Universidad de la República

Facultad de Ingeniería

Procesamiento digital de señales de audio

Practico 1

Autor:
Federico Bello

13 de septiembre de 2024

UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

Índice

1. Ejercicio 1

Este ejercicio tiene como objetivo estudiar el muestreo y la cuantización de señales de audio.

1.1. Parte 1 - Sampling/Aliasing

Se comienza el problema con un audio, cuyo espectrograma se observa en la figura 1. Se observa como el mismo consiste de dos tonos, uno en $6000Hz$ seguido de uno en $2000Hz$. Por el teorema de muestro, se sabe que es necesario tener muestras cada al menos el doble de la frecuencia máxima de la señal para poder recuperar la misma sin pérdida de información. Es decir, si f_s es la frecuencia de muestreo y f_B el ancho de banda de la señal de interés, se debe cumplir que $f_s \geq f_B$ para poder recuperar la señal. El punto donde esta condición se cumple con igualdad se conoce como la frecuencia de Nyquist. En caso de no cumplirse no es posible recuperar la señal, obteniendo el efecto de *aliasing* si se intentase. Por lo tanto, para poder recuperar la señal será necesario muestrear a al menos $12kHz$, aunque muestreando a al menos $4kHz$ se podría recuperar la segunda parte de la señal (correspondiente a $2kHz$).

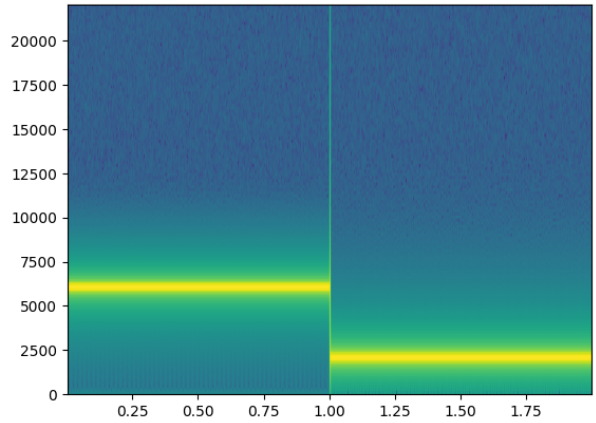


Figura 1: Espectrograma del audio *waves*

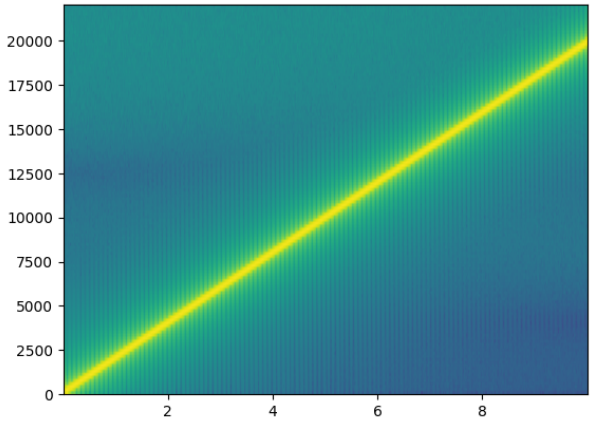


Figura 2: Espectrograma del audio *chirp*

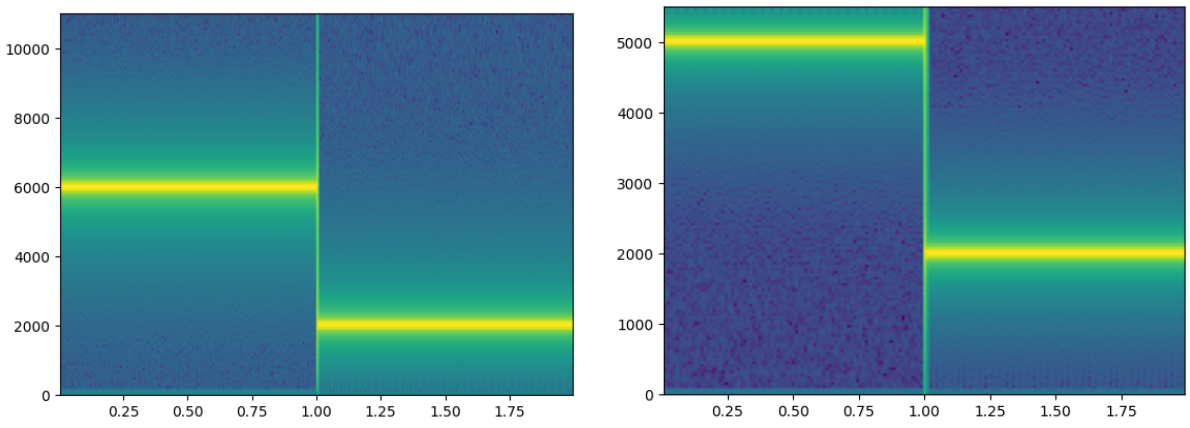
Además, del caso anterior, se tiene un segundo audio, cuyo espectrograma puede visualizarse en la figura 2. En este caso se tiene un tono el cual aumenta su frecuencia hasta llegar a un valor de aproximadamente $20kHz$. Por lo tanto, para no perder información, es necesario muestrear a una tasa de al menos $40kHz$.

