

Taller de Aprendizaje Automático

Segundo Proyecto

Procesamiento de Señales de Audio

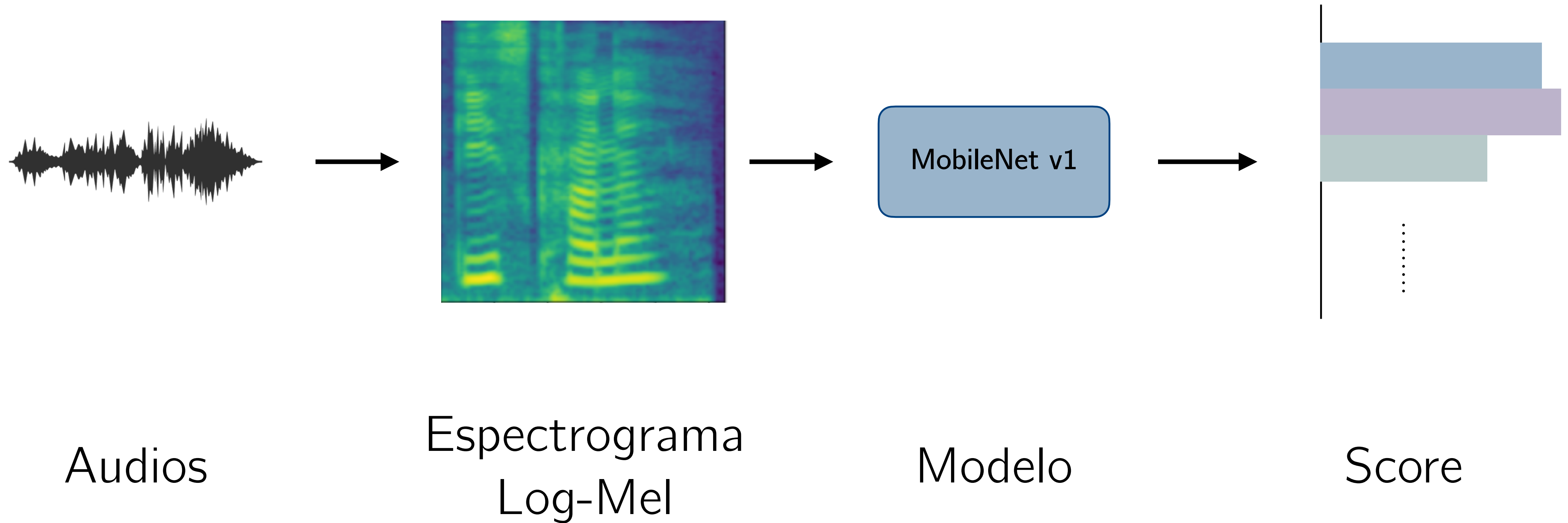
Instituto de Ingeniería Eléctrica
Facultad de Ingeniería



UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

Montevideo, 2024

Baseline



¿Por qué utilizar el espectrograma?

- Gran cantidad de la información de un sonido está contenida en la **distribución relativa de energía en las distintas frecuencias.**
- La transformación más utilizada para las señales de audio es la **transformada discreta de Fourier**
- Las **señales de audio suelen ser no estacionarias.** Por ello, se extraen características utiliza el método de procesamiento en tiempo corto.

