UNTREF UNIVERSIDAD NACIONAL DE TRES DE FEBRERO

INGENIERÍA DE SONIDO

Dereverberación del habla a partir de algoritmos de aprendizaje profundo†

Autor: Martin Bernardo Meza
Tutores: Ing. Leonardo Pepino

(†) **Tesis para optar por el titulo de ingeniero/a de Sonido**.

Octubre 2021

Índice

1. Introducción **1**

1.1. Fundamentación 1

1.2. Objetivos 2

1.2.1. Objetivo general 2

1.2.2. Objetivos específicos 2

1.3. Estructura de la Investigación 2

2. Estado del Arte **4**

3. Marco Teórico **7**

3.1. Representación temporal y frecuencial de señales 7

3.1.1. Transformada de corto término de Fourier (STFT) 8

3.2. Respuesta al impulso y reverberación 11

3.2.1. Relación directo-reverberado (DRR) 14

3.3. Inteligibilidad y parámetros de calidad de percepción 15

3.3.1. Relación energía de modulación de voz a reverberación 16

3.3.2. Inteligibilidad objetiva de corto termino extendida 16

3.3.3. Relación señal a distorsión 16

3.4. Redes neuronales y algoritmos de aprendizaje profundo 17

3.4.1. La neurona artificial 17

3.4.2. Modelos basados en redes neuronales 18

3.4.3. Entrenamiento y aprendizaje 19

3.4.4. Redes neuronales convolucionales 23

3.5. Dereverberación por filtrado temporal-frecuencial 26

3.5.1. Máscaras de amplitud 26

3.5.2. Síntesis de audio a partir de espectrogramas 28

4. Metodología **30**

4.1. Análisis de datos 30

4.2. Base de datos de respuestas al impulso 30
4.2.1. Respuestas al impulso reales 30
4.2.2. Respuestas al impulso generadas 31
4.2.3. Respuestas al impulso generadas por aumentación 33
4.3. Bases de datos de señales de habla 36
4.3.1. Pre-procesamiento de datos 36
4.4. Modelo propuesto 38
4.5. Especificaciones de la arquitectura implementada 41
4.6. Evaluación del modelo 43
4.6.1. Combinaciones de bases de datos 43
4.6.2. Ordenamiento de los datos durante el entrenamiento 44

5. Resultados y Discusiones **45**
5.1. Bases de datos de respuestas al impulso 45
5.2. Funcionamiento del sistema 47
5.3. Reconstrucción de audio dereverberado 49
5.4. Dereverberación del habla y manejo de datos 52
5.5. Aprendizaje por curriculum 56

6. Conclusiones **60**

7. Lineas futuras de investigación **61**

Bibliografía **63**

Anexos **68**

A. Aumentación de tiempo de reverberación **68**

Índice de figuras

1. Proceso de obteción de la STFT. Extraido y adaptado de [37]. 9
2. Ventanas **(a)** rectangular, **(b)** triangular y **(c)** Hann, con sus respectivas respuestas en frecuencia. Extraido de [38]. 10
3. Espectrograma de una señal de audio. 10
4. Efecto del solapamiento entre ventanas en la STFT. 11
5. Secciones temporales de una respuesta al impulso. Extraido de [39]. 13
6. Modelos analíticos de cálculo de la respuesta al impulso de un recinto. Extraido de [39]. 14
7. Esquema de neurona artificial. Extraído de [48]. 17
8. Funciones de activación y sus derivadas. Extraído de [27]. 18
9. Ejemplo de red neuronal multicapa con alimentación hacia adelante. 19
10. Diagrama de flujo del bucle de entrenamiento de una sistema de red neuronal 21
11. Curva de costo para un parámetro 22
12. Funcionamiento de un filtro de convolución. Extraido de [48]. 24
13. Representación de la aplicación de un filtro bidimensional sobre una imagen . 24
14. Parámetros de procesamiento en capas convolucionales 25
15. Capas convolucionales con campos receptivos locales rectangulares 26
16. Disposición de las entradas y salidas del modelo para la estimación de máscaras de amplitud 28
17. Diagrama en bloques del algoritmo de Griffin-Lim. 29
18. Recintos y puntos receptor-fuente generados para la simulación de respuestas al impulso 32
19. Señales involucradas en el proceso de aumentación de DRR 34
20. Estructura general de un autoencoder. 38
21. Esquema básico de una red tipo 'U-NET' con conexiones de salto. 39
22. Modelo de red neuronal convolucional implementado 41
23. Conjunto de respuestas al impulso reales. 45
24. Conjunto de respuestas al impulso generadas. 46

25. Conjunto de respuestas al impulso aumentadas. 46
26. The average and standard deviation of critical parameters 48
27. Espectrogramas de magnitud y fase de los audios para entrenamiento 49
28. Influencia del número de iteraciones del algoritmo de Griffin-Lim. 50
29. Variaciones de SDR para el primer conjunto de pruebas. 52
30. Variaciones de SRMR para el primer conjunto de pruebas. 53
31. Variaciones de ESTOI para el primer conjunto de pruebas. 53
32. Variaciones de SDR para el segundo conjunto de pruebas. 54
33. Variaciones de SRMR para el segundo conjunto de pruebas. 54
34. Variaciones de ESTOI para el segundo conjunto de pruebas. 55
35. Respuestas al impulso generadas por aumentación. 56
36. Comparación de SDR entre tipos de ordenamiento de datos durante el entrenamiento. 57
37. Comparación de SRMR entre tipos de ordenamiento de datos durante el entrenamiento. 57
38. Comparación de ESTOI entre tipos de ordenamiento de datos durante el entrenamiento. 58
39. Descomposición de la respuesta al impulso a procesar durante la aumentación temporal 69
40. Banco de filtros Butterworth 70
41. Sub-bandas obtenidas luego de aplicar el banco de filtros 71
42. Estimación paramétrica de la pendiente de caída. 73
43. Respuesta original y extendida sin piso de ruido. 74
44. Aumentación del tiempo de reverberación alterando la envolvente de caída original. 74
45. Resultado del proceso de aumentación del tiempo de reverberación de una respuesta al impulso 75

Índice de tablas

1. Configuración del primer conjunto de pruebas. 43
2. Configuración del segundo conjunto de pruebas. 44
3. Comparación de métodos de reconstrucción de espéctrograma complejo para generar audio 51
4. Resultados de las métricas sobre los conjuntos reverberados 52
5. Medianas correspondientes a cada esquema de entrenamiento. 58

RESUMEN

En este trabajo se estudia la dereverberación de señales de habla a partir de algoritmos de aprendizaje profundo. Se implementa una de red neuronal convolucional tipo autoencoder con conexiones de salto, basada en el estado del arte actual, para estimar máscaras de amplitud que realicen la dereverberación del habla en el dominio de la transformada de tiempo corto de Fourier. Uno de los problemas de esta tarea es la escasa cantidad de datos disponibles, por lo que se analizan técnicas de generación y aumentación de datos, evaluando su impacto en el desempeño del sistema. Además, se evalúa el efecto que tiene el ordenamiento de los datos de entrenamiento y el tratamiento de la información de fase. Los resultados indican que las técnicas de generación y aumentación de datos permiten mejorar el rendimiento final del sistema. A su vez, ordenar los datos de entrenamiento con un tiempo de reverberación creciente tuvo un impacto positivo en las métricas de evaluación. Por último, se proponen mejoras al enfoque utilizado, y líneas futuras de investigación.

Palabras clave: “Dereverberación del habla”; “Redes Neuronales Convolucionales”; “Respuestas al Impulso”

SUMMARY

…

Keywords: “Speech dereverberation”; “Convolutive Neural Networks”; “Room Impulse Response”

CAPÍTULO 1: INTRODUCCIÓN

1.1 FUNDAMENTACIÓN

Las tecnologías de procesamiento digital de señales de voz mostraron grandes avances en las últimas décadas, llegando a ocupar un rol importante en nuestro día a día. Las investigaciones realizadas en este campo impulsaron diversas aplicaciones basadas en el análisis de la voz humana [1][2]. Estas aplicaciones, en mayor o menor medida, deben lidiar con una característica intrínseca a cualquier emisión sonora dentro de un recinto: la reverberación. Esto se debe principalmente a que las señales de voz se obtienen a partir de un transductor que no siempre se encuentra cercano a la fuente que se desea registrar, provocando que la señal registrada contenga la reverberación propia del entorno. Esta reverberación interfiere en con la señal de voz, produciendo una reducción en el rendimiento de aquellas aplicaciones que dependen de la integridad de dicha señal, como por ejemplo:

- Reconocimiento del habla [3]
- Verificación del hablante[1] [4]
- Localización del hablante [5]
- Mejora del habla [6]

Si bien esta problemática fue abordada desde el enfoque de diversas técnicas de procesamiento de señales, en los últimos años ocurrieron grandes avances producto de la implementación de una tecnología emergente de amplio crecimiento en el ambiente científico: los algoritmos de aprendizaje profundo. La capacidad y robustez de estos métodos a la hora de resolver problemas pertinentes al procesamiento de imágenes y detección de patrones se vio también reflejada en el campo de la dereverberación de habla. Actualmente, los sistemas basados en modelos de aprendizaje profundo representan el estado del arte, tanto en dereverberación del habla como otras tareas de audio, desplazando a enfoques mas clásicos del procesamiento de señales. Sin embargo, aún quedan desafíos por resolver como por ejemplo: la falta de bases de

[1] Debe distinguirse entre reconocimiento del habla y verificación del hablante. Lo primero refiere a poder distinguir qué palabras fueron dichas, y lo segundo refiere a identificar quién es el que está pronunciando las palabras.

datos masivas de señales acústicas específicas como respuestas al impulso reales, la selección de una representación óptima de las señales que permita explotar sus características intrínsecas, las formas de evaluar y cuantificar el desempeño de los sistemas, entre otros.

Por este motivo, esta investigación pretende realizar un análisis de esta problemática desde el punto de vista de la ingeniería de sonido, para comprender las limitaciones de los modelos actualmente utilizados en este campo de estudio, analizar alternativas posibles a la escasez de datos y poder aportar al progreso y mejora del rendimiento de dichos modelos.

1.2 OBJETIVOS

1.2.1 Objetivo general

El objetivo general de esta investigación es implementar un algoritmo de dereverberación de señales de voz a partir del uso de redes neuronales y algoritmos de aprendizaje profundo.

1.2.2 Objetivos específicos

- Realizar una revisión de las técnicas utilizadas para resolver el problema de dereverberación.
- Diseñar e implementar una arquitectura de red neuronal para dereverberación de señales de voz en lenguaje Python.
- Analizar técnicas de pre y post procesamiento de datos, estudiando el impacto que tienen en el rendimiento del algoritmo.
- Optimizar el sistema propuesto, y evaluar los resultados obtenidos de manera objetiva.
- Estudiar y analizar técnicas de generación y aumentación de datos, evaluando su impacto en el desempeño del sistema implementado.

1.3 ESTRUCTURA DE LA INVESTIGACIÓN

En el capítulo 2 se presenta el estado del arte referido a las técnicas de dereverberación de señales del habla. En el capítulo 3 se detalla el marco teórico necesario para el seguimiento

y comprensión de este trabajo. En este se abordan tres temáticas principales: la representación de señales de audio en el dominio espectral mediante la transformada de tiempo corto de Fourier, el concepto de reverberación y su relación con la respuesta al impulso, y por último la aplicación de redes neuronales convolucionales y algoritmos de aprendizaje junto con las principales técnicas de procesamiento. En el capítulo 4 se especifica de manera detallada la metodología seguida a lo largo de este trabajo, y se brinda toda la información necesaria para replicar los experimentos realizados. En el capítulo 5 se presentan los resultados de los experimentos y se hace un análisis crítico de los mismos. En el capítulo 6 se exponen las conclusiones generales del trabajo, y por último en el capítulo 7 se proponen líneas futuras de investigación relacionadas con el presente trabajo.