En la figura 3 se observan ambos audios muestreados a una frecuencia de $22kHz$ y $11kHz$. Se ve como al muestrear el primer audio a una frecuencia alta la señal no se distorsiona (figura 3a), mientras que al disminuir la frecuencia de muestreo la primera mitad de la señal es distorsionada (figura 3b). En las figuras 3c y 3d se observa un patrón aun mas interesante, donde el tono es reconstruido correctamente hasta que alcanza la frecuencia de Nyquist, a partir de ese punto comienza a verse *aliasing*.

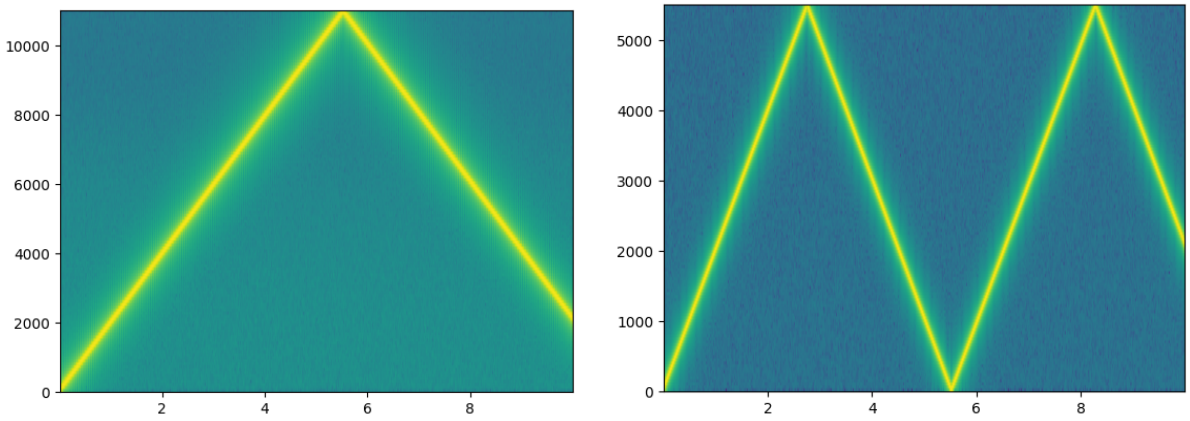
Una alternativa para prevenir el aliasing seria agregar un filtro anti aliasing previo a submuestrear. Esto es, un filtro pasabajos de frecuencia de corte la frecuencia de Nyquist. De esta forma, si bien se pierde la información de frecuencias altas, la informacion obtenida es correcta. Esto es lo que se puede observar en las figuras 4, donde las partes de la señal a mayor frecuencia que la frecuencia de Nyquist se encuentran atenuadas gracias al filtro.

1.2. Parte 2 - Quantization/Dithering

Dada una señal de entrada $\mathbf{x}$ se llamara la salida del paso de cuantización de Q niveles $\mathbf{x_Q}$. Se define $Q = \frac{2A}{2^n}$, donde A es la amplitud de la señal $\mathbf{x}$ y n es la cantidad de bits de cuantización. Si el el error de



(a) *Waves* $f_s = 22kHz$ (b) *Waves* $f_s = 11kHz$



(c) *Chirp* $f_s = 22kHz$ (d) *Chirp* $f_s = 11kHz$

Figura 3: Submuestro de las señales *wave* y *chirp* a distintas frecuencias descartando una de cada dos o cuatro muestras

cuantización es uniforme con distribución $\mathcal{U}\left(-\frac{Q}{2}, \frac{Q}{2}\right)$, la relación señal a ruido SNR para un sistema de n bits está dada por:

$$SNR = \frac{P_x}{E}$$

donde P_x es el valor de la potencia de la señal de entrada y E es la potencia del error de cuantización. Es posible calcular la potencia del ruido y de la señal como:

