

Trabajo de Fin de Grado
Grado en Ingeniería de las Tecnologías de
Telecomunicación

Un Algoritmo para la Detección Automática de
Falsetas de Guitarra Flamenca.

Autor: Javier Ivar Advani Aguilar

Tutor: José Miguel Díaz Báñez

Dep. Matemática Aplicada II
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2016



Trabajo de Fin de Grado
Grado en Ingeniería de las Tecnologías de Telecomunicación

Un Algoritmo para la Detección Automática de Falsetas de Guitarra Flamenca.

Autor:

Javier Ivar Advani Aguilar

Tutor:

José Miguel Díaz Bañez

Profesor titular

Dpto. de Matemática Aplicada II
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2016

Trabajo de Fin de Grado: Un Algoritmo para la Detección Automática de Falsetas de Guitarra Flamenca.

Autor: Javier Ivar Advani Aguilar

Tutor: José Miguel Díaz Báñez

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2016

El Secretario del Tribunal

A mi padre

Agradecimientos

Infinitamente agradecido a María José Aguilar y a María del Carmen Gutiérrez, por haber creído en mí y en mi futuro desde el primer momento. A Jesule del Puerto, por instruirme y enseñarme el valor del flamenco, más allá de los compases. A José Miguel Díaz, Nadine Kroher e Inmaculada Morales Peinado por la amabilidad que han tenido al atender al impaciente repleto siempre de preguntas, esperando tras la puerta del despacho.

Javier Ivar Advani Aguilar

Sevilla, 2016

Resumen

La terminología acuñada en el ámbito de la guitarra flamenca es pobre y en ocasiones algo ambigua, lo que genera gran controversia al estudiarla o interactuar con otros músicos. Así, términos como *falseta*, *melodía* o *variación melódica*, se aplican de forma indiscriminada. Con el fin de arrojar información adicional y facilitar un estudio más explícito y objetivo, se plantea un algoritmo de extracción automática de aquellas secciones de audio en que sólo intervenga la guitarra ejecutando una falseta.

Así, este proyecto tiene como objetivo fundamental la identificación automática de falsetas flamencas, entendiendo éstas como determinados fragmentos musicales interpretados por la guitarra flamenca que poseen entidad musical propia y se ejecuta entre dos interpretaciones vocales. La investigación ha sido realizada a partir de la colección de una serie de grabaciones en las que intervienen la guitarra, la voz del cantaor e incluso, en algunos casos, la percusión.

Abstract

The terminology associated to the field of the flamenco guitar is occasionally poor and ambiguous, so it generates controversy by studying it or by playing and interacting with other musicians. Thereby, concepts like *falseta*, *melody* or *melodic variation*, are applied with no discrimination. With the final purpose of giving additional information and making easier a more objective and explicit study, it is proposed an algorithm for an automatic extraction of all the audio sections where exclusively the guitar takes part.

This Project has as main objective the automatic identification of flamenco *falsetas*, understanding them as the musical fragments played by the flamenco guitar, with their own musical entities, and executed between two vocal interpretations. The research has been done starting from a collection of audios containing the guitar, the voice and, occasionally, some percussion.

Índice

Agradecimientos	7
Resumen	9
Abstract	11
Índice	13
1 Introducción	17
1.1. <i>Breve Historia del Flamenco. El papel de la guitarra en la Música Flamenca</i>	
1.2. <i>El Flamenco como proceso comunicativo: Falsetas</i>	
1.3. <i>MIR y Etnomusicología Computacional.</i>	
1.4. <i>Rasgos y aportaciones. Proyecto CoFla.</i>	
2 Entorno de Desarrollo del Algoritmo	19
2.1. <i>Herramientas utilizadas para la extracción automática</i>	
3 Algoritmo de Extracción Automática.	25
3.1. <i>Algoritmo de Extracción Automática de Falsetas</i>	
3.1.1. <i>Separación de canales</i>	
3.1.2. <i>Definición de las Bandas de Bark.</i>	
3.1.3. <i>Filtrado Gaussiano de muestras. Obtención de frames vocales.</i>	
3.1.4. <i>Frames de guitarra. Definición de falsetas.</i>	
3.2. <i>Implementación del Algoritmo.</i>	
4 Obtención de Resultados. Conclusiones	37
4.1. <i>Histogramas.</i>	
4.2. <i>Precisión y Exhaustividad.</i>	
4.3. <i>Resultados. Optimización futura del algoritmo</i>	
4.4. <i>Conclusiones</i>	
Anexo I: Código de Matlab	41
Anexo II: Corpus Utilizado	45
Glosario	49
Referencias	53

1 INTRODUCCIÓN

La música empieza donde acaba el lenguaje.

- Hoffmann -

El flamenco es un género musical que integra fundamentalmente el cante, el baile y el toque de guitarra [33]. Es una expresión artística resultado del paso y la convivencia de diversas culturas por nuestro país a lo largo de la Historia. Sin duda, en la actualidad se ha convertido en una de las manifestaciones socioculturales más representativas no sólo de Andalucía sino también de España. Debido a que desde sus primitivos orígenes ha logrado expandirse y popularizarse principalmente mediante la transmisión oral, tratar de formalizar su análisis encierra una gran complejidad, pues nos enfrentamos junto a la escasez de documentación disponible, con la disparidad y controversia en los métodos didácticos existentes para la conservación y transmisión de su conocimiento.

Esta manifestación sociocultural, conocida por su gran intensidad emocional, ha logrado extenderse universalmente siendo, el Arte Flamenco, declarado Patrimonio Cultural Inmaterial de la Humanidad en el año 2010.

1.1. Breve Historia del Flamenco. El papel de la guitarra en la música flamenca.

Casi todas las opiniones están de acuerdo en señalar que el flamenco es la consecuencia final de la unión de dos elementos fundamentales: lo gitano y lo andaluz [34], dos culturas resultantes de la fusión que a lo largo de los siglos experimentaron estos pueblos en convivencia con el paso de diferentes civilizaciones.

Podríamos decir que los orígenes del flamenco comenzaron a fraguarse en el s. XV con la llegada de los gitanos a Andalucía [35], donde encontraron un clima que facilitó su asentamiento definitivo en estas tierras entre los sectores más desarraigados y marginados del pueblo andaluz [36] [37] (moriscos, judíos conversos, campesinos sin tierras, gentes huidas por la amenaza de la Inquisición o el destierro,...) hallando entre ellos la hospitalidad que les había faltado en tiempos pasados, en que fueron víctimas de persecuciones y atropellos por las drásticas leyes dictadas contra los gitanos, viéndose obligados a llevar una vida errante, azarosa y clandestina, desperdigados por toda Europa.

En esta convivencia, amasada en el dolor y en la miseria de todos los días, fundirá el gitano las riquísimas y milenarias formas folclóricas andaluzas, surgiendo de esta genial unión el flamenco como respuesta capaz de expresar una similar concepción del hombre y de la vida.

Tras una primera y oscura etapa inicial de anonimato doméstico, el pueblo gitano comienza a divulgar abiertamente el flamenco durante la segunda mitad del siglo XVIII, gracias a una ley promulgada por Carlos III [38] en 1763, que permite a estos asentarse libremente en las distintas poblaciones de la geografía andaluza, poniendo así fin a las grandes persecuciones a las que habían sido sometidos.

El flamenco posee una clara similitud con las modulaciones y melismas [39] provenientes de los cantos monocordes (monofónicos) islámicos e influencias de las melodías salmodiales de las sinagogas y el sistema musical judío, junto con los modos jónico y frigio del canto bizantino, los antiguos sistemas musicales hindúes, y canciones populares mozárabes, de donde procedieron posteriormente las *jarchas* y *zambras*¹.

Es importante destacar que esta música, durante su desarrollo, fue disgregándose en distintas variedades tradicionales, denominadas *palos* que, según las situaciones geográficas y sociales, desembocaron y dieron lugar a diferentes subgéneros. En líneas generales, estos palos se fraguaron también para distintas situaciones o emociones. Así, el cante *jondo* [40], o cante grande del flamenco, expresa sentimientos profundos y normalmente dolorosos. Los cantes de fiesta o festeros, expresan alegría, sensualidad y pasión, no exentos de influencias caribeñas gracias al monopolio de las rutas comerciales de nuestro país con América desde los puertos de Sevilla y más tarde Cádiz.

La música flamenca, en sus representaciones más primitivas, se trataba de un canal expresivo limitado a la interpretación vocal y coreográfica, que con el tiempo se enriqueció y potenció fuertemente, gracias a la intervención de instrumentos musicales de cuerda, como la guitarra.

Desgraciadamente, a pesar de que la guitarra ha ido adquiriendo con el paso del tiempo un protagonismo elevado (hasta el punto de asumir un papel protagonista o solista en sus vertientes más contemporáneas, gracias al aporte realizado por guitarristas de renombre, como son Manolo Sanlúcar o Paco de Lucía), es bien notable que tan solo ha llamado la atención a un reducido número de estudios [32], en los que se observa un claro predominio del cante en sus escritos. Podríamos afirmar que, debido a ello, la guitarra ha estado sometida a un doble estado de marginación: desde *fuera*, por su consideración de “popular y carente de reglas artísticas descifrables”, y desde *dentro*, considerada como elemento menor por los aficionados e intérpretes en el cante y el baile que, desde un comienzo, interactuaban con ella.

No es hasta la aparición de los primeros escritos de Falla [41] o de Lorca [40] sobre ella, donde vemos que los músicos que se han detenido en escuchar atentamente a los *tocaos* (guitarristas), se han percatado de la peculiaridad musical idiomática y han llamado la atención sobre ello.

Por último, cabe destacar que, a pesar de que en apariencia y, originariamente en timbre, la guitarra flamenca no difería demasiado de la guitarra clásica, la necesidad de poder alcanzar una potencia sonora similar a la de la percusión, implícita en el baile, o el torrente de voz en el cante, ha desembocado en la construcción de un instrumento de uso específico.

¹ Jarcha: Composición lírica al final de poemas mozárabes o moaxajas. Zambras: danza popular morisca ejecutada durante festejos.

Existen diferencias de construcción entre la guitarra flamenca [27] y clásica, como son:

- 1-Diferentes posiciones de las barras armónicas en el interior de la guitarra (anverso de la tapa).
- 2-Dimensiones de los arcos: Suelen ser más estrechos en las flamencas.
- 3-Acción de las cuerdas: Es mucho más baja en la flamenca. No sólo es cuestión de bajar las silletas, si no que se utiliza en la construcción de la flamenca un ángulo distinto de unión del mango con los arcos y la tapa que en la clásica.
- 4- Tipo de Madera utilizada. Por tradición, en la guitarra flamenca, se utiliza el ciprés macizo español para los arcos y el fondo y el abeto alemán para la tapa. En la clásica, se utiliza el palosanto para los aros y el fondo y para la tapa el abeto y el cedro. Hay que señalar que, dada la escasez y encarecimiento del ciprés, en los últimos años se han popularizado y extendido gracias a conocidos artistas las guitarras flamencas llamadas “negras”, cuyos aros y fondo están elaborados con palosanto.