Cálculo del Espectrograma

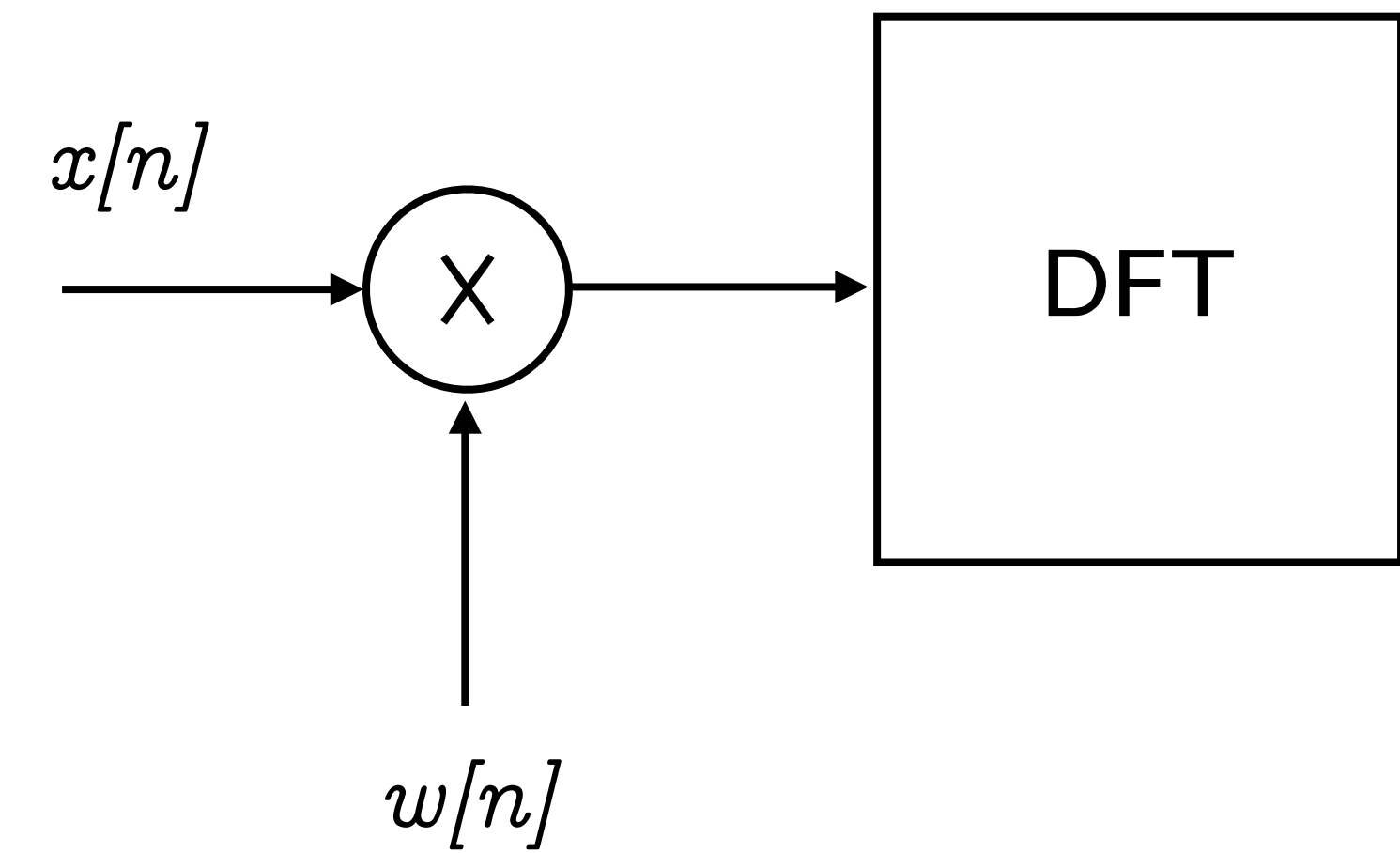
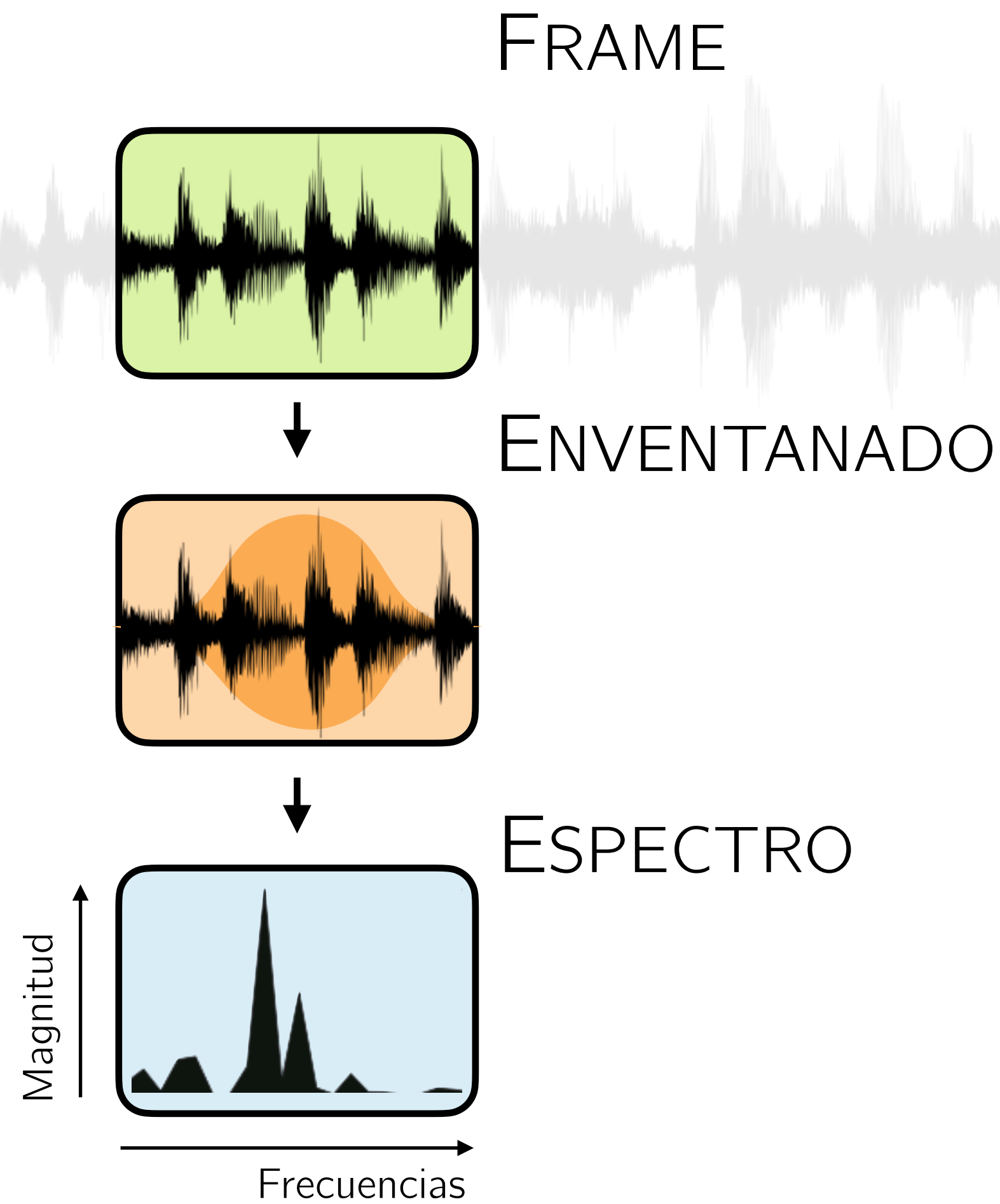
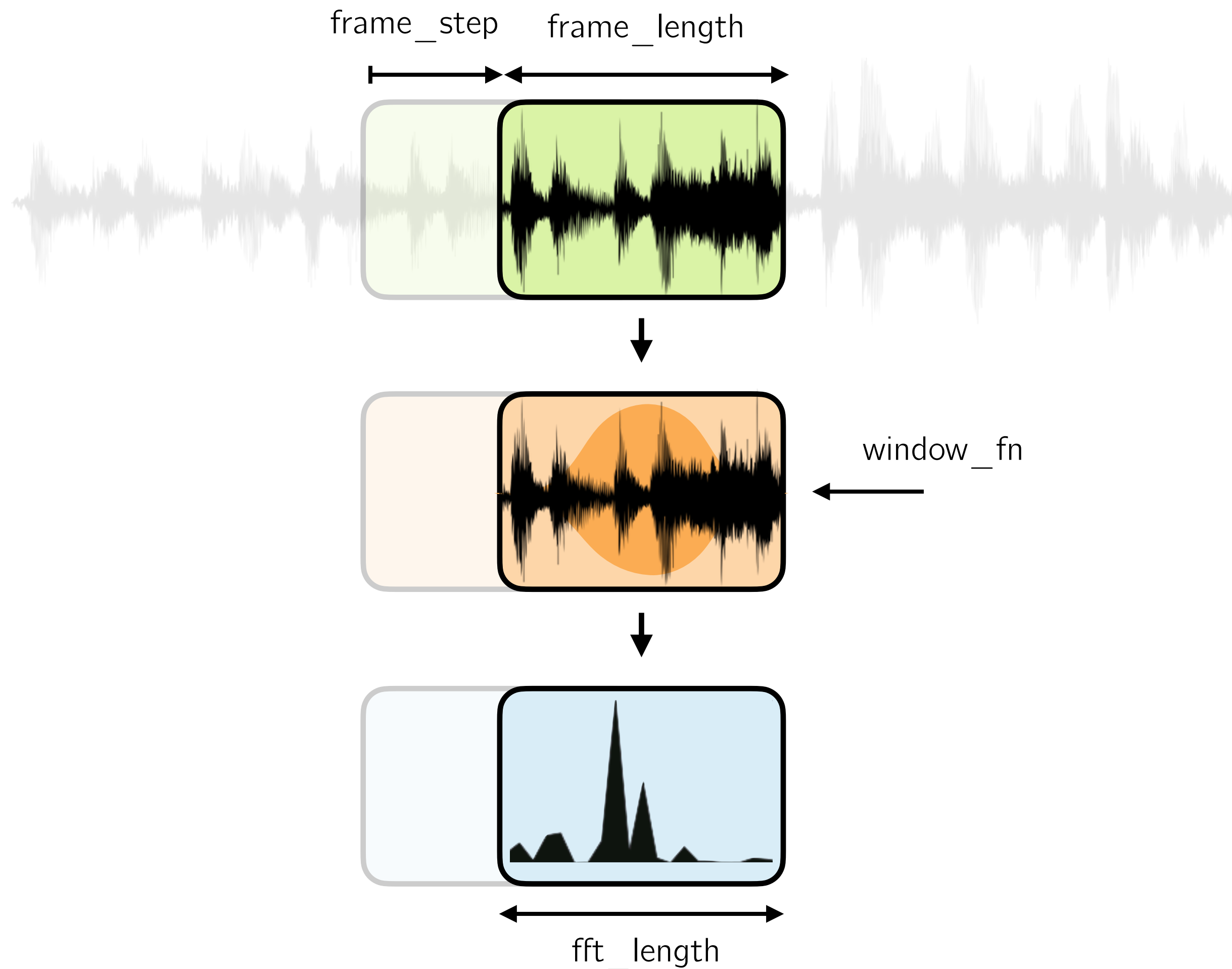


Diagrama de bloques para cálculo de la STFT

Figura Adaptada de [2]

Cálculo del Espectrograma



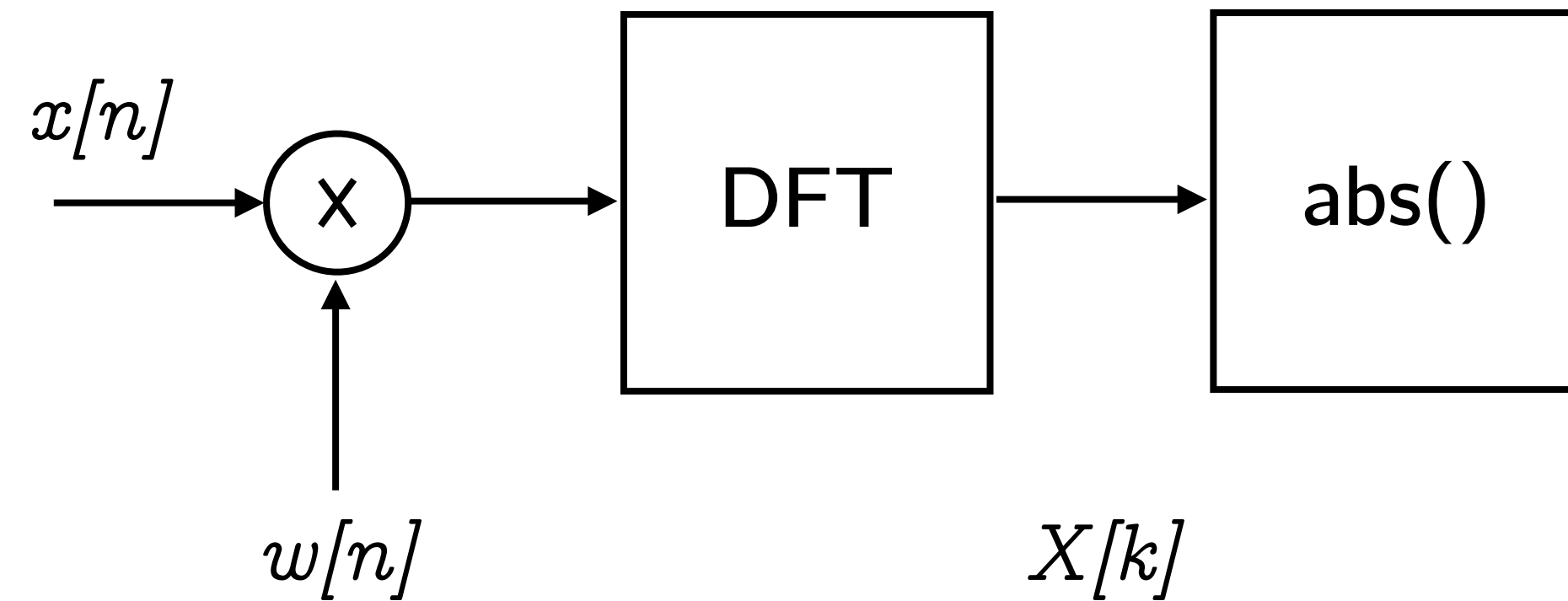
```
# Convertir la una señal de audio en un  
espectrograma mediante la STFT
```

```
stft = tf.signal.stft(  
    waveform,  
    frame_length = 1024,  
    frame_step = 512,  
    window_fn = tf.signal.hann_window,  
    fft_length = 1024,  
)
```

[Documentación de Tensorflow](#)

