



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior de Linares

Trabajo Fin de Grado

HERRAMIENTA DE AYUDA A LA EDUCACIÓN MUSICAL DEL INSTRUMENTO “FLAUTA DULCE” BASADA EN EL ANALISIS DE SEÑALES SONORAS

Alumno: Alberto Sima Andong

Tutor: Prof. D. Francisco Jesús Cañadas Quesada
Depto.: Ingeniería de Telecomunicación

Julio, 2017

ÍNDICE GENERAL

1	CAPITULO 1. INTRODUCCIÓN	3
1.1	Resumen	3
1.2	Introducción	4
1.3	Estructuración de la memoria	7
1.4	Características físicas del sonido	9
1.4.1	Atributos subjetivos del sonido	12
1.5	Teoría musical.....	20
1.5.1	La música como concepto	20
1.5.2	La escala musical	21
1.5.3	Clasificación de los sonidos.....	23
1.5.4	Breve historia de la flauta dulce	30
2	CAPÍTULO 2. ESTADO DEL ARTE	33
2.1.1	Algoritmos en el dominio del tiempo	35
2.1.2	Algoritmos espectrales.....	44
2.1.3	Algoritmos perceptuales.....	46
3	CAPITULO 3. OBJETIVOS	48
3.1	OBJETIVO PRINCIPAL.....	48
3.2	OBJETIVOS SECUNDARIOS	49
4	CAPÍTULO 4. DESARROLLO	50
4.1	PUNTO DE PARTIDA Y PLANTEAMIENTO	50
4.2	MÉTODOS Y MATERIALES	52
4.2.1	Base de datos de música.....	54
4.2.2	Base de datos Audacity	57
4.2.3	Representación tiempo-frecuencia	61
4.2.4	ESTIMACIÓN DE PITCH	68
5	CAPITULO 5. EVALUACIÓN DEL SISTEMA Y RESULTADOS	103
5.1	1ª PARTE. EVALUACIÓN A NIVEL DE FRECUENCIA	104
5.1.1	Base de datos.....	104

5.1.2	Inicialización	108
5.1.3	Métricas utilizadas.....	112
5.1.4	Resultados.....	113
5.2	2ª PARTE. EVALUACIÓN A NIVEL DE TRAMAS.....	119
5.2.1	Base de datos	119
5.2.2	Inicialización	121
5.2.3	Métricas utilizadas.....	121
5.2.4	Resultados.....	122
5.3	EJERCICIOS DEL TUTOR.....	133
6	CAPITULO 6. CONCLUSIONES	137
7	CAPITULO 7. LÍNEAS DE FUTURO.....	138
8	CAPITULO 8. ANEXOS.....	140
8.1	ANEXO I. MANUAL DE USUARIO	140
8.2	ANEXO II. NOMENCLATURAS.....	147
8.3	ANEXO III: ÍNDICE DE FIGURAS	148
8.4	ANEXO III: ÍNDICE DE TABLAS	152
9	REFERENCIAS BIBLIOGRÁFICAS	154

1 CAPÍTULO 1. INTRODUCCIÓN

1.1 Resumen

Una práctica muy común que tienen los amantes de la música a la hora de aprender a tocar un instrumento, es que se ponen a tocar y por medio del oído van evaluando si las notas tocadas son las correctas, y muy a menudo pueden llegar a equivocarse tocando notas que en realidad no son las deseadas; y en vista a esta deficiencia, este Trabajo Fin de Grado, viene a poner a disposición del usuario una herramienta, profesor virtual, que le sirva en todo momento de ayuda en el aprendizaje del instrumento “flauta dulce”. De tal forma que un usuario se disponga a tocar una partitura compuesta de varias notas a tocarse en un determinado tiempo y la herramienta sea capaz de indicarle el porcentaje de aciertos y fallos basados en el análisis de parámetros como la frecuencia que indica la nota tocada y el tiempo que marca el ritmo que debe tocarse cada una de las notas por el usuario.

La herramienta presentada para el aprendizaje de la flauta es un sistema de entrenamiento muy ambicioso en el que el usuario tiene que realizar los ejercicios del tutor que consisten en determinadas pruebas de nivel. Donde el usuario para pasar de un determinado nivel de aprendizaje al siguiente debe obtener como resultado para el nivel en prueba un porcentaje de aciertos en promedio igual o superior al 50% y un mínimo del 40% en cada una de las pruebas; solo entonces el usuario puede pasar de nivel y en caso contrario pues suspende y tiene que repetir el ejercicio. Para realizar las pruebas se establecen condiciones de error en frecuencia y en tiempo. Estas condiciones son en principio impuestas por el tutor que solo permite un error igual o inferior a 40 Hz en frecuencia; mientras que en tiempo solo permite un error de dos segundos. Pero la herramienta es flexible y permite a los usuarios establecer las condiciones que consideren necesarias para su aprendizaje. Los diferentes niveles de aprendizaje que recogen los ejercicios del tutor son:

Nivel principiante: está compuesto por diez pistas con un total de tres notas aleatorias.

Nivel intermedio: está compuesto por diez pistas con un total de cinco notas aleatorias.

Nivel experto: está compuesto por diez pistas con un total de ocho notas aleatorias.

A lo largo de esta memoria, se irán desarrollando los conceptos físicos y matemáticos enfocados en la teoría musical, que servirán para el entendimiento de la implementación de este sistema.

En este Trabajo Fin de Grado, el desarrollo de esta herramienta, su posterior evaluación, así como los resultados obtenidos de éstos se realizarán en el entorno Matlab.

1.2 Introducción

La música, es una forma de expresión cultural, a través de la cual los humanos se expresan y representan diferentes pensamientos, sensaciones e ideas que identifican la cultura y tradiciones que les rodean; por lo tanto, con la música se logra transmitir diferentes sensaciones como miedos, alegrías, penas, profundos sentimientos; etc. Es por eso que la música recobra vital importancia en los humanos si se tiene en cuenta que, a lo largo de la historia, todas las civilizaciones han creado música; de hecho, hasta los instrumentos musicales están entre los objetos más antiguos creados por el ser humano, ejemplo de ello es que según **Steven Mithen** profesor de arqueología de la Universidad de Reading del condado de Berkshire, en Inglaterra, Reino Unido, en su libro “ La prehistoria de la mente: Los orígenes cognitivos del arte, la religión y la ciencia” traducido del inglés [1], sostiene que la flauta más antigua reconocida por la comunidad científica hasta mayo del 2012 es de 36 mil años descubierta en el yacimiento de Geissenklösterle, Alemania. Es por eso que esta retrospectiva histórica nos muestra la importancia que recobra la música para los hombres.

El **Doctor Daniel Joseph Levitin**, neuro científico especializado en la música, director del Laboratorio de Música, Percepción, Cognición y Experiencia en la Universidad de McGill, decano fundador de Artes y Humanidades de la Escuela de Minerva y un miembro distinguido de la Facultad de la Escuela de Negocios Hass de la Universidad de California en Berkeley quien en su libro “Este es tu cerebro en la música” traducido del inglés[2], plantea que esta disciplina afecta a muchas partes del órgano vital; es por eso que escuchar una música alegre puede ayudar a levantar los ánimos; mientras que una triste provoca el efecto contrario. El mismo sostiene que el cerebro interpreta la música en tres áreas distintas:

- 1) El ritmo que es interpretado en la corteza frontal izquierda, la corteza parietal izquierda y el cerebro derecho.
- 2) El tono se procesa en la corteza pre frontal, el cerebro y el lóbulo temporal.
- 3) Por último, la letra, que es descifrada por el área de Wernicke, el área de Broca, la corteza motora, la corteza visual y las zonas correspondientes a las respuestas emocionales.

Por tanto, el doctor divide las áreas perceptivas de la música en tres a efectos de explicarnos que el nivel de percepción e interpretación de la música está relacionado con el cerebro basado en parámetros fundamentales como la melodía, armonía y ritmo; lo que nos lleva a pensar que la música es una fusión coherente de sonidos y silencios, la cual es capturada y procesada mediante nuestro sistema auditivo. Mientras esté sonando una música el ser humano es capaz de capturarla (por el oído), procesarla, analizarla y quedarse con la información de su interés (la nota tocada, el instrumento usado; etc.).

A partir de la premisa de que todo este análisis es realizado por el sistema auditivo, el cual puede tener sus errores en la percepción del sonido debido a factores como enfermedades que afectan a la audición, etc. Y teniendo en cuenta la evolución tecnológica del ser humano, nos vemos abocados a presentar un proyecto más confortable que sea capaz de solucionar estos problemas, basado en el procesado y análisis de señales sonoras en el mundo de la música.

El foco principal de este Trabajo Fin de Grado (de ahora en adelante TFG) está en lograr una herramienta que funcione inteligente y automáticamente para así prescindir del mecanismo a veces erróneo del sistema auditivo humano; siendo indispensables para lograr dicho objetivo los conocimientos adquiridos a lo largo de la carrera en asignaturas como teoría de la comunicación, técnicas de codificación y transmisión, procesado digital de la señal, fundamentos de ingeniería acústica, transmisión digital y el uso del entorno Matlab.

La Figura 1 representa el resumen general del sistema que se ha seguido para conseguir este propósito. El sistema consta de un primer bloque que es el de grabación donde se captarán las señales analógicas tocadas por la flauta, y transformadas a digitales; mientras que en el segundo bloque se enventana la señal grabada para trabajar con segmentos cortos. En el tercer bloque mediante el algoritmo de estimación de pitch (Algoritmo de Yin, que se estudiará más adelante) se calcula la frecuencia fundamental de cada nota tocada sin tener en cuenta los silencios, para eso se debe conocer el inicio (onset) y el final (offset) de cada nota; mientras que el último bloque lo que hace es convertir esta frecuencia a notación musical MIDI.

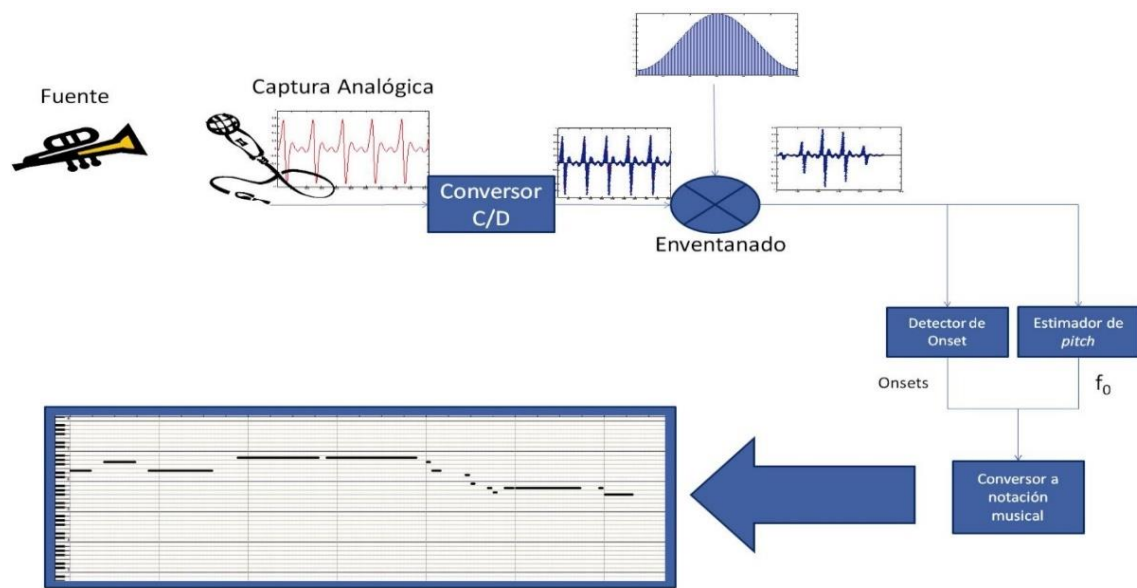


Figura 1: Esquema general del Trabajo Fin de Grado [3]

1.3 Estructuración de la memoria

Este TFG, está estructurado en varios capítulos que están recogidos en el índice y que a continuación se va a describir:

Capítulo 1: Este capítulo está formado por cuatro puntos. En el punto 1.4 se describen todos los conceptos físicos y atributos relacionados al sonido, que ayudaran a comprender mejor el desarrollo y las pautas llevadas a cabo en este Trabajo Fin de Grado.

En el punto 1.5, se introduce la teoría musical necesaria para que el usuario pueda entender el origen y desarrollo del entorno musical; así como dotarle de una cierta preparación respecto al lenguaje empleado en la música.

Capítulo 2: En este capítulo se detalla el estado del arte, con el fin de conocer los avances de la ciencia respecto a la creación de algoritmos utilizados para la estimación de los pitches tocados por distintos instrumentos musicales.

Capítulo 3: En este capítulo se marcan los objetivos que se persiguen en este Trabajo Fin de Grado, de tal forma que se sea capaz de desarrollarlos y conseguir resultados satisfactorios.

Capítulo 4: En este capítulo se detalla la implementación en el entorno Matlab de este sistema, explicando los pasos dados para la consecución de dicho objetivo. Por tanto también se explicarán detalladamente los tres bloques diferentes que forma este sistema como son la grabación, la estimación de pitch (teniendo en cuenta el inventariado y la detección de onset y offset); y la transformación a midi. Estos tres bloques están implementados de forma aislada para después complementarse y formar el conjunto del sistema.

Capítulo 5: En este capítulo se analizarán los resultados obtenidos por el sistema general; teniendo como pautas principales el sometimiento al estudio del cómo reacciona el sistema ante distintos valores de entrada de la relación señal a ruido (SNR), y el comportamiento del sistema respecto al número de tramas detectadas como sonoras o como silencio.

Capítulo 6: Tras haber evaluado el comportamiento que adopta el algoritmo mediante varios parámetros, en este capítulo se sacan las conclusiones más acertadas y aconsejables a la hora de perseguir el objetivo que se ha marcado.

Capítulo 7: Un ejemplo claro del curso de la vida y sobre todo del mundo de la tecnología es que nada es definitivo y todo está en constante evolución, por tanto, en este capítulo se explora diferentes vías o líneas futuras para seguir mejorando este Trabajo Fin de Grado, incluyendo mejoras en este ámbito.

Capítulo 8: Este capítulo lo que pretende es dejar al usuario varios anexos que puedan facilitarle el pleno entendimiento de este trabajo. En esta misma línea, se pone a disposición del usuario un manual sobre el funcionamiento de la interfaz gráfica para su correcta utilidad.

Capítulo 9: Contiene la bibliografía consultada para la realización de este trabajo.

1.4 Características físicas del sonido

En física, el sonido es una vibración propagada a través de un medio elástico, sólido o gaseoso; generalmente aire, con un rango de frecuencias entre 20 y 2000 Hz [4]. Teniendo en cuenta esta definición se puede considerar el sonido como una onda propagándose por el aire, que puede ser digitalizado, medido y analizado.

En un enfoque más perceptual y subjetivo, en el diccionario esencial de la lengua española, la RAE (Real Academia Española), se describe el sonido como la sensación que produce el oído debido a la vibración de los cuerpos al estar en movimiento en un medio elástico como puede ser el aire [5].

A partir de la descripción física y objetiva del sonido como una onda sonora, se puede deducir que la onda propagada por el medio material elástico es de naturaleza senoidal. Las ondas senoidales se caracterizan por tres parámetros: frecuencia, amplitud y fase. Donde la frecuencia se describe como el número de oscilaciones completas en un determinado intervalo de tiempo (segundos). Siendo la amplitud de una onda senoidal la diferencia entre su máximo y mínimo de presión. En una onda senoidal si se tiene en cuenta la respuesta del oído a los cambios de la presión, la amplitud alta (el máximo de la presión) es la que se capta con un volumen alto mientras que la amplitud baja (el mínimo de la presión) es la que se capta con un volumen bajo o suave. Y la fase indica la localización temporal de la onda senoidal.

Un ejemplo ilustrativo de onda senoidal es la que se representa en la Figura 2, donde se sitúa el umbral en un valor nominal de tal forma que las regiones por encima de este valor son de altas presiones o región de amplitudes altas y las regiones por debajo de este valor son de bajas presiones; siendo considerado el aire como el medio de propagación.

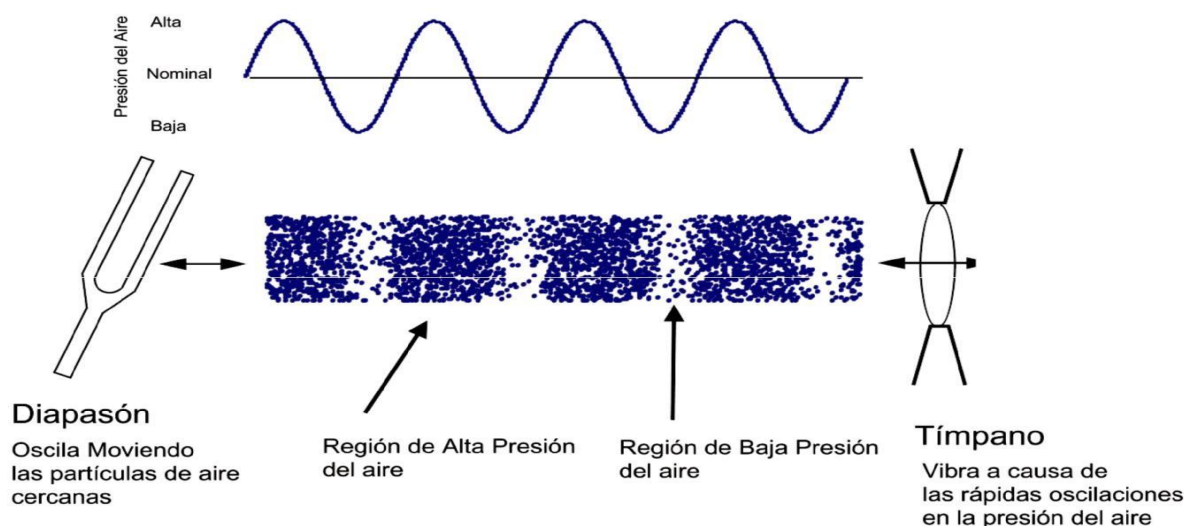


Figura 2: Representación del sonido como una onda de presión. Fuente [6]

La información sobre los parámetros de una onda senoidal se puede obtener analizando la onda en sus dos dominios (el dominio del tiempo y el dominio de la frecuencia). Por ejemplo, si se representa una onda senoidal de 50 Hz (Hz, es la abreviatura de Hertzios que hace honor al físico alemán Heinrich Rudolph Hertz) en el dominio del tiempo en el intervalo de un segundo, y atendiendo a la descripción anterior de la frecuencia como el número de oscilaciones completas la onda oscilará 50 veces en el intervalo de un segundo tal como se puede ver en la Figura 3.

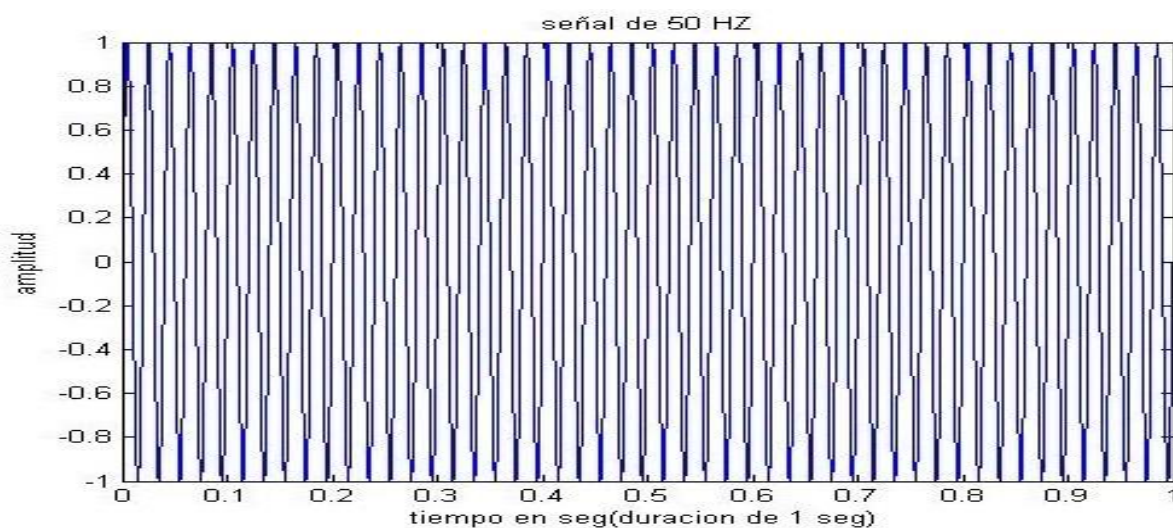


Figura 3: Tono puro de 50 Hz de un segundo de duración

Si representamos solo los primeros 100 milisegundos tendremos 5 ciclos de esta señal en un intervalo de 0.1 lo que implica que en un segundo (que es lo que dura la señal), habrá 50 ciclos (50 Hz), de tal forma que de esta forma también podemos obtener los parámetros de la onda. Tal como se puede observar en la Figura 4.

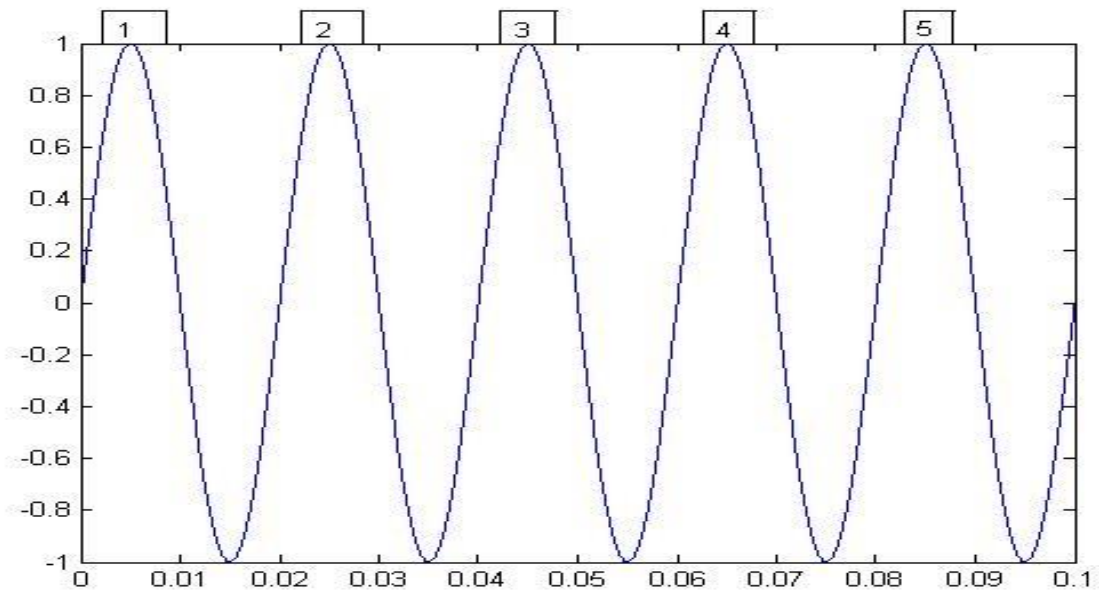


Figura 4: Tono puro de 50 Hz de 0.1 segundos de duración

A partir de este punto nos planteamos el análisis para una señal equivalente a la suma de dos tonos puros. Por ejemplo, en la Figura 5 se representa una señal formada por la suma de un tono de 50 Hertzios y un tono de 100 Hertzios.

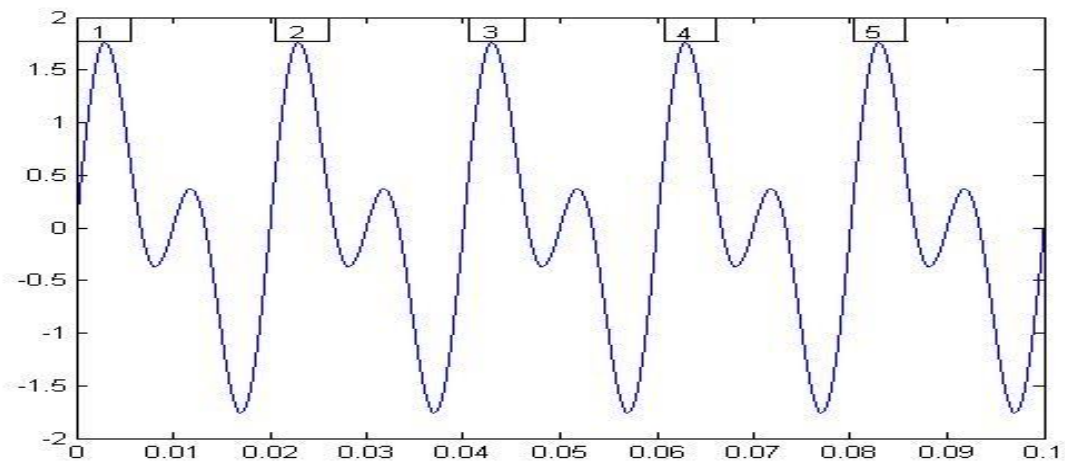


Figura 5: Suma de dos tonos de 50 Hz y 100 Hz

Si nos fijamos en la Figura 5 se observa que la señal ha cambiado en forma respecto a la Figura 4 para los mismos 100 milisegundos, pero sigue teniendo 5 ciclos por tono en 0.1 segundos pese que contiene también un tono de 100 Hz, cuyo número de oscilación natural debería ser de 10 oscilaciones por 0.1 segundos; por lo que se hace difícil y complejo determinar los tonos puros existentes en este periodo de tiempo (0.1 segundos).

El matemático y físico francés **Jean Baptiste Joseph Fourier** mediante sus trabajos sobre descomposición de funciones periódicas en series trigonométricas que llevan su nombre (Series de Fourier) pudo demostrar que una señal periódica cualquiera puede descomponerse (y reconstruirse) en una suma de tonos puros [7] [8]. Por lo que aplicando la Transformada de Fourier a la señal anterior se puede comprobar que existen dos tonos puros, tal como se puede observar en la Figura 6.

Realizar un estudio completo de las señales sonoras abarca estudiar tanto los parámetros físicos objetivos que les caracterizan (amplitud, frecuencia, fase) así como también tener información sobre los parámetros o atributos perceptuales subjetivos. Los atributos subjetivos importantes que caracterizan a las señales se pueden clasificar en: timbre, sonoridad, duración, frecuencia fundamental o pitch [9]. Aunque es importante señalar que la frecuencia fundamental es en realidad un parámetro físico objetivo como anteriormente se ha estudiado, pero sí que está asociado directamente con el pitch como atributo perceptual subjetivo.

Es muy importante e interesante para el lector conocer los atributos perceptuales subjetivos que caracterizan a las señales sonoras con la finalidad de que más adelante pueda entender el desarrollo, la evaluación y la consecución de los resultados que marcan los objetivos de este TFG. Por lo que una vez visto la clasificación de los atributos del sonido que se considerarán para el desarrollo de este TFG, se va a describir los mismos para que el usuario se sitúe a partir de ahora en todo lo que se irá abordando, al ser estos atributos a los que se harán referencia a lo largo de este TFG; sobre todo la frecuencia fundamental o pitch.

1.4.1 Atributos subjetivos del sonido

1.4.1.1 Frecuencia fundamental (f_0)

No se puede entender la frecuencia fundamental sin conocer previamente el concepto de periodo fundamental. Por tanto, supongamos una señal infinita en el tiempo $x(t)$ se dice que es periódica si se repite de forma consecutiva en cada determinado intervalo de tiempo T . Siendo el periodo fundamental T_0 , expresado en segundos, el menor valor de T que cumpla que $x(t) = x(t + T_0), \forall t$.

$$x(t) = x(t + nT_0), n \in (-\infty, +\infty), n \in \mathbb{Z} \quad (1)$$

La ecuación anterior (ver ecuación (1)), demuestra que una señal de duración infinita, si es periódica y por tanto de periodo fundamental T_0 es lógico pensar que tendrá infinitos periodos donde la señal $x(t)$ será periódica, o lo que es lo mismo decir que se repetirá consecutivamente cada T_0 infinitas veces. En este mismo orden de ideas, si se tiene en cuenta que las señales periódicas del día a día (señales de audio, video, imagen; etc) con las que se trabaja en este TFG son de duración finita, causales ($x(t) = 0$), si $(-\infty \leq t \leq 0)$, y reales. Por tanto, en consecuencia, cumplirán con la misma ecuación arriba propuesta y su periodo fundamental será de T_0 , es decir se repetirán por cada determinado intervalo T_0 en un tiempo finito. La frecuencia fundamental f_0 expresada en Hertzios se define como la inversa del periodo fundamental para una señal periódica $x(t)$ (ver ecuación (2)).

$$f_0 \text{ (Hz)} = \frac{1}{T_0 \text{ (s)}} \quad (2)$$

En este TFG, se trabaja siempre con miras al modelado de señales musicales (LTI), donde principalmente debe cumplirse que cualquier señal a la entrada de un sistema lineal invariante en el tiempo (LTI), a su salida se obtenga la misma señal, pero escalonada en amplitud. Y atendiendo al Teorema de Fourier [10] se llega a la determinación de que cualquier señal periódica infinita en el tiempo puede ser descompuesta en suma de infinitas sinusoides ($s_k(t)$) que están relacionados armónicamente entre sí. Estas sinusoides pueden ser descritas como la parte real de una exponencial compleja caracterizada por los parámetros físicos de (amplitud $|a_k|$, frecuencia f_0 y la fase ϕ_k). Las ecuaciones (3, 4, 5, 6, 7, 8, 9, 10, 11) se han referenciado en [11] y vienen descritas de la siguiente forma:

$$s_k(t) = \text{Re}(a_k \cdot e^{jk2\pi f_0 t}) \quad ; \quad (3)$$

Por tanto, la señal periódica de duración infinita $x(t)$ quedará descompuesta como sigue:

$$x(t) = \sum_{k=-\infty}^{\infty} s_k(t) = \sum_{k=-\infty}^{\infty} \text{Re}(a_k \cdot e^{jk2\pi f_0 t}) \quad (4)$$

Si se desarrolla esta expresión suponiendo tres intervalos de la señal en tiempo $[t \in (-\infty, \dots, -1), t = 0, t \in (1, \dots, \infty)]$ y suponiendo ahora $x(t)$ real, $a_0 = 0$ y $a_k = |a_k| \cdot e^{j\phi_k}$, por consiguiente $x(t) = x^*(t) \Rightarrow a_k = a_{-k}^*$; entonces el desarrollo de la expresión es:

$$x(t) = \sum_{k=-\infty}^{\infty} (a_k \cdot e^{jk2\pi f_0 t}) = \sum_{k=-\infty}^{-1} (a_k \cdot e^{jk2\pi f_0 t}) + \sum_{k=1}^{\infty} (a_k \cdot e^{jk2\pi f_0 t}) \quad (5)$$

Por lo que al ser hipótesis de señal real se queda solo con la parte positiva de la señal [11]:

$$= \sum_{k=1}^{\infty} (a_{-k} \cdot e^{-jk2\pi f_0 t}) + a_0 + \sum_{k=1}^{\infty} (a_k \cdot e^{jk2\pi f_0 t}) \quad (6)$$

$$= \sum_{k=1}^{\infty} (a_{-k} \cdot e^{-jk2\pi f_0 t} + a_k \cdot e^{jk2\pi f_0 t}) \quad (7)$$

$$= \sum_{k=1}^{\infty} (a_k^* \cdot e^{-jk2\pi f_0 t} + a_k \cdot e^{jk2\pi f_0 t}) \quad (8)$$

$$\sum_{k=1}^{\infty} 2Re(a_k \cdot e^{jk2\pi f_0 t}) = \sum_{k=1}^{\infty} 2Re(|a_k| \cdot e^{-j\phi_k} \cdot e^{jk2\pi f_0 t}) \quad (10)$$

Poniendo la ecuación en función del coseno resulta:

$$x(t) = \sum_{k=1}^{\infty} (|a_k| \cdot \cos(2\pi f_k t + \phi_k)) \quad (11)$$

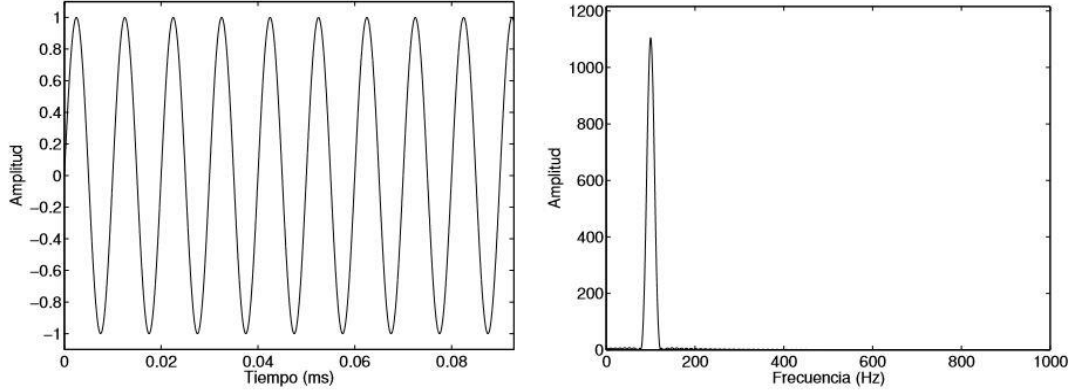
De esta ecuación se deduce que en una señal real (como con la que trabajaremos) los armónicos son múltiplos de la frecuencia fundamental ($f_k = kf_0, k \in \mathbb{N}$) [12]

Por tanto, se ha visto que las señales musicales con las que se trabaja en la vida real son ‘armónicos’. Aunque esta afirmación no termina siendo del todo cierta debido a que las propiedades estadísticas como media (μ_x) y autocorrelación (R_x) de las señales musicales no son constantes en un tiempo extendido, sino que sufren modificaciones o lo que es lo mismo decir que las señales musicales no son estacionarias para extendidas duraciones. Para evitar este problema en telecomunicaciones se trabaja con bloques pequeños de la señal (en orden de veintenas de milisegundos) que generalmente se llaman tramas (frames). Donde las propiedades estadísticas se mantienen invariantes y por tanto pueden ser consideradas estacionarias o cuasi-estacionarias. Es en este sentido que el Teorema de Fourier [10] sostiene que: sea $c(t)$ una trama cuasi-periódica, ésta puede descomponerse a su vez en un número determinado (M) sinusoides que están relacionados entre sí armónicamente:

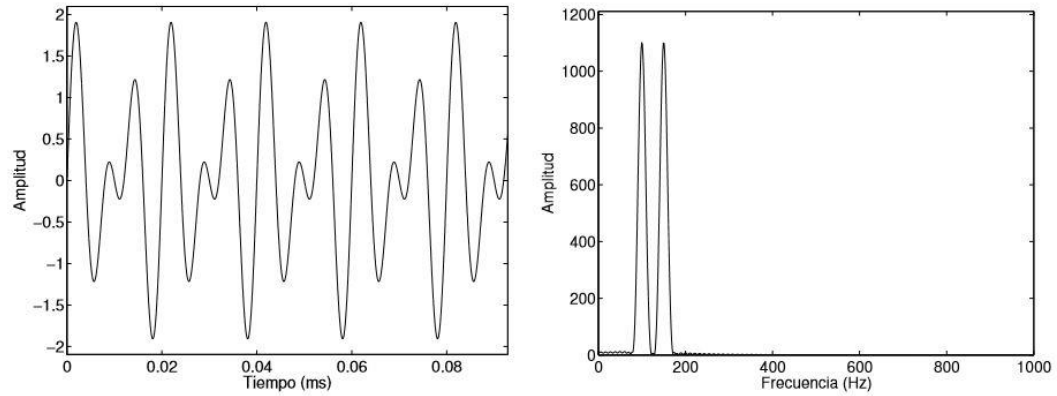
$$c(t) = \sum_{k=1}^{\infty} (A_p \cdot \cos(2\pi f_p t + \phi_k)) \quad (12)$$

La cuasi-periodicidad de las tramas se debe a que no son armónicos puros, sino que sus componentes frecuenciales sufren pequeñas desviaciones respecto a los múltiplos de la frecuencia fundamental ($f_p \approx pf_0, p \in \mathbb{N}$).

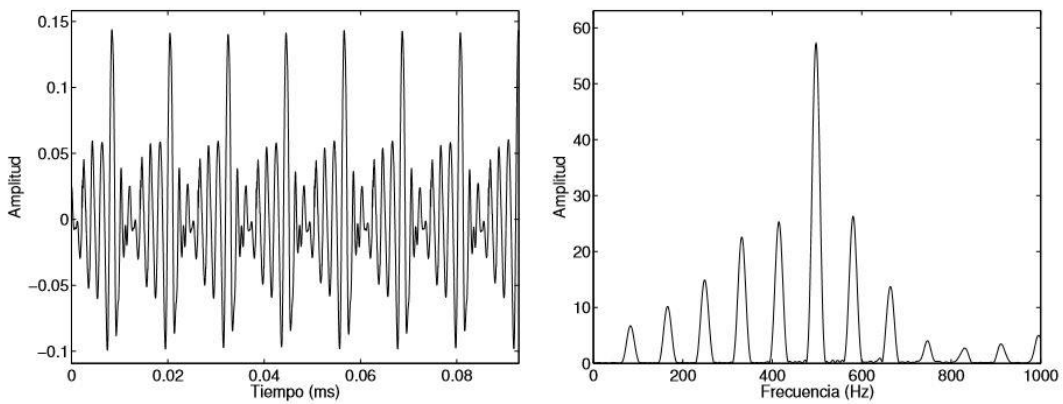
En la Figura 6 se puede ver un ejemplo claro de todo lo explicado anteriormente. Se representa en 6(a) una señal periódica formada por una senoide, otra señal formada por la suma de dos senoideos 6(c), y la representación de una trama cuasi-periódica 6(f) de la nota Sí4 de la flauta dulce donde se puede ver claramente que sus picos están en cada pf_0



(a) Señal s_1 en el dominio del tiempo (b) Señal s_1 en el dominio de la frecuencia



(c) Señal s_2 en el dominio del tiempo (d) Señal s_2 en el dominio de la frecuencia



(e) Señal s_3 en el dominio del tiempo (f) Señal s_3 en el dominio de la frecuencia

Figura 6: Ejemplos de señales periódicas y cuasi-periódicas. Las figuras (a) y (b) muestran una señal periódica S_1 de una senoide con $T_0 = 6.66\text{ms}$ y $f_0 = 150\text{Hz}$ en el dominio del tiempo y la frecuencia respectivamente. Las figuras (c) y (d) muestran una señal periódica S_2 que está compuesta por la suma de dos senoideas. La primera de ellas con $T_0 = 6.66\text{ms}$ y $f_0 = 100\text{Hz}$, y la segunda con $T_0 = 5\text{ms}$ y $f_0 = 200\text{Hz}$. La señal cuasi-periódica S_3 está generada por la nota Sí4 con $T_0 = 2.02\text{ ms}$ y $f_0 = 495\text{Hz}$ de la flauta dulce de la base de datos RWC.

En la Tabla 1 se puede ver el rango de frecuencias posibles que puede generarse por cada instrumento musical.

Familia musical	Instrumento	Rango de frecuencias (Hz)
Vocal	Soprano	250-1000
	Contralto	200-700
	Baritono	110-425
	Bajo	80-350
Viento	Flautín	630-5000
	Flauta	250-2500
	Oboe	250-1500
	Clarinete (Bb)	125-2000
	Clarinete (Eb)	200-2000
	Fagot	55-575
	Trompeta	90-1000
	Saxofón Alto	125-900
	Saxofón Tenor	110-630
	Saxofón Soprano	225-1000
Metal	Trompeta	170-1000
	Trombon Tenor	80-600
	Trombon Bajo	63-400
	Tuba	45-375
Cuerda	Violin	200-3500
	Viola	125-1000
	Chelo	63-630
	Guitarra	80-630
	Piano	28-4100
Organo	Organo	20-7000
Percusión	Celesta	260-3500
	Timbales	90-180
	Carrillón	63-180
	Xilófono	700-3500

Tabla 1: espectro de frecuencias de varios instrumentos musicales. Fuente [11]

1.4.1.2 Pitch

El pitch es el atributo perceptual que permite ordenar los sonidos en una escala musical de frecuencia logarítmica. Es tomado como la nota que se asocia a un determinado evento musical.

Existen varias definiciones de pitch pero la que más convence es según [13] que mantiene que los sonidos tienen pitch si son emitidas a una frecuencia tal que esté dentro del rango audible. De esta forma se puede ver la relación que guarda el pitch con la frecuencia como parámetro físico para señales periódicas o cuasi-periódicas [14] [15].

Otras definiciones que se pueden encontrar sobre el pitch como concepto se tienen por ejemplo en el American National Standards Institute (ANSI) donde se sostiene que el pitch es un atributo auditivo que ordena los sonidos en escala de menor a mayor, y depende principalmente de la estimulación sonora, la presión de las ondas sonoras y la forma de onda del propio estímulo [16]. Mientras que a nivel de la psicoacústica se define el pitch en base a la voz hablada, siendo tal que es la vibración de las cuerdas vocales [17]

En este TFG será muy recurrente utilizar de forma similar la frecuencia y el pitch pese a la sutil diferencia entre ambas definiciones.

1.4.1.3 Sonoridad, duración y timbre

La sonoridad como atributo perceptivo, permite la ordenación de los sonidos en una escala que abarca desde los sonidos más bajos hasta los sonidos más altos en intensidad. Depende de factores como la potencia del sonido, la frecuencia y la duración del mismo.

Para establecer la sonoridad en frecuencia se establecen unas curvas, llamadas curvas isofónicas (ver Figura 8) de igual sonoridad con la referencia en 1KHz. Estas curvas son variantes con la frecuencia, por ello dos sonidos con igual potencia, pero distintas frecuencias no tendrán la misma sonoridad, puesto que es un atributo perceptual y no físico. La unidad de medida de sonoridad es el fonio o fon. La sonoridad está estrechamente relacionada con la dinámica musical, la cual se refiere al volumen o al nivel de intensidad sonora de un sonido emitido. Calidad con la que se puede distinguir un sonido suave de uno fuerte. Esta relación se puede ver en la tabla 2

Sonoridad de interpretación	Notation	Indicaciones de dinámica
Muy suave	<i>pp</i>	pianissimo
Suave	<i>p</i>	piano
Medio suave	<i>mp</i>	mezzopiano
Medio fuerte	<i>mf</i>	mezzoforte
Fuerte	<i>f</i>	forte
Muy fuerte	<i>ff</i>	fortissimo
Forzando	<i>sfz</i>	sforzando
Incremento gradual	<	crescendo
Decremento gradual	>	decrescendo

Tabla 2: Relación sonoridad y dinámica. Fuente [11]

La duración es el periodo de tiempo en el que una nota está activa en la señal de audio. Se relaciona la duración con la terminología de onset y offset, ya que éstos indican el inicio y final del evento o nota. La duración de las notas musicales puede ser representada mediante cuatro valores: redonda, blanca, negra, corchea, semicorchea, fusa y semifusa. La redonda es la representación de mayor duración mientras que la semifusa es la de menor duración. En la tabla 3 se puede observar la relación entre las distintas formas de representar la duración de las notas musicales.

VALOR 4	VALOR 2	VALOR 1	VALOR $\frac{1}{2}$	
REDONDA	BLANCA BLANCA	NEGRA NEGRA NEGRA NEGRA	CORCHEA CORCHEA CORCHEA CORCHEA CORCHEA CORCHEA CORCHEA	CADA CORCHEA EQUIVALE A DOS SEMICORCHEAS...

Tabla 3: Duración de notas musicales. Fuente [18]

Como se puede observar en la tabla 3 una redonda es igual a dos blancas, una blanca a dos negras y así sucesivamente hasta el último.

En la Figura 7, se puede observar también el tiempo de duración de las notas, donde las negras por ejemplo deben durar un segundo, las blancas dos segundos, la corchea medio segundo y así sucesivamente



Figura 7: Representación de la duración de una nota musical. Fuente [18].

El timbre es un atributo que relaciona la calidad sonora del sonido de un instrumento [19]; gracias al cual se puede distinguir y localizar fuentes sonoras [20]. Además, que con el timbre se puede distinguir parámetros como la sonoridad, duración y el pitch [21]

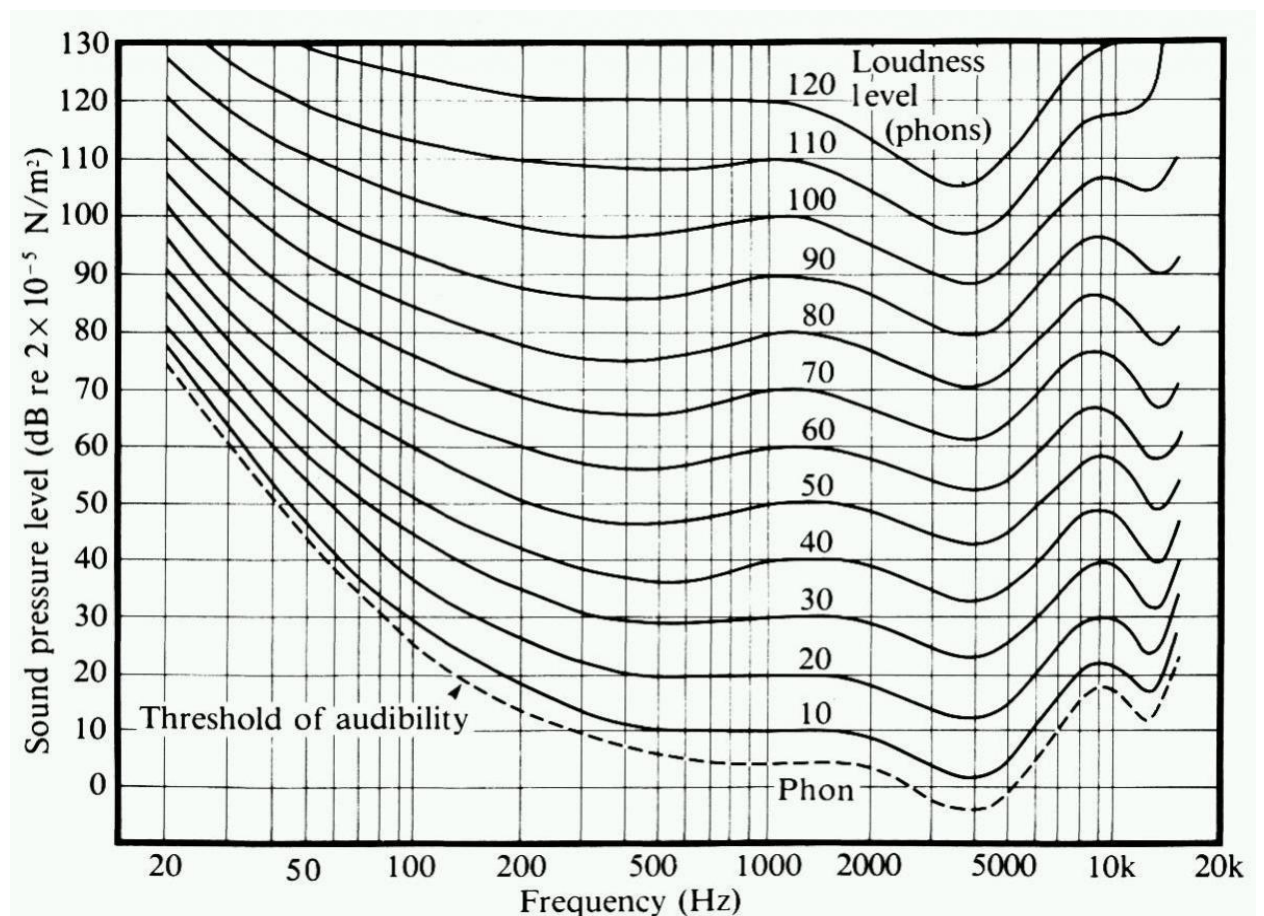


Figura 8: Curvas isofónicas. Fuente [22]

1.5 Teoría musical

Cualquier composición musical puede ser representada en una partitura (ver figura 9). La partitura está formada por una notación simbólica. En ella también se incluye información como la velocidad de interpretación, intensidad y matices de interpretación a fin que quede descrita de forma completa.



Figura 9: Notación simbólica de una partitura. Fuente [23]

1.5.1 La música como concepto

Existen muchas definiciones de la música ya que es interpretada de forma distinta por las distintas culturas, lo que hace que este concepto desencadene diferentes interpretaciones a la hora de especificar claramente su significado.

Tradicionalmente se definía la música como el arte capaz de combinar de forma coherente y sensible los sonidos y silencios regidos por la armonía y el ritmo. [24].

Otra definición de la música se puede encontrar en la Real Academia Española de la lengua que la define como el conjunto de sonidos en los que se varían sus cualidades con fines armónicos para recrear el oído [25]

Las definiciones anteriormente descritas de la música responden a la percepción de cada individuo. Por lo que una descripción propiamente de la música sería la que se considera a la música como un conjunto de eventos o notas que llevan un determinado ritmo y una sucesión musical establecida [26]. Y siendo la combinación de estas notas el acorde.

1.5.2 La escala musical

Al conjunto ordenado de sonidos musicales o notas se llama escala musical. Existen varios tipos de escalas musicales atendiendo cada uno al orden específico de un grupo de sonidos en los que sus alturas guardan una relación entre sí dependiendo del modelo de afinación; es decir la escala musical divide el rango de frecuencias audibles. Aunque está demostrado que el ser humano no hace uso de todas las escalas musicales debido a que el cerebro tiene una tendencia a resumir el espectro audible. De hecho, en la naturaleza existen sonidos que son captados por nosotros como los mismos, cuando en realidad son emitidos a frecuencias diferentes.

La escala musical también puede ser vista como un parámetro didáctico y de clasificación mediante el cual la interpretación y análisis de las señales musicales se hace más sencillo [27].

Para que dos notas suenen como una única deben tener una relación de pitch igual a la potencia. Por tanto, las notas que guardan esta relación de pitch vienen agrupadas en una misma clase y se les da el mismo nombre. Las clases de pitch son representadas según región tal como viene recogido en la tabla 4.

Región	1ª	2ª	3ª	4ª	5ª	6ª	7ª
España	Do	Re	Mi	Fa	Sol	La	Si
Italia	Do	Re	Mi	Fa	Sol	La	Si
Francia	Do	Re	Mi	Fa	Sol	La	Si
Países anglófonos	C	D	E	F	G	A	B
Norte de Europa	C	D	E	F	G	A	H

Tabla 4: Notas musicales o clases de pitch por región

Si estas notas se agrupan en un intervalo de ocho notas se denomina a este intervalo como octava musical. Para diferenciar las notas con la misma clase de pitch pero que se sitúan en octavas distintas se combina el nombre de la nota con un número que designa la octava específica. Por ejemplo, el pitch de la nota Sí₄ es de 493.9 Hz o el de Sí₃ es 296.9 Hz.

El pitch que posee exactamente cada nota se define relacionado con el pitch de un sonido estándar, consensuado por los organismos de estandarización. Aunque han existido, a lo

largo del tiempo, otros sonidos de referencia. En 1936 la asociación de Estándares Americana estableció tal sonido de referencia como el correspondiente a la nota La4, de frecuencia fundamental 440 Hz. Más tarde se creó un estándar internacional sobre la nota La4 y los 440Hz [28].

La fórmula que calcula el pitch de una nota viene descrita de la siguiente forma:

$$f = 2^{\frac{n}{12}} \times 440 \text{ Hz} \quad (13)$$

donde n es la distancia, en semitonos de la nota deseada a La4. Desarrollando la ecuación (13) se obtiene la siguiente tabla (ver tabla 5) donde en cada celda se representa la frecuencia en Hz y entre paréntesis se encuentra la diferencia en semitonos respecto a La4 (n).

Notas musicales (Hz - MIDI)									
Nota / Octava	0	1	2	3	4	5	6	7	8
C	16.35 (12)	32.70 (24)	65.41 (36)	130.8 (48)	261.6 (60)	523.3 (72)	1047.0 (84)	2093.0 (96)	4186.0 (108)
C#	17.32 (13)	34.65 (25)	69.30 (37)	138.6 (49)	277.2 (61)	554.4 (73)	1109.0 (85)	2217.0 (97)	4435.0 (109)
D	18.35 (14)	36.71 (26)	73.42 (38)	146.8 (50)	293.7 (62)	587.3 (74)	1175.0 (86)	2349.0 (98)	4699.0 (110)
D#	19.45 (15)	38.89 (27)	77.78 (39)	155.6 (51)	311.1 (63)	622.3 (75)	1245.0 (87)	2489.0 (99)	4978.0 (111)
E	20.60 (16)	41.20 (28)	82.41 (40)	164.8 (52)	329.6 (64)	659.3 (76)	1319.0 (88)	2637.0 (100)	5274.0 (112)
F	21.83 (17)	43.65 (29)	87.31 (41)	174.6 (53)	349.2 (65)	698.5 (77)	1397.0 (89)	2794.0 (101)	5588.0 (113)
F#	23.12 (18)	46.25 (30)	92.50 (42)	185.0 (54)	370.0 (66)	740.0 (78)	1480.0 (90)	2960.0 (102)	5920.0 (114)
G	24.50 (19)	49.00 (31)	98.00 (43)	196.0 (55)	392.0 (67)	784.0 (79)	1568.0 (91)	3136.0 (103)	6272.0 (115)
G#	25.96 (20)	51.91 (32)	103.80 (44)	207.7 (56)	415.3 (68)	830.6 (80)	1661.0 (92)	3322.0 (104)	6645.0 (116)
A	27.50 (21)	55.00 (33)	110.0 (45)	220.0 (57)	440.0 (69)	880.0 (81)	1760.0 (93)	3520.0 (105)	7040.0 (117)
A#	29.14 (22)	58.27 (34)	116.50 (46)	233.1 (58)	466.2 (70)	932.3 (82)	1865.0 (94)	3729.0 (106)	7459.0 (118)
B	30.87 (23)	61.74 (35)	123.50 (47)	246.9 (59)	493.9 (71)	987.8 (83)	1976.0 (95)	3951.0 (107)	7902.0 (119)

Tabla 5: Relación frecuencia MIDI de las notas. Fuente [11]

Con esta tabla se puede deducir que una octava la componen 12 notas, es decir las siete notas básicas más 5 alteraciones que, en este caso, aumentan el pitch de la nota modificada en una razón de $\sqrt[12]{2}$. La escala es dividida en semitonos, que es la mínima separación entre notas.

1.5.3 Clasificación de los sonidos

Los sonidos pueden clasificarse atendiendo al número de canales, el grado de periodicidad, el número de timbres y el número de eventos.

1.5.3.1 Sonidos armónicos

Se dice que un sonido es armónico cuando las sinusoides de las que se compone están relacionadas entre sí [29]; es decir que sus componentes están separados por cada múltiplo de f_0 [30].

Pero los sonidos musicales siempre están ligados al concepto de inarmonía o la inarmonicidad. El cual genera una desviación de los armónicos de un sonido. Un ejemplo de inarmonía lo tenemos cuando en la nota fundamental La 440 su segundo armónico se produce en 880.4 Hz en vez de en 880 Hz; esta leve variación es el efecto de inarmonicidad.

1.5.3.2 Sonido monoaural, estéreo y multicanal

Los sonidos monoaurales provienen de un solo canal o han sido grabados por un solo micrófono y generan una señal duplicada de tal forma que oímos lo mismo tanto en el oído izquierdo como en el derecho. El análisis de los sonidos monofónicos es complejo debido a que no tienen información espacial de las fuentes [31].