CAPÍTULO 2: ESTADO DEL ARTE

En los últimos años se ha registrado un marcado desarrollo y progreso en el campo de el procesamiento de señales del habla. En este campo, la dereverberación ocupa un rol crucial, debido al impacto negativo que genera la presencia de reverberación en muchas aplicaciones del procesamiento de señales de habla.

Los primeros enfoques que apuntaron a resolver el problema de la dereverberación se orientaron al modelado o registro de las respuestas al impulso y la estimación de filtros inversos a partir de estas [7]. Como el efecto de la reverberación en una señal se puede pensar como el resultado de una convolución entre una señal anecoica y una respuesta al impulso, este enfoque apunta a estimar la respuesta al impulso con el fin de poder generar un filtro inverso que permita realizar una deconvolución de la señal para poder revertir el efecto de la respuesta del recinto, recuperando la señal en su estado anecoico. Sin embargo, esta metodología presenta varios inconvenientes, como el hecho de considerar que las respuestas al impulso son lineales e invariantes en el tiempo, lo cual no siempre se cumple en la práctica [8], o bien el hecho de que la respuesta no siempre pueda ser deducida de manera directa y deba ser estimada.

También surgieron trabajos enfocados en modelar matemáticamente la señal de habla anecoica [9]. Algunos de estos trabajos consistían en estimar la señal de habla mediante predicción lineal, y calcular el residuo, el cual contiene información sobre la reverberación en la señal. Esta señal de residuo se utilizó para estimar filtros variantes en el tiempo que al aplicarse a la señal de habla lograban eliminar parte de la reverberación [10]. Otro enfoque consistió en utilizar múltiples transductores y aplicar técnicas de factorización matricial, como la descomposición en valores singulares (SVD), sobre las señales captadas [11]. Algunas características propias de las señales de habla, como la estructura armónica [12] y el índice de modulación [13], fueron explotadas para eliminar los efectos de la reverberación.

Posteriormente, se aplicó la idea de la sustracción espectral [14] [15] que básicamente consiste en la estimación del espectro de potencia generado por la reverberación a partir de modelos estadísticos. En 2006, Wang et. al. aplicaron este enfoque combinado con el de la estimación de filtros inversos logrando presentar avances importantes en la efectividad de los algoritmos [16].

El uso de máscaras binarias ideales en el dominio temporal-frecuencial para extraer las señales buscadas [17] es un enfoque muy utilizado, el cual tiene su origen en el campo del análisis computacional de escenas auditivas [18]. Las máscaras se definen como ideales ya que su obtención requieren del conocimiento de la señal buscada y de la señal que interfiere. El uso de estas máscaras implica primero realizar una transformación de la señal de entrada de manera de trasladarla al dominio tiempo-frecuencia (por ejemplo un espectrograma, o un cocleograma) y luego asignarle a cada punto del espacio temporal-frecuencial un valor de 1 cuando su energía mayormente pertenece a la señal objetivo, y un valor de 0 en el caso contrario. Roman et. al. [19] aplicaron este concepto para tratar el problema de dereverberación, donde se busca estimar la máscara binaria ideal tomando como señal objetivo la señal del habla en condiciones anecoicas y como interferencia a la parte reverberante. Para conseguir la dereverberación, este método requiere seleccionar de manera correcta parámetros como el punto desde el cual se distingue la parte temprana y tardía de la reverberación,y el nivel del umbral en base al cual se identifica a un punto específico como parte de la señal deseada o de la interferencia [20]. Hazrati et al. [21] propusieron estimar la máscara binaria a partir de un parámetro dependiente de la varianza de la señal, la cual define un umbral adaptativo, obteniendo mejores resultados.

A partir del año 2007, se comenzaron a aplicar redes neuronales en la tarea de dereverberación. Jin y Wang [22] utilizaron perceptrones multicapa para estimar las mascaras binarias necesarias para la separación de la componente reverberante en una señal voz. La red neuronal aprende a estimar máscaras binarias a partir de la representación tiempo-frecuencia de la señal reverberada. Mas adelante, con el avance de los modelos de aprendizaje profundo, esta técnica se iría perfeccionando reflejándose en mejores resultados en la tarea de dereverberación. Los enfoques basados en la estimación de máscaras tuvieron variantes como por ejemplo la estimación de máscaras ideales reales [23], mascaras ideales complejas [24] y mascaras sensibles a la fase [25].

En 2014 Kun et al. [26] proponen el uso de redes neuronales profundas para aprender el mapeo espectral de señales reverberantes hacia señales anecoicas. Esto quiere decir, en otras palabras, que se entrena una red neuronal profunda para que sea capaz de estimar el espectro anecoico a partir de la señal reverberada. Nuevamente se vuelve al planteo de la búsqueda del filtro inverso que permita la deconvolución de la señal reverberante para obtener su versión

anecoica, pero en este caso se utilizaran redes neuronales para estimar ese filtro.

Se han explorado una gran cantidad de arquitecturas y tipos de redes neuronales profundas. Las más utilizadas son las redes neuronales convolucionales (CNN), que surgieron del estudio de la corteza visual del cerebro y han sido muy exitosas en algunas tareas visuales complejas [27]. Las CNN en general trabajan sobre espectrogramas de magnitud y tienen una estructura de codificador-decodificador [28]. Estas son eficientes en términos de parámetros, aunque requieren de una gran cantidad de capas (o profundidad). Esto se debe a que cada capa convolucional modela su entrada de forma local, con un campo receptivo limitado, y es necesario colocar muchas capas en serie para ampliar ese campo receptivo y abarcar la totalidad del espectrograma de entrada.

Otro enfoque es utilizar redes neuronales que modelen el espectrograma de forma global, como es el caso de las redes recurrentes [29]. Estas permiten aprovechar el contexto temporal de una secuencia de espectros, por lo cual pueden extraer estructuras de corto y largo término, solucionando problemas de las arquitecturas convolucionales como la discontinuidad entre espectrogramas.

También se implementaron sistemas que combinan dos o más arquitecturas. Por ejemplo, la combinación de redes convolucionales y redes recurrentes [30]. De esta manera, la red convolucional permite analizar características locales en un contexto temporal fijo, y la red recurrente permite conservar información estructural de largo plazo. Otro ejemplo es la combinación de redes convolucionales y recurrentes con redes generativas adversarias (GAN) [25], lo cual produce una mejora en la calidad percibida del audio dereverberado generado.

Gran parte de los trabajos procesan la magnitud del espectrograma, ignorando la información de fase. En estos casos, al realizar la inversión del espectrograma estimado, algunos sistemas [26] utilizan el algoritmo de Griffin Lim [31], el cual permite invertir espectrogramas utilizando solamente su magnitud. Otros trabajos utilizan la fase original de la señal reverberada [32, 28], lo cual es una solución sencilla aunque subóptima. Por último, en trabajos recientes la información de fase es utilizada por las redes neuronales, ya sea porque trabajan con el espectrograma complejo [23], o porque modelan directamente la forma de onda [33].

CAPÍTULO 3: MARCO TEÓRICO

3.1 REPRESENTACIÓN TEMPORAL Y FRECUENCIAL DE SEÑALES

El sonido se genera cuando un disturbio que se propaga por un material elástico causa una alteración de la presión o un desplazamiento de las partículas del material que puedan ser reconocidos por una persona o por un instrumento [34]. En este trabajo manipularemos señales de audio digitales, por lo cual se realiza un muestreo periódico de la señal de audio analógica. La distancia temporal entre dos muestras contiguas se determina según la frecuencia máxima que se desea representar, acorde al teorema de Nyquist [35]. Entonces, una señal continua $x_c(t)$ que es muestreada a una frecuencia de f_s muestras por segundo, produce una señal discreta $x[n]$ como se expresa en la ecuación 1.

$$x[n] = x_c(nT_s) \qquad -\infty < n < \infty \tag{1}$$

En donde T_s es el período de muestreo, y es equivalente al inverso de la frecuencia de muestreo f_s. Partiendo de esta representación temporal discreta de la señal, se puede obtener una representación frecuencial (espectro) de la misma a partir de la Transformada Discreta de Fourier (DFT) [35]. Dada una señal discreta $x[n]$ con N muestras, su transformada discreta de Fourier se define como:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-jn2\pi k/N} \qquad k = 0, 1, 2, ..., N-1 \tag{2}$$

En donde X es la transformada de Fourier discreta de x, y cada muestra de $X[k]$ es el resultado del producto interno entre x y una exponencial compleja de frecuencia $2\pi k/N$. La DFT es invertible, por lo cual es posible recuperar la señal $x[n]$ a partir de $X[k]$ utilizando la transformada discreta de Fourier inversa (IDFT):

$$x[n] = 1/N \sum_{k=0}^{N-1} X[k] e^{jnk2\pi/N} \qquad 0 \leq n \leq N-1 \tag{3}$$

Existen algoritmos eficientes para el cálculo de la DFT denominados algoritmos de transformada rápida de Fourier (FFT) [36]. Estos consiguen realizar el cálculo de la DFT reduciendo

considerablemente la complejidad computacional (de O(N^2) a O$(Nlog(N))$).

3.1.1 Transformada de corto término de Fourier (STFT)

Las señales sonoras a analizar pueden ser no estacionarias. En este caso, la forma de onda de la señal nos brinda información del orden de aparición de los eventos sonoros, pero no sobre sus características frecuenciales. Por otro lado, la DFT nos permite tener información sobre la estructura frecuencial de la señal, pero resignando información sobre la evolución temporal. Entonces, para representar adecuadamente este tipo de señales tanto en tiempo como en frecuencia se utiliza la transformada de corto término de Fourier (STFT). La misma consiste en calcular la DFT sobre una ventana temporal que limita la cantidad de muestras a utilizar. Esta ventana se desplaza a lo largo de la señal, de manera tal que el resultado final pueda representar las caracteristicas frecuenciales y sus variaciones en el tiempo. Matemáticamente, la STFT se define como:

$$X[t,k] = \sum_{n=0}^{N-1} w[n]x[tH+n]e^{-j\frac{2\pi kn}{N}} \tag{4}$$

en donde:

- $X[t,k]$ es la transformada de corto término de Fourier de $x[n]$.
- t y k son los indices temporales y frecuenciales respectivamente.
- $w[n]$ es la ventana utilizada.
- N es el número de muestras de la ventana
- H es el número de muestras que se desplaza la ventana. Se lo denomina tamaño de salto o *hop size*, y determina el factor de solapamiento entre ventanas.

En la figura 1 se ilustra el proceso descrito para la obtención de la STFT.

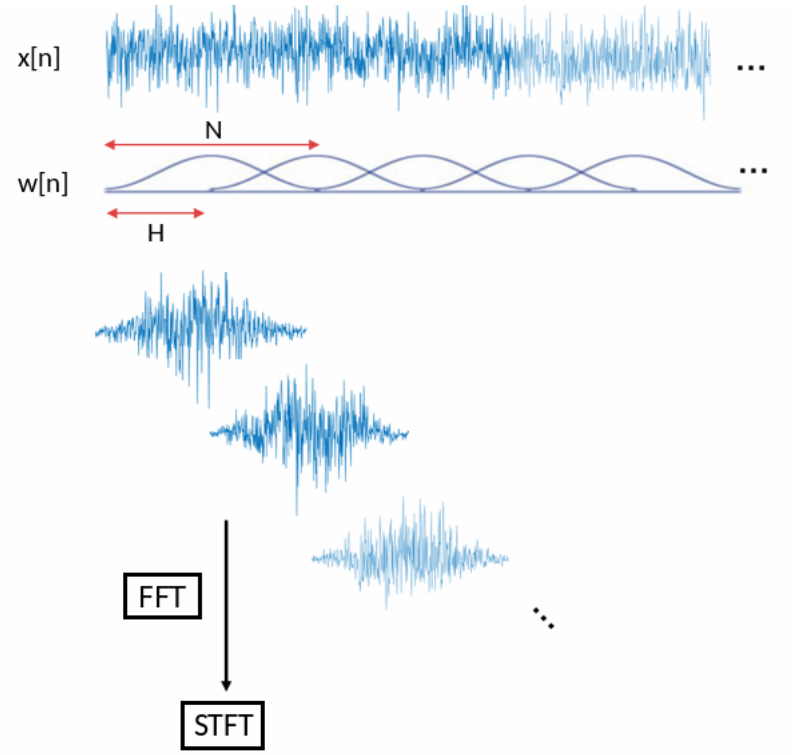


Figura 1: Proceso de obteción de la STFT. Extraido y adaptado de [37].