La guitarra flamenca, frente a la clásica, nos ofrece un sonido más nasal, agudo, brillante y más redondo. La clásica, ofrecerá más armónicos, y de un carácter más suave, frente a ese otro más seco y metálico de la flamenca. Finalmente estos sonidos se logran actuando sobre el volumen de la caja, que suele ser algo más estrecho. La pulsación es también mucho más baja en la flamenca que en la clásica, lo que hace que sea mucho más blanda de toque. Al estar esta distancia más próxima al traste, limitamos la oscilación de las cuerdas dentro de ese límite físico. A estas características propias que repercuten en el sonido del toque de una guitarra flamenca, por su artesanal elaboración siempre único, habrá que añadir la personalidad del intérprete.

En cuanto al modo de tocar la guitarra podemos distinguir dos formas: El toque “por lo fino”, llamado comúnmente también como “toque por arriba”, basado en la realización de trémolos, arpeggios y picados, y el toque “por lo flamenco” o toque “por abajo” en el que predominan los rasgueos y alzapúas, básicos en la expresión flamenca. Este último fundamenta su armonía y ritmo en la estructura melódica que sigue a la voz en aquellas partes de la pieza en que está presente. Podría afirmarse que el toque flamenco es el acompañamiento al canto y al baile durante las partes en que toman protagonismo.

Por otro lado, el toque “por lo fino”, ha sido el estilo precursor de una escuela de guitarristas que enfocaron su música como un ente independiente del acompañamiento, en que la integridad de las piezas son ejecutadas por un concertista o solista.

1.2. El flamenco como proceso comunicativo: Falsetas.

Numerosos expertos coinciden en que cualquier pieza flamenca en la que intervienen varios intérpretes, se establece un proceso de diálogo entre ellos a través de sus ejecuciones. Cada uno de ellos (cantaos, guitarristas y bailaores) implicados en una misma actuación intervendrán de forma coherente y estructurada, es decir, en función de las acciones que uno de los componentes lleve a cabo en el transcurso de la pieza flamenca, los demás aportarán una respuesta lo más adecuada posible, permitiendo un fluido diálogo.

En este continuo intercambio de información musical, es de suma importancia la *falseta*, frase melódica que se intercala entre las sucesiones de acordes de la guitarra (“toque por lo flamenco”) destinados a acompañar al canto para marcar y establecer la propia estructura de la pieza musical. Para ser más exactos, la falseta es una “micro composición” que merece ser explorada independientemente y a fondo pues goza de identidad musical propia, dado que no necesariamente sigue la estructura armónica del acompañamiento que la precede².

² Es importante señalar que muchos a la hora de definir la falseta la denominan también “floreo” y no es correcto, pues el floreo es, dentro una variación, un interludio melódico que sí depende de la estructura armónica que la precede.

Así, la falseta es una de las unidades más importantes en este proceso comunicativo de diálogo que tiene lugar durante la ejecución musical, porque permite avisar al resto de los que intervienen del comienzo o final de una letra o interludio coreográfico.

Para el mayor entendimiento entre las partes, ha de existir forzosamente un profundo nivel de conocimiento del “lenguaje” de los otros que intervienen y con los que uno pretende dialogar. Es por esto que, durante el aprendizaje, desde el punto de vista musical, se requiere llevar a cabo un riguroso análisis de la ejecución del resto de artistas que intervienen al mismo tiempo.

Sin embargo, debido a que buena parte de los intérpretes y aficionados a este género no han recibido una preparación académica en el marco de la teoría musical, se ha fomentado una gran heterogeneidad conceptual en su estudio.

La dificultad ofrecida por el flamenco para ser estudiado obedece a la transmisión oral en que se apoya; su complejidad, a la cultura popular a la que pertenece. Tiene vertientes artísticas musicales y plásticas, (...) políticas, sociológicas, migratorias, económicas, laborales, comerciales, (...) lingüísticas, literarias, poéticas, religiosas, étnicas. Sus apoyos documentales son escasos y, si hemos de ser rigurosos, estos empiezan con (...) el gramófono (...) [1].

Los avances en la tecnología musical y su industria durante los siglos XX y comienzos del XXI, marcaron un gran cambio en el mundo del flamenco. Insignes analistas, como José Antonio Rodríguez, inciden en la importancia y trascendencia de las nuevas tecnologías en la música [2] conocida también como Informática musical, habiendo logrado no sólo nuevos medios para almacenarla y reproducirla fielmente en un formato compacto, sino mejorar el rendimiento y la obtención de una mayor capacidad (especialmente con los formatos digitales) lo que ha provocado y facilitado el interés de profesionales y aficionados por inmortalizar letras, melodías y subgéneros que anteriormente sólo podían haber sido transmitidos de generación en generación.

A su vez, los nuevos objetivos logrados en cuanto a la capacidad de grabar, procesar, editar y aislar por pistas los distintos instrumentos que intervienen en una grabación, facilitan la asimilación de técnicas, giros y secciones melódicas que en una simple escucha podrían resultar más difíciles de asimilar o incluso pasar desapercibidos.

Los vertiginosos avances tecnológicos de las últimas décadas y la popularización de Internet, junto con creación de servicios en línea, permitieron a su vez que podamos tener la posibilidad de reproducir miles de canciones grabadas a través de nuestros dispositivos móviles, ordenadores o tablets. Sin embargo, esta misma ventaja, nos exige *buscar nuevas formas de indexación, clasificación o análisis* de archivos musicales.

Ante esta oscilación pendular de las épocas y sus gustos y una flamencología esteticista, casticista, mediatizada por una afición pasional, se impone una flamencología objetiva, que use del método científico con rigor, lo cual no puede ni debe impedir que exista la pasión de siempre (...). Estudiar el fenómeno flamenco no es controlar su vida o evolución, aunque un buen estudio del fenómeno sirva para controlarlo en la dirección que se quiera [1].

Para cubrir parte de las necesidades anteriormente señaladas, se están aplicando técnicas computacionales y modelos matemáticos utilizados en el campo de la *Music Information Retrieval* (Recuperación de la Información Musical, MIR) [21], para desarrollar la investigación de la Teoría Computacional del flamenco.

1.3. MIR y Etnomusicología Computacional

MIR (*Music Information Retrieval*) es un área interdisciplinar dedicada a la recuperación de información de la música, abarcando todos los procesos de acceso a material digital sobre la misma [26]. Durante los últimos diez años, MIR ha detectado y resuelto una gran variedad de problemas tales como [25]:

* Identificación de audio or *fingerprinting* (huella digital, mecanismo de defensa de derechos de autor).

* Identificación de versiones de canciones (identificación de infracciones de Copyrights).

*Alineamiento y sincronización de audios, *matching* [24] (asociación de grabaciones a patrones musicales).

*Reconocimiento de instrumentos y separación de una grabación por pistas.

**Query by humming* (identificación de melodías cantadas o tarareadas).

* *Music Generation* (generación automática de música)

MIR es un ámbito en continuo crecimiento que involucra a distintas ramas de conocimiento como pueden ser: la Psicología, la Formación Musical Académica, Tratamiento de Señal, Acústica, Matemáticas, Informática, etc. Todas ellas comparten una misma visión en el diseño y la construcción de herramientas que nos ayuden a organizar, comprender y buscar en grandes colecciones de música.

Las aplicaciones del MIR pueden ser, entre otras:

- Sistemas de recomendación musical.
- Separación y reconocimiento de instrumentación.
- Transcripción automática.
- Caracterización automática.

Desde los comienzos de MIR como campo de conocimiento, la mayoría de las técnicas y modelos utilizados, han sido desarrollados para la música tradicional, conocida como música popular de tradición occidental. En los últimos años ha surgido un especial interés en aplicar las técnicas desarrolladas por el MIR para el estudio de otras músicas, también tradicionales, como el folk o la música étnica de lo cual podrían beneficiarse considerablemente etnomusicólogos y profesionales de la música en el curso de sus investigaciones. El reciente interés por el estudio tecnológico interdisciplinar de las músicas de tradición oral, ha llevado a la consideración de una nueva área de investigación conocida como *Etnomusicología Computacional*.

La Etnomusicología Computacional se remonta a 1978 [3] cuando Halmos, Köszei y Mandler, dos matemáticos y un ingeniero proporcionaron una interesante reflexión sobre el papel de los ordenadores en las cinco principales áreas de investigación en Etnomusicología: recopilación de datos, administración, notación, selección y sistematización, y tratamiento científico. El término fue recientemente redefinido por Tzanetakis [28] como “el diseño, desarrollo y utilización de herramientas informáticas que tienen el potencial para ayudar en la investigación etnomusicológica”.

En lo referente a la etnomusicología computacional, Tzanetakis señala que existe una necesidad de colaboración entre los etnomusicólogos y técnicos para crear un campo de investigación interdisciplinar; que las grandes colecciones de audio son importantes para desarrollar enfoques automatizados, utilizando

técnicas de aprendizaje de máquinas que requieran grandes cantidades de datos para el entrenamiento, anotadas manualmente por expertos; que las técnicas de dominio específico son necesarias para estudiar las características y limitaciones de los diferentes estilos musicales; que el contenido y el contexto de interfaces que permiten la conservación pre-activa deben desarrollarse para una mayor accesibilidad de los usuarios no especialistas; y que la captura de movimiento corporal durante el proceso de hacer música mediante el uso de sensores colocados en el cuerpo y o instrumento, proporciona una nueva metodología que ofrece perspectivas interesantes, que podrían ser aplicadas a la música étnica. Este último reconoce de alguna manera el hecho de que el audio podría no ser suficiente para describir y / o recuperar la música étnica.

Con este enfoque, la Etnomusicología es la ciencia que, sirviéndose de los avances en otras disciplinas tales como Matemáticas Computacionales, Musicología o estudios de carácter socio-cultural, posibilitará la reestructuración de problemas y la percepción de las relaciones entre sus elementos bajo una perspectiva diferente [22].

1.4 Rasgos y aportaciones. Proyecto CoFla

Las herramientas utilizadas en el área de MIR son técnicas genéricas de procesamiento de señales de audio, aprendizaje automático (*machine-learning*) o algoritmos de modelado de la percepción, entre otros. Debido a que MIR se encuentra en un ámbito de investigación relativamente reciente, los sistemas existentes todavía no están preparados para su uso específico en algunos estilos musicales, por lo que se requiere poseer conocimientos técnicos y artísticos para su configuración y uso.

Ante la necesidad de adaptar y optimizar soluciones para el flamenco surge el *Proyecto CoFla*³, estableciendo las bases para la investigación de la *Teoría Computacional del Flamenco* [29] donde, a través de modelos computacionales, se pretende apoyar el análisis, caracterización y síntesis del flamenco, conceptos en los que principalmente se fundamentan las líneas de investigación más actuales de este grupo interdisciplinar; en el cual expertos de distintas materias, tales como la etnomusicología, literatura, matemáticas, ingeniería o psicología musical, colaboran entre sí para aproximarnos a una mejor comprensión de este singular género.

