f}{440})}{log(2)}$.

Un valor apropiado es el período de pulso, como en la notación musical. Lamentablemente, en el caso de una melodía cantada, no siempre es posible estimar el pulso, por lo que este criterio no es aplicable. Sin conocer el tempo, no existe una normalización de las duraciones completamente aceptable. Una alternativa sencilla utilizada frecuentemente es considerar d_{ref} como la duración de la nota previa [Pardo y Birmingham, 2002]. Dada la secuencia de duraciones $D = (d_1, d_2, \ldots, d_n)$, la representación invariante al tempo utilizada es la secuencia de duraciones relativas $\overline{D} = (\frac{d_2}{d_1}, \frac{d_3}{d_2}, \ldots, \frac{d_n}{d_{n-1}})$. Esta secuencia se cuantiza a un alfabeto discreto de enteros por medio de la función logarítmica $r(i) = \text{round}\left(10 \log_{10}\left(\frac{d_i}{d_{i-1}}\right)\right)$ [Pollastri, 2003]. El logaritmo suaviza la relación de duraciones de forma de atenuar las grandes aproximaciones de duración que se cometen al cantar despreocupadamente. Un error común de los cantantes inexpertos es finalizar prematuramente las notas, por lo que se considera el intervalo entre inicios de nota sucesivos (IOI, Inter Onset Interval) como una representación más consistente de las duraciones. La secuencia de intervalos de altura no se cuantiza (codificación en intervalos exactos), pero en la etapa de comparación se utilizan criterios flexibles.

Comparación La etapa de comparación consiste en encontrar buenas ocurrencias de la secuencia de notas codificada en la base de datos. Para ello es necesario definir una medida de distancia. Se ha demostrado que la Distancia de Edición (Edit Distance) es la mejor medida de similitud para la comparación de melodías [Uitdenborgerd y Zobel, 1999]. La distancia de edición consiste en el número mínimo de alteraciones necesarias para transformar una cadena de caracteres en otra y se determina a través del algoritmo conocido como Programación Dinámica (DP, Dynamic Programming) [Lemström, 2000].

El algoritmo de comparación calcula la distancia de edición combinando la información de duración y altura. Al realizar la combinación se prioriza la información de altura ya que permite mayor discriminación que la duración. Adicionalmente la información de duración es menos confiable debido a las grandes aproximaciones que se cometen al cantar y a que la secuencia de duraciones es más sensible a los errores de la transcripción automática[4].

Sean a_i^a y b_j^a los intervalos exactos de altura y a_i^d y b_j^d los intervalos de duración codificados de las secuencias a comparar. La distancia de edición se calcula llenando recursivamente una matriz D_{DP} en la que cada elemento d_{ij} se obtiene como,

$$d_{ij} = \min \begin{cases} d_{i-1,j} + i \\ d_{i,j-1} + o \\ d_{i-1,j-1} + s \\ d_{i-1,j-1} + c & |a_i^a - b_j^a| < 2 \text{ y} |a_i^d - b_j^d| < 2 \\ d_{i-1,j-1} + s_d & |a_i^a - b_j^a| < 2 \end{cases}$$

Costo

i	o	s	c	s_d
1	1	1	-1	0

donde i es el costo de una inserción, o el de una omisión, s el de una sustitución, c el de una coincidencia de duración y altura y s_d el de una sustitución de duración. Cabe señalar que a pesar de usar intervalos exactos de altura, dos notas que difieren en un semitono se consideran iguales en el cálculo de la distancia de edición. Se introduce también tolerancia en los intervalos de duración.

A partir de la distancia de edición se define una medida de similitud S que toma valores entre 0 y 100 como $S = 100\ \frac{(m-1)-E}{2(m-1)}$, con E la distancia de edición y m el número de notas de la consulta. Se comparan todos los elementos de la base de datos con la consulta codificada y se selecciona un conjunto de las mejores ocurrencias en función del valor de similitud.

[4]Por ejemplo, si no se segmentan dos notas consecutivas de la misma altura, se afecta solo un intervalo de altura y tres intervalos de duración.