Por lo contrario, un sonido estéreo es una señal que se graba con dos canales o micrófonos iguales de tal forma que cuando oímos lo grabado se escucha un determinado sonido por el oído izquierdo diferente al escuchado por el oído derecho. Los sonidos estereofónicos tienen información espacial sobre la localización de fuentes.

Los sonidos multicanal, son aquellos generados a partir de cuatro canales de audio separados.

1.5.3.3 Sonidos monofónicos y polifónicos

Se dice que un sonido es monofónico cuando es emitido por un solo instrumento en cada instante de tiempo; es decir aquellos sonidos en los cuales aparece solo una determinada frecuencia. Mientras que los sonidos polifónicos son los generados por varios instrumentos [32].

1.5.3.4 Sonidos monotímbricos y multitímbricos

En base al número de instrumentos, se puede considerar una interpretación musical como mono tímbrica o multi-tímbrica. Si las piezas musicales son interpretadas por un único instrumento se dice que el sonido es monotímbrico; mientras que si son varios es politímbrico.

1.5.3.5 Conceptos musicales previos al análisis de la estimación de pitch

En este punto se va a ver algunos conceptos que serán de utilidad para comprender mejor este TFG.

Antes del procesamiento de la señal de entrada, se debe configurar varios parámetros para el correcto funcionamiento del algoritmo. Para esto concretamente se realizará un previo estudio de la señal de entrada. Estos parámetros son los siguientes:

- **Frecuencia de muestreo**

En este TFG, la fuente que generará la señal de audio es la “flauta dulce” siendo ésta captada por el micrófono de un ordenador. La señal generada por la flauta es analógica por tanto deberá ser convertida a digital mediante un conversor analógico/digital (A/D) que ya viene incorporado en la tarjeta de sonido del ordenador. A partir de este punto se encuentra con el primer factor, la calidad de la señal resultante, que depende de la frecuencia de muestreo; por lo que se tendrá que configurar la frecuencia de muestreo.

La frecuencia de muestreo viene definida como el número de muestras por unidad de tiempo que es tomada de una señal continua para generar una señal discreta, durante el proceso de convertirla de analógica a digital.

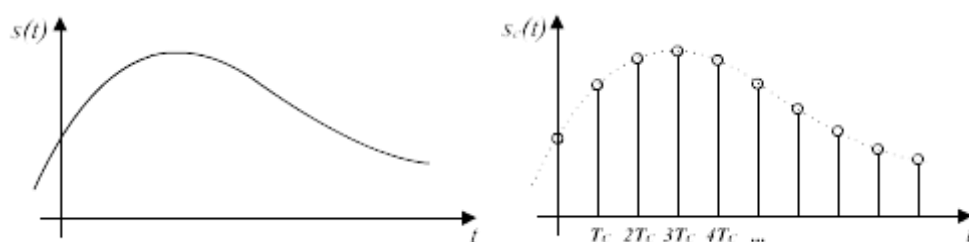


Figura 10: A la izquierda señal continua y a la derecha señal muestreada. Fuente [33]

Como la finalidad es obtener una representación discreta similar o que se adecue a la forma de onda de la señal continua; para ello es importante cumplir con el Teorema de Nyquist-Shannon [34]. Según este teorema, para poder replicar con exactitud la forma de

onda de una señal continua, la frecuencia de muestreo (f_s) debe ser igual o superior al doble de la frecuencia máxima f_{max} a muestrear:

$$f_s \geq 2f_{max} \quad (14)$$

Cuanto mayor es la cantidad de muestras, mejor será la conversión A/D, lo que implica una mayor calidad de la señal resultante. Teniendo en cuenta el Teorema de Nyquist (ver ecuación 14) se indica que para digitalizar una señal continua ésta debe ser limitada en banda. La frecuencia de muestreo indica la máxima componente frecuencial que se registra a la hora de realizar el proceso de digitalización; por tanto por ejemplo si se quiere digitalizar una señal de audio con una frecuencia máxima de A Hz, entonces se debe usar una frecuencia de muestreo mínima de 2A Hz. Si no se cumple esta condición tal como explica el Teorema de Nyquist se produce un solapamiento espectral, lo que implica que en el dominio del tiempo las frecuencias superiores a $\frac{f_s}{2}$ harían que las muestras no coincidan a la hora de reconstruir la señal original, teniendo como resultado una señal que suene distinta a la original limitada en $\frac{f_s}{2}$. Por consiguiente en muchas de las aplicaciones de audio se suele usar frecuencias de muestreo iguales a 44100 Hz ya que con $f_s = 44100 \text{ Hz}$ no existe la posibilidad de que hayan componentes espectrales por encima de $\frac{f_s}{2} = 22050 \text{ Hz}$ que es superior que cualquier componente del espectro musical (ver tabla 5).

La configuración de la frecuencia de muestreo debe restringirse en función de los resultados que se desea alcanzar. Para ello, en este TFG, se centrará exclusivamente en el estudio del rango de frecuencias que es producido por la flauta, con el objetivo de conocer la mínima frecuencia de muestreo a la que se puede utilizar para la captura o grabación de todas las frecuencias audibles producidas por la flauta (ver tabla1). Como recordatorio para el lector, en la Figura 11, se puede ver el espectro audible de la flauta.



Figura 11: Rango de frecuencias de la flauta

En la Figura 11, se puede observar que la frecuencia fundamental máxima del espectro de la flauta dulce es de 2349 Hz, y teniendo en cuenta que si se toca la nota con la flauta dulce se genera una onda sonora la cual viene acompañada por una serie de armónicos y cada uno con amplitudes distintas por lo que si se hace uso del Teorema de Nyquist [34] considerando que los armónicos son múltiplos de la frecuencia fundamental según se deduce del Teorema de Fourier [10] ($f_k = kf_0, k \in \mathbb{N}$) ; entonces la frecuencia de muestreo mínima (f_{s_min}) que se puede utilizar para nuestro sistema debe ser de:

$$f_{s_min} \geq 2xf_k \geq 2xkf_0 \geq 2x2349xk \geq 4698xk \text{ Hz} \quad (15)$$

También es importante resaltar la dependencia directa que tiene la frecuencia de muestreo sobre el bloque de muestras o tramas, si bien se considera que el bloque de muestras será la principal unidad del procesamiento de nuestra señal y dependerá de la frecuencia de muestreo que se utilice, por lo que, para la configuración de la frecuencia de muestreo, se debe valorar la repercusión que tiene la misma sobre el bloque de muestras. Un ejemplo claro sobre la influencia de la frecuencia de muestreo respecto al bloque de muestras es que una frecuencia de muestreo grande provoca que el tamaño de la ventana de la señal tenga un intervalo de tiempo más corto (que es lo que se persigue si se tiene en cuenta el Teorema de Fourier, donde se puede deducir que las señales de audio tienen carácter periódico para intervalos de tiempo cortos). Lo que significa que para ventanas de la señal de igual número de muestras si se utiliza una frecuencia de muestreo grande esto genera que estas muestras estén representadas en un intervalo de tiempo más corto que al utilizar frecuencias de muestreo inferiores. En la Figura 1 y Figura 13 se puede observar cómo para el mismo bloque de muestras (1024 muestras) de una señal, utilizar frecuencias de muestreo grandes, por ejemplo en orden de 44100 Hz, el bloque es representado en un intervalo de tiempo más pequeño; mientras que para frecuencias de muestreo inferiores ,4698 Hz, es representado en intervalos más grandes.

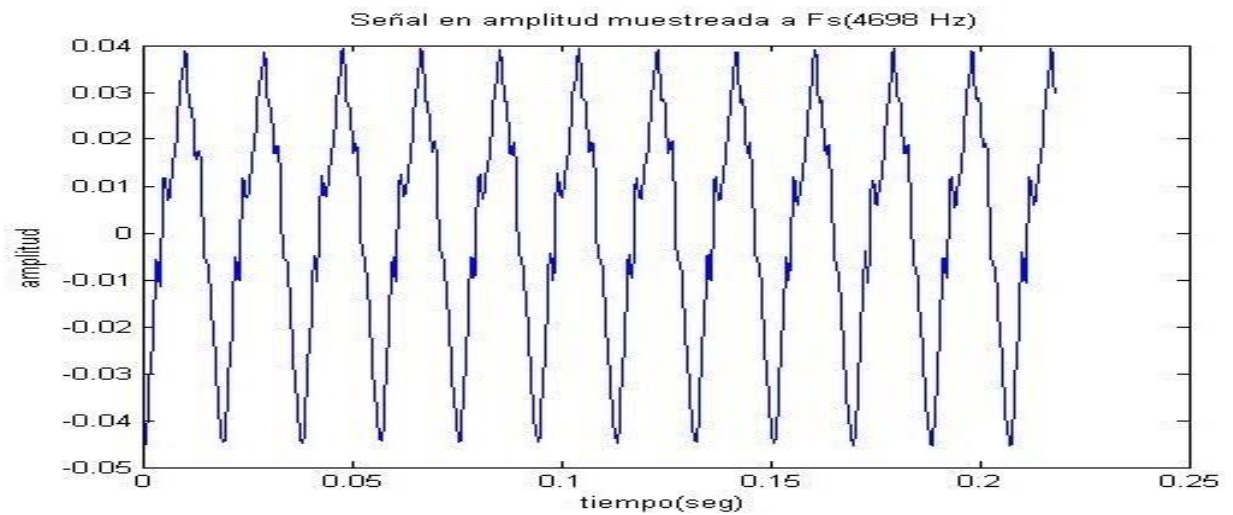


Figura 12: Señal representada en tiempo (de duración 220ms) y muestreada a una frecuencia de muestreo (f_s) igual a 4698 Hz.

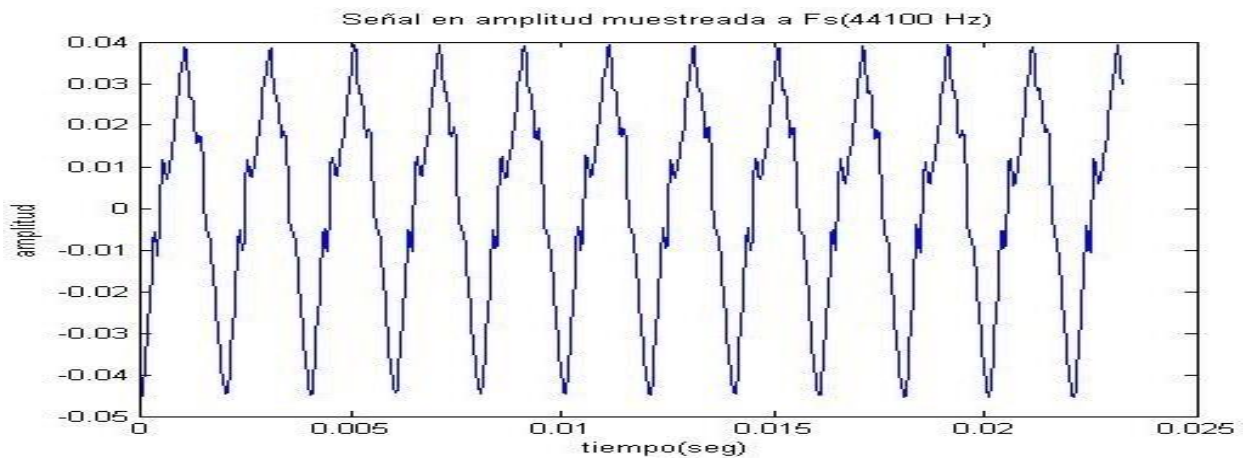


Figura 13: Señal representada en tiempo (de duración 23.22ms) y muestreada a una frecuencia de muestreo igual a 44100 Hz.

De estas Figuras (12 y 13) se puede constatar la diferencia de tiempos de un determinado bloque de muestras o trama en función de la frecuencia de muestreo que se utilice, lo cual viene bien diferenciado en la siguiente Figura 14, que es complementaria.

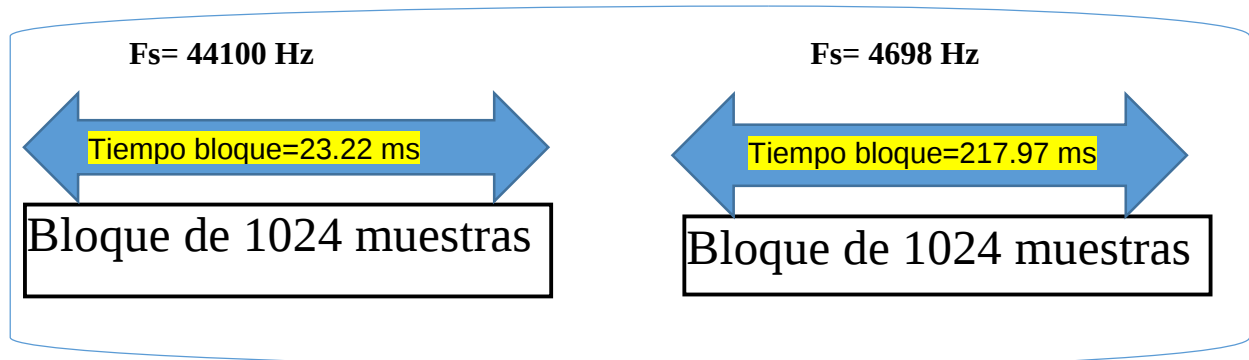


Figura 14: Diferencia de tiempo del bloque o trama

De la Figura 14 anterior, se puede determinar que el número de muestras que forma la trama también es imprescindible para determinar el cálculo del tiempo de la trama. Desde este punto de vista, se puede determinar que, para un tiempo de trama fijo, las frecuencias de muestreo mayores necesitarán un bloque de muestras mayores que para frecuencias de muestreo inferiores.

Se debe tener también en cuenta que, si se tiene un periodo de tiempo de señal muy corto, esta señal es propensa a ser insuficiente para su correcto análisis frecuencial. La frecuencia de muestreo influirá también en la carga del proceso de nuestra herramienta, porque cuanto mayor es el número de muestras de una trama, mayor serán las muestras procesadas.

También es importante para la determinación de este estudio, tener en cuenta la resolución temporal y frecuencial, que serán obtenidas mediante la frecuencia de muestreo, el tamaño en muestras de la trama y el número de puntos de la FFT (Transformada rápida de Fourier).

- **Enventanado**

Como se ha estudiado con anterioridad, las señales musicales no son estacionarias a largo plazo; es decir recobran un carácter pseudo-estacionario sólo a corto plazo. Entonces esto implica que para analizar y procesar una señal musical se debe dividirla en bloques o tramas. Y el mecanismo que nos facilita este análisis en corto plazo se llama enventanado. El enventanado es la multiplicación de una señal por una determinada ventana o trama, con lo que en consecuencia la señal fuera del intervalo es nula. El enventanado se puede expresar como:

$$x(m) = s(n) \cdot w(m - n) \quad (16)$$

Donde $s(n)$ es en nuestro caso la señal musical de flauta de duración larga, $w(m - n)$ la ventana aplicada. Por lo que la trama que resulta de dividir a la señal de larga duración sería $x(n)$ que valdría cero en el intervalo $n \in [m - N + 1, m]$ donde N es la duración de la trama.

La problemática de la aplicación de ventanas cortas a la señal está en la elección del número de muestras que debe tener la ventana, que está en base a la resolución temporal. Si la ventana tiene mayor número de muestras, se provoca una pérdida de resolución temporal, es decir que existe de posibilidad de que una sola ventana contenga información sobre dos notas distintas. Por lo contrario si el número de muestras de la ventana es menor, no habrá suficiente resolución frecuencial a fin de que se pueda distinguir en frecuencia dos tonos que están próximos. Por tanto para reducir las probabilidades de error y dar posible salida a esta problemática se hace uso de las técnicas de solapamiento. Que consiste en analizar ventanas que contengan un tanto por ciento de muestras de la ventana anterior. En la Figura 15 se puede observar ventanas de una señal solapadas al 50%.

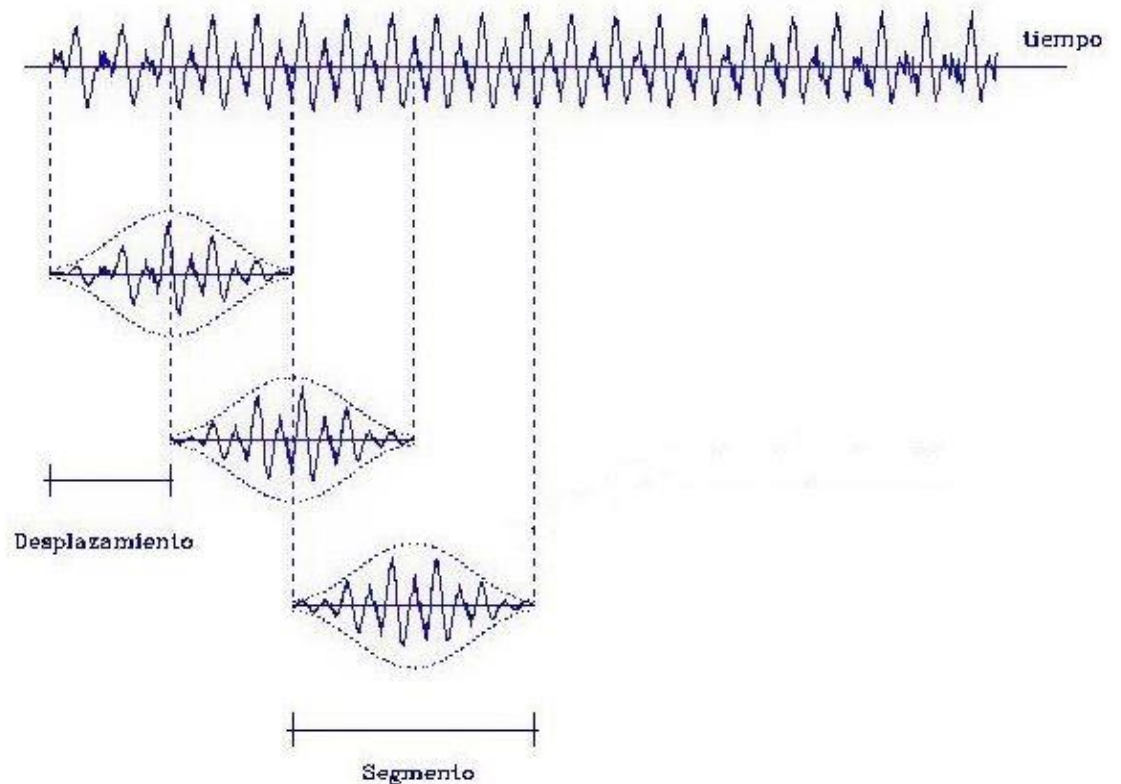


Figura 15: Solapamiento del 50% entre tramas. Fuente [36]

- **Conversor musical midi**

Para transformar un pitch a notación musical MIDI se hace uso de la tabla 5, que relaciona la frecuencia y midi para cada nota.

MIDI (Musical Instrument Digital Interface) es estándar tecnológico descrito por un protocolo que permite a varios instrumentos musicales (electrónicos, tarjetas de sonido, etc.) conectarse y comunicarse entre ellos. El protocolo MIDI ubica o almacena en sí eventos que informan sobre notación musical, velocidad, tono y señales de control para determinados parámetros musicales (dinámica, el pitch, duración, intensidad, volumen, vibrato, el ritmo, instrumento musical, etc.). [36]

- **Onset y Offset**

Onset y Offset en términos musicales hacen referencia a dónde empieza y dónde termina una nota [37]. Esta información es relevante a la hora de segmentar las señales musicales para su posterior análisis y procesado. A este tema se han dedicado números estudios y se ha llegado a la conclusión de que la detección de onset y offset ayuda a la clasificación de las señales musicales; así como a la localización de determinadas notas dentro de una pista o canción [38] [39].

La forma más lógica de detectar los inicios y finales de una señal, es conociendo cómo transita, es decir conociendo su distribución de energía. Porque probablemente los inicios y finales de las notas se sitúen donde se efectúan cambios bruscos de energía en la señal.

1.5.4 Breve historia de la flauta dulce

La flauta dulce es un instrumento de viento de madera. Está formado por un tubo cilíndrico y tiene ocho orificios. De estos, siete están situados en la parte delantera y uno en la parte trasera. La flauta emite sonido cuando es soplada mediante la boquilla que está en su parte frontal provocando un choque entre el aire y el bisel, mientras los dedos van siendo utilizados para tapar y destapar los agujeros, obteniendo de esta manera varias notas musicales.

El origen de la flauta data desde la Edad Media, que fue cuando adquirió gran popularidad, esta tradición continuó hasta la época del Renacimiento y el Barroco. El

Clasicismo y Romanticismo fueron de las épocas en que la flauta cayó en desuso por la aparición de otros instrumentos. Pero a partir del siglo XX volvió a ser parte de los instrumentos utilizados con frecuencia. Ya a partir del Renacimiento es cuando se fabrican las primeras familias nuevas de la flauta dulce como el violín, la viola y el contrabajo. Existen flautas que tienen unos 15 centímetros de longitud, hasta los hay de aproximadamente dos metros y medio. [40]

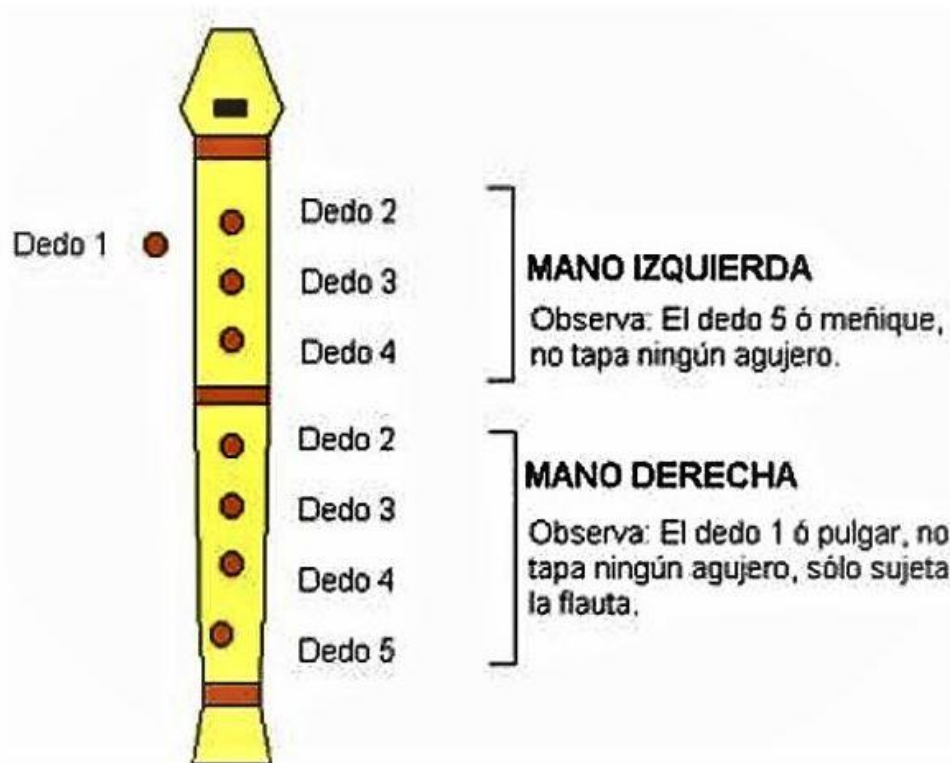


Figura 16: Flauta dulce. Fuente [40]

La flauta dulce está formada por un total de 25 notas [41] siendo la menor de todas Sí4 y la mayor de Sí6. Por tanto, más adelante veremos cómo nuestro sistema está condicionado a la resolución temporal; es decir a la detección de la menor nota del espectro de la flauta (Sí4).

2 CAPÍTULO 2. ESTADO DEL ARTE

Si ya hacia el 1863, **Helmholz** fisiológico y físico alemán en su libro ‘‘**On Sensations of tone**’’, se pregunta ante todo por la naturaleza del sonido. Y a través de estudios experimentales con una sirena polifónica de construcción propia, estableció que cuando la variación de las frecuencias de dos ondas originarias es leve (para un numero de 132 Hz), en general se obtiene disonancia. A través de estas investigaciones llegó a la explicación de la armonía demostrando así la relación entre frecuencia y pitch. Mediante esta demostración desarrolló un microscopio de vibraciones, que puede ser considerado de los primeros estimadores de pitch [42] [43].

No fue hasta el 1877 cuando **Thomas Edison** inventa el fonógrafo un instrumento capaz con su aguja de trazar sobre el estaño un surco continuo helicoidal que contiene toda la información sobre las características del sonido al realizar en él el proceso de grabación [44] [45].

Con el paso de los años en la revista acústica americana [46] se propone un grabador de pitch en el que se pueda amplificar el sonido, controlar de forma automática el volumen, así como medir la frecuencia; lo que a todas luces era una gran apuesta en la estimación de pitch [47]

Años más tarde en 1939, **Homer Dudley**, durante la celebración de la Exposición universal de Nueva York el primer, presentó el vocoder [48], una aplicación capaz de analizar una trama de voz hablada con el fin de estimar parámetros como el pitch de las vocales sonoras, el ruido de las vocales sordas así como la respuesta al impulso del tracto vocal.

En el año 1951 **Licklider** propuso para la estimación de pitch de cualquier sonido el uso de la función de autocorrelación que calcula el periodo fundamental y con él (hallando su inversa) la frecuencia fundamental. [49]

Con el avance de la tecnología a partir de 1960 se conoció un gran despliegue a nivel práctico en el sector de la estimación de pitch siendo determinante en esta época la propuesta alternativa de Noll (Cepstrum) al método de la autocorrelación que era utilizado con anterioridad, donde la idea que también en ámbito temporal era obtener el periodo de la señal

a través del análisis de periódico de la misma comprimida [50]. Este método se conoce como Cepstrum.

Otro algoritmo de ámbito temporal fue propuesto por Ross y Shaffer y tiene relación con la autocorrelación a diferencia de que computacionalmente es menos costosa. Este algoritmo se conoce como función de diferencias [51], que guarda relación con la autocorrelación pero es computacionalmente menos costosa.

Ya hacia los años 70 se multiplicaron los algoritmos de estimación de pitch mediante el análisis del espectro de la señal [52] [53] [54].

La publicación del libro de **W.J. Hess** hacia el año 1983 iba a ser la referencia más adelante por su contenido. Este libro guarda principalmente la realización ordenada del historial de los algoritmos de estimación de pitch hasta la época, así como propuestas sobre la búsqueda de algoritmos eficientes [55]. Y gracias a este libro se puede recoger la clasificación de los algoritmos de estimación de pitch en tres bloques: algoritmos temporales, espectrales y perceptuales.

Todos los algoritmos que han sido propuestos con posterioridad son solo revisiones de los tres bloques que recoge el libro de **W.J. Hess**. Algunos algoritmos que han introducido modificación con posterioridad los podemos encontrar en: [56] [57]

Haciendo utilidad del libro de Hess, vamos a describir distintos algoritmos de estimación de pitch clasificados en tres grandes bloques:

- Algoritmos temporales
- Algoritmos espectrales
- Algoritmos perceptuales

2.1.1 Algoritmos en el dominio del tiempo

2.1.1.1 Short- Term Autocorrelation function (ACF). [58]

La función de autocorrelación que fue propuesta por Licklider consiste en la operación de multiplicar a una señal por su réplica retardada. La operación de autocorrelación nos indica el parecido de una señal consigo misma en un tiempo determinado (ver ecuación 17) [58] [17].

$$r_t(\tau) = \sum_{j=t+1}^{t+W} x_j x_{j+\tau} \quad (17)$$

Donde r_t es la función de autocorrelación que representa la multiplicación muestra por muestra en tiempo entre una señal x_j por su réplica $x_{j+\tau}$ retardada τ muestras. Siendo señal x_j enventanada con una ventana temporal de W muestras de longitud.

La función de autocorrelación utiliza como mecanismo de estimación de pitch la propiedad de que la autocorrelación de una señal periódica es también periódica. Por tanto su parecido es máximo cuando ésta está desplazada en todas sus muestras. De esta forma el periodo fundamental viene marcado por el primer máximo.

Otra forma de expresar la función de autocorrelación viene descrita en [17] de la siguiente forma:

$$r'_t(\tau) = \sum_{j=t+1}^{t+W-\tau} x_j x_{j+\tau} \quad (18)$$

La única diferencia con la ecuación 17, está en que la ventana disminuye a mediación que aumenta el retardo; o lo que es lo mismo decir que la función de autocorrelación tiende a decrecer para grandes retardos [17].

Lo más importante que se debe tener en cuenta sobre la función de autocorrelación a la hora de estimar el pitch es que el periodo fundamental equivale a la muestra de la señal retardada en que se sitúe el primer máximo distinto de cero. También cabe resaltar que entre las dos ecuaciones arriba propuestas (ecuación 17 y 18), la ecuación 18 suele ser la que se emplea más para las distintas investigaciones.

En la Figura 17, se puede observar la función de autocorrelación en tiempo (en segundos) de una trama sonora de 1024 muestras de la nota Sí4 de la flauta dulce.

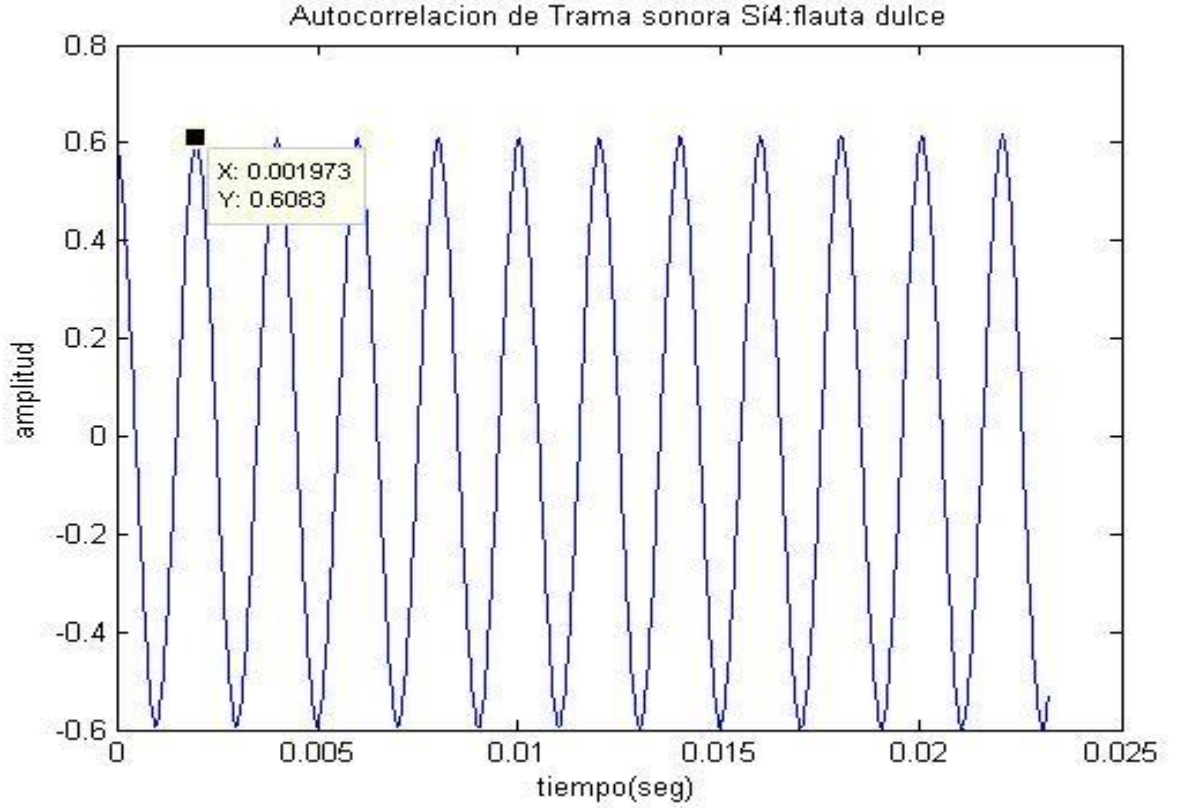


Figura 17: Función de autocorrelación trama 70 de la nota sí4 de la flauta dulce

Donde efectivamente se puede observar que la muestra situada en el primer máximo distinto de cero, marca el periodo fundamental de la señal.

2.1.1.2 Difference function [58]

Este algoritmo es muy similar a la función de autocorrelación si se tiene en cuenta que ambos estiman el periodo fundamental de la señal que se sitúa en la muestra del primer máximo distinto de cero. Pero la diferencia entre ambos consiste en que la función de diferencias en vez de la multiplicación de la señal por su réplica desfasada, utiliza como operación la diferencia entre la señal con su réplica desfasada. Esta función en comparación a la autocorrelación es menos costosa en términos de coste computacional debido a que no incluye multiplicaciones (ver ecuación 19). [51] [58] [17].

$$d_t(\tau) = \sum_{j=t+1}^{t+W} (x_j - x_{j+\tau})^2 \quad (19)$$

Donde d_t es la función de diferencias que representa la diferencia entre una señal x_j y su réplica $x_{j+\tau}$ retardada τ muestras. Siendo esta señal x_j enventanada con una ventana temporal de W muestras de longitud.

La función de diferencias devuelve una señal en forma de valles en vez de picos como la autocorrelación. Por tanto, la muestra del primer valle distinto de cero marcará el periodo fundamental, como se puede observar en la Figura 18.

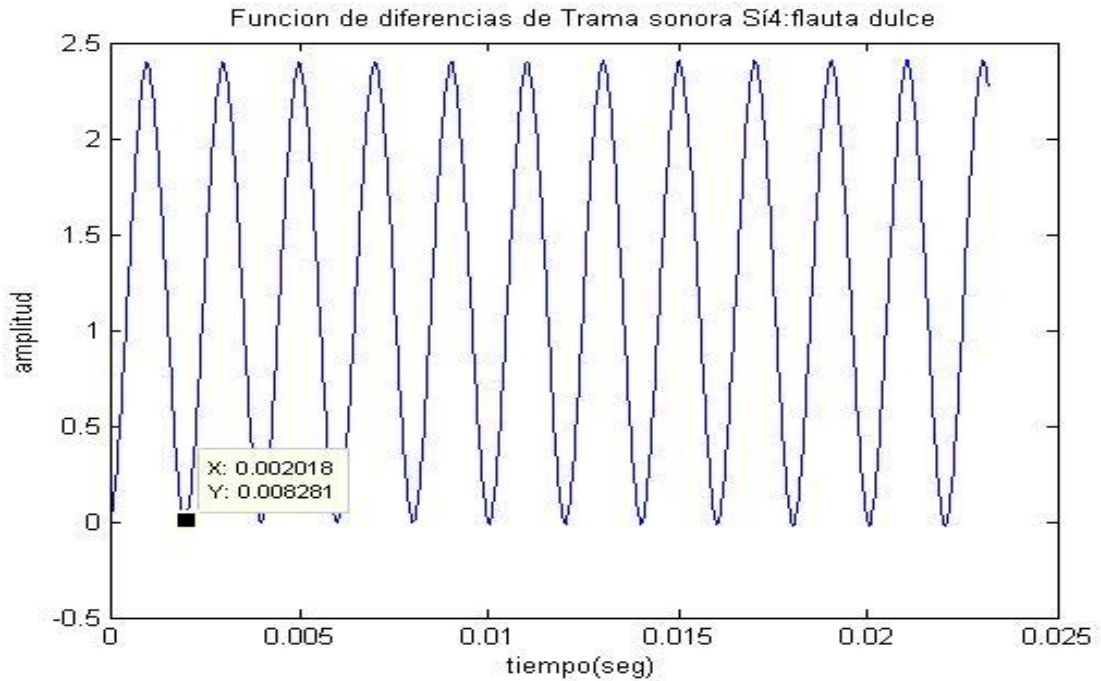


Figura 18: Función de diferencias 70 de la nota sí4 de la flauta dulce

Si hacemos una comparativa entre ambas funciones (autocorrelación y de diferencias) encontramos una similitud en que ambas funciones estiman el periodo fundamental basándose en el parecido de una señal con su réplica retardada. Por lo que es obvio que ambas funciones estén relacionadas entre sí. Esta relación viene descrita en la ecuación 20 como sigue [17] [51]:

$$d_t(\tau) = 2[r_t(0) - r_t(\tau)] \quad (20)$$

Donde d_t es la función de diferencias que representa la diferencia entre la autocorrelación en el origen $r_t(0)$ de una señal x_j y la autocorrelación $r_t(\tau)$ de la misma

señal muestra por muestra en tiempo retardada τ muestras. Siendo la señal x_j enventanada con una ventana temporal de W muestras de longitud.

Por tanto, tal como muestra la Figura 18 y atendiendo a la ecuación 20, la función de diferencias es una versión invertida de la función de autocorrelación; donde el primer mínimo marcaría el periodo fundamental.

2.1.1.3 Cepstrum [58]

Este algoritmo parece situarse en el bloque de algoritmos frecuenciales. Pero debido a que utiliza muestras retardadas y por guardar relación con los algoritmos anteriormente presentados se le clasifica como algoritmo temporal. [50] [58]

El Cepstrum se expresa en el dominio de tiempo al aplicar la Transformada de Fourier Inversa como el logaritmo de la Transformada de Fourier de una señal. Tiene relación con la función de autocorrelación de tal forma que es igual a $\frac{1}{2}\log(r_t(\tau))$. De tal forma que si la función de autocorrelación se expresa como sigue [50] [58]:

$$r_t(\tau) = IDFT\{|DFT[x_t(n)]|^2\} \quad (21)$$

Entonces la función cepstral será:

$$c_t(\tau) = IDFT\{\log(|DFT[x_t(n)]|)\} \quad (22)$$

Donde $(c_t(\tau))$ es la función cepstral que corresponde a la transformada discreta de Fourier inversa del espectro de una señal $x_t(n)$ en unidades logarítmicas.

El logaritmo lo que provoca es aplanar las funciones, lo que tiene como consecuencia que una señal inarmónica sea aproximada a un sistema lineal. El inconveniente está en que no tiene buena resolución espectral, por tanto este algoritmo es propenso al ruido y deberá utilizarse en entornos de bajo nivel de ruido.

2.1.1.4 Generalized autocorrelation Function [58]

Debido a que la función cepstral tiene un gran problema en señales que incluyen ruido. Se propone como solución a este problema un algoritmo generalizado de la función de autocorrelación que a diferencia de la ecuación anterior (ver ecuación 21) la función generalizada de autocorrelación sustituye el cuadrado de la ecuación por dos tercios (ver ecuación 23) [59] [58]

$$r_t(\tau) = IDFT \left\{ |DFT[x_t(n)]|^{\frac{2}{3}} \right\} \quad (23)$$

2.1.1.5 Algoritmos de detección de eventos. Zero Crossing Rate (ZCR) [58]

Este se basa en recomtar el número de eventos que suceden en una señal periódica. Ya que se fundamenta este algoritmo en que, si una señal es periódica, sus eventos también serán periódicos. Por tanto, el número de eventos por segundos marcará el periodo de la señal. En una forma más resumida, el algoritmo ZCR calcula el número de veces que la señal cruza por cero por unidad temporal [60] [58].

2.1.1.6 Algoritmo de YIN

Este es el algoritmo que más se ha utilizado para la detección de pitch. Está basado en la función de autocorrelación al que se suman otros pasos para mejorar los resultados [17].

Una de las características importantes de este algoritmo es que tiene una tasa de error muy baja si se comparan con otros resultados. Es capaz de detectar cualquier rango de frecuencia por lo que puede ser utilizado para instrumentos de diversos tipos hasta para la voz humana. Este algoritmo se describe en cuatro pasos [17]:

Paso 1: Función de autocorrelación

En este paso se plantea la función de autocorrelación de las dos formas arriba descritas (ecuación (17) y ecuación (18)). Aunque por lo general trabajan con la ecuación (17) debido a que no se ve afectada por el tiempo a diferencia de la ecuación (18) donde pudimos comprobar que conforme aumenta el retardo la función se decrece.

Paso 2: Función de diferencias (AMDF)

En este paso los autores proponen trabajar con la función de diferencias relacionando ésta con la función de autocorrelación como se pudo estudiar anteriormente. Para esto se utiliza la ecuación (20):

Utilizar esta función respecto a la función de autocorrelación del paso uno tiene como consecuencia una reducción considerable del error bruto que se comete en la estimación de pitch; porque pasamos de tener un error bruto de 10% a uno de 1,95% (ver tabla 6) por lo que hay una mejora considerable. Ya que este algoritmo pretende demostrar que conforme

se avanza en los pasos que contiene hay una mejora considerada en términos de error de pitch estimado.

Paso 3: Función de diferencias normalizada de media acumulativa

Esta función resulta de dividir la función de diferencias muestra a muestra por su media. La diferencia notoria que tiene con la función de diferencias es que en el origen comienza en 1 y no en 0 como lo hace la función de diferencias. Como hemos explicado con anterioridad este paso reduce el error en 1.69 % respecto al paso anterior es decir una reducción de un 0.26 % de error bruto como viene recogido en la tabla 6. El tercer paso el periodo fundamental será el primer valle distinto de uno [17].

$$d'_t(\tau) = \begin{cases} 1, & \tau = 0 \\ \frac{d_t(\tau)}{\left(\frac{1}{\tau}\right) \sum_{j=1}^{\tau} d_t(\tau)}, & \tau \neq 0 \end{cases} \quad (24)$$

Donde d'_t es la función de diferencias normalizada de media acumulativa que representa la división entre la función de diferencias $d_t(\tau)$ muestra a muestra por su media. Esta función tiene amplitud unidad en el origen y distinta para el resto de muestras de la señal.

Paso 4: Función de diferencias normalizada de media acumulativa

Este paso teniendo en consideración de que en la búsqueda de valles existe la posibilidad de que la caída de un armónico distinto al del primer valle (la frecuencia fundamental) sea más acusada por lo que se seleccionaría de forma errónea como frecuencia fundamental este armónico. O lo que es lo mismo que se cometería errores de octava. Y para solucionar este problema los autores proponen un umbral absoluto de tal forma que se escoja el mínimo de periodo que proporciona un valor inferior al umbral fijado en 0.1. Entonces si ningún valor satisface esta condición se elige el mínimo global.

En la Figura 19 se puede observar hasta el cuarto paso del algoritmo Yin:

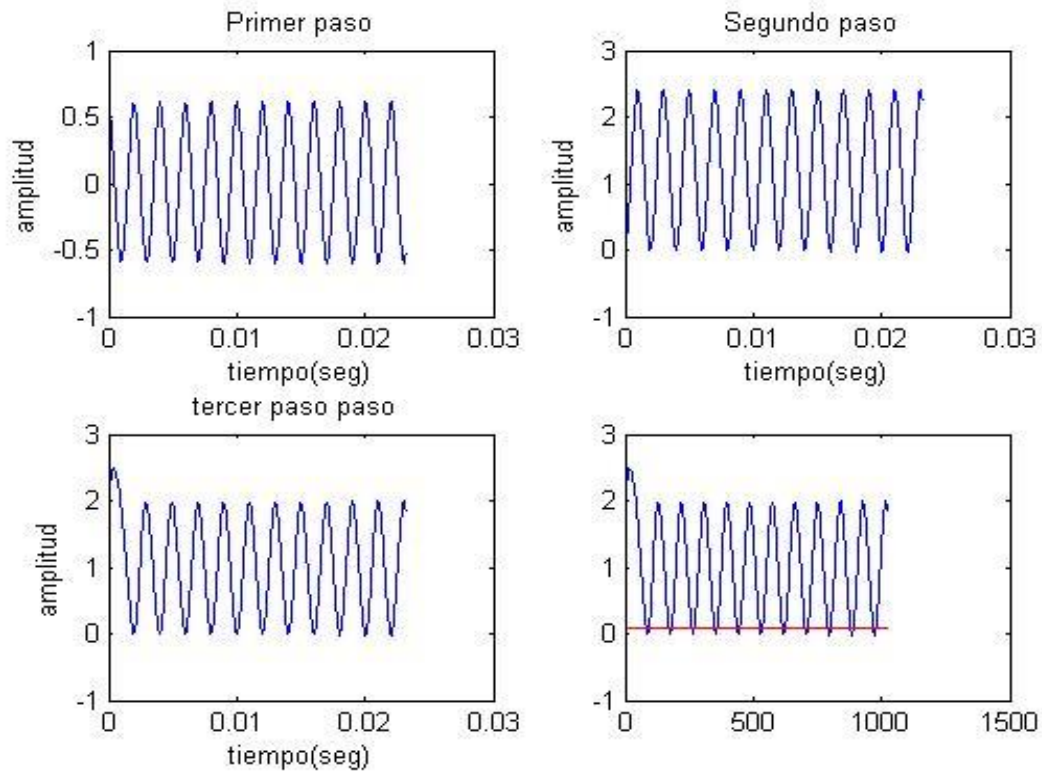


Figura 19: Pasos del algoritmo de YIN

Paso 5: Interpolación parabólica

Una manera de probar el correcto funcionamiento de los pasos anteriores es ver si cumplen con la condición que proponen los autores en este paso. Consiste en que el periodo estimado para cada mínimo local de la función de diferencias normalizada junto a sus vecinos forma una parábola. Entonces la ordenada del mínimo interpolado es utilizada para seleccionar la caída. Por tanto, la abscisa cuyo mínimo es seleccionado se utiliza para estimar el periodo a un múltiplo del periodo de muestreo. Si no se cumple esta condición la estimación puede resultar un tanto incorrecta.

Paso 6: estimación del mejor local

Existe la probabilidad de inestabilidad o fluctuación en la estimación del periodo fundamental pese a la mejora que incluyen los pasos anteriores. Entonces para tener la seguridad de que el valor seleccionado es el óptimo los autores proponen el mecanismo de optimización global basado en una programación difusa con valores que pertenezcan a intervalos distintos y que sean aproximadamente iguales al periodo fundamental estimado por los pasos anteriores.

Versión	Error Bruto (%)
Paso 1	10
Paso 2	1.95
Paso 3	1.69
Paso 4	0.78
Paso 5	0.77
Paso 6	0.50

Tabla 6: Porcentaje de Error Bruto del algoritmo de Yin. [17]

2.1.1.7 Autopitch [61]

Este algoritmo se basa en la búsqueda de un patrón de periodicidad mediante mecanismos automáticos. De tal forma que se toma un determinado intervalo t y se compara con todos los intervalos de la muestra de igual longitud. A través de esta comparación se obtiene un valor P que es el resultado de la suma del valor absoluto de la resta del intervalo original con todos los otros intervalos de la muestra. Este proceso se repite para intervalos de distinta duración y el periodo fundamental es el intervalo que da el P mínimo, por consiguiente su inversa sería la frecuencia fundamental [61].

2.1.1.8 Separación y Acumulación armónica (SAA).

Este algoritmo consiste en obtener el nombre de la nota musical y la octava [62].

El algoritmo se fundamenta en un procedimiento de tres etapas para hallar el valor del pitch [63][65].

Etapas 1: Obtener el nombre de la nota.

Para esto en primer lugar se debe crear doce señales musicales cualesquiera de escala cromática construidas en el dominio de tiempo y que cubran un periodo de seis octavas [63]:

$$N_n(t) = \sum_{n=1}^6 \text{Sen}(2\pi f_n t) \quad (25)$$

Por lo que para hallar la frecuencia de la octava n de f_i se emplea la siguiente ecuación [63]:

$$f_n = f_i * 2^{(n-1)} \quad (26)$$

Donde f_i se obtiene a partir de [64]:

$$f_i = f_0 \cdot \sqrt[12]{2^i} \quad (27)$$

Siendo $f_0 = 110 \text{ Hz}$ que corresponden a la frecuencia de A2 a fin de mantener la afinación estándar de A4 (440 Hz).

En segundo lugar se realiza la multiplicación de cada una de las doce señales por la trama de audio; es decir se realiza una convolución en frecuencia [62].

$$X_n(t) = S(t) * N_n(t) \quad (28)$$

En tercer lugar se calcula la energía de cada una de las señales que resultan de la multiplicación anterior [62]

$$E_n = \sum |X_n(t)|^2 \quad (29)$$

Donde $X_n(t)$ es la señal que resulta de la multiplicación de la señal musical n por la trama de audio.

En cuarto lugar se debe hallar el valor más significativo de la energía; es decir el que se difiera más en amplitud respecto a sus vecinos locales (no necesariamente es siempre el máximo). Esta búsqueda de valores significativos se puede llevar a cabo con la ecuación 30). [63].

$$\gamma_n = |E_n - E_{(n-1)}| + |E_n - E_{(n+1)}| \quad (30)$$

Donde se llamara δ_n al valor más significativo del vector γ_n .

En quinto lugar se busca la posición que ocupa el valor más significativo dentro de los nueve primeros armónicos de la serie armónica.



Figura 20: Serie armónica de la nota C2. Fuente [66]

Etapas 2: Obtener el número de octava de la nota.

En primer lugar tras conocer el nombre de la nota en la etapa anterior, se obtiene por tanto su frecuencia en la octava inferior usada y se crean cinco señales en cada una de las seis octavas.

$$N_n(t) = \sin(2\pi f_n t) \quad (31)$$

En segundo lugar se multiplica cada una de las seis señales por la trama de audio (ecuación 31) [62].

En tercer lugar se calcula la energía de cada una de las señales resultantes de la multiplicación.

En cuarto lugar se halla el valor de la energía más significativa de las seis señales que coincidirá con el número de la octava.

Etapa 3: Hallar la frecuencia fundamental

La frecuencia fundamental se halla mediante la siguiente ecuación.

$$f_0(Hz) = 55x 4^{\frac{(2+a)}{24}} x 2^{(b-1)} \quad (32)$$

Donde a es el valor numérico de la nota en semitonos que resulta de la etapa 1, b es el número de octava que resulta de la etapa 2.

2.1.2 Algoritmos espectrales

2.1.2.1 Harmonic Product Spectrum (HPS) [58]

Este algoritmo se basa en el cálculo de la máxima coincidencia de los armónicos para cada trama espectral tal como viene descrito en la ecuación (25) [50] [58].

$$Y(\omega) = \prod_{r=1}^R |X(\omega r)| \quad (33)$$

$$\forall = \max_{\omega i} \{Y(\omega i)\} \quad (34)$$

Donde R es el número de armónicos considerados para el producto, ωi es el rango de frecuencias candidatos a ser la frecuencia fundamental. En la función $\forall$ se queda con el valor máximo devuelto por la ecuación 33.

Se recomienda la implementación de este algoritmo a través del espectro comprimido de la misma señal de tal forma que cuando se haga la comparación del espectro con sus réplicas se pueda comprobar que las frecuencias están alineadas con la frecuencia fundamental.

Las ventajas que incluye este algoritmo es que es robusto frente al ruido. Y las desventajas son que es costoso, comete errores de octava y su resolución depende de la ventana utilizada. Es considerable pues que si se quiere obtener mayor resolución se debe aumentar la ventana en tiempo.

2.1.2.2 Subharmonic Summation Algorithm (SHS) [58]

El algoritmo HPS surge como una alternativa al algoritmo HPS, ya que este último necesita señales armónicas con características constantes para que puedan funcionar de forma correcta; si se tiene en cuenta que en las señales donde falta un solo armónico este algoritmo no funciona de forma óptima. Es por esto que surge el algoritmo HPS [58].

Este algoritmo se fundamenta en HPS anteriormente vista [67]. La única diferencia que incluye es que opera con sumas en vez de multiplicaciones lo que ayuda a que a falta de armónicos en una señal no se detenga el análisis, con lo que es una mejora considerable respecto al algoritmo al que hace referencia.

La ecuación que describe este algoritmo es la ecuación (35) [58] [67].

$$Y_{SHS}(\omega) = \sum_{r=1}^R |X(\omega r)| \quad (35)$$

$$\forall_{SHS} = \max_{\omega i} \{Y_{SHS}(\omega i)\} \quad (36)$$

2.1.2.3 A Method Combining LPC-BASED Cepstrum and HPS for Pitch Detection [58]

Este algoritmo se basa en la estimación de pitch para señales vocales. De tal forma que se elimina la información del tracto vocal así como el ruido de alta frecuencia con LPC [68]. Este método está orientado a la estimación del pitch de señales vocales. Para ello el algoritmo elimina la información del tracto vocal y el ruido de alta frecuencia mediante LPC [17]. Por lo que este algoritmo está proyectado a trabajar en entornos con un ruido considerable.

2.1.2.4 FFT-método [69]

Este algoritmo utiliza el método de estudio directo del espectro de la señal dentro de un rango de frecuencias situado entre 95 y 500 Hz. Está basado en los siguientes pasos [69] [70]:

- 1) Se selecciona la frecuencia f que tiene el máximo del espectro dentro del intervalo [95, 500] Hz.
- 2) Se seleccionan todos los máximos locales en el intervalo anterior que tenga una intensidad inferior por lo menos del 10% de la intensidad de la máxima frecuencia anterior.
- 3) De los posibles candidatos se seleccionan los que son división entera de f .
- 4) La frecuencia fundamental correspondería al candidato que tenga el menor divisor encontrado.

2.1.2.5 Método de multi-resolución

Es una mejora que se puede aplicar a los algoritmos de estimación de pitch en el dominio espectral. De tal forma que la estimación de pitch que se obtiene de un algoritmo

pueda ser rechazada o aceptada utilizando el mismo algoritmo en distintas resoluciones; lo que significa que se usa una ventana más grande o pequeña para realizar el análisis espectral. Si se tiene el mismo valor para distintas resoluciones, pues se valida la estimación. [70] [71] [72] [73]

Este procedimiento en la mayoría de los casos obtiene resultados satisfactorios pero es muy costoso computacionalmente.

2.1.2.6 *Máxima verosimilitud (ML)*

En este algoritmo se busca en un conjunto de espectros ideales predefinidos el que mayor similitud tenga con el espectro de la trama que se analiza. Si la diferencia entre dos espectros es mínima significa que el espectro que se analiza tiene gran similitud con la trama en cuestión por lo que se le asigna su frecuencia fundamental [74].

2.1.2.7 *Modelo de Goldstein*

Este algoritmo pretende determinar la frecuencia fundamental a partir de sonidos complejos que tengan más de dos componentes frecuenciales obteniendo el mejor patrón de armónicos que aparece en el espectro. Se fundamenta en los siguientes pasos [75]:

- 1) Análisis con FFT de una muestra de 1024 muestras
- 2) Selección de picos : se calcula el máximo global del espectro, y se seleccionan los máximos locales que no sean inferiores al 15% del máximo global calculado
- 3) Se procesan los picos en busca del mejor patrón de armónicos.
- 4) La frecuencia fundamental es seleccionada a partir del estimador de máxima verosimilitud.

2.1.3 *Algoritmos perceptuales*

Haciendo un resumen general de los tipos de algoritmos vistos hasta ahora y el método utilizado para estimar el pitch podemos decir que el primer tipo visto son los algoritmos temporales que estiman el pitch por las propiedades de la periodicidad de señales que incluye la función de autocorrelación. El otro tipo visto hasta ahora son los algoritmos espectrales, los cuales estiman el pitch por propiedades de periodicidad espectral. Por lo que el último bloque a presentar son los algoritmos perceptuales, los cuales estiman el pitch midiendo la envolvente de la señal temporal basado en el estudio del modelado del oído

humano. El cálculo de la envolvente se obtiene a través de un rectificador de media seguido de un filtro paso bajo. Este rectificador se describe de la siguiente forma [58]:

$$HWR(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (37)$$

Un algoritmo perceptual lo tenemos en el modelo unitario de percepción de pitch.

2.1.3.1 Modelo unitario de percepción de pitch

Es un modelo computacional de percepción de pitch propuesto por Meddis y O'Mard. En este modelo se realiza el análisis en las distintas salidas de un banco de filtros paso banda [76].