Este es el marco donde se desarrolla este proyecto. Algunos de los problemas relevantes en el flamenco y la *Flamencología Computacional* merecen un enfoque diferente en comparación con otras vertientes musicales. Así, se han tratado problemas de investigación desde distintas presteativas como:

a) *Matemáticas y Flamenco*. [15, 16] El estudio desde una perspectiva matemática de ciertas características de esta música, fue el punto de partida para el grupo de investigación CoFla y del área que fue establecida y fundamentada [20] por éste: *Análisis Computacional de la música Flamenca (Musicología Computacional)*.

³ <http://www.cofla-project.com/>

b) *Extracción de Ritmo y Armonías*. [30] Comprobamos en numerosas grabaciones antiguas que el cantaor iba *por libre* ejecutando una letra sin un patrón rítmico o armónico claramente definido, viéndose el guitarrista obligado a predecir intuitivamente las notas que se suceden durante la interpretación de una o varias letras, añadiendo además la armonía y ritmos necesarios. En determinadas piezas, como los tientos o las alegrías, en las que se suele comenzar con un ritmo más lento, un *allegro* (aceleración rítmica) o un cambio de tono enfatizan el final de la letra de un cante e incluso el cambio de un palo a otro. Si bien es cierto que, la práctica y ensayo constantes permitirán con el tiempo a ambos músicos alcanzar una buena sincronización, indudablemente les sería de gran ayuda y utilidad gozar de una herramienta capaz de predecir estos cambios [23] en una única escucha, facilitando la laboriosidad didáctica y profesional que implica actualmente.

c) *Transcripción Automática del Cante*. Algunos de los algoritmos de MIR permiten encontrar con facilidad un patrón melódico y su transcripción a partitura como, por ejemplo, la transcripción de sinfonías de música clásica a partir del archivo de audio. Aplicar este tipo de algoritmos al flamenco es algo que aporta malos resultados. Originariamente, debido a las toscas condiciones y la falta de rigurosidad a la hora de ejecutar una pieza flamenca, el cantaor solía dar su tonalidad para que el guitarrista afinase su instrumento, imponiendo en numerosas ocasiones la aparición de afinaciones microtonales (distancias más pequeñas a un semitono). Si a esto unimos las modulaciones y melismas que suele realizar la voz [13], extraer y transcribir todo ello se convierte en una tarea harto compleja.

d) *Similitud Musical*. [31] El problema de la similitud melódica del flamenco [12] se comenzó a tratar desde el principio del desarrollo del proyecto. Resulta uno de los problemas de mayor dificultad por afrontar en el ámbito de la Musicología Computacional [6] y, para el grupo CoFla, ha desembocado en numerosos subproblemas a desarrollar (limpieza de audios y ampliación de corpus de tonás [10], realización de tests que muestren ausencia de *overfitting*, segmentación de *pitch* (contorno melódico), estudio de la voz flamenca [19], timbre y sistemas de clasificación [14], segmentación de Melodías,...)

Otros problemas que han abarcado en sus estudios son los siguientes:

- *Clasificación de estilos / palos flamencos*.
- *Análisis de estructuras métricas* [40].
- *Detección de patrones melódicos*.
- *Análisis Semántico*.
- *Identificación de cantaores* [17] [18].
- *Música y emocionalidad*.

Al igual que la mayoría de los problemas computacionales, es importante subdividir su estudio en problemas más sencillos, “*divide et vinces*”. En el caso del análisis musical, puede separarse su estudio en tres aspectos esenciales: **La melodía, armonía y ritmo**. Como bien puede apreciarse en los puntos anteriormente mencionados, los problemas caracterizados por CoFla, versan, esencialmente, en el ámbito de la melodía y la armonía y es en este contexto en el que da comienzo la búsqueda para la solución a un problema específico por resolver, que será el objeto de estudio de este proyecto.

A fin de facilitar el proceso de indexación y clasificación de archivos musicales, o su posterior comparación con fragmentos monofónicos, este proyecto pretende realizar un algoritmo que, basado en la detección melódica de voz junto con otros parámetros, permita **identificar el número de falsetas de guitarra en una grabación y extraerlas automáticamente** para facilitar posteriores estudios de la música flamenca. Además, por tratarse de un medio automático, posibilitará una forma más accesible para la conservación del patrimonio cultural en una basta colección de grabaciones.

2. ENTORNO DE DESARROLLO DEL ALGORITMO

2.1. Herramientas utilizadas para la extracción automática.

1. **Sonic Visualiser.** Software de carácter libre que facilita el estudio de grabaciones musicales, especialmente diseñado para musicólogos o investigadores de procesamiento de señales de audio. Algunas de las funcionalidades que incorpora (y que son utilizadas en el proceso de extracción) son la visualización del espectrograma de una grabación, anotación de datos específicos del audio a través de etiquetas, extracción de características a través de *plugins*, tales como detectores de tonalidad (*pitch detectors*) o seguidores de ritmo (*beat trackers*).

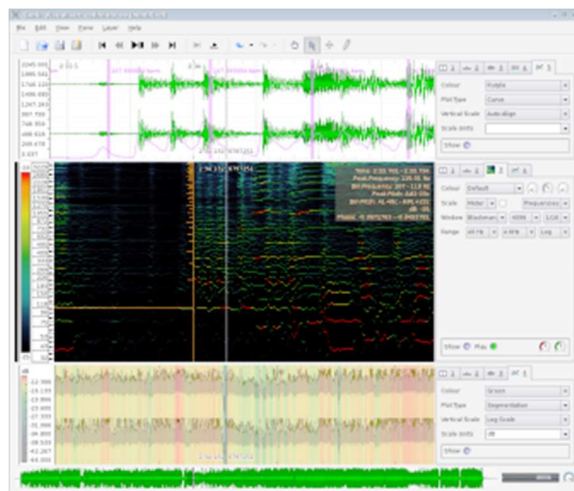


Figura 2-1. Interfaz gráfica de Sonic Visualiser con un archivo de audio abierto, en que se muestra una grabación de audio en tiempo y en frecuencia.

2. **Matlab R2015a.** (*MATrix LABORatory*), es una herramienta de software matemático que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación estructurado, disponible para diversos sistemas operativos. Aunque se trata de una distribución por la que es necesario abonar para su uso, existe una versión de estudiante con algo de descuento. La necesidad de recurrir a este IDE y no otros libres, como Octave, se debe a que incluye toolboxes de procesamiento de señal y audio (*Audio System Toolbox, Signal Processing and Communications Toolbox*) que facilitan algunas de las tareas que se llevan a cabo (apertura / guardado de archivos de audio, análisis frecuencial,...).

3. **Plugin MELODIA.** MELODIA [2] es un plugin desarrollado por el Music Technology Group de la Universitat Pompeu Fabra, a través del cual, dada una grabación de audio de una pieza de música, realiza una estimación automática de los tonos (“*pitch contour*”) dentro de la melodía polifónica que conforma una canción, el contorno melódico, compuesto por las notas tónicas que aparecen en cada instante de la grabación.

Con *polifonía* nos referimos a la música en que dos o más notas pueden sonar simultáneamente, debido a la simultaneidad sonora de dos o más instrumentos (ej. Una voz, una guitarra y un bajo sonando a la vez) o bien porque un instrumento concreto pueda ser capaz de tocar más de una nota al mismo tiempo (ej. piano).

Para comprender con claridad el concepto de contorno melódico, es necesario definir correctamente lo que es una *melodía*. Aunque posee diversas acepciones, tomaremos como referencia la definición musicológica que adoptó el MIR:

“La melodía es una secuencia de tonos monofónicos que el oyente puede reproducir si se le pide silbar o tararear una pieza polifónica, además de poder reconocerla como la “esencia” de esa música en comparación con la original”.

Así, para extraer el contorno melódico de un archivo de audio y exportarlo como texto plano, hemos de abrirlo en Sonic Visualiser. Una vez abierto, basta con hacer click en: Transform > Analysis by Category > MELODIA –Melody Extraction...

Dentro de la ventana emergente, debemos indicar los siguientes parámetros:

PROGRAM: POLYPHONIC

Min Frequency 120 Hz

Max Frequency 720 Hz

Voicing Tolerance 0,2

Monophonic Noise Filter 0,0

El motivo por el que se escogió este intervalo frecuencial, es que, a través de diversas pruebas, se descubrió es el que mejor capta y extrae el registro de voz en grabaciones polifónicas. Así, el archivo de texto resultante de la extracción polifónica, se trata de una tabla en que se indica la cantidad de energía que hay para cada instante de tiempo dentro de la duración total del audio. Si la energía es positiva, significa que, para ese instante de tiempo, se ha registrado contorno melódico en la canción. En caso de ser negativa o nula, se tratan de instantes de tiempo en que no se detectó tonalidad comprendida en el rango vocal.

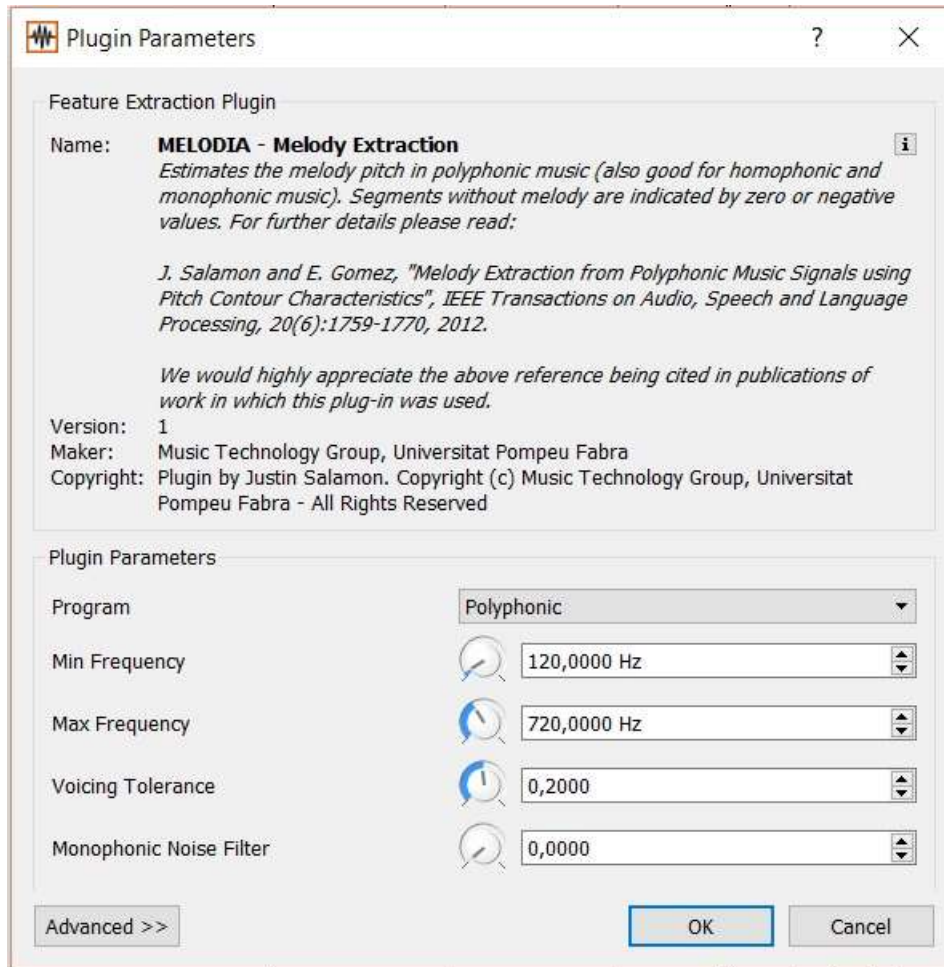


Figura 2.2. Parámetros a introducir en el plugin MELODIA.

```

114.726893424,206.456
114.729795918,206.456
114.732698412,206.456
114.735600907,207.652
114.738503401,208.855
114.741405895,210.065
114.744308390,211.282
114.747210884,-280.403
114.750113378,-280.403
114.753015873,-278.788
114.755918367,-277.183
114.758820861,-278.788
114.761723356,-278.788
114.764625850,-278.788