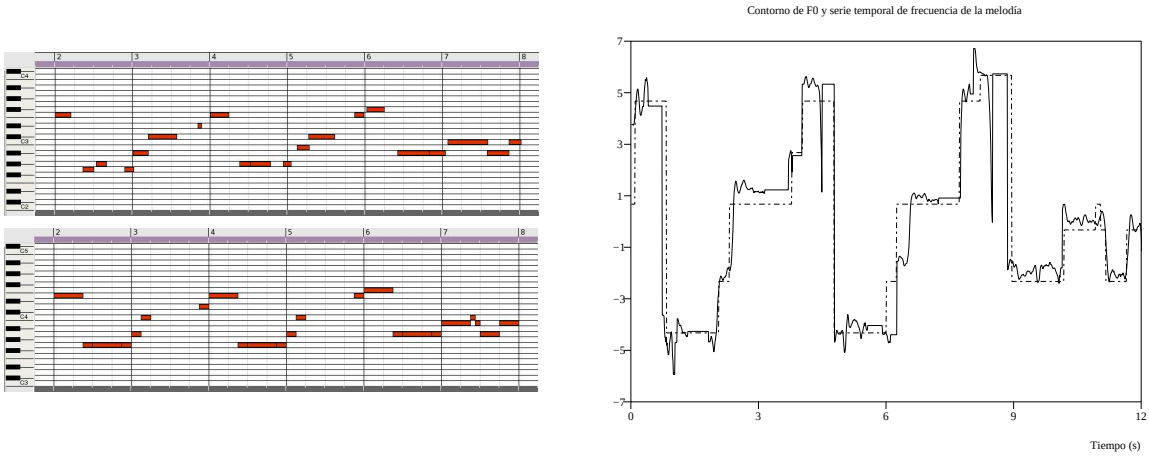

Figura 4: Transcripción de la consulta y ocurrencia en la base de datos (izquierda) y las correspondientes series temporales normalizadas y alineadas por el sistema (derecha).

3.2. Comparación basada en series temporales

Como alternativa al enfoque basado en comparación de notas, recientemente se estudia la posibilidad de identificar melodías directamente a partir de características derivadas de la señal de voz. Este enfoque consiste en comparar el contorno de F0 de la consulta con melodías codificadas como series temporales de frecuencia. Intuitivamente la forma de comparar dos series temporales de distinto largo es, en primer lugar, ajustarlas al mismo largo y luego compararlas punto a punto permitiendo cierta deformación temporal. La técnica de Deformación Temporal Dinámica Local (LDTW, Local Dynamic Time Warping) permite comparar series temporales de esta forma.

Además del alto costo computacional, una restricción del uso de LDTW es que algún elemento de la base de datos debe corresponder exactamente a la consulta, ya que no se puede buscar subsecuencias dentro de secuencias manteniendo invarianza a la trasposición de altura y al tempo [Dannenberg y Hu, 2002] [Shasha y Zhu, 2003]. Por esta razón el proceso de construcción de la base de datos es complejo ya que es necesario identificar dentro de la melodía original los fragmentos más probables de ser cantados.

En el sistema implementado se seleccionan las mejores ocurrencias del patrón buscado a través de la comparación de notas. Durante este proceso se identifican los tramos que se asemejan a la consulta dentro de las melodías seleccionadas. Luego se construyen series temporales de frecuencia de esos tramos que se comparan con el contorno de F0 de la consulta. De esta forma se aplica LDTW sobre un conjunto reducido de candidatos sin imponer restricciones sobre la consulta (ver figura 4).

El primer paso para aplicar LDTW es ubicar el comienzo y fin de las mejores ocurrencias del patrón en las melodías de la base de datos. Esto corresponde a encontrar el alineamiento entre las secuencias, lo que puede hacerse a través del camino mínimo[5] de la matriz D_{DP}. Luego se normalizan las duraciones de las ocurrencias para igualarlas a la de la consulta. Esto es necesario debido a que para aplicar LDTW se requiere que las series a comparar sean del mismo largo. En esta etapa se descartan candidatos de tempo excesivamente distinto al de la consulta. A partir de la secuencia de notas de la ocurrencia normalizada en el tiempo se construye una serie temporal de valores de altura. El siguiente paso es llevar las series a forma normal para poder compararlas, es decir transformarlas en series de varianza uno y media nula. Finalmente se calcula la distancia LDTW para cada una de las ocurrencias con un factor de deformación máximo de un segundo.

[5]El camino mí nimo es el camino de menor costo.

El sistema devuelve un único elemento de la base de datos como resultado de la consulta si es capaz de diferenciarlo del conjunto de los candidatos en función de los valores de similitud S y distancia LDTW. En caso contrario se retorna la lista de candidatos ordenada según la distancia LDTW.

4. Evaluación

El sistema desarrollado denominado Tararira fue implementado en C++[6] y se construyó una base de datos de melodías MIDI monofónicas con la colección completa de The Beatles (208 temas). Se condujo una evaluación en la que participaron más de 30 personas sin entrenamiento musical. En la tabla de la figura 5 se presentan los resultados. Si bien el tamaño de la base de datos es acotado, el desempeño obtenido es alentador.

	Voz cantada	
	Cortas	Largas
Consultas	68	242
Notas promedio	11.15	25.81
Primero (%)	73.53	76.86
Primeros 10 (%)	80.88	83.06
	Tarareo	
	Cortas	Largas
Consultas	85	32
Notas promedio	12.03	22.40
Primero (%)	75.00	83.52
Primeros 10 (%)	90.62	89.41

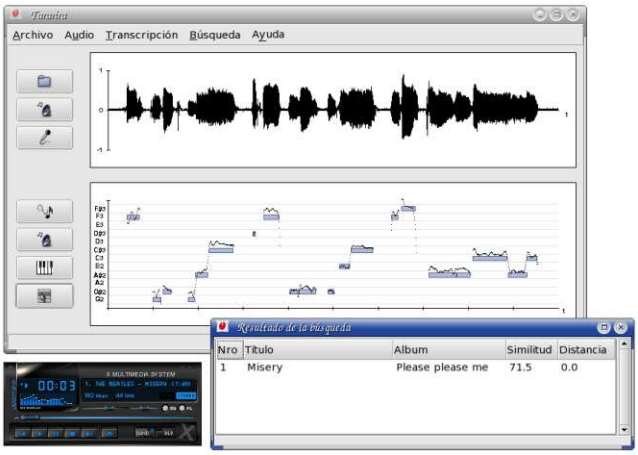

Figura 5: Resultados de la evaluación y aspecto de la aplicación.