El resultado de la STFT depende tanto de la señal bajo análisis como del tipo de ventana utilizada. Esto se debe a que la señal y la ventana se multiplican en el dominio del tiempo, lo que equivale a una convolución en frecuencia. Es decir que el espectro frecuencial de la señal se verá distorsionado por el espectro frecuencial de la ventana utilizada. Sabiendo esto, las ventanas se diseñan para controlar la distorsión espectral que producen sobre la señal bajo análisis. Existen dos tipos principales de distorsión espectral: el manchado espectral y la fuga espectral. El manchado espectral refiere a una pérdida de resolución en frecuencia y esta relacionado al ancho del lóbulo principal del espectro de la ventana utilizada. Por otro lado, la fuga espectral consiste en la aparición de componentes frecuenciales que no corresponden a la señal bajo análisis y esta relacionada a la amplitud relativa del lóbulo principal con respecto a los lóbulos secundarios. En la figura **??** se muestran algunas ventanas con sus respectivas respuestas en frecuencia.

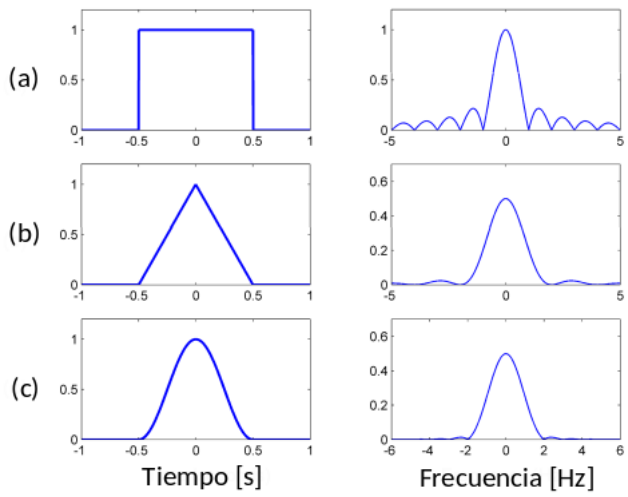


Figura 2: Ventanas (a) rectangular, (b) triangular y (c) Hann, con sus respectivas respuestas en frecuencia. Extraido de [38].

EL resultado de aplicar la STFT es una matriz de números complejos. Dicha matriz puede descomponerse en componentes de magnitud y fase. Comunmente, en tareas de procesamiento de audio, la información de fase se descarta y se trabaja únicamente con la magnitud. Esto se debe a que la magnitud presenta mayor información estructural que la fase. Tomando la magnitud de la STFT, las amplitudes suelen representarse en una escala de colores para producir una visualización como la que se muestra en la figura 3 a la que se denomina espectrograma.

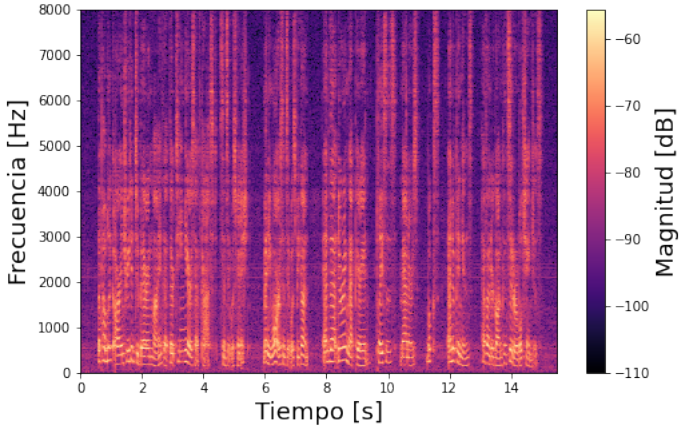


Figura 3: Espectrograma de una señal de audio.

La STFT es una transformación reversible. Para obtener una señal temporal partiendo de una STFT se aplica la técnica de solapamiento y suma (overlap-add). La misma consiste en calcular la

IDFT para cada t de la STFT y sumar las señales resultantes aplicando el mismo desplazamiento que se utilizó en el proceso de análisis. En terminos matemáticos, esta operación se define como:

$$x[n] = \sum_{t=0}^{L-1} Shift_{tH}[\frac{1}{N}\sum_{k=0}^{N-1} X[t,k]e^{j\frac{2\pi kn}{N}}] \tag{5}$$

En donde L es la cantidad de cuadros temporales presentes en la STFT. Para que la reconstrucción de la señal sea correcta, la ventana utilizada tiene que cumplir con el criterio de solapamiento y suma constante (COLA):

$$\sum_{t=0}^{L-1} w[n-tH] = \alpha \ \ \forall n \in \mathbb{Z} \tag{6}$$

donde α es una constante. Cuando $\alpha = 1$ la reconstrucción es perfecta. Para otros valores de α se deben aplicar compensaciones de amplitud sobre la señal reconstruida. La ventana de Hann cumple el criterio COLA siempre que la relación $\frac{N}{H}$ sea un número entero mayor a 1. En la figura 4 se muestran ejemplos de diferentes grados de solapamientos. Se puede ver que cuando no se cumple la relación anteriormente mencionada entre el largo de la ventana y el tamaño de salto, se produce una modulación en la amplitud de la señal recuperada.

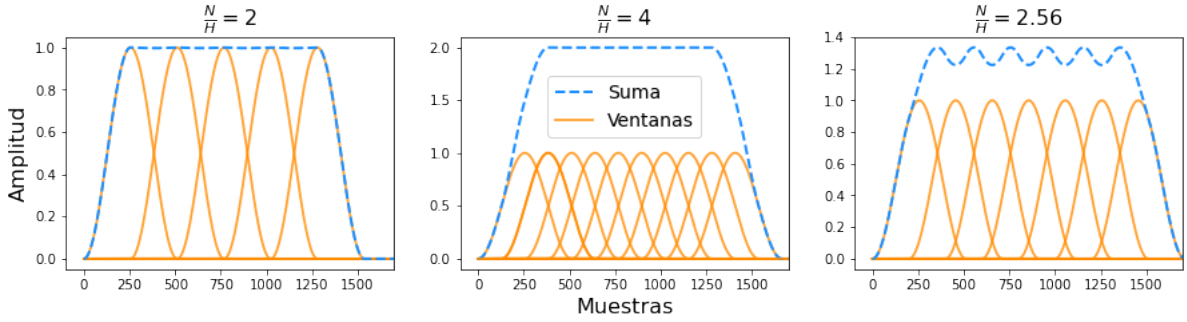


Figura 4: Efecto del solapamiento entre ventanas en la STFT.

3.2 RESPUESTA AL IMPULSO Y REVERBERACIÓN

Si en un recinto se tiene una fuente sonora y un micrófono captando a una cierta distancia, las ondas sonoras emitidas se reflejarán en las paredes del recinto y alcanzarán el micrófono inmediatamente después que la onda sonora directa. Las reflexiones continúan ocurriendo, y

cada instancia de reflexión supone una disminución de la energía de la onda, principalmente causada por el efecto de absorción acústica de las superficies que producen las reflexiones. En un determinado tiempo, la energía sonora decaerá en todo el recinto hasta ubicarse por debajo del ruido de fondo. A este proceso se lo denomina reverberación. Al camino mas corto entre la fuente y el punto de captura se denomina camino directo, y a la relación de nivel entre la presión sonora que genera la onda propia del camino directo y la presión que genera el efecto de reverberación se lo conoce como relación directo-reverberado.

Si el micrófono se ubica cerca de la fuente va a captar en mayor medida la señal correspondiente al camino directo, y una pequeña porción del sonido reverberado. Es decir, una relación directo-reverberado alta. A medida que el punto de captura se aleja de la fuente va a captar una menor cantidad del sonido correspondientemente al camino directo, mientras que el campo reverberado se mantendrá aproximadamente invariante. Esto se traduce en una disminución de la relación directo-reverberado. De esta manera, habrá una distancia específica para la cual el nivel de presión sonora generado por la fuente sera igual al nivel de presión sonora generado por el efecto de la reverberación. Esta distancia se conoce como distancia crítica. Esta depende tanto de las condiciones del recinto como de las características del micrófono.

Si pensamos a la fuente y el micrófono dentro del recinto como un sistema, es de interés estudiar su respuesta al impulso $h(n)$ para poder calcular una serie de parámetros que describan las características acústicas del recinto. Como su nombre lo indica, la respuesta al impulso equivale a la respuesta del sistema cuando se lo excita con un impulso infinitamente angosto (delta de Dirac). La respuesta al impulso sera diferente para cada par de puntos fuente-receptor dentro del recinto.

La figura 5 muestra una respuesta al impulso junto con un esquema temporal de la misma. En dicho esquema podemos identificar 3 partes: primero, el nivel de sonido directo (producido por la onda que viaja a través del camino directo), las reflexiones tempranas (cuyo limite temporal vendrá definido por las características propias de cada recinto) y por último la cola reverberante. Se puede distinguir la parte de reflexiones tempranas y la cola reverberante partiendo de la suposición de que las reflexiones tempranas ocurren en un proceso determinístico, siendo altamente sensibles a pequeños cambios en la geometría del recinto, mientras que la cola reverberante es mas bien un proceso estocástico, y al depender de un mayor número de

reflexiones no varía drásticamente frente a pequeños cambios de geometría.

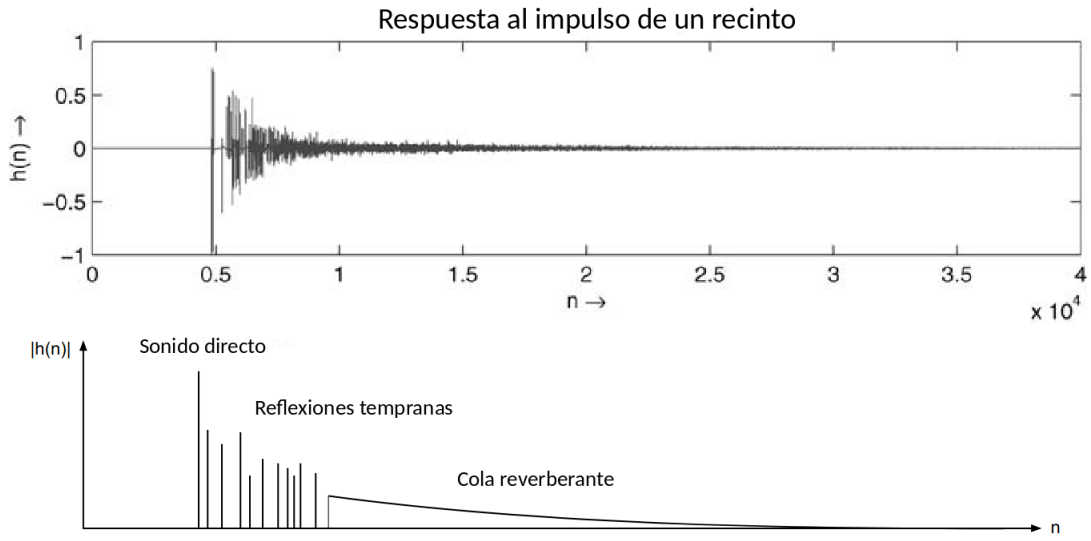


Figura 5: Secciones temporales de una respuesta al impulso. Extraido de [39].