```

Figura 2.3. Tabla resultante de la extracción de voz, dentro del fichero de extensión (.csv). Se muestra un intervalo de tiempo (114.7268s hasta 114.7646s) en que se pasa de tener energía positiva a negativa. Esto significa, en 114.7472s, la voz deja de estar presente, pero hay otros contornos melódicos presentes. Tener un 0.00 significa que en ese momento hay silencio en la grabación.

3 ALGORITMO DE EXTRACCIÓN DE FALSETAS

La función de un buen software es hacer que lo complejo aparente ser simple.

- Grady Booch -

SI bien ya fueron introducidas en el apartado anterior las herramientas que procesarán un corpus de archivos de audio y extraerán toda la información relevante, a continuación se detalla, paso a paso, cada uno de las etapas (y funciones Matlab) que hacen posible la implementación de un algoritmo de extracción, así como las razones por las que han sido incorporados.

En un primer experimento, se trató de extraer falsetas de guitarras proveniente de un corpus de canciones en que tan sólo intervienen guitarra y voz, tan solo con el Plugin MELODIA, pues al tener la posibilidad de devolver el contorno melódico de un instrumento polifónico, debía ser una tarea que pudiese llevar a cabo sin complicaciones. Sin embargo, los resultados de este procedimiento fueron pésimos. El motivo fundamental se debe a que la voz, con un rango frecuencial aproximado de entre 70-700Hz, pisa buena parte de las frecuencias fundamentales de una guitarra (82-988 Hz, sin contar los armónicos). Si en determinados instantes de la grabación la cantidad de energía de ambas fuentes de sonido es parecida, durante el proceso de extracción polifónica, los contornos melódicos de guitarra y voz se confundirían, lo que dificulta la obtención de un patrón fiable para determinar la sección en que transcurre una falseta. La implementación de alternativas para la extracción automática de segmentos de audio de un instrumento de cuerda [11] aporta resultados poco favorables debido a este mismo problema, por lo que fue necesario plantear una estrategia diferente.

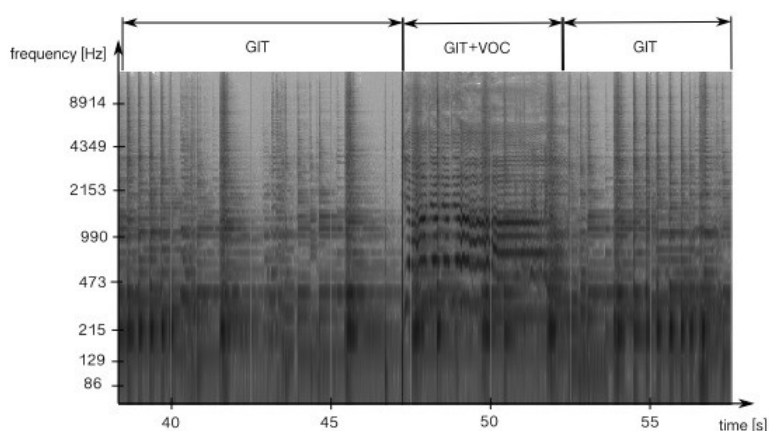


Figura 3.1. Espectrograma obtenido con Sonic Visualiser. GIT representa el grupo de frames en que solo hay presente contorno melódico de guitarra. VOC aquellos frames en que hay voz. La voz siempre tiene contorno melódico de guitarra superpuesto, lo que supone un problema para la extracción polifónica de falsetas.

Nuestra propuesta en ese proyecto se basa en un método de extracción y transcripción de voz a partir de grabaciones polifónicas [4]. Partiendo del supuesto de que las falsetas se encontrarían en aquellas secciones de la canción en que no interviene la voz, basta con restar los intervalos de tiempo en que intervienen la voz a la canción completa. Esto nos permite obtener secciones de audio más pequeñas en las que acotar predictivamente una falseta de guitarra.

El algoritmo de extracción automática de falsetas consta de los siguientes pasos:

- a) Obtención de archivo de datos (.csv) para cada grabación de audio (en .wav)
- b) Separación de canales estéreo, tomando aquel de los dos en que la voz tiene una mayor presencia.
- c) Extracción de contorno melódico predominante en toda la grabación.
- d) Filtrado gaussiano a partir de las bandas de Bark. Definición de *frames* vocales, aquellos en que aparece exclusivamente la voz.
- e) Extracción de guitarra. Restando los *frames* de voz a la grabación completa, se obtienen los *frames* que contienen información de guitarra.
- f) Delimitación temporal de los *frames* de guitarra. Descartamos intervalos menores a una duración indicada en segundos. Aquellos que prevalecen, son considerados *falseta*.

3.1. Algoritmo de extracción automática de falsetas.

Partiendo de que, para cada canción a analizar siempre estamos en posesión del archivo de datos correspondiente, obtenidos a través del plugin MELODIA, así como la grabación en formato .wav, con una compresión de 44100 Hz, se realiza el análisis y extracción de falsetas:

3.1.1. Separación de canales estéreo.

La selección automática de canal se basa en la distribución espectral de energía. Cuando la voz está presente, hay un incremento en el rango frecuencial, que va de 500Hz a 6kHz. Por ello seleccionamos el canal que tiene mayor presencia media en este rango. Siendo la frecuencia de muestreo 44.1kHz, se aplica la transformada en tiempo de Fourier, STFT (*Short Time Fourier Transform*) para un tamaño de ventana de 4096 muestras, zero padding factor $m=2$, y tamaño de salto de $h_s = 1024$ muestras.

$$k(f) = \text{round} \left(\frac{f \cdot m \cdot N}{f_s} \right)$$

La caracterización de presencia de la voz, $S[n]$, se realiza a través un ratio de banda espectral, un cociente de la suma de magnitudes en la banda superior ($f_{21} = 500 \text{ Hz} < f < f_{22} = 6 \text{ kHz}$) con la banda inferior ($f_{11} = 80 \text{ Hz} < f < f_{12} = 400 \text{ Hz}$):

$$S[n] = 20 \cdot \log_{10} \left(\frac{\sum_{k(f_{21} < k < f_{22})} |\dot{X}[k, n]|}{\sum_{k(f_{11} < k < f_{12})} |\dot{X}[k, n]|} \right)$$

A fin de eliminar parte del volumen excesivo, se divide el espectro de magnitud del frame por su valor máximo (la norma, $|\hat{X}[k, n]|$), para un tiempo de frame n .

3.1.2. Definición de las bandas de Bark.

La escala Bark es una escala psicoacústica que fue establecida como una de las primeras medidas subjetivas de audio, la distinción del *timbre*.

El *timbre*, es una cualidad esencial que caracteriza un sonido. Puede ser agudo o grave, según la altura de la nota que corresponde a su resonador predominante. Se trata de una de las cuatro cualidades esenciales del sonido, junto con el tono, la duración y la intensidad). Tiene un rango de 1 a 24 y corresponde con las primeras 24 bandas frecuenciales críticas del oído.

La conversión de una frecuencia a bandas de Bark se realiza a través de la siguiente expresión:

$$Bark = 13 \cdot \arctan\left(\frac{0.76 \cdot f}{1000}\right) + 3.5 \cdot \arctan\left(\left(\frac{f}{7500}\right)^2\right)$$

En el caso de la voz en el flamenco, tal y como se detalla en [53], las 12 primeras bandas frecuenciales son las que determinan la existencia de cante o no. Con ello, para distinguir los intervalos de tiempo o *frames* que contienen información vocal, se considerará un vector multidimensional de 12 variables, $x[n]$, en que se estudia, para una primera pasada en el archivo de audio, la distribución gaussiana de la energía para las 12 primeras bandas de frecuencias más bajas, 50 Hz, 100 Hz, 150 Hz, 200 Hz, 300 Hz, 400 Hz, 510 Hz, 630 Hz, 770 Hz, 920 Hz, 1080 Hz y 1270 Hz.

La campana de Gauss obtenida en cada una de estas bandas, es notoriamente característica, y poseerá siempre una morfología muy diferente al contorno que se generaría a partir de información vocal.

A pesar de que la información frecuencial de la distribución difiere considerablemente de aquellos segmentos en que tan sólo aparece la guitarra, no se puede generalizar (el timbre entre una voz y otra distinta varían notoriamente, incluso para una misma voz en distintas grabaciones), por lo que es un paso que no puede obviarse para una futura optimización.

3.1.3. Filtrado Gaussiano de muestras. Obtención de frames vocales.

Basándonos en el algoritmo detallado en [5], se efectúa un suavizado (smooth) para 4096 muestras, con un tamaño de ventana de 1024 muestras, un salto de 128 muestras.

La ventana (también de 12 dimensiones), se va desplazando por todas las muestras de la canción, aplicando la transformada de Fourier en cada salto. El resultado de esta transformada de Fourier se filtra a través de las distribuciones gaussianas correspondientes al timbre vocal para cada una de las bandas críticas. El audio que prevalece así en cada ventana, es el correspondiente a aquel que contiene información vocal.

La aplicación de este método, aporta unos resultados muy fiables para la distinción de voz. Este método, en que la covarianza y la media obtenidas nos ayudan a determinar la probabilidad con la que existe o no voz en ese frame, garantiza la obtención de un 90% de los *frames* que contienen voz.

Con ello, la energía obtenida a través de una banda de Bark en el algoritmo viene dada por:

$$B[n] = \sum_{k(f1,m) < k < k(f2,m)} |X[k, n]|^2$$

La información energética se guarda en un vector $x[n] = \langle B[n, 1], \dots, B[n, 12] \rangle$ que contiene la información vocal para las doce bandas para un determinado *frame*.