La evaluación confirma algunas hipótesis sobre el problema. Las características de la voz cantada con letra dificultan su transcripción lo que se refleja en los resultados. Se confirma que al aumentar el largo de la consulta mejora el desempeño, ya que la melodía se torna más identificable.

5. Conclusiones

Un sistema ideal de búsqueda de música por melodía además de ser rápido y eficaz, debe tolerar errores en la consulta, no restringir al usuario en la forma de cantar y retornar únicamente la pieza buscada. Otra característica importante es que el agregado de piezas musicales a la base de datos sea sencillo. Si bien aún no existe un sistema que cumpla con estos requerimientos, Tararira fue diseñado tomándolos en cuenta.

A los efectos de contemplar las distintas formas de cantar se desarrolló un sistema de transcripción robusto, integrando y adaptando técnicas que representan el estado del arte. Se buscó perfeccionar la segmentación integrando de manera efectiva la información de energía y altura de la señal de voz.

Los enfoques de búsqueda de música basados en comparación de notas y comparación de series temporales son considerados antagónicos y no se conocen antecedentes de usarlos en forma conjunta. Con el objetivo de aumentar la capacidad de discriminación del sistema de búsqueda se combinó de forma novedosa ambos enfoques aprovechando las

[6]El programa compilado para Linux puede obtenerse de la página web `http://iie.fing.edu.uy/investigacion/grupos/gmm/proyectos/tararira/`

ventajas de cada uno. De esta forma, la construcción y extensión de la base de datos del sistema es relativamente sencilla ya que no es necesario extraer los fragmentos más representativos de la melodía, como en los sistemas de comparación de series temporales.

Se desarrolló un sistema de búsqueda completo que funciona correctamente. En trabajos futuros se abordará la validación exhaustiva del sistema y su extensión a aplicaciones de mayor escala.

Referencias

Dannenberg, R. B. y Hu, N. (2002). A comparison of melodic database retrieval techniques using sung queries. *JCDL*, pages 301–307.

de Cheveignè, A. y Kawahara, H. (2002). Yin, a fundamental frequency estimator for speech and music. *JASA*, 111:1917–1930.

Dixon, S. (2001). Learning to detect onsets of acoustic piano tones. *Proc. of the Workshop on Current Directions in Computer Music Research*.

Ghias, A., Logan, J., Chamberlin, D., y Smith, B. C. (1995). Query by humming: Musical information retrieval in an audio database. *Proc. ACM Multimedia*, pages 231–236.

Klapuri, A. P. (1999). Sound onset detection by applying psichoacoustic knowldege. *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*.

Lemström, K. (2000). *String Matching Techinques for Music Retrieval*. PhD thesis, Department of Computer Science, University of Helsinki.

Mazzoni, D. y Dannenberg, R. B. (2001). Melody matching directly from audio. *Proc. of ISMIR*.

McNab, R. J., Smith, L. A., Witten, I. H., Henderson, C. L., y Cunningham, S. J. (1996). Towards the digital music library: Tune retrieval from acoustic input. *Proc. of the ACM Digital Libraries*, pages 11–18.

Pardo, B. y Birmingham, W. (2002). Encoding timing information for musical query matching. *ISMIR*, pages 267–268.

Pollastri, E. (2003). *Processing Singing Voice for Music Retrieval*. PhD thesis, Università Degli Studi Di Milano.

Pollastri, E. y Haus, G. (2001). An audio front end for query-by-humming systems. *Proc. of ISMIR*.

Scheirer, E. D. (1998). Tempo and beat analysis of acoustic signals. *JASA*, pages 588–601.

Schloss, W. A. (1985). *On the Automatic Transcription of Percusive Music: From Acoustic Signal to High Level Analysis*. PhD thesis, Stanford University, CCRMA.

Shasha, D. y Zhu, Y. (2003). Warping indexes with envelope transforms for query by humming. *Proc. of the 2003 ACM SIGMOD Conference on Management of Data*, pages 181–192.

Uitdenborgerd, A. y Zobel, J. (1999). Melodic matching techniques for large music databases. *Proc. of the ACM Multimedia*, pages 57–66.

Viitaniemi, T., Klapuri, A., y Eronen, A. (2003). A probabilistic model for the estimation of single-voice melodies. *Finnish Signal Processing Symposium, Tampere University of Technology*.