2.1.3.2 Algoritmo de Terhardt

Este algoritmo está basado en aspectos psicoacústicos a partir de señales temporales. Se fundamenta en los siguientes pasos [77]:

- 1) Análisis espectral: Con el algoritmo fft y una ventana de haming se calcula el espectro de la señal.
- 2) Extracción de componentes tonales
- 3) Evaluación de los efectos de enmascaramiento
- 4) Pesos de componentes
- 5) Extracción de pitch virtual

3 CAPITULO 3. OBJETIVOS

3.1 OBJETIVO PRINCIPAL

Este TFG pretende implementar una herramienta que simule la función de un profesor virtual, de tal forma que dicha herramienta informe al alumno de si la secuencia de notas musicales que se dispone a tocar es o no la correcta, indicando incluso cuáles de ellas (las notas tocadas) han sido erróneas y si el error se ha cometido en frecuencia o en tiempo a fin de que el alumno pueda mejorar en estos dos aspectos. El cumplimiento de este objetivo se debe efectuar al final en una interfaz gráfica (más adelante se va a detallar el funcionamiento del mismo, confeccionando los ejercicios del tutor), la cual contendrá la información visual del objetivo principal planteado.

En la Figura 21 se tiene un ejemplo de la interfaz gráfica que dará respuesta a los objetivos propuestos en este TFG.

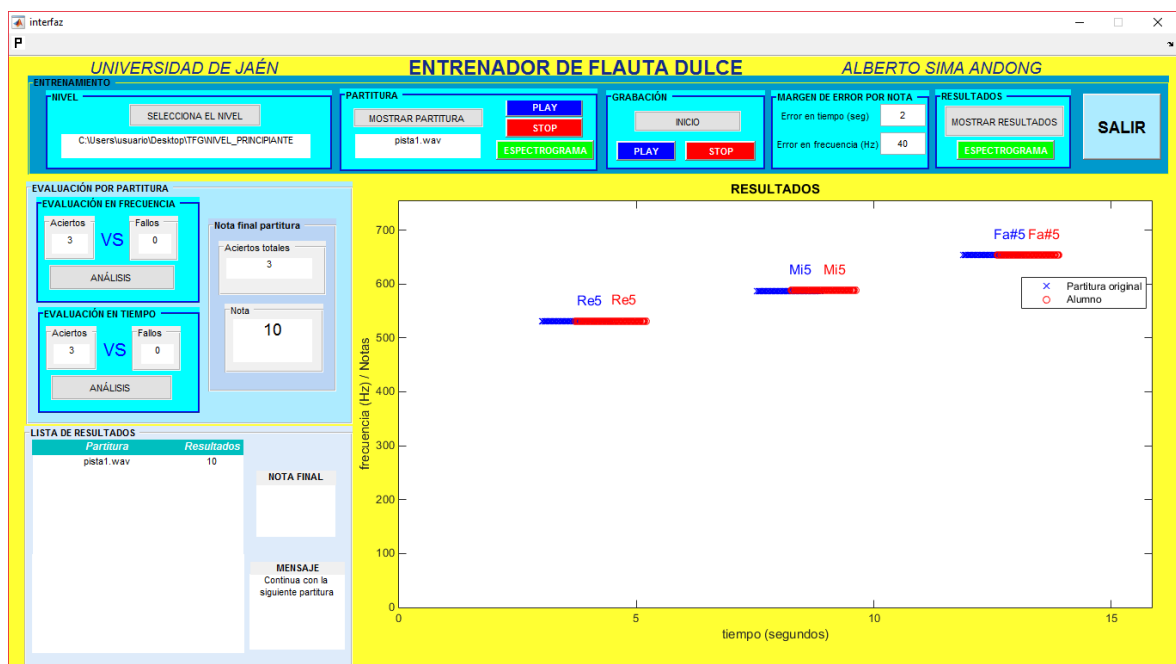


Figura 21: Ejemplo del funcionamiento de la interfaz gráfica cumpliendo los objetivos marcados: análisis en frecuencia y en tiempo.

3.2 OBJETIVOS SECUNDARIOS

Para conseguir el objetivo principal debemos trabajar para la consecución de varios sub-objetivos de los que depende:

- A) Grabación: es necesario implementar un sistema capaz de grabar el sonido que emitamos con el instrumento flauta.
- B) Enventanar la señal grabada: para trabajar con tramos cortos de la misma, si tenemos en cuenta el carácter de periodicidad de las señales, nos indica que las señales sonoras no son periódicas en un tiempo indefinido, es por esto que es aconsejable trabajar con tramas cortas de señal sonora que son periódicas o cuasi periódicas.
- C) Detección de onset y offset: como la finalidad de nuestra herramienta es estimar la frecuencia fundamental, y a sabiendas que las señales sonoras están compuestas de silencios y parte sonora, por lo que es imprescindible para el correcto funcionamiento de nuestra herramienta que el análisis solo sea aplicable a tramas sonoras de nuestra señal; y para esto tendremos que conocer el inicio (onset) y final (offset) de cada nota.
- D) Estimación de pitch: la estimación de pitch o frecuencia fundamental la realizaremos haciendo uso de la implementación del algoritmo de estimación de pitch Yin, que nos devuelve el periodo fundamental y siendo la inversa del mismo la frecuencia fundamental.
- E) Transformación a notación musical midi: consistente en convertir las frecuencias fundamentales devueltas por el Yin a midi.
- F) Creación de una interfaz gráfica multidisciplinar que permita al usuario obtener información visual y auditiva de las pruebas que esté realizando. De esta forma tendría la información completa y un espacio didáctico que le sirva de ayuda durante el aprendizaje de la flauta dulce.

4 CAPÍTULO 4. DESARROLLO

4.1 PUNTO DE PARTIDA Y PLANTEAMIENTO

Tras realizar un estudio del estado del arte de los distintos algoritmos de estimación de pitch en el Capítulo 2. Y siguiendo como referencia [58], el algoritmo con mejores prestaciones es el algoritmo temporal de estimación de pitch, Función Diferencial Normalizada de Media Acumulativa (CMNDF), cuyo corazón se basa en el algoritmo YIN [17], utilizando para el análisis y procesado ventanas rectangulares de 1024 muestras. Por tanto este será el algoritmo que se va a implementar para llevar a cabo el desarrollo de este TFG.

La implementación del algoritmo CMNDF que se va a desarrollar en este TFG, propone la implementación del YIN hasta los pasos que no se relacionen con la esencia de las señales sonoras; por lo que abarca el YIN hasta el cuarto paso, si bien se tiene en cuenta que el quinto paso de la interpolación parabólica no es imprescindible al no ofrecer mejoras sustanciales, mientras el sexto paso requiere de la iteración de un nuevo algoritmo lo que a efectos prácticos supondría un coste elevado para la mejora que ofrece (ver tabla 6). Esta implementación se realizará en el entorno Matlab.

A continuación, se va recordar al lector las ecuaciones que rigen este algoritmo para que cuando más tarde se implemente tenga un seguimiento hilvanado sobre todos los pasos que se realizarán a partir de ahora.

- **Cumulative Mean Normalized Difference Function (CMNDF)**

La Función Diferencia Normalizada de Media Acumulativa, es una función que resume el algoritmo de Yin explicado anteriormente en cuatro pasos:

Paso 1: La función de autocorrelación

En este paso se va a utilizar la función de autocorrelación a corto plazo (ACF) presentada por los autores, que se ha explicado anteriormente (ver ecuación (17)), que a continuación se va a recordar:

$$r_t(\tau) = \sum_{j=t+1}^{t+W} x_j x_{j+\tau}$$

Donde r_t es la función de autocorrelación que representa la multiplicación muestra por muestra en tiempo entre una señal x_j por su réplica $x_{j+\tau}$ retardada τ muestras. Siendo la señal inventanada con una ventana temporal de W muestras de longitud.

Paso 2: La función de diferencia

En este paso se va a utilizar la función de diferencia de media acumulativa, adoptando la ecuación que relaciona ésta con la función de autocorrelación arriba descrita y como bien viene explicado en el correspondiente apartado de Yin (ver ecuación (20)) Para su uso se va a recordar la siguiente ecuación:

$$\mathbf{d}_t(\tau) = 2[\mathbf{r}_t(\mathbf{0}) - \mathbf{r}_t(\tau)]$$

Donde d_t es la función de diferencias que representa la diferencia entre la autocorrelación en el origen $r_t(0)$ de una señal $\mathbf{x}_j$ y la autocorrelación $r_t(\tau)$ de la misma señal muestra por muestra en tiempo retardada τ muestras. Siendo la señal $\mathbf{x}_j$ enventanada con una ventana temporal de W muestras de longitud.

Paso 3: La función de diferencias normalizada de media acumulativa

Se debe prestar especial atención a este paso, porque generalmente a partir de él se puede estimar el periodo fundamental con un error inferior. Esta función como se puede ver en la ecuación que la describe, depende de la función de diferencias arriba planteada, la diferencia entre ambas consiste en que ésta normaliza a la anterior dividiendo cada muestra de ésta por su media, que será cada vez más extensa conforme incrementamos el retardo. También es importante señalar que una de las mejoras que incluye esta nueva función es que a diferencia de la función anterior que comenzaba en cero, ésta comienza en uno, por lo que tiende a permanecerse alto para pequeños retardos; y la ecuación que vamos a emplear en este paso es (ver ecuación 24)

$$\mathbf{d}'_t(\tau) = \begin{cases} \mathbf{1}, & \tau = \mathbf{0} \\ \frac{\mathbf{d}_t(\tau)}{\left(\frac{1}{\tau}\right) \sum_{j=1}^{\tau} \mathbf{d}_t(\tau)}, & \tau \neq \mathbf{0} \end{cases}$$

Donde $\mathbf{d}'_t$ es la función de diferencias normalizada de media acumulativa que representa la división entre la función de diferencias $\mathbf{d}_t(\tau)$ muestra a muestra por su media. Esta función tiene amplitud unidad en el origen y distinta para el resto de muestras de la señal.

Paso 4: Umbral absoluto

Como antes fue explicado, y se puede observar en la Figura 19, se sabe que la función anterior devuelve una función de valles, por lo que el sentido del umbral es marcar el intervalo de los posibles candidatos a ser periodos fundamentales; es por eso que los

autores [17] proponen que el periodo fundamental sea el del primer valle que supera el umbral, estando situado éste en 0.1. Aquí se debe resaltar que en las sucesivas pruebas realizadas para la implementación del algoritmo se ha modificado este umbral haciéndolo dinámico para que sea reajutable con cualquier señal de entrada, ya que todas las señales no tendrán la misma energía; es por esto que fijar el umbral en un determinado valor puede inducir a errores; por esto se creó conveniente que el umbral sea dinámico y reajutable dependiendo de la energía de la señal de entrada. Esto se explicará con más detalle más adelante.

Una vez retomado la base en la que se sustenta el algoritmo CMNDF, en el siguiente capítulo se va a desarrollar los procedimientos efectuados para la implementación de este algoritmo que nos ayudarán a alcanzar el objetivo trazado anteriormente.

4.2 METODOS Y MATERIALES

Este Trabajo Fin de Grado, nos presenta una herramienta dividida en tres fases, todas implementadas en el entorno Matlab. La primera fase consiste en un sistema de grabación que permite capturar las señales de audio emitidas por nuestra fuente, “flauta dulce”, ya que si se tiene en cuenta uno de los objetivos (ver apartado 3.2.A), no se entiende el cumplimiento del conjunto de los objetivos marcados si no se es capaces de capturar la señal de nuestra fuente para su posterior análisis.

La segunda fase consiste en el análisis del procesado de la señal grabada, llevada a cabo mediante el algoritmo de estimación mono-pitch CMNDF, que ya fue explicado en el capítulo anterior cuya implementación se abordara más adelante. Gracias a la implementación de este algoritmo; el usuario podrá conocer la información de cuán parecida o no es la nota que esté tocando con la flauta y los onsets y offsets de la misma en comparación a la que se dispone a tocar marcada en la partitura, con el fin de ofrecer al usuario final una herramienta útil que pueda servirle de entrenamiento para su aprendizaje a la flauta.

La tercera fase, lo que hace es simplemente transformar la nota tocada correspondiente a una determinada frecuencia a notación musical MIDI, así como dar nombres a las notas tocadas, para la correcta interpretación del resultado de aprendizaje para aquellos usuarios que no conozcan las notas tocadas por sus frecuencias ni midi, sino por su nomenclatura musical.

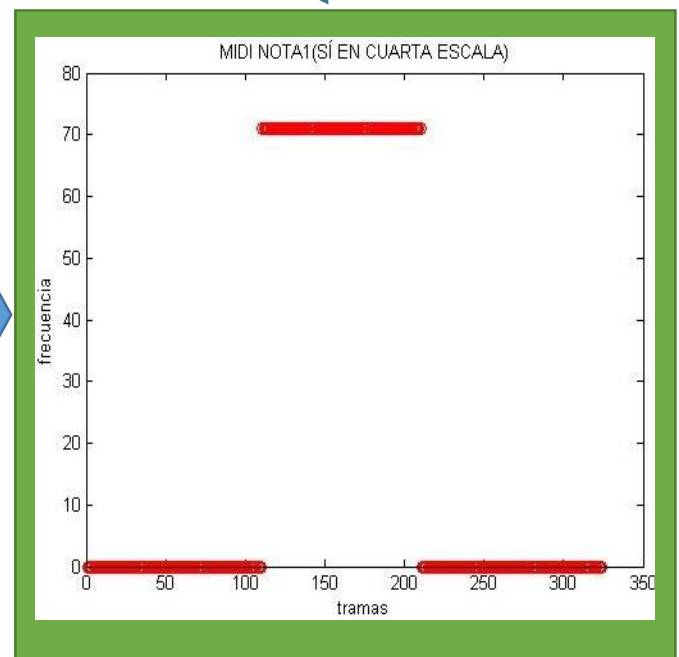


Figura 22: Esquema del sistema a implementar en el TFG.

A continuación, antes de explicar detalladamente los mecanismos seguidos para la consecución de cada fase, primero se pondrá de manifiesto el estudio de las diversas bases de datos con las que se va a trabajar en este TFG.

4.2.1 Base de datos de música

4.2.1.1 Introducción

Para realizar la implementación de este algoritmo se necesitará una base de datos que nos sirva de señal de entrada, para así poder comprobar los resultados que nos devuelva nuestro algoritmo, con lo que nos surge la idea de buscar una base de datos que satisfaga nuestro propósito. Para esto primero se describirá al lector las distintas bases de datos con las que se puede trabajar en este TFG para luego especificarle la base de datos que se ha seleccionado. Para llevar a cabo dicha selección se ha tenido en consideración la base de datos que ha sido utilizada ampliamente para trabajos de las mismas características que el nuestro y también en la rigurosidad y aceptación que ha tenido dicha base de datos en la comunidad científica.

Según el Doctor en telecomunicaciones **Francisco Jesús Cañadas Quesada**, en su tesis doctoral [11] sostiene que las bases de datos se pueden clasificar en dos grupos: base de datos de sonidos sintéticos y base de datos de sonidos reales. Siendo las primeras generadas por un sintetizador de ficheros MIDI (sintetizadores profesionales, que pueden generar sonidos semejantes a los generados por instrumentos reales); mientras que las bases de datos de sonidos reales son generadas por instrumentos físicos reales tocados por profesionales con la particularidad de que cuentan con la interpretación del músico, las características acústicas de la sala y del propio instrumento.

Teniendo en cuenta esta clasificación de las distintas bases de datos, nos surge la idea de que es aconsejable trabajar con el tipo de base de datos de instrumentos reales ya que se asemejan más al detalle con el ámbito real y la información que contienen es más comparable con las particularidades de un instrumento real.

En esta misma línea el doctor prosigue con la clasificación de las bases de datos dentro del tipo de sonidos reales, clasificándolos en dos sub-grupos:

Bases de datos de sonidos individuales: está compuesta por un conjunto de notas musicales tocadas de manera aislada por distintos instrumentos y de distintas familias, músicos, estilos; etc. Las notas se encuentran bajo un orden aleatorio, siguiendo ciertos criterios como el número de instrumentos, número de notas concurrentes; etc. Estas notas se encuentran de forma aislada de tal forma que podemos obtener todas las notas del espectro o rango dinámico de un determinado instrumento que deseemos analizar. De entre este tipo de bases de datos las más conocidas son: Musical Instrument Sound Database de Real World Computing (RWC), [78] [79], University of Iowa Musical Instrument Samples [80], y McGill University's Master Samples (MUMS) [81]. Estas bases de datos se usan para realizar entrenamiento de modelos de instrumentos, o bien, para construir señales de entrada con determinadas características deseadas.

Bases de datos compuestas de fragmentos de composiciones musicales: Se utilizan para evaluar sistemas de separación de fuentes, e incluyen diferentes instrumentos, son grabadas en distintas salas acústicas, distinto nivel de polifonía; etc. Los más usados en este tipo de base de datos son: Classical Music Database and the Jazz Music Database de Real World Computing (RWC) [78] [79] y The Bach Chorals Dataset propuesta en [82].

4.2.1.2 RWC Musical Instrument Sound Database [83]

Esta base de datos [83] está constituida por 50 instrumentos musicales; y ofrece tres variaciones para cada instrumento para un total de 150 actuaciones con el fin de tener una amplia variedad de sonidos, de las que ofrece:

- Variaciones (3 fabricantes, 3 músicos). Contiene grabaciones del mismo instrumento fabricado por tres fabricantes diferentes e interpretado por tres músicos diferentes.
- Estilo de juego. Contiene grabaciones con diferentes técnicas interpretativas y podemos mencionar algunos como normal, staccato, vibrato, todos grabados siempre dentro del espectro de notas que pueden ser generados por cada diferente instrumento.
- Pitch (todo el espectro). Para cada estilo de juego de un determinado instrumento, el músico toca todo el espectro de notas que puede ser generado por el instrumento utilizado, siempre con intervalos de un semitono entre dos notas consecutivas.

- Niveles dinámicos. Para cada grabación realizada por un determinado instrumento se registraron tres niveles dinámicos (forte, mezzo, piano), siempre dentro del mismo espectro de notas posibles, generados por el instrumento en cuestión.

Un ejemplo concreto de esta base de datos se puede encontrar en la creación de la base de datos RWC-MDB-I-2001 No. 01(Piano) donde las 88 notas del piano están grabados usando tres pianos de distintos fabricantes (Yamaha, Bosendorfer y Steinway) con diferentes estilos de juego (normal, staccato, pedal, etc) y con tres niveles dinámicos (forte, mezzo y piano).

Los 50 instrumentos que conforman esta base de datos, sus sonidos están grabados con una frecuencia de muestreo de 44100 Hz, 16 bits; siendo en su totalidad unos 3544 archivos de audio que ocupan una capacidad total de 29.1 Gbyte y un tiempo total de reproducción de 91.6 horas donde vienen incluidos los intervalos de silencio (ver información [83]).

4.2.1.3 *McGill University's Master Samples (MUMS)*

Esta base de datos MUMS [81] está compuesta por 6000 muestras de sonido distintos tocados por un gran número de instrumentos musicales clásicos y con distintos estilos de interpretación. Las notas grabadas se dividen en 2204 de cuerda, 1595 de teclado, 1197 de viento de madera, 1087 de percusión y 463 de viento metal, almacenando en concreto un total de 6546 notas. La duración de las notas es de unos 2 o 10 segundos dependiendo del instrumento que se interprete. Todas las notas están grabadas independientemente bajo una frecuencia de muestreo de 44100 Hz y 24 bits.

4.2.1.4 *RWC Classical Music Database [84]*

Esta base de datos [84] está compuesta por 50 piezas:

- Sinfonías: 4 piezas
- Conciertos: 2 piezas
- Música orquestal: 4 piezas
- Música de cámara: 10 piezas
- Ejecuciones individuales: 24 piezas
- Actuaciones vocales: 6 piezas.

4.2.1.5 *Bach Chorals Dataset [85]*

Esta base de datos [85] se suele utilizar en varias líneas de investigación como en la estimación multi-pitch, Audio-score Alignment, separación de fuentes; etc. Está compuesta por grabaciones de cada instrumento del grupo de diez piezas corales de J.S. Bach con sus correspondientes ficheros MIDI.

Cada pieza está grabada en cuatro partes independientes (soprano, alto, tenor y bajo). Para la realización de estas grabaciones los distintos instrumentos utilizados fueron el violín, el saxofón, el clarinete, y el fagot.

Se ha obtenido de cada parte sus correspondientes valores de pitch o notas de forma independiente para luego combinar todas las fuentes en un solo archivo que contenga la información representativa de las grabaciones por partes.

Para analizar estos archivos se utilizaron tramas de 46ms y salto de 10ms. Mientras que para conseguir el valor de pitch de cada instrumento que forma esta base de datos han empleado el algoritmo de Yin [17]. Gracias a este algoritmo han realizado el procesamiento de todas las grabaciones de cada instrumento trama por trama hasta conseguir el valor de pitch de cada determinada nota.

Tras hacer un breve estudio de las diferentes bases de datos y teniendo en cuenta que se va a trabajar sobre un algoritmo mono-pitch donde la única fuente es la flauta, por lo que en consecuencia se ha determinado que la base de datos que más se ajusta a nuestras necesidades es **RWC Musical Instrument Sound Database [83]**. Es importante mencionar en esta línea que esta base de datos seleccionada con la que hemos abordado este TFG fue conseguida gracias a mi profesor tutor del mismo, el **Doctor D. Francisco Jesús Cañadas Quesada [11]** del departamento de área de teoría de la señal y comunicaciones.

4.2.2 *Base de datos Audacity*

Teniendo ya seleccionada nuestra base de datos, es importante aclarar que la misma que fue proporcionada por el tutor consta de 25 notas distintas que parten desde la nota B4 (nota Sí en cuarta escala, ver tabla 5) hasta B6 (nota Sí en sexta escala) todas en una sola melodía, entonces para el inicio de la implementación se ha creído conveniente trabajar primero con notas aisladas; por lo que se ha creado con el Audacity una base de datos de las

25 notas pero separadas unas con otras. Para esto se va a explicar al lector el funcionamiento del programa Audacity.

- **Audacity**

Audacity es un programa gratuito y su código es abierto con una licencia GNU para editar y grabar audios. Trabaja gráficamente con la forma de onda del sonido que en él se carga; también puede servir de conversión a notación musical midi. Soporta muchos formatos de audio, pudiendo mencionar: wav, mp3, ogg, wma, ac3, flac o eif; incluyendo al mismo tiempo opciones de recortes, borrado y copia. También cuenta con numerosos efectos a los que se puede aplicar a nuestro audio en cuestión; además que permite ajustar el nivel de ruido, modificar la forma de onda incrementando o disminuyendo su intensidad, y es capaz de eliminar la voz entre tantas propiedades de las que goza. Para más información sobre sus propiedades, uso y descarga ver la bibliografía [86] [87].

En la Figura 23 se puede observar la ventana que se abre al arrancar el programa:

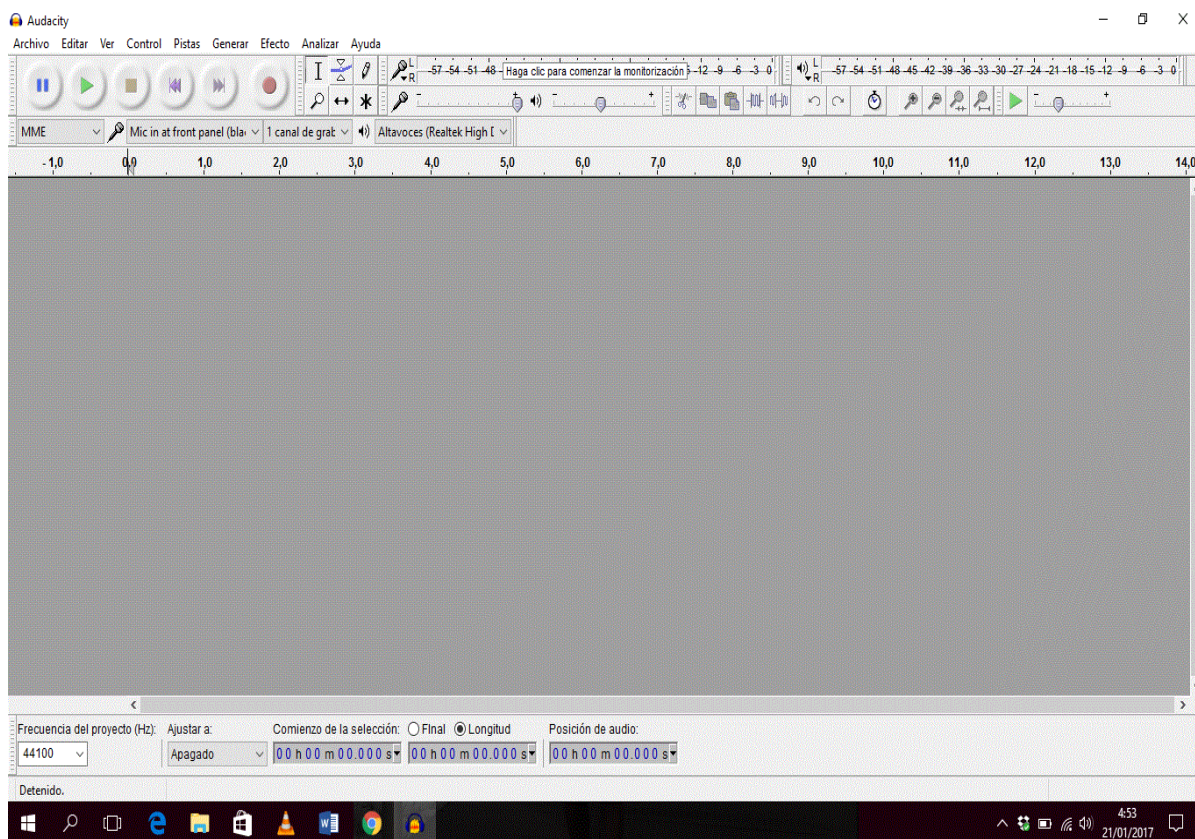


Figura 23: Ventana Audacity sin sonido cargado

En la Figura 24 se ilustra como importar un audio, que en nuestro caso será el ofrecido por nuestra base de datos **RWC Musical Instrument Sound Database** (de 25 notas):

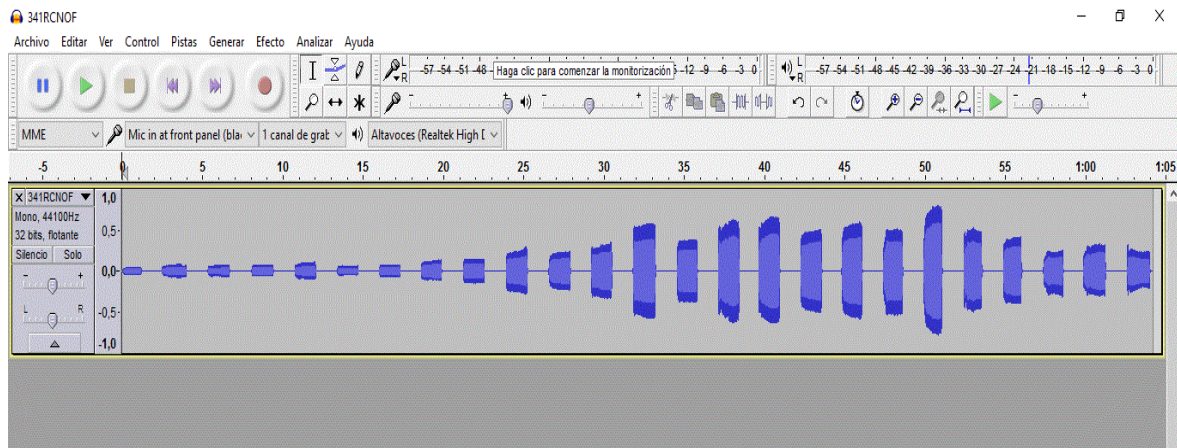


Figura 24: Pista de 25 notas en audacity

En la Figura 25 se puede observar primero una nota sombreada con un círculo amarillo que será la que se cortará de la pista a través de una de las funciones de cortado que ofrece el programa Audacity y que se ha sombreado con un círculo negro y al final se puede ver el resultado del mismo:

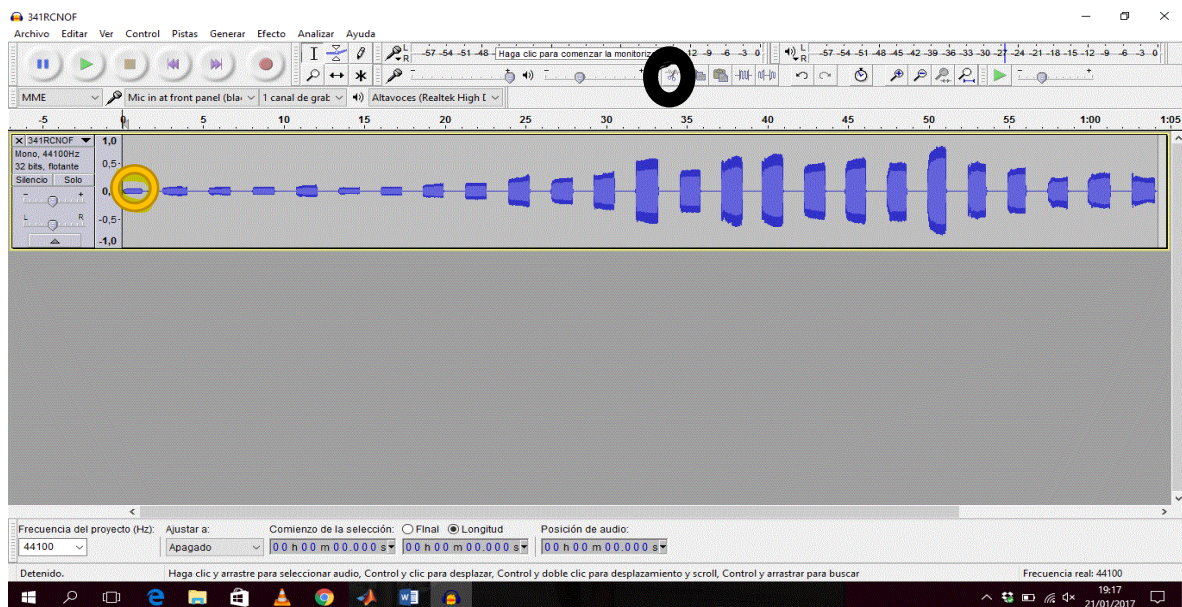


Figura 25: Selección de la nota a separar (usando tijeras) de la pista

Mientras que en la Figura 26 se puede observar el resultado de una nota cortada con el Audacity.

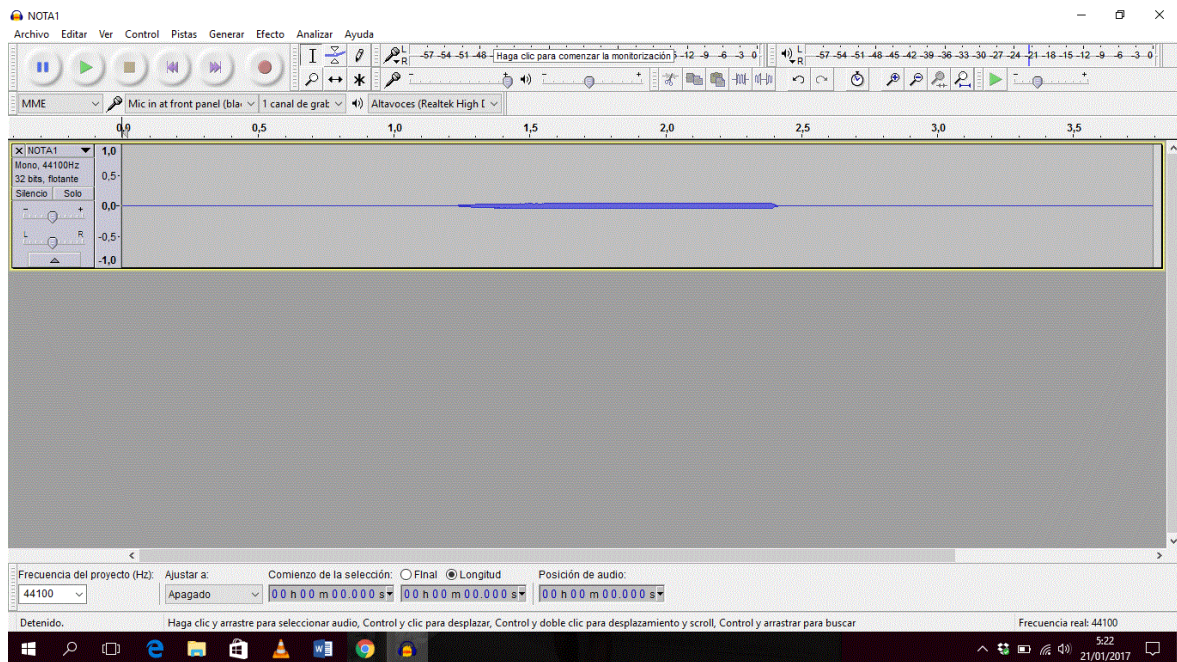


Figura 26: Resultado de separar la nota marcada con el círculo

Con este procedimiento en Audacity separamos las 25 notas y creamos una base de datos que con ella trabajaremos, la cual se encuentra en la carpeta: “ notas_unicas”.

4.2.3 Representación tiempo-frecuencia

Después de conformar nuestra base de datos, el siguiente paso es conocer la frecuencia fundamental de cada una de las notas que contiene. Como se ha explicado anteriormente, la información de las señales sonoras puede ser conocida analizando los dos dominios, el dominio temporal y el frecuencial. Teniendo en cuenta que el algoritmo que se implementará más adelante concretamente es un algoritmo temporal que devuelve como resultado el periodo fundamental al que aplicando su inversa se obtiene la frecuencia fundamental de las notas de entrada, por lo que se hace imprescindible conocer la frecuencia real de las notas con las que se va a trabajar.

Para conocer la información frecuencial de las notas de entrada, se aplicará la Transformada de Fourier de Tiempo Corto (STFT, Short Time Fourier Transform) estudiada en clase, que es un eficiente algoritmo con el que se puede calcular la Transformada de Fourier de señales discretas (DFT). Este algoritmo pone algunas restricciones sobre la señal que se necesita conocer su espectro. De tal forma que se es más eficiente computacionalmente cuando el número de muestras de la señal es igual a la potencia de dos. Generalmente se basa en tomar valores discretos o tramas de una señal continua y se va aplicando la transformada de Fourier a estas tramas obteniendo de esta forma el correspondiente espectro de la señal introducida, lo que se reduce en un proceso de digitalización de la señal continua. En la mayoría de los casos se admite la transformación de un total de muestras (512, 1024, 2048, 4096 muestras); aunque el rango de frecuencias o la precisión en frecuencia depende tanto de las muestras tomadas de la señal continua como de la frecuencia de muestreo.

Si tomamos $x_0, \dots, x_{N-1}$ como un intervalo discreto de números complejos, la Transformada Discreta de Fourier (DTF, por sus siglas en inglés) se define como:

$$X_K = \sum_{n=0}^{N-1} x_n e^{-j2\pi k \frac{n}{N}} \quad k = 0 \dots, N-1 \quad (38)$$

Si se evalúa de forma directa esta ecuación se va a necesitar (N^2) operaciones aritméticas, mientras que con una FFT se puede conseguir el mismo resultado solo con $(n \log n)$ operaciones. Esta optimización que aporta la FFT se debe a la división de la transformada a realizar en otras más reducidas hasta alcanzar la transformada de dos elementos donde k puede tomar valores entre 0 y 1. Después de resolver las transformadas

más reducidas éstas deben ser agrupadas en un grupo superior y así sucesivamente hasta alcanzar el más alto. Un ejemplo ilustrativo del proceso de la FFT lo podemos observar en la siguiente Figura 27:

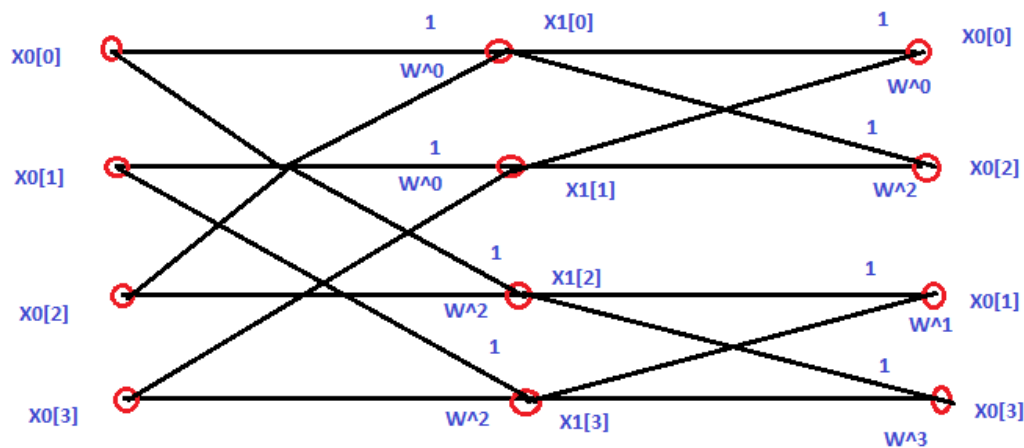


Figura 27: Gráfica de flujo de señales para FFT de cuatro puntos. Fuente [88]

El proceso explicado arriba es el que se ha aplicado a nuestras señales (notas) para conocer su frecuencia fundamental y en consecuencia conocer su nomenclatura y escala musical al que pertenecen (ver tabla 5). El lenguaje de programación (Matlab) utilizado para realizar este TFG incluye una sintaxis para el cálculo de la FFT:

$y = \text{fft}(x, n)$ que devuelve la DFT de n puntos; siendo x la señal de entrada temporal.

La función que nos permite obtener la frecuencia fundamental para cada nota es:

```
function [freq]= BuenFFT (x)
```

Donde la señal de entrada de esta función es:

- x : que es la variable que contiene las notas cuyas frecuencias deseamos conocer.

La señal de salida de esta función es:

- freq : que es la variable que almacena los valores calculados de la frecuencia fundamental a través de la sintaxis fft .

A continuación, se va a explicar el procedimiento para ejecutarlo:

Primero se carga la señal (nota), que se encuentra en nuestra base de datos creada con el Audacity ('' notas_unicas''). En la Figura 28 se puede observar la ventana que se abre cuando ejecutamos el programa:

Para realizar esta operación primero se debe clicar en ''run'' (sombreado en amarillo) y después matlab pedirá que se cambie de carpeta y se tendrá que clicar en ''Change Folder'' que está sombreado en azul; tal como muestra la Figura 28.

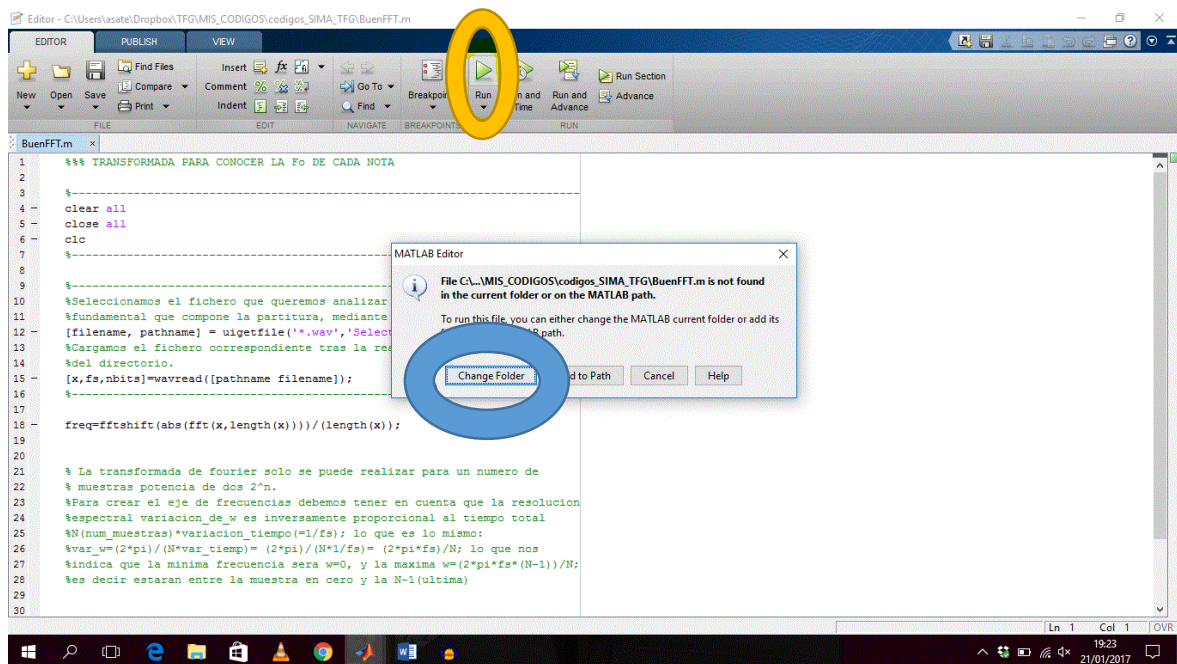


Figura 28: Ejecución del programa en Matlab.

Una vez realizada esta operación el programa solicitará seleccionar la carpeta o la base de datos situada en algún directorio del PC donde se tiene almacenadas las notas cuyas frecuencias se quiere conocer. En la Figura 29 se puede observar cómo se selecciona el directorio que contiene la base de datos ('' notas_unicas''):

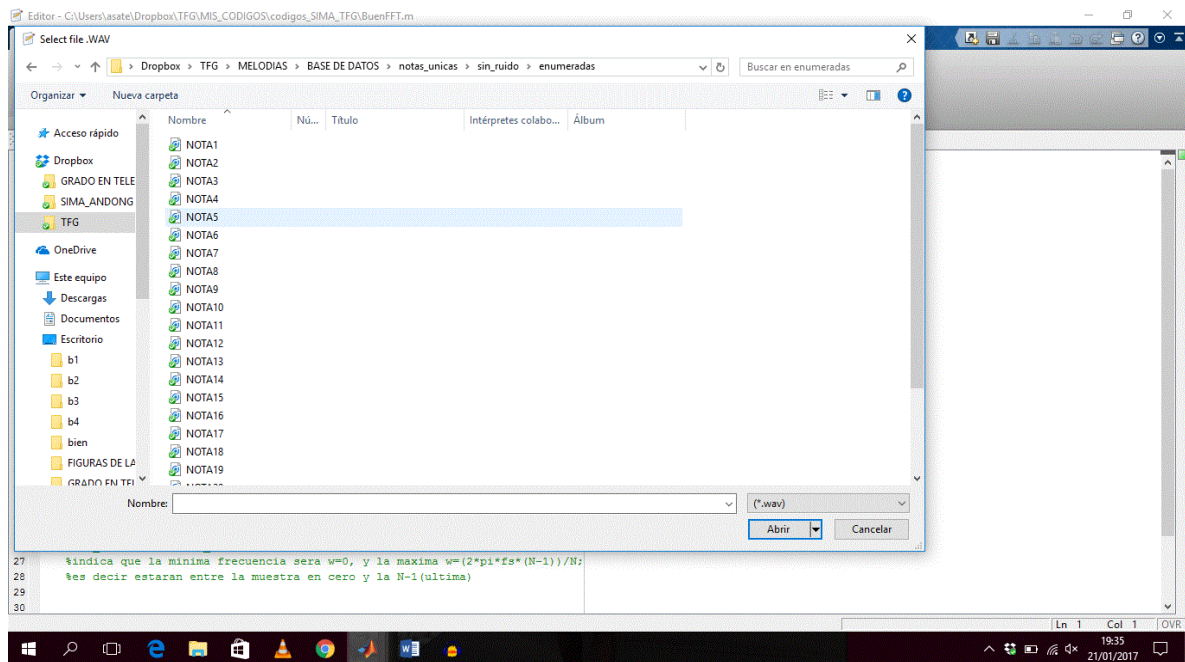


Figura 29: Selección de notas en la base de datos: notas_unicas

Una vez situado dentro de la base de datos solo basta con seleccionar cualquiera de las notas cuya frecuencia se quiere conocer y que como se puede ver están enumeradas (de la Nota1 hasta la Nota 25) porque a priori no se conoce la nomenclatura a la que hacen referencia.

A continuación, se pueden observar las Figuras (30 y 31) el resultado de haber realizado la FFT de un par de notas de nuestra base de datos. Observando la distribución espectral de las mismas se puede ver que es simétrica y centrada en cero siendo la frecuencia fundamental el primer armónico distinto de cero o la frecuencia más baja del espectro. Si nos fijamos bien en la distribución espectral, los siguientes armónicos son cuasi-múltiplos de la frecuencia fundamental y es debido a que la señal no es periódica a largo plazo. Por ejemplo, en la Figura 30, los armónicos de la “Nota2” se sitúan en las siguientes frecuencias:

- Primer armónico (frecuencia fundamental): 533.5 Hz
- Segundo armónico (múltiplo de la frecuencia fundamental): 1067 Hz
- Tercer armónico (múltiplo de la frecuencia fundamental): 1600.5 Hz

Con esta información ya se sabe que las señales con las que se va trabajar son periódicas o cuasi- periódicas por lo que tendrá sentido aplicarles nuestro algoritmo si se tiene en cuenta que la base de éste está en la periodicidad de las señales.

Para observarlo mejor, la frecuencia fundamental se encuentra donde se ha situado el cursor de color negro en las Figuras 30 y 31.

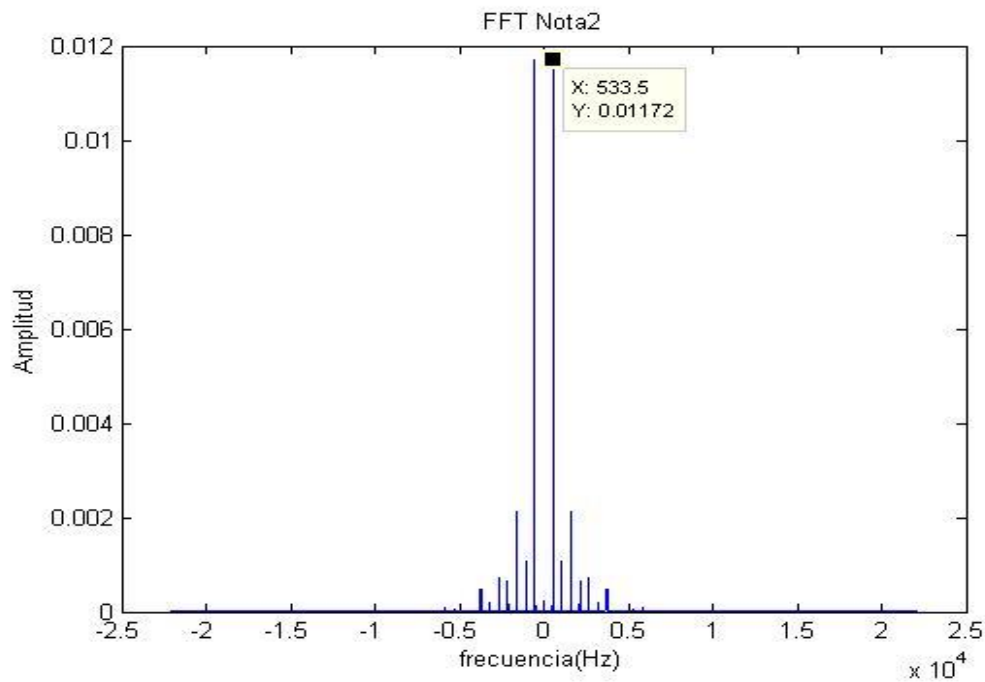


Figura 30: Frecuencia fundamental (situado en el cursor) de Nota2 de la base de datos (notas únicas)

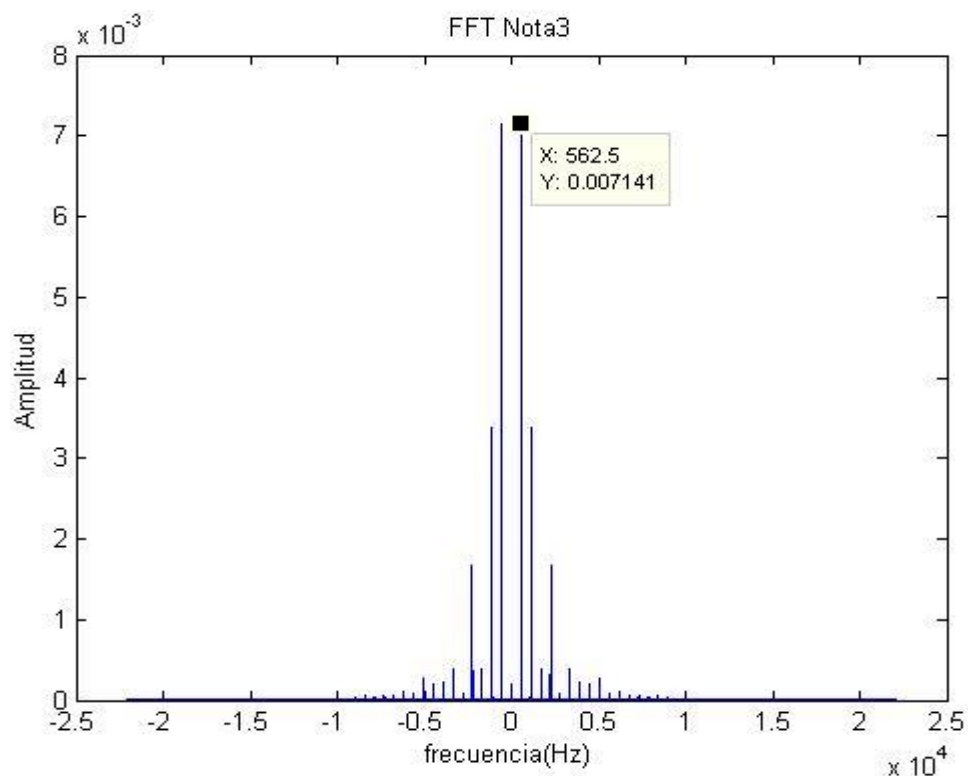


Figura 31: Frecuencia fundamental (situado en el cursor) de Nota3 de la base de datos (notas únicas)

La finalidad con la que se han calculado las frecuencias fundamentales de cada una de las notas de nuestra base de datos, es de conocer la información de la misma como por ejemplo su correspondiente nombre, frecuencia real, midi, y escala. Y para esto una vez calculadas las respectivas frecuencias fundamentales mediante la transformada rápida de Fourier, para sacar esta información se tendrá que usar la tabla 5.

Con la tabla 5 y conociendo la frecuencia fundamental de cada nota obtenida mediante FFT se puede relacionar y sacar el midi, el nombre y la escala de la nota. Basta con buscar donde se encuentra la frecuencia fundamental calculada dentro de la tabla 5 y conocer toda esta información. Por ejemplo, conociendo la frecuencia fundamental de la Nota1 (499.2 Hz) y relacionándola con la tabla 5 se puede determinar que esta frecuencia pertenece a la nota Sí en cuarta escala y de midi 71. La Nota2 (533.5 Hz) pertenece a la nota Do en quinta escala y de midi 72; mientras que la Nota3 (562.5 Hz) pertenece a la nota Do# (Do bajo) en quinta escala y de midi 73.

Volviendo a resumir lo realizado hasta ahora:

1°. Se dispone de una base de datos compuesta por notas aisladas extraídas con el Audacity de la base de datos **RWC Musical Instrument Sound Database**.

2°. Se calcula la frecuencia fundamental mediante la FFT de cada nota y a través de la relación dispuesta con la tabla 5 se extrae la información de la misma (nombre, escala y midi).

De esta forma ya cuando se trabaje con alguna de estas notas en el algoritmo que se implementará, ya se sabrá de qué nota se trata, sus características; así como el resultado que se espera conseguir de ésta.

A continuación, se puede observar la tabla 7 con la información obtenida tras realizar la FFT de todas las notas de nuestra base de datos, así como la información extraída mediante la relación con la tabla 5 sobre su midi, escala, y nomenclatura:

BASE DE DATOS : “notas_unicas”				
NOTAS	FRECUENCIA FUNDAMENTAL (Hz)	MIDI	ESCALA	NOMBRE
Nota 1	499.2	71	4 ^a	Sí
Nota 2	533.5	72	5 ^a	Do
Nota 3	562.2	73	5 ^a	Do#
Nota 4	598.5	74	5 ^a	Re
Nota 5	628.2	75	5 ^a	Re#
Nota 6	667.8	76	5 ^a	Mi
Nota 7	716.8	77	5 ^a	Fa
Nota 8	747.2	78	5 ^a	Fa#
Nota 9	792.3	79	5 ^a	Sol
Nota 10	834.8	80	5 ^a	Sol#
Nota 11	894.2	81	5 ^a	La
Nota 12	943.3	82	5 ^a	La#
Nota 13	1001	83	5 ^a	Si
Nota 14	1062	84	6 ^a	Do
Nota 15	1135	85	6 ^a	Do#
Nota 16	1191	86	6 ^a	Re
Nota 17	1267	87	6 ^a	Re#
Nota 18	1337	88	6 ^a	Mi
Nota 19	1428	89	6 ^a	Fa
Nota 20	1504	90	6 ^a	Fa#
Nota 21	1603	91	6 ^a	Sol
Nota 22	1697	92	6 ^a	Sol#
Nota 23	1792	93	6 ^a	La
Nota 24	1912	94	6 ^a	La#
Nota25	2016	95	6 ^a	Si

Tabla 7: Relación FFT, MIDI, escala y nomenclatura de cada nota de la base de datos notas_unicas

4.2.4 ESTIMACIÓN DE PITCH

4.2.4.1 Implementación del algoritmo CMNDF

Este algoritmo se fundamenta en calcular el periodo fundamental a través del cálculo matemático de la función de autocorrelación; por lo que es trivial pensar que este algoritmo funcionará de forma eficiente para señales periódicas, y aunque los resultados obtenidos al realizar la FFT de cada nota indican que las señales son periódicas al ser sus respectivos armónicos múltiplos de la frecuencia fundamental. Aunque se debe decir que fijándose en la evolución espectral de los mismos resultados los armónicos no son múltiplos exactos de la frecuencia fundamental por lo que existe algo de aperiodicidad, lo que nos llevó a pensar en trabajar con tramas más cortas de la misma nota para asegurar mejor su carácter periódico, ya que esta será la propiedad principal de la que se aprovechará el algoritmo que se va a implementar. Y teniendo en cuenta el apartado anterior donde se configuran los parámetros que tendremos ahora en consideración, hemos llegado a la conclusión de que la trama o ventana que mejor responde en cuanto a la periodicidad es la trama de 1024 muestras o 23.22 milisegundos, muestreadas a una frecuencia de muestreo de 44100 Hz y con un solapamiento del 50 % entre tramas. Esto se debe principalmente al hecho de que el análisis de señales musicales requiere de un mayor procesamiento a nivel de muestra. Además, el hecho de escoger ese tamaño de ventana con el solapamiento seleccionado permite llegar a un equilibrio efectivo entre eficacia y coste computacional. La frecuencia más alta reproducible es la mitad de la frecuencia de muestreo empleada, lo que se conoce como la frecuencia Nyquist. Si utilizamos una frecuencia de muestreo de 44.1kHz, obtendremos una frecuencia Nyquist de 22.050Hz, que estará justo por encima de nuestro umbral de audición. Por lo consiguiente, en la siguiente Figura 32 se va a mostrar la representación en tiempo (Figura izquierda) de la Nota1 de la base de datos correspondiente a Sí en 4ª escala (MIDI 71); y la representación en muestras (Figura derecha) de la misma señal. Como se puede observar, la Nota1 tiene una duración en tiempo de 2.4 segundos (ver el posicionamiento del cursor) y como según los parámetros configurados la frecuencia de muestreo empleada será de $f_s = 44100$ Hz, por lo que para calcular el número de muestras que tiene la Nota1 (Sí4) se aplicará la siguiente ecuación:

$$\text{muestras} = t * f_s \quad (39)$$

Donde t es el tiempo que dura la nota y f_s es la frecuencia de muestreo. Por tanto se puede determinar que el número de muestras de la señal será:

$$muestras = 2.4(seg) * 44100 \left(\frac{muestras}{seg} \right) = 105840 \text{ muestras}$$

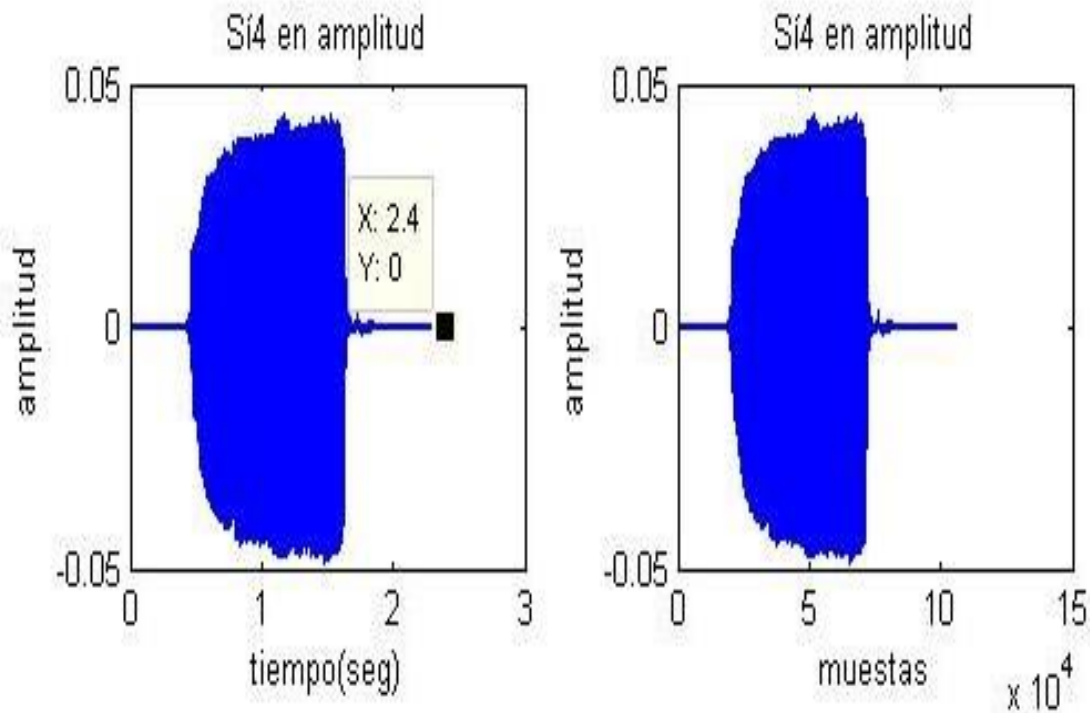


Figura 32: Nota Sí4 representada en tiempo (izquierda) y en tramas (derecha)

En la siguiente Figura 33 se va a representar la nota Sí4 segmentada en tramas de 1024 muestras solapadas al 50%; es decir que si anteriormente se ha visto que esta señal tiene un total de 105840 muestras por lo que si se la divide en tramas de 1024 muestras el número total de tramas resultantes responderá a la siguiente ecuación:

$$N^{\circ}_{tramas} = \frac{muestras}{tamaño\ trama} = \frac{105840\ muestras}{1024\ muestras} = 103\ tramas \quad (40)$$

Pero este número de tramas responde a trazar las tramas o ventanas de forma consecutiva acorde a la longitud en muestras de la nota (esta longitud en muestras es tomada como "*muestras*" en la ecuación anterior (ver ecuación 40)). Mientras que, según los parámetros configurados, siempre con el objetivo de asegurar el carácter periódico de las señales se ha determinado que las tramas deben tener 1024 muestras, pero no consecutivas como se realiza en el cálculo anterior (ver ecuación (32)) sino que estas tramas deben solaparse al 50% o lo que es lo mismo decir que una trama debe tener 512 muestras de la

anterior. El solapamiento de señales se emplea con el fin de reducir el tiempo de análisis. Se puede solapar cualquier tipo de ventanas, pero la Hanning es ideal para este propósito.

Un ejemplo ilustrativo de las tramas solapadas la podemos ver en la Figura 33 donde en la primera fila se tiene una señal con un determinado número de muestras (N). En la segunda fila se tiene esta señal segmentada en tres tramas consecutivas de igual número de muestras formadas de tal forma que la primera trama está compuesta por (TA y TB) siendo estas cada una de la mitad de la primera trama, (TC y TD) son ambas cada una de la mitad de la segunda trama, mientras que TE y TF son cada una de la mitad de la tercera trama consecutiva de la señal. En la tercera fila se empieza a formar el conjunto de tramas solapadas, aunque al ser este el inicio no se solapa y es la misma que la primera trama de la señal. A partir de la cuarta fila se puede ver el solapamiento entre tramas si bien se tiene en cuenta que en la cuarta fila se tiene una trama compuesta por la mitad de las muestras de la trama de la fila tercera, es decir la mitad de la anterior más la mitad de la siguiente ($TB+TC$) y así sucesivamente se van formando tramas hasta la última muestras. Es importante explicar que en la última trama solo contendrá la mitad de la última trama consecutiva de la señal mientras que la otra mitad puede ser completada con ceros, pero esto supondrá tener más muestras de las que parte la señal inicialmente, aunque una opción sería completar la señal con ceros para que tenga la misma longitud. Pero otra opción que es la que se ha utilizado es considerar todas las tramas solapadas excepto la última para así no tener que modificar la señal inicial ya que la última trama nula no aporta información a la señal por lo que puede ser prescindida.

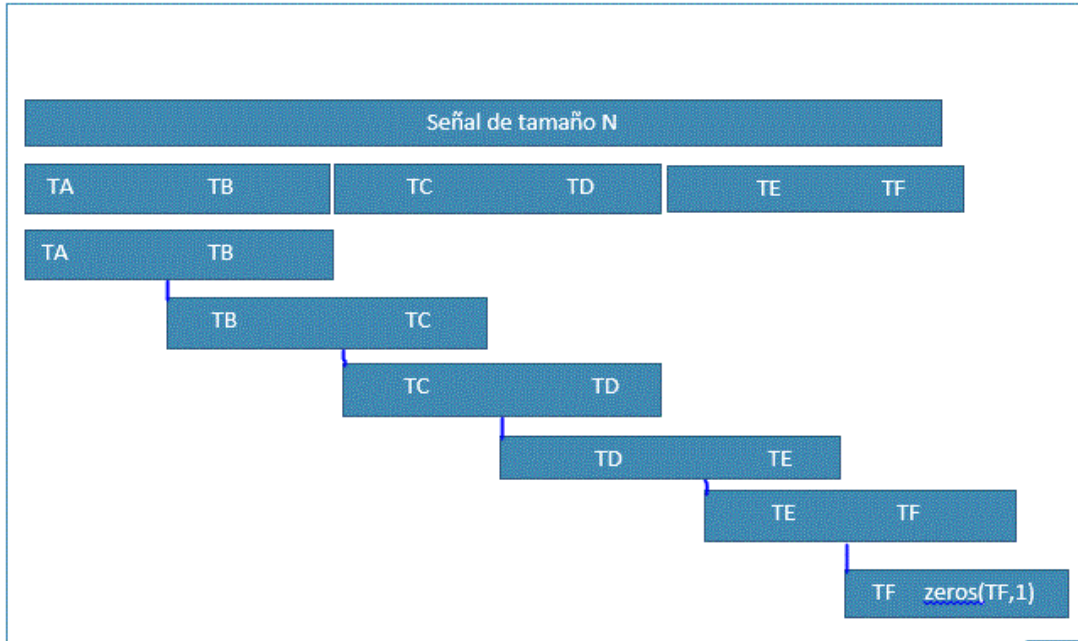


Figura 33: Ejemplo de solapamiento al 50% entre tramas.

En términos matemáticos trabajar con tramas solapadas al 50% respecto a las tramas consecutivas resulta ser el doble del número de tramas como muestra la siguiente ecuación:

$$solape = 2 * (N^{\circ}_{tramas}) \quad (41)$$

Donde *solape* es el número total de tramas solapadas añadiendo ceros a la última trama. Aplicando esta ecuación se tendrá el número de muestras solapadas añadiendo ceros (muestras nulas) que será:

$$solape = 2 * (N^{\circ}_{tramas}) = 2 * (103) = 206 \text{ tramas} \quad (42)$$

Teniendo en cuenta lo explicado anteriormente (para no tener que añadir ceros a la última trama solapada y por tanto a la señal inicial) y considerando que ésta no aporta información relevante a la señal, en nuestra implementación se preferido no añadir ceros y descartar la última trama solapada, por lo que la ecuación quedaría:

$$ncol = (solape - 1) \text{ tramas} \quad (43)$$

Donde $ncol$ es el número total de tramas solapadas que se ha utilizado en nuestra implementación y aplicando esta ecuación se tendrá que:

$$ncol = (solape - 1) = 206 - 1 = 205 \text{ tramas} \quad (44)$$

Llegados a este punto, se puede comprobar que la señal con la que se está trabajando (SÍ4) tiene un total de 205 tramas solapadas.

En la Figura 34 se puede ver la señal (SÍ4) segmentada en tramas de 1024 muestras consecutivas (103 tramas).

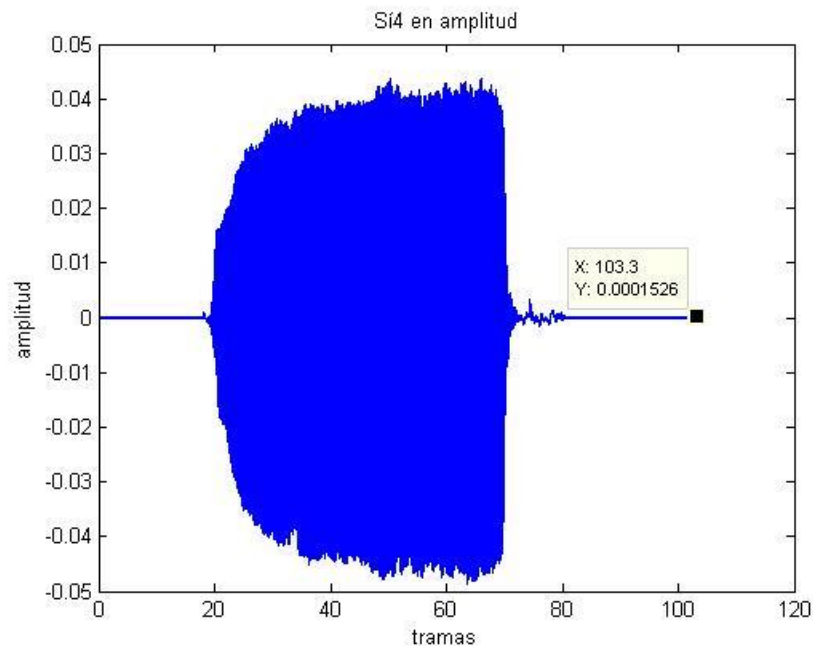


Figura 34: Nota (SÍ) segmentada en tramas consecutivas de 1024 muestras (103 tramas)

Por lo que como se puede observar, la señal (SÍ4) segmentada en tramas consecutivas tiene un total de 103 tramas que son 206 tramas solapadas, es decir el doble, quedando así estipulada la relación que hay entre el número de tramas consecutivas y el de tramas solapadas al 50%.

El objetivo de segmentar la señal era de trabajar con tramas cortas de la señal que nos permitiesen llevar a cabo un procesamiento más rápido y de esta forma llevar un coste computacional inferior. Pero el siguiente problema que se nos vuelve a presentar teniendo en cuenta que el algoritmo se fundamenta en el carácter periódico de la función de autocorrelación es que segmentando la señal no se consigue que la nota sea completamente periódica. A sabiendas de que las señales sonoras están formadas por tramas de silencio y

tramas sonoras; por lo que se nos hace la idea de que, para aplicar el algoritmo en condiciones óptimas, se deberá determinar de la señal qué tramas son sordas y cuáles son las sonoras.

La Figura 35 muestra el esquema que se va a seguir a partir de ahora. Nuestra implementación va a consistir en cinco pasos. El primer paso que se ha llamado “paso cero” no es concretamente del algoritmo CMNDF, pero es imprescindible para conocer las tramas sonoras ya que a éstas se aplicarán el algoritmo a implementar, a partir del paso 1 (función de autocorrelación) comenzará la implementación propia del algoritmo CMNDF.

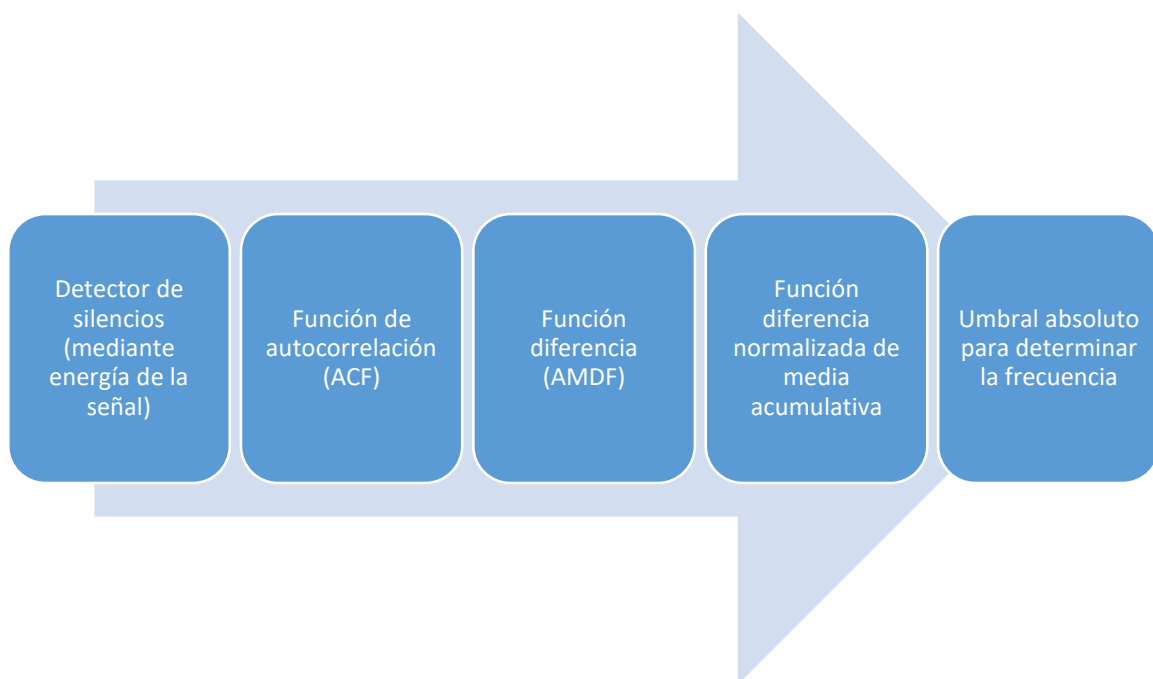


Figura 35: Esquema para la implementación del algoritmo de estimación de pitch

- Paso cero: energía (sonoro/silencio)

Con la distribución de energía podremos conocer las tramas sonoras y las tramas de silencio con el fin de quedarnos solo con las tramas sonoras a las que aplicaremos nuestro algoritmo. La distribución de energía de la señal normalizada por el número de muestra de la señal de entrada viene descrita en la ecuación 45:

$$Ener = \frac{1}{N} \sum_{k=1}^{ncol} |x|^2 \quad (45)$$

Donde $Ener$ es la energía de la señal, x es la señal de entrada, N es el número de muestras de la señal de entrada, $(k = 1, \dots, ncol)$ es el intervalo del número de tramas totales de la señal.

En la Figura 36 se puede observar la distribución de energía de la nota (Sí4)

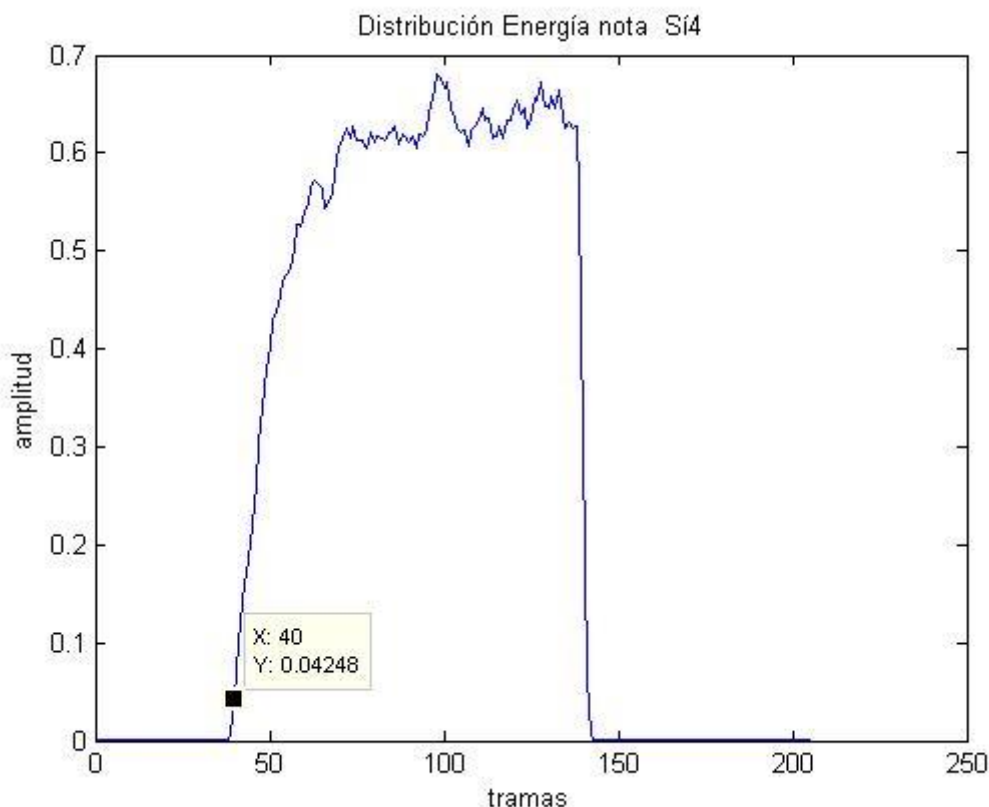


Figura 36: Distribución de energía de la nota (Si4) de 205 tramas

Como se puede observar las tramas sordas son las que tienen una amplitud de energía igual a cero o muy pequeña comparado con el máximo de energía calculado del

fragmento de la señal de entrada (Sí4); por lo se queda solo con las tramas sonoras, que como se puede observar en el posicionamiento del cursor, comienzan en la trama 40. Por ejemplo si se representan tramas inferiores a ésta (ejemplo la trama 36, 37, 38 y 39) se puede observar que la señal es aperiódica, tal como muestra la Figura 37:

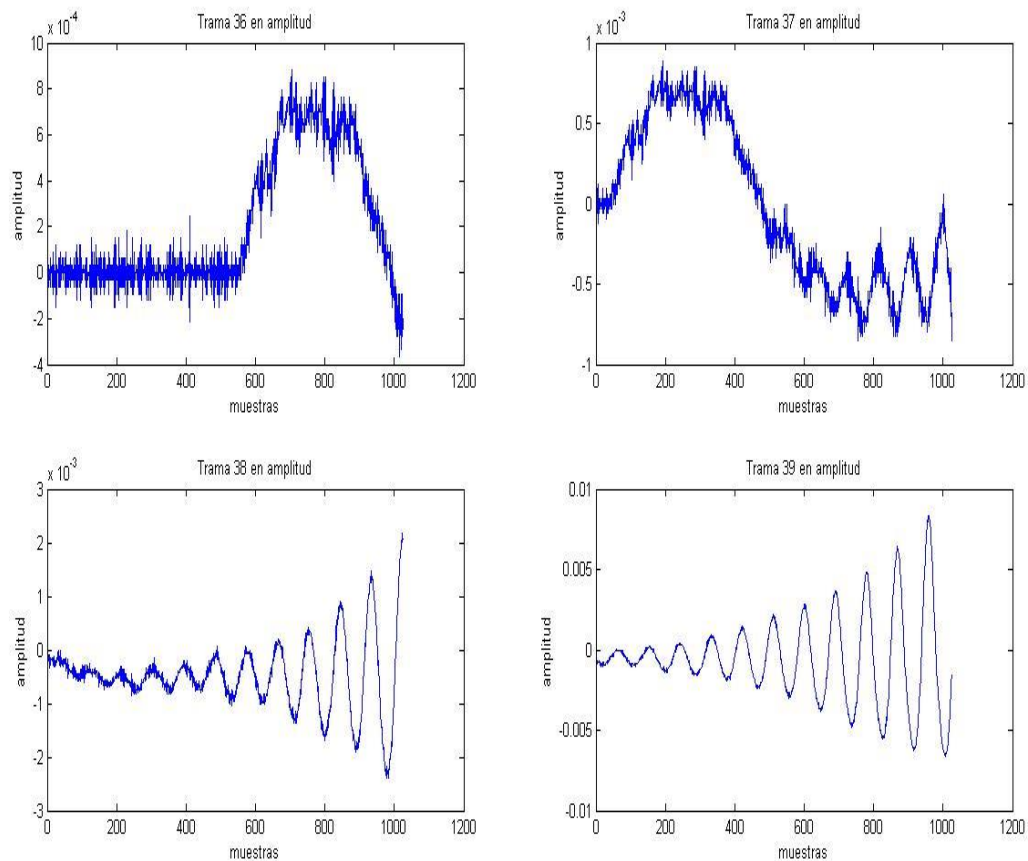


Figura 37: Representación de tramas de silencio (tramas 36, 37, 38 y 39) en muestras (1024 muestras)

Mientras que si representan algunas tramas sonoras se puede observar que efectivamente empiezan a ser periódicas a partir de la trama 40. En la Figura 38 se puede ver representadas las tramas (40, 41, 42, 43, y 44):

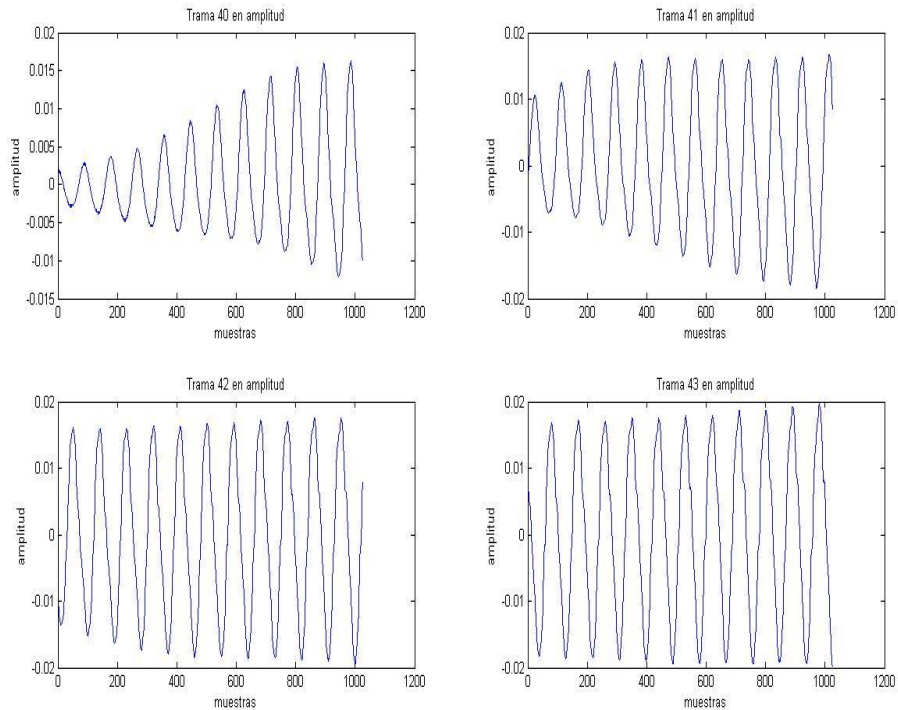


Figura 38: Representación de tramas de sonoras (tramas 40, 41, 42 y 43) en muestras (1024 muestras)

Más adelante cuando se realice la evaluación del algoritmo se verificará que al añadir ruido a nuestra base de datos que se supone sin ruido, será difícil diferenciar a simple vista mediante la distribución de energía las tramas sonoras de las tramas de silencio, porque éstas últimas ganarán amplitud de energía. Este ejemplo se puede ver en la Figura 39.

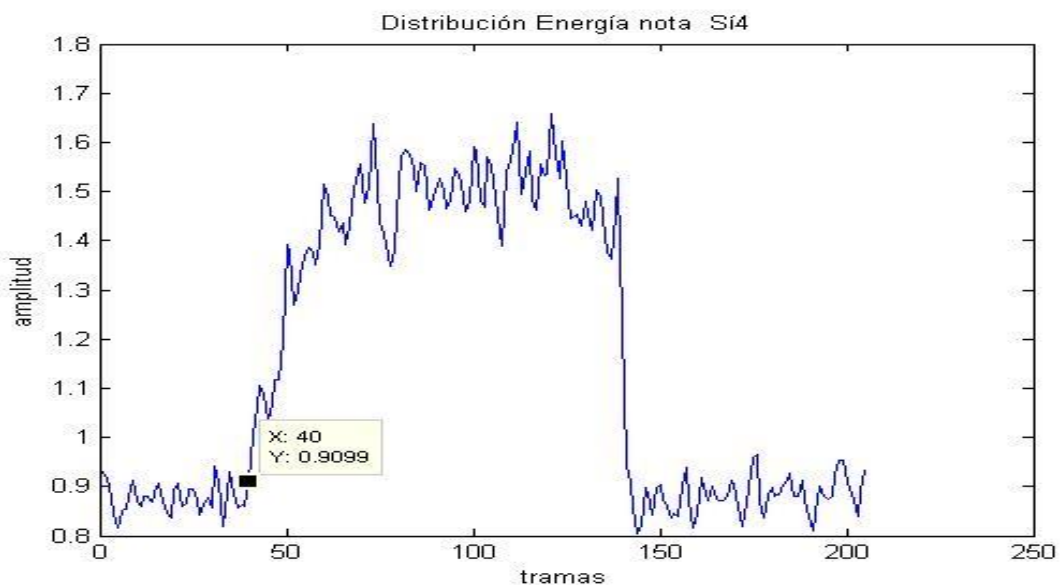


Figura 39: Distribución de energía nota Sí4 de 205 tramas con SNR (- 5db)

Como se puede observar, se hace difícil vislumbrar que a partir de la trama 40 (donde está situado el cursor) comienzan las tramas sonoras si se tiene en cuenta que tal como se puede observar hay tramas inferiores que tienen mayor energía que ésta. Para evitar estos altibajos de energía y para solucionar este problema se ha decidido dividir las tramas de 1024 muestras en pequeñas tramas de cinco muestras, de esta forma lo que se hace es agrupar la señal en su conjunto en pequeñas subsecciones del mismo valor, con la ventaja fundamental de que en lugar de tener 1024 valores distintos de energía tendríamos la quinta parte y de esta forma se hace más fácil el hecho de escoger un umbral que permita diferenciar entre sonoro y sordo. Como desventaja se pueden producir pequeños errores al suponer que las tramas vecinas a la trama a analizar tienen el mismo valor. Sin embargo, el error no llega a ser considerable. Con grupos de 5 tramas se llega al equilibrio entre eficacia a la hora de escoger el umbral (sonoro/sordo) y error. En la Figura 40 se puede observar que con la distribución de energía dividida en sub-tramas de cinco muestras se puede diferenciar claramente las tramas sonoras de las sordas y así fijar un umbral.

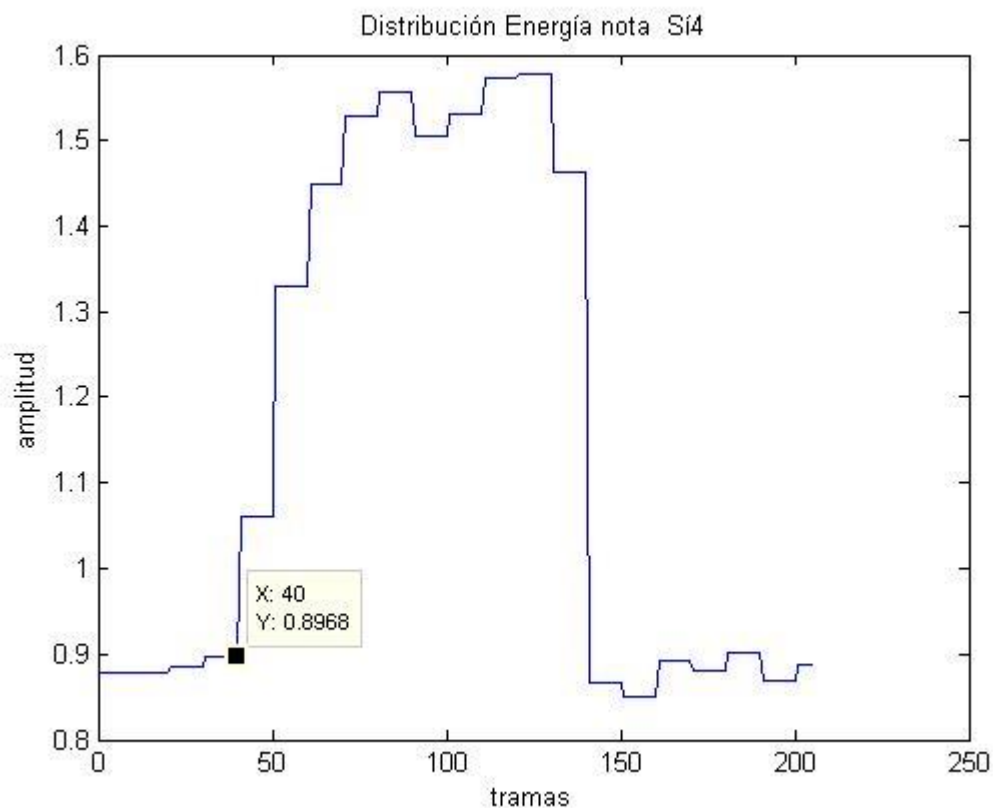


Figura 40: Distribución de energía nota S4 de 205 con SNR (- 5db), con tramas de 1024 muestras divididas en pequeñas tramas de 5 muestras.

Después de definir mediante la distribución de energía las tramas sonoras y sordas, y con el objetivo de trabajar con tramas cortas sonoras, se tomará por ejemplo la trama 70 como señal de entrada en nuestra implementación a partir de los siguientes pasos propios del algoritmo. La Figura 41 representa la trama 70 de 1024 muestras:

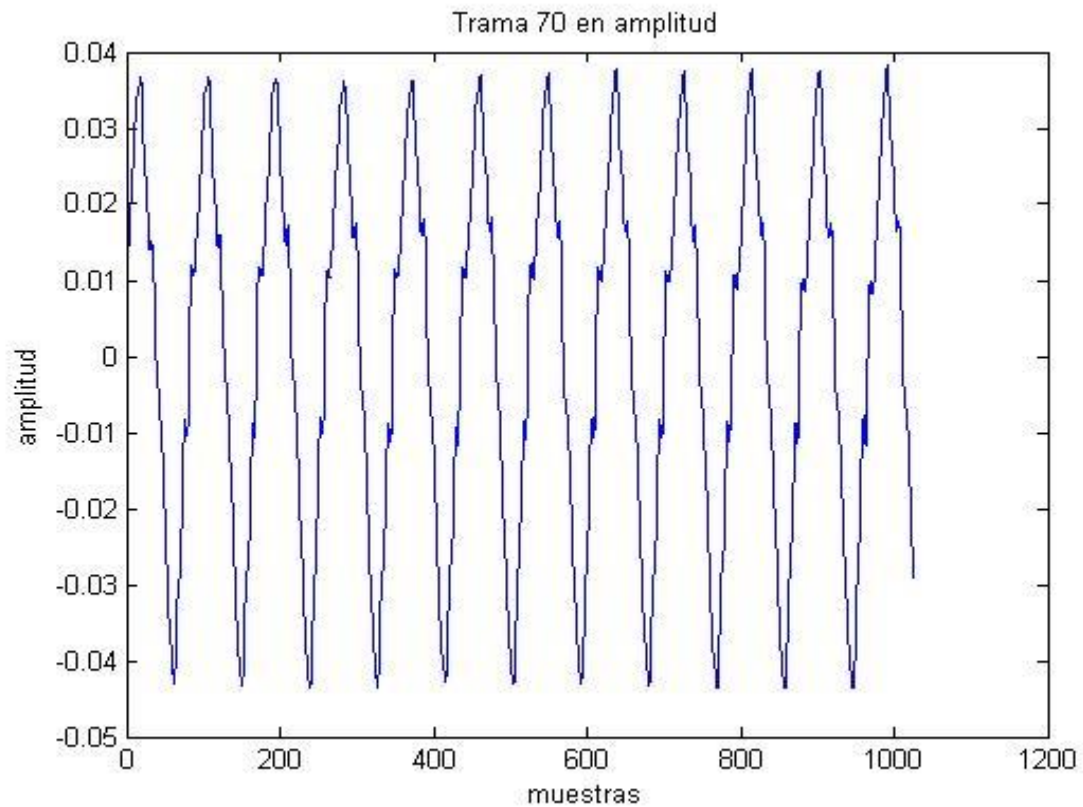


Figura 41: Trama 70 en muestras

A continuación, se va a detallar la realización del algoritmo CMNDF formado por cuatro pasos que se han implementado íntegramente en este TFG para poder cumplir con los objetivos anteriormente marcados.

- **Paso 1. Función de Autocorrelación (ACF)**

Una de las utilidades de la función de autocorrelación es que a través de ella se puede encontrar patrones que se repiten en una misma señal como por ejemplo el periodo que es la información que a nosotros nos interesa. La función de autocorrelación compara una señal con su réplica desplazada τ muestras. Retomando la expresión de la autocorrelación sería:

$$r_t(\tau) = \sum_{j=t+1}^{t+W} x_j x_{j+\tau},$$

Donde r_t es la función de autocorrelación que representa la multiplicación muestra por muestra en tiempo entre una señal x_j por su réplica $x_{j+\tau}$ retardada τ muestras. Siendo la señal x_j enventanada con una ventana temporal de W muestras de longitud.

En nuestro caso, en particular la autocorrelación será la multiplicación dentro de una determinada ventana de la señal x_j que corresponde a la trama 70 por su réplica desplazada $x_{j+\tau}$, siendo este desplazamiento el número de muestras de la trama 70 (1024 muestras). Por lo que es importante señalar que la autocorrelación es periódica siendo su periodo el primer armónico situado en un pico distinto de cero.

La Figura 42 muestra la representación de la función de autocorrelación en muestras de la trama 70 de la nota Sí4:

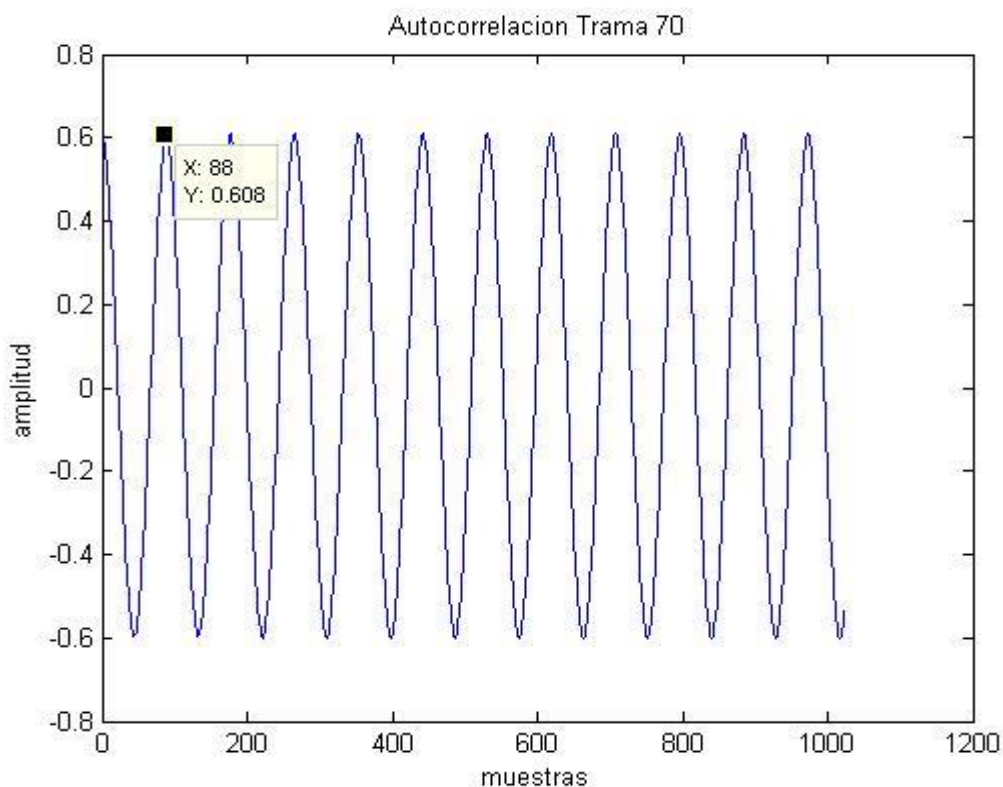


Figura 42: Autocorrelación de la trama 70 en muestras

Se puede observar de la Figura 42 que la frecuencia fundamental está situada en la muestra 88, por lo que haciendo una transformación en tiempo sería:

$$muestras * \frac{1}{f_s} = 88muestras * \frac{1seg}{44100muestras} \approx 0.00199seg = 1.99mseg$$

En la Figura 43 se puede observar mejor la representación de la función de autocorrelación de la trama 70 en tiempo.

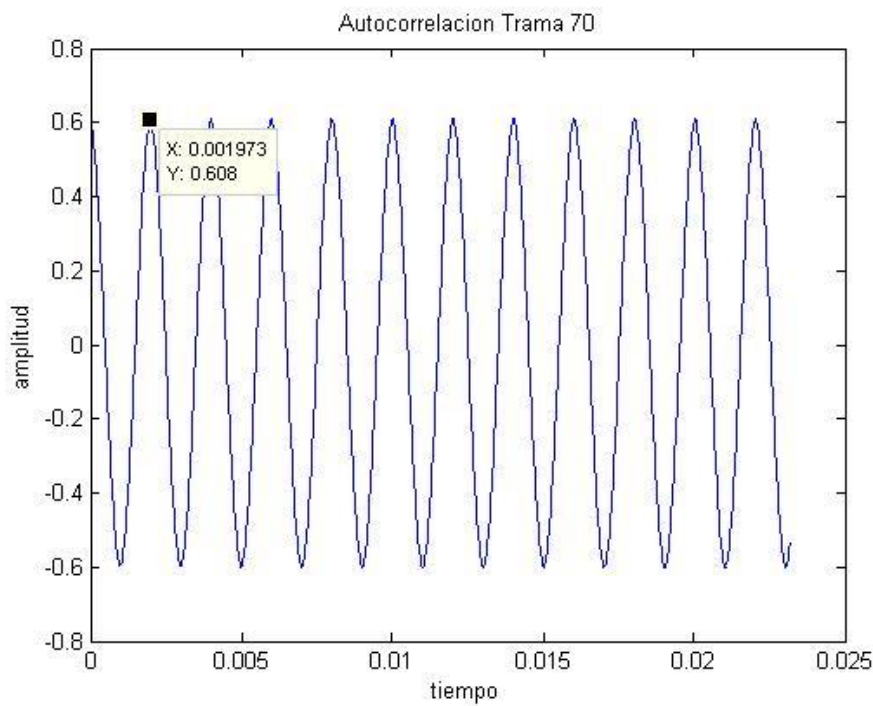


Figura 43: Autocorrelación de la trama 70 en segundos

Como se puede observar en la Figura 43, el primer armónico de la trama 70 se da en el instante 0.001973 segundos. Por tanto, ya se sabe que el periodo de la trama según el cálculo de la función de autocorrelación es de unos 0.001973 segundos y si hacemos la inversa de la ecuación que describe la relación periodo y frecuencia fundamental podremos deducir que la frecuencia fundamental de la trama se rige por la siguiente ecuación:

$$f_{fundamental} = \frac{1}{periodo} = \frac{1}{0.001973} \approx 506.8 \text{ Hz} \quad (46)$$

Teniendo en cuenta la tabla 7, se puede ver que la nota (Sí4) cuya trama (trama70) se está realizando la autocorrelación, su frecuencia fundamental ideal (la calculada con la

FFT) es de 499.2 Hz con lo que se puede comprobar que $506.8 \text{ Hz} \neq 499.2 \text{ Hz}$. Aunque se pueden considerar próximos; pero el paso de la autocorrelación comete un error absoluto de:

$$error_{absoluto} = (f_{ideal} - f_{estimada}) = (499.2 - 506.8) \text{ Hz} = 7.6 \text{ Hz} \quad (47)$$

- **Paso 2. Función de diferencias (AMDF)**

Al igual que la función de autocorrelación anterior esta función calcula el periodo fundamental de la señal con la diferencia de que la función de autocorrelación es un tren de armónicos en forma de picos, mientras que la función de diferencia es invertida; es decir sus armónicos se sitúan en los valles, donde el primer valle distinto de cero marcará el periodo de la trama. Retomando la expresión de la función de diferencias sería:

$$d_t(\tau) = 2[r_t(0) - r_t(\tau)]$$

Donde d_t es la función de diferencias que representa el doble de la diferencia entre la autocorrelación en el origen $r_t(0)$ de una señal x_j y la autocorrelación $r_t(\tau)$ de la misma señal muestra por muestra en tiempo retardada τ muestras. Siendo la señal x_j enventanada con una ventana temporal de W muestras de longitud.

En nuestro caso, en particular la función de diferencias será el doble de la diferencia entre la autocorrelación de la señal x_j correspondiente a la trama 70 cuando es máxima (autocorrelación en cero) y la autocorrelación con un retardo de τ muestras; siendo este desplazamiento el número de muestras de la trama 70 (1024 muestras). Por lo que es importante señalar que la función de diferencias es periódica siendo su periodo el primer armónico situado en un valle distinto de cero.

La Figura 44 muestra la representación de la función de diferencias en muestras de la trama 70:

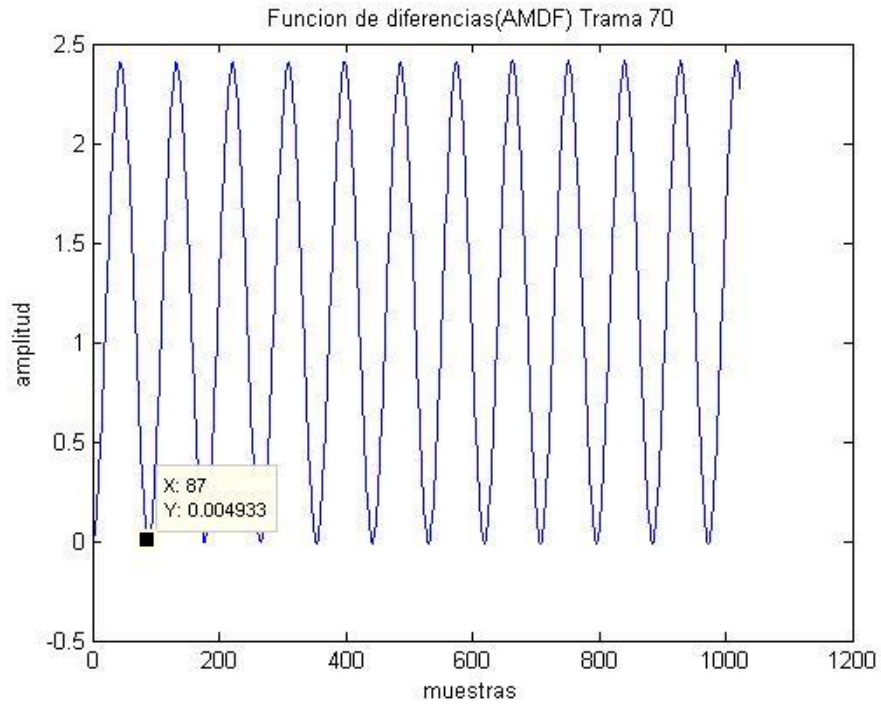


Figura 44: Función de diferencias de la trama 70 en muestras

Podemos observar de la Figura 43 que el primer armónico está situado en la muestra 87, por lo que haciendo una transformación en tiempo sería:

$$muestras * \frac{1}{fs} = 87muestras * \frac{1seg}{44100muestras} \approx 0.00197seg = 1.97 msec$$

En la Figura 45 se puede observar mejor la representación de la función de diferencias de la trama 70 en tiempo.

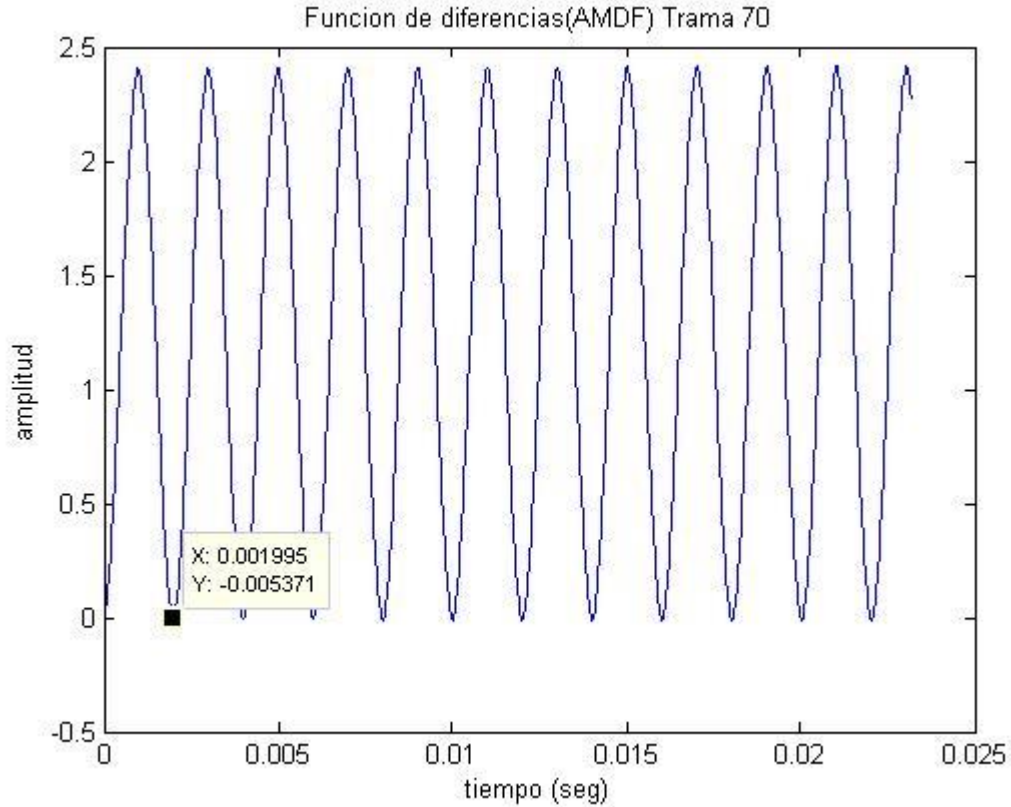


Figura 45: Función de diferencias de la trama 70 en segundos

Como se puede observar en la Figura 45, el primer valle de la trama 70 se da en el instante 0.001995 segundos. Por tanto, ya se sabe que el periodo de la trama según el cálculo de la función de diferencias es de unos 0.00195 segundos y si se hace la inversa de la ecuación que describe la relación periodo y frecuencia fundamental podremos deducir que la frecuencia fundamental de la trama se rige por la siguiente ecuación.

$$f_{fundamental} = \frac{1}{periodo} = \frac{1}{0.001995} \approx 501.2Hz \quad (48)$$

Si nos vamos a la tabla 7, se ve que la nota (S4) cuya trama (trama70) se está realizando la función de diferencias su frecuencia fundamental ideal (la calculada con la FFT) es de 499.2 Hz, con lo que se puede observar que $501.2 Hz \neq 499.2 Hz$. Aunque se pueden considerar próximos; pero el paso de la función de diferencias comete un error absoluto de:

$$error_{absoluto} = (f_{ideal} - f_{estimada}) = (499.2 - 501.2)Hz = 2 Hz \quad (49)$$

Se puede observar que este paso mejora en cuanto al error cometido en relación al del primer paso (el error de 2Hz, es mejor que el error de 7.6 Hz). Por lo que hay una mejora de (7.6Hz -2Hz = 5.6 Hz).

- **Paso 3. Función de diferencias normalizada de media acumulativa (CMNDF)**

Se debe prestar especial atención a esta función porque generalmente a partir de ella se puede estimar el periodo fundamental con un porcentaje de error relativamente inferior (ver tabla 6). Esta función como se puede ver en la ecuación que la describe, depende de la función de diferencias arriba planteada, la diferencia entre ambas consiste en que ésta normaliza a la anterior dividiendo cada una de sus muestras por su media, que será cada vez más extensa conforme se incremente el retardo. También es importante señalar que una de las mejoras que incluye esta nueva función es que a diferencia de la función anterior que comenzaba en cero, ésta comienza en uno, por lo que tiende a permanecerse alto para pequeños retardos. Por tanto, el periodo se situará en el primer valle distinto de cero cuya amplitud es igual a la unidad. Retomando la expresión de la función de diferencias normalizada de media acumulativa sería:

$$d'_t(\tau) = \begin{cases} 1, & \tau = 0 \\ \frac{d_t(\tau)}{\left(\frac{1}{\tau}\right) \sum_{j=1}^{\tau} d_t(j)}, & \tau \neq 0 \end{cases}$$

Donde d'_t es la función de diferencias normalizada de media acumulativa que representa la división entre la función de diferencias $d_t(\tau)$ muestra a muestra por su media. Esta función tiene amplitud unidad en el origen y distinta para el resto de muestras de la señal.

La Figura 46, muestra la representación de la función de diferencias normalizada de media acumulativa en muestras para la trama 70 de la nota Sí4:

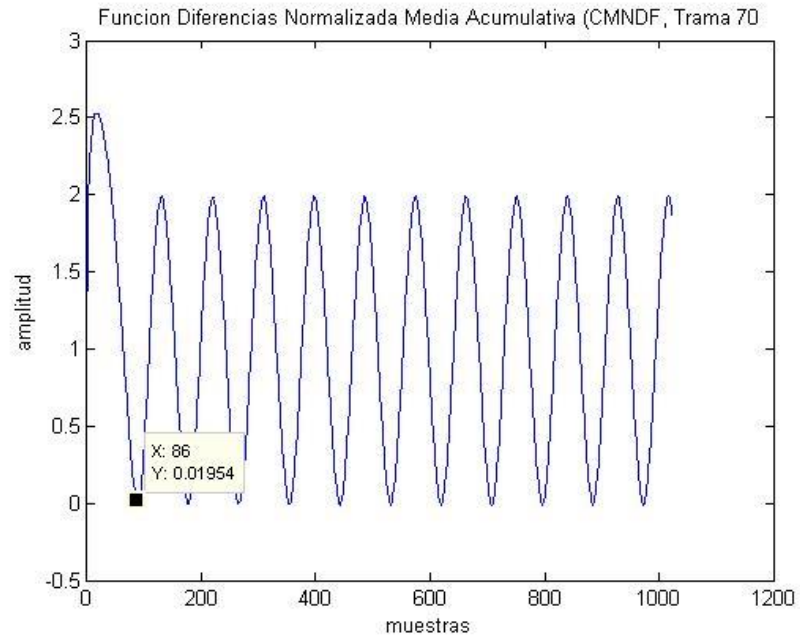


Figura 46: Función CMNDF trama 70 en muestras

Se puede observar de la Figura 46 que el primer valle está situado en la muestra 86, por lo que haciendo una transformación en tiempo sería:

$$\text{muestras} * \frac{1}{f_s} = 86\text{muestras} * \frac{1\text{seg}}{44100\text{muestras}} \approx 0.00195\text{seg} = 1.95 \text{ mseg}$$

En la Figura 47, se puede observar mejor la representación de la función de diferencias normalizada de media acumulativa de la trama 70 en tiempo:

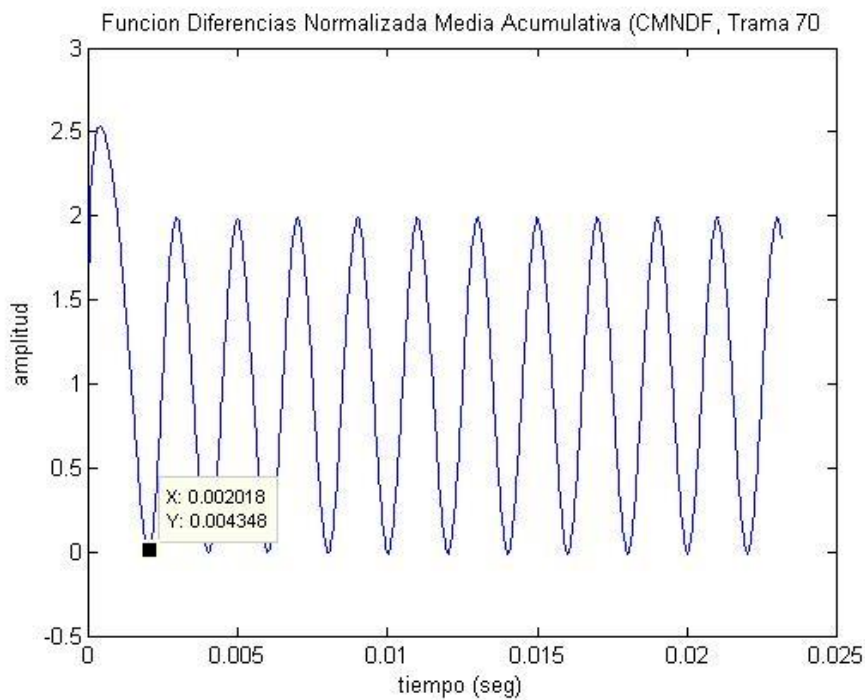


Figura 47: Función CMNDF de la trama 70 en segundos

Como se puede observar en la Figura 47, el primer valle de la trama 70 se da aproximadamente en el instante 0.002018 segundos. Por tanto, ya se sabe que el periodo de la trama según el cálculo de la función de diferencias es de unos 0.002018 segundos y si se hace la inversa de la ecuación que describe la relación periodo y frecuencia fundamental se puede deducir que la frecuencia fundamental de la trama se rige por la siguiente ecuación.

$$f_{fundamental} = \frac{1}{periodo} = \frac{1}{0.002018} \approx 495.5 \text{ Hz} \quad (50)$$

Si nos vamos a la tabla 7, se puede ver que la nota (Sí4) cuya trama (trama70) se realizando la función de diferencias su frecuencia fundamental ideal (la calculada con la FFT) es de 499.2 Hz con lo que se puede observar que $500 \text{ Hz} \neq 499.2 \text{ Hz}$. Aunque se pueden considerar próximos; pero el paso de la función de diferencias comete un error absoluto de:

$$error_{absoluto} = (f_{ideal} - f_{estimada}) = (499.2 - 500) \text{ Hz} = 0.8 \text{ Hz} \quad (51)$$

Con el error cometido se puede observar que conforme se avanza en los pasos que componen la implementación del algoritmo CMNDF realizado en este TFG, el error es cada vez menor y se asemeja a la frecuencia real. La mejora en este paso para la trama 70 respecto al segundo paso es de $(2 \text{ Hz} - 0.8 \text{ Hz} = 1.2 \text{ Hz})$.

- **Paso 4. Umbral**

El algoritmo que se está implementando, como se puede observar hasta el tercer paso lo que incluye es la estimación de periodicidad. El cuarto paso plantea un algoritmo de detección y selección de valles de la función CMNDF. Para esto los autores del algoritmo Yin [17] al que se sustenta este TFG sostiene que la búsqueda de valles se debe regir por un umbral absoluto que en su bibliografía lo sitúan en un valor igual a 0.1 de tal forma que los valles que superen este umbral sean seleccionados como candidatos a ser frecuencia fundamental y el menor de todos sea el seleccionado como la frecuencia fundamental. Por lo que este paso se fundamenta en encontrar los mínimos de la función de diferencias normalizada de media acumulativa. Un apunte muy importante es que después de realizar varias pruebas con distintas notas, se ha comprobado que el umbral estático que los autores

plantean (de valor igual a 0.1) no siempre selecciona la frecuencia fundamental correcta. Si se considera el siguiente ejemplo:

Considerando un umbral estático de 0.1 y suponiendo que el primer valle que nos encontramos en la función CMNDF tiene una amplitud de 0.05, mientras que el segundo valle que nos encontramos tiene una amplitud de 0.03, en este caso y según la regla definida anteriormente de escoger el valle de menor valor por debajo del umbral estático de 0.1, se escogería el segundo valle como frecuencia fundamental. Esta decisión sería errónea, ya que la frecuencia fundamental vendría determinada por el primer valle. Este planteamiento sería un tanto erróneo y desde este punto de vista el algoritmo es propenso a cometer errores de octava al seleccionar como frecuencia fundamental los valles más acusados.

Una vez analizado el problema tras llevar a cabo una evaluación exhaustiva de los resultados, se comprobó que se producían errores al determinar la frecuencia fundamental y esto se debía principalmente a que el umbral estático no especificaba de forma exacta el valle que determinaría la frecuencia fundamental. Es por ello que se planteó la idea de utilizar un umbral dinámico que se ajustase a cada una de las tramas analizadas.

Para que la última fase del proceso funcione de forma correcta y se obtengan resultados fiables es necesario que se cumplan 2 condiciones.

Primera condición

Que se analicen aquellas tramas que se consideren sonoras y de esta forma se evite un análisis innecesario de las tramas sordas. Para ello se hace uso del paso 0 que permite, mediante el cálculo de la energía, determinar entre sonoro y sordo.

Problemática del umbral de energía: Tras varias pruebas hemos visto que, si este umbral es muy grande, el algoritmo implementado tomará a muchas tramas de la nota como silencio cuando en realidad son tramas sonoras (falsos negativos). Mientras que si este umbral es muy pequeño tomará a muchas tramas como sonoras cuando en realidad son silencio (falsos positivos).

Por tanto, hemos decidido tras varios experimentos que el umbral de energía no sea constante para todas las señales de entrada, sino que se reajuste en función del 10% de la

diferencia entre el máximo y el mínimo valor de energía de cada nota o señal. Este umbral dinámico se adapta al comportamiento de cada una de las señales. El hecho de elegir un valor mediante la función definida, es para asegurar que estamos eligiendo un valor que se encuentre por encima de la señal considerada como silencio y de esta forma se pueda segmentar entre lo que no es silencio y lo que sí. Para una explicación más completa, si a una señal le añadimos una relación señal/ruido del orden de -5dB y analizamos su comportamiento podremos observar que en el caso de que haya ruido la diferencia de energía entre lo que es silencio y no, se va haciendo cada vez más insignificante. Por esa razón se ha pensado en trabajar con un umbral que se encuentre al 10 % de la distancia que existe entre el valor máximo y el mínimo, tal como indica la siguiente ecuación:

$$corte = 0.1(maximo_{energia} - minimo_{energia}) \quad (52)$$

Donde *corte* es el umbral de energía, *maximo_{energia}* es el valor máximo de la energía de la señal, y *minimo_{energia}* es el valor mínimo de la energía de la señal; por tanto, todas las tramas mayores a este umbral de energía serán consideradas como tramas sonoras. En la Figura 48 podemos observar cómo se seleccionan las tramas sonoras dentro de una determinada nota (S14) con diferentes presentaciones de la distribución de energía anteriormente comentadas:

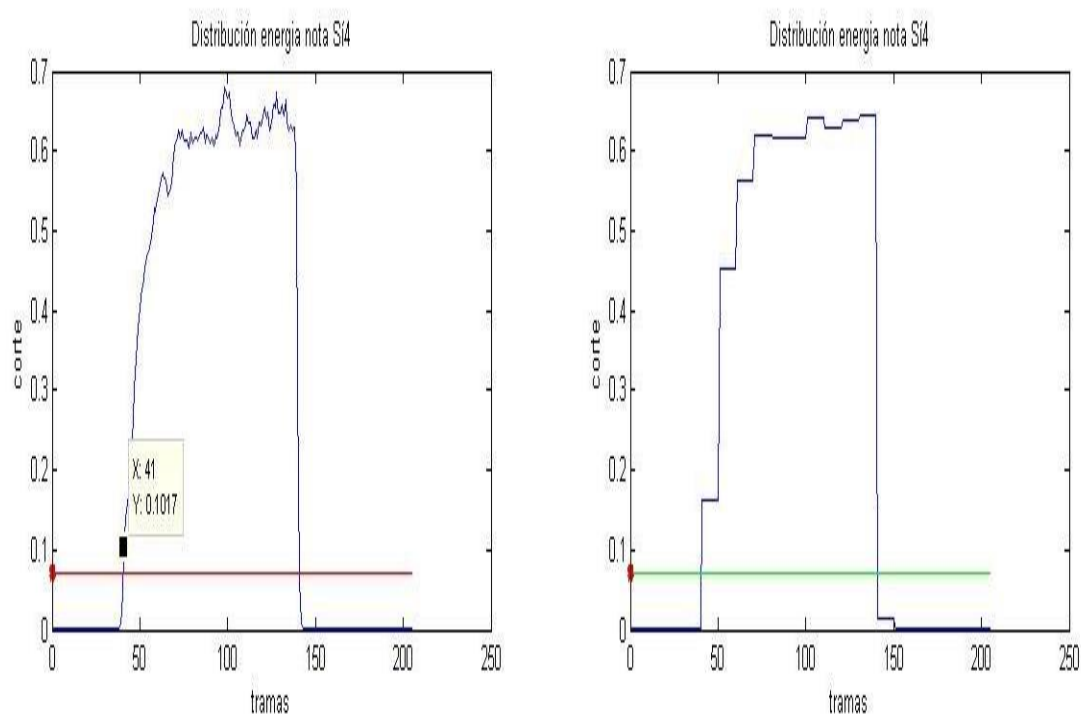


Figura 48: Selección de tramas sonoras mediante la distribución de energía de la nota S4. A la izquierda se representa la distribución de energía de la nota S4. A la derecha se representa la distribución de la energía de la nota S4 dividida en pequeñas tramas.

Para dejarlo más claro al lector, el objetivo de conocer las tramas sonoras en este punto, es porque al ser ya el último paso del algoritmo y el que devuelve la frecuencia fundamental de cada nota, se necesitará que devuelva la información (la frecuencia fundamental estimada) solo de las tramas sonoras; por lo que en definitiva en el apartado anterior se plantea eliminar los silencios por selección del umbral de energía.

Segunda condición

Si el objetivo es seleccionar la muestra (situada en un valle de la función de diferencias normalizada) que corresponda a cualquiera de las frecuencias de la flauta, entonces la segunda condición que es aplicable exclusivamente a las tramas que han sido seleccionadas como sonoras mediante la primera condición del umbral de energía, debe consistir en restringir la señal obtenida en la función CMNDF para quedarnos únicamente con aquellas muestras dadas en tiempo, que al pasarlas a frecuencia nos de valores propios de la frecuencia fundamental que pueden tener las distintas notas tocadas por la flauta (ver tabla 1).

Por lo tanto, una vez que se tiene claro cuáles son las condiciones que permiten obtener un umbral efectivo, se lleva a cabo el desarrollo que permite determinar el umbral dinámico que se adaptara a cada una de las tramas analizadas para determinar la frecuencia fundamental en cada una de ellas, siempre y cuando teniendo en cuenta que únicamente se analizan las tramas sonoras y que de alguna forma se restringe el análisis para poder seleccionar únicamente frecuencias fundamentales que correspondan a las notas de la flauta.

Para determinar el umbral dinámico con el cual se va a seleccionar la muestra temporal de la señal CMNDF que nos va a permitir determinar la frecuencia fundamental en cada trama, se aplicará un umbral dinámico con un margen del 25%.

Este umbral se basa en el siguiente proceso interactivo:

1º.- En primer lugar, nos quedamos con el valor mínimo de la función CMNDF, siempre y cuando ese valor pueda determinar una frecuencia acorde a las notas de la flauta y no supere el rango, ni este por debajo.

2º.- El siguiente paso es aplicar un 25% de margen que permite analizar las muestras cercanas al valle seleccionado y quedarnos con el menor. Esto es útil, porque a la hora de escoger el valor mínimo en el primer paso no existe ninguna garantía de que el valle seleccionado sea el primero, en otras palabras, el correcto.

Dependiendo de si el valor mínimo es positivo o negativo se tendría que seguir el proceso de la siguiente forma:

3º.- Si esta amplitud es negativa; entonces que el algoritmo seleccione solo las muestras cuyos valles tengan una amplitud menor que el umbral estático por 0.75 (25% por debajo); y de entre estas muestras que seleccione la primera de todas ellas que será la que corresponda a la frecuencia fundamental en la que se emita una nota de la flauta (el primer valle).

4º.- Si esta amplitud es positiva; entonces que el algoritmo seleccione solo las muestras cuyos valles tengan una amplitud menor que el umbral estático por 1.25 (25% por

arriba); y de entre estas muestras que seleccione la primera de todas ellas que será la que corresponda a frecuencia fundamental en la que se emita una nota de la flauta.

Un ejemplo numérico con la nota Sí4 lo tenemos como sigue:

A) El valor mínimo de la función CMNDF es:

Umbral estático = 0.0867 de amplitud (valor mínimo)

B) Como el umbral estático es positivo se ejecutará el cuarto punto donde las muestras seleccionadas como posibles candidatos serían Candidatos= [87 88 89 90 91] muestras; y de entre estas muestras nos quedamos con la primera que es la muestra 87 y es la que nos asegura que nos encontramos en el primer valle, que corresponde a la frecuencia de alguna nota de la flauta.

Por lo consiguiente el periodo fundamental será:

$$periodo_{CMNDF} = muestras * \frac{1}{f_s} = 87 * \frac{1}{44100} = 0.001972seg \quad (53)$$

Y la frecuencia fundamental será:

$$f_{CMNDF} = \frac{1}{periodo_{CMNDF}} = \frac{1}{0.001972seg} = 500Hz \quad (54)$$

Un ejemplo ilustrativo que acompaña a este procedimiento numérico se puede ver en la Figura 49.

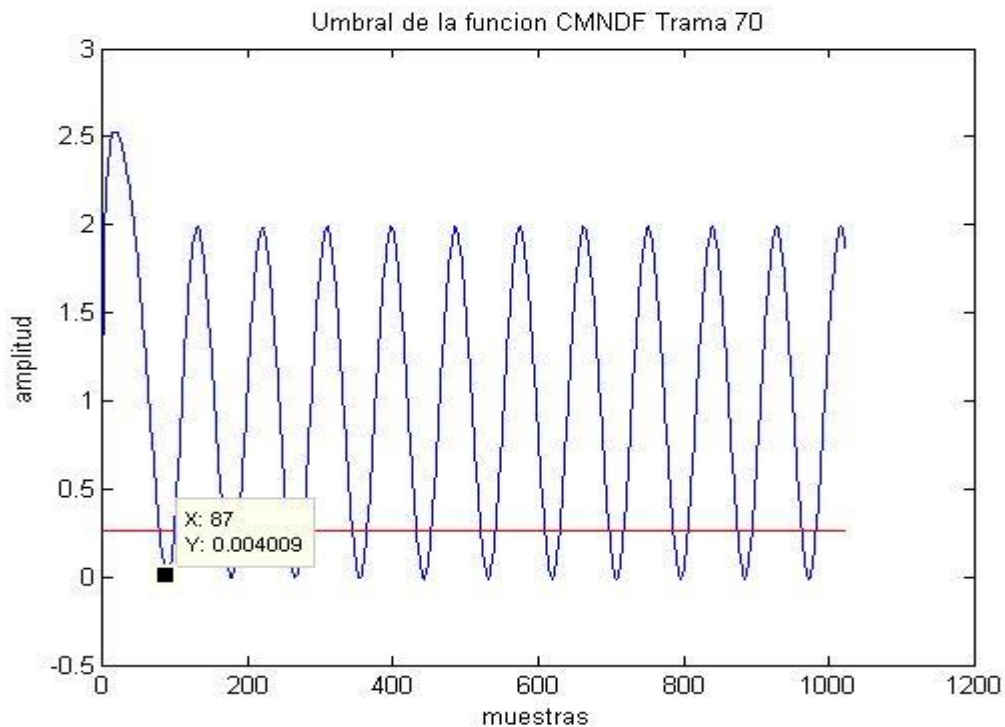


Figura 49: Selección de muestra por umbral

Una vez realizado los pasos de los que consta el algoritmo CMNDF para una única trama (en todos los ejemplos se ha trabajado con la trama 70 de la nota Sí4) obteniendo la frecuencia estimada, su correspondiente frecuencia fundamental, y siempre teniendo en cuenta las restricciones arriba indicadas; el siguiente paso es conocer la frecuencia fundamental de toda la nota y no de una sola trama como antes quedó demostrado.

Para obtener la frecuencia fundamental de todas las tramas que forman una nota se creará una máscara de tal forma que se asigne un peso o valor igual a cinco a aquellas tramas que superen el umbral de energía arriba explicado, es decir las tramas sonoras; mientras que tomaran un peso o valor igual a uno las que sean consideradas como silencio. Por tanto, para realizar el cálculo de todas las frecuencias fundamentales, se establecerá que al algoritmo de que realice el tercer y cuarto paso del algoritmo a todas las tramas que sean asignadas por un valor igual a cinco en la máscara hasta detectar un valor uno que pertenece a tramas de silencio, para así evitar los silencios. Tal como se puede observar en la Figura 50.

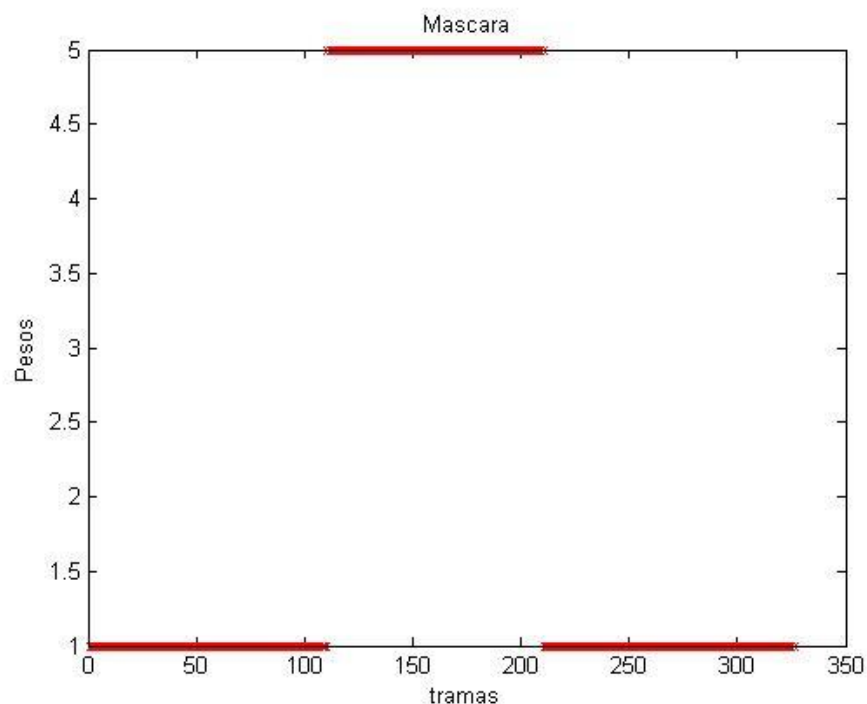


Figura 50: Tramas en las que se calculara la frecuencia fundamental

De esta forma ya conociendo las tramas sonoras, la selección de sus respectivas muestras en este intervalo es llevada a cabo bajo las condiciones arriba explicadas mientras

que a las muestras de las tramas de silencio se le asignan un valor igual a la frecuencia de muestreo (44100). Este ejemplo se puede observar en la Figura 51:

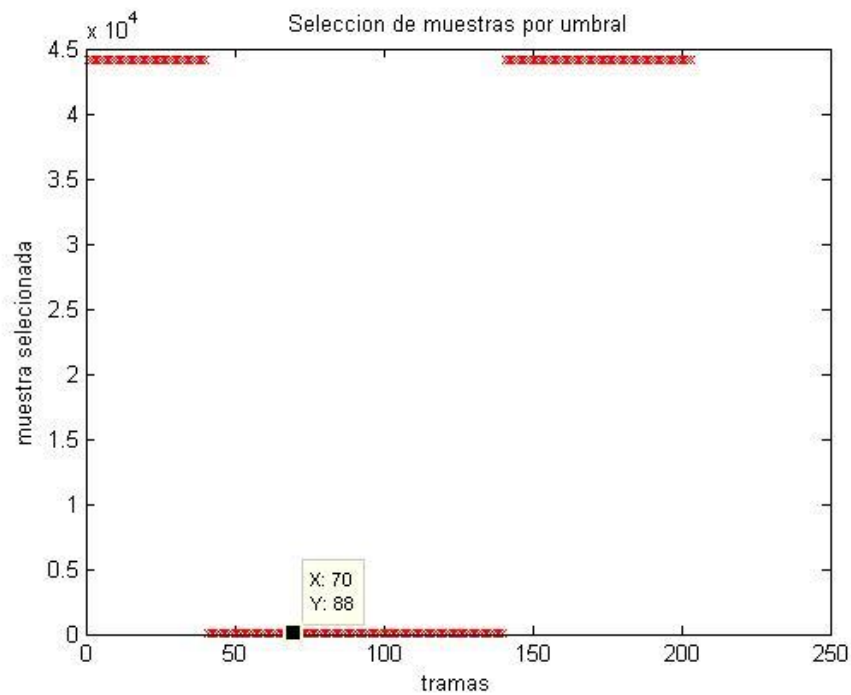


Figura 51: muestras donde se dan la nota vs tramas de la nota

Como se puede observar, a modo de ejemplo puntual se ha situado el cursor (punto marcado en negro) en un punto que relaciona la trama 70 con la que se ha estado trabajando anteriormente con su respectiva muestra seleccionada y se puede verificar que efectivamente la muestra donde el algoritmo estima que se dará la nota es la muestra 88 que pasando de muestras a frecuencia equivale a una frecuencia de 500 Hz aproximados. Cada una de las tramas tienen su correspondiente muestra, aunque no muy dispares entre sí; y en consecuencia tendrían cada una de estas tramas distintas frecuencias fundamentales. Si se hace un zoom a la Figura 51 se podrá observar como todas las tramas no tienen la misma muestra estimada tal como indica la Figura 52.

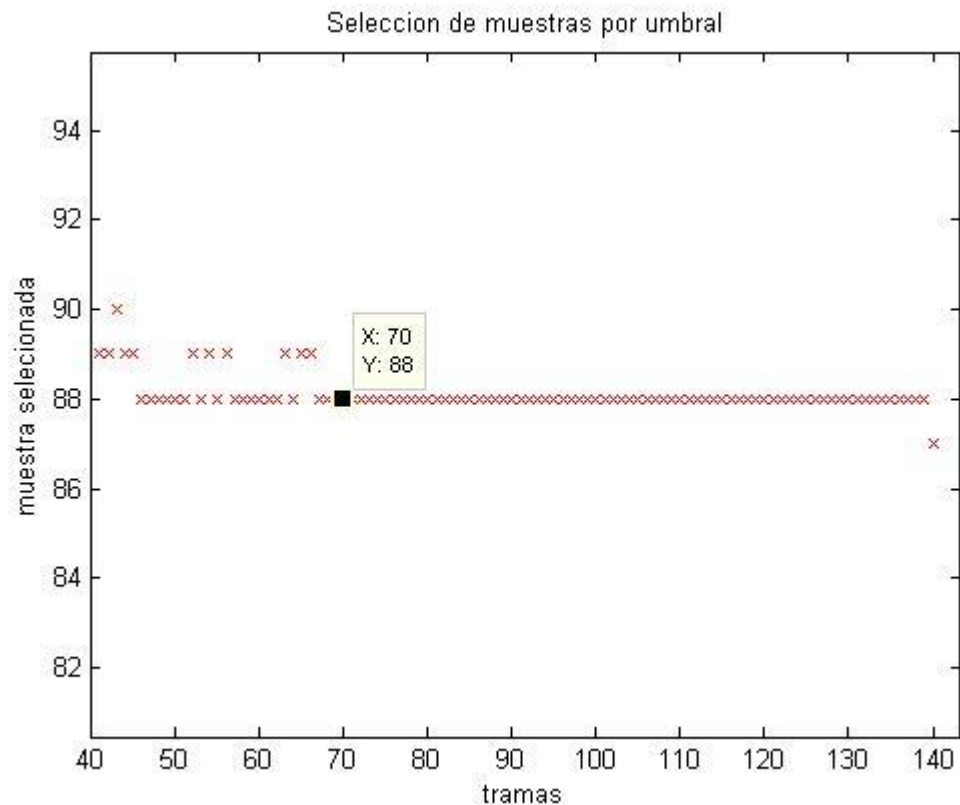


Figura 52: Zoom muestras donde se dan la nota vs tramas de la nota

En la Figura 53 se tiene la frecuencia fundamental que corresponde a la muestra seleccionada de la nota Sí4 con la que hemos trabajado anteriormente:

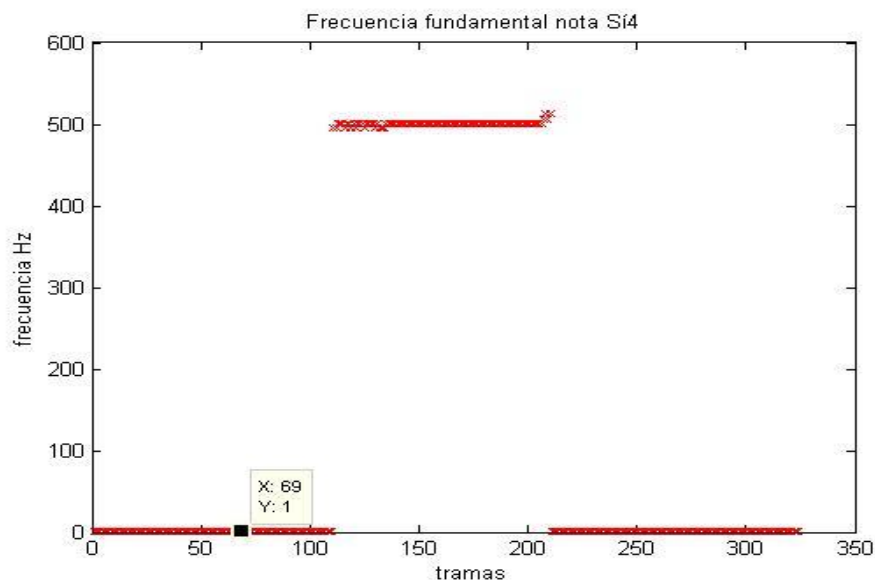


Figura 53: Frecuencia fundamental de todas las tramas sonoras

Como se puede observar fuera del intervalo sonoro la frecuencia vale uno, es decir es considerado silencio; mientras que dentro del intervalo sonoro de la nota Sí4 la frecuencia de las respectivas tramas es distinta de uno. También se puede observar que todas las tramas

sonoras no tienen la misma frecuencia. Si se hace un zoom a la Figura 53 se puede observar mejor cómo todas las tramas sonoras no tienen la misma frecuencia tal como indica la Figura 54.

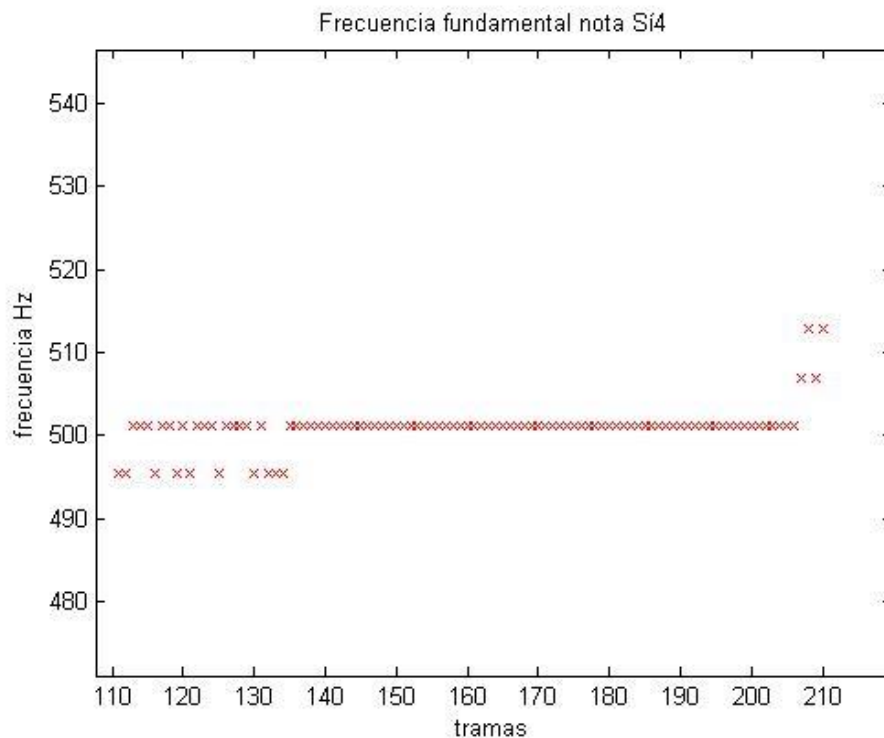


Figura 54: Zoom frecuencia fundamental de todas las tramas sonoras

La función que realiza todos los procesos arriba explicados es:

```
function [f0,salto,tsalto]= yin_Albert (x,fs)
```

La señal de entrada de esta función es:

- x: es la variable que contiene la señal de entrada
- fs: corresponde a la frecuencia de muestreo

La salida de esta función es:

- f0: es la variable que almacena la frecuencia fundamental de cada señal estimada por el algoritmo trama por trama.
- salto: corresponde a la longitud en muestras de cada trama
- tsalto: es el tiempo que dura la trama

Problemática de la función y_{in} Albert

Esta función devuelve la frecuencia fundamental de la nota trama por trama. Pero todas las tramas de la nota no tienen la misma frecuencia fundamental, lo que es un tanto incongruente si se trata de tramas correspondientes a la misma nota.

Para corregir esta incongruencia, nos surge la idea de obtener una frecuencia única para toda la nota, que de ahora en adelante se llamará ‘frecuencia justa’. Para llevarlo a cabo se han tenido dos consideraciones:

1º.- Realizar la media a la matriz de frecuencias que devuelve el algoritmo correspondiente a cada trama. Es decir, asignar a la nota la frecuencia media resultante.

2º.- Realizar la moda matemática a la matriz de frecuencias devuelta por el algoritmo correspondiente a cada trama. Es decir, asignar a la nota la frecuencia más repetitiva.

Después de realizar varias pruebas se ha llegado a la conclusión de que es mejor la segunda opción porque la media no es representativa, debido a que si por ejemplo existiera una frecuencia de alguna trama muy distinta o inferior a las demás ésta puede provocar que la media sea inferior a la frecuencia verdadera, mientras que esta situación se evita cuando simplemente se escoge la frecuencia más repetitiva.

En la Figura 55 (a) se puede observar la frecuencia justa de la nota La6 de la flauta dulce devuelta por el algoritmo cuando se realiza la media en comparación a la frecuencia real de la nota La6; y en 55 (b) se puede observar la frecuencia justa de la nota La6 que devuelve el algoritmo cuando se realiza la moda matemática en comparación a la frecuencia real de la nota La6. Donde se puede observar claramente que la mejor opción es determinar la frecuencia justa mediante la realización de la moda matemática.

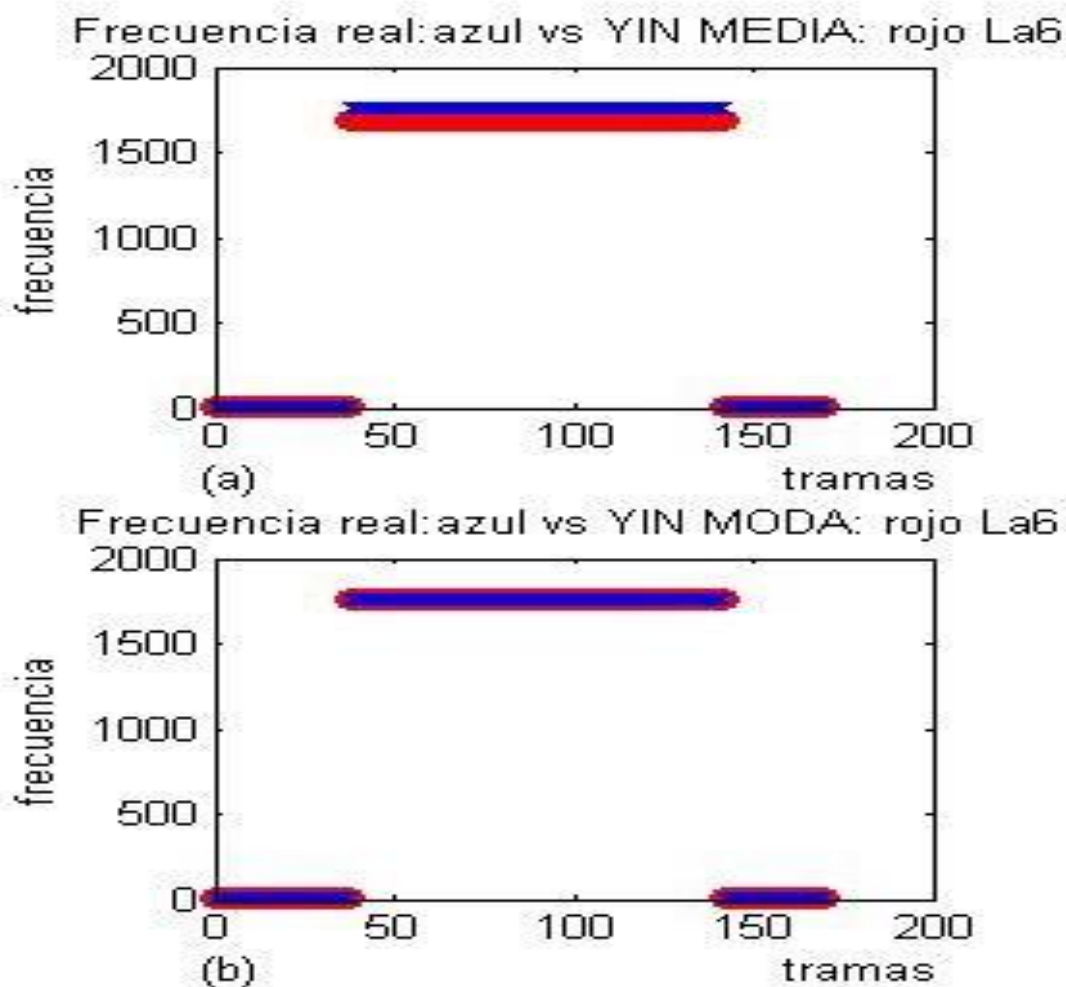


Figura 55: Función Yin ajustada con la media vs ajustada con la moda matemática. La Figura (a) y (b) muestran los resultados de calcular una la frecuencia fundamental única para todas las tramas que forman la nota La6 mediante la moda matemática y la media respectivamente.

Para realizar el proceso de asignar a todas las tramas sonoras la frecuencia más repetitiva (la moda matemática) como ilustra la Figura 55 se deben conocer la trama de inicio de nota (onset) y la trama de final (offset) de la nota. Como se ha llevado a cabo un cálculo de la energía para cada una de las tramas presentes en la señal, el hecho de determinar los inicios y los finales queda resuelto de forma rápida y sencilla. Estos van a marcar el intervalo de inicio y final de cada nota y se aplicará la moda matemática a este intervalo de tramas. La función que realiza el proceso de ajustar la frecuencia devuelta por el algoritmo es:

```
function [f0,inicios,finales,f0n,midi,midin]= freq_justa (f0)
```

La señal de entrada de esta función es:

- f_0 : es la frecuencia fundamental para cada trama devuelta por el algoritmo

La salida de esta función es:

- f_0 : frecuencia fundamental, pero la misma para todas las tramas de la nota.
- $inicios$: indica la trama de inicio de la nota
- $finales$: indica la trama de final de nota
- f_{0n} : variable que va a almacenar el valor de la frecuencia fundamental única por nota que resulta de realizar la moda matemática.

Llegados a este punto, lo último que queda para concluir el sistema implementado es transformar las frecuencias obtenidas a notación musical (MIDI). Para esto se utiliza la relación frecuencia y MIDI registrada en la tabla 5 de tal forma que para cada frecuencia de la tabla se crea un intervalo con un rango superior y un rango inferior. Siendo el rango superior:

$$rango_{superior} = f_{tabla} + \frac{\left(2^{\left(\frac{1}{12}\right)}\right)}{2} * f_{tabla} \quad (55)$$

Y el rango inferior:

$$rango_{inferior} = f_{tabla} - \frac{\left(2^{\left(\frac{1}{12}\right)}\right)}{2} * f_{tabla} \quad (56)$$

Por lo consiguiente se formaría un intervalo de decisión:

$$intervalo_{decision} = [rango_{inferior}, \dots, rango_{superior}] \quad (57)$$

De tal forma que si la frecuencia estimada por el algoritmo cae dentro del intervalo de alguna de las frecuencias de la tabla 5 se le asigna su correspondiente MIDI.

En la Figura 56 se puede observar la frecuencia estimada de la nota $Sí4$ en (a) mientras que en (b) su correspondiente midi. Y efectivamente el midi de la nota $Sí4$ es 71 por lo que se ha realizado de forma correcta la transcripción.

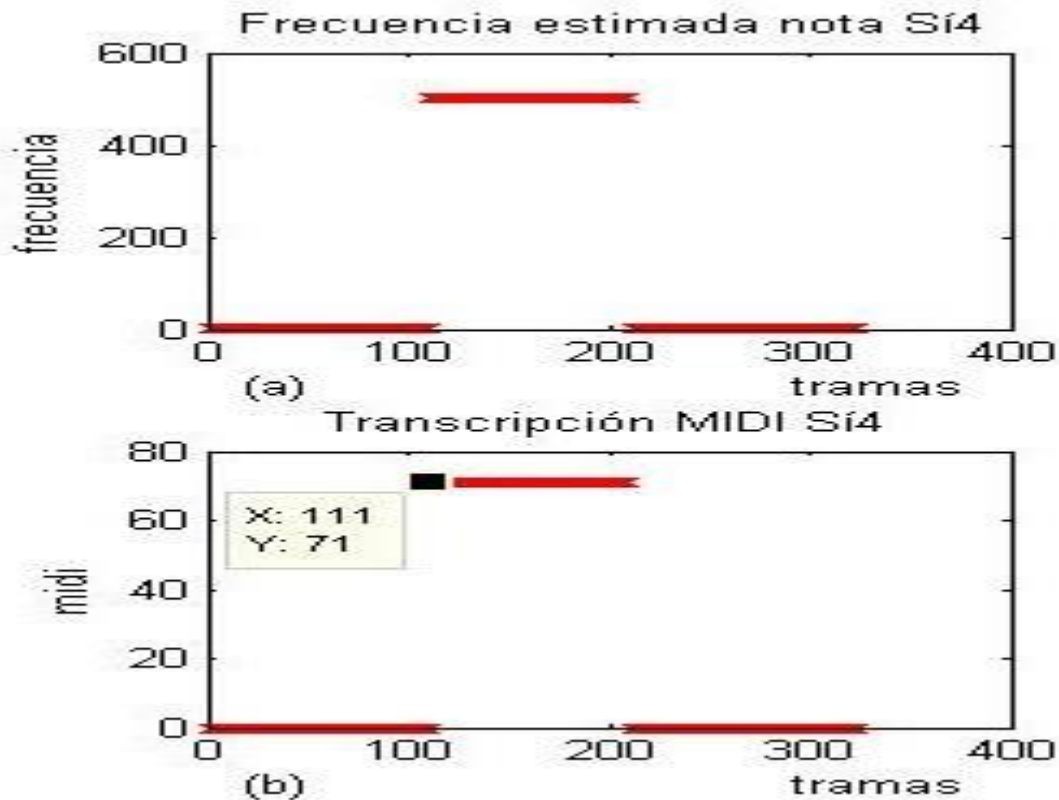


Figura 56: Frecuencia fundamental transformada a MIDI. La Figura (a) representa la frecuencia fundamental de la nota S4 estimada con el algoritmo implementado en este TFG y en la Figura (b) se representa la transformación a midi de esta frecuencia.

La función que realiza la transcripción de la frecuencia devuelta a notación musical MIDI es:

```
function [midi,notas]= freq_mid(f0)
```

La señal de entrada de esta función es:

- f0: es la frecuencia fundamental ya de la nota.

La salida de esta función es:

- midi: corresponde al midi que se le relaciona con la frecuencia estimada de la nota.
- notas: corresponde a la matriz de caracteres que almacena la nomenclatura correspondiente al midi de la nota en cuestión

El sistema implementado también es capaz de estimar las frecuencias y sus respectivos MIDI de una pista compuesta de más de una nota como muestra la Figura 57 se tiene en 57 (a) las frecuencias estimadas de una pista compuesta por tres notas

(Sol6, La6, La#6) y en 57 (b) sus midi's correspondientes. En 57 (c) las frecuencias estimadas de una pista compuesta por tres notas (Mi5, Fa5, Fa#5, Sol5, Sol#5) y en 57 (d) sus midi's correspondientes. En 57 (e) las frecuencias estimadas de una pista compuesta por tres notas (sí4, Do5, Do#5, Re5, Mi5, Fa5, Fa#5, Sol5) y en 57 (f) sus midi's correspondientes.

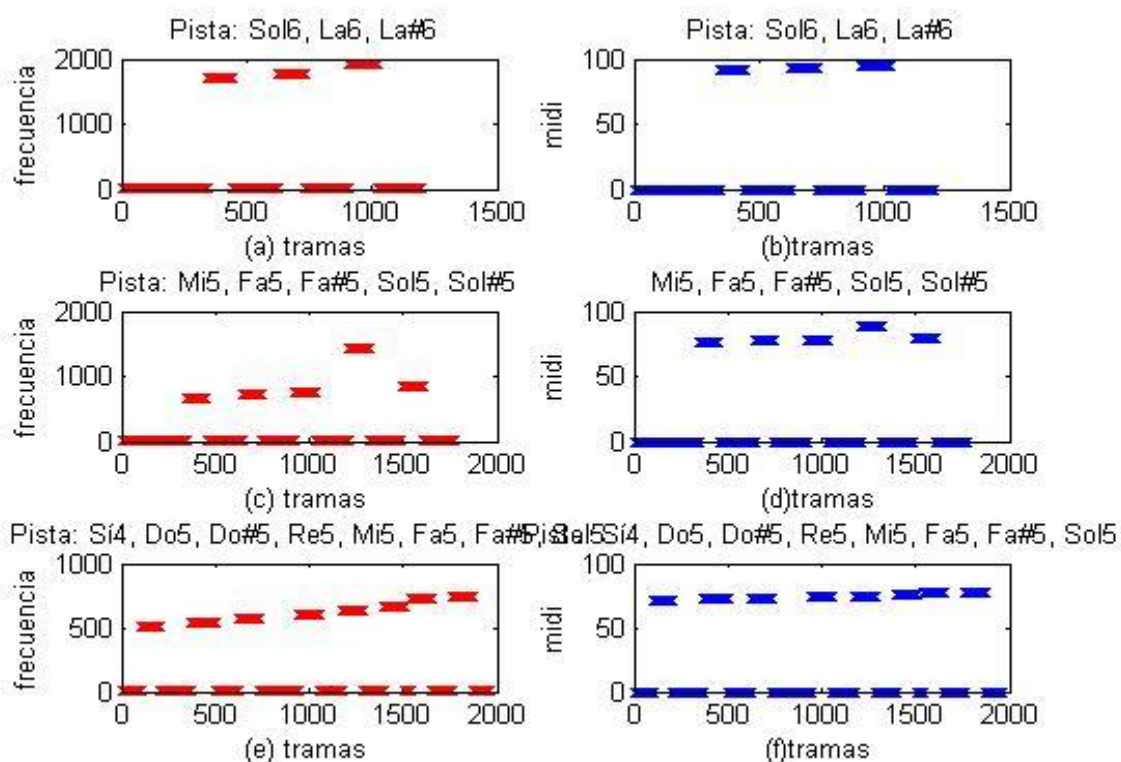


Figura 57: Transformación de frecuencia a midi de varias pistas. Las Figuras (a) y (b) muestran la frecuencia fundamental de la pista formada por tres notas (Sol6, La6, La#6) estimada con el algoritmo implementado y su correspondiente transformación a midi respectivamente. Las Figuras (c) y (d) muestran la frecuencia fundamental estimada de la pista formada por cinco notas (Mi5, Fa5, Fa#5, Sol5, Sol#5) y su correspondiente transformación a midi respectivamente. Las Figuras (e) y (f) muestran la frecuencia fundamental estimada de la pista formada por ocho notas (Sí4, Do5, Do#5, Re5, Mi5, Fa5, Fa#5, Sol5) y su correspondiente transformación a midi respectivamente.

Para terminar con el desarrollo de la implementación de nuestro sistema se va a ilustrar en la Figura 58 todas las frecuencias que estima nuestro sistema. Mientras que en la Figura 59 se puede observar la comparación entre las frecuencias estimadas por nuestro

sistema (representados en color azul) y las frecuencias reales de las 25 notas que dispone la flauta dulce (representados en color rojo)

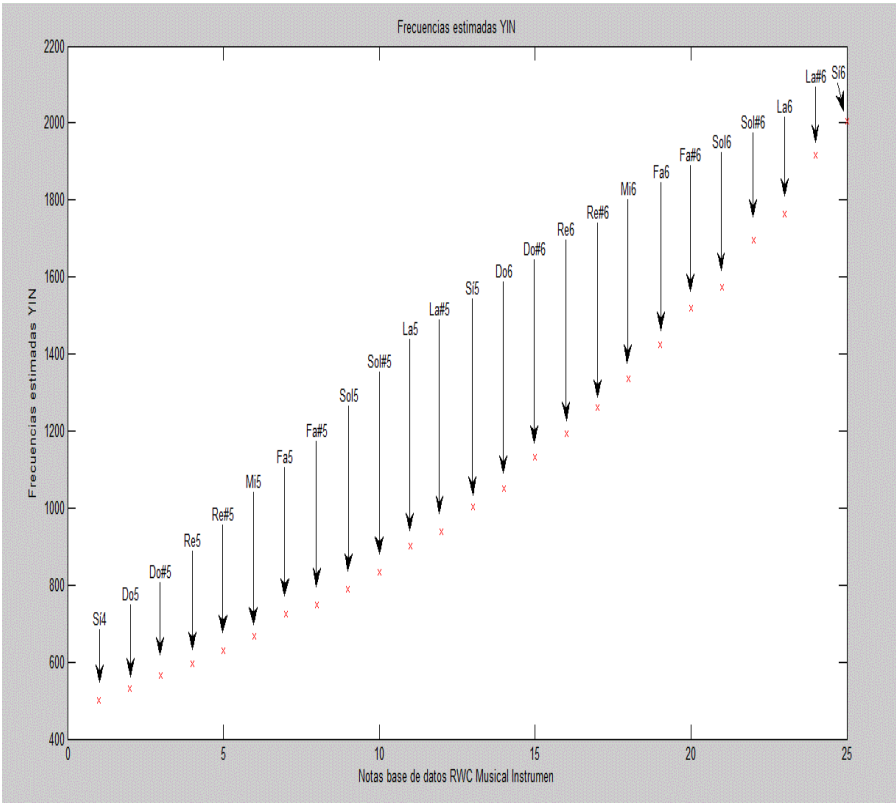


Figura 58: espectro de la flauta dulce estimado por nuestro algoritmo

En la Figura 59, se puede observar la comparación de las frecuencias devueltas por nuestro algoritmo y las frecuencias reales.

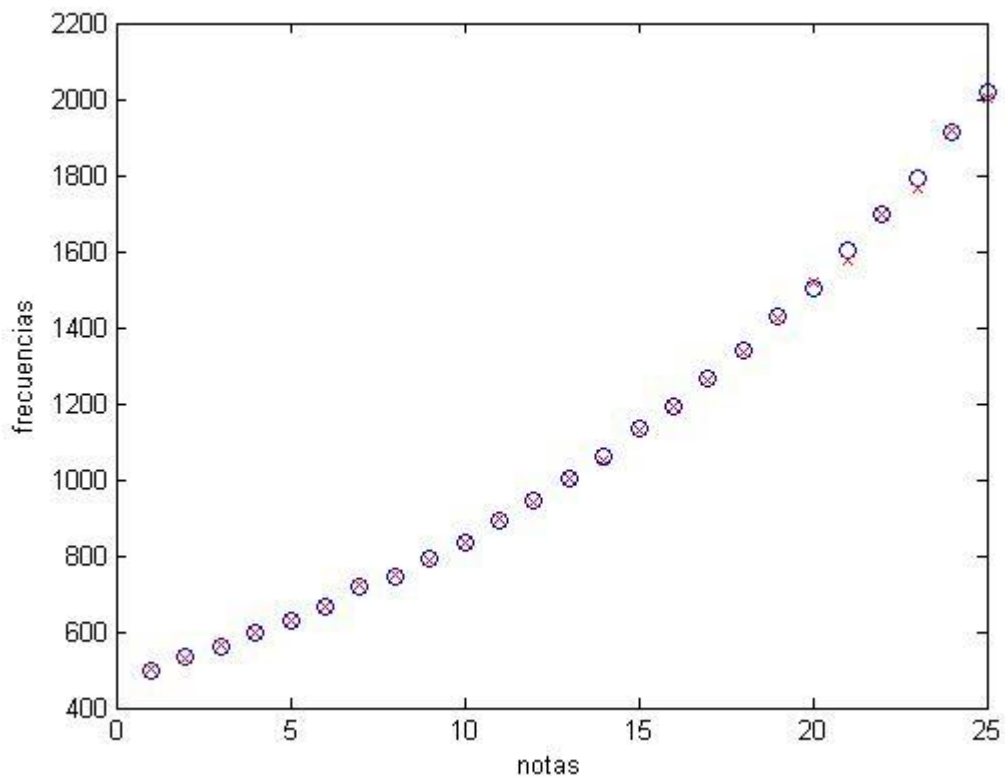


Figura 59: Frecuencias estimadas vs frecuencias reales

Teniendo en cuenta la Figura 59 anterior se puede concluir que el sistema implementado es eficiente porque devuelve frecuencias muy similares o cercanas a las reales por cada nota, aunque como se puede observar existen desviaciones leves.

5 CAPITULO 5. EVALUACIÓN DEL SISTEMA Y RESULTADOS

En el anterior capítulo se han puesto las bases para poder comprender en qué consiste el sistema que se ha previsto desarrollar; de tal forma que se ha explicado con detalle la implementación del algoritmo de detección de pitch CMNDF que se ha realizado en este TFG. Para esta implementación se ha usado el entorno Matlab. Y tras comprobar el correcto funcionamiento del mismo si se tiene en cuenta que las frecuencias estimadas por nuestro algoritmo son muy similares a las frecuencias reales (ver Figura 59), el siguiente paso es evaluar los resultados que devuelve el sistema implementado. Teniendo en cuenta que anteriormente se ha trabajado con notas grabadas profesionalmente por lo que se supone que no incluyen ruido. Y con la necesidad de evaluar nuestro sistema en un ámbito más cotidiano, se va valorar cómo responde el sistema en frecuencia y en tramas bajo la intromisión de distintos niveles de ruido en las notas grabadas; con la clara finalidad de conocer qué tan robusto es nuestro sistema frente al ruido. Para ello primero se explicará el mecanismo realizado para introducir el ruido a las distintas notas de la base de datos para luego adoptar unas determinadas métricas objetivas distintas tanto para la evaluación a nivel de frecuencia como a nivel de tramas, de tal forma que los resultados que obtengamos de ellas sirvan de indicadores de fiabilidad, eficiencia y rendimiento sobre el sistema implementado.

Esta evaluación generalmente consiste en dar respuesta a dos preguntas:

- 1) **¿cuánto se equivoca nuestro sistema sobre las frecuencias estimadas respecto a las frecuencias reales de las notas que componen el espectro de la flauta dulce?**
- 2) **¿Cuánto se equivoca nuestro sistema sobre las tramas detectadas respecto a las tramas propias que contienen las distintas notas de la flauta dulce?**

Este capítulo se va a dividir en dos partes, para diferenciar claramente las dos evaluaciones consideradas.

- **1ª PARTE:** En esta primera parte se realiza una evaluación basada en la comparación de frecuencias y teniendo en cuenta los resultados obtenidos en consecuencia se dará respuesta a la primera pregunta arriba propuesta.
- **2ª PARTE:** En esta parte se realiza una evaluación basada en la comparación de trama por trama y teniendo en cuenta los resultados obtenidos en consecuencia se dará respuesta a la segunda de las preguntas arriba propuestas.

Estos dos parámetros (frecuencia y tiempo) serán utilizados más adelante en el apartado de ejercicios del profesor para igualmente valorar en qué nivel de aprendizaje se le sitúa al usuario mientras realiza los entrenamientos o ejercicios del tutor.

5.1 1ª PARTE. EVALUACIÓN A NIVEL DE FRECUENCIA

5.1.1 Base de datos

Para abordar esta parte de la evaluación consistente en la comparación entre las frecuencias estimadas por nuestro sistema frente a las frecuencias de las notas que forman el espectro de la flauta dulce; y para posteriormente obtener los resultados se ha evaluado cinco bases de datos:

- **Base de datos correspondiente a la carpeta: -5db**

Esta base de datos está contenida en la carpeta llamada ‘‘1ª PARTE’’ y está formada por 25 notas distintas, generalmente cada una de las notas dura aproximadamente unos tres segundos. En esta base de datos se ha introducido un nivel de SNR (relación señal a ruido) de -5db.

CARPETA: 1ª PARTE

Etiqueta	Pista con SNR: -5db
Nota1	Sí4
Nota2	Do5
Nota3	Do#5
Nota4	Re5
Nota5	Re#5
Nota6	Mi5
Nota7	Fa5
Nota8	Fa#5
Nota9	Sol5
Nota10	Sol#5
Nota11	La5
Nota12	La#5
Nota13	Sí5
Nota14	Do6
Nota15	Do#6

Nota16	Re6
Nota17	Re#6
Nota18	Mi6
Nota19	Fa6
Nota20	Fa#6
Nota21	Sol6
Nota22	Sol#6
Nota23	La6
Nota24	La#6
Nota25	Sí6

Tabla 8: Notas pertenecientes a la base de datos -5db. 1ª PARTE

- **Base de datos correspondiente a la carpeta: 0db**

Esta base de datos está contenida en la carpeta llamada “1ª PARTE” y está formada por 25 notas distintas, generalmente cada una de las notas dura aproximadamente unos tres segundos. En esta base de datos se ha introducido un nivel de SNR (relación señal a ruido) de 0db.

CARPETA: 1ª PARTE

Etiqueta	Pista con SNR: 0db
Nota1	Sí4
Nota2	Do5
Nota3	Do#5
Nota4	Re5
Nota5	Re#5
Nota6	Mi5
Nota7	Fa5
Nota8	Fa#5
Nota9	Sol5
Nota10	Sol#5
Nota11	La5
Nota12	La#5

Nota13	Sí5
Nota14	Do6
Nota15	Do#6
Nota16	Re6
Nota17	Re#6
Nota18	Mi6
Nota19	Fa6
Nota20	Fa#6
Nota21	Sol6
Nota22	Sol#6
Nota23	La6
Nota24	La#6
Nota25	Sí6

Tabla 9: Notas pertenecientes a la base de datos 0db. 1ª PARTE

- **Base de datos correspondiente a la carpeta: 5db**

Esta base de datos está contenida en la carpeta llamada ‘‘1ª PARTE’’ y está formada por 25 notas distintas, generalmente cada una de las notas dura aproximadamente unos tres segundos. En esta base de datos se ha introducido un nivel de SNR (relación señal a ruido) de 5db.

CARPETA: 1ª PARTE

Etiqueta	Pista con SNR: 5db
Nota1	Sí4
Nota2	Do5
Nota3	Do#5
Nota4	Re5
Nota5	Re#5
Nota6	Mi5
Nota7	Fa5
Nota8	Fa#5
Nota9	Sol5

Nota10	Sol#5
Nota11	La5
Nota12	La#5
Nota13	Sí5
Nota14	Do6
Nota15	Do#6
Nota16	Re6
Nota17	Re#6
Nota18	Mi6
Nota19	Fa6
Nota20	Fa#6
Nota21	Sol6
Nota22	Sol#6
Nota23	La6
Nota24	La#6
Nota25	Sí6

Tabla 10: Notas pertenecientes a la base de datos 5db. 1ª PARTE

- **Base de datos correspondiente a la carpeta: 10db**

Esta base de datos está contenida en la carpeta llamada ‘‘1ª PARTE’’ y está formada por 25 notas distintas, generalmente cada una de las notas dura aproximadamente unos tres segundos. En esta base de datos se ha introducido un nivel de SNR (relación señal a ruido) de 10 dB.

CARPETA: 1ª PARTE

Etiqueta	Pista con SNR: 10db
Nota1	Sí4
Nota2	Do5
Nota3	Do#5
Nota4	Re5
Nota5	Re#5
Nota6	Mi5

Nota7	Fa5
Nota8	Fa#5
Nota9	Sol5
Nota10	Sol#5
Nota11	La5
Nota12	La#5
Nota13	Sí5
Nota14	Do6
Nota15	Do#6
Nota16	Re6
Nota17	Re#6
Nota18	Mi6
Nota19	Fa6
Nota20	Fa#6
Nota21	Sol6
Nota22	Sol#6
Nota23	La6
Nota24	La#6
Nota25	Sí6

Tabla 11: Notas pertenecientes a la base de datos 10db. 1ª PARTE

5.1.2 Inicialización

En este apartado se va a explicar todos los métodos y pautas seguidas para acondicionar nuestro ámbito de evaluación con el fin de evaluar la robustez de nuestro sistema bajo condiciones similares al mundo real.

En primer lugar, como la herramienta esta prevista a que sea utilizada en el mundo real, donde se debe tener en cuenta un factor importante como es el ruido, es por esto que se ha pensado evaluar nuestra herramienta bajo condiciones ruidosas semejantes a las que se pueden presentar en un ámbito real. Para esto se han creado diversas bases de datos correspondientes a distintos niveles de relación señal a ruido: -5db, 0db, 5db, 10db. Pero

antes de explicar en profundidad el método programado para insertar el ruido a nuestras bases de datos, primero se va a situar al lector definiéndole brevemente el concepto de ruido y sus tipos.

- **El ruido:**

Se define el ruido como cualquier perturbación que experimenta la señal en un sistema de comunicación. Más concretamente se puede considerar el ruido como una señal no deseada que se suma o se mezcla a la señal útil, la cual se desea transmitir; y como consecuencia esta señal indeseable llega a perturbar a la señal útil de tal forma que tiende a enmascarar la información que porta ésta cuando se sitúa en la banda de frecuencias del espectro útil o lo que es lo mismo dentro del ancho de banda de la señal útil.

En los sistemas de comunicaciones, existen varios tipos de ruidos según las características que interesen, se puede mencionar:

- Según el tiempo: ruido estacionario, ruido fluctuante, ruido intermitente, ruido impulsivo [89].
- Según la frecuencia: ruido blanco, ruido rosa, ruido tonal y ruido de bajas frecuencia [89].

Es de interés para este TFG que a la hora de evaluar nuestro sistema con ruido se elija el tipo de ruido con características semejantes que se pueden encontrar en un ámbito real, se centrará el estudio en los ruidos que tienen una distribución gaussiana y una densidad espectral plana dentro de un amplio rango de frecuencias. Al tipo de ruidos cuya intensidad es constante para todas las frecuencias del espectro se le conoce como ruido blanco y gaussiano. Este tipo de ruido es muy utilizado en sistemas de comunicaciones. Por lo que se asume que el ruido blanco y gaussiano es el que se adapta a las condiciones en las que se quiere someter a evaluación nuestro sistema; por lo que seguidamente se va a explicar cómo se ha añadido a nuestra base de datos sin ruido distintos valores de SNR.

Se parte desde el punto de que se dispone de una base de datos llamada ‘sin_ruido’ grabada por profesionales donde se presupone que se grabó en salas que guardan las condiciones acústicas requeridas para estas características a fin de aislar el ruido; por lo que esta carpeta la se supone sin ruido. Para crear nuestra base de datos con distintos niveles de ruido, se ha utilizado el entorno Matlab para llevar a cabo este propósito.

El archivo Matlab que contiene el código fuente para añadir distintos niveles de señal a ruido a las notas es: `baseruido.m`

Este archivo .m realiza tres operaciones:

1º) Selecciona el directorio de trabajo donde se encuentran las notas sin ruido incluido, que en nuestro caso es la carpeta “sin_ruido”

2º) Con la sentencia `awgn`, se añade ruido a nuestra base de datos seleccionada:

```
mi_senal_ruidosa = awgn(x, SNR, 'measured');
```

En este punto se va a requerir que se ingrese el valor de la SNR, por lo que será: -5db, 0db, 5db, 10db dependiendo del caso que nos ocupe.

3º) Se guardan las notas ya con ruido dentro de la carpeta que en este caso se ha llamado ‘Music’.

```
Wavwrite(mi_senal_ruidosa, fs, [pathname filesep 'Music' filesep filename(1: end-4) '.wav']);
```

La Figura 60 muestra la estructura de las bases de datos para añadir distintos valores de ruido.

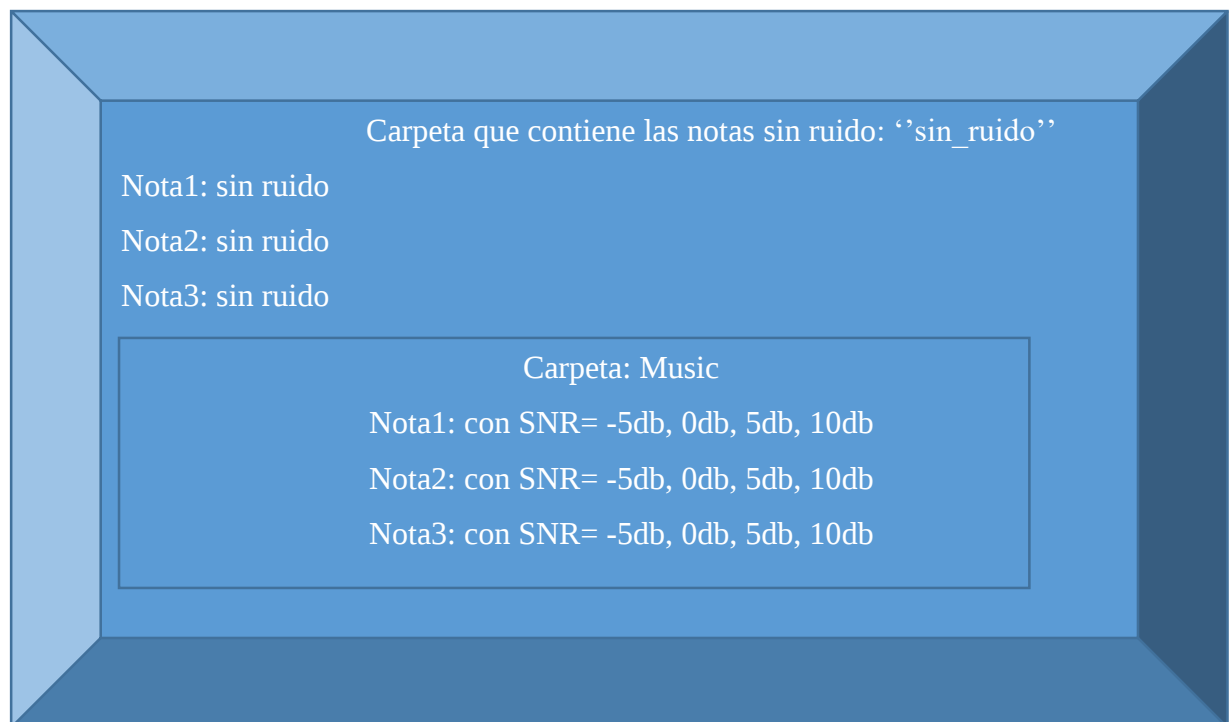


Figura 60: Esquema que resume como añadir ruido

En la Figura 61, se puede observar una señal sin ruido. Mientras que en la Figura 62 se observa el resultado de añadir ésta diferentes valores de ruido.

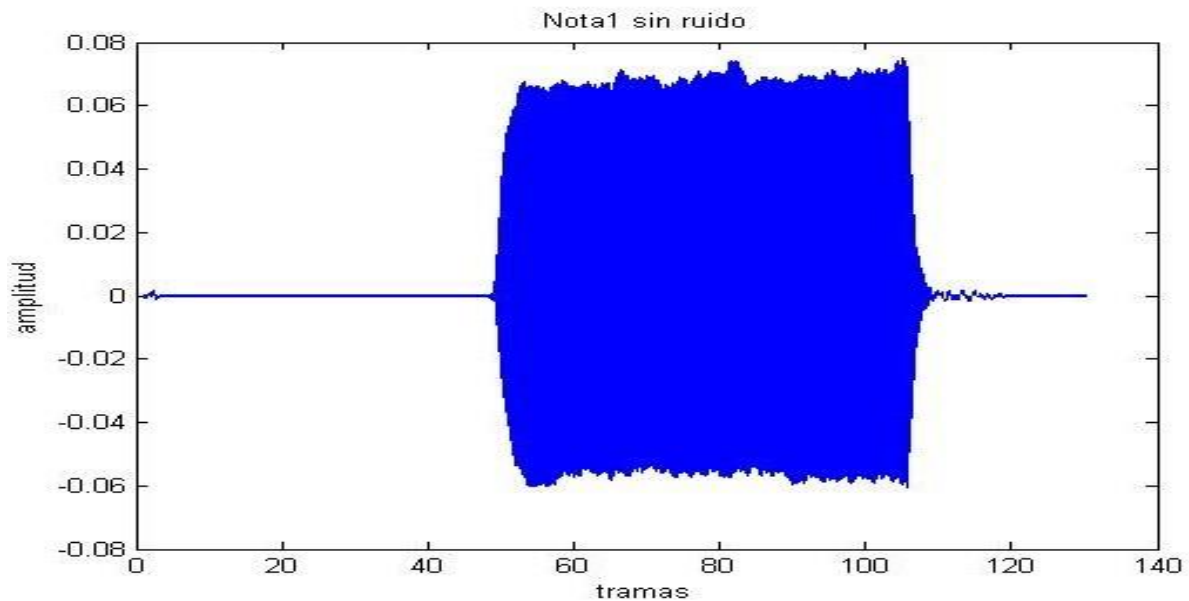


Figura 61: Señal sin ruido

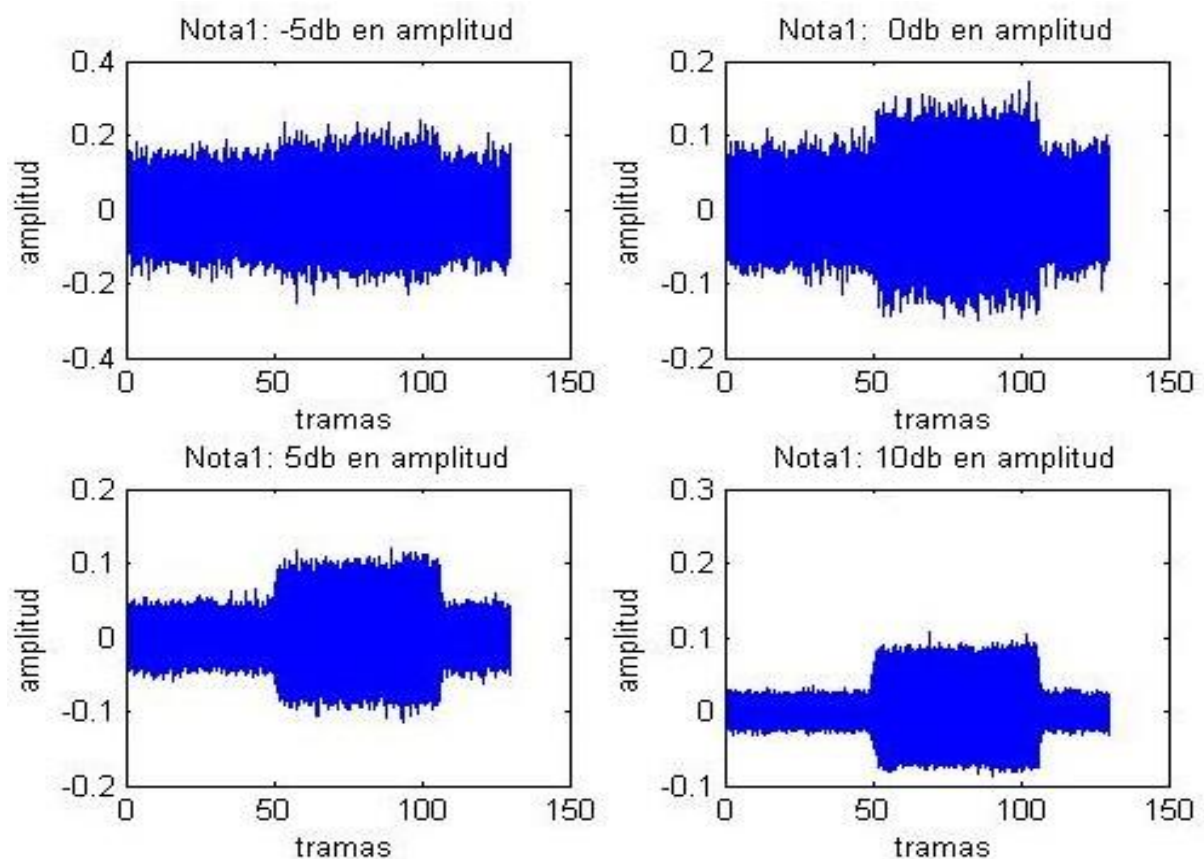


Figura 62: Señal resultante al añadir distintos niveles de ruido. La primera fila de la Figura 62 de izquierda a derecha representa el resultado de incluir a una señal (Nota1) un nivel de ruido de (-5db y 0db) respectivamente. La segunda fila de la Figura 62 de izquierda a derecha representa el resultado de incluir a una señal (Nota1) un nivel de ruido de (5db y 10db) respectivamente.

Como se puede observar la amplitud de la señal es diferente cuando se la introduce ruido, pero también es distinta dependiendo de la potencia de ruido introducida. Un detalle que puede ser interesante para el lector es que aquí se introduce el ruido en términos de relación de potencias; es decir que para conocer la potencia de ruido propia de cada nota dentro de la misma relación SNR, se deberá tener en cuenta la relación entre la potencia y la energía que viene descrita por la siguiente ecuación:

$$SNR(db) = 10 \log_{10} \left(\frac{P_S}{P_N} \right) \quad (58)$$

5.1.3 Métricas utilizadas

Una vez preparado nuestro ámbito de evaluación con cuatro bases de datos distintas. Y con la finalidad de evaluar los resultados obtenidos a nivel de frecuencia por el algoritmo de estimación de pitch implementado, se han determinado dos medidas objetivas.

Las métricas que se han considerado son, **el error absoluto y el error relativo**. Se ha seguido como referencia el artículo [90]

- Error absoluto: Es la diferencia entre el valor obtenido de una medida determinada y el valor que se toma como verdadero o real [90].

$$E_{abs} = |x - x_0| \quad (59)$$

Donde x corresponde al valor obtenido de la medida, y x_0 corresponde al valor real.

La magnitud a medir es la frecuencia fundamental de las notas correspondientes a la flauta dulce; por consiguiente el valor real o verdadero corresponde al espectro de la flauta dulce (ver tabla 12) y el valor medido corresponde a la frecuencia que se estima con el algoritmo implementado en este TFG.

- Error relativo: Es el cociente entre error absoluto y el valor real. Y multiplicado por cien se obtiene en valores porcentuales [90].

$$E_{rel}(\%) = \frac{E_{abs}}{x_0} \times 100 \quad (60)$$

Donde $E_{rel}(\%)$ es el error relativo en porcentaje.

Con estas métricas se consigue conocer el comportamiento del sistema implementado a nivel de detección de frecuencia sometido bajo distintas condiciones de ruido.

5.1.4 Resultados

En este apartado se analizan los resultados obtenidos con nuestro sistema a nivel de evaluación en frecuencia para cada una de las cuatro bases de datos arriba propuestas. En realidad, lo que aquí se intenta juzgar es la robustez o no de nuestro sistema frente a distintas condiciones ruidosas. Es decir, valorar su rendimiento o eficiencia sujeto a cómo varía la frecuencia de cada nota para las distintas condiciones de ruido.

En este apartado se va a evaluar el sistema mediante los dos métodos utilizados en este TFG para estimar la frecuencia fundamental. Estos dos métodos son: asignar la frecuencia fundamental de una determinada nota a la media de las frecuencias de las tramas sonoras que forman la nota, y asignar la frecuencia fundamental de una determinada nota a la frecuencia más repetitiva de las tramas sonoras que forman la nota; es decir la frecuencia que resulta de aplicar la moda matemática a las tramas sonoras.

- **Asignación de la frecuencia fundamental mediante la media**

A continuación, se puede ver en primer término en la tabla 12 los resultados en frecuencia de las distintas condiciones de ruido y en segundo lugar los resultados obtenidos objetivamente mediante las métricas utilizadas; habiendo utilizado la media para asignar la frecuencia de la nota.

NOTA/SNR	-5 dB	0 dB	5 dB	10 dB	original
Nota 1	3662.3	1075.2	644.7	499.9	499.2
Nota 2	4171.6	1687.5	755.5	618	533.5
Nota 3	1969.7	1062.8	564.1	563.9	562.2
Nota 4	3714.5	955.3	597.5	598.5	598.5
Nota 5	2215.2	629.1	627.8	628.14	628.2
Nota 6	3210.5	912.4	665.8	667	667.8
Nota 7	3713	1096.3	762.6	724.4	716.8
Nota 8	2315.2	1021.6	811.6	759.2	747.2
Nota 9	3208.3	1240	920	855	792.3
Nota 10	3240.3	1193.1	960	840.7	834.8
Nota 11	2880.6	1153.9	1024.1	989.3	894.2
Nota 12	4079	1307.2	1228.4	986.8	943.3
Nota 13	1742.9	850.2	857.6	1038.7	1001
Nota 14	2149.4	866.9	922.12	1004.7	1062
Nota 15	2176.9	1012.8	1082.5	1093.2	1135
Nota 16	1981.2	970.9	1031.4	1136.7	1191
Nota 17	3028.7	1101.2	1203.9	1219.7	1267
Nota 18	1876.9	1147.4	1258	1336.8	1337
Nota 19	1120.4	1255.1	1533.5	1355	1428
Nota 20	1068.2	1179.9	1298.7	1468	1504
Nota 21	1351.5	1343.8	1331.8	1441.7	1603
Nota 22	1346.2	1371.4	1675.8	1684.4	1697
Nota 23	1579.3	1139.6	1422.9	1678.1	1792
Nota 24	1483.4	1548.6	1794.7	2017.5	1912
Nota 25	1413.3	1485.5	1972	1982.1	2016

Tabla 12: Resultados obtenidos en frecuencia para distintos valores de ruido, asignando la frecuencia estimada a la media de las tramas sonoras.

Una forma más cómoda de ver estos resultados se tiene en la Figura 63 donde se puede observar que en las 25 notas de las bases de datos la frecuencia estimada mejora cuanto mayor es la SNR.

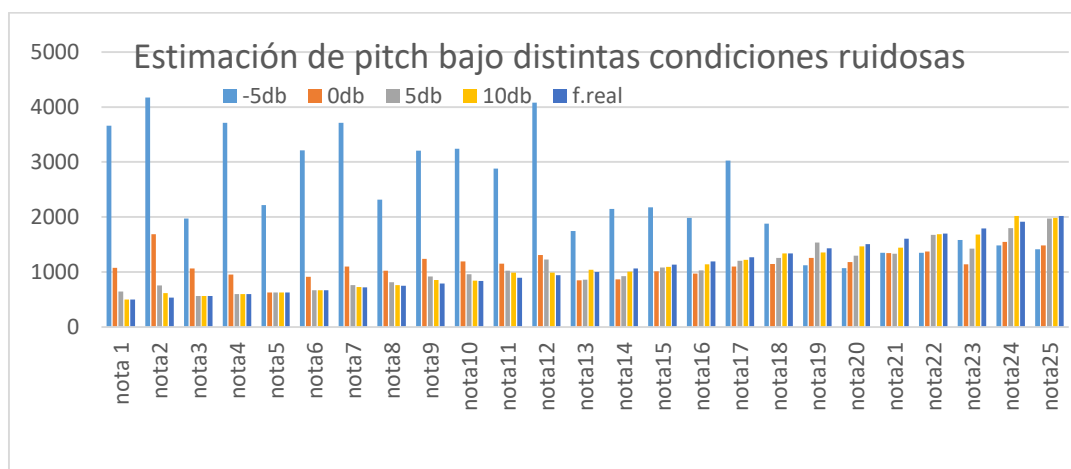


Figura 63: Resultados en frecuencia de las 25 notas bajo distintas condiciones de SNR, asignando la frecuencia estimada a la media de las tramas sonoras.

La tabla 13 muestra los resultados derivados de las métricas fijadas para esta evaluación.

NOTA/SNR	-5db		0db		5db		10db	
	Eabs	Erel (%)	Eabs	Erel (%)	Eabs	Erel (%)	Eabs	Erel (%)
Nota 1	3163.1	633.6	576	115.3	145.5	29.15	0.7	0.1
Nota 2	3638.1	681.9	154	216.3	222	41.6	84.5	15.8
Nota 3	1407.5	250.3	500.6	89	1.9	0.3	1.7	0.3
Nota 4	3116	520.6	356.8	59.6	1	0.1	0	0
Nota 5	1587	252.6	9	0.1	0.4	0.06	0.06	0.009
Nota 6	2542.7	380.7	244.6	36.6	2	0.3	0.8	0.1
Nota 7	2996.2	417.9	379.5	52.9	45.8	6.4	7.6	1.06
Nota 8	1568	209.8	274.4	36.7	64.4	8.6	12	1.6
Nota 9	2416	304.9	447.7	56.5	127.7	16.1	62.7	7.9
Nota 10	2405.5	288.1	358.3	42.9	125.2	14.9	5.9	0.7
Nota 11	1986.4	222.1	259.7	29	129.9	14.5	95.1	10.6
Nota 12	3135.7	332.4	363.9	38.5	285.1	30.2	43.5	4.6
Nota 13	741.9	74.1	150.8	15	143.4	14.3	37.7	3.7
Nota 14	1087.4	102.3	195.1	18.3	139.8	13.1	57.3	5.3
Nota 15	1041.9	91.7	122.2	10.7	52.5	4.6	41.8	3.6
Nota 16	790.2	66.3	220.1	18.4	159.6	13.4	54.3	4.5
Nota 17	761.7	139	165.8	13	63.1	4.9	47.3	3.7
Nota 18	539.9	40.3	189.6	14.1	79	5.9	0.2	0.01
Nota 19	307.6	21.5	172.9	12.1	105.5	7.3	73	5.1
Nota 20	435.8	28.9	324.1	21.5	205.3	13.6	36	2.3
Nota 21	251.5	15.6	259.2	16.1	271.2	16.9	161.3	10.06
Nota 22	350.8	20.6	325.6	19.1	21.2	1.2	12.6	0.7
Nota 23	212.7	11.8	652.4	36.4	369.1	20.5	113.9	6.3
Nota 24	428.6	22.4	363.4	19	117.3	6.1	105.5	5.5
Nota 25	602.7	29.8	530.5	26.3	44	2.1	33.9	1.6
MEDIA	1500.6	206.3	303.8	40.5	116.8	11.4	43.5	3.8
MEDIANA	1087.4	139	274.4	26.3	117.3	8.6	37.7	3.6

Tabla 13: Resultados de error absoluto y relativo, asignando la frecuencia estimada a la media de las tramas sonoras.

En la Figura 64, se puede ver el resultado de las métricas de forma más cómoda.

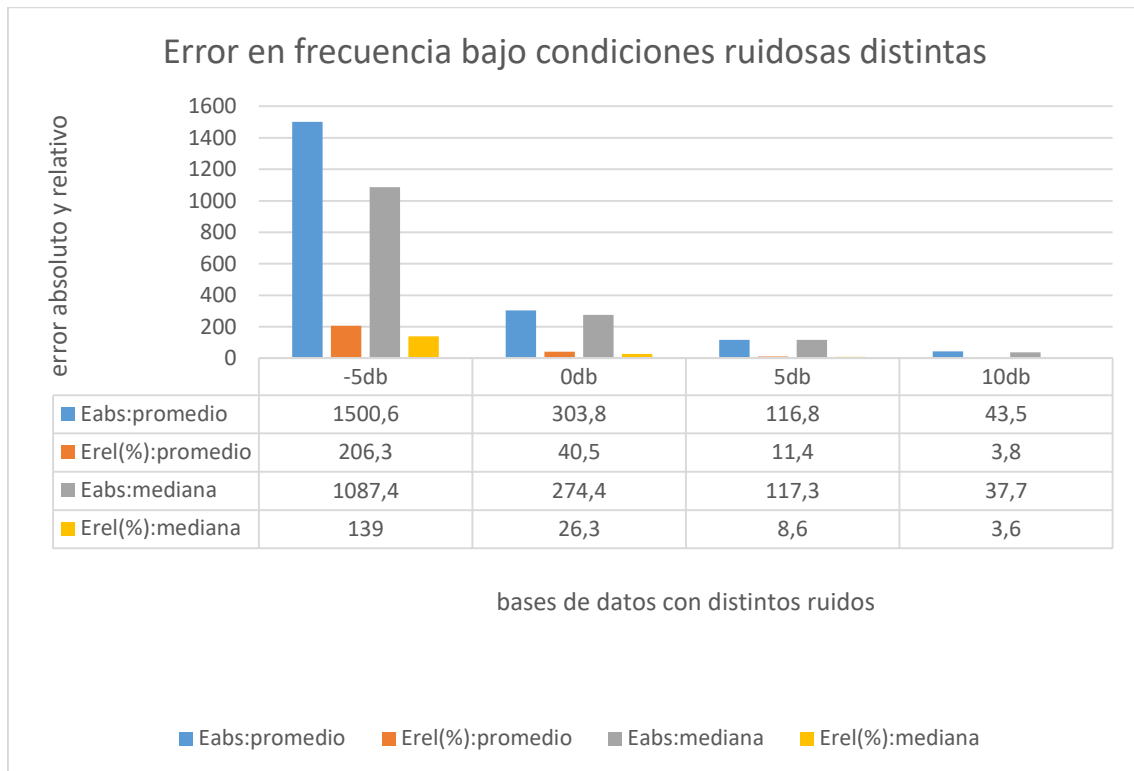


Figura 64: Resultados del error absoluto y relativo, asignando la frecuencia estimada a la media de las tramas sonoras

• Discusión de los resultados

En la Figura 63 se puede observar que conforme mayor es la SNR, mejor serán los resultados que se obtiene de nuestro sistema.

Otra valoración importante se puede deducir de la Figura 64 donde los parámetros de medida son el error absoluto y relativo. Ambos errores son indicadores de desviación de la frecuencia. El error absoluto es un indicador de imprecisión sobre las frecuencias estimadas por nuestro sistema; por tanto el error absoluto y relativo aportan información sobre la robustez del sistema frente al ruido. Como se puede observar en la Figura 64 el error absoluto y relativo disminuye en cuanto mejora la SNR. En el caso de SNR peores; por ejemplo -5db por ejemplo el error relativo es de superior (139 %); mientras que por ejemplo para 10db es de 3.6 %, o sea una disminución de error de 135%; lo que implica que en escenarios de niveles de ruidos considerados, el sistema no funcionará adecuadamente, porque el ruido enmascara al sonido procedente de la flauta.

- **Asignación de la frecuencia fundamental mediante la moda matemática**

A continuación, se puede ver en primer término en la tabla 14 los resultados en frecuencia de las distintas condiciones de ruido y en segundo lugar los resultados obtenidos objetivamente mediante las métricas utilizadas; habiendo utilizado la moda matemática para asignar la frecuencia de la nota.

NOTA/SNR	-5 dB	0 dB	5 dB	10 dB	f.real
Nota 1	501.13	501.13	501.13	501.13	499.2
Nota 2	537.80	531.32	531.32	531.32	533.5
Nota 3	565.38	565.38	565.38	565.38	562.2
Nota 4	588	595.94	595.94	595.94	598.5
Nota 5	630	630	630	630	628.2
Nota 6	678.46	668.18	668.18	668.18	667.8
Nota 7	711.29	711.29	722.95	722.95	716.8
Nota 8	747.45	747.45	747.45	747.45	747.2
Nota 9	787.50	787.50	787.50	787.50	792.3
Nota 10	848.07	848.07	832.07	832.07	834.8
Nota 11	900	900	900	900	894.2
Nota 12	958.69	938.29	938.29	938.29	943.3
Nota 13	1002.27	1002.27	1002.27	1002.27	1001
Nota 14	1075.60	1050	1050	1050	1062
Nota 15	1130.76	1130.76	1130.76	1130.76	1135
Nota 16	1191.89	1191.89	1191.89	1191.89	1191
Nota 17	1260	1260	1260	1260	1267
Nota 18	1336.36	1336.36	1336.36	1336.36	1337
Nota 19	1422.58	1422.58	1422.58	1422.58	1428
Nota 20	1520.68	1520.68	1520.68	1520.68	1504
Nota 21	1633.33	1633.33	1633.33	1633.33	1603
Nota 22	1696.15	1696.15	1696.15	1696.15	1697
Nota 23	1764	1764	1764	1764	1792
Nota 24	1917.39	1917.39	1917.39	1917.39	1912
Nota 25	2004.54	2004.54	2004.54	2004.54	2016

Tabla 14: Resultados obtenidos en frecuencia para distintos valores de ruido, asignando la frecuencia estimada a la más repetitiva (moda matemática) de las tramas sonoras

Una forma más cómoda de ver estos resultados se tiene en la Figura 65 donde se puede ver que las 25 notas tienen la misma frecuencia para los distintos valores de ruido.

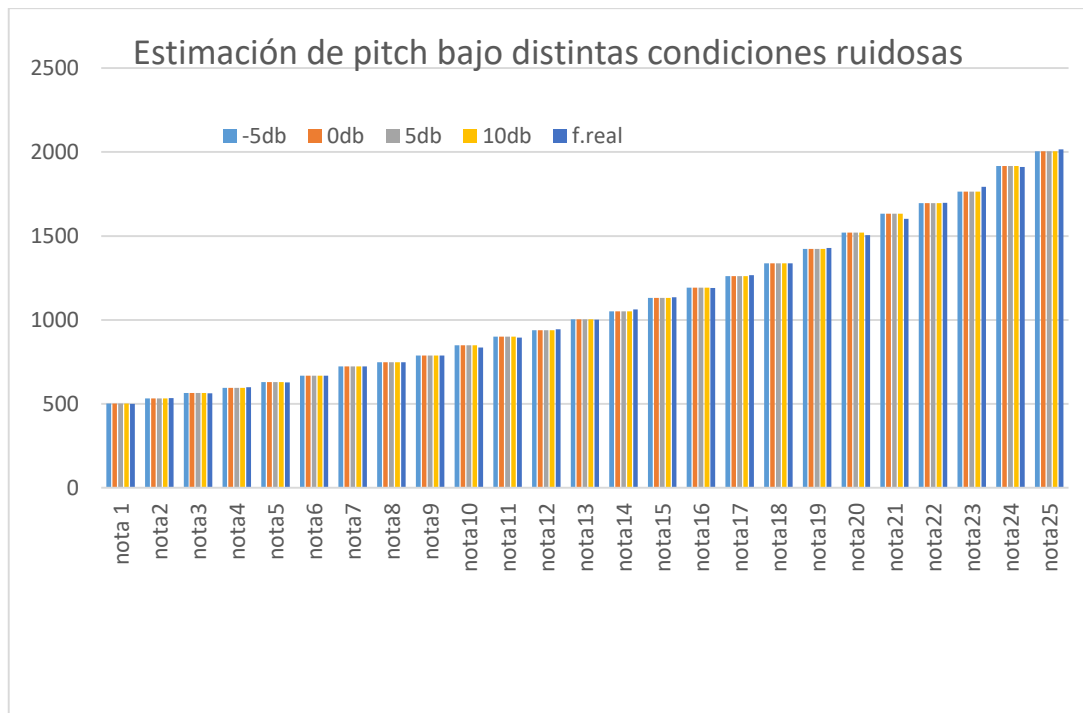


Figura 65: Resultados en frecuencia de las 25 notas bajo distintas condiciones de SNR, asignando la frecuencia estimada a la más repetitiva (moda matemática) de las tramas sonoras.

La tabla 15 muestra los resultados derivados de las métricas fijadas para esta evaluación.

NOTA/SNR	-5db		0db		5db		10db	
	Eabs	Erel (%)	Eabs	Erel (%)	Eabs	Erel (%)	Eabs	Erel (%)
Nota 1	3163.1	633.6	576	115.3	145.5	29.15	0.7	0.1
Nota 2	3638.1	681.9	154	216.3	222	41.6	84.5	15.8
Nota 3	1407.5	250.3	500.6	89	1.9	0.3	1.7	0.3
Nota 4	3116	520.6	356.8	59.6	1	0.1	0	0
Nota 5	1587	252.6	9	0.1	0.4	0.06	0.06	0.009
Nota 6	2542.7	380.7	244.6	36.6	2	0.3	0.8	0.1
Nota 7	2996.2	417.9	379.5	52.9	45.8	6.4	7.6	1.06
Nota 8	1568	209.8	274.4	36.7	64.4	8.6	12	1.6
Nota 9	2416	304.9	447.7	56.5	127.7	16.1	62.7	7.9
Nota 10	2405.5	288.1	358.3	42.9	125.2	14.9	5.9	0.7
Nota 11	1986.4	222.1	259.7	29	129.9	14.5	95.1	10.6
Nota 12	3135.7	332.4	363.9	38.5	285.1	30.2	43.5	4.6
Nota 13	741.9	74.1	150.8	15	143.4	14.3	37.7	3.7
Nota 14	1087.4	102.3	195.1	18.3	139.8	13.1	57.3	5.3
Nota 15	1041.9	91.7	122.2	10.7	52.5	4.6	41.8	3.6
Nota 16	790.2	66.3	220.1	18.4	159.6	13.4	54.3	4.5
Nota 17	761.7	139	165.8	13	63.1	4.9	47.3	3.7
Nota 18	539.9	40.3	189.6	14.1	79	5.9	0.2	0.01
Nota 19	307.6	21.5	172.9	12.1	105.5	7.3	73	5.1
Nota 20	435.8	28.9	324.1	21.5	205.3	13.6	36	2.3
Nota 21	251.5	15.6	259.2	16.1	271.2	16.9	161.3	10.06
Nota 22	350.8	20.6	325.6	19.1	21.2	1.2	12.6	0.7
Nota 23	212.7	11.8	652.4	36.4	369.1	20.5	113.9	6.3
Nota 24	428.6	22.4	363.4	19	117.3	6.1	105.5	5.5
Nota 25	602.7	29.8	530.5	26.3	44	2.1	33.9	1.6
MEDIA	1500.6	206.3	303.8	40.5	116.8	11.4	43.5	3.8
MEDIANA	1087.4	139	274.4	26.3	117.3	8.6	37.7	3.6

Tabla 15: Resultados de error absoluto y relativo, asignando la frecuencia estimada a la más repetitiva (moda matemática) de las tramas sonoras

En la Figura 66, se puede ver el resultado de las métricas de forma más cómoda.

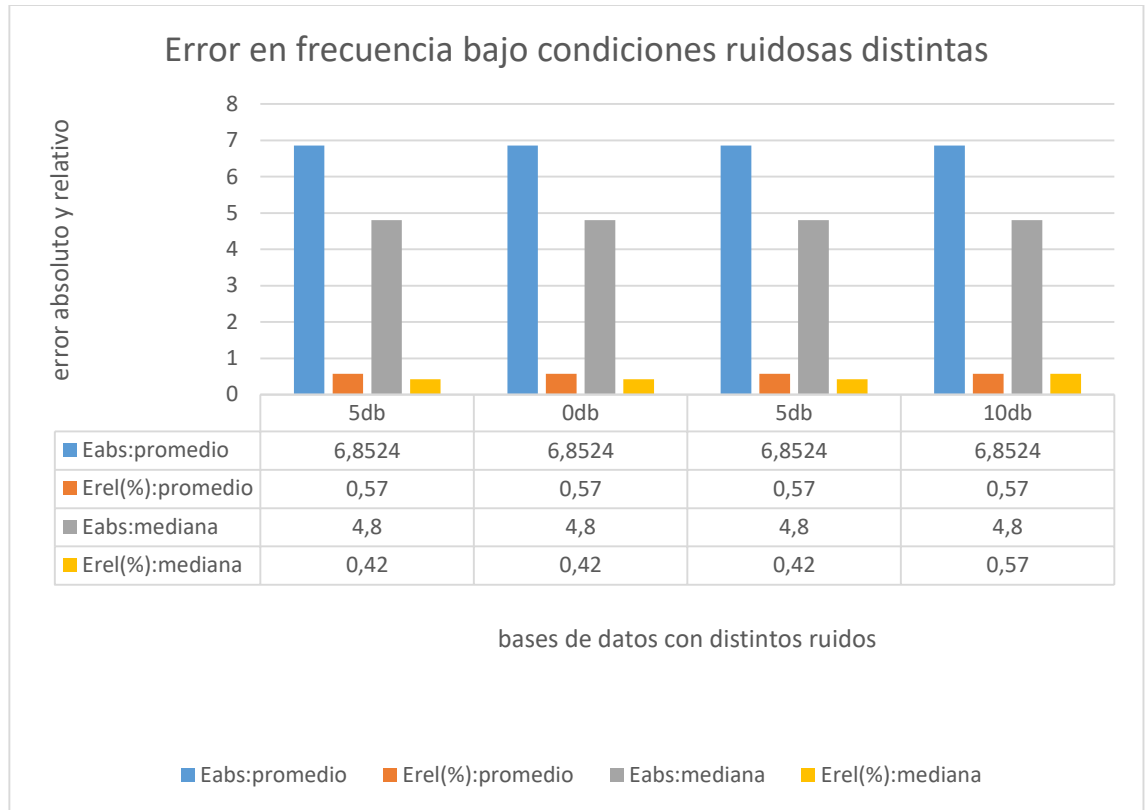


Figura 66: Resultados dl error absoluto y relativo, asignando la frecuencia estimada a la más repetitiva (moda matemática) de las tramas sonoras.

• Discusión de los resultados

En una primera valoración como se puede observar en la Figura 65 los resultados obtenidos por nuestro sistema en frecuencia son iguales para las cuatro condiciones de ruido propuestas, por lo que se puede decir que nuestro sistema es muy robusto frente al ruido. Otra valoración importante se puede deducir de la Figura 66 donde los parámetros de medida son el error absoluto y relativo. Ambos errores son indicadores de desviación de la frecuencia. El error absoluto es un indicador de imprecisión sobre las frecuencias estimadas por nuestro sistema. Esto significa que las frecuencias que estime nuestro sistema por cada nota respecto a las frecuencias reales del espectro de la flauta dulce guardaran aproximadamente la siguiente relación:

$$f_{estimada} \approx f_{real} \pm Eabs \quad (61)$$

Es decir, si se tiene en cuenta que se ha obtenido un error absoluto de 6.58 Hz; esto implica que la frecuencia estimada para una determinada nota será 6.58 Hz superior o inferior que la frecuencia real. Pero si se mira con detenimiento la tabla 14 la diferencia verdadera entre la frecuencia real y la estimada no es en la mayoría de las notas de 6.58 Hz, salvo en algunas notas como la nota 23 y 25 donde es mayor que el promedio de error absoluto, lo que provoca que la media del error absoluto sea un tanto superior a la verdadera medida del error absoluto. Por lo que teniendo la necesidad de ajustarse a la medida verdadera que nos indique la desviación propia en frecuencia se va valorar a partir de la mediana ya que nos indica el verdadero término central del error absoluto cometido por nuestro sistema y este valor verdadero es un error absoluto igual a 4.8 Hz, inferior que el obtenido por la media. Este valor de error es más coherente con los resultados.

Otro indicador importante es el error relativo que es un indicador de calidad y con la mediana tenemos que es de 0.42 %.

5.2 2ª PARTE. EVALUACIÓN A NIVEL DE TRAMAS

5.2.1 Base de datos

Para abordar esta parte de la evaluación consistente en la comparación trama por trama entre las que detecta el sistema implementado y las que realmente forman la nota; y para posteriormente obtener los resultados hemos evaluado cuatro bases de datos:

- **Base de datos correspondiente a la carpeta: -5db**

Esta base de datos está contenida en la carpeta llamada ‘‘2ª PARTE’’ y está formada por 10 pistas que tienen 5 notas aleatorias, generalmente cada una de las pistas dura aproximadamente unos trece segundos. En esta base de datos se ha introducido un nivel de SNR (relación señal a ruido) de -5db.

Etiqueta	Notas compuestas : 5db
pista1	Notas: 2,3,4,5,6
pista2	Notas: 7,8,9,10,11
pista3	Notas: 12,13,14,15,16
pista4	Notas: 17,18,19,20,21
pista5	Notas: 2,22,23,24,25
pista6	Notas: 2,4,6,8,10

pista7	Notas: 12,14,16,18,20
pista8	Notas: 3,5,7,9,11
pista9	Notas: 13,15,17,19,21
pista10	Notas: 21,22,23,24,25

Tabla 16: Notas pertenecientes a la base de datos 5db. 2ª PARTE

- **Base de datos correspondiente a la carpeta: 0db**

Esta base de datos está contenida en la carpeta llamada ‘‘2ª PARTE’’ y está formada por 10 pistas que tienen 5 notas aleatorias, generalmente cada una de las pistas dura aproximadamente unos trece segundos. En esta base de datos se ha introducido un nivel de SNR (relación señal a ruido) de 0db.

Etiqueta	Notas compuestas : 0db
pista1	Notas: 2,3,4,5,6
pista2	Notas: 7,8,9,10,11
pista3	Notas: 12,13,14,15,16
pista4	Notas: 17,18,19,20,21
pista5	Notas: 2,22,23,24,25
pista6	Notas: 2,4,6,8,10
pista7	Notas: 12,14,16,18,20
pista8	Notas: 3,5,7,9,11
pista9	Notas: 13,15,17,19,21
pista10	Notas: 21,22,23,24,25

Tabla 17: Notas pertenecientes a la base de datos 5db. 2ª PARTE

- **Base de datos correspondiente a la carpeta: 5db**

Esta base de datos está contenida en la carpeta llamada ‘‘2ª PARTE’’ y está formada por 10 pistas que tienen 5 notas aleatorias, generalmente cada una de las pistas dura aproximadamente unos trece segundos. En esta base de datos se ha introducido un nivel de SNR (relación señal a ruido) de 5db.

Etiqueta	Notas compuestas : 5db
pista1	Notas: 2,3,4,5,6
pista2	Notas: 7,8,9,10,11

pista3	Notas: 12,13,14,15,16
pista4	Notas: 17,18,19,20,21
pista5	Notas: 2,22,23,24,25
pista6	Notas: 2,4,6,8,10
pista7	Notas: 12,14,16,18,20
pista8	Notas: 3,5,7,9,11
pista9	Notas: 13,15,17,19,21
pista10	Notas: 21,22,23,24,25

Tabla 18: Notas pertenecientes a la base de datos 5db. 2ª PARTE

5.2.2 Inicialización

Al igual que la evaluación en frecuencia se pretende evaluar el sistema implementado a nivel de tramas. Es decir, valorar qué tramas sonoras se detectan como silencio cuando en realidad las son sonoras, o qué tramas de silencio se detectan como sonoras cuando en realidad no las son, así como determinar la precisión en tramas verificando en cómputo si se detecta como equitativas la mayoría o minoría de las tramas. Para llevar a cabo esta evaluación se tendrá que realizarlo bajo ciertas métricas objetivas que permitirán hacer una discusión coherente de los resultados y en consecuencia definirnos sobre el rendimiento del sistema.

5.2.3 Métricas utilizadas

Las métricas que se han considerado son, las tramas de falsos negativos (TFN), las tramas de falsos positivos (TFP) y la precisión (Acc). Se ha seguido como referencia el artículo [91].

- Trama de falsos negativos: con esta métrica se comprueba cuántas tramas de una señal se rechazan como tramas sonoras cuando en realidad sí son sonoras; es decir con esta métrica se indica cuantas tramas sonoras son detectadas como silencio, cuando en realidad son sonoras [91]. Viene descrita en términos porcentuales por:

$$TFN = \frac{TFN}{TDC} \times 100 \quad (62)$$

Donde *TFN* corresponde al porcentaje de tramas de falsos negativos obtenidos de una señal sonora; es decir el número de tramas que se detectan como silencio cuando en

realidad son tramas sonoras y TDC corresponde al número de tramas totales que contiene la señal sonora.

- Trama de falsos positivos: con esta métrica se comprueba cuántas tramas de una señal son detectadas como tramas sonoras cuando en realidad no son sonoras; es decir con esta métrica se indica cuántas tramas de silencio son detectadas como sonoras cuando en realidad son silencios[91]. Viene descrita en términos porcentuales por:

$$TFP = \frac{TFP}{TDC} \times 100 \quad (63)$$

Donde TFP corresponde al porcentaje de tramas de falsos positivos obtenidos de una señal sonora y TDC corresponde al número de tramas totales que contiene la señal sonora; es decir el número de tramas que se detectan como silencio cuando en realidad son tramas sonoras.

- Precisión (Accuracy, Acc): Esta medida establece la relación entre el número de tramas correctamente detectados o el número de tramas totales de la señal (TDC) y la suma del número de tramas totales de la señal, el número de tramas de falsos negativos (TFN) y el número de tramas de falsos positivos (TFP) [91].

$$Acc = \frac{TDC}{(TDC+TFN+TFP)} \times 100 \quad (64)$$

Donde Acc corresponde al valor porcentual de la precisión del sistema que se mide; ofrece información sobre el número de las tramas correctamente detectadas.

5.2.4 Resultados

En este apartado se analizan los resultados obtenidos con nuestro sistema a nivel de evaluación en tramas para cada una de las cuatro bases de datos arriba propuestas. En realidad, lo que aquí se juzga es si el algoritmo detecta tramas erróneas o no, así como conocer la precisión del mismo. Es decir, valorar su rendimiento o eficiencia sujeto al error que hace en detectar tramas sonoras bajo distintas condiciones ruidosas.

Por simplicidad aquí solo se va a presentar las gráficas que resultan de realizar las métricas por cada 10 pistas formadas por cinco notas. Es decir, cada base de datos en realidad tiene un total de 50 notas, y al ser tres bases de datos los evaluados entonces se trata de 150 notas evaluadas en su totalidad. Por tanto, por razones de espacio solo se mostrarán los resultados resumidos por cada base de datos y al final los resultados totales.

A continuación, se puede observar en la Figura 67 que representa los resultados obtenidos por la base de datos con un SNR igual a -5db

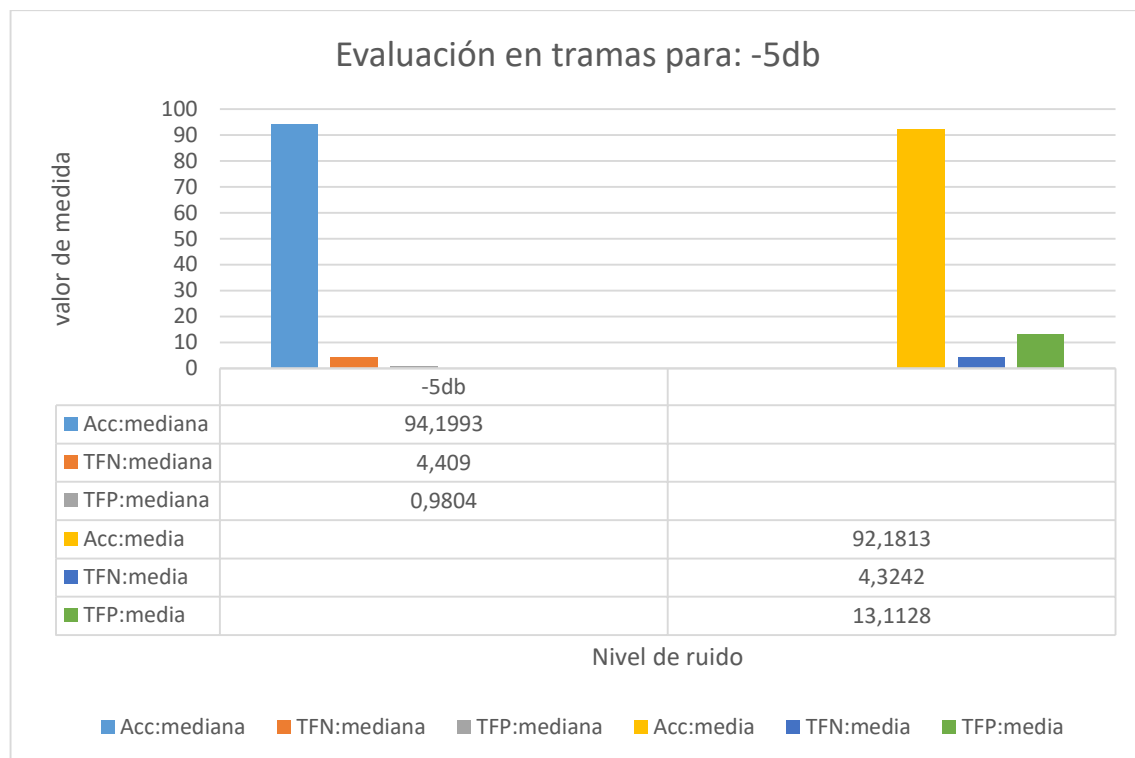


Figura 67: Resultados de la base de datos SNR igual a -5db.

En la Figura 68 se puede observar los resultados de la base de datos de 0db.

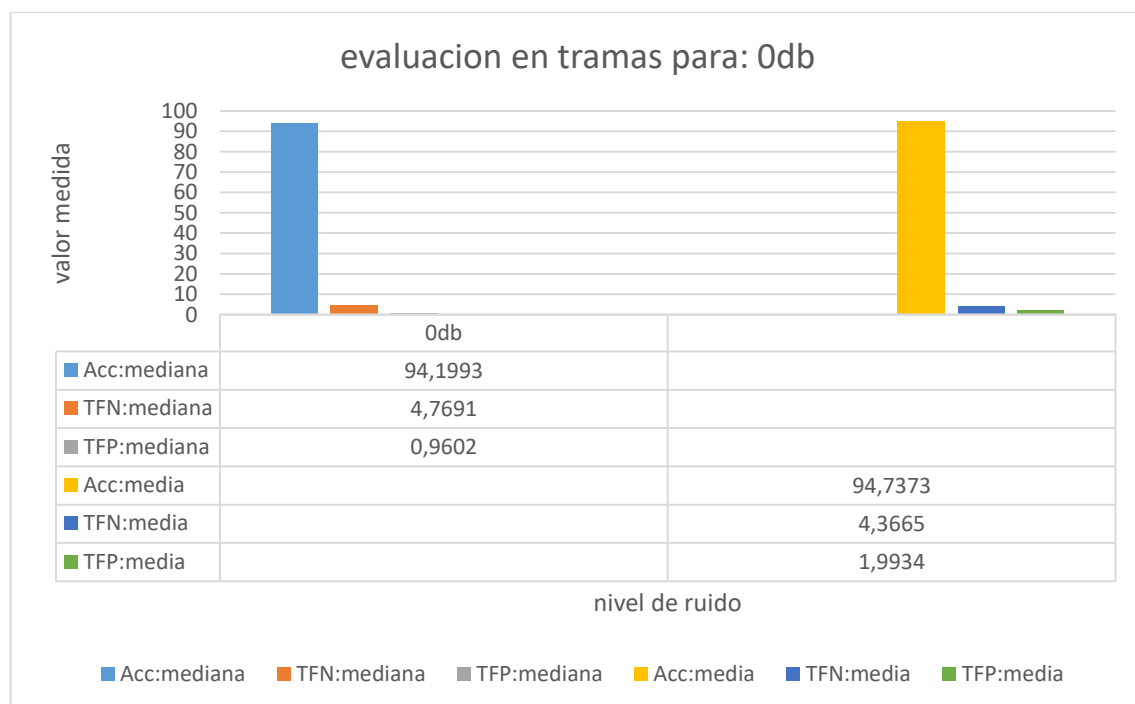


Figura 68: Resultados de la base de datos SNR igual a 0db

En la Figura 69 se puede observar los resultados de 5db.

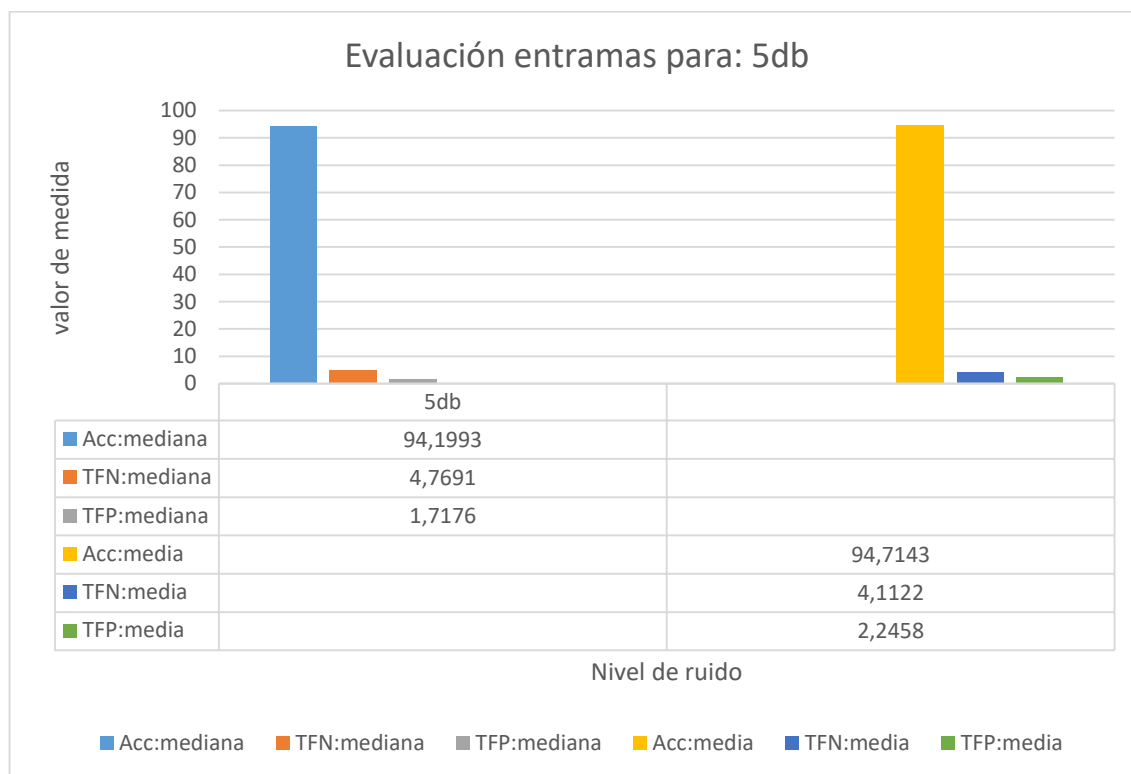


Figura 69: Resultados de la base de datos SNR igual a 5db

En la Figura 70 se puede observar los resultados totales de todas las bases de datos

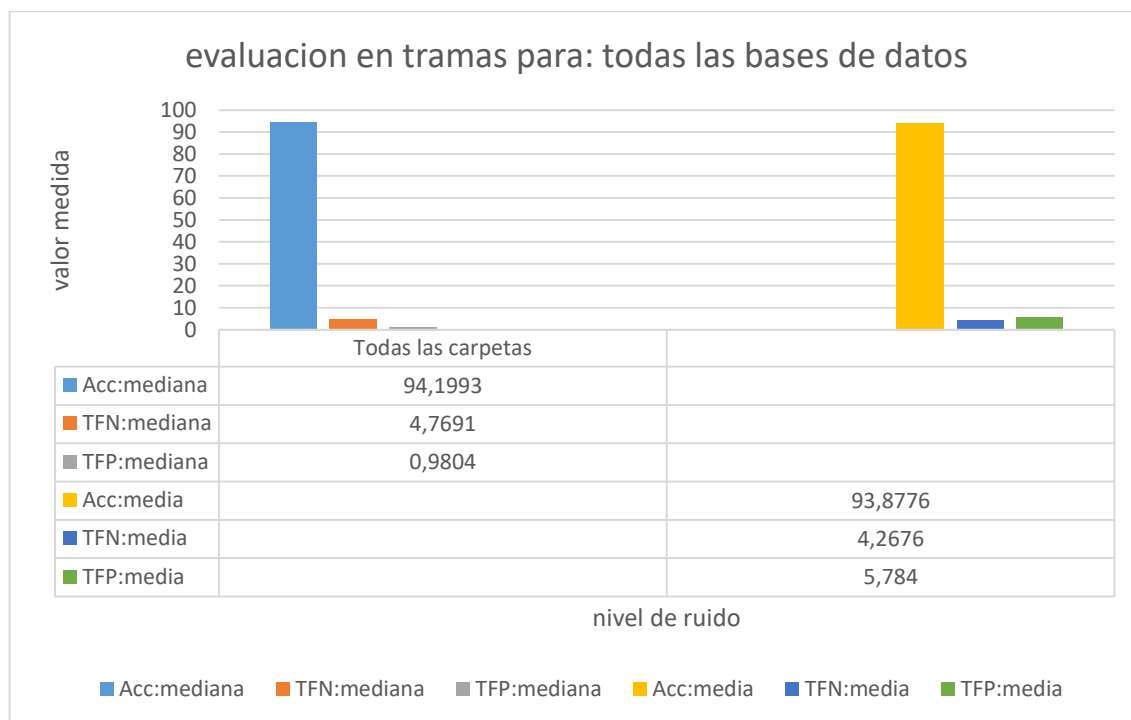


Figura 70: Resultados de la base de datos para todas las bases de datos

Los resultados que hasta ahora se han presentado se pretende con ellos comprobar y conocer el nivel de precisión y el número de tramas de falsos negativos y positivos detectados por el sistema implementado sobre distintas bases de datos y en condiciones de ruido distintas. Pero también es importante señalar que la detección de falsos positivos y negativos puede ocurrir al inicio o final de una nota. Y en este sentido también se ha realizado un estudio complementario para conocer el porcentaje en que ocurren al inicio o al final de las notas.

A continuación, se puede observar la Figura 71 que representa los resultados obtenidos de la detección de tramas de falsos negativos y positivos al inicio de las notas de la base de datos con un SNR igual a -5db

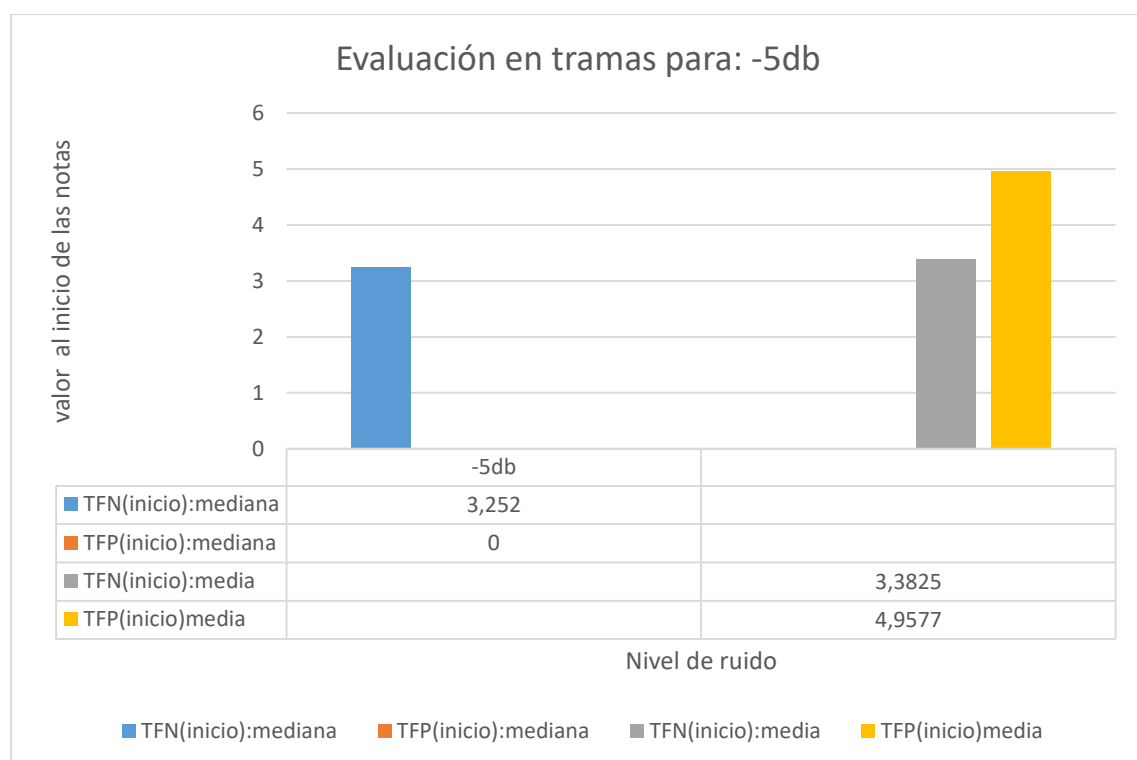


Figura 71: Resultados de las tramas de falsos negativos y falsos positivos al inicio de las notas para la base de datos SNR igual a -5db

En la Figura 72 se puede observar los resultados de la detección de falsos negativos y positivos al inicio de las notas para la base de datos de 0db.

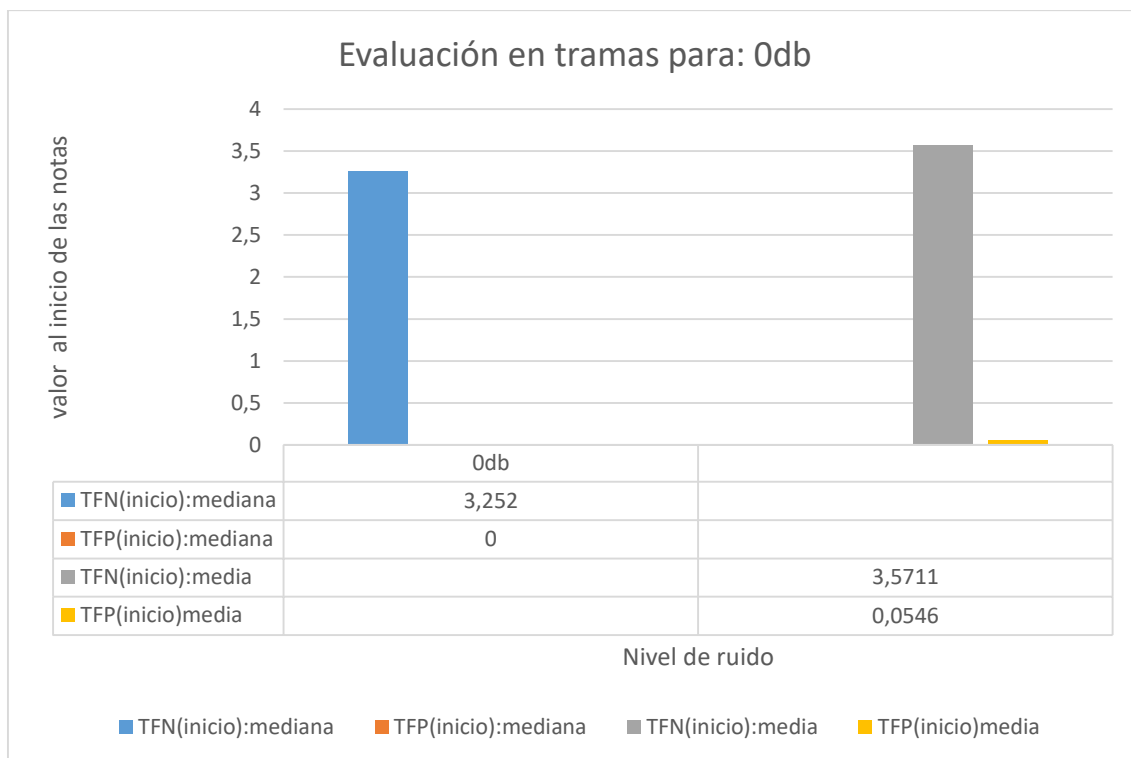


Figura 72: Resultados de las tramas de falsos negativos y falsos positivos al inicio de las notas para la base de datos SNR igual a 0db

En la Figura 73 se puede observar los resultados de la detección de falsos negativos y positivos al inicio de las notas para la base de datos de 5db

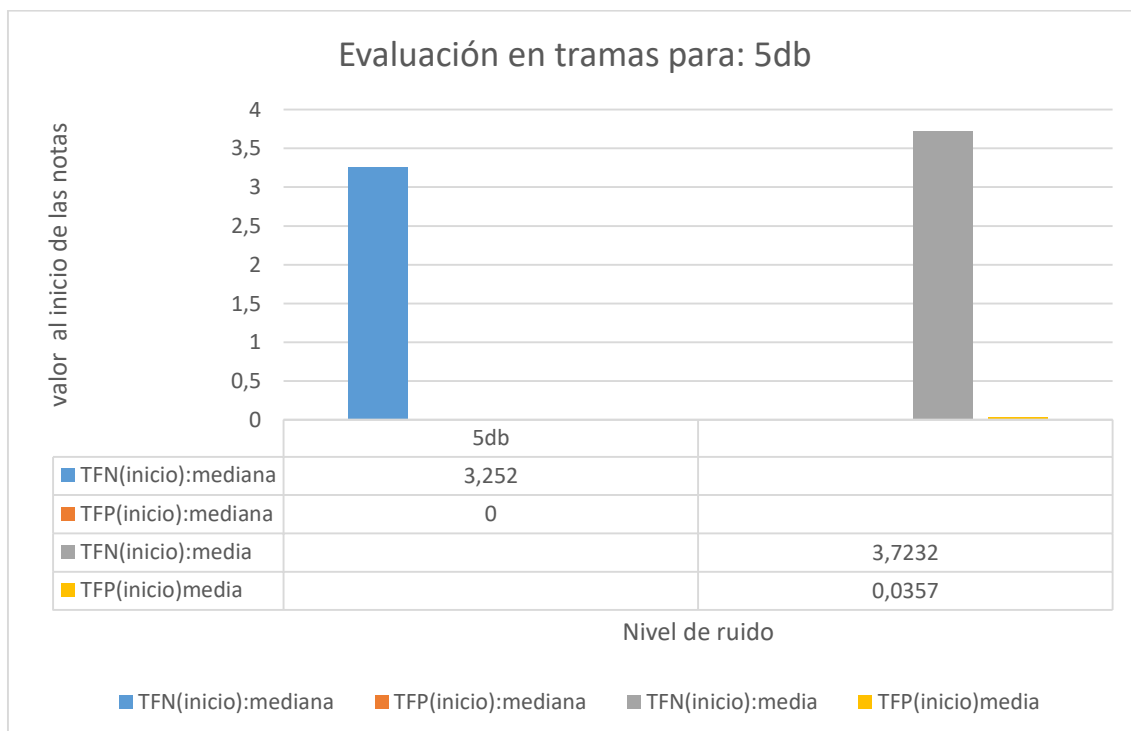


Figura 73: Resultados de las tramas de falsos negativos y falsos positivos al inicio de las notas para la base de datos SNR igual a 5db

En la Figura 74 se puede observar los resultados totales de la detección de falsos positivos y negativos al inicio de las notas para todas las bases de datos.

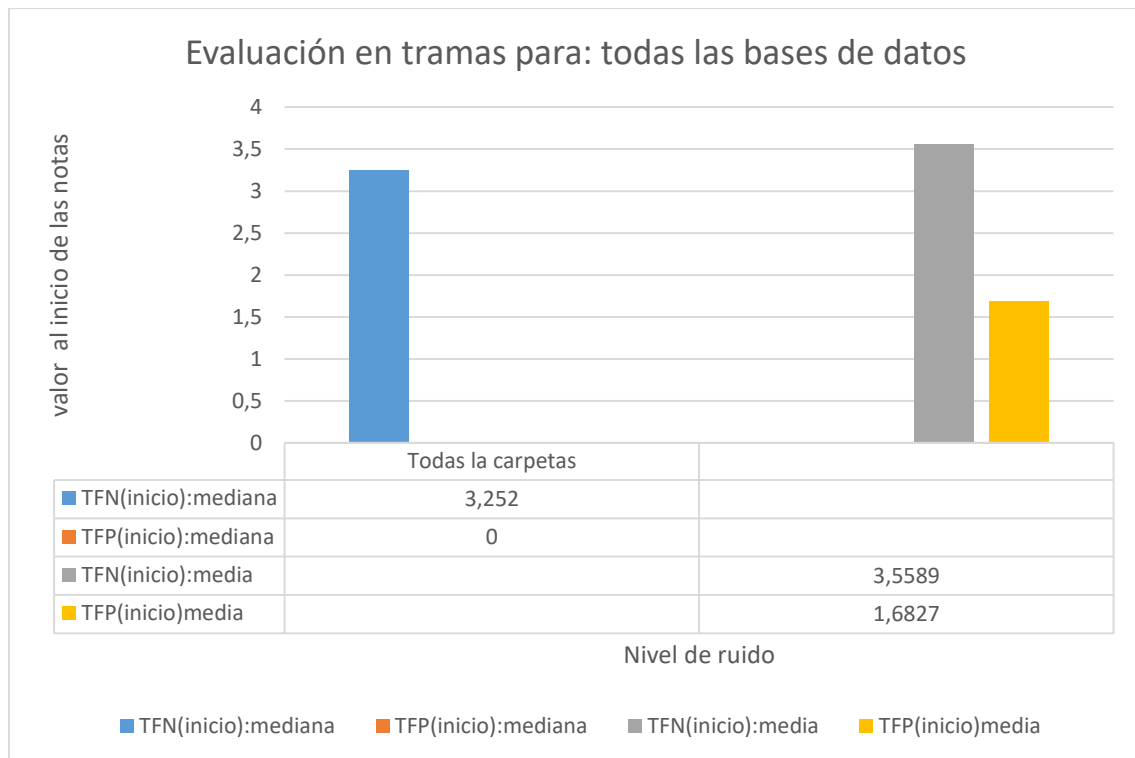


Figura 74: Resultados de las tramas de falsos negativos y falsos positivos al inicio de las notas para todas las bases de datos

A continuación, se puede observar la Figura 75 que representa los resultados obtenidos de la detección de tramas de falsos negativos y positivos al final de las notas de la base de datos con un SNR igual a -5db.

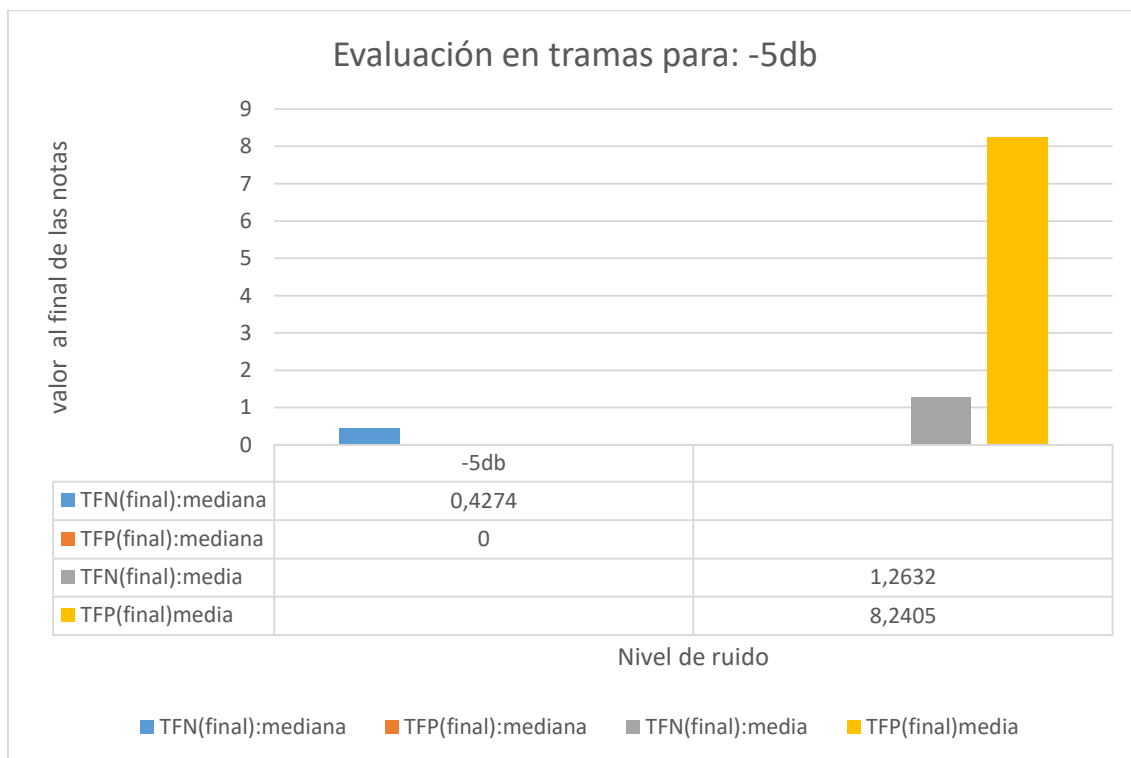


Figura 75: Resultados de las tramas de falsos negativos y falsos positivos al final de las notas para la base de datos SNR igual a -5db

En la Figura 76 se puede observar los resultados de la detección de falsos negativos y positivos al final de las notas para la base de datos de 0db.

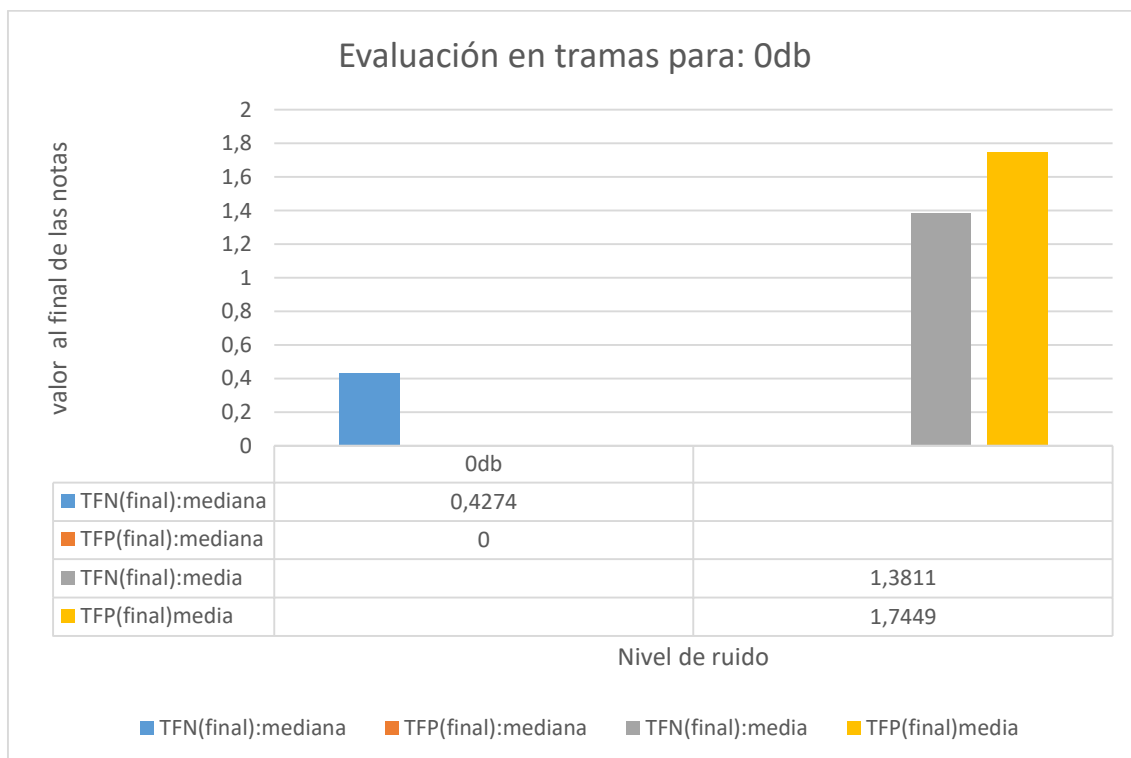


Figura 76: Resultados de las tramas de falsos negativos y falsos positivos al final de las notas para la base de datos SNR igual a 0db

En la Figura 77 se puede observar los resultados de la detección de falsos negativos y positivos al final de las notas para la base de datos de 5db.

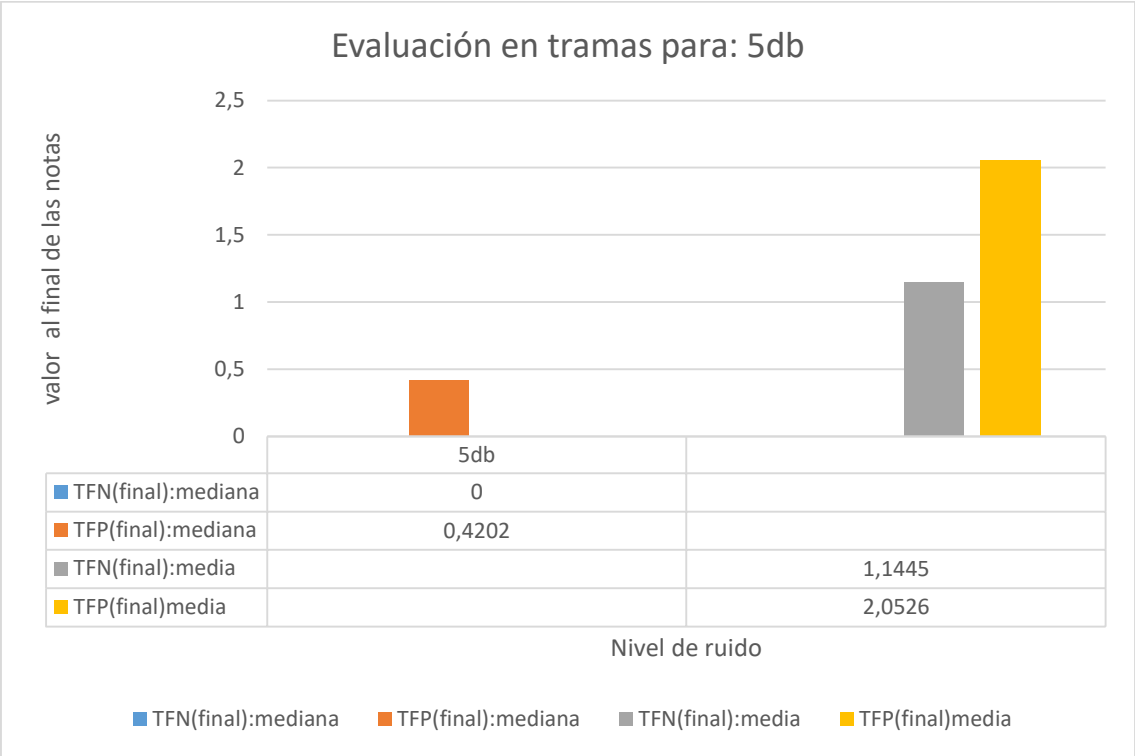


Figura 77: Resultados de las tramas de falsos negativos y falsos positivos al final de las notas para la base de datos SNR igual a 5db

En la Figura 78 se puede observar los resultados totales de la detección de falsos positivos y negativos al final de las notas para todas las bases de datos.

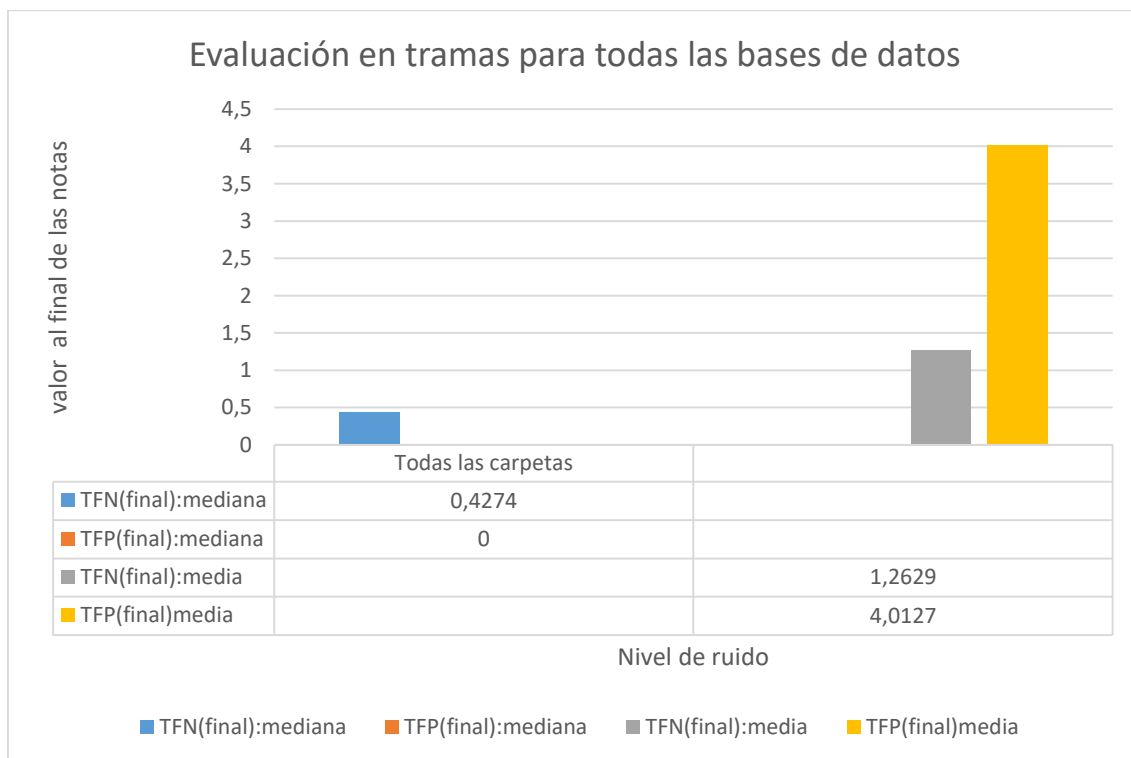


Figura 78: Resultados de las tramas de falsos negativos y falsos positivos al final de las notas para todas las bases de datos

En la Figura 79 se puede observar los resultados generales de la evaluación realizada; de tal forma se ilustran los resultados totales de la precisión del sistema así como la detección de falsos negativos y positivos tanto generales como en las distintas posiciones donde ocurren (al inicio o final de las notas) para todas las bases de datos de forma general.

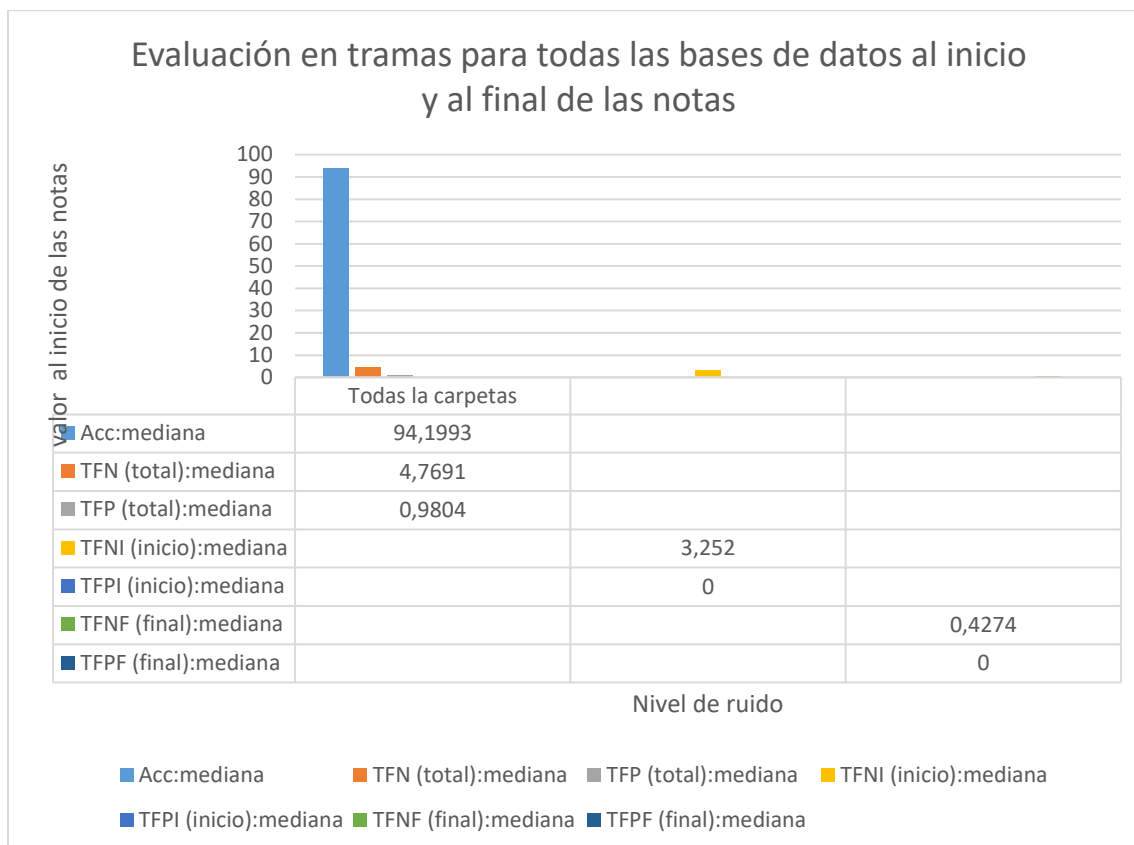


Figura 79: Resultados de las tramas de falsos negativos y falsos positivos totales al inicio y final de las notas para todas las bases de datos.

• **Discusión de los resultados**

Fijándose en la Figura 78 que representa los resultados globales se observa que el sistema detecta menos falsos positivos (0.98%) que falsos negativos (4.76%). Por lo que al detectar menos tramas de falsos positivos, implica que en un bajo porcentaje (0.98%) se detectan tramas de silencio como tramas sonoras. Por consiguiente es un buen resultado, si se tiene en cuenta que si el sistema en su mayoría detectara las tramas de silencio como sonoras (que no es el caso); entonces no se acertaría en la estimación de la frecuencia fundamental. Este resultado indica que el sistema de estimación implementado así como el umbral prefijado detectan de forma correcta las notas del espectro de la flauta en la mayoría de los casos. Este buen resultado también se reafirma con la medida de la precisión reflejada en un 94,19%; lo que se traduce en que el sistema en la mayoría de los casos acierta la frecuencia fundamental de las tramas sonoras que componen las notas de la flauta.

La detección de falsos negativos y falsos positivos que provocarían una estimación equivocada o desacertada de la frecuencia fundamental de las tramas sonoras se puede dar tanto al inicio como al final de la misma.

Aparecen más tramas de falsos negativos al inicio de las notas evaluadas en un porcentaje de 3.25% que al final de las mismas con un porcentaje de 0.42 % (ver Figura 78). Este error se puede dar como bueno porque fijándose en la energía instantánea de las señales sonoras se puede observar que ninguna nota musical comienza de forma abrupta, sino que la envolvente de la nota va siendo creciente con el tiempo por esto se comete este error; pero aun así en menor medida que se consideraría despreciable (3.25%).

En la tabla 19 se puede observar el número de tramas que tienen cada una de las cinco notas que forman las diez distintas pistas evaluadas bajo distintas condiciones de nivel de ruido.

	Pista1	Pista2	Pista3	Pista4	Pista5	Pista6	Pista7	Pista8	Pista9	Pista10
Nota1	132	123	112	114	112	102	117	114	123	102
Nota2	119	114	123	109	123	101	107	109	114	101
Nota3	119	118	107	106	107	106	114	106	118	106
Nota4	113	106	111	101	111	117	111	101	106	117
Nota5	113	102	112	101	112	125	105	101	102	125
TOTAL	596	563	565	531	565	551	554	531	563	551

Tabla 19: Número de tramas totales por pista

Si se suman todas las tramas totales de las diez pistas (596, 563, 565, 531, 565, 551, 554, 531, 563, 551) resulta que se tendría 5570 tramas por base de datos; por lo que al haber realizado la evaluación para tres bases de datos con diferentes ruidos, entonces multiplicado por tres se tendría un total de 16710 tramas evaluadas. Y como se ha fijado que una trama son 1024 muestras, por tanto se trata de haber evaluado 17111040 muestras que transformados a tiempo por la relación con la frecuencia de muestreo (44100 Hz) (ver ecuación 39) se tendría que se ha evaluado un total de señal de 388 segundos. Por lo tanto al tener un porcentaje de 3.25% de falsos negativos al inicio para las tres bases de datos evaluadas se traduce en que el umbral establecido se carga al inicio un total de 12 segundos. Y al tener un porcentaje de 0.42% de falsos negativos al final significa que se carga un total de 1 segundo.

Por otra parte los falsos positivos representados por el 0.98% (ver Figura 78) que es un error despreciable son cometidos al final de la nota más que al inicio debido a que en realidad las notas no terminan de forma abrupta y tienden a expandirse muestras al final de la nota.

También se puede deducir de los resultados que cuanto mayor es la SNR, mejores son los resultados que se obtiene. En el caso de SNR=-5db la trama de falsos positivos representan una mediana de 0.98% o una media de 13.11% (ver Figura 67) y para una SNR=0db representan una mediana de 0.96 % o una media de 1.9% (ver Figura 68) lo que significa que a mayor SNR en las tramas sonoras se acierta la frecuencia fundamental. Por lo que el algoritmo funciona suficientemente bien para SNR mayores de -5db; ya que para niveles de ruido peores se observa que el ruido enmascara al sonido procedente de la flauta.

En definitiva al mejorar la detección conforme mayor es la SNR; entonces para escenarios reales por encima de los 20db el sistema funcionará mucho mejor.

5.3 EJERCICIOS DEL TUTOR

El objetivo principal de este TFG es ayudar al usuario a tocar la flauta o a mejorar su nivel de experiencia; por lo que para dicho propósito se han creado los ejercicios del tutor que servirán de mecanismo de evaluación.

Estos ejercicios están formados por tres niveles de aprendizaje. El primero es el nivel principiante, seguido del nivel intermedio y luego del nivel experto. Por lo que en primer lugar se tendrá que ir seleccionando un determinado nivel de aprendizaje para realizar las pruebas.

Cada nivel está formado por diez partituras distintas. Es por eso que en segundo lugar se tendrá que ir seleccionando las distintas partituras que conforman el nivel en cuestión, para llevar a cabo el entrenamiento para cada partitura determinada.

Para realizar los ejercicios del tutor se tendrá que ajustar, según le convenga al tutor, los parámetros de error por nota; es decir establecer cuántos segundos se puede equivocar el alumno en tiempo y cuanto se puede equivocar en frecuencia; o lo que es lo mismo establecer un error que se le admite poder cometer tanto en tiempo así como en frecuencia (que en principio son de dos segundos y 40 Hz; aunque para hacer más flexible el aprendizaje se le permite al alumno la opción de establecer las condiciones que considere oportunas).

Por tanto para realizar las pruebas se tendrá que ir tocando y grabando las distintas notas que contiene la partitura que visualmente aparecen en la ventana de representación.

Entonces el tutor procederá a ponerle una calificación según el resultado obtenido en cada nota tocada de la partitura. El método de evaluación del tutor; es decir el tutor para dar una determinada calificación, tendrá en cuenta el resultado obtenido tanto en tiempo como en frecuencia de la nota tocada; es decir previamente procederá a realizar un análisis o evaluación en tiempo como en frecuencia de las notas que ha tocado el alumno. Lo que se

traduce en que el resultado que se vaya a obtener por cada nota tocada de una partitura, dependerá del análisis o evaluación del número de aciertos y fallos en tiempo como en frecuencia conseguidos por el alumno.

Un acierto es considerado como total para una nota tocada siempre que se acierte en tiempo y en frecuencia. Por lo que la calificación final sería la media de todas las calificaciones obtenidas en el nivel; es decir la media de las calificaciones obtenidas por partitura dentro del nivel.

Una vez realizadas las pruebas para las diez partituras que forman el nivel; el alumno pasará de un nivel de aprendizaje al siguiente siempre y cuando se cumpla que la media de las calificaciones obtenidas por partitura dentro de este nivel sea mayor o igual que cinco y se haya sacado como mínimo un cuatro en todas las partituras evaluadas; con el fin de asegurarse el tutor que el alumno ha realizado un nivel más o menos aceptable en todas las partituras.

Un ejemplo ilustrativo de los ejercicios del tutor se puede observar en el Figura 80 donde ajustando un error de dos segundos y 40 Hz se observa que el resultado conseguido por el alumno en esta determinada partitura es satisfactorio porque se consigue un acierto del cien por cien; es decir tres aciertos en tiempo y frecuencia y cero fallos para las tres notas que forman esta partitura, obteniendo así una calificación igual a diez puntos.

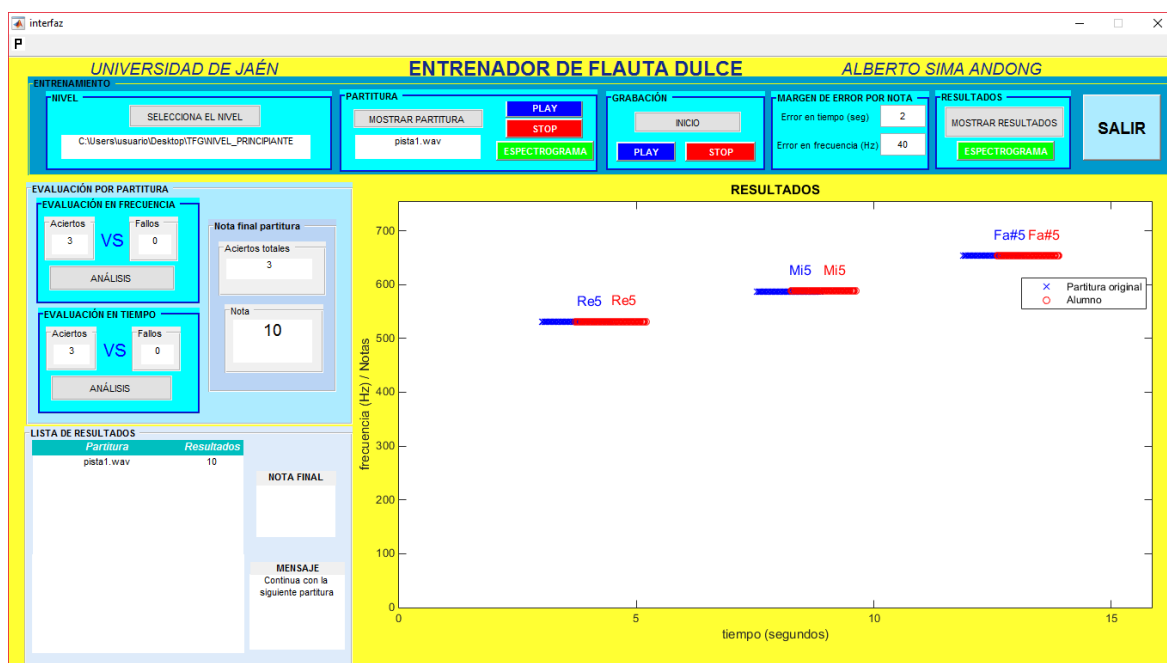


Figura 80: Ejemplo de funcionamiento de los ejercicios del tutor con un acierto del 100 %

Otro ejemplo se ilustra en la Figura 81 donde para la prueba se ha establecido el mismo margen de error en tiempo y en frecuencia. Se puede observar que para este caso a diferencia del anterior se tiene un fallo en frecuencia en solo una nota. Por lo que en tiempo se tendrían tres aciertos y cero fallos; mientras que en frecuencia se tendrían dos aciertos y un fallo, lo que se traduce en dos aciertos totales (considerando como acierto total, las notas en que se acierta tanto en tiempo como en frecuencia).

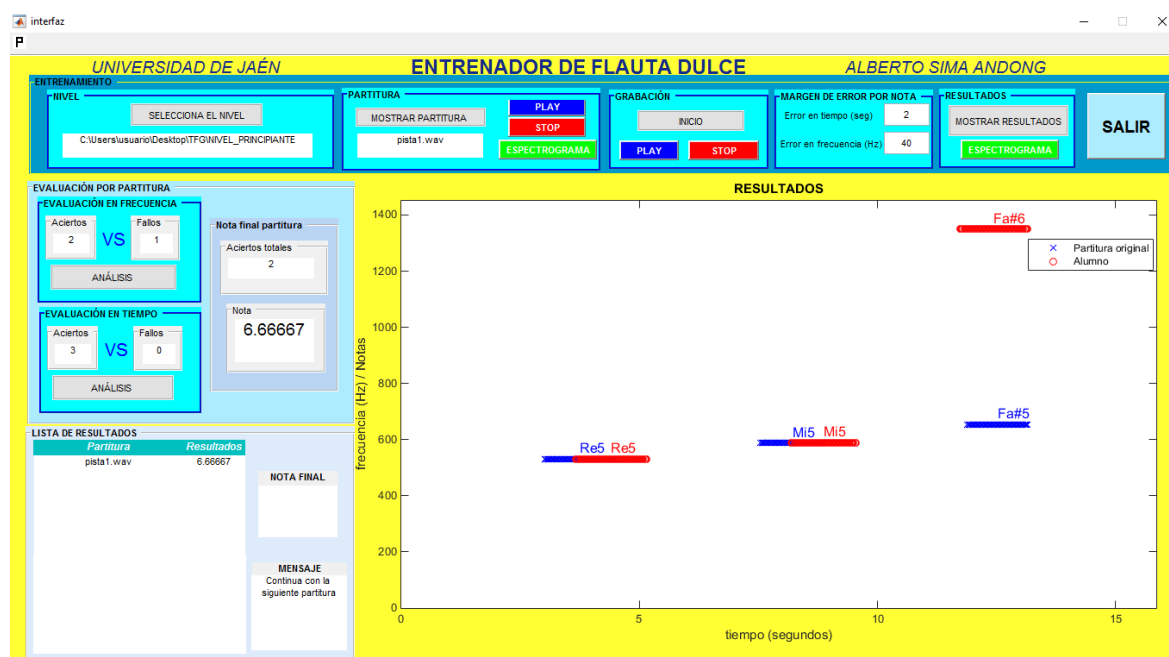


Figura 81: Ejemplo de funcionamiento de los ejercicios del tutor con un acierto del 66.7%. El alumno consigue dos aciertos totales y un fallo.

En la Figura 82 se tiene un ejemplo sobre la condición de superar un determinado nivel de aprendizaje donde se puede observar que en el listado de puntuaciones las calificaciones que se han conseguido en las distintas partituras son todas iguales o superiores que cinco por tanto se cumple con la condición impuesta y en este caso se puede observar la anotación del tutor informando al alumno que puede pasar de nivel. Mientras que en la Figura 83 se puede observar el ejemplo de que, aunque la media obtenida al finalizar el nivel es superior a 5, existe una partitura donde la media es inferior a 4 (pista 3) y por lo tanto no se cumplen las dos condiciones. Mostrándose en este caso la necesidad de repetir el mismo nivel.

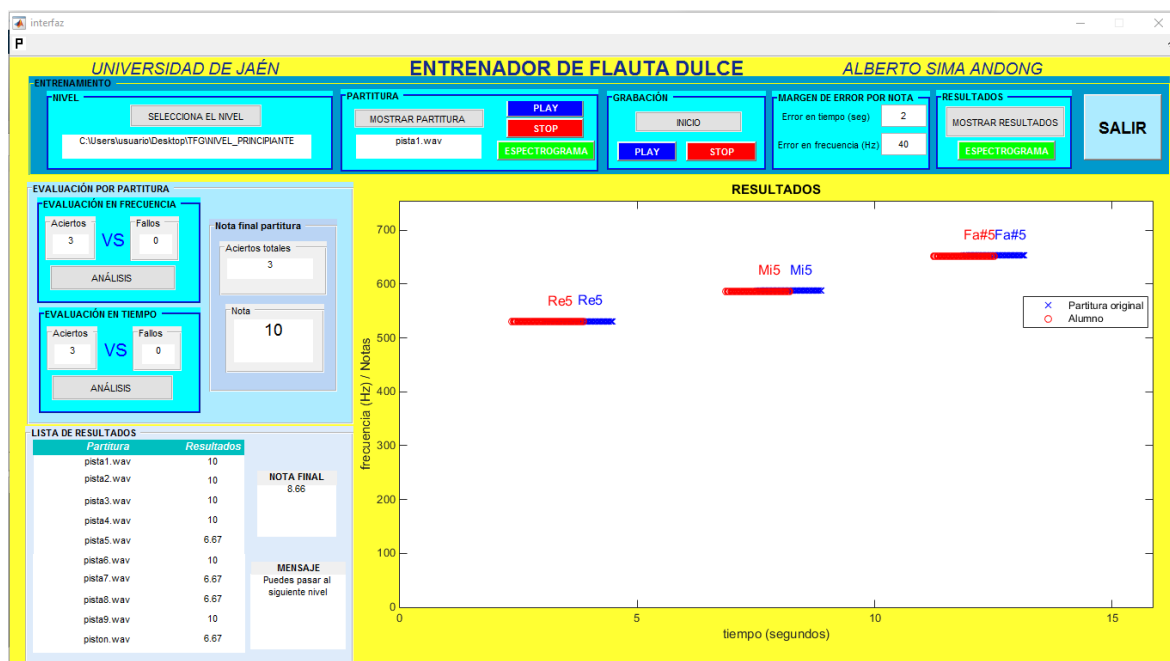


Figura 82: Ejemplo del funcionamiento de los ejercicios del tutor donde se le informa al alumno que puede pasar de nivel

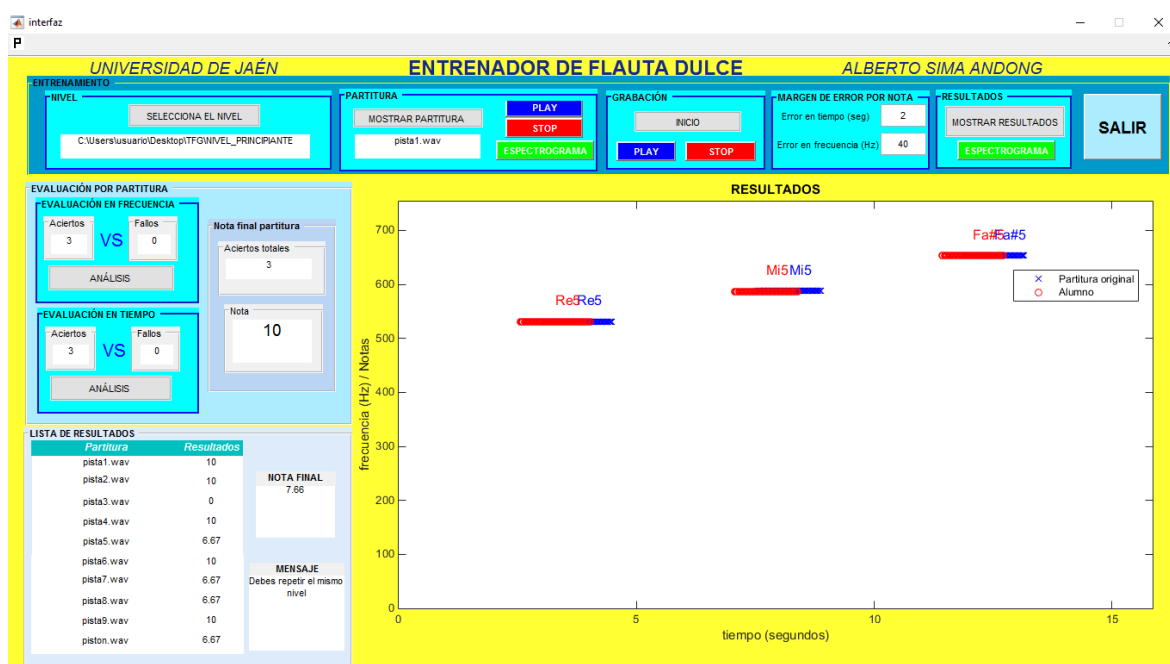


Figura 83: Ejemplo del funcionamiento de los ejercicios del tutor donde se le informa al alumno que no puede pasar de nivel

6 CAPITULO 6. CONCLUSIONES

Después de verificar el funcionamiento del algoritmo de estimación de pitch, Yin, utilizado para la implementación de este sistema, y tras analizar las pruebas, evaluar los resultados mediante varias métricas se puede determinar por tanto que se han alcanzado con éxito los objetivos marcados en este TFG. De tal forma que se ha creado un sistema que funciona de forma automática, capaz detectar las notas tocadas de la flauta dulce, así como marcar los tiempos que se han empleado para lo mismo con el fin de estimar la nota tocada por el usuario. En definitiva, se puede afirmar que con este TFG se ha puesto a disposición del usuario una herramienta útil para el aprendizaje de la flauta.

El gran atractivo de este TFG consiste en la implementación de una herramienta que sirve al usuario de entrenador orientado a la flauta; sobre todo, si se tiene en cuenta su carácter automático en su funcionamiento, que se manifiesta cuando se es capaz de detectar y estimar eficientemente y de forma autónoma las tramas correspondientes de las notas tocadas con la flauta.

La razón de utilizar distintas bases de datos parte de la necesidad de evaluar nuestra herramienta en un entorno real, ya que en el mundo real se cuenta con el ruido como factor indeseable. Por tanto después de evaluar nuestro sistema adecuando un entorno real con distintos niveles de SNR, se puede concluir que la estimación de la frecuencia fundamental que realiza nuestro sistema para ámbitos de ruido normales es muy acertada con una precisión del 94%; lo que se resume en que el sistema implementado es muy robusto frente al ruido.

Por otra parte aun en condiciones con ruido de fondo, nuestra herramienta funciona de forma un tanto aceptable en base a los resultados obtenidos de la evaluación en frecuencia cuando se asigna como frecuencia fundamental de la nota, aquella que es la más repetitiva de las tramas sonoras (cálculo de la frecuencia fundamental mediante la moda matemática); lo cual significa que el umbral dinámico establecido para la detección de tramas sonoras funciona de forma correcta, posibilitando quedarse mayoritariamente con la parte de la trama correspondiente a la zona instrumental de la flauta; evadiendo considerablemente el ruido. Lo que se entiende mejor si se tiene en cuenta que una señal con SNR -5db, es un ruido considerable para entornos de grabación de música, pero nuestro sistema aun en estas condiciones es capaz de detectar con éxito las notas tocadas con la flauta. Aunque se debe señalar que lo cual no implica que funcione correctamente para entornos peores de ruido, porque solo se ha realizado la evaluación teniendo como peor de los casos un entorno con SNR igual a -5db.

7 CAPITULO 7. LÍNEAS DE FUTURO

Este trabajo fin de grado ha consistido en realizar una herramienta de ayuda a la educación musical del instrumento flauta dulce basado en el análisis de señales sonoras monofónicas. Se ha comenzado realizando el proceso de grabación para poder capturar la señal analógica procedente de la flauta mediante el proceso de conversión de analógico a digital (A/D), se ha inventariado la señal grabada con el fin de analizar tramas periódicas de la señal, se ha realizado el proceso de detección de pitch mediante un algoritmo en el dominio del tiempo, el algoritmo Yin, para ello primero se ha tenido que realizar la detección del inicio y final de la nota procesada (onset y offset) mediante el análisis de la distribución de energía; es decir observando los máximos locales o cambios bruscos de la señal tal como aconseja el mencionado algoritmo [17], se ha transcrito a notación musical midi el pitch estimado, y por último se han cumplido los objetivos por los que se creó este TFG pudiendo analizar así los errores y aciertos en tiempo y en frecuencia que se cometen a la hora de aprender a tocar este instrumento. Todo este estudio lleva a la finalización de este trabajo fin de grado; pero se debe tener en cuenta que el mundo está en constante evolución por lo que este TFG es mejorable como todo en la vida. Por lo consiguiente se puede trasladar este TFG a diferentes contextos. Por ejemplo que la fuente no sea de una señal monofónica sino polifónica o que para el mismo propósito se pueda implementar algoritmos espectrales; también se podría trabajar con otras fuentes independientes así como realizar más estudios sobre la detección del onset y el offset.

El primer contexto supone añadir más dificultad a la problemática de la estimación de pitch si se tiene en cuenta que las señales esta vez serian una mezcla de varias fuentes o distintos instrumentos lo que supone tener que complementar la información que se ha manejado hasta ahora en este TFG. Lo que supone tener que hacer un estudio de la información espacial que permitiría realizar la separación de las distintas fuentes que forman la señal polifónica. Por lo que si además de esta nueva información, la espacial, se cuenta con la información temporal y espectral que se ha manejado en este TFG se puede obtener resultados óptimos para una estimación de pitch en el escenario de una señal polifónica. Por lo que realizar la estimación de pitch para señales polifónicas usando la complementariedad de ambas informaciones puede ser muy eficiente.

En el segundo contexto, utilizar algoritmos espectrales para la estimación de pitch, es ampliamente demandado en la industria musical actual. Este planteamiento puede tener varias coincidencias con el algoritmo implementado en este TFG, si bien tenemos en cuenta que el algoritmo temporal implementado en este TFG se difiere de los algoritmos espectrales

respecto a la información que se puede extraer para calcular la frecuencia fundamental o el pitch. Pero la información que se puede manejar para ambos algoritmos puede ser complementaria ya que existe una relación entre los dominios de tiempo y frecuencia mediante la Transformada de Fourier. Por lo que la estimación de pitch a pesar de ser posible usando el algoritmo de Yin implementado en este TFG, también se podría llevar a cabo implementando algoritmos espectrales ya que el primero analiza el periodo de la señal en el dominio temporal y el segundo bloque de algoritmos lo hace analizando la periodicidad en el dominio de la frecuencia. Aunque también han habido y se puede realizar estudios e implementaciones para la estimación de pitch con algoritmos perceptuales analizando la periodicidad en tiempo de la envolvente de la señal.

En el tercer contexto se considera que este TFG ha sido abordado mediante el algoritmo de estimación de pitch, Yin, usando señales monofónicas teniendo como fuente la flauta dulce. Pero es razonable pensar que esta misma técnica se puede implementar generalmente para estimar el pitch de otras fuentes (la guitarra, el violín; etc); de forma independiente. Por lo que queda visto que en el futuro se puede trabajar sobre la estimación de pitch de cualquier instrumento distinto a la flauta; con lo cual entre otras consideraciones se tendría que trabajar en base a la resolución temporal y la resolución frecuencial ya que cada instrumento tiene su rango espectral (ver tabla 1).

El cuarto contexto consistiría en seguir evolucionando en el sector de la detección del onset y offset realizando un estudio e implementación de varias de las técnicas innovadoras que existen en este campo; y que sobre todo se pueda realizar de forma automática, para así conociendo el inicio y final de una trama sonora se pueda extraer con absoluta precisión la información instrumental de las distintas fuentes musicales.

8 CAPITULO 8. ANEXOS

8.1 ANEXO I. MANUAL DE USUARIO

En este manual se incluyen todos los aspectos fundamentales para el manejo de la interfaz gráfica creada para este TFG.

Esta interfaz gráfica ha sido creada mediante la utilización de la herramienta GUIDE. Es una herramienta del entorno de programación Matlab pensado para la programación visual con el fin de realizar o ejecutar programas que necesiten trabajar con varios datos. Tiene características básicas similares a otros programas visuales tales como Visual Basic o Visual C++.

Este manual de usuario está enfocado como una guía donde se pretende ofrecer ayuda para poder entender el funcionamiento de la interfaz resultante de este TFG; o bien educar al usuario sobre el aprendizaje de la flauta dulce de forma didáctica e interactiva mezclando opciones visuales como audibles. Por tanto se va a detallar las instrucciones que se deben seguir para que el usuario pueda utilizar este producto fruto de este TFG; es decir se van a detallar las funciones de teclado; así como las diferentes opciones disponibles que se pueden desplegar mediante varios menús; etc.

En primer lugar en la Figura 84 se puede observar el entorno de trabajo que se despliega de esta interfaz.

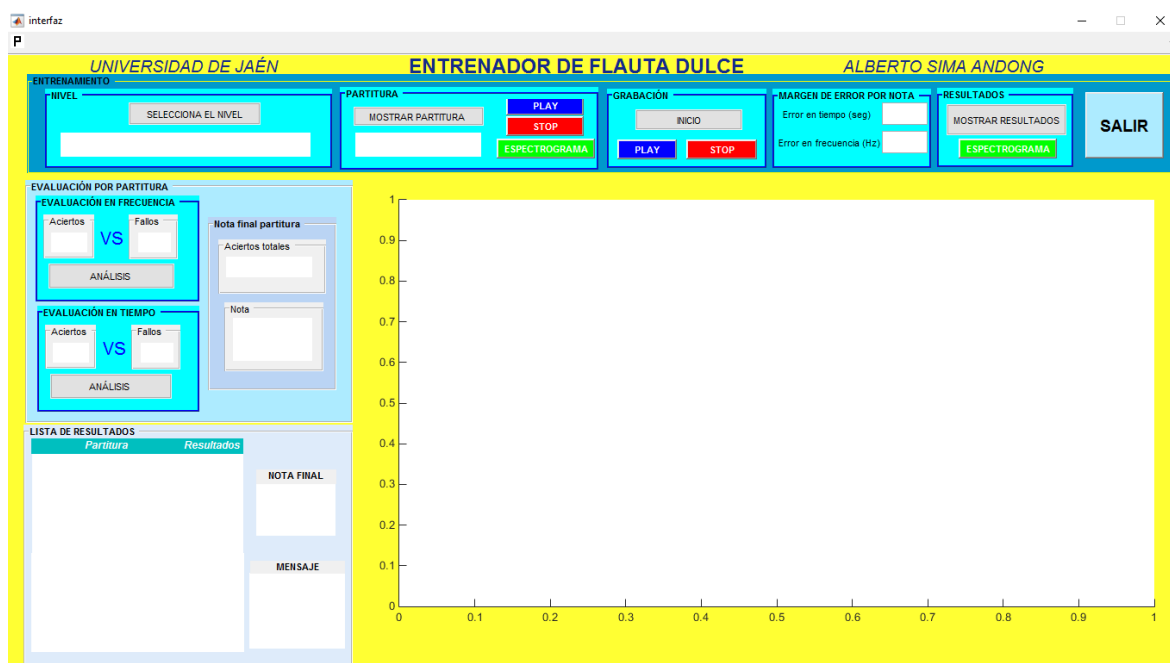


Figura 84: Entorno de trabajo de la Interfaz gráfica.

De la Figura 84 anterior se puede observar que el entorno de trabajo habilitado para esta interfaz es muy intuitivo, ameno y de fácil manejo.

Esta interfaz gráfica, tal como se puede observar en la Figura 85 está formada por tres bloques o partes visualmente diferenciados y que vienen señalizados por tres flechas de distintos colores. El color azul señala el primer bloque de la interfaz. El color verde señala el segundo bloque; mientras que el color anaranjado señala el tercer bloque de la interfaz. Esta interfaz también incluye un entorno o ventana de representación señalado con una flecha negra donde se podrán observar tanto las notas que se dispone a tocar el usuario como las resultantes de haber tocado realmente.

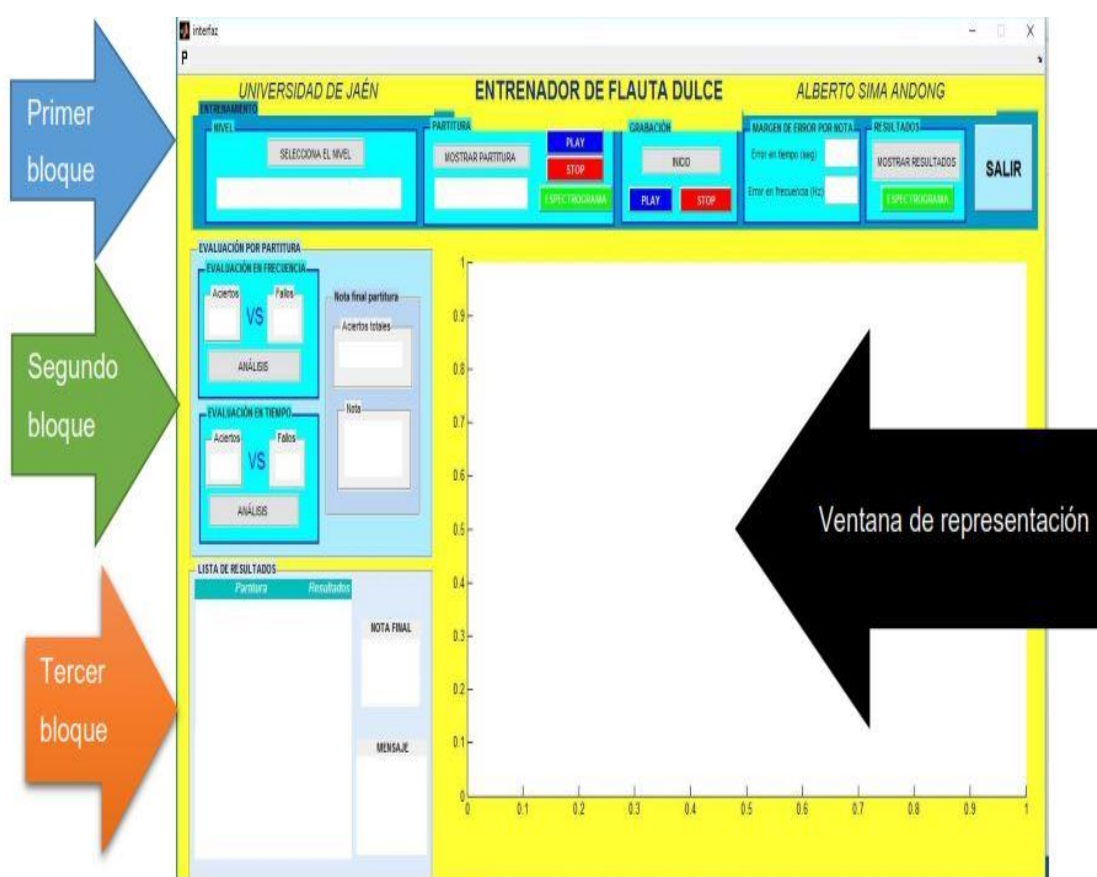


Figura 85: Interfaz gráfica mostrada en tres bloques

A continuación se va a describir el funcionamiento de cada bloque y detallar las instrucciones, las funciones de teclado así como las diferentes opciones disponibles que derivan de cada uno de los bloques en los que se divide esta interfaz.

➤ **PRIMER BLOQUE**

El primer bloque, que se ilustra en la Figura 86, es el bloque de entrenamiento que permite llevar a cabo el entrenamiento para una partitura determinada.



Figura 86: Primer bloque de la interfaz gráfica

Este bloque está dividido por cuatro secciones. En la primera sección se selecciona un determinado nivel de aprendizaje, ya sea principiante, intermedio o experto. Para seleccionar el nivel de aprendizaje, se debe accionar pulsando sobre el botón “ SELECCIONA EL NIVEL” tal como se puede observar en la Figura 87 donde se ha seleccionado el nivel principiante.

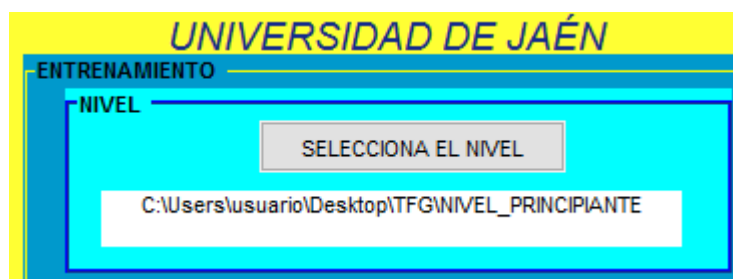


Figura 87: primera sección del primer bloque. Botón para seleccionar un determinado nivel de aprendizaje. El nivel seleccionado es el nivel principiante

En la segunda sección se pueden realizar varias ejecuciones: se puede mostrar la partitura cuya prueba se va a realizar accionando sobre el botón “ MOSTRAR PARTITURA”. Se puede escuchar, parar y mostrar el espectrograma de la partitura cuya prueba se va a realizar, accionando sobre los botones “ PLAY”, “STOP” y “ ESPECTROGRAMA” respectivamente. Y lo más importante, también se puede grabar accionando sobre el botón “INICIO”, que marca el inicio de la grabación; y también se puede escuchar o parar la nota grabada. La grabación que se realice deberá ser evaluada por el tutor bajo ciertas condiciones, para eso en esta sección se ha habilitado el margen de error por nota, donde se ajustaran las condiciones establecidas por el tutor.

En la Figura 88 se puede observar un claro de ejemplo donde tras seleccionar el nivel principiante, el alumno debe disponerse a tocar o grabar la pista 1 o primera partitura del nivel bajo un margen de error de dos segundos en tiempo y 40 Hz en frecuencia



Figura 88: Segunda sección del primer bloque. Botón para mostrar la partitura, escuchar la partitura mostrada, pararla, mostrar su espectrograma, y grabar la misma partitura bajo un margen de error de 2 segundos en tiempo y 40 Hz en frecuencia.

Para realizar la grabación de las distintas notas que forma la partitura, se hace uso de un cursor que es el que va marcando visualmente los tiempos en que se debe ir tocando cada una de las notas de la partitura. Un claro ejemplo de cómo se realiza la grabación teniendo en cuenta la evolución del curso se puede observar en la Figura 89

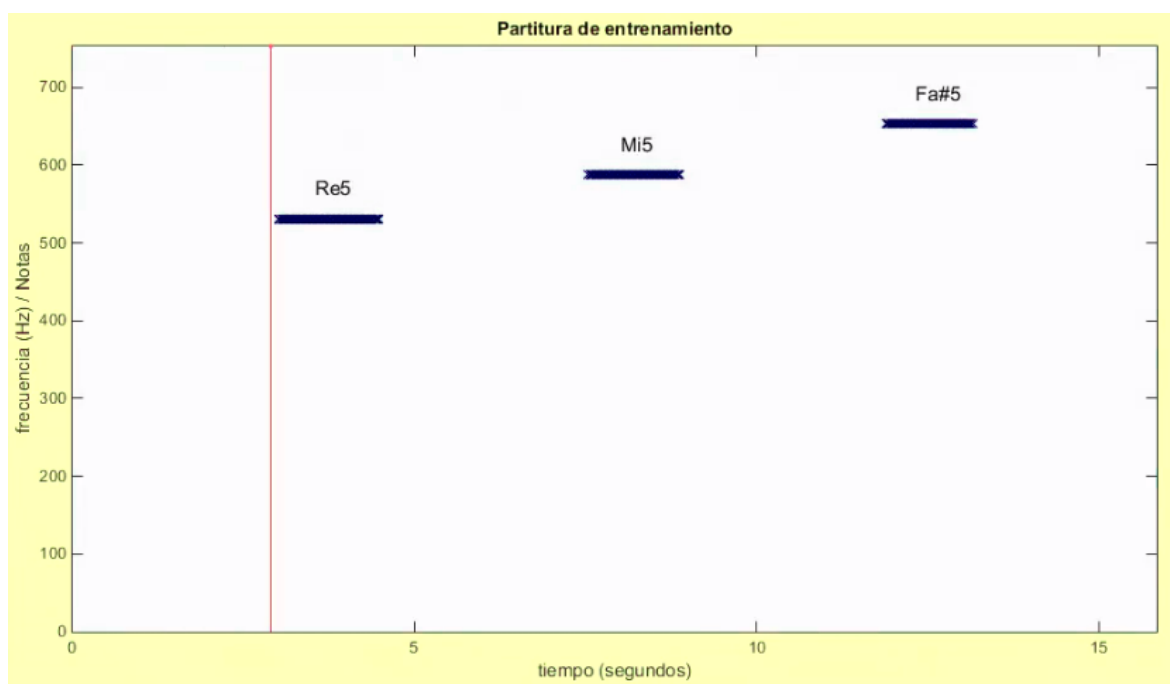


Figura 89: Segunda sección del primer bloque. Cursor que marca los tiempos en la grabación de las notas que forman la partitura

En la tercera sección del primer bloque se pueden realizar las siguientes ejecuciones: se puede mostrar los resultados; es decir visualizar sobre la misma ventana y de forma solapada la partitura que se dispone tocar el alumno y la partitura que realmente ha tocado, ambas en diferentes colores, para que el alumno pueda tener una información visual de los

posibles errores cometidos tanto en tiempo como en frecuencia; todo eso con accionar sobre el botón “ MOSTRAR RESULTADOS”. También se puede visualizar el espectrograma de la partitura grabada por el alumno, accionando sobre el botón“ ESPECTROGRAMA”; tal como se puede observar en la Figura 90

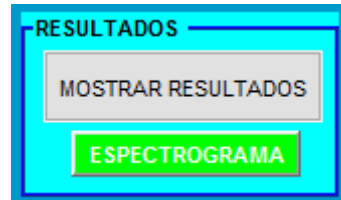


Figura 90: Tercera sección del primer bloque. Botón para mostrar resultados, y mostrar el espectrograma de la partitura grabada bajo un margen de error de 2 segundos en tiempo y 40 Hz en frecuencia.

Un ejemplo del resultado visual que se muestra en la ventana de representación tras accionar sobre el botón que ejecuta la acción de mostrar los resultados lo podemos observar en la Figura 91

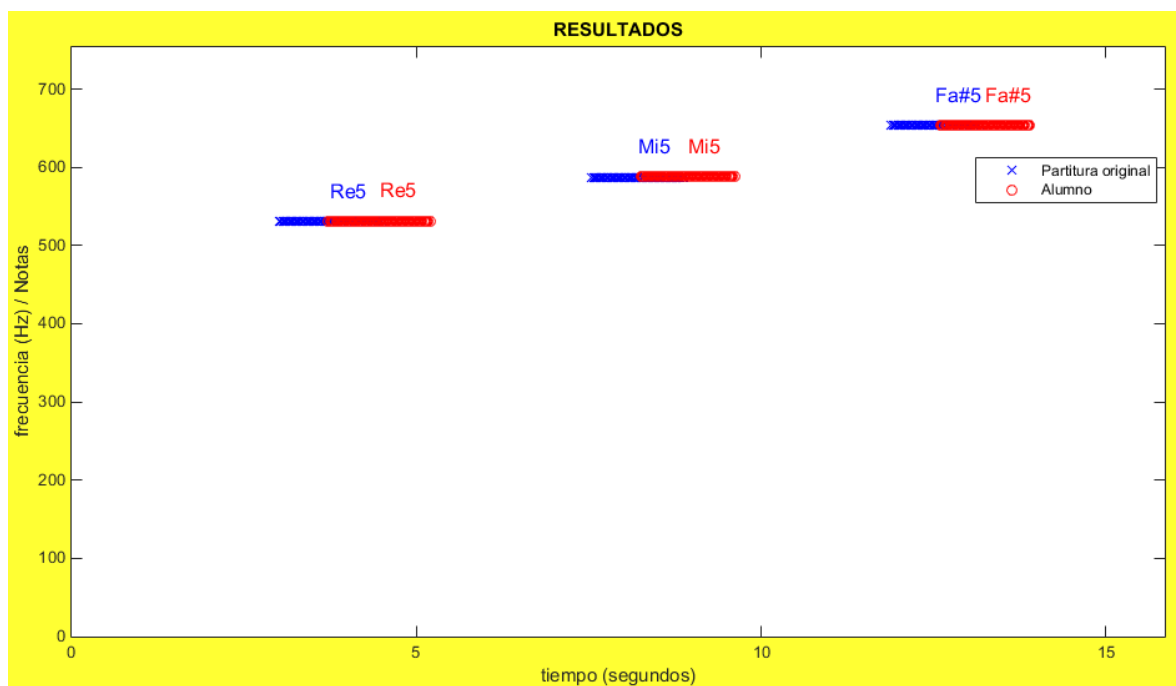
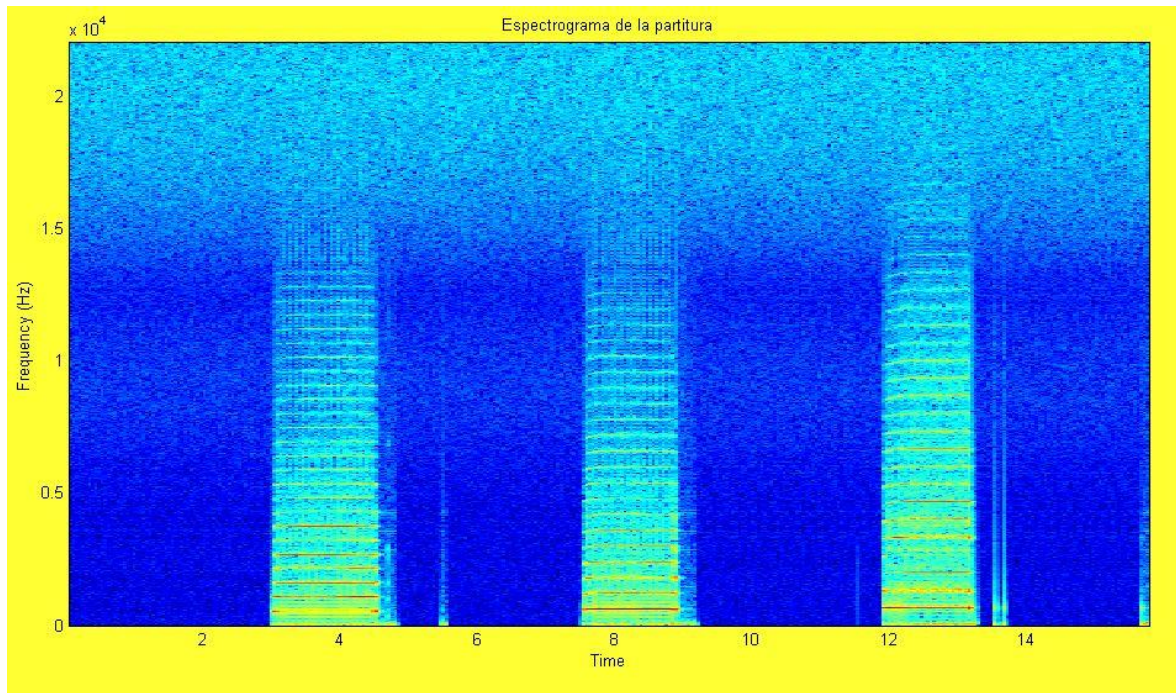


Figura 91: Resultado de la partitura grabada por el alumno en color rojo en comparación a la que se debería haber tocado realmente en color azul.

Mientras que en la Figura 92 se puede observar el espectrograma de la partitura grabada.



➤ **SEGUNDO BLOQUE**

El segundo bloque de la interfaz que se ilustra en la Figura 93 es el bloque correspondiente a la evaluación por partitura, donde se llevará a cabo una evaluación para cada una de las partituras que está tocando el alumno, disponiendo de una lista de resultados para que se vea un análisis de cómo va evolucionando el alumno.

EVALUACIÓN POR PARTITURA

EVALUACIÓN EN FRECUENCIA

Aciertos	VS	Fallos
3		0

ANÁLISIS

EVALUACIÓN EN TIEMPO

Aciertos	VS	Fallos
3		0

ANÁLISIS

Nota final partitura

Aciertos totales: 3

Nota: 10

Figura 93: Segundo bloque de la interfaz gráfica

Para cada una de las partituras se podrá llevar a cabo la evaluación en frecuencia y en tiempo. En la evaluación en frecuencia aparecerá el número de aciertos y el número de fallos en frecuencia, incluso se podrá hacer un análisis de los fallos que se han tenido en frecuencia; donde aparecerán en la ventana de representación las notas que se han fallado, cuál ha sido la que ha tocado el alumno y cuál es la que se debería haber tocado realmente.

En cuanto a la evaluación en el tiempo sería un análisis similar, aparecerán el número de aciertos y fallos en el tiempo. En caso de que haya habido fallos, nos aparecerán las notas que han tocado y las que realmente se deberían haber tocado en otro color para que se vea en que se ha equivocado. Y por último los aciertos totales los cuales para que un acierto se pueda considerar como final, se debe cumplir que sea acierto tanto en frecuencia como en tiempo.

Un ejemplo ilustrativo lo tenemos en la Figura 94, donde tras realizar la evaluación de la partitura formada por las notas (Sol5,sol6,Si5), en comparación a la prueba realizada por el alumno, se puede observar en el segundo bloque que el alumno comete tres fallos en frecuencia y acierta en tiempo en las tres notas, por lo consiguiente obtiene una calificación igual a cero al no acertar en frecuencia y en tiempo.

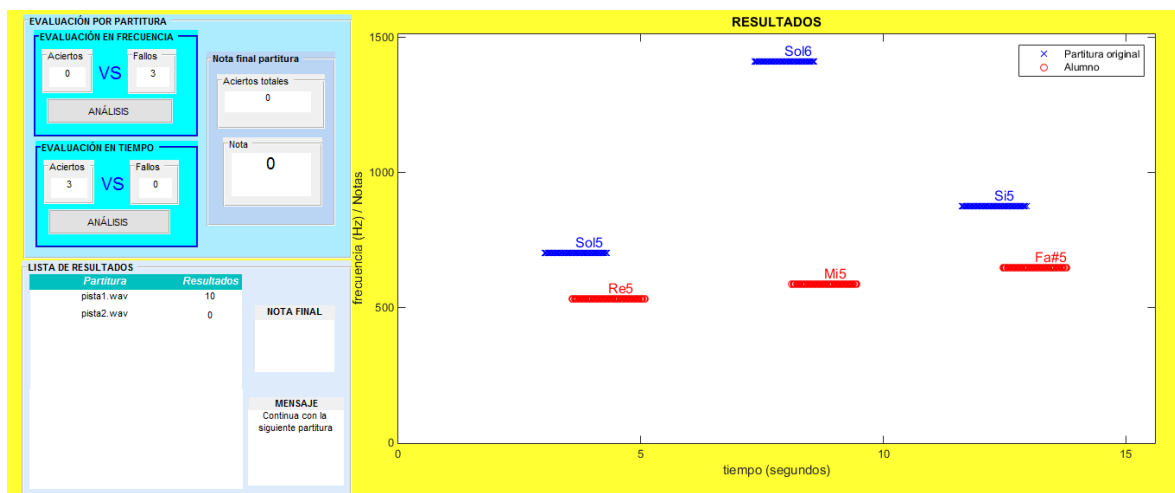


Figura 94: Segundo bloque de la interfaz gráfica: evaluación y análisis en frecuencia y en tiempo de la partitura (Sol5, Sol6, Si5).

➤ TERCER BLOQUE

El tercer bloque de la interfaz que se ilustra en la Figura 95, corresponde a la ventana de la lista de resultados.

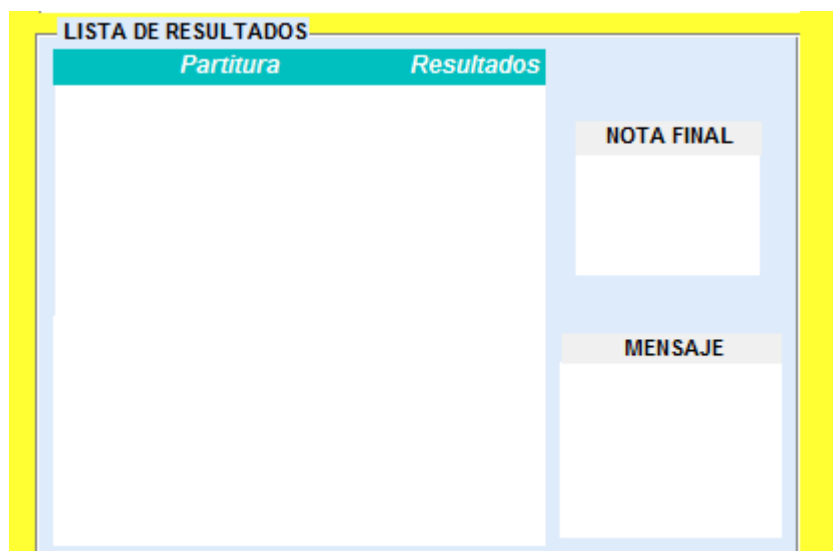


Figura 95: Tercer bloque de la interfaz gráfica

Donde se muestra el nombre de la partitura, la puntuación por partitura que se ha obtenido y finalmente cuando se haya completado la base de datos formada por diez partituras aparecerá la puntuación o nota final y un mensaje indicando si se pasa de nivel o no; de tal forma que se puede pasar de nivel cuando se cumpla que la media de las calificaciones obtenidas por partitura dentro de este nivel sea mayor o igual que cinco y se haya sacado como mínimo un cuatro en todas las partituras evaluadas; con el fin de asegurarse el tutor que el alumno ha realizado un nivel más o menos aceptable en todas las partituras (ver ejemplo en las Figuras 82 y 83).

8.2 ANEXO II. NOMENCLATURAS

TFG: Trabajo Fin de Grado

SNR:	Signal-to-noise ratio
ACF:	Short Term Autocorrelation function
FFT:	Fast Fourier Transform
SFFT:	Short Time Fourier Transform
AMDF:	Average Magnitude Difference Function
ZCR:	Zero Crossing Rate
HPS:	Harmonic Product Spectrum
PDA:	Pitch Detection Algorithm
SHS:	Subharmonic Summation Algorithm
ACC:	Accuracy (tasa de exactitud)
DFT:	Discrete Fourier Transform
RWC:	Musical Instrument Sound Database de Real World Computing
MUMS:	McGill University's Master Samples
BCD:	Bach Chorals Dataset
TFP:	Trama de falsos positivos
TFN:	Trama de falsos negativos
MIDI:	Musical Instrument Digital Interfaz
EABS:	Error absoluto
EREL:	Error relativo
F_0 :	Frecuencia fundamental
F_{0n} :	Frecuencia fundamental de toda la nota
x:	señal de entrada
F_s :	Frecuencia de muestreo
SALTO:	Longitud del bloque de muestras
TSALTO:	Tiempo que dura el bloque de muestras
INICIOS:	Indica el inicio de las tramas sonoras
FINALES:	Indica el final de las tramas sonoras
MIDIN:	Valor que se asigna al midi de todo la nota.

8.3 ANEXO III: ÍNDICE DE FIGURAS

<i>Figura 1: Esquema general del Trabajo Fin de Grado [3]</i>	6
<i>Figura 2: Representación del sonido como una onda de presión. Fuente [6]</i>	10
<i>Figura 3: Tono puro de 50 Hz de un segundo de duración</i>	10
<i>Figura 4: Tono puro de 50 Hz de 0.1 segundos de duración</i>	11

Figura 5: Suma de dos tonos de 50 Hz y 100 Hz	11
Figura 6: Ejemplos de señales periódicas y cuasi-periódicas. Las figuras (a) y (b) muestran una señal periódica s_1 de una senoide con $T_0 = 6.66\text{ms}$ y $f_0 = 150\text{Hz}$ en el dominio del tiempo y la frecuencia respectivamente. Las figuras (c) y (d) muestran una señal periódica s_2 que está compuesta por la suma de dos sinusoides. La primera de ellas con $T_0 = 6.66\text{ms}$ y $f_0 = 100\text{Hz}$, y la segunda con $T_0 = 5\text{ms}$ y $f_0 = 200\text{Hz}$. La señal cuasi-periódica s_3 está generada por la nota Si_4 con $T_0 = 2.02\text{ms}$ y $f_0 = 495\text{Hz}$ de la flauta dulce de la base de datos RWC.	16
Figura 7: Representación de la duración de una nota musical. Fuente [18].	19
Figura 8: Curvas isofónicas. Fuente [22].	19
Figura 9: Notación simbólica de una partitura. Fuente [23].	20
Figura 10: A la izquierda señal continua y a la derecha señal muestreada. Fuente [33]	24
Figura 11: Rango de frecuencias de la flauta	25
Figura 12: Señal representada en tiempo (de duración 220ms) y muestreada a una frecuencia de muestreo (f_s) igual a 4698 Hz.	27
Figura 13: Señal representada en tiempo (de duración 23.22ms) y muestreada a una frecuencia de muestreo igual a 44100 Hz.	27
Figura 14: Diferencia de tiempo del bloque o trama	28
Figura 15: Solapamiento del 50% entre tramas. Fuente [36].	29
Figura 16: Flauta dulce. Fuente [40].	31
Figura 17: Función de autocorrelación trama 70 de la nota si_4 de la flauta dulce	36
Figura 18: Función de diferencias 70 de la nota si_4 de la flauta dulce	37
Figura 19: Pasos del algoritmo de YIN.	41
Figura 20: Serie armónica de la nota C2. Fuente [66]	43
Figura 21: Ejemplo del funcionamiento de la interfaz gráfica cumpliendo los objetivos marcados: análisis en frecuencia y en tiempo.	48
Figura 22: Esquema del sistema a implementar en el TFG.	53
Figura 23: Ventana Audacity sin sonido cargado	58
Figura 24: Pista de 25 notas en audacity.	59
Figura 25: Selección de la nota a separar (usando tijeras) de la pista	59
Figura 26: Resultado de separar la nota marcada con el círculo	60
Figura 27: Gráfica de flujo de señales para FFT de cuatro puntos. Fuente [88]	62
Figura 28: Ejecución del programa en Matlab.	63
Figura 29: Selección de notas en la base de datos: notas_unicas	64
Figura 30: Frecuencia fundamental (situado en el cursor) de Nota2 de la base de datos (notas únicas)	65
Figura 31: Frecuencia fundamental (situado en el cursor) de Nota3 de la base de datos (notas únicas)	66
Figura 32: Nota Si_4 representada en tiempo (izquierda) y en tramas (derecha)	69
Figura 33: Ejemplo de solapamiento al 50% entre tramas.	71
Figura 34: Nota (Si) segmentada en tramas consecutivas de 1024 muestras (103 tramas)	72

<i>Figura 35: Esquema para la implementación del algoritmo de estimación de pitch.....</i>	<i>73</i>
<i>Figura 36: Distribución de energía de la nota (Si4) de 205 tramas</i>	<i>74</i>
<i>Figura 37: Representación de tramas de silencio (tramas 36, 37, 38 y 39) en muestras (1024 muestras)</i>	<i>75</i>
<i>Figura 38: Representación de tramas de sonoras (tramas 40, 41, 42 y 43) en muestras (1024 muestras)</i>	<i>76</i>
<i>Figura 39: Distribución de energía nota Si4 de 205 tramas con SNR (- 5db).....</i>	<i>76</i>
<i>Figura 40: Distribución de energía nota Si4 de 205 con SNR (- 5db), con tramas de 1024 muestras divididas en pequeñas tramas de 5 muestras.....</i>	<i>77</i>
<i>Figura 41: Trama 70 en muestras.....</i>	<i>78</i>
<i>Figura 42: Autocorrelación de la trama 70 en muestras</i>	<i>79</i>
<i>Figura 43: Autocorrelación de la trama 70 en segundos.....</i>	<i>80</i>
<i>Figura 44: Función de diferencias de la trama 70 en muestras.....</i>	<i>82</i>
<i>Figura 45: Función de diferencias de la trama 70 en segundos.....</i>	<i>83</i>
<i>Figura 46: Función CMNDF trama 70 en muestras.....</i>	<i>85</i>
<i>Figura 47: Función CMNDF de la trama 70 en segundos.....</i>	<i>85</i>
<i>Figura 48: Selección de tramas sonoras mediante la distribución de energía de la nota Si4. A la izquierda se representa la distribución de energía de la nota Si4. A la derecha se representa la distribución de la energía de la nota Si4 dividida en pequeñas tramas.</i>	<i>89</i>
<i>Figura 49: Selección de muestra por umbral</i>	<i>91</i>
<i>Figura 50: Tramas en las que se calculara la frecuencia fundamental.....</i>	<i>92</i>
<i>Figura 51: muestras donde se dan la nota vs tramas de la nota.....</i>	<i>93</i>
<i>Figura 52: Zoom muestras donde se dan la nota vs tramas de la nota.....</i>	<i>94</i>
<i>Figura 53: Frecuencia fundamental de todas las tramas sonoras.....</i>	<i>94</i>
<i>Figura 54: Zoom frecuencia fundamental de todas las tramas sonoras</i>	<i>95</i>
<i>Figura 55: Función Yin ajustada con la media vs ajustada con la moda matemática. La Figura (a) y (b) muestran los resultados de calcular una la frecuencia fundamental única para todas las tramas que forman la nota La6 mediante la moda matemática y la media respectivamente.</i>	<i>97</i>
<i>Figura 56: Frecuencia fundamental transformada a MIDI. La Figura (a) representa la frecuencia fundamental de la nota Si4 estimada con el algoritmo implementado en este TFG y en la Figura (b) se representa la transformación a midi de esta frecuencia.</i>	<i>99</i>
<i>Figura 57: Transformación de frecuencia a midi de varias pistas. Las Figuras (a) y (b) muestran la frecuencia fundamental de la pista formada por tres notas (Sol6, La6, La#6) estimada con el algoritmo implementado y su correspondiente transformación a midi respectivamente. Las Figuras (c) y (d) muestran la frecuencia fundamental estimada de la pista formada por cinco notas (Mi5, Fa5, Fa#5, Sol5, Sol#5) y su correspondiente transformación a midi respectivamente. Las Figuras (e) y (f) muestran la frecuencia fundamental estimada de la pista formada por ocho notas (Si4,Do5, Do#5, Re5, Mi5, Fa5, Fa#5, Sol5) y su correspondiente transformación a midi respectivamente.....</i>	<i>100</i>
<i>Figura 58: espectro de la flauta dulce estimado por nuestro algoritmo</i>	<i>101</i>
<i>Figura 59: Frecuencias estimadas vs frecuencias reales.....</i>	<i>102</i>

<i>Figura 60: Esquema que resume como añadir ruido</i>	<i>110</i>
<i>Figura 61: Señal sin ruido</i>	<i>111</i>
<i>Figura 62: Señal resultante al añadir distintos niveles de ruido. La primera fila de la Figura 62 de izquierda a derecha representa el resultado de incluir a una señal (Nota1) un nivel de ruido de (-5db y 0db) respectivamente. La segunda fila de la Figura 62 de izquierda a derecha representa el resultado de incluir a una señal (Nota1) un nivel de ruido de (5db y 10db) respectivamente.</i>	<i>111</i>
<i>Figura 63: Resultados en frecuencia de las 25 notas bajo distintas condiciones de SNR, asignando la frecuencia estimada a la media de las tramas sonoras.</i>	<i>114</i>
<i>Figura 64: Resultados del error absoluto y relativo, asignando la frecuencia estimada a la media de las tramas sonoras.....</i>	<i>115</i>
<i>Figura 65: Resultados en frecuencia de las 25 notas bajo distintas condiciones de SNR, asignando la frecuencia estimada a la más repetitiva (moda matemática) de las tramas sonoras.</i>	<i>117</i>
<i>Figura 66: Resultados dl error absoluto y relativo, asignando la frecuencia estimada a la más repetitiva (moda matemática) de las tramas sonoras.....</i>	<i>118</i>
<i>Figura 67: Resultados de la base de datos SNR igual a -5db.....</i>	<i>123</i>
<i>Figura 68: Resultados de la base de datos SNR igual a 0db</i>	<i>123</i>
<i>Figura 69: Resultados de la base de datos SNR igual a 5db</i>	<i>124</i>
<i>Figura 70: Resultados de la base de datos para todas las bases de datos</i>	<i>124</i>
<i>Figura 71: Resultados de las tramas de falsos negativos y falsos positivos al inicio de las notas para la base de datos SNR igual a -5db.....</i>	<i>125</i>
<i>Figura 72: Resultados de las tramas de falsos negativos y falsos positivos al inicio de las notas para la base de datos SNR igual a 0db.....</i>	<i>126</i>
<i>Figura 73: Resultados de las tramas de falsos negativos y falsos positivos al inicio de las notas para la base de datos SNR igual a 5db.....</i>	<i>126</i>
<i>Figura 74: Resultados de las tramas de falsos negativos y falsos positivos al inicio de las notas para todas las bases de datos.....</i>	<i>127</i>
<i>Figura 75: Resultados de las tramas de falsos negativos y falsos positivos al final de las notas para la base de datos SNR igual a -5db</i>	<i>128</i>
<i>Figura 76: Resultados de las tramas de falsos negativos y falsos positivos al final de las notas para la base de datos SNR igual a 0db.....</i>	<i>128</i>
<i>Figura 77: Resultados de las tramas de falsos negativos y falsos positivos al final de las notas para la base de datos SNR igual a 5db.....</i>	<i>129</i>
<i>Figura 78: Resultados de las tramas de falsos negativos y falsos positivos al final de las notas para todas las bases de datos.....</i>	<i>130</i>
<i>Figura 79: Resultados de las tramas de falsos negativos y falsos positivos totales al inicio y final de las notas para todas las bases de datos.</i>	<i>131</i>
<i>Figura 80: Ejemplo de funcionamiento de los ejercicios del tutor con un acierto del 100 %.....</i>	<i>134</i>

<i>Figura 81: Ejemplo de funcionamiento de los ejercicios del tutor con un acierto del 66.7%. El alumno consigue dos aciertos totales y un fallo.</i>	135
<i>Figura 82: Ejemplo del funcionamiento de los ejercicios del tutor donde se le informa al alumno que puede pasar de nivel</i>	136
<i>Figura 83: Ejemplo del funcionamiento de los ejercicios del tutor donde se le informa al alumno que no puede pasar de nivel.</i>	136
<i>Figura 84: Entorno de trabajo de la Interfaz gráfica.</i>	140
<i>Figura 85: Interfaz gráfica mostrada en tres bloques</i>	141
<i>Figura 86: Primer bloque de la interfaz gráfica</i>	142
<i>Figura 87: primera sección del primer bloque. Botón para seleccionar un determinado nivel de aprendizaje. El nivel seleccionado es el nivel principiante</i>	142
<i>Figura 88: Segunda sección del primer bloque. Botón para mostrar la partitura, escuchar la partitura mostrada, pararla, mostrar su espectrograma, y grabar la misma partitura bajo un margen de error de 2 segundos en tiempo y 40 Hz en frecuencia.</i>	143
<i>Figura 89: Segunda sección del primer bloque. Cursor que marca los tiempos en la grabación de las notas que forman la partitura</i>	143
<i>Figura 90: Tercera sección del primer bloque. Botón para mostrar resultados, y mostrar el espectrograma de la partitura grabada bajo un margen de error de 2 segundos en tiempo y 40 Hz en frecuencia.</i>	144
<i>Figura 91: Resultado de la partitura grabada por el alumno en color rojo en comparación a la que se debería haber tocado realmente en color azul.</i>	144
<i>Figura 92: tercera sección del primer bloque. Botón para mostrar resultados, y mostrar el espectrograma de la partitura grabada bajo un margen de error de 2 segundos en tiempo y 40 Hz en frecuencia.</i>	145
<i>Figura 93: Segundo bloque de la interfaz gráfica</i>	146
<i>Figura 94: Segundo bloque de la interfaz gráfica: evaluación y análisis en frecuencia y en tiempo de la partitura (Sol5, Sol6, Si5).</i>	146
<i>Figura 95: Tercer bloque de la interfaz gráfica</i>	147

8.4 ANEXO III: ÍNDICE DE TABLAS

<i>Tabla 1: espectro de frecuencias de varios instrumentos musicales. Fuente [11]</i>	16
<i>Tabla 2: Relación sonoridad y dinámica. Fuente [11]</i>	18
<i>Tabla 3: Duración de notas musicales. Fuente [18]</i>	18
<i>Tabla 4: Notas musicales o clases de pitch por región</i>	21

<i>Tabla 5: Relación frecuencia MIDI de las notas. Fuente [11].....</i>	<i>22</i>
<i>Tabla 6: Porcentaje de Error Bruto del algoritmo de Yin. [17].....</i>	<i>42</i>
<i>Tabla 7: Relación FFT, MIDI, escala y nomenclatura de cada nota de la base de datos notas_unicas</i>	<i>67</i>
<i>Tabla 8: Notas pertenecientes a la base de datos -5db. 1ª PARTE.....</i>	<i>105</i>
<i>Tabla 9: Notas pertenecientes a la base de datos 0db. 1ª PARTE</i>	<i>106</i>
<i>Tabla 10: Notas pertenecientes a la base de datos 5db. 1ª PARTE</i>	<i>107</i>
<i>Tabla 11: Notas pertenecientes a la base de datos 10db. 1ª PARTE</i>	<i>108</i>
<i>Tabla 12: Resultados obtenidos en frecuencia para distintos valores de ruido, asignando la frecuencia estimada a la media de las tramas sonoras.</i>	<i>113</i>
<i>Tabla 13: Resultados de error absoluto y relativo, asignando la frecuencia estimada a la media de las tramas sonoras.</i>	<i>114</i>
<i>Tabla 14: Resultados obtenidos en frecuencia para distintos valores de ruido, asignando la frecuencia estimada a la más repetitiva (moda matemática) de las tramas sonoras</i>	<i>116</i>
<i>Tabla 15: Resultados de error absoluto y relativo, asignando la frecuencia estimada a la más repetitiva (moda matemática) de las tramas sonoras.....</i>	<i>117</i>
<i>Tabla 16: Notas pertenecientes a la base de datos 5db. 2ª PARTE.....</i>	<i>120</i>
<i>Tabla 17: Notas pertenecientes a la base de datos 5db. 2ª PARTE</i>	<i>120</i>
<i>Tabla 18: Notas pertenecientes a la base de datos 5db. 2ª PARTE</i>	<i>121</i>
<i>Tabla 19: Número de tramas totales por pista.....</i>	<i>132</i>

9 REFERENCIAS BIBLIOGRÁFICAS

- [1] S. Mithen, The Singing Neanderthals: the Origins of Music, Language, Mind and Body. London: Weidenfeld & Nicolson, 2005.
- [2] D. J. Levitin, This Is Your Brain on Music: The Science of a Human Obsession. New York: Dutton/Penguin, 2006.
- [3] J.R. Suarez Caballero, F.J. Cañadas Quesada (academic tutor), Implementation and analysis of Pitch estimation applied to monophonic musical signals, final Project, Jaen University (Spain), 2011.
- [4] Goldstein, Blackweel handbook of sensation and perception. Oxford: Blackweel Publishing, 2005.
- [5] ESPASA, Diccionario de la lengua española. Pozuelo de Alarcón: Espasa-Calpe, 2005.
- [6] W.A. Sethares, Tuning, Timbre, Spectrum, Scale, Segunda ed. Berlin: Springer–Verlag, 2004.
- [7] S. S Soliman, Continous and discrete signals and systems, Segunda ed. Lebanon, Indiana: Prentice Hall, 1998.
- [8] A. Pinkus, S. Zafrany, Fourier Series and Integral Transforms. Cambridge: Cambridge University Press, 1997.
- [9] T. D. Rossing, The Science of Sound. New York: Springer, 1990.
- [10] A Oppenheim. Signals and Systems, 2nd ed. Upper Saddle River, New Jersey: Prentice Hall, 1997.
- [11] F. J. Cañadas Quesada, Investigation and Development of Multi-Pitch Estimation Techniques and its Application to Automatic Transcription of Polyphonic Musical Sinals, Doctoral Thesis, Jaen University (Spain), 2009.
- [12] W. DeLiang, G. Brown, Computational Auditory Scene Analysis. Sheffield: University of Sheffield &, Ohio State University, 2006.

- [13] P. Lindsay, D. Norman, Human information processing: An introduction to psychology. New York: Academic Press, 1977.
- [14] Perfecto Herrera-Boyer, Anssi Klapuri, and Manuel Davy, Signal Processing Methods for Music Transcription. Boston: Springer US, MA, 2006.
- [15] W. M. Hartmann, "Pitch, periodicity, and auditory organization," Journal of the Acoustical Society of America, vol. 6, no. 100, pp. 3491–3502, 1996.
- [16] ANSI, "American National Estandard Acoustic Terminology," 1994.
- [17] A. de Cheveigné and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," Journal of the Acoustical Society of America, vol. 4, no. 111, pp. 1917-1930, 2002.
- [18] Bh Leo. (2012). Duration of musical notes. [online].
<http://www.notasmusicales.org/duracion-notas-musicales/>
- [19] E. Wold, T. Blum, D. Keislar, and J Wheaton, " Contentbased Classification, Search and Retrieval of Audio". IEEE Multimedia, 3(3):27–36, 1996.
- [20] A. Bregman, Auditory Scene Analysis: The Perceptual Organization of Sound. Montreal: MIT Press, 1990.
- [21] S. Martin, American national standard psychoacoustical terminology. New York: American National Standards Institute, 1973.
- [22] Juan Pablo Lobaton. (2015). Investigation: Fletcher- Munson diagram, Isofonic curve. [online].
<http://jplobaton.blogspot.com.es/2015/07/investigacion-diagrama-de-fletcher.html>
- [23] María Bermudez. (2007). Music sheet. [online].
http://www.doslourdes.net/partitura_cigue%C3%B1a.htm
- [24] Encyclopedia Britannica. (2011) Encyclopedia Britannica Online. [Online].
<http://www.britannica.com/EBchecked/topic/398918/music>

- [25] L. Brown, The New Shorter Oxford English Dictionary. New York: Oxford University Press, 1993.
- [26] J.J. Nattiez, Music and Discourse: Toward a Semiology of Music. Princeton: Princeton University Press, 1990.
- [27] ISO, "Acoustics - Standard tuning frequency (Standard musical pitch)". Geneva, 1975.
- [28] J.M. Pisano. (2006) Mustech.Net. [online].
<http://mustech.net/2006/09/midstandards-a-brief-history-and-explanation/>
- [29] R. Paiva, Melody Detection in Polyphonic Audio. PhD thesis, Computer Science Department, Science and Technology, University of Coimbra (Portugal), 2006.
- [30] A. Klapuri, Signal Processing Methods for the Automatic Transcription of Music. PhD thesis, Tampere University of Technology, 2004
- [31] Li Yipeng, Monaural Musical Sound Separation. PhD thesis, The Ohio State University, 2008.
- [32] M. D. Plumbley, S. A. Abdallah, J. P. Bello, M. E Davies, G. Monti, and M. B. Sandler, Automatic music transcription and audio source separation. *Cybernetics and Systems*, 33(6):603–627, 2002.
- [33] Bruce Fries, Marty Fries. (2005). Practical Digital Audio: Fundamental keys for the Integral Tratement. [online].
<http://serbal.pntic.mec.es/srug0007/archivos/radiocomunicaciones/3%20SE%D1ALES%20DIGITALES/Muestreo%20digital.pdf>
- [34] Barco Rodrigo Ricardo. (2013). Discretization of signals. [online].
<http://www.ingelec.uns.edu.ar/pds2803/materiales/cap05/05-cap05.pdf>
- [35] E. Barrull. (1992). Analysis of Articulatory Verbal Behavior in Spontaneous Group Conversations. [online].
http://www.biopsychology.org/tesis_esteve/procana/procana.htm

[36] J. M. Pisano. (2006) Mustech.Net [online] <http://mustech.net/2006/09/midistandards-a-brief-history-and-explanation/>

[37] A. P. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," IEEE Trans. Speech and Audio Proc., 11(6), 804–816, 2003.

D. Moelants, C. Rampazzo, A Computer System for the Automatic Detection of Perceptual Onset, 1997.

[38] S. Dixon, F. Gouyon, G. Widmer, "Towards Characterisation of Music Via Rhythmic Patterns", Proceedings of the International Symposium on Music Information Retrieval, 2004.

[39] J. P. Bello, "Complex Domain Onset detection for musical signals". In International Conference on Digital Audio Effects, 2003.

[40] Elisa. (2012). Position of the flute, fingering, study tips. [online]. https://es.wikipedia.org/wiki/Flauta_dulce

[41] Lawrence Fritts. (1997). University of Iowa Electronic Music Studios Web Site. [Online]. <http://theremin.music.uiowa.edu/>

[42] Online bibliography encyclopedia. (2004). Helmholtz bibliography. [online]. <https://www.biografiasyvidas.com/biografia/h/helmholtz.htm>

[43] T.D. Rossing, Springer Handbook of Acoustics. New York: Springer, 2007.

[44] Online bibliography encyclopedia. (2004). Thomas Edison bibliography. [online]. <http://www.biografiasyvidas.com/monografia/edison/fotos5.htm>

[45] Jody Rosen. (2008). The phonograph of Edison. [online]. <https://es.wikipedia.org/wiki/Fon%C3%B3grafo>

[46] Acoustical Society of America. (2011) Journal of the Acoustical Society of America. [Online]. <http://asadl.org/jasa/>

- [47] J. Obata and R. Kobayashi, "A Direct-Reading Pitch Recorder and Its Applications to Music and Speech," *Journal of the Acoustical Society of America*, vol. 9, no. 2, pp. 156-161, 1937.
- [48] Dudley, "Remarking Speech," *Journal of the Acoustical Society of America*, vol. 11, no. 2, pp. 169-177, 1939.
- [49] J.C.R. Licklider, "A duplex theory of pitch perception," *Experientia*, vol. 7, no. 4, pp. 128-134, 1951.
- [50] A.M. Noll, "Short-Time Spectrum And 'Cepstrum' Techniques For Vocal Pitch Detection," *Journal of the Acoustical Society of America*, vol. 36, no. 2, pp. 296–302, 1964.
- [51] M.J. Ross and H.L. Shaffer, "Average Magnitude Difference Extractor," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 22, no. 5, pp. 353–362, 1974
- [52] J.L. Goldstein, "An optimum processor theory for the central formation of pitch complex tones," *Journal of the Acoustical Society of America*, vol. 54, no. 6, pp. 1496-1516, 1973
- [53] F.L. Wightman, "The pattern-transformation model of pitch," *Journal of the Acoustical Society of America*, vol. 54, no. 2, pp. 407-416, 1973.
- [54] E. Terhardt, "Pitch, consonance and harmony," *Journal of the Acoustical Society of America*, vol. 55, no. 5, pp. 1061-1069, 1974.
- [55] W. Hess, *Pitch Determination of Speech Signals*. Berlin: Springer-Verlag, 1983.
- [56] Ray Meddis and Lowell O'Mard, "A Unitary Model of Pitch Perception," *Journal of the Acoustical Society of America*, vol. 102, no. 3, pp. 1811-1820, 1997.
- [57] A. J. M. Houtsma, "Pitch Perception," in *Hearing—Handbook of Perception and Cognition*. San Diego: San Diego, CA: Academic, 1995, ch. 8.
- [58] J.R. Suarez Caballero, F.J. Cañadas Quesada (academic tutor), *Implementation and analysis of Pitch estimation applied to monophonic musical signals*, final Project, Jaen University (Spain), 2011.

- [59] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 708–716, Nov. 2000.
- [60] B. Kedem, "Spectral analysis and discrimination by zero-crossings," *Proceedings of the IEEE*, vol. 74, no. 11, pp. 1477-1493, 1986.
- [61] E. Yumoto, W. J. Gould and T. Baer, "Harmonic-to-noise ratio as an index of the degree of hoarseness", *J. Acoust. Soc. Am.* 71, 1544-1549 .1982.
- [62] Serquera Peyro, Jaime and corral Gonzales, Juan Luis. (2003). Algorithm for tune real time detection in monophonic music signals by separation-harmonic accumulation (SAA). [online].
http://www.sea-acustica.es/fileadmin/publicaciones/publicaciones_4375cd002.pdf
- [63] Coca S. Andrés Eduardo, Estimation of the pitch in monophonic signals of sung voice, Doctoral thesis, National University of Colombia, Manizares Heardquarters, 2004.
- [64] Deolazabal, Tirso. *Acústica Musical y Organología*. Editorial Ricordi Americana S.A.E.C. Buenos Aires, Argentina. 1993.
- [65] Andrés Eduardo Coca. (2017). Pitch estimation on monophonic voice signals Sung. [online].
http://www.academia.edu/760347/Estimaci%C3%B3n_del_pitch_en_senales_monof%C3%B3nicas_de_voz_cantada
- [66] J. H. Fowler. (2006). Harmonic series. [online].
<http://www.burmusik.com/3-armoniacutea.html>
- [67] D. J. Hermes, "Measurement of Pitch by Subharmonic Summation," *Journal of the Acoustical Society of America*, vol. 83, no. 1, pp. 257-264, 1988
- [68] A. V. Oppenheim, *Discrete-Time Signal Processing*, Segunda ed. Upper Saddle River, NJ: Prentice Hall, 1999.
- [69] N. Hiraoka, Y. Kitaoe, H. Ueta, S. Tanaka and M Tanabe "Harmonic-intensity analysis of normal and hoarse voices", *J. Acoust. Soc. Am* 76, 1648- 1651. 1984.

[70] J. Romero, S. Cerdá. (2008). Using the analysis of Multiresolution for calculate the pitch of signals in Presence of noise. [online].

https://www.researchgate.net/publication/242621429_Uso_del_analisis_de_multirresolucion_para_calcular_el_pitch_de_senales_en

[71] I. Daubechies, Ten Lectures on wavelets. Philadelphia: SIAM, 1992.

[72] C.K. Chui, "An Introduction to wavelets", Academic, San Diego, 1992.

[73] Cerdá, Salvador, Romero, J: "Use of multiresolution analysis to calculate pitch in presence of noise". J. Acoust. Soc. Am. Volume 99, Issue 4, pp. 2497-2500, 1996.

[74] De la Cuadra Patricio, Aaron Master and Craig Sapp. (2001). Efficient Pitch Detection Techniques For Interactive Music. [online].

https://ccrma.stanford.edu/~pdelac/research/MyPublishedPapers/icmc_2001-pitch_best.pdf

[74] Torres Ortega J. M. (2014). Afinador Polifonico para Hipone. [online]

<http://tauja.ujaen.es/bitstream/10953.1/1856/1/TFG-Torres-Ortega-Jose-M.pdf>

[75] H.Duifhuis, L.F Wilems and R.J Sluyter, " Measuremenst of pitch in speech: An Implementation of Goldstein's theory of pitch perception", J. Acoust. Soc. Am.71, 1568-1579, 1982.

[76] Ray Meddis and Lowel O'Mard, "A Unitary Model of Pitch Perception," Journal of the Acoustical Society of America, vol. 102, no. 3, pp. 1811-1820, 1997.

[77] E. Terhardt, G Stoll and M. Seewann " Algorithm for extraction of pitch salience from complex tonal signals", J. Acoust. Soc. Am. 71, 679-688. 1982.

[78] M. Goto, H Hashiguchi, T Nishimura, and R Oka. RWC musicdatabase: Popular, classical, and jazz music database. in Proc. Int. Symp. Music Inf. Retrieval, 2002.

[79] M. Goto. RWC Music Database. National Institute of Advanced Industrial Science and Technology (AIST), 2005.

- [80] The University of Iowa. Iowa Music Database. University of Iowa, 2006.
- [81] F Opolko and J Wapnick. McGill University Master Samples. McGill University, 1992.
- [82] Z. Duan and B. Pardo. Soundprism: An online system for score-informed source separation of music audio. *Selected Topics in Signal Processing, IEEE Journal of*, 5(6):1205– 1215, 2011.
- [83] RCW Music Database. (2003). Musical Instrument Sound. [online].
<https://staff.aist.go.jp/m.goto/RWC-MDB/rwc-mdb-i.html>
- [84] RCW Music Database. (2003). Classical Music. [online].
<https://staff.aist.go.jp/m.goto/RWC-MDB/rwc-mdb-c.html>
- [85] F. J. Rodríguez Serrano, Separation of sound sources de fuentes sonoras, Doctoral Thesis, Jaen University, 2014.
- [86] Audacity. (2016). Descarga audacity. [online].
<https://audacity.softonic.com/>
- [87] Audacity. (2016).Funcionamiento Audacity. [online].
http://www.jesusda.com/docs/ebooks/ebook_tutorial-edicion-de-sonido-con-audacity.pdf
- [88] Van Loan, C., Computational Fremeworks for the Fas Fourier Transform. Philadelphia, PA: SIAM, 1992.
- [89] D. M. Muñoz, P. J. Reche Lopez, N. R. Reyes, P. Vera Candeas, Digital transmission. Jaen: Jaen University, 2009.
- [90] Albiol Colomer, Antonio, Tratamiento digital de la señal: teoría y aplicaciones. Valencia: Universidad Politecnica de Valencia, 2007.
- [91] Chacon-Rodriguez, A., Julian, P., Castro, L., Alvarado, P. and Hernandez, N., 'Evaluation of Gunshot Detection Algorithms', *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2011, vol. 58, pp. 363 – 373.