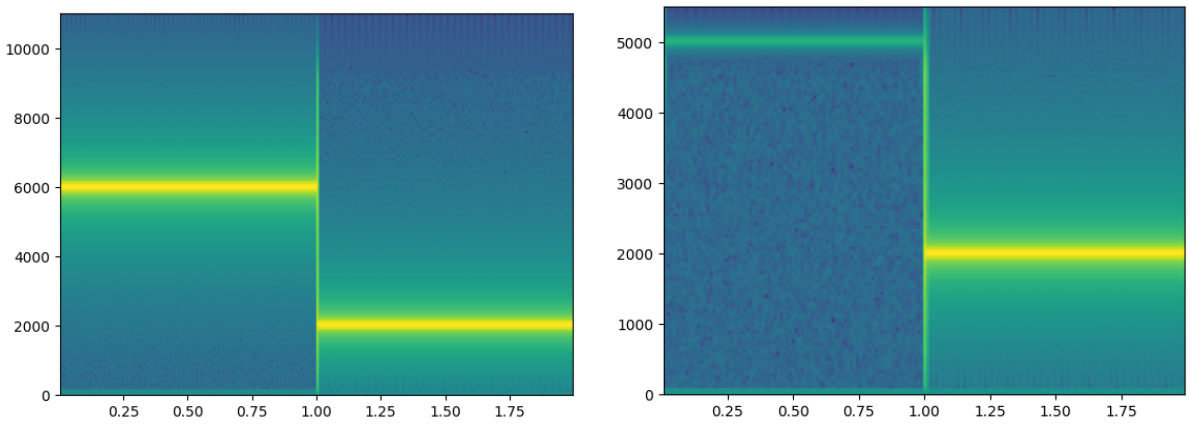
$$P_x = \left(\frac{Q2^{(n-1)}}{\sqrt{2}}\right)^2 = \frac{Q^2 2^{2(n-1)}}{2}$$

$$\mathbb{E}[\mathcal{E}^2] = \frac{1}{Q}\int_{-Q/2}^{Q/2} e^2 de = \frac{1}{Q}\left(\frac{(Q/2)^3}{3} - \frac{(-Q/2)^3}{3}\right) = \frac{Q^2}{12}$$

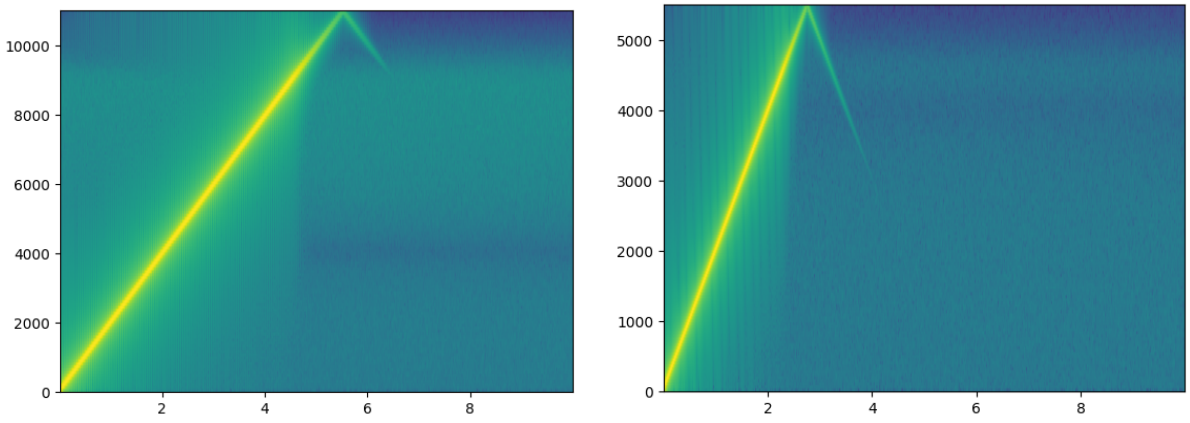
Por lo tanto, el SNR es:

$$SNR = \frac{Q^2 2^{2(n-1)}}{2}\frac{12}{Q^2} = 6 \cdot 2^{2(n-1)}$$

Expresando en decibeles, se tiene que:



(a) *Waves* $f_s = 22kHz$ (b) *Waves* $f_s = 11kHz$



(c) *Chirp* $f_s = 22kHz$ (d) *Chirp* $f_s = 11kHz$

Figura 4: Submuestro de las señales *wave* y *chirp* a distintas frecuencias utilizando *decimate*.

$$SNR_{dB} = 10\log\left(3 \cdot 2^{2n}\right) = 10\log\left(3\right) + 20n\log\left(2\right)$$

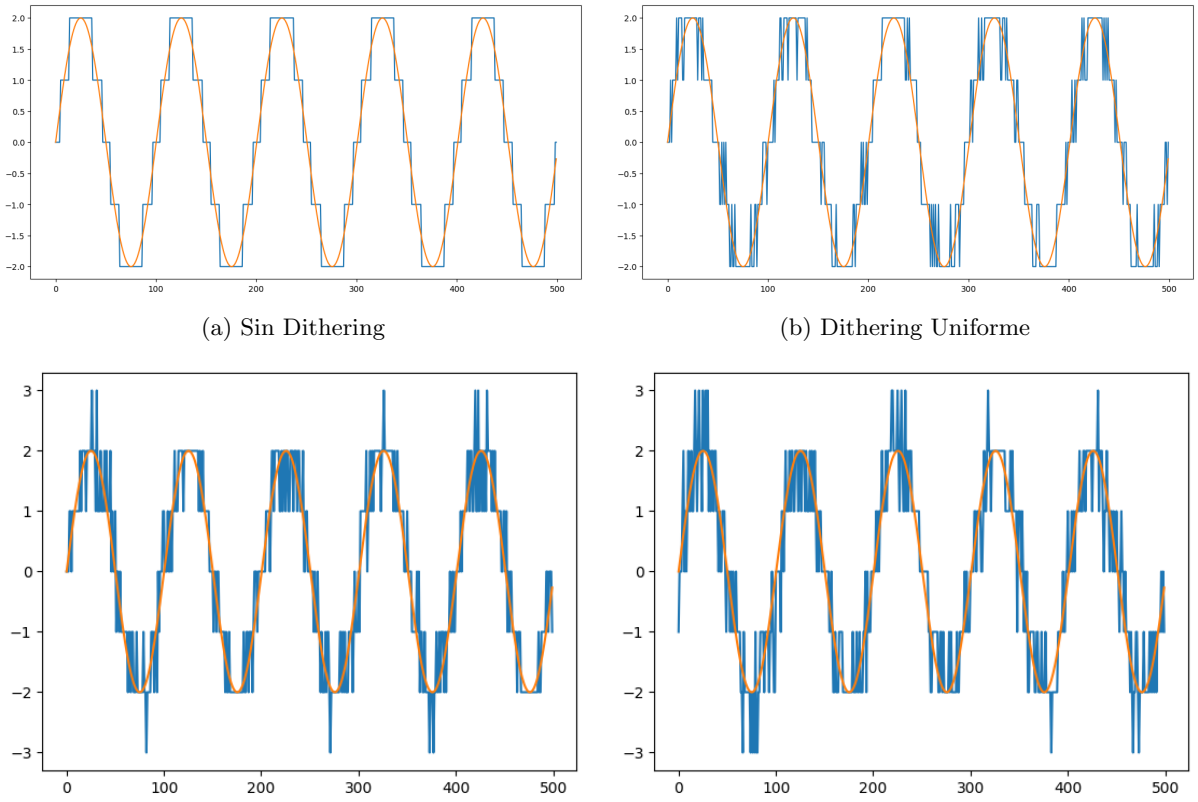
Por lo tanto, por cada bit extra que se utilice se esta mejorando en $20log(2)$ dB el ruido, cerca de 6 dB por bit, 12 dB por 2 bits.

Sin dithering la potencia del ruido de cuantizacion es la calculada anteriormente de $\frac{Q^2}{12}$. Al agregar dithering se le esta agregando un ruido previo a realizar la cuantización. En particular, en el caso del dithering triangular:

$$P_e = \int_{-Q}^{0} e^2 \left[\frac{e}{Q^2} + \frac{1}{Q}\right] de + \int_{0}^{Q} e^2 \left[-\frac{e}{Q^2} + \frac{1}{Q}\right] de = \left[\frac{e^4}{4Q^2} + \frac{e^3}{3Q}\right]_{-Q}^{0} + \left[\frac{-e^4}{4Q^2} + \frac{e^3}{3Q}\right]_{0}^{Q} = \frac{Q^2}{6}$$

Al cuantizar la señal con pocos niveles el error deja de cuantización deja de ser independiente entre una muestra y la otra. Cuando esto sucede, el error de cuantización deja de ser percibido como ruido, y comienza a ser un fenómeno de distorsión mas nocivo. Una forma de atacar este problema es agregar algún tipo de ruido previo al paso de cuantización. Con esto se vuelven a cumplir las hipótesis de la cuantización, a costa de un compromiso por un menor SNR. Esto es lo que se puede observar en las figuras 5, donde el agregar el ruido convierte la señal mas ruidosa pero tiene la ventaja que su forma vuelve a aproximarse mas a la de una sinusoide.

En la tabla 1 se ve el error entre la sinusoide original y las distintas cuantizaciones realizadas. Se ve



(a) Sin Dithering

(b) Dithering Uniforme

(c) Dithering Triangular

(d) Dithering Gaussiano

Figura 5: Señal cuantizada con distintos tipos de dithering.

como el error es considerablemente menor sin dithering. Sin embargo, perceptivamente se percibe mejor con dithering que sin dithering.

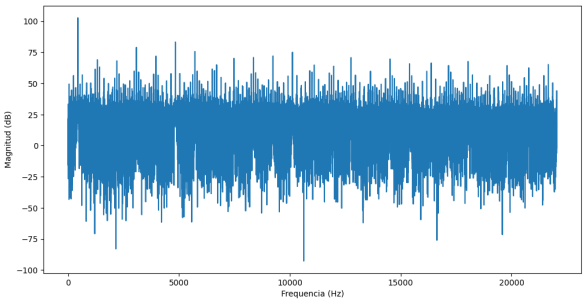
Tipo de Cuantización	**Potencia del Error**	**Potencia del Error en dB**
Sin Dithering	0.070	-11.6
Dithering Uniforme	0.147	-8.4
Dithering Triangular	0.251	-6.0
Dithering Gaussiano	0.333	-4.8

Cuadro 1: Potencia del error de cuantización y en dB para diferentes métodos.

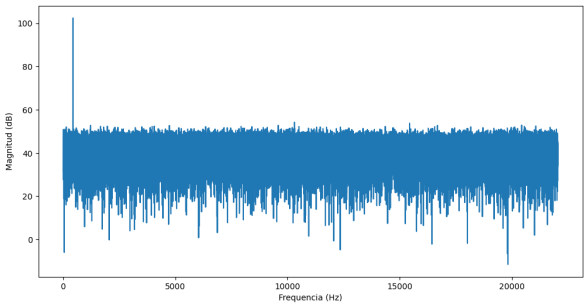
Finalmente, lo que se observo en la figura 5 en el tiempo se puede observar en la figura 6 en el espectro. Se ve como al agregar dithering todos los espectros se asemejan considerablemente mas a una sinusoide, es decir, una delta.

Por último, se implementó una técnica de *noise shaping*, la cual consiste en manipular el ruido de cuantización utilizando principios psicoacústicos para hacerlo menos audible. Esta técnica permite redistribuir el espectro del ruido hacia frecuencias más altas, donde el oído humano es menos sensible, reduciendo así su impacto perceptivo y mejorando la calidad percibida del audio.

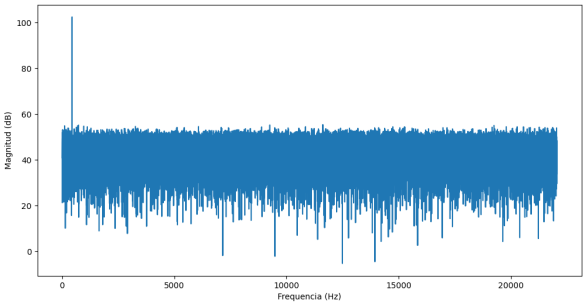
La figura **??** muestra el espectro de la señal (magnitud en logaritmo) de la señal sin *noise-shaping*. Se ve claramente como al utilziar la tecnica de *noise-shaping* (figura 8 el ruido en el espectro disminuye a frecuencias bajas, mientras aumenta a frecuencias altas. Sin embargo, estas frecuencias altas no son audibles,



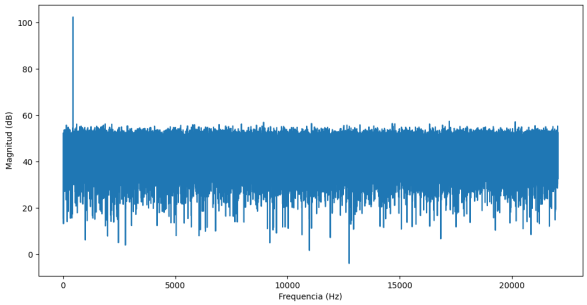
(a) Sin Dithering



(b) Dithering Uniforme



(c) Dithering Triangular



(d) Dithering Gaussiano

Figura 6: Espectro de la señal cuantizada con distintos tipos de dithering.

por lo que la calidad del audio mejora de forma considerable.