Idealmente, el micrófono captura una señal que corresponde a la convolución entre la respuesta al impulso del recinto y la señal fuente, como se ve en la ecuación 7. Esto equivale a una multiplicación en el dominio de la frecuencia de acuerdo con la transformada de Fourier, como se ve en la ecuación 8.

$$x(t) = h(t) * s(t) \tag{7}$$

$$X(f) = H(f)S(f) \tag{8}$$

De esta manera se puede ver que la respuesta al impulso conserva toda la información sobre la influencia de la reverberación del recinto sobre la señal captada por el micrófono.

La respuesta al impulso de un recinto real se mide actualmente utilizando una técnica de barrido frecuencial [40]. Además, existen modelos geométricos que se utilizan para determinar la respuesta al impulso de manera analítica. Los principales son el modelo de trazado de rayos [41] y el modelo fuente imagen [42]. Estos modelos asumen la propagación del sonido como rayos.[en] lugar de ondas. En la figura 6 se ilustran ambos modelos. El trazado de rayos consiste en considerar un punto de fuente que emite radialmente. La longitud de los caminos de cada

rayo y los coeficientes de absorción acústica de las superficies del recinto se utilizan para determinar la respuesta al impulso del recinto. Por otro lado, el modelo de fuente imagen se basa en el principio de que una reflexión especular puede ser definida geométricamente espejando la fuente respecto del plano de reflexión. De esta forma se genera una imagen especular de la fuente por cada superficie de reflexión, y esto se aplica de manera recursiva. La suma de todas las fuentes imágenes con sus respectivos retardos y atenuaciones conforman la respuesta al impulso estimada del recinto.

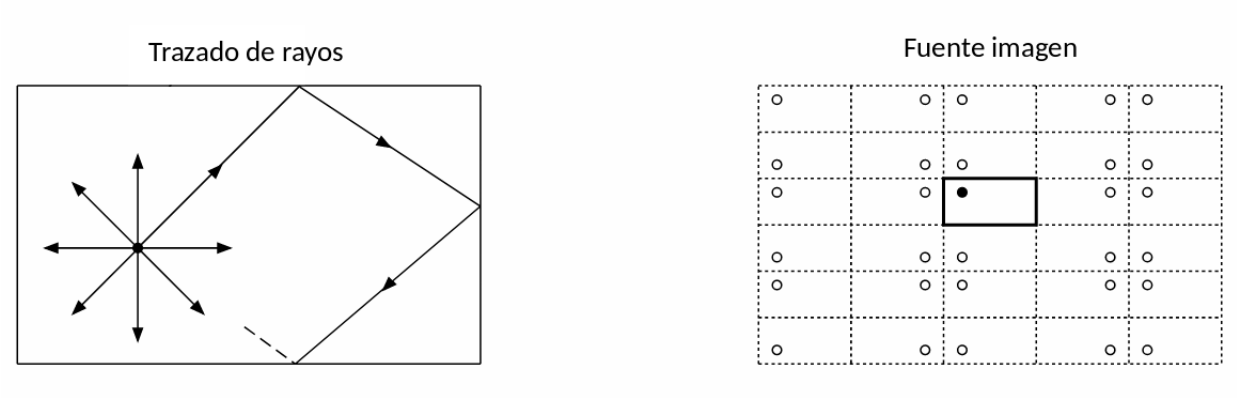


Figura 6: Modelos analíticos de cálculo de la respuesta al impulso de un recinto. Extraido de [39].

3.2.1 Relación directo-reverberado (DRR)

Es un descriptor acústico que se aplica sobre respuestas al impulso. Se define según la ecuación 9 en la cual $h(n)$ representa la respuesta al impulso discreta obtenida. Los índices desde cero hasta n_d representan las muestras correspondientes a la señal directa, y las muestras que continúan luego de n_d representan solo la reverberación producida por las reflexiones.

$$DRR[dB] = 10Log_{10}(\frac{\sum_{n=0}^{n_d} h^2(n)}{\sum_{n=n_d+1}^{\infty} h^2(n)}) \tag{9}$$

Este parámetro es dependiente de la distancia entre el punto emisor y receptor, y del tiempo de reverberación del recinto.

Como esta definición inicialmente se piensa en un dominio continuo, la primera intuición es pensar que el camino directo está fielmente representado por la mayor magnitud en la parte temprana de la respuesta al impulso. Sin embargo, esto solo es correcto cuando el tiempo de propagación entre la fuente y el receptor es un múltiplo entero del período de muestreo. Por

esto, trabajar con frecuencias de muestreo finitas (dominio discreto) en general deriva en que la representación del camino directo se produzca a través de una función seno cardinal ($Sinc$) correspondiente a la ventana de muestreo, centrada de acuerdo al retardo correspondiente al tiempo de propagación. En cambio, cuando se trata de respuestas al impulso sintéticas, el camino directo puede ser computado de forma separada del resto. Es decir, se puede determinar con exactitud el aporte del campo directo y del campo reverberado, lo que permite el cálculo del parámetro DRR con una mayor exactitud.

3.3 INTELIGIBILIDAD Y PARÁMETROS DE CALIDAD DE PERCEPCIÓN

Para caracterizar la señal del habla propagándose en condiciones reverberantes se utilizan métricas objetivas derivadas de la respuesta al impulso del recinto en cuestión, como por ejemplo el tiempo de reverberación o la relación energética entre la señal directa y el campo reverberado. En cambio, al considerar el proceso de dereverberación de estas señales las respuestas al impulso requieren ser estimadas, lo que usualmente conduce a una caracterización de baja calidad. Además, los algoritmos de dereverberación pueden introducir artefactos audibles a la señal voz, los cuales no son contemplados por las respuestas al impulso estimadas. Es por esto que es preciso utilizar métodos de medida de calidad basados en la señal dereverberada. Las pruebas subjetivas son el método más confiable para evaluar la calidad percibida de una señal de habla dereverberada. Sin embargo, este método es costoso y requiere mucho tiempo, por lo cual se vuelve inviable su aplicación para procesamientos en tiempo real. Para aplicaciones prácticas se definieron entonces métodos objetivos de medición de calidad basados en la señal dereverberada como reemplazo de las pruebas subjetivas. Estos métodos consisten en algoritmos que de manera objetiva y repetible buscan estimar la calidad percibida de la señal, por lo cual, un método resulta efectivo cuando logra obtener una alta correlación con las respuestas subjetivas. Estos métodos se clasifican en intrusivos o no intrusivos, dependiendo de si requieren o no una señal de referencia para realizar la estimación. Poder contar con una señal de referencia para realizar estas estimaciones es usualmente una dificultad, por lo cual se presta mayor interés en aquellos métodos no intrusivos.

3.3.1 Relación energía de modulación de voz a reverberación

Este parámetro de medida de calidad para señales dereverberadas se basa en obtener características de la reverberación partiendo del espectro de modulación de la señal [43]. La formulación de este parámetro se basa en el hecho de que la cola reverberante de una respuesta al impulso puede ser modelada como ruido blanco Gaussinano exponencialmente amortiguado. Esta característica puede ser explotada en el análisis del espectro de modulación de la señal bajo análisis para obtener descriptores del efecto de la reverberación.

3.3.2 Inteligibilidad objetiva de corto termino extendida

Este parámetro está basado en características extraídas a partir de la correlación de corto término entre la señal limpia y la señal procesada. Es aplicable para evaluar aquellos procesos que realizan transformaciones no lineales [44]. Su funcionamiento se basa en aplicar una ventana de análisis de $384\ ms$ en las envolventes de amplitud de las subbandas de la señal analizada. Estas ventanas temporales se aplican en pos de contemplar frecuencias de modulación que son relevantes para la inteligibilidad. En estos lapsos temporales se calculan coeficientes de correlación espectrales que son luego promediados. De esta manera, este parámetro puede ser interpretado en términos de una descomposición ortogonal de espectrogramas energéticamente normalizados que son luego ordenados de acuerdo a su contribución a la inteligibilidad estimada.

3.3.3 Relación señal a distorsión

Este descriptor fue ampliamente utilizado en tareas de separación de fuentes y refuerzo de señales de habla. Esta basado en el cómputo de la relación señal a interferencia (SIR), y en la relación señal a artefacto (SAR) [45]. En las tareas de dereverberación, estas medidas pueden ser interpretadas como proporcionales a la supresión de componentes reverberantes tardías e inversamente proporcionales a la distorsión en la señal del habla, respectivamente. Contemplando estos valores, el parámetro final mide la calidad general de la señal dereverberada.

3.4 REDES NEURONALES Y ALGORITMOS DE APRENDIZAJE PROFUNDO

Las redes neuronales artificiales son herramientas de modelado computacional que comparten algunas propiedades con el funcionamiento de las neuronas biológicas. Estas conforman el estado del arte actual para la resolución y modelado de problemas de alta complejidad en diversas disciplinas [46].

3.4.1 La neurona artificial

El bloque básico de los modelos de aprendizaje profundo es la neurona artificial. Esta consiste en un modelo que parte de los principios de funcionamiento de las neuronas biológicas [47]. EL esquema de una neurona artificial se puede ver en la Figura 7. Sus componentes principales son:

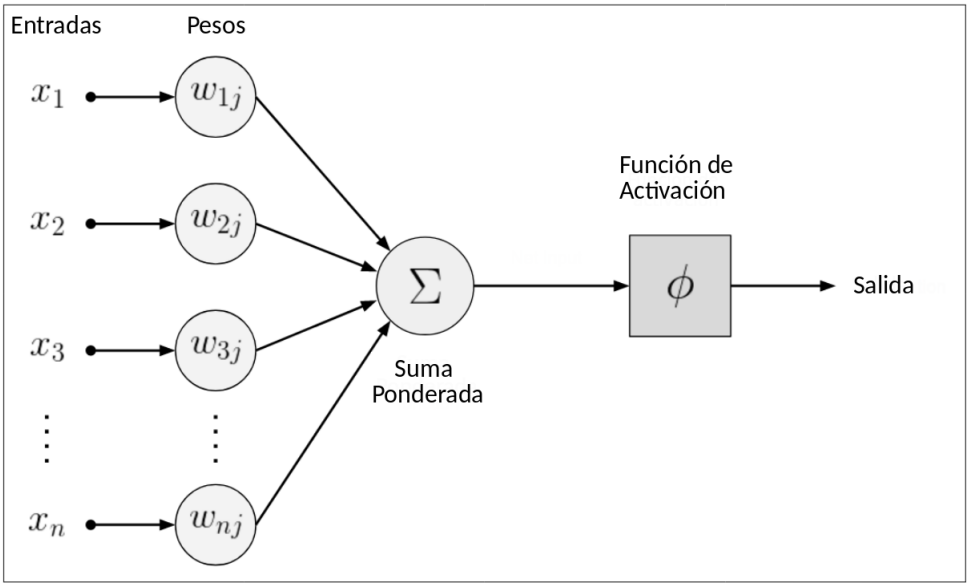


Figura 7: Esquema de neurona artificial. Extraído de [48].

- **Entradas**: son los datos que van a ser procesados en esta unidad.
- **Pesos sinápticos**:son parámetros asociados a cada entrada que se van ajustando durante la etapa de entrenamiento.
- **Suma ponderada**: Sintetiza la entrada a la función de activación. Consiste en el producto interno entre el vector de entradas y el vector de pesos sinápticos. Matemáticamente, se

expresa como:

$$\sum_{i=1}^{n} x_i w_i \tag{10}$$

donde x son las entradas, w los pesos sinápticos y n el número de entradas.

- **Función de activación**: Función que se aplica a la salida de la suma ponderada, y genera la salida de la neurona. La misma introduce alinealidades en la neurona, haciendo que el modelo resultante sea no lineal. Algunos ejemplos de funciones de activación se pueden ver en la figura 8.