3.1.4. Frames de guitarra. Definición de falsetas.

Una vez extraídas las bandas de Bark con la información vocal, la obtención de la información sonora de la guitarra (falsetas) se encontrará en las partes en que no se han detectado *frames* de voz ($1 - x[n]$)

Los primeros resultados arrojados con esta suposición tenían una duración temporal muy heterogénea (se daban casos de “falsetas” de 3s hasta de 40s – 1min). Por ello, siguiendo las indicaciones del musicólogo, los *frames* de guitarra de duración menor a 15s, al igual que los de voz, son descartados.

El código consta de un script principal (*final.m*) que llama a varias funciones, las cuales se detallarán en las próximas páginas. Como argumentos de entrada, recibe los archivos de audio en .wav, los correspondientes archivos de texto de la extracción con MELODIA, y el tiempo (en segundos) de falseta que hayamos escogido. La salida, es la duración de cada una de las falsetas detectadas, representada matricialmente junto a la canción que corresponde y el número de falsetas en esa canción. Además, se crean automáticamente los archivos de audio en que tan sólo se oye la guitarra, las falsetas que se han detectado en cada muestra de audio.

3.2 Implementación del algoritmo

El script *final.m* (ver código en Anexo I), es aquel que gestiona y controla globalmente la extracción automática de falsetas.

Recibe como parámetro único la duración mínima de las falsetas que se desean extraer. Devuelve un vector con todos los tiempos de falseta obtenidos para un corpus, (compuesto de archivos wav+csv), del que debemos indicar el directorio en que se encuentra (las variables música y files almacenan las ubicaciones de .wav y .csv, respectivamente). El algoritmo prosigue su ejecución, siempre y cuando exista un igual número de archivos y audios, que han de coincidir en nombre (*Ej: a audio_ejemplo1.wav, le correspondería un fichero audio_ejemplo1.csv*).

Una vez se comprueba esto, el script procede a examinar de forma independiente, uno a uno, cada uno de los archivos de audio, junto con su información correspondiente, llamando a la función *SegmentGuitarVocals*, que recibe como parámetros de entrada el fichero de audio .wav, el .csv y el tiempo de falseta.

A continuación, se realiza llamada a *SegmentGuitarVocals.m*. En esta función se elabora la mayor parte del procesado de las canciones. Recibiendo los parámetros anteriormente comentados (canción, fichero correspondiente y tiempo de falsetas), devuelve un vector que contiene la segmentación de la pieza (0 si es guitarra, 1 si es voz). Un vector con los tiempos de falseta y un número *h*, número de falsetas dentro de la canción.

En primer lugar, se lee el fichero de audio, en que se encontraba el contorno melódico de la canción a procesar. La primera columna contiene el instante de tiempo, la segunda la energía vocal en dicho instante. Si el valor es negativo significa que no hay melodía presente, por lo que se coloca a 0.

Tras ello, la función *barkbands.m*, extrae los frames con información vocal, tal y como se comentaba anteriormente. Cada Banda de Bark es colocada de forma consecutiva a través de la creación de una tabla que, dinámicamente, varía su tamaño según se van obteniendo nuevos frames.

Este vector, contiene el valor en segundos del comienzo y final de cada frame. Para determinar un frame de guitarra a partir de uno vocal, basta con considerar que la terminación de un segmento de voz es el comienzo de uno de guitarra. A su vez, el comienzo del siguiente segmento de voz determina el final del frame de guitarra. Esto nos permite obtener el contorno melódico en que tan solo suena la guitarra con bastante probabilidad.

Debido a que los primeros resultados obtenidos con el corpus daban resultados en que existían gran cantidad de falsetas (del orden de 20 y 30), se aplicó una nueva restricción a través del tiempo mínimo de duración (30 segundos, 20, 18, ...) hasta obtener unos resultados similares a la duración estimada que determinaron los expertos.

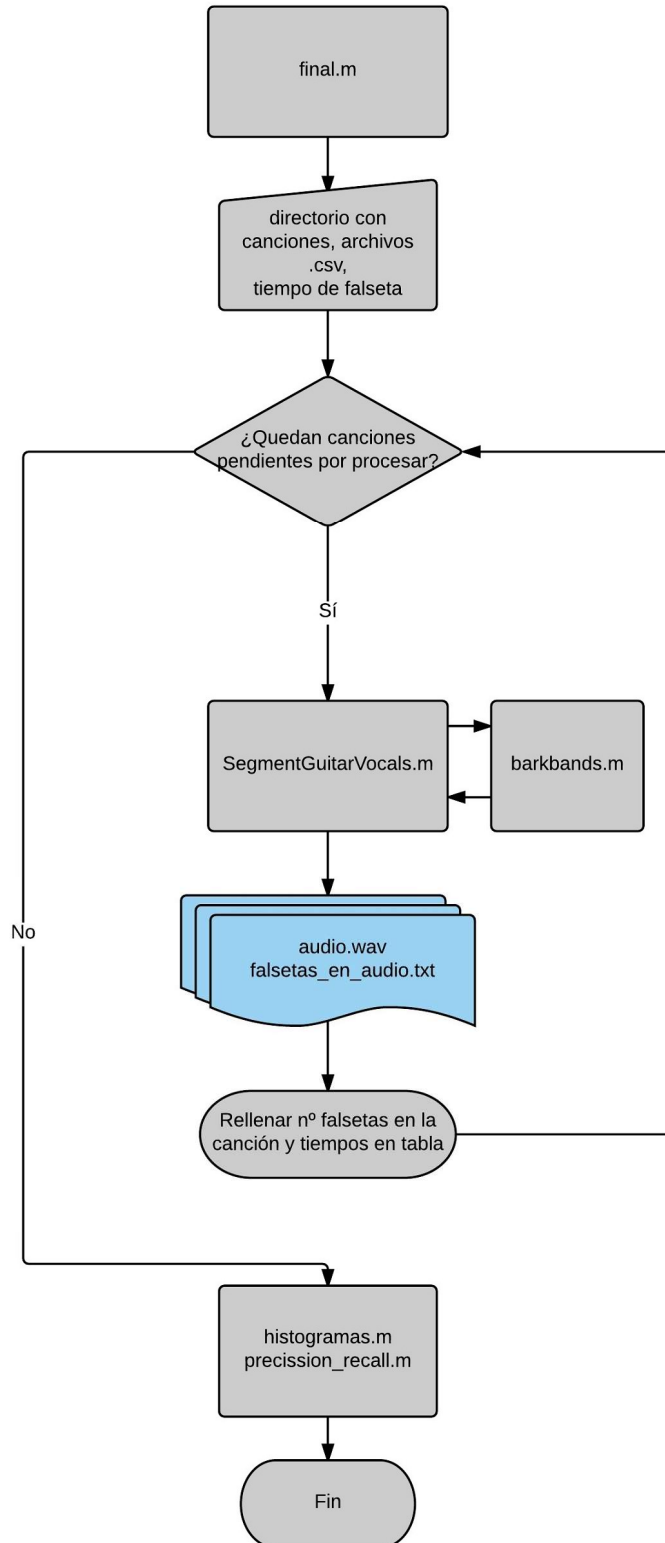


Figura 2.4. Diagrama de flujo del algoritmo de extracción de falsetas.

4 OBTENCIÓN DE RESULTADOS. CONCLUSIONES.

4.1. Histogramas.

Una vez obtenidos los vectores con el total de falsetas (un vector para las registradas manualmente, otro para los resultados automáticos obtenidos con MatLab), se realizó una representación que permitiese visualizar la relación del número de falsetas encontradas en total con respecto a la duración de cada una de ellas.

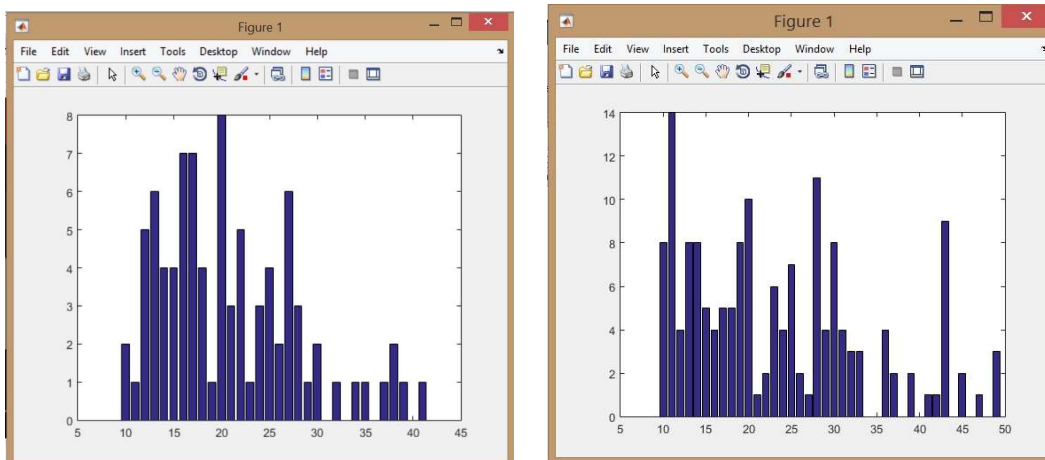


Figura 4.1. Histogramas de falsetas: a) Detectadas manualmente; b) Detectadas con el algoritmo.

Tal y como puede apreciarse, existen una cantidad de intervalos menores a 15s que Matlab ha considerado como falsetas y secciones mayores a 35-40s que también fueron consideradas falsetas. ¿Pero hasta qué punto se trata de una falseta o no lo detectado? Podría haberse dado un error de transcripción manual o bien el error puede provenir de la extracción automática por parte de Matlab. Para contrastar esta información con otros estadístico usaremos medidas de rendimiento usuales en MIR.

4.2. Precision-Recall (Precisión y Exhaustividad).

Se tratan de medidas de relevancia, empleadas en la medida del rendimiento de los sistemas de búsqueda y recuperación de información y reconocimiento de patrones. En este contexto, se denomina *precision* como la fracción de instancias recuperadas que son relevantes, mientras que *recall* es la fracción de instancias relevantes que han sido recuperadas.

		Condición predecida	
		Condición predecida como positiva	Condición predecida como negativa
Condición Verdadera	Condición Positiva	Verdadero Positivo (tp)	Falso Negativo (fn)
	Condición Negativa	Falso Positivo (fp)	Verdadero Negativo (tn)

Tabla 4.1. Recoge los parámetros con los que se calculan la precisión y exhaustividad.