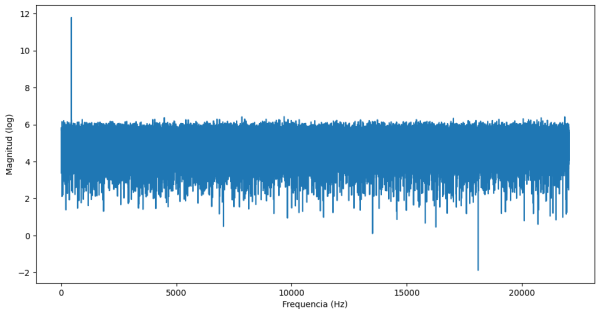


Figura 7: Espectro del audio sin *noise-shaping*

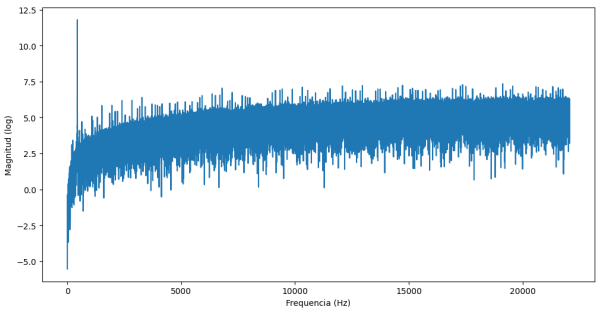


Figura 8: Espectro del audio con *noise-shaping*

2. Ejercicio 2

2.1. Parte 1

La energía y la magnitud en el tiempo corto de una señal $x[n]$, E_n y M_n respectivamente, se definen como:

$$E_n = \sum_{m=-\infty}^{\infty} x^2[m]w[n-m] \qquad M_n = \sum_{m=-\infty}^{\infty} |x[m]|w[n-m]$$

Para la ventana $w_a[n]$ definida como:

$$w_a[n] = \begin{cases} a^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

Por lo tanto, se tiene que:

$$E_n = \sum_{m=-\infty}^{\infty} x^2[m]a^{n-m} = \sum_{m=-\infty}^{n} x^2[m]a^{n-m} = a \sum_{m=-\infty}^{n-1} x^2[m]a^{(n-1)-m} + x^2[n]$$

El primer termino de la sumatoria es $a \cdot E_{n-1}$. Por lo tanto, se llega a la relación recursiva:

$$E_n = aE_{n-1} + x^2[n]$$

De forma análoga se llega a que:

$$M_n = aM_{n-1} + |x[n]|$$

Las figuras 9 y 10 muestran la energía y magnitud de la señal de interés utilizando las distintas ventanas. En este caso, cada pico en la energía o la magnitud se corresponde con una palabra. Sin embargo, esta técnica no es útil para tratar de detectar fonemas sordos, como la "s." la "p". Para esto, suelen utilizarse otras métricas, como puede ser la tasa de cruces por cero.

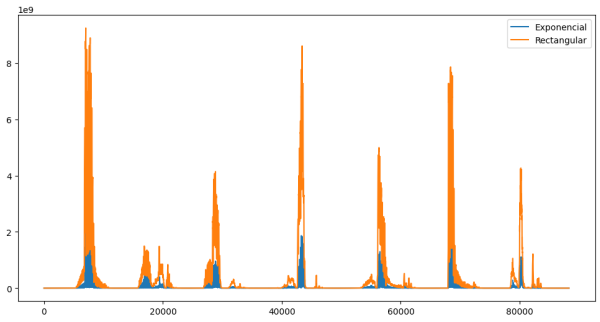


Figura 9: Energía para cada ventana

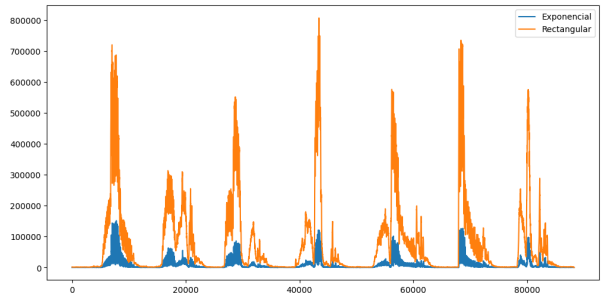


Figura 10: Magnitud para cada ventana

2.2. Parte 2

Como se menciono, la tasa de cruces por cero es una métrica utilizada principalmente para detectar los fonemas sordos del lenguaje.

$$Z_n = \sum_{m=-\infty}^{\infty} |\text{sign}(x[m]) - \text{sign}(x[m-1])|\ w[n-m] = \frac{1}{2N} \sum_{m=n-N+1}^{n} |\text{sign}(x[m]) - \text{sign}(x[m-1])|$$

donde $\text{sign}(x[m]) = \begin{cases} 1, & x[m] \geq 0 \\ -1, & x[m] < 0. \end{cases}$ y $w[n] = \begin{cases} \frac{1}{2N}, & 0 \leq n \leq N-1 \\ 0, & \text{en otro caso} \end{cases}$ Como indica el nombre, es una métrica que indica cuantas veces la señal cambia de signo en una ventana temporal.

Si se quiere escribir de forma recursiva utilizando Z_{n-1} se debe tener en cuenta que al ir moviendo la ventana en el tiempo de a una muestra, lo que se tiene es que la muestra mas vieja (muestra $x[n-N-1]$ ya no entra dentro de la ventana, mientras que una nueva muestra comienza a entrar en juego $x[n]$. Con esta

observacion, se pude plantear la resta

$$Z_n - Z_{n-1} = \frac{1}{2N} \sum_{m=n-N+1}^{n} |\text{sign}(x[m]) - \text{sign}(x[m-1])| - \frac{1}{2N} \sum_{m=n-1-N+1}^{n-1} |sign(x[m]) - sign(x[m-1])|$$

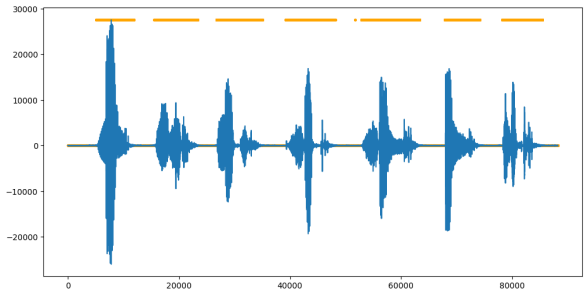
donde todos los términos que difieren son $|\text{sign}(x[n]) - \text{sign}(x[n-1])|$ y $|\text{sign}(x[n-N]) - \text{sign}(x[n-N-1])|$, que se repiten. De esta forma, se llega a que:

$$Z_n = Z_{n-1} + \frac{1}{2N} \left[|\text{sign}(x[n]) - \text{sign}(x[n-1])| - |\text{sign}(x[n-N]) - \text{sign}(x[n-N-1])| \right]$$

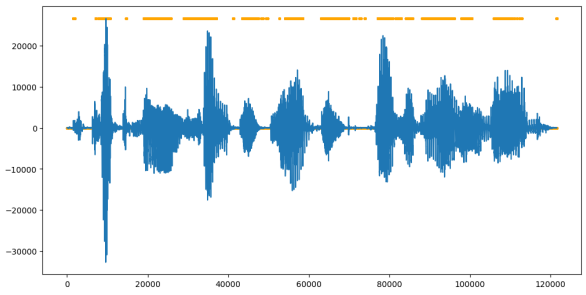
2.3. Parte 3

La figura 11 muestra el detector de palabras implementado para distintos audios. Las zonas naranjas indican que sucede una palabra en ese instante, mientras que la falta de estas indica que no se esta diciendo una palabra. Por lo tanto, para indicar el inicio y el final de una palabra alcanza con agrupar estos puntos. Una primera observación es que las palabras parecen detectarse en su mayoría de forma correcta, tanto cuando tienen fonemas sordos como cuando no, indicando que la combinación entre la métrica de energía y cruces por cero aporta valor a la solución.

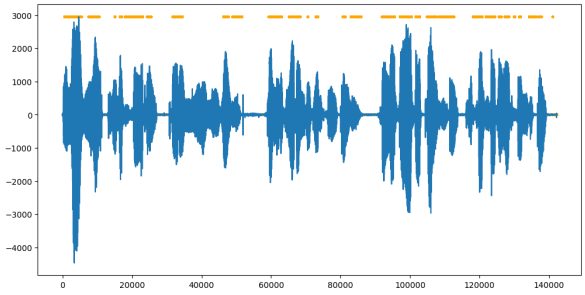
Una clara desventaja es que los parámetros utilizados dependen fuertemente de los audios. Una alternativa seria normalizar los audios, aunque seguirá dependiendo de las características especificas de los mismos.



(a) *ejemplo_palabras.wav*



(b) *fox.wav*



(c) *voice_2.wav*

Figura 11: Detector de Palabras en distintos audios

3. Ejercicio 3

3.1. Parte 1

La autocorrelación $R_n[k]$ de la señal $x[m]$ modulada por la ventana $w[n]$ se define como:

$$
\begin{aligned}
R_n[k] &= \sum_{m=-\infty}^{\infty} x[m]w[n-m]x[m+k]w[n+k-m] \\
&\overset{\tau=m+k}{=} \sum_{\tau=-\infty}^{\infty} x[\tau-k]w[n-\tau+k]x[\tau]w[n-\tau] \\
&= \sum_{\tau=-\infty}^{\infty} \underbrace{x[\tau]w[n-\tau]}_{R_n[\tau]} \underbrace{x[\tau-k]w[n-\tau+k]}_{R_n[-k]} \\
&= R_n[-k]
\end{aligned}
$$

Como se puede observar, al reorganizar los términos, se identifica la forma de la autocorrelación pero para $-k$. A partir de esto, se puede escribir la autocorrelación de manera más compacta:

$$
\begin{aligned}
R_n[k] &= R_n[-k] \\
&= \sum_{m=-\infty}^{\infty} x[m]w[n-m]x[m-k]w[n+k-m] \\
&= \sum_{m=-\infty}^{\infty} x[m]x[m-k] \underbrace{w[n-m]w[n+k-m]}_{h_k[n]}
\end{aligned}
$$

Donde $h_k[n] = w[n]w[n+k]$ encapsula la modulación de la ventana en los puntos n y $n+k$, simplificando la expresión original de la autocorrelación.

3.2. Parte 2

Para esta parte se utilizara la correlación para detectar las frecuencias fundamentales de una voz cantando. La frecuencia fundamental es la frecuencia más baja en un sonido complejo y es la que define el tono o "pitch" que percibimos en la voz o en un instrumento musical. Es el componente básico de una señal periódica, y todas las demás frecuencias presentes son armónicos de esta frecuencia.

Usar la autocorrelación tiene sentido para detectar la frecuencia fundamental porque esta técnica mide la similitud de una señal consigo misma en distintos tiempos. En una señal periódica, como la voz cantada, la autocorrelación ayuda a identificar el período de la señal, que corresponde directamente a la frecuencia fundamental. Al encontrar el primer pico significativo en la autocorrelación, podemos calcular el período y, a partir de él, obtener la frecuencia fundamental, lo que hace que sea un método eficaz para este tipo de análisis.

Siguiendo esta técnica, se paso a detectar la frecuencia fundamental de una mujer cantando. Luego de un post procesado de la señal (filtro pasa altos, disminuir el valor de las anomalías) se obtuvo la señal de la figura 13, la cual se asemeja al valor de referencia.

3.3. Parte 3

Se implementó una técnica de procesamiento de señales en ventanas de tiempo corto, en la cual la señal de audio se divide en pequeñas tramas o fragmentos. Para cada una de estas tramas, se calcula el valor máximo de la señal, creando así un contorno que representa su forma general a lo largo del tiempo. Este enfoque permite simplificar la representación de la señal, capturando solo los picos más relevantes de cada fragmento, en lugar de procesar todas las muestras individuales.

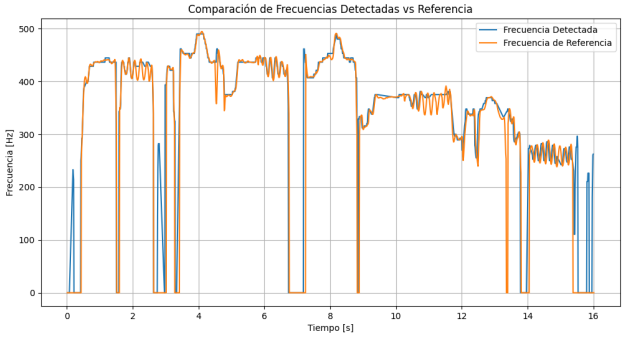


Figura 12: Frecuencia Fundamental Detectada y Real

Esta técnica es útil en aplicaciones de edición y visualización de audio, ya que reduce significativamente la cantidad de datos a manejar. Al trabajar con una versión más resumida de la señal, se requieren menos recursos computacionales, lo que mejora el rendimiento de la interfaz gráfica y permite que responda de manera más ágil. Esto es especialmente valioso en situaciones donde se necesita manipular audio en tiempo real o visualizar señales largas sin que el sistema se sobrecargue.

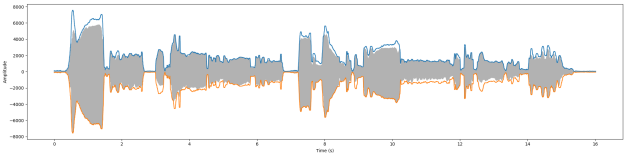


Figura 13: Contorno de la Forma de Onda

Referencias