- **Salida**: Es el resultado de aplicar el proceso completo al conjunto de entradas. En una red neuronal, esta salida puede ser la entrada de una o varias unidades subsiguientes.

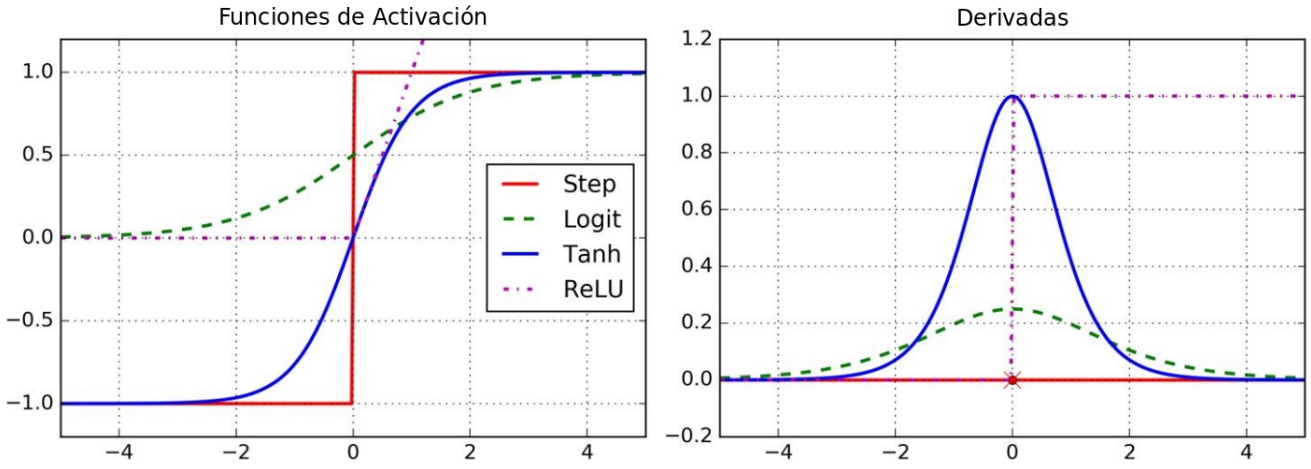


Figura 8: Funciones de activación y sus derivadas. Extraído de [27].

3.4.2 Modelos basados en redes neuronales

Una red neuronal suele orgaizarse en capas, las cuales poseen una cierta cantidad de neuronas. El comportamiento de la red neuronal se define en base a su arquitectura. La arquitectura depende principalmente de:

- Número de capas.

- Número de neuronas por capas.

- Tipos de conexiones entre las capas.

La red neuronal con alimentación hacia adelante fue el primer tipo de red neuronal implementado. En esta red neuronal, la información fluye en un solo sentido, desde la entrada hacia la salida.

Una de las redes neuronales mas estudiadas es la red neuronal multicapas con alimentación hacia adelante (feed-forward multilayer neural network). Esta consta de una capa de entrada, una o varias capas ocultas, y una capa de salida. Cada capa puede contener un número distinto de neuronas, y cada capa se encuentra completamente conectada a la capa adyacente. Esta red neuronal es capaz de representar cualquier función dado un número suficiente de neuronas. Un ejemplo de esta red se ilustra en la figura 9.

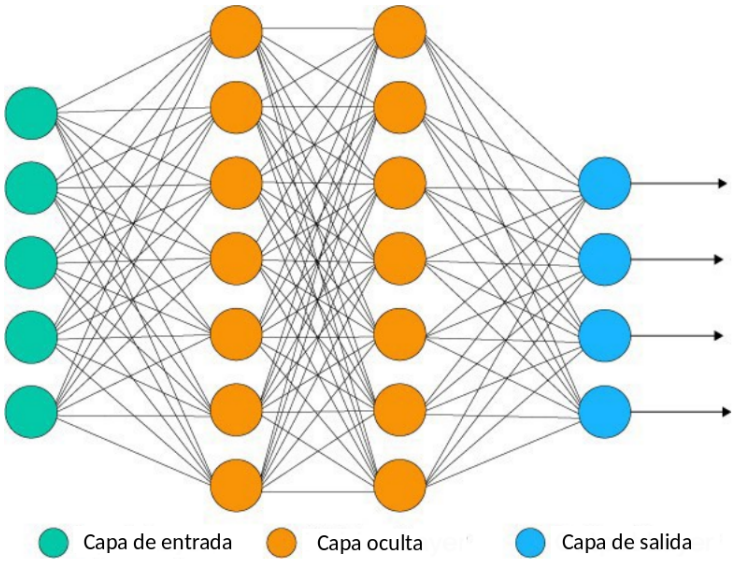


Figura 9: Ejemplo de red neuronal multicapa con alimentación hacia adelante.

Frecuentemente se cataloga a las redes neuronales multicapas como algoritmos de aprendizaje profundo. Esto se debe a que contienen múltiples capas de procesamiento no lineal que aprenden diferentes niveles de representación formando una jerarquía de características desde un nivel de abstracción mas bajo a uno mas alto [49].

3.4.3 Entrenamiento y aprendizaje

Los algoritmos de aprenedizaje por máquina se entrenan a partir del procesamiento de datos. Cuando se trata de un modelo de aprendizaje supervisado, los datos de entrenamiento se

presentan de a pares (x, y) en donde y es el valor objetivo o la salida que se espera obtener para el valor de entrada x. En el contexto del entrenamiento de una red neuronal, se define al aprendizaje como la búsqueda de una determinada configuración de los parámetros entrenables de la red que produzcan que la entrada x genere la salida y. En general, inicialmente las entradas van a generar salidas $\hat{y}$ aleatorias que difieren del valor objetivo y. Entonces, es necesario tener una medida de esta diferencia. De eso se encarga la función de costo (también denominada función objetivo o función de pérdida).

La función de costo recibe las salidas de la red y las salidas esperadas, y luego calcula una medida de error a partir de una función matemática. Esta función se escoge de acuerdo a la tarea que se busca realizar. Entonces, para cada estimación de la red, la función de costo otorga un puntaje que explica cuan lejos está el valor estimado del valor objetivo.

El paso siguiente en el proceso de entrenamiento es utilizar la salida de la función de costo como una señal de realimentación para poder ajustar los parámetros entrenables de la red neuronal (pesos sinápticos, umbrales, etc) de manera tal de minimizar la función de costo. Esta tarea es realizada por la función de optimización. La misma aplica el algoritmo de propagación del error hacia atrás para computar el gradiente de la función de costo respecto a los parámetros entrenables de la red neuronal. En base a este gradiente y al valor de tasa de aprendizaje definido en la función de optimización, se puede determinar como modificar los parámetros entrenables para lograr disminuir el error de salida. Este bucle de entrenamiento se ilustra en el esquema de la Figura 10. Repetir este ciclo un número suficiente de veces conduce a la convergencia del valor de error.

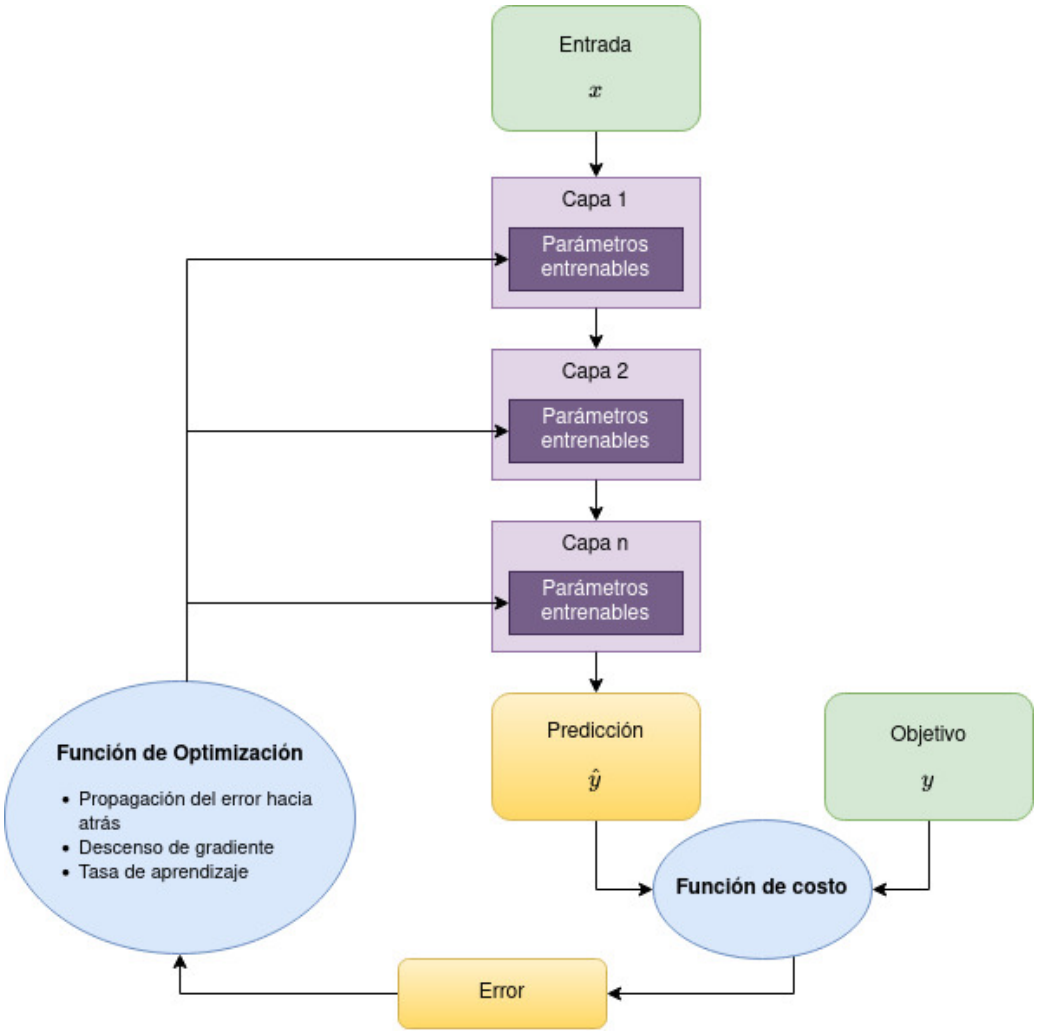


Figura 10: Diagrama de flujo del bucle de entrenamiento de una sistema de red neuronal

En este trabajo el conjunto de datos de entrenamiento se segmenta en lotes. En cada iteración de entrenamiento la red neuronal recibe un lote, lo procesa, aplica la función de costo y ajusta los parámetros de cada capa. Cuando la red procesó todos los lotes que componen el conjunto de datos de entrenamiento se dice que transcurrió una época. El proceso de entrenamiento depende en cierta medida del tamaño de los lotes [49]. Si consideramos la curva de la función de costo de un parámetro como la de la Figura 11, vemos que existen mínimos locales y mínimos globales a lo largo de la misma. En el proceso de entrenamiento se busca minimizar este valor de costo. Si se toman lotes muy pequeños, lo que se traduce en desplazamientos pequeños a lo largo de esta curva, se corre el riesgo de quedar confinado en un mínimo local. De igual manera, un conjunto demasiado grande produciría saltos demasiado grandes en compa-

ración a las fluctuaciones de esta curva, haciendo que se obtengan valores de costo aleatorios.