En la detección automática de falsetas, se puede dar el caso de que un *frame* obtenido en MatLab corresponda a un instante de tiempo dentro de los intervalos que se preestablecieron como tiempo de falseta (verdadero positivo), que donde debía haber falseta, no se ha detectado (falso negativo). Que se haya obtenido un *frame* considerado falseta, sin serlo (falso positivo) o que, simplemente, el algoritmo de MatLab y la extracción manual, coincidieron en que para un *frame* concreto, no existía falseta (verdadero negativo).

$$Precisión = \frac{tp}{tp+fp} \qquad Exhaustividad = \frac{tp}{tp+fn}$$

La media armónica o balanceada de estos dos términos se denomina F-measure.

$$F - measure = 2 \cdot \frac{precisión \cdot exhaustividad}{precisión + exhaustividad}$$

4.3. Resultados. Optimización futura del algoritmo.

Los resultados de mayor relevancia, además de los archivos de audio en que tan solo suenan las falsetas, automáticamente exportados y guardados en formato .wav, se obtuvieron a partir de la aplicación de los estadísticos anteriormente explicados:

$$precisión = 0.5695$$

$$recall = 0.8563$$

$$f-measure = 0.6841$$

Una precisión del 56% nos indica que el algoritmo de extracción automática aún debe mejorarse. En lo que sigue analizamos posibles mejoras del algoritmo.

Al escuchar parte de las falsetas extraídas y cotejarlas con el tramo originalmente preestablecido, se observa que en gran parte de éstas, existe una gran cantidad de falsos positivos y verdaderos negativos, en que la falseta está “recortada” debido a los *jaleos* de los palmeros o el mismo cantautor al inicio de la falseta, elogios verbales que exclaman el resto de intérpretes cuando el guitarrista demuestra gran habilidad técnica y/o emocional en la ejecución de una sección en que es protagonista. Al ser los jaleos considerados por el ordenador como *frames* de tipo vocal, esto hace que el algoritmo considere que se da el comienzo y el final de una falseta de guitarra entre éstos, lo que me subdivide una misma falseta en 2. Omitir los contornos melódicos vocales menores de 2 a 5 segundos (sería necesario realizar nuevos experimentos con el corpus) mejoraría considerablemente el grado de precisión del algoritmo.

Por otra parte, dentro de lo obtenido como falseta de guitarra, se dan situaciones en que el guitarrista prolonga las variaciones y/o acompañamiento antes de dar comienzo a la microcomposición. La adaptación de algún algoritmo de detección de patrones melódico-rítmicos para encontrar rasgueos (lo que fuertemente caracteriza al “toque flamenco”, acompañamiento, y no al toque “por lo fino”, las falsetas) afinaría aún más la predicción y localización de secciones de guitarra.

4.4. Conclusiones.

La guitarra flamenca está adquiriendo un protagonismo excepcional en la cultura contemporánea gracias a la imparable expansión y auge internacional del flamenco. Hasta hace muy pocos años, el aprendizaje y la transmisión del conocimiento del toque estaba limitado a unos pocos, social y geográficamente. Eran las peñas flamencas y las escuelas de música privadas las que, en un principio, promovieron su aprendizaje hasta que su enseñanza fue incluida en los planes de estudio de los Conservatorios Superiores de Música localizados en algunas provincias españolas.

La aparición de los primeros métodos didácticos traducidos a otros idiomas, videolecciones, especialmente los medios de música en red y *streaming* (Spotify, Youtube) han permitido que sean cada vez más los profundos admiradores y aficionados de la guitarra en el flamenco.

Haber realizado una labor de investigación en este ámbito, no sólo despliega la oportunidad de implementar, en un futuro, algún método innovador que acelere y facilite el proceso de aprendizaje de la guitarra flamenca, si no que abre sus puertas a una nueva fórmula de preservar y divulgar este estilo musical, Patrimonio de la Humanidad desde el año 2010.

Como bien se comenta en la sección 1.2., profundizar en la conceptualización del término *falseta* a través de los experimentos realizados, permitirá mejorar la interacción del guitarrista profesional durante la ejecución e interpretación colectiva.

El estudio de las falsetas de guitarra abre a su vez una nueva línea de estudio dentro CoFla y la Flamencología Computacional, pues hasta ahora no se habían realizado estudios similares. La implementación de este algoritmo se podría aplicar en futuros estudios musicológicos para fines como:

* El estudio etimológico de falsetas flamencas. Con una simple búsqueda de una determinada falseta de guitarra en un amplio corpus, podría determinarse el origen de melodías popularizadas a través de falsetas.

* El análisis comparativo entre falsetas y letras cantadas. Durante el desarrollo del flamenco a lo largo de su historia, la guitarra se incorporó posteriormente al canto, a modo de acompañamiento. Pero es a partir de la aparición de guitarristas contemporáneos como Paco de Lucía y otros solistas, cuando se han dado situaciones en que el cantaor es ahora el que trata de imitar (acompañar) a la melodía de la guitarra, y adaptar letras a la estructura armónica de una falseta. Un algoritmo de detección de melodías de una falseta en el canto daría luz a este tipo de investigación.

* Definición formal de falseta. Aunque se arrojó una primera aproximación del término durante el primer capítulo, es necesario ahondar aún más en su conceptualización, ayudado de la colaboración estrecha con otras disciplinas (musical, etnológica,...). Un método automático de extracción de falsetas resulta un operador objetivo para ponderar una buena definición de falseta.

Anexo I: Códigos de Matlab

a) Final.m

```
function [vect_grande]=final(time)

%given a 'falseta' time, final extracts falsetas from directory if the
%audio file has already a melodia file.
%mp3towav;

musica=dir('C:\Users\Javier\Desktop\dale\*.wav');
files=dir('C:\Users\Javier\Desktop\dale\*.csv');

if length(musica)~=length(files)
    disp('ERROR 1: Did you extracted the MELODIA file to each song?')
    return
end
suma=0;
tam=0;
tam_total=0;
cd('C:\Users\Javier\Desktop\dale\');
vect_grande=[0];
for m=1:size(musica,1)

    [~,vect, ~]=SegmentGuitarVocals([files(m).name],[musica(m).name],time);

    if m==1
        vect_grande=vect;
    else
        vect_grande=[vect_grande; vect];
    end
end
end
```

b) SegmentGuitarVocals.m

```

function [ seg,vect, h] = SegmentGuitarVocals( f0File, audioFile, t_int )

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Segments a given flamenco recording into guitar and vocals. %%
%% INPUT: f0File - .csv file containing predominant melody, ex-%%
%%             tracted with MELODIA. %%
%%             audioFile - .wav or .mp3 of the original recording. %%
%%             t_int 'falseta' time duration required %%
%% OUTPUT: seg - vector containing the segmentation: %%
%%             0.. guitar; 1.. vocals %%
%%             <audioFile>_guitar.wav - audio file is created in %%
%%             which vocal parts are %%
%%             silenced. %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%cd('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1 guitarrra\dale\');
%% LOAD FILES
[w, fs] = audioread(audioFile); % read audio file
f0 = load(f0File); % read predominant melody csv file
t = f0(:,1); % first column contains time stamps
f0 = f0(:,2); % second column contains frequency
f0(f0<0)=0; % if the frequency is negative, no melody is present -> set this
to zero

%% CHECK SAMPLE RATE
if fs ~= 44100
    disp('You are using a weird sample rate, please use 44.1kHz!')
    return
end

%% CONVERT TO MONO (here we skip the channel selection...)
if (size(w, 2)==2)
    w = mean(w',' );
end

%% EXTRACT BARKBANDS
B = barkbands(w, fs); % extract bark bands
if length(f0)>size(B,2) % adjust difference in length if necessary
    f0 = f0(1:size(B,2));
    t = t(1:size(B,2));
end

%% VOCAL DETECTION
v_init = zeros(length(f0),1); % initial vocal detection function
v_init(f0>0) = 1; % initial estimation: vocals are where MELODIA estimates a
frequency

B_u = B(:,v_init == 0); % barkbands in voiced frames
B_v = B(:,v_init > 0); % barkbands in unvoiced frames

GM_v=fitgmdist(B_v',1); % fit distribution for voiced frames
GM_u=fitgmdist(B_u',1); % fit distribution for unvoiced frames

voc = zeros(length(v_init),1); % vocal probability

```

```

git = zeros(length(v_init),1); % guitar probability
voc_det = zeros(length(v_init),1); % initialize vocal detection output

for i = 1 : length(voc)
    voc(i) = pdf(GM_v,B(:,i)'); % probability that this frame contains vocals
    git(i) = pdf(GM_u,B(:,i)'); % probability that this frame contains vocals
    if git(i) <= voc(i)
        voc_det(i) = 1; % decision: vocals or guitars
    end
end

%% SMOOTH OUTPUT
voc_det=smooth(voc_det,345);
voc_det(voc_det<0.5)=0;
voc_det(voc_det>0.5)=1;

%% FIND SEGMENTS OF CONSCUTIVE VOICED FRAMES -> CONTOURS (from initial
estimation)
startC=[];
endC=[];

startG=[];
endG=[];

if f0(1)>0
    startC=[startC;1];
end
for i=1:length(f0)-1
    if(abs(f0(i+1))>0 && f0(i)==0)
        startC=[startC;i+1];
    end
    if(f0(i+1)==0 && abs(f0(i))>0)
        endC=[endC;i];
    end
end
end

if length(endC)<length(startC)
    endC=[endC; length(f0)];
end

if f0(1)>0
    startC=[startC;1];
end

%% CLASSIFY EACH CONTOUR
seg = zeros(length(f0),1);
for i = 1 : length(startC)
    v_conf = sum(voc_det(startC(i):endC(i)));
    if v_conf>0.0 % contour is not vocals, if all frames are classified as
unvoiced
        seg(startC(i):endC(i)) = 1; % the seg vector is returned...
    end
end

%% FIND REMAINING CONTOURS
%% FIND SEGMENTS OF CONSCUTIVE VOICED FRAMES -> CONTOURS (from initial
estimation)
startC=[];
endC=[];

```

```

if seg(1)>0
    startC=[startC;1];
end
for i=1:length(seg)-1
    if(abs(seg(i+1))>0 && seg(i)==0)
        startC=[startC;i+1];
    end
    if(seg(i+1)==0 && abs(seg(i))>0)
        endC=[endC;i];
    end
end

if length(endC)<length(startC)
    endC=[endC; length(seg)];
end

%%%%%%GUITAR EXTRACTION%%%%%%%%
startG=ones(length(endC)+1,1);
endG=ones(length(startC),1);
for k=1:length(endC) startG(2:end)=endC; end
for k=1:length(startC) endG(1:end)=startC; end

if length(endG)<length(startG)
    endG=[endG; length(f0)];
end

if f0(1)>0
    startG=[startG;1];
end
%TIEMPOS GUITARRA
t_g=ones(length(endG),1); %aquí almacenamos intervalos de tiempo de guitarra
for index1=1:length(endG)
    t_g(index1)=t(endG(index1))-t(startG(index1)); %rellenamos con indices
end

for m=1:(length(t_g)) %elimina muestras menores de t_int segundos
    if t_g(m)<=t_int
        t_g(m)=0;
    end
end
%guardo resultados en fichero

fileID = fopen([audioFile(1:end-4) '_falsetas.txt'],'w');
nbytes = fprintf(fileID,'intervalos guitarra (s)\r\n');
nbytes = fprintf(fileID,'%3.2f\r\n',t_g);
fclose(fileID);

h=0;
for m=1:length(t_g)
    if t_g(m)~=0.00
        h=h+1;
    end
end
vect=zeros(h,1);

for n=1:(h-1)

```

```

    for m=1:length(t_g)
    if t_g(m)~=0.00
        vect(n)=t_g(m);
        n=n+1;
    end

    end

end
%media=suma/h;
%% ELIMINATE VOCAL SECTIONS FROM AUDIO
tAudio = 0 : 1/fs : (length(w)-1)*1/fs; % audio time stamps
for i = 1 : length(startC)
    % frame contains vocals --> set to zero

    [~,s] = min(abs(tAudio-t(startC(i)))); % start sample
    [~,e] = min(abs(tAudio-t(endC(i)))); % end sample
    w(s:e) = 0;
end

for i = 1 : length(startG)
    if t_g(i)<=t_int %extraemos muestras menores de t_int (s) de guitarra
solo
        [~,s] = min(abs(tAudio-t(startG(i)))); % start sample
        [~,e] = min(abs(tAudio-t(endG(i)))); % end sample
        w(s:e) = 0;
    end
end

%% EXPORT AUDIO FILE
audiowrite([audioFile(1:end-4) '_falsetas.wav'], w, fs);

```

c) barkbands.m

```
function [ B ] = barkbands( w, fs )

%% BARKBANDS
% Extracts frame-wise bark band values from an .wav file. A sample rate of
% 44.1 kHz is assumed.

% parameters
winLen=1024;
hopSize=128;
numSamples=length(w);
bands=[0.0, 50.0, 100.0, 150.0, 200.0, 300.0, 400.0, 510.0, 630.0, 770.0,
920.0, 1080.0, 1270.0];%, 1480.0, 1720.0, 2000.0, 2320.0, 2700.0, 3150.0,
3700.0, 4400.0];
win=window(@hamming,winLen);
numFrames=floor((numSamples-winLen)/hopSize);
numBands=length(bands)-1;
B=zeros(numBands,numFrames);
freqScale = (fs*0.5) / (winLen-1);

% extract bands
for i=1:numFrames
    frame=w((i-1)*hopSize+1:(i-1)*hopSize+winLen);
    frame=frame.*win;
    spec=(abs(fft(frame,winLen)));
    spec(spec==--Inf)=0;
    spec=spec(1:round(0.5*length(spec)));
    for ii=1:numBands
        startBin=round(bands(ii)/freqScale+0.5);
        startBin=max(1,startBin);
        endBin=round(bands(ii+1)/freqScale+0.5);
        endBin=min(endBin,length(spec));
        B(ii,i)=sum(spec(startBin:endBin).^2);
    end
    B(:,i)=B(:,i)./max(B(:,i));
end

End
```

d) histogramas.m

```
function histogramas(~)

%representa el histograma de falsetas detectadas por el músico
%y las falsetas detectadas por Matlab.
% we consider falsetas last 60 seconds (max)
matriz_inma=falsetas_teoricos();
matriz_matlab=falsetas_practico();
falsetas_inma=matriz_inma(1:end,2)-matriz_inma(1:end,1);
falsetas_matlab=matriz_matlab(1:end,2)-matriz_matlab(1:end,1);

round(falsetas_matlab); %approach to ints
round(falsetas_inma);
%matlab falsetas histogram
nbins = 130;
ax1 = subplot(2,1,1);
hist(ax1,falsetas_matlab,nbins)
h = findobj(gca,'Type','patch');
h.FaceColor = [0 0.5 0.5];
h.EdgeColor = 'b';
%musician falsetas histogram

ax2 = subplot(2,1,2);
hist(ax2,falsetas_inma,nbins)
h = findobj(gca,'Type','patch');
h.FaceColor = [0.5 0.5 0];
h.EdgeColor = 'b';
end
```

e) precision_recall.m

```

tp = 0;
tn = 0;
fp = 0;
fn = 0;
music=dir('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\*.wav');

files_matlab=dir('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\tiempos_matlab\*.csv');

files_musician=dir('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\tiempos_inma\*.csv');
cd('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1 guitarrra\dale\')
if(isequal(length(files_matlab),length(files_matlab),length(music))==1)

    for m=1:size(music,1)

        cd('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\tiempos_inma\');
        mat_inma=load(files_musician(m).name);
        cd('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\tiempos_matlab\');
        mat_mat=load(files_matlab(m).name);
        cd('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\')
        [y,fs] = audioread(music(m).name);
        SongTime = length(y)./fs;
        t=0:0.1:SongTime;

G = zeros(length(t),1); % ground truth falsetas binary vector
for ii = 1 : size(mat_inma,1)
    [~,sSam] = min(abs(mat_inma(ii,1)-t));
    [~,eSam] = min(abs(mat_inma(ii,2)-t));
    G(sSam:eSam) = 1;
end

E = zeros(length(t),1); % estimated falsetas binary vector
for ii = 1 : size(mat_mat,1)
    [~,sSam] = min(abs(mat_mat(ii,1)-t));
    [~,eSam] = min(abs(mat_mat(ii,2)-t));
    E(sSam:eSam) = 1;
end

tp = tp + sum(E==1 & G==1);
tn = tn + sum(E==0 & G==0);
fp = fp + sum(E==1 & G==0);
fn = fn + sum(E==0 & G==1);
end
precision = tp / (tp+fp);
recall = tp / (tp+fn);
fmeasure = 2 * precision * recall / (precision + recall);

```



```

disp('precision recall f-measure')
%disp([precision recall fmeasure])
else

    disp('Inequality between theoretic-real csv files or/and wav files')
    return
end

```

f) falsetas_practico.m

```

function matriz_total=falsetas_practico(~)
%devuelve la matriz con los comienzos y finales de las falsetas estimadas en
Matlab
matriz=[];
matriz_total=[];
musica=dir('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\tiempos_matlab\*.csv');

cd ('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\tiempos_matlab\');
for m=1:size(musica,1)

matriz=csvread(musica(m).name);
matriz_total=[matriz; matriz_total];

end
cd ('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1 guitarrra\dale\');
end

```

g) falsetas_teorico

```

function matriz_total=falsetas_teoricos(~)
%devuelve la matriz con los comienzos y finales de las falsetas de Inma
(preestablecidos los tiempos previamente)

musica=dir('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\tiempos_inma_mios\*.csv');
matriz=[];
matriz_total=[];
cd ('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1
guitarrra\dale\tiempos_inma_mios\');
for m=1:size(musica,1)

matriz=csvread(musica(m).name);
matriz_total=[matriz; matriz_total];

end
cd ('C:\Users\Javier\Desktop\Disc. Camaron. Falsetas 1 guitarrra\dale\');
end

```


Anexo II: Corpus utilizado

A continuación se detalla el corpus utilizado para las pruebas de extracción automática de falsetas. La gran mayoría de estas canciones pertenecen a la discografía de Camarón de la Isla, en su primera etapa con Paco de Lucía (1969-1977), época en que lanzaron 9 discos, sin contar con las numerosas grabaciones inéditas que aficionados y expertos han recopilado en posteriores antologías. Por otra parte, se han incluido también dos canciones del guitarrista Tomatito en su etapa solista. En el siguiente listado, se puede leer la anotación manual de todas las falsetas que aparecen en estas canciones, junto con el nombre abreviado por el que se referenció para indexar y dar nombre a los parámetros de entrada y resultados a través del código de Matlab.

1. Donde se divisa el mar (cartagenera) 0'-28''; 2'03''-2'23''; "divisaelmar".
2. Hermano mío (seguiriya) 0'-38''; 2'26''-2'53'' "hermanito".
3. No dudes de la nobleza (fandango) 0'-0'30''; 2'07''-2'23''. "nobleza".
4. Reniego haber encontrado (tientos) 0'-0'16''; 1'16''-1'38''. "reniego".
5. Sólo vivo pa' quererte (granaína) 0'-0'38''; 0'56''-1'23''. "paquererte".
6. Y no llegaste a quererme (granaína) 0'-0'41''; 1'15''-1'35''. "aquererme".
7. Maldito yo estaba (seguiriya) 0'-0'24''; 0'56''-1'08'' (Variación); 2'39''-2'49''. "maldito".
8. Mis penas lloraba yo (canastera) 0'-0'16''; 1'43''-2'08''. "canastera".
9. Se me partió la barrena (taranto) 0'-0'35''; 1'36''-2'10''. "barrena".
10. Pueblos de la tierra mía (alegrías) 0'50''-1'08''; 2'17''-2'31''. "pueblos".
11. Que no se quita con ná' (fandangos) 0'-0'22''; 1'37''-1'57''. "nosequita".
12. Calabosito oscuro (seguiriya) 0'-0'25''; 2'01''-2'22''. "calabosito".
13. De tus ojos soy cautivo (soleá) 0'-0'16''; 2'17''-2'35''. "cautivo".
14. Moral (fandangos) 0'-0'17''; 1'06-1'21''. "moral".
15. Yo soy el viento (bulerías) [Tomatito] 0'19''-0'30''; 1'17''-1'37''. "viento".
16. En una piedra me acosté (fandangos) 0'-0'24''; 1'50''-2'05''. "piedra".
17. Sin motivo ni razón (fandango) 0'-0'15''; 1'22''-1'43''. "sinmotivo".
18. Moraito como un lirio (tientos) 0'-0'27''; 1'35''-1'47''; 2'48''-3'10''. "moraito".

19. No quisiera que te fueras (bulerías) 0'10''-0'40''; 2'10''-2'23''. "quetefueras".
20. Sentao en el valle (fandangos) 0'-0'14''; 1'15''-1'28''. "sentao".
21. Ni que me manden a mi (fandangos) 0'-0'28''; 2'02''-2'15''. "memanden".
22. Camina y dime (taranto) 0'-39'' --- 0'39''-1'08''; 3'12''-3'30''. "caminaydime".
23. Las campanas también lloran (tientos) 0'-0'26''; 1'45''-2'04''. "campanas".
24. En tu puerta da la luna (taranto) 0'-0'22''; 2'36''-3'03''. "dalaluna".
25. Los dos se juegan la vida (taranto) 0'-0'32''; 1'59''-2'20''. "lavida".
26. Quisiera volverme pulgar (tangos malagueños) 0'-0'22''; 1'44''-2'00''. "pulgar".
27. El espejo donde te miras (soleá) 0'-0'20''; 0'55''-1'10''; 1'54''-2'08''. "espejo".
28. Arte y majestad (bulerías) 0'49''-1'16''; 1'43''-1'56''. "majestad".
29. Bulerías de la perla (bulerías) 0'-0'27''; 2'33''-2'57''. "laperla".
30. Salud antes que dinero (fandangos) 0'-0'20''; 1'19''-1'35''. "dinero".
31. La vi por primera vez (fandangos) 0'-0'17''; 1'42''-1'59''. "primeravez".
32. Me la tienes controlada (fandangos) 0'-0'17''; 1'18''-1'30''. "controlada".
33. Se pelean en mi mente (tarantos) 0'-0'20''; 0'42''-0'58''. "enmimente".
34. Y tú no me respondías (soleá) 0'-0'26''; 2'06''-2'18''. "respondias".
35. La víbora rabiosa (fandango) [Tomatito] 1'54''-2'12''; 3'08''-3'25''. "vibora".
36. A los santos del cielo (seguriya) 0'-0'28''; 2'42''-2'59''. "santosdelcielo".
37. A un sabio le oí decir (soleá) 0'11-0'26''; 1'26''-1'38'' (Variación). "sabio".
10. Por cositas malas (seguriya) 0'-0'20''; 1'56''-2'08''. "malas".
11. Y me gustan las mujeres (tangos) 0'29''-0'42''; 1'55''-2'12''. "mujeres".
40. Las penas de mi madre (seguriya) 0'-0'35''; 1'02''-1'28''. "penas".

GLOSARIO

Flamenco: Género musical de canto y de baile que se caracteriza por la fusión de elementos musicales andaluces, gitanos y orientales.

Cante Jondo: Canto popular flamenco, de sentimiento profundo y tono quejumbroso.

MIR: *Music Information Retrieval* (ver capítulo 1).

CoFla: *Computational Flamenco* (ver capítulo 1).

Melisma: Grupo de notas sucesivas que forman un neuma o adorno sobre una misma vocal, dentro de un contorno melódico.

Falseta: Microcomposición característica de la guitarra flamenca, de identidad propia, que no necesariamente sigue la estructura armónica del resto de la canción.

Variación: Interludio melódico adherido al acompañamiento entre letra y letra que, a diferencia de una falseta, sí depende de la estructura armónica que la precede.

Contorno melódico: En lingüística y música, el contorno melódico (pitch contour) de un sonido es la curva que sigue el tono percibido en un sonido o canción a lo largo del tiempo.

Frame: En este contexto, es la unidad digital mínima de información energética de una señal de audio en el tiempo. Decimos que un frame no está vacío si contiene parte del contorno melódico a extraer. A partir de un conjunto ordenado de frames, se puede componer el contorno melódico (monofónico o polifónico) de una canción.

REFERENCIAS

- [1] A. Gómez, "Pasado, Presente y Futuro de los Estudios Flamencos", *Seminario - Curso 2000-2001. Servicio de Publicaciones de la Universidad de Córdoba*, 2002, pp 15 - 29.
- [2] J. Salamon, E. Gómez, "Melody Extraction from Polyphonic Music Signals using Pitch Contour Characteristics" *IEEE Transactions on Audio, Speech and Language Processing*, 20(6):1759-1770, Aug. 2012, pp 1-5
- [3] I. Halmos, "Computational Ethnomusicology in Hungary". *MI: MPublishing, University of Michigan Library*, 1978, pp 775-783.
- [4] N. Kroher, E. Gomez, "Automatic Transcription of Flamenco Singing from Polyphonic Music Recordings", *IEEE Transactions on Audio, Speech and Language Processing*, 2015, pp 1-3.
- [5] J. Salamon, E. Gómez, "Predominant fundamental frequency estimation vs singing voice separation for the automatic transcription of accompanied flamenco singing", *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR)*, 2012, pp 602-604.
- [6] W. B. Hewlett, E. Selfridge-Field, (1998). "Melodic similarity: Concepts, procedures, and applications" (Vol. 11). *The MIT Press*. Holzapfel, A., (2010). "Similarity methods for computational ethnomusicology" (*Doctoral dissertation, PhD thesis, University of Crete*), pp. 261.
- [7] D. H. Huson, "SplitsTree: analyzing and visualizing evolutionary data". *Bioinformatics*, 1998, pp 68-73. Itakura, F, "Minimum prediction residual principle applied to speech recognition". *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 1975, pp 67- 72.
- [8] A. Klapuri, M. Davy, "Signal processing methods for music transcription". *Springer*, 2006; Kramer, C., & Plenckers, L. J: "The structure of the saeta flamenca: An analytical study of its music". *Yearbook for traditional music*, 1998, pp 102-132.
- [9] N. Kroher, "Automatic Characterization of Flamenco Singing by Analyzing Audio Recordings", *Master thesis, Master Program in Sound and Music Computing, Universitat Pompeu Fabra*, 2013, pp 5-9.
- [10] P. Lefranc, J.M.C. Bonald, "El Cante Jondo: Del Territorio a los Repertorios: Tonás, Siguiriyas, Soleares". *Servicio de Publicaciones de la Universidad de Sevilla*, 2010, pp 21-22.
- [11] K. Lemström, "String Matching Techniques for Music Retrieval. *PhD thesis*", *University of Helsinki, Department of Computer Science. Report*, 2000, pp 1-9.
- [12] N. Kroher, E. Gómez, C. Guastavino, F. Gómez, "Computational models for perceived Melodic

Similarity in a Cappella Flamenco Singing”, *Universitat Pompeu Fabra* 2013-14, pp. 1-6.

[13] J.Sundberg: “The acoustics of the singing voice,” *Scientific American*, Vol. 236 (3), 1977, pp.104-116.

[14] J. M. Díaz Báñez, J.C.Rizo, “An efficient DTW-based approach for melodic similarity in flamenco singing”. *SISAP conference. Lecture Notes in Computer Science*, (ISBN 978-3-319-11987-8), Springer, 2014, Vol. 8821. pp. 289–300.

[15] F. Gómez, J. M. Díaz-Báñez, J. Mora, E. Gómez, “La música flamenca y su estudio computacional – I”. *Revista en línea Divulgamat*, Abril de 2014, pp. 1- 8.

[16] F. Gómez, J. M. Díaz-Báñez, J. Mora, E. Gómez, E. “La música flamenca y su estudio computacional – II”. *Revista en línea Divulgamat*, Mayo de 2014, pp. 3-5.

[17] N. Kroher, E. Gómez, C. Guastavino, J. Bonada, P. Gómez, (2014) “Computational models for perceived melodic similarity in a cappella flamenco cantes”, *15th International Society for Music Information Retrieval Conference, Taipei, Taiwan*, October 2014.

[18] N. Kroher, E. Gómez, “Automatic Singer Identification For Improvisational Styles Based On Vibrato, Timbre And Statistical Performance Descriptors”, *11th Sound and Music Computing Conference. Greece*, 2014.

[19] N. Kroher, E. Gómez, “Improving accompanied Flamenco singing voice transcription by combining vocal detection and predominant melody extraction”. *11th Sound and Music Computing Conference*. 2014.

[20] N. Kroher, E. Gómez, M. Sordo, F. Gómez-Martín, J.M. Díaz-Báñez, J. Mora, “Computational ethnomusicology: a study of flamenco and Arab-Andalusian vocal music” *4th International Workshop On Folk Music Analysis*, 2014, pp. 2-3.

[21] N. Orio, “Music Retrieval: A Tutorial and Review (Foundations and Trends in Information Retrieval)”, *Now Publishers Inc*, Oct. 2006.

[22] Gutiérrez-Cordero, R. “CoFla, un ejemplo de Investigación Interdisciplinar” *Sociedad Española de Musicología*, Mayo 2013, pp 881-896.

[23] Zapata, J. R., Davies M. E. P., Gómez E. (2014), Multi-feature beat tracking. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. 22(4), pp. 816 – 825.

[24] N. Hu, R.B. Dannenberg, G. Tzanetakis, “Polyphonic Audio Matching and Alignment for Music Retrieval”, *2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2003, pp. 1-4.

[25] M- Schedl, E. Gómez, J. Urbano, "Music Information Retrieval: Recent Developments and Applications", *Now The Essence of Knowledge*, 2014, pp. 132-143.

[26] A. Cortesi, N. Chaki, K. Saeed, S. Wierzchoń, "Computer Information Systems and Industrial Management", *11th IFIP TC 8 International Conference, CISIM 2012, Venice, Italy, Sept. 2012*, pp. 36-45.

[27] <http://www.guitarristaonline.es/noticia/5856/tecnica/que-diferencias-hay-entre-una-guitarra-clasica-y-una-flamenca.html>

[28] G. Tzanetakis, A. Kapur, W.A. Schloss, "the design, development and usage of computer tools that have the potential to assist in ethnomusicological research." "Computational Ethnomusicology" *M. Wright Journal of Interdisciplinary Music Studies*, 1(2), 2007, pp. 1.

[29] J. M. Díaz-Báñez, "Mathematics and Flamenco: An Unexpected Partnership". *Mathematical Intelligencer*, 2016.

[30] J.P. Bello, R. Rowe, C. Guedes, G. Toussaint, "Five perspectives on musical rhythm". *Journal of New Music Research*, 2015, 44(1), pp. 112.

[31] N. Kroher, J. M. Díaz Báñez, J. Mora, E. Gómez, "Corpus Cofla: A research corpus for the computational study of flamenco music" *ACM Journal on Coputing and cultural Heritage*, 2014, pp 10-21.

[32] N. Torres Cortés, "La guitarra flamenca actual", *Seminario - Curso 2000-2001. Servicio de Publicaciones de la Universidad de Córdoba*, 2002, pp 104 -106.

[33] B. Infante, "Orígenes de lo Flamenco y del Cante Jondo", Junta de Andalucía, *Consejería de Cultura. Instituto de las Artes y las Letras*, Reedición 1980, pp 27.

[34] Diccionario de la Real Academia Española, definición de flamenco, 4ª acepción: "De carácter popular andaluz, y vinculado a menudo con el pueblo gitano." <http://dle.rae.es/?id=I2kiw28>

[35] T. San Román, "Vecinos gitanos", *Akal*, 1976, pp. 27-28.

[36] J.C. de Luna, "Gitanos en la Bética", *Epesa. Madrid*, 1951, pp. 44.

[37] L. Corzo, "Gitanos en Castilla", *Boletín de la Sociedad Castellonense de Cultura*, 1964, pp. 11.

[38] M. Martínez, "Los gitanos y las gitanas de España a mediados del siglo XVIII: El fracaso del proyecto de Exterminio (1748-1765)", *Editorial Universidad de Almería*, 2015, pp. 42-43.

[39] A. Guerrero, "La técnica vocal en el cante flamenco", Encuentro para la Plataforma Independiente de Estudios Flamencos Modernos y Contemporáneos (PIE.FMC), *Sevilla, Noviembre 2013*, pp 1-16.

[40] F.G. Lorca, “Importancia histórica y artística del primitivo canto andaluz llamado *cante jondo*”, *Conferencia leída en el «Centro Artístico» de Granada*, Febrero 1922, pp. 2-9.

[41] M. Falla, “Escritos sobre Música y Músicos”, *Espasa-Calpe*, 1950, pp. 40-62.