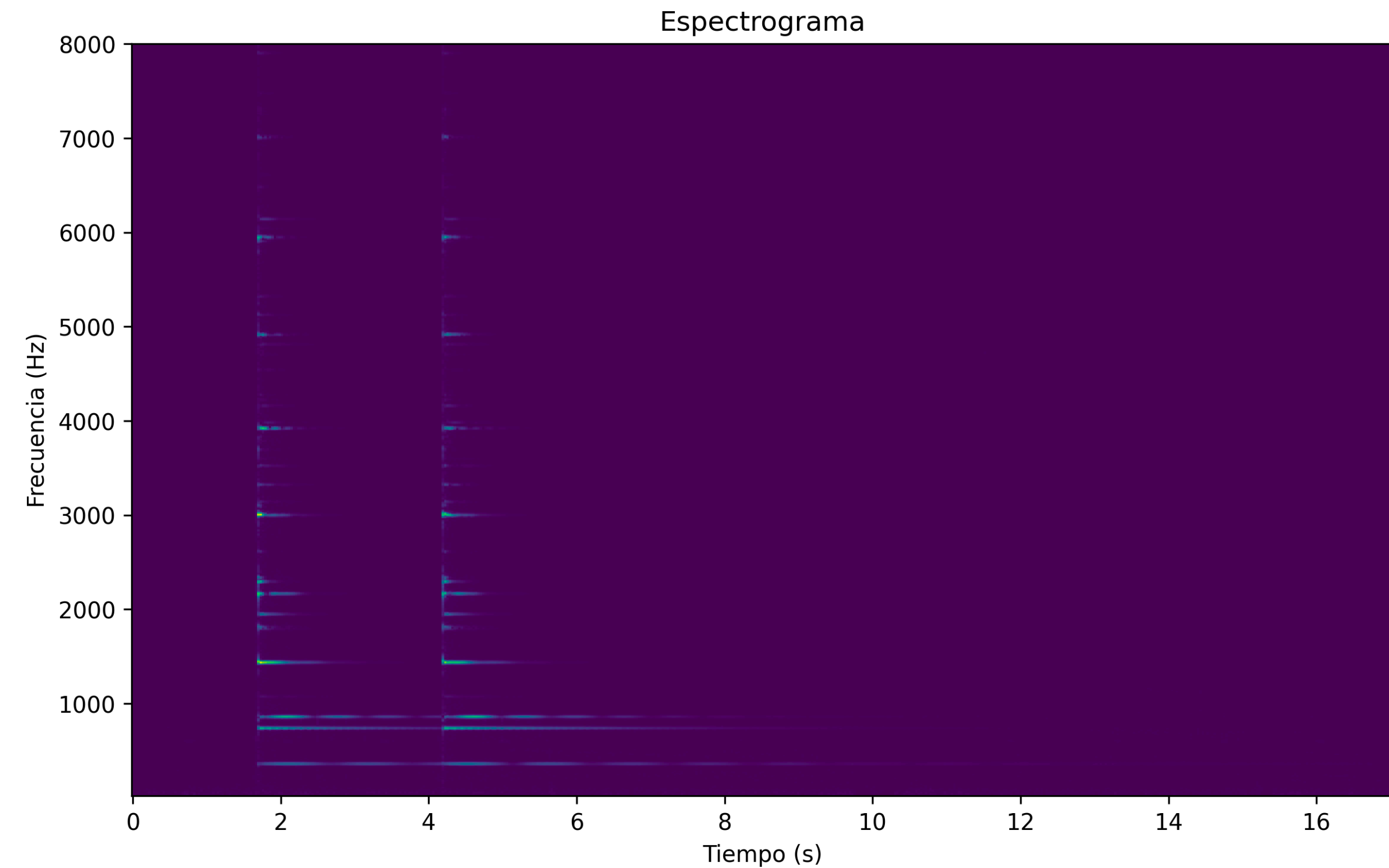
Figura Adaptada de [2]

Cálculo del Espectrograma

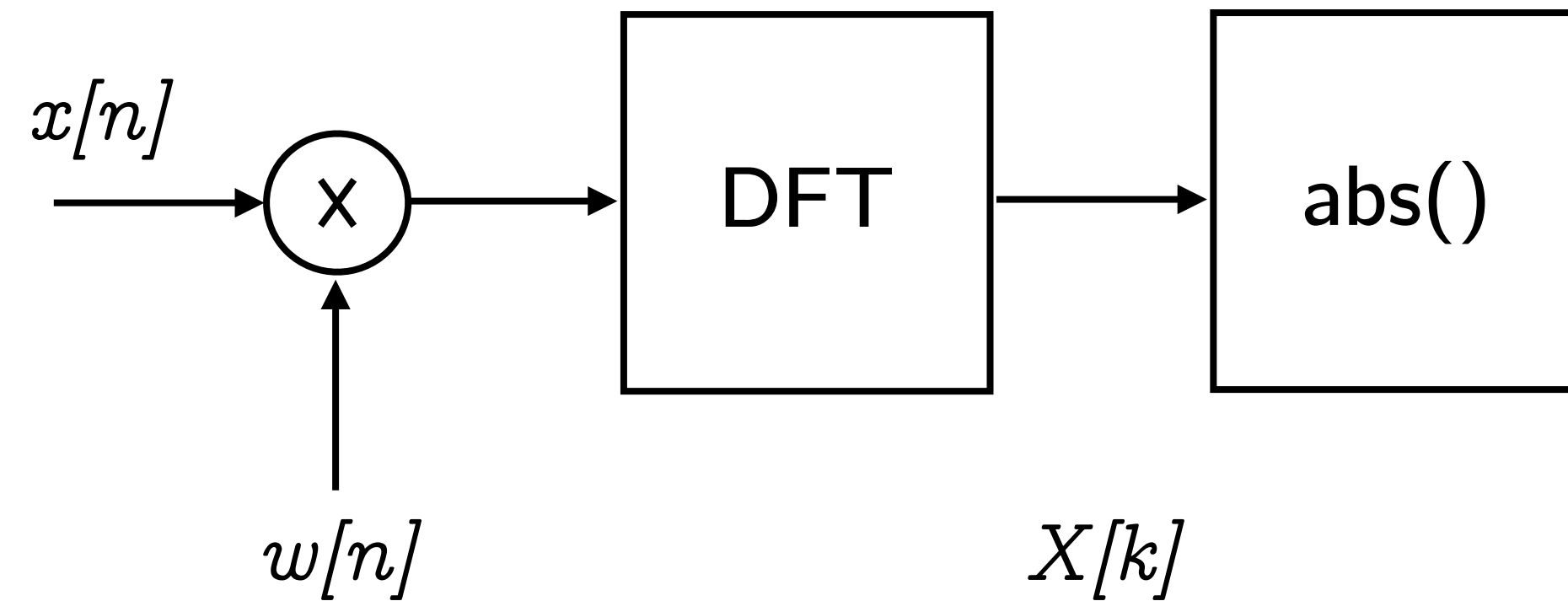


```
# Calculo de la STFT
stft = tf.signal.stft(
    waveform,
    frame_length = 1024,
    frame_step = 512,
    window_fn = tf.signal.hann_window,
    fft_length = 1024,
)

# Obtener la magnitud de la STFT.
spectrogram = tf.abs(stft)
```

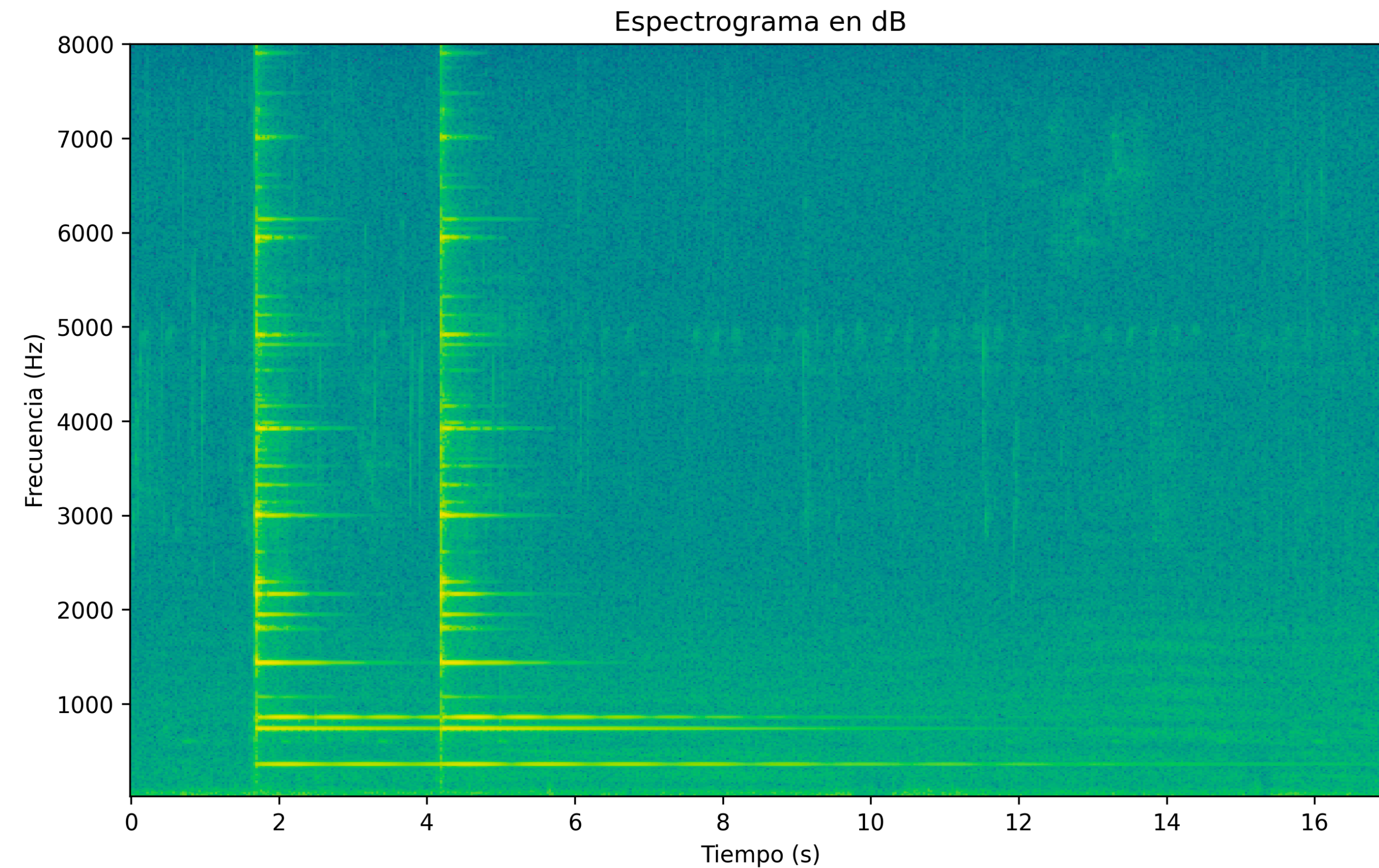


Cálculo del Espectrograma



```
# Calculo de la STFT
stft = tf.signal.stft(
    waveform,
    frame_length = 1024,
    frame_step = 512,
    window_fn = tf.signal.hann_window,
    fft_length = 1024,
)

# Obtener la magnitud de la STFT.
spectrogram = tf.abs(stft)
```



¿Por qué aplicar transformaciones en el espectrograma?

- Tanto la percepción de las magnitudes como la percepción de las frecuencias son **no lineales**.
 - ▶ Se utiliza una representación no lineal para las magnitudes - **Logaritmo**
 - ▶ Se utiliza una escala de frecuencias no lineal implementada mediante bancos de filtros - **Escala Mel**

Cálculo del Espectrograma Log-Mel

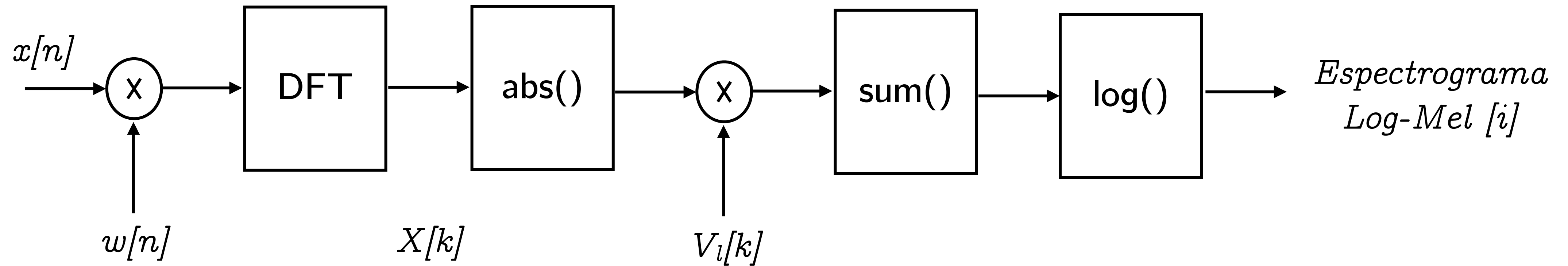


Diagrama de bloques para cálculo del espectrograma log-mel

Cálculo del Espectrograma Log-Mel

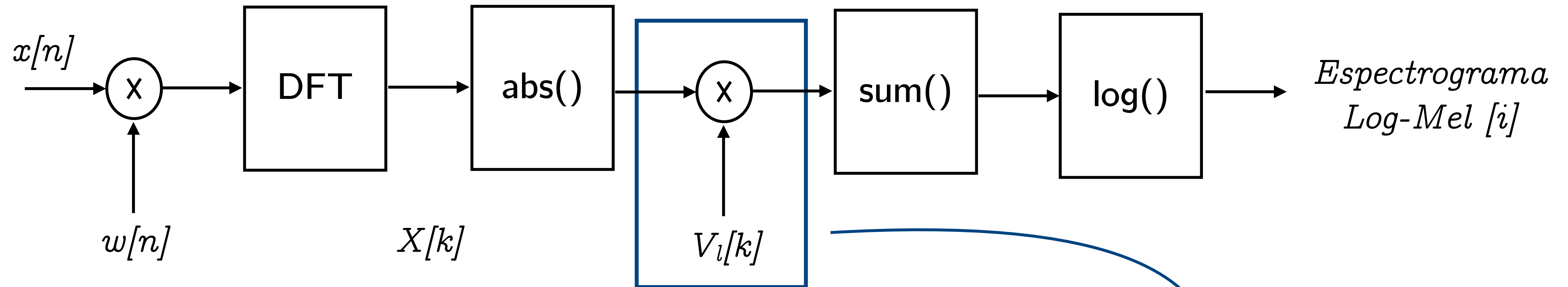
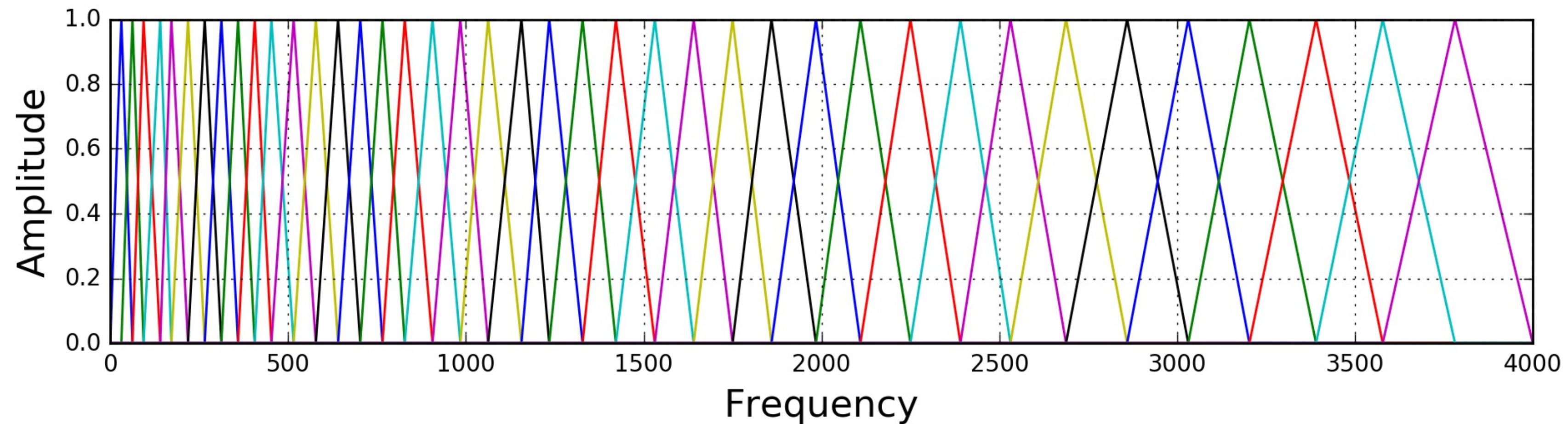
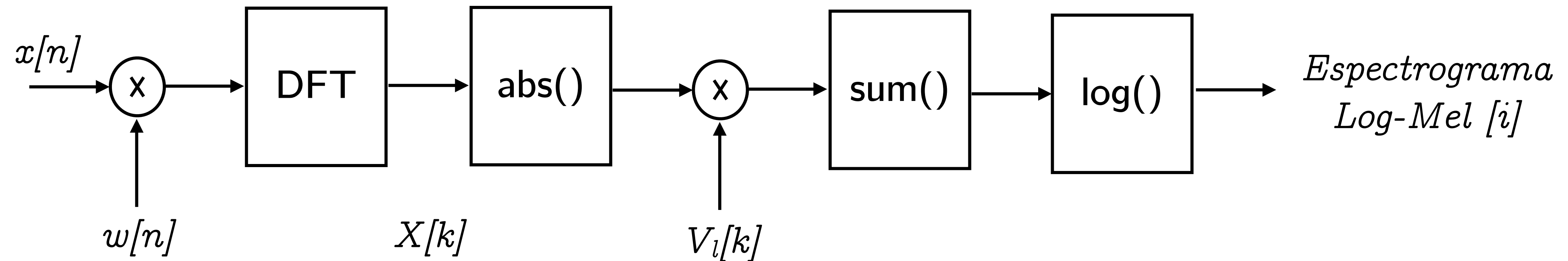


Diagrama de bloques para cálculo del espectrograma log-mel



Cálculo del Espectrograma Log-Mel



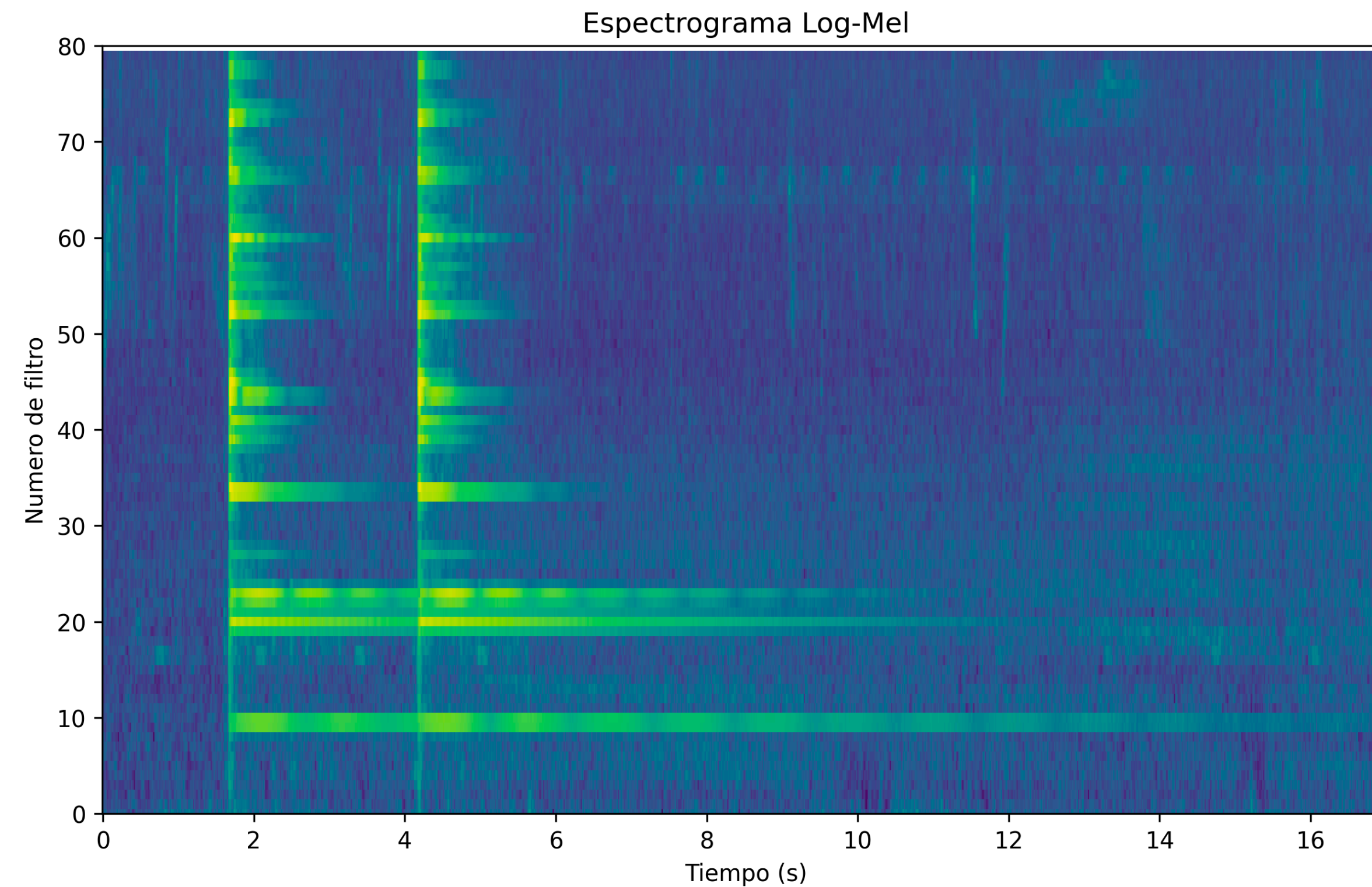
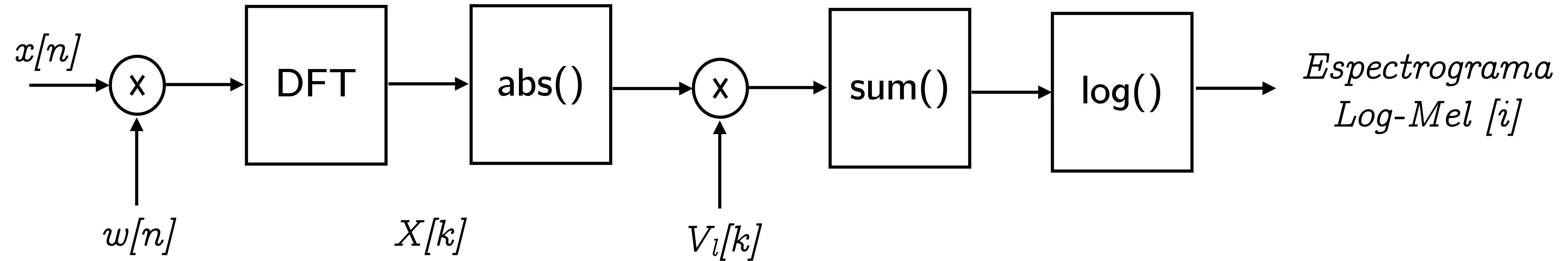
```
# Calculo de la STFT
stfts = tf.signal.stft( waveform, frame_length = 1024, frame_step = 512, fft_length = 1024)
# Obtener la magnitud de la STFT.
spectrogram = tf.abs(stfts)

# Se define el banco de filtros a utilizar
num_spectrogram_bins = stfts.shape[-1]
lower_edge_hertz, upper_edge_hertz, num_mel_bins = 0, 4000, 80
linear_to_mel_weight_matrix = tf.signal.linear_to_mel_weight_matrix (num_mel_bins, num_spectrogram_bins, sample_rate,
                                                                    lower_edge_hertz, upper_edge_hertz)

# Se aplica el banco de filtros sobre el espectrograma
mel_spectrograms = tf.tensordot(spectrograms, linear_to_mel_weight_matrix, 1)
mel_spectrograms.set_shape(spectrograms.shape[:-1].concatenate(linear_to_mel_weight_matrix.shape[-1:]))

# Calculo el Espectrograma en magnitud logaritmica y escala mel
log_mel_spectrograms = tf.math.log(mel_spectrograms + 1e-6)
```

Cálculo del Espectrograma Log-Mel



Cálculo del Espectrograma Log-Mel

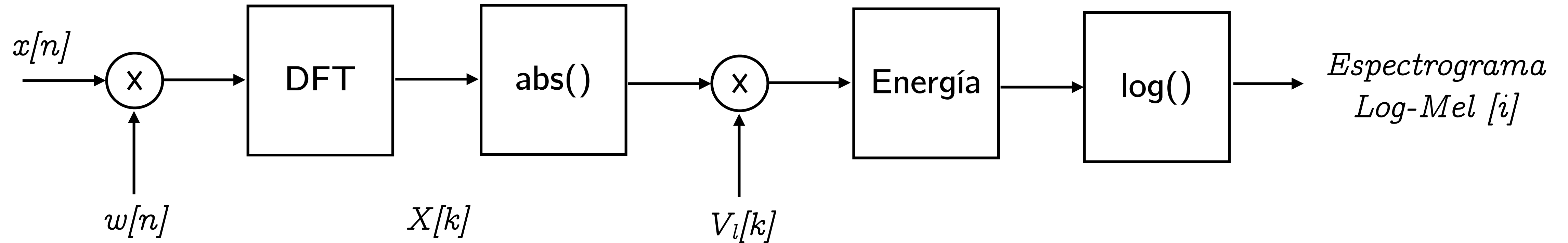
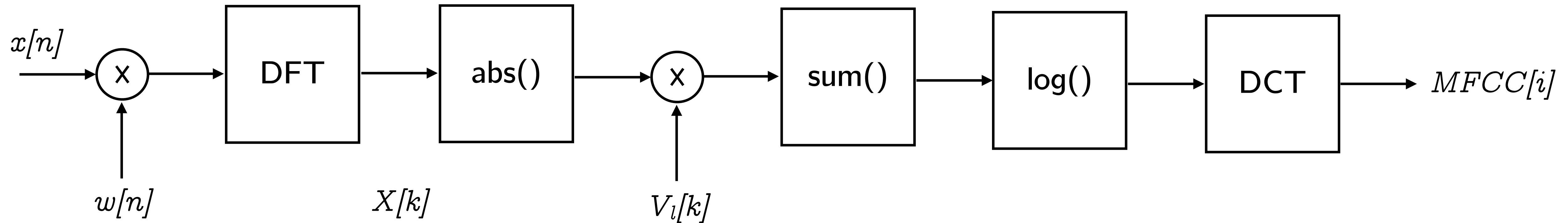


Diagrama de bloques para cálculo espectrograma log-mel

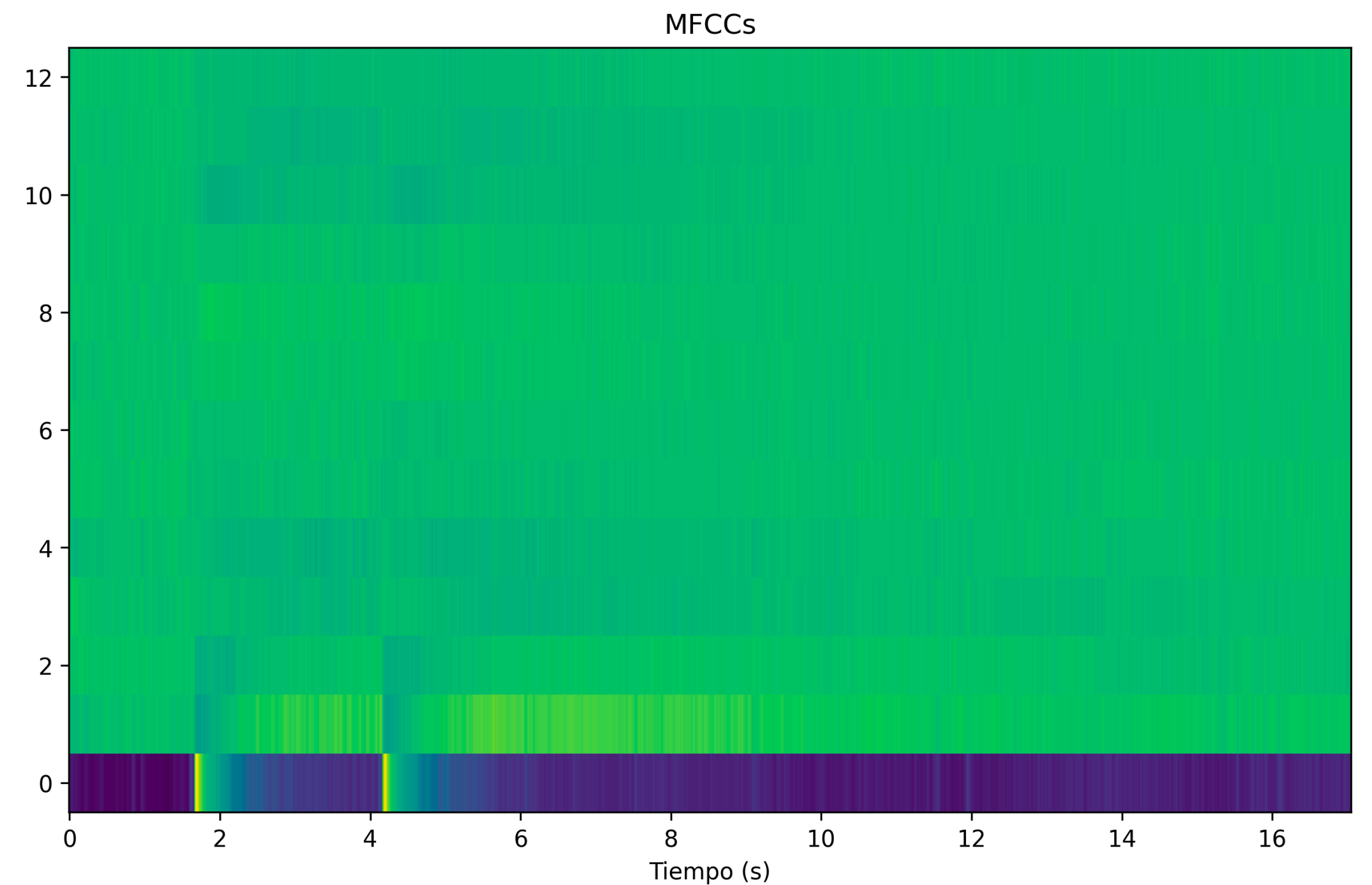
Cálculo de los MFCCs



```
# Calculo el Espectrograma en magnitud logaritmica y escala mel
log_mel_spectrograms = tf.math.log(mel_spectrograms + 1e-6)

# Calculo los MFCCs a partir del log_mel_spectrograms y tomo
los primeros 13
mfccs = tf.signal.mfccs_from_log_mel_spectrograms(
    log_mel_spectrograms)[..., :13]
```

[Documentación de Tensorflow](#)



Referencias

- [1] Virtanen, T., Plumbley, M. D., & Ellis, D. (Eds.). (2018). Computational Analysis of Sound Scenes and Events. Springer.
- [2] Quateri, T.F. (2001). Discrete-Time Speech Signal Processing: Principles and Practice. Upper Saddle River, NJ: Prentice Hall PTR
- [3] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," in IEEE Transactions on Acoustics, Speech, and Signal Processing