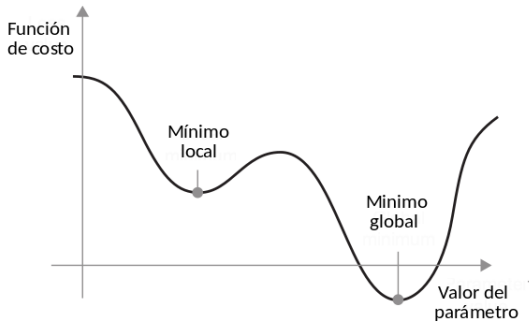


Figura 11: Curva de costo para un parámetro

El orden en el que los datos son presentados ante el modelo puede influenciar positiva o negativamente en el resultado final del entrenamiento. Existen técnicas de optimización que se desarrollaron partiendo de cualidades propias del aprendizaje en seres humanos y anaimales, como el hecho de que el aprendizaje resulta mejor cuando las instancias de aprendizaje estan organizadas en un orden significativo, añadiendo gradualmente mayor cantidad de conceptos y por ende una mayor complejidad. Esta idea se traslada al entrenamiento de algoritmos de aprendizaje profundo con el desarrollo de una estrategia denominada aprendizaje por currículum [50]. La misma consiste en seleccionar cuales datos y en que orden presentarlos al sistema durante el aprendizaje, de manera de guiar el entrenamiento para que inicialmente se aprendan los conceptos mas sencillos del problema, e ir gradualmente aumentando el grado de complejidad de la tarea. El aprendizaje por curriculum se define como una estrategia de optimización global. Dependiento de la tarea sobre la que se aplica, puede lograr en menor o mayor medida que un sistema logre un mejor nivel de generalización asi como también llegar al punto de convergencia en un menor tiempo de entrenamiento.

Por último, la segmentación de los datos que la red neuronal recibe y procesa en cada etapa del desarrollo también influye en el desempeño de la misma. El objetivo final del sistema es alcanzar un grado de generalización que le permita procesar adecuadamente instancias de datos que no hayan sido reveladas ante la red en la etapa de entrenamiento. Por esto, el conjunto total de los datos se divide en tres subgrupos:

- **Conjunto de entrenamiento**: Este conjunto de datos es el que se utiliza en la etapa de en-

trenamiento para optimizar los parámetros de la red. Aquí se concentra el mayor volumen de datos.

- **Conjunto de validación**: Sobre este conjunto se mide el desempeño del sistema a lo largo de su entrenamiento. Los resultados obtenidos del procesamiento de este conjunto sirven para ajustar variables que requieren ser especificadas de manera previa al entrenamiento. Estas variables se denominan hiper parámetros.

- **Conjunto de prueba**: Este conjunto es el que se utiliza para medir el rendimiento final del sistema. Como contiene instancias que no fueron utilizadas en las etapas de entrenamiento y ajuste de parámetros, el análisis del procesamiento de este conjunto sirve para medir el nivel de generalización que el sistema logró alcanzar.

3.4.4 Redes neuronales convolucionales

Las redes neuronales convolucionales emergen del estudio de la corteza visual del cerebro animal [**animales**]. En los últimos años, estas estructuras fueron utilizadas para resolver tareas visuales complejas (análisis de imágenes). El bloque básico de las redes neuronales convolucionales es la capa convolucional. La misma posee las siguientes características:

- **Campo receptivo limitado**: las neuronas que conforman la capa convolucional no están conectadas a todas las entradas, sino que solo se conectan con una porción de las mismas que se denomina campo receptivo. Esto hace que se modelen estructuras locales, es decir, presentes dentro de este campo receptivo.

- **Parámetros compartidos**: los pesos sinápticos de las neuronas se comparten. Por esto, los patrones aprendidos por las neuronas son invariantes a la traslación. Esto quiere decir que si se aprende de un patrón ubicado en un lugar específico del volumen de entrada, este mismo patrón puede luego ser identificado en cualquier otra ubicación de la entrada.

- **Convolución**: se aplican filtros de convolución sobre las entradas, que comúnmente son imágenes. Si bien la literatura utiliza el término convolución, matemáticamente se realiza una correlación cruzada. La figura 12 muestra en que consiste este proceso, y como el filtro se desplaza sobre las entradas para generar las salidas. En cada paso, el filtro se

multiplica por una sección de la imagen de entrada (producto interno) y el resultado corresponde a un solo valor en la imagen de salida. Las salidas de una capa convolucional se denominan mapas de características.

El funcionamiento a partir de campos receptivos y el hecho de que los parámetros se comparten entre neuronas producen que las capas convolucionales sean mas eficientes en términos de cantidad parámetros a entrenar.

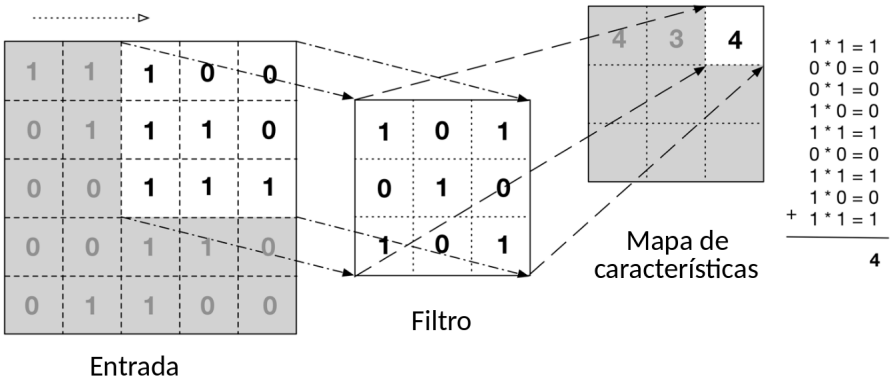


Figura 12: Funcionamiento de un filtro de convolución. Extraido de [48].

De esta manera, una capa convolucional aplica uno o varios filtros bidimensionales sobre la imagen de entrada, generando mapas de características que representan la presencia del patrón del filtro a lo largo de la imagen, como se puede apreciar en la Figura 13. En este tipo de capas, el aprendizaje se traduce en determinar la forma de los filtros que se deben aplicar para conseguir los resultados esperados.

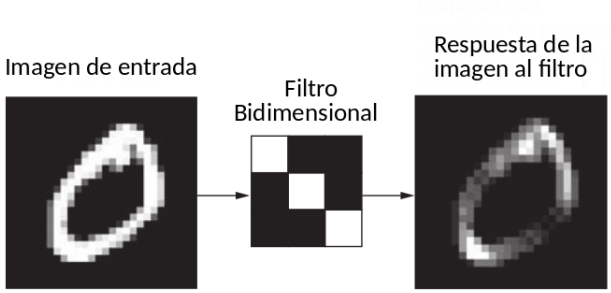


Figura 13: Representación de la aplicación de un filtro bidimensional sobre una imagen

Los hiperparámetros que se deben definir en cada capa convolucional son:

- **Tamaño del filtro**: Define el tamaño del campo receptivo de cada unidad de procesamiento de la capa. Valores comunes son $3x3$ o $5x5$. En la Figura 14 se ve un ejemplo de un filtro de tamaño $3x3$.

- **Tamaño del salto**: Determina la distancia horizontal y vertical entre campos receptivos de dos unidades contiguas. Hacer que este valor sea mayor a uno permite reducir las dimensiones de la imagen de entrada al atravesar la capa convolucional. Esto se puede apreciar en la Figura 14 en donde se aplica un tamaño de salto igual a dos (tanto en sentido vertical como horizontal).

- **Relleno de ceros**: Cuando se pretende mantener invariables las dimensiones de entrada y salida de una capa convolucional, se suele aplicar un relleno con ceros en los contornos de la imagen. La cantidad de ceros agregados dependerá de las características del filtro a aplicar. Aplicar un relleno de ceros produce un fenómeno denominado efecto de borde [27].

- **Cantidad de filtros aplicados**: El número de filtros convolucionales que se aplican sobre la entrada. Es equivalente al número de mapas de características que se generan.

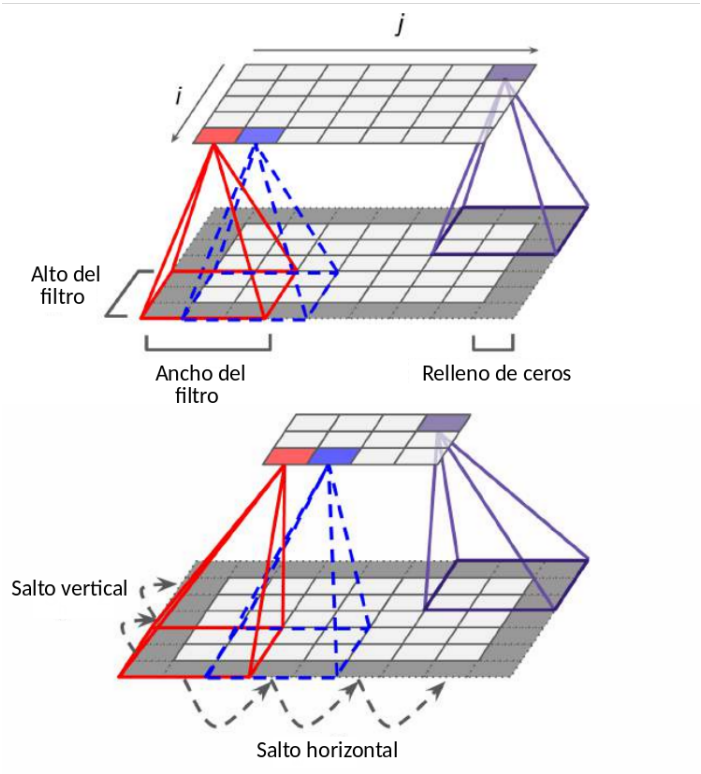


Figura 14: Parámetros de procesamiento en capas convolucionales. Extraido de [27].

Para el análisis de imágenes, estas capas se concatenan de manera que la primera capa no contempla cada píxel de la imagen, sino que solo se enfoca un número acotado de píxeles que caen dentro de su campo receptivo. De igual manera, las capas subsiguientes se enfocan en las salidas de un conjunto acotado de neuronas de la capa precedente. Este funcionamiento se ilustra en la Figura 15.

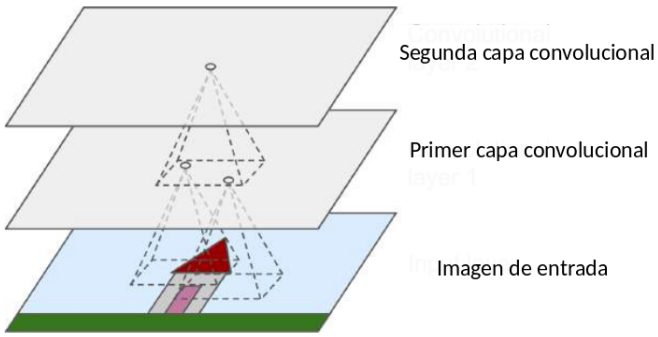


Figura 15: Capas convolucionales con campos receptivos locales rectangulares

Formar esta estructura le permite a la red aprender diferentes patrones estructurales locales de manera jerárquica [27]. En conclusión, a diferencia de las redes completamente conectadas,

las redes convolucionales logran la generalización de conceptos visuales complejos a partir de una menor cantidad de parámetros, distribuidos estratégicamente a lo largo de la arquitectura.

3.4.5 Arquitectura U-NET

3.5 DEREVERBERACIÓN POR FILTRADO TEMPORAL-FRECUENCIAL

3.5.1 Máscaras de amplitud

Existen numerosas maneras de representar las señales de audio para dereverberarlas. En la actualidad, los espectrogramas son la más utilizada ya que su cálculo es rápido y sencillo, son interpretables, es posible invertirlos y pueden ser facilmente aprovechados por modelos de aprendizaje profundo como las redes neuronales convolucionales. Partiendo de una señal con reverberación, se extrae una representación en tiempo-frecuencia a partir de transformaciones como la transformada de corto término de Fourier (STFT). Una vez obtenido este espectrograma, lo que se busca es descifrar el proceso necesario para obtener un nuevo espectrograma que se corresponda con la señal anecoica (descartando el efecto de la reverberación). Entonces, el proceso de dereverberación se puede resumir a la estimación de un filtro variable con el tiempo que se aplica sobre el espectrograma con reverberación. Estudios previos afirman que la fase no aporta información significativa para estas tareas de mejora del habla [51][52], por lo cual se suelen realizar estos procesos únicamente sobre la magnitud de los espectrogramas, utilizando la información de fase del audio original. Considerando esto, el proceso de dereverberación se reduce a la expresión de la ecuación 11, en donde $STFT_Y$ es la magnitud del espectrograma de la señal anecoica, $STFT_X$ es la magnitud del espectrograma de la señal con reverberación y M es la máscara ideal que representa el filtrado en el dominio tiempo-frecuencia. En otras palabras, la magnitud del espectro de la señal dereverberada se obtiene aplicando la máscara ideal sobre la magnitud del espectrograma de la señal con reverberación.

$$STFT_Y(t,f) = M(t,f)STFT_X(t,f) \tag{11}$$

Considerando que el espectro reverberado tendrá siempre mayor amplitud que el espectro anecóico, se trasladan los rangos de ambos espectros al intervalo $(0,1]$. De esta manera, las máscaras resultantes estarán dentro del mismo intervalo. Este proceso de escalado de las en-

tradas, resulta beneficioso para los modelos de redes neuronales artificiales [27]. Luego, la red neuronal tiene el objetivo de estimar la máscara ideal a partir del espectrograma reverberado. En la figura 16 se muestra un diagrama de cómo se estructura el sistema y las señales disponibles para lograr estimar la máscara de manera indirecta. El espectrograma con reverberación ingresa a la red neuronal artificial, la cual procesa esta entrada y produce una salida intermedia de las mismas dimensiones que la entrada. Luego, esta salida intermedia se multiplica con el espectrograma de entrada y el resultado es comparado con el espectrograma sin reverberación, mediante la función de costo. De esta manera, los pesos de la red neuronal se van a modificar en pos de que la salida del sistema sea lo más semejante posible al espectro sin reverberación. Cuando la función de costo tienda a cero, esto significará que la salida de la red neuronal tiende a igualarse con una máscara ideal, que al aplicarse sobre el espectrograma reverberado produce el espectrograma anecóico.

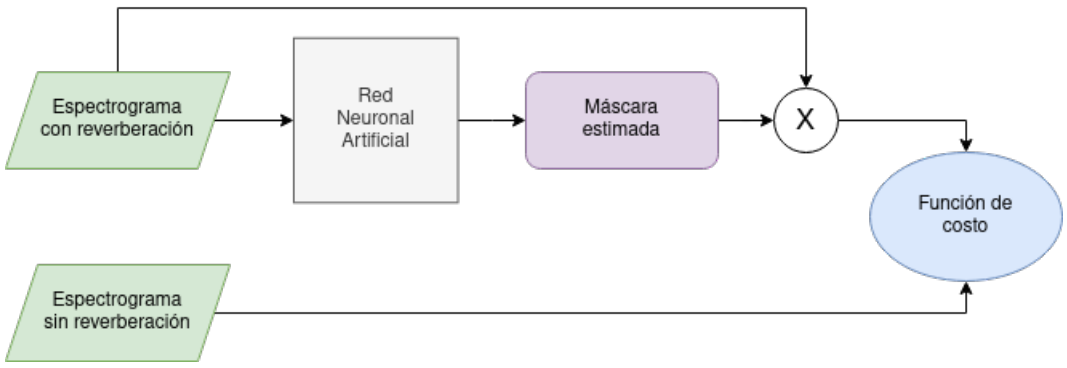


Figura 16: Disposición de las entradas y salidas del modelo para la estimación de las máscaras de amplitud

3.5.2 Síntesis de audio a partir de espectrogramas

En este trabajo, como en muchas otras tareas de procesamiento de audio (y procesamiento de señales en general), se llevan a cabo los siguientes pasos:

1. Tomar una señal en el dominio del tiempo $x[n]$ y convertirla en un espectrograma $X[t, f]$ a traves de la STFT.
2. Modificar el espectrograma $X[t, f]$ para obtener $\tilde{X}[t, f]$.
3. Convertir el espectrograma modificado $\tilde{X}[t, f]$ nuevamente a una señal en el dominio del tiempo $\tilde{x}[n]$ a través de la ISTFT.

El último paso de la lista conlleva un problema, ya que ciertos procesos pueden generar espectrogramas que no sean consistentes, es decir, que no haya ninguna señal en el dominio temporal cuyo espectrograma sea el generado. Para solucionar esta cuestión se desarrollaron algoritmos que buscan estimar una señal temporal cuyo espectrograma sea el más cercano posible al espectrograma que se quiere invertir. Este es el caso del algoritmo propuesto por Griffin y Lim [31]. El algoritmo consiste en un bucle iterativo que busca minimizar el error cuadrático medio entre el espectrograma de la señal estimada y el espectrograma modificado. En la figura 17 se muestra un diagrama de bloques que explica el funcionamiento básico del algoritmo.

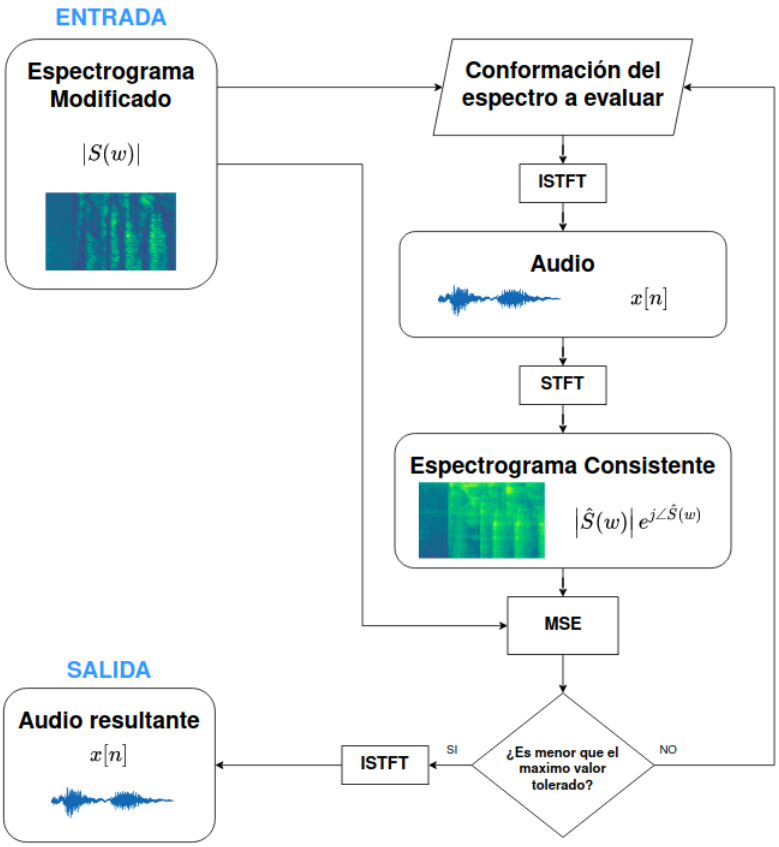


Figura 17: Diagrama en bloques del algoritmo de Griffin-Lim.

El espectrograma de entrada $|S(w)|$ inicialmente se combina con una fase aleatoria para formar un espectrograma complejo. Si se cuenta con una estimación previa de la fase, puede utilizarse en lugar de la fase aleatoria al iniciar el proceso. Luego, el espectrograma complejo conformado se antitransforma obteniendo una señal de audio, la cual vuelve a ser transformada obteniéndose un nuevo espectrograma complejo $|\hat{S}(w)|\, e^{j\angle\hat{S}(w)}$. Este espectrograma es

consistente, pues deriva de la transformación de una señal de audio real. Se puede probar que combinar esta fase resultante $\angle\hat{S}(w)$ con el espectrograma modificado de entrada $|S(w)|$ disminuye el error cuadrático entre los espectrogramas evaluados (es decir, entre el espectrograma consistente y el inconsistente). De esta manera, este proceso se repite hasta lograr que el error cuadrático medio descienda hasta un cierto valor deseado. Cuando esta condición se cumple, el espectrograma complejo que causa esta condición se antitransforma generando la señal de audio resultante final.

CAPÍTULO 4: METODOLOGÍA

4.1 GENERACIÓN DE DATOS

Para poder entrenar un algoritmo de aprendizaje profundo se requiere un conjunto de datos extensos. Este conjunto debe poder representar de la mejor manera posible el fenómeno que se quiere procesar. Además, se debe poder asegurar que todas las instancias que componen la base de datos tengan características homogéneas que se adecuen a los procesos subsiguientes.

Para este trabajo, es necesario partir de un conjunto de datos conformados por dos tipos de elementos principales: Respuestas al impulso, y grabaciones de voz. Con estos elementos, es posible generar instancias que comprendan información de audio con reverberación y su correspondiente versión anecoica, siendo esta última la que un sistema de dereverberación tiene como objetivo.

Además, se debe tener en cuenta que se busca formar tres grandes conjuntos de datos: conjunto de entrenamiento, conjunto de validación y conjunto de prueba. Las características de estos conjuntos deberán variar de acuerdo al propósito de cada uno para lograr optimizar cada etapa, o bien, para evaluar ciertos aspectos de estos procesos.

4.2 BASES DE DATOS DE RESPUESTAS AL IMPULSO

Para este trabajo se utilizaron respuestas al impulso reales y simuladas. A partir de este punto, se utilizarán los términos respuestas simuladas y respuestas generadas de manera indistinta. A su vez, también se trabaja con un tercer conjunto formado a partir de la aumentación de respuestas al impulso reales. Esto es, partiendo de un subconjunto de respuestas al impulso reales, se alteran estas señales de manera controlada para producir nuevas respuestas al impulso con diferentes características acústicas.

4.2.1 Respuestas al impulso reales

Las respuestas al impulso reales se obtienen del conjunto de datos C4DM [53]. Este conjunto consiste en una colección de respuestas al impulso que fueron medidas en tres recintos: una sala multipropósito con aproximadamente 800 asientos, un edificio victoriano construido en

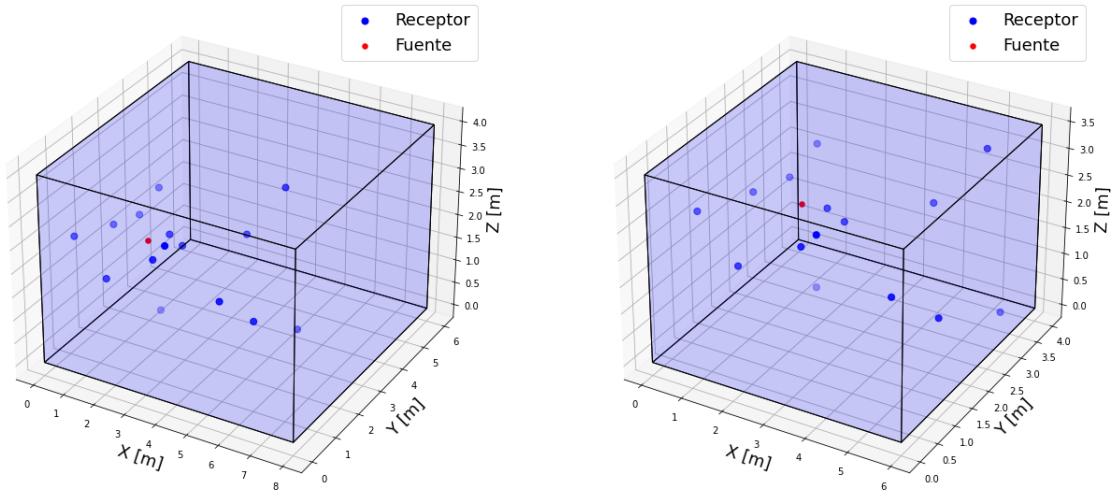
1988 originalmente diseñado para ser una biblioteca, y una sala de clases de una universidad. Las mediciones fueron realizadas utilizando la técnica del barrido frecuencial [40]. Para todas estas respuestas al impulso, el tiempo de reverberación es de aproximadamente 2 segundos.

4.2.2 Respuestas al impulso generadas

En cuanto a las respuestas al impulso generadas, se utilizó la librería de Python 'PyRoomAcoustics' [54] para sintetizarlas. Esta librería brinda un software de generación de respuestas al impulso basado en el método de fuente imagen [55]. El algoritmo está implementado en el lenguaje de programación C, permitiendo una rápida simulación de la propagación del sonido en recintos poliédricos. Los parámetros que se deben indicar a la hora de generar una respuesta al impulso son:

- Dimensiones del recinto (largo, ancho y alto).
- Posiciones de fuente y receptor, en coordenadas tridimensionales.
- Coeficientes de absorción de las superficies.
- Orden máximo de reflexiones a computar.

Para generar los datos se proponen dos recintos, el primero de dimensiones $8m\,x\,6m\,x\,4m$ que se denominará 'Recinto 1' , el segundo de dimensiones $6m\,x\,4m\,x\,3{,}5m$ que se denominará 'Recinto 2'. En la figura 18 se pueden visualizar ambos recintos generados.



(a) Recinto 1 *(b) Recinto 2*

Figura 18: Recintos y puntos receptor-fuente generados para la simulación de respuestas al impulso

Para controlar los demás parámetros que refieren a las condiciones del recinto, se subordina el orden máximo de reflexiones y los coeficientes de absorción a un tiempo de reverberación esperado. Esto es, teniendo un cierto recinto se determina un valor de tiempo de reverberación T60 inicial. Este se utiliza para estimar un valor de un coeficiente de absorción promedio mediante la ecuación de Sabine y también en base a este tiempo se determina el orden de reflexiones necesario para poder representar la reverberación. Por último, las posiciones de fuente y receptor se generan aleatoriamente para poder generar diferentes respuestas al impulso a partir de un mismo recinto. De esta manera, los datos que se deben determinar son las dimensiones del recinto, un tiempo de reverberación inicial y la cantidad de respuestas al impulso que se busca generar.

Con esto, para formar el conjunto de respuestas al impulso generadas de entrenamiento se utilizó el recinto 1 para 3 tiempos de reverberación principales: $0,5s$ como reverberación baja, $0,75s$ como reverberación media y $1,0s$ como reverberación alta. Partiendo de estos tiempos, se generan 30 respuestas al impulso para cada uno, variando aleatoriamente los puntos de fuente y receptor. Esto resulta en un total de 90 respuestas al impulso con tiempos de reverberación de entre aproximadamente $0{,}5$ segundos a $1{,}0$ segundos. Para generar las respuestas destinadas

a evaluación se realiza el mismo procedimiento pero utilizando el recinto 2 y generando 15 respuestas por cada tiempo de reverberación, formando un total de 45 respuestas al impulso generadas.

4.2.3 Respuestas al impulso generadas por aumentación

Este conjunto se genera partiendo de un subconjunto de respuestas al impulso reales. El propósito de este proceso es partir de un conjunto de impulsos escasos con determinadas características, y generar un conjunto mucho más grande de respuestas al impulso controlando de manera paramétrica ciertos descriptores acústicos como el tiempo de reverberación $T60$ y la relación directo-reverberado DRR, de manera tal que se pueda asegurar un cierto balance en el conjunto conformado [56]. El proceso de aumentación entonces se divide en dos procesos principales: una alteración de amplitud en la parte temprana de la respuesta al impulso para controlar la relación directo-reverberado, y una alteración de envolvente de caída para controlar el tiempo de reverberación.

Para el primer proceso, a la parte temprana de la respuesta al impulso $h_e(t)$ se le aplica una ganancia definida por un factor α el cual se calcula para obtener el valor de DRR deseado generando una nueva señal $\tilde{h}_e(t)$. Para evitar generar discontinuidades durante el proceso, se aplican ventanas complementarias a la parte temprana obteniendo una parte temprana ventaneada y un residuo ventaneado. A partir de esto, la parte temprana se puede definir según la ecuación 12.

$$h_e(t) = \alpha w_d(t) h_e(t) + [1 - w_d(t)] h_e(t) \tag{12}$$

En donde $w_d(t)$ corresponde a una ventana Hann de $5ms$ de longitud. De esta manera, partiendo de esta última definición junto con la expresión del parámetro DRR expresado en la ecuación 9 se plantea un sistema de ecuaciones a partir del cual se puede definir un valor pretendido de DRR y despejar el correspondiente valor de α. En la figura 19 se puede observar una representación de una parte temprana $h_e(t)$, las ventanas aplicadas, el efecto del factor de ganancia α y la nueva señal $\tilde{h}_e(t)$ generada. Finalmente, esta parte temprana modificada se concatena con el resto de la respuesta al impulso completando así el proceso de aumentación

referido a la relación directo-reverberado. Se debe tener en cuenta que para tiempos de reveberación cortos, puede ser un problema generar relaciones directo-reverberado demasiado bajas, ya que la energía de la parte tardía ya es de por si muy baja. Para esos casos, se definen valores límites para la ganancia aplicada a la parte temprana.

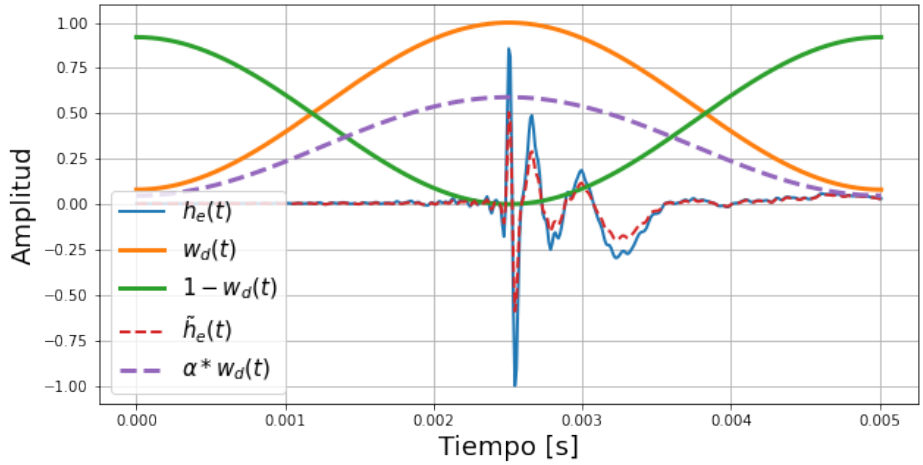


Figura 19: Señales involucradas en el proceso de aumentación de DRR

Luego, para la modificación del tiempo de reverberación T_{60} se trabaja únicamente con la parte tardía de la respuesta al impulso. Esta puede ser modelada como ruido Gaussiano con una caída de nivel exponencial dependiente de la frecuencia, sumado a un determinado piso de ruido. Como esta pendiente de caída varía con la frecuencia, se analiza la respuesta al impulso en bandas de tercio de octava para contemplar esta dependencia frecuencial. Este modelo se expresa en la ecuación 13.

$$h_m(t) = A_m e^{\frac{-(t-t_0)}{\tau_m}} n(t)u(t-t_0) + \sigma_m n(t) \tag{13}$$

En donde A_m es la amplitud inicial, τ_m es la tasa de caída, σ_m es el nivel del piso de ruido, $n(t)$ es ruido Gaussiano de media cero y desvio estándar unitario, t_0 es el instante temporal en donde comienza la parte tardía de la respuesta al impulso, m es el índice que refiere a una sub-banda frecuencial y $u(t)$ es un escalón unitario. En este modelo, el tiempo de reverberación T_{60} se relaciona directamente con el parámetro τ según la ecuación 14.

$$T_{60} = ln(1000)\tau T_s \tag{14}$$

En donde T_s es el período de muestreo. Dado este modelo, se aplican métodos de optimización no lineales para estimar el conjunto de parámetros $\left\{\hat{A}_m; \hat{\tau}_m; \hat{\sigma}_m\right\}$ que mejor aproximen la envolvente de caída de la respuesta al impulso. Con estos parámetros sumados a la tasa de caída deseada $\tau_{m,d}$ calculada a partir del tiempo de reverberación deseado, se modifica la parte tardía de la respuesta al impulso inicial multiplicandola por una envolvente exponencial creciente o decreciente según corresponda como se muestra en la ecuación 15.

$$h_m{}'(t) = h_m(t)e^{-(t-t_0)\frac{\hat{\tau_m}-\tau_{m,d}}{\hat{\tau_m}\tau_{m,d}}} \tag{15}$$

En donde $h_m{}'(t)$ representa a la nueva parte tardía de la respuesta al impulso generada para obtener el tiempo de reverberación deseado. En términos generales, el proceso consiste en modificar la pendiente de caída para obtener la pendiente de caída deseada por cada banda de frecuencia. Al final, las sub-bandas generadas se suman para obtener el resultado final que contemple todo el espectro de la señal. Hasta aquí este proceso funciona satisfactoriamente cuando se generan tiempos de reverberación menores al de la respuesta al impulso inicial, es decir, siempre que se multiplica la respuesta al impulso por envolventes exponenciales decrecientes. En cambio, cuando se busca generar tiempos de reverberación mayores la envolvente por la que se multiplica la respuesta inicial es creciente, lo que produce una amplificación de la parte tardía de la respuesta al impulso. Esto muchas veces equivale a amplificar el piso de ruido presente en la señal, lo que puede producir pendientes de caída inestables que no se corresponden con el comportamiento propio de la respuesta al impulso ya que no es información del sistema acústico sino simplemente ruido. Para evitar este efecto adverso del proceso de aumentación anteriormente propuesto se debe estimar el piso de ruido de la respuesta al impulso. Esto se realiza a través del método iterativo propuesto por Lundeby et. al. [57]. Una vez estimado el piso de ruido de la señal, la respuesta final se obtiene haciendo un cross-fade en el inicio del piso de ruido entre la parte tardía generada y una cola reverberante sintética creada a partir de multiplicar ruido Gaussiano con una envolvente exponencial decreciente, utilizando los parámetros previamente calculados. Una explicación más detallada de este proceso se

puede encontrar en el anexo A.

Para realizar este proceso se determinan límites de relación directo-reverberado y tiempo de reverberación medio. La relación directo-reverberado va desde $-3\ dB$ a $10\ dB$ con saltos de $1\ dB$, lo que se considera una diferencia de nivel promedio acorde a la mínima perceptible por el oido humano. Con respecto al tiempo de reverberación, se generan desde $0{,}1\ s$ a $1{,}2\ s$ para estar dentro del rango de las respuestas al impulso generadas, con un paso de $0{,}05s$ basado en estudios previos realizados sobre la minima diferencia perceptible entre tiempos de reverberación [58]. Una vez obtenido el conjunto de respuestas al impulso aumentadas, se seleccionan aleatoriamente 135 respuestas para equiparar al número de respuestas al impulso generadas.

4.3 BASES DE DATOS DE SEÑALES DE HABLA

Las señales del habla necesarias para formar los pares anecoico-reverberados se obtienen de la librería LibriSpeech [59] la cual consiste en un conjunto de datos que reúne 100 horas de audio correspondientes a lecturas en idioma inglés. Los datos corresponden a programas tipo audiolibros. Las señales poseen bajo nivel de reverberación, y provienen de una aplicación en la cual la inteligibilidad es primordial, lo cual hace que esta base de datos sea adecuada para utilizarse en este trabajo.

4.3.1 Pre-procesamiento de datos

Partiendo de audios de voz y respuestas al impulso, el modelo de red neuronal propuesto requiere generar instancias de espectrogramas de magnitud y máscaras ideales para poder entrenarