

Introducción al análisis de una señal musical

Duque Karla A.
kduque@fi.uba.ar
108406

Rodriguez Guido E.
guerodriguez@fi.uba.ar
108723

Vargas Joaquín
jvargas@fi.uba.ar
104323

Dietrich Pablo
pdietrich@fi.uba.ar
83383

Abstract—Se estudia su estructura interna de un archivo WAV, se representa la información en el dominio temporal para luego analizarla estimando la frecuencia fundamental de distintas regiones. Además, se aplican filtros mediante convolución con diferentes respuestas al impulso para caracterizar su comportamiento como filtros pasa-altos y pasa-bajos.

I. INTRODUCCIÓN

En este trabajo se emplean archivos de audio en formato WAV de los cuales se extraerán las señales, se graficarán y analizará algunas componentes periódicas de las mismas.

A. Formato del archivo de audio WAV

WAV de las siglas en inglés *Waveform Audio File Format* es un formato de almacenamiento de archivos de audio digital de tipo *lossless*, es decir, sin pérdida de información. Esto implica que una vez es digitalizada la señal, los datos se conservan fielmente con la información original o en su defecto fueron comprimidos mediante *lossless compression* en la cual se emplean algoritmos sin pérdida como lo son los algoritmos Huffman [1], LZW [2], entre otros.

La ventaja de utilizar este formato es que nos permite centrarnos en el análisis de la señal sin preocuparnos por distorsiones, *artifacts* entre otros factores producto de una *lossy compression*.

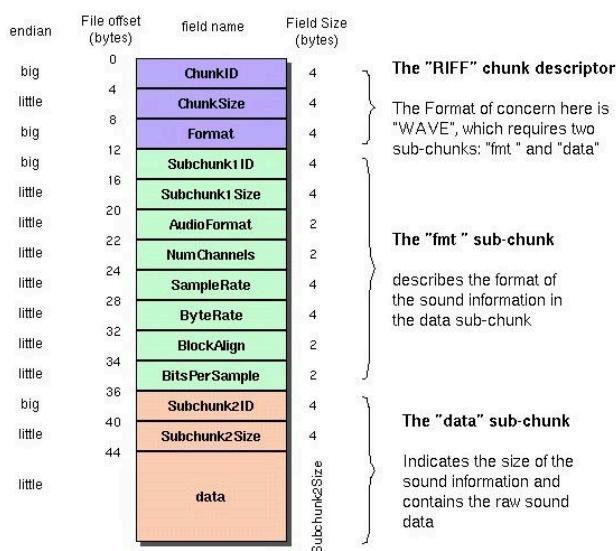


Figura 1. Estructura de datos de un archivo WAV

En base a la especificación [3] del formato de presentado en la Figura 1 podemos identificar 3 bloques o *chunks* principales, RIFF, FMT y DATA, de los cuales 2 son *metadata*.

a) **RIFF (Resource Interchange File Format)**: Este chunk es el *header* del archivo y define que el contenido corresponde al formato WAV (más allá de su extensión), el tamaño del archivo y un *string* "WAVE" que indica el formato. Todos sus campos son de 4 bytes.

b) **FMT (Format)**: En este *sub-chunk* se detalla información relevante respecto a la señal de audio como lo son el formato de audio (habitualmente PCM sin compresión

del cual haremos mención posteriormente), la cantidad de canales utilizados en la grabación, la frecuencia de muestreo de la señal, el *byterate*, la cantidad de bytes que representan una muestra de la señal (*block align*) y la cantidad de bits (declarados) utilizados durante el proceso de muestreo.

c) **DATA (Información de audio)**: En este *sub-chunk* se hallan 2 parámetros relevantes que son la cantidad de bytes en la señal de audio y la señal de audio muestreada.

B. Modulación PCM

La modulación utilizada habitualmente en archivos sin compresión es PCM [4] que son las siglas de Pulse Code Modulation, esta es la forma más básica de digitalizar una señal haciendo uso de un ADC (Analog to digital converter). El procedimiento se basa en tres etapas:

- **Muestreo**: donde la señal se discretiza en intervalos regulares de tiempo definidos por la frecuencia de muestreo f_s del ADC.
- **Cuantificación**: cada muestra se aproxima a un nivel dentro de un conjunto finito de valores dado por la resolución en bits (ej: 16 Bits equivalen a 65536 niveles).
- **Codificación**: se asigna a cada valor cuantificado su correspondiente valor codificado en binario.

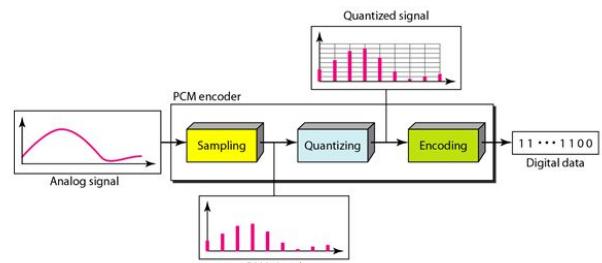


Figura 2. Diagrama esquemático del proceso de conversión de una señal de tiempo continuo a discreto

II. DESARROLLO

A continuación se presenta el desarrollo involucrado para la obtención de la información de los archivos WAV, la identificación de segmentos quasi-periódicos de la canción 1, segmentos no periódicos de la canción 2 y la estimación de las frecuencias a las que fueron muestreadas.

A. Obtención de la información de un archivo WAV

Con el objetivo de lograr una implementación transparente basada en la estructura interna del formato WAV, se desarrolló un script en python que procesa el archivo de audio y obtiene la información relevante del mismo.

Inicialmente la estrategia consistió en definir un diccionario que distingue entre la *metadata* y los datos de la

señal en el cual se describieron de forma explícita los campos que conforman los bloques **RIFF**, **FMT** y **DATA**, detallando para cada uno su tamaño y *endianess*.

Posteriormente, se efectuó la lectura completa del archivo en modo binario y la asignación de cada campo, considerando su *endianess* previamente definido y aplicando los *casts* necesarios para la correcta interpretación de la información, tal es el caso del identificador de formato, el número de canales, la frecuencia de muestreo, el *byte rate*, el tamaño de bloque de muestra y la cantidad de bits con la que fue muestreada la señal.

Luego de ser recuperada la información bruta de la señal de audio se procedió a reconstruirla a partir del bloque **DATA** en función de la metadata obtenida anteriormente para dicho motivo se considerará que cada muestra debe ser interpretada como un número entero en formato *little endian* con signo de un tamaño en bytes igual al valor que nos indica el campo *block_align*.

Para representar la señal en el dominio temporal se generaron pares de la forma tiempo/valor, donde el tiempo asociado a la muestra n se calculó según (1) de la cual f_s es la frecuencia de muestreo. Para la señal de la canción 1 se identificó una $f_s = 44,1$ kHz y para la canción 2 una $f_s = 48$ kHz.

$$t_n = \frac{n}{2 \cdot f_s} \quad (1)$$

Finalmente, las muestras se graficaron tal como se presenta en la Figura 3 y Figura 4 mediante la biblioteca `matplotlib`, obteniendo la representación temporal de la señal de audio y por medio de inspección visual se destacaron tres regiones con cierta periodicidad local en su forma de onda.

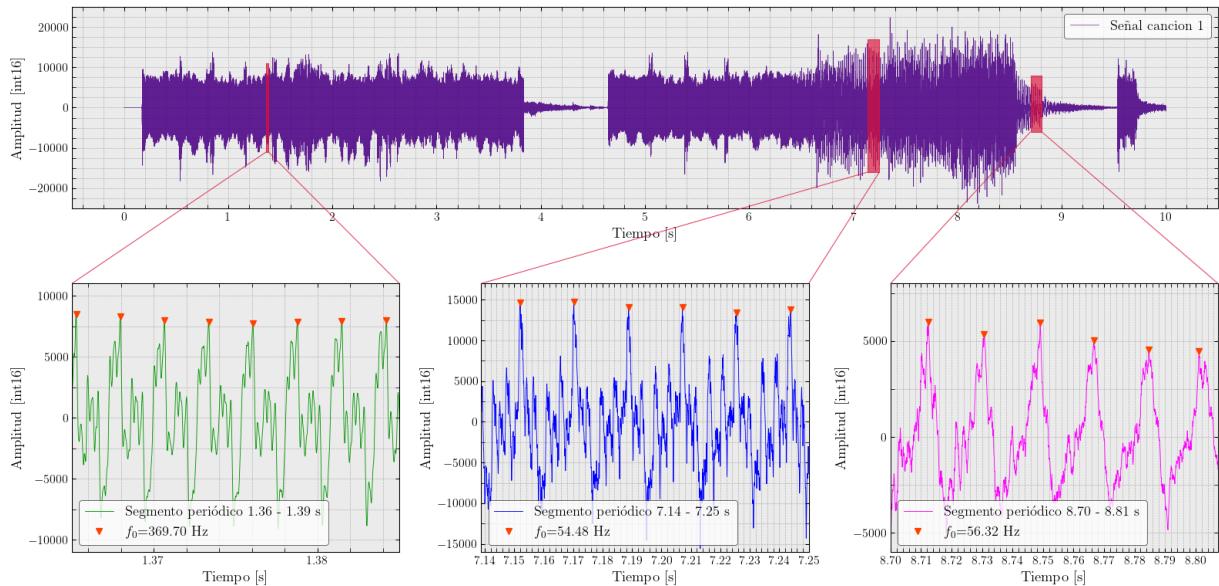


Figura 3. Representación de la señal de audio extraída de la canción 1 y la identificación de 2 regiones quasi-periódicas

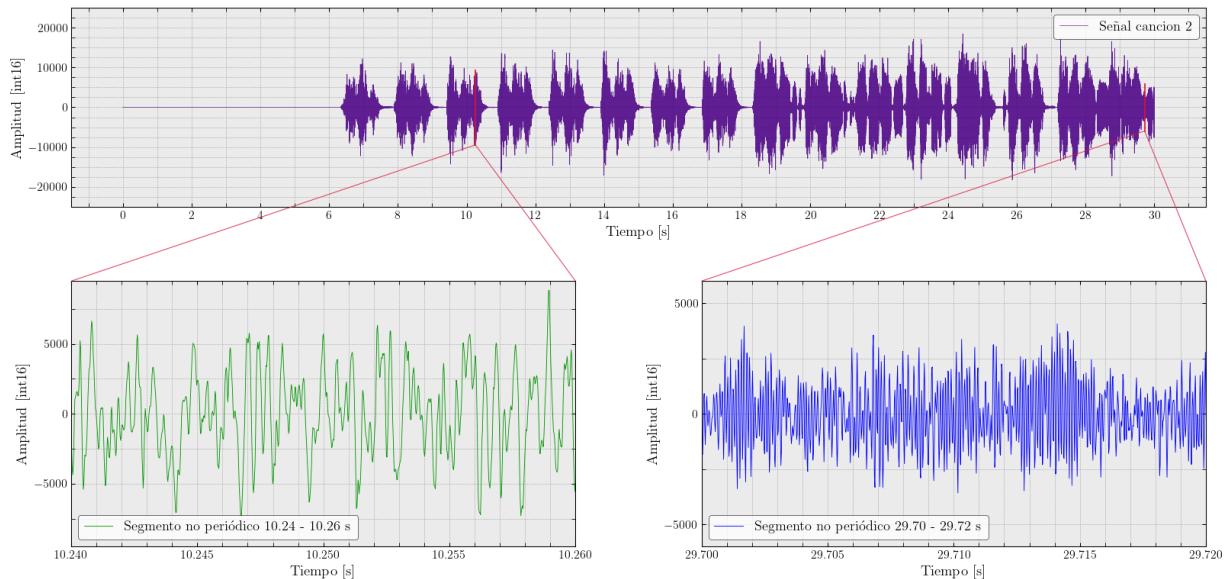


Figura 4. Representación de la señal de audio extraída de la canción 2 y la identificación de 2 regiones no periódicas

B. Estimación de la frecuencia fundamental del segmento periódico

Haciendo uso de la grilla generada en el gráfico presentado en la Figura 3 podemos estimar el período para cada segmento mediante la ecuación (2), en ella podemos calcular la frecuencia como el inverso de la diferencia temporal de dos muestras o como la relación entre la frecuencia de muestreo (f_s) y la diferencia del número de muestra asociada a dos picos consecutivos.

$$f_0 = \frac{1}{t_1 - t_0} = \frac{f_s}{n_1 - n_0} \quad (2)$$

Alternativamente, se desarrolló un algoritmo de detección de picos que consiste en identificar máximos locales de la señal que superen entre ellos una diferencia de N muestras, la cual fue determinada de forma empírica de tal forma que esta sea lo suficientemente grande para no tomar picos cercanos al de interés (ruido de alta frecuencia) y lo suficientemente angosta para no perder los picos relevantes de la señal. Una vez obtenidos los picos que cumplen nuestra restricción, se estima el período fundamental como el promedio de las diferencias consecutivas de tiempo entre ellos. De esta forma, la frecuencia fundamental se calcula como el inverso de dicha

No obstante, esta periodicidad aparente corresponde al compás¹ de la canción, por lo que para identificar segmentos no periódicos fue necesario analizar intervalos más acotados dentro de beat

C. Síntesis de señales (La3)

Luego de sintetizar 3 formas de ondas (senoidal, triangular y cuadrada), todas a 440 Hz, se procedió a graficarlas como se muestra en la Figura 5. En ellas se logran observar diferencias notorias en las formas de onda a pesar de ser todas iguales en su frecuencia fundamental, dichas diferencias son producto, tal como se detalló anteriormente, de sus componentes armónicas. Esto, se evidenciaría aún más si observáramos el espectro en frecuencia de las señales.

D. Convolución con filtro pasa-bajos y pasa-altos

En la Figura 6 y Figura 7 como también en la Figura 8 y Figura 9 se presentan la canción 1 y 2 respectivamente, filtradas por dos sistemas distintos como resultado del proceso de convolución entre las señales y las respuestas al impulso de cada sistema. Para determinar el tipo de filtro de cada sistema se analizará su efecto individual sobre los segmentos periódicos presentados anteriormente (Figura 3), es decir, en los segmentos de “baja” (≈ 55 Hz) y “alta” frecuencia (≈ 370 Hz).

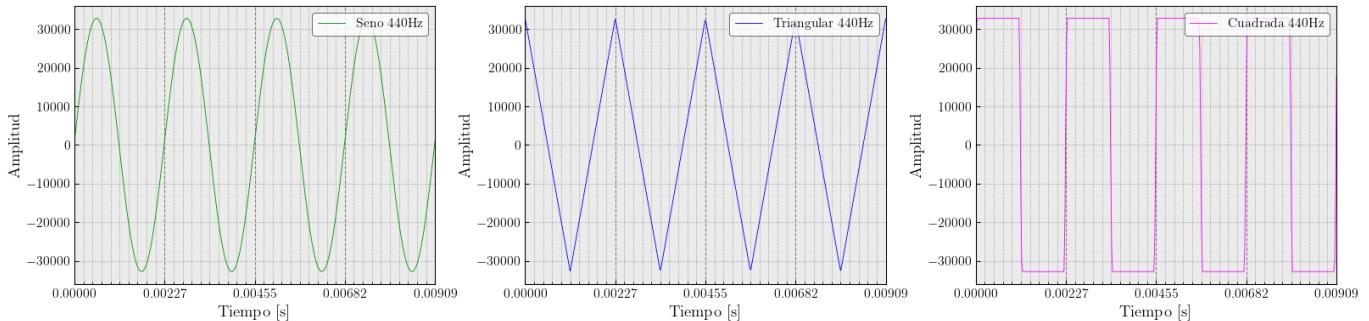


Figura 5. Diferentes señales a 440 Hz

diferencia promedio, lo que equivale a la expresión de la ecuación (2) dando como resultado $f_0 = 369,70$ Hz para el primer segmento, $f_0 = 54,48$ Hz para el segundo y $f_0 = 56,32$ Hz para el tercero.

Observar que dichas frecuencias responden en gran similitud las expresadas en tablas musicales [5] donde se tiene un Fa sostenido en 369,99 Hz en cuarta octava y un La en 55 Hz en primera octava. Se hace mención que a pesar de corresponder a la misma nota el segundo y tercer segmento, su forma de onda es distinta, esto es debido a las diferencias en sus armónicos. Los valores obtenidos muestran que el sonido del primer segmento se percibe más agudo en comparación con el segundo y tercero, que resultan más graves, esto nos muestra que las frecuencias más altas se corresponden a tonos agudos, mientras que las frecuencias más bajas se asocian a tonos graves.

En la canción 2 de la Figura 4 puede apreciarse una periodicidad global asociada al *tempo*² de 81 BPM [6], marcado principalmente por aplausos y golpes.

Para el sistema 1, se observa en la Figura 6 que las señales de baja frecuencia son atenuadas significativamente en el segundo y tercer segmento periódico (azul y rosa) indicando que se trata de un filtro pasa-altos. Por otro lado, en la convolución con el sistema 2 no se perciben cambios notables en la señal por lo que este sistema no afecta notoriamente a la canción 1 por lo que se intentará determinar su comportamiento mediante la canción 2.

En la canción 2, ambos segmentos tienen la misma duración (0,02 segundos), lo que permite una comparación directa. En el segundo segmento, se observa una mayor variabilidad en la señal, lo que sugiere la presencia de componentes de alta frecuencia a diferencia del primer segmento..

Al aplicar la convolución con el sistema 2, el segundo segmento se ve considerablemente atenuado, mientras que el primer segmento, con menos variabilidad, no muestra cambios indicando que se trata de un filtro pasa-bajos

¹El compás es la división del tiempo musical en unidades regulares, que agrupa un número fijo de pulsos o beats

²El tempo de una canción hace referencia a la velocidad con la que es ejecutada la pieza, en este caso 81 BPM (Beats Per Minute) equivalen a una frecuencia de 1,35 Hz

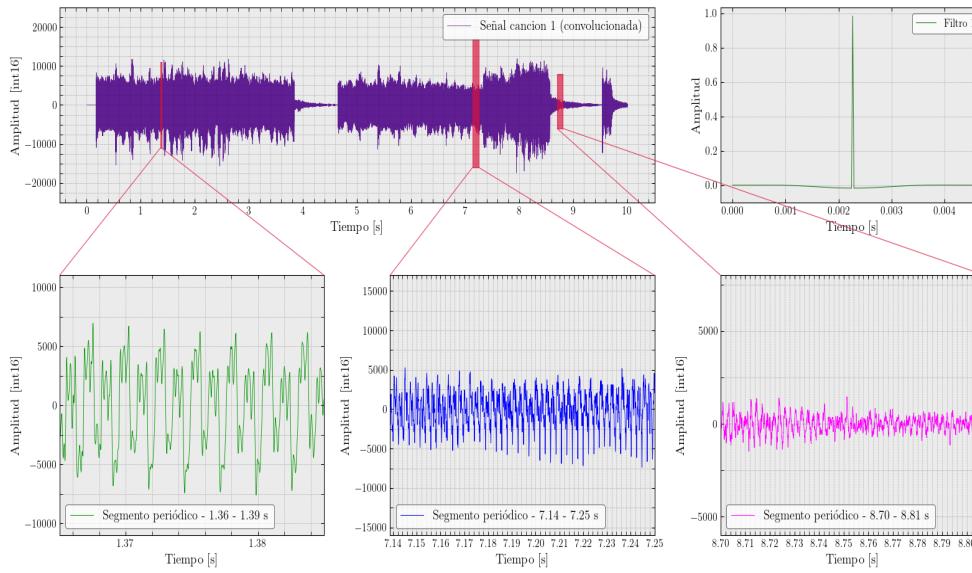


Figura 6. Convolución de canción 1 con filtro 1

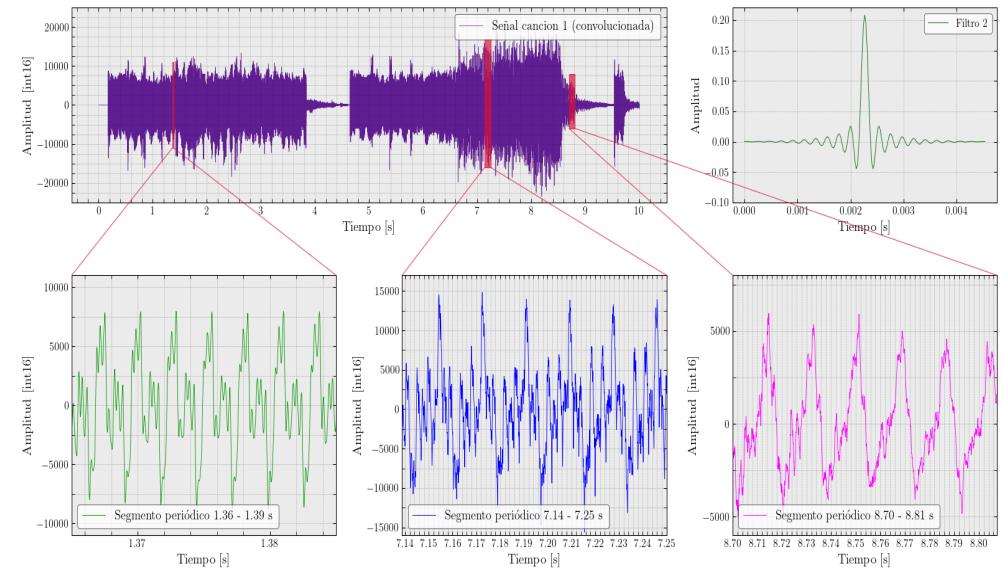


Figura 7. Convolución de canción 1 con filtro 2

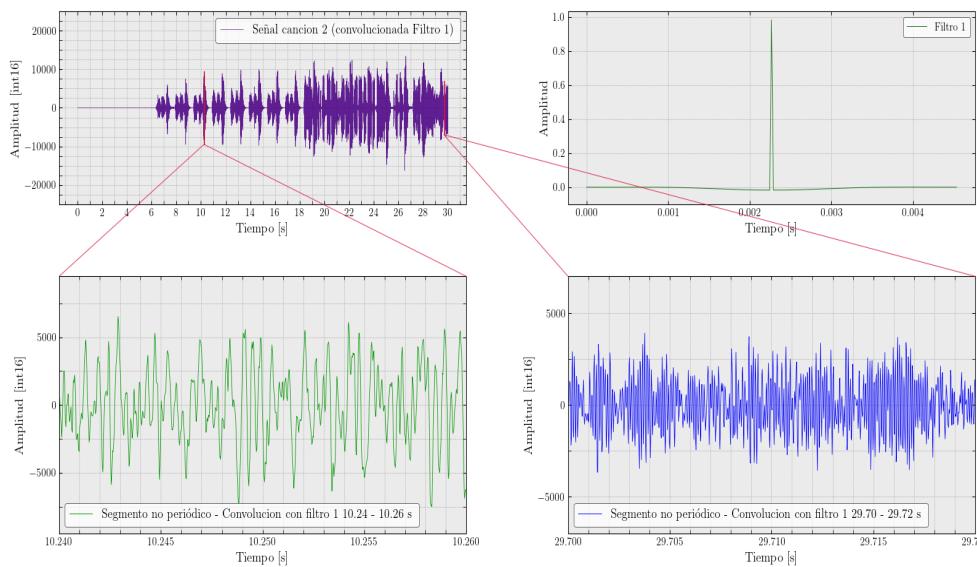


Figura 8. Convolución de canción 2 con filtro 1

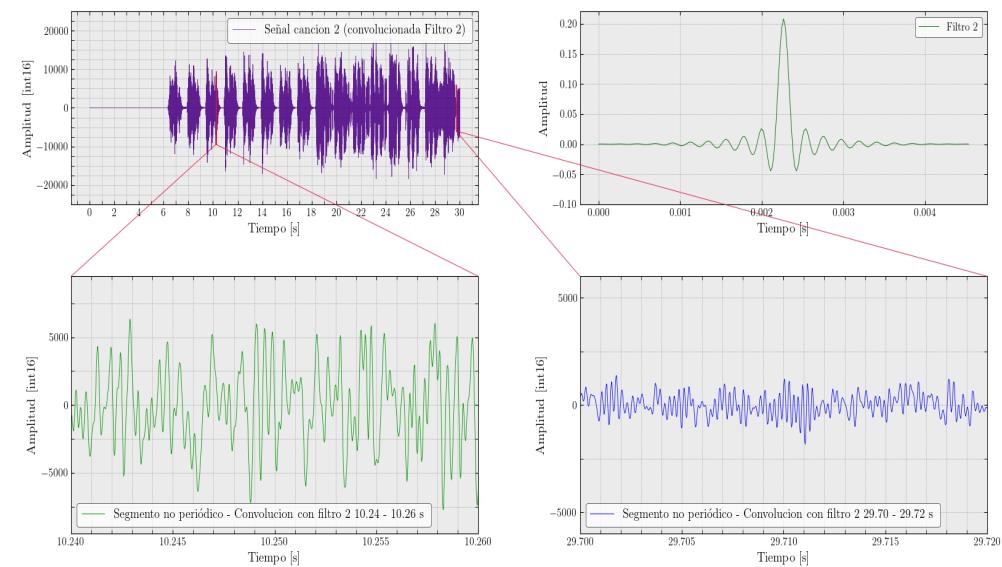


Figura 9. Convolución de canción 2 con filtro 2

III. CONCLUSIONES

- Pudimos observar que a pesar de que dos señales tengan la misma frecuencia fundamental su forma de onda es distinta, esto se debe a la diferencia en su componente armónica
- Percibimos que los tonos agudos corresponden a las frecuencias altas, mientras que las frecuencias más bajas se asocian a tonos graves
- Observando la señal convolucionada de ambas canciones con la “delta de Dirac”, concluimos que ésta última se comporta como un filtro pasa altos. En contraparte, la convolución con la señal seno cardinal nos muestra que ésta se corresponde con un filtro pasa bajos
- La comparación entre las señales sintetizadas (senoidal, triangular y cuadrada) nos muestra que la percepción del sonido no se basa solo en la frecuencia fundamental sinó que también depende de los armónicos

IV. ANEXO

- Link al repositorio donde se encuentra el código desarrollado: [Repositorio \(click acá\)](#)

REFERENCIAS

- [1] Wikipedia contributors, “Huffman coding – Wikipedia, The Free Encyclopedia.” [Online]. Available: https://en.wikipedia.org/w/index.php?title=Huffman_coding&oldid=1308318399
- [2] Wikipedia contributors, “Lempel–Ziv–Welch – Wikipedia, The Free Encyclopedia.” [Online]. Available: <https://en.wikipedia.org/w/index.php?title=Lempel%E2%80%93Ziv%E2%80%93Welch&oldid=1307959679>
- [3] “Microsoft WAVE soundfile format.” [Online]. Available: <http://soundfile.sapp.org/doc/WaveFormat/>
- [4] Wikipedia contributors, “Pulse-code modulation – Wikipedia, The Free Encyclopedia.” [Online]. Available: https://en.wikipedia.org/w/index.php?title=Pulse-code_modulation&oldid=1309515144
- [5] “Notas musicales y su correspondiente frecuencia.” [Online]. Available: <https://mixbutton.com/music-tools/frequency-and-pitch/music-note-to-frequency-chart>
- [6] “Analisis of the tempo of We Will Rock You by Queen.” [Online]. Available: <https://songbpm.com/@queen/we-will-rock-you-fce1d43f-b953-4dfb-af21-38e15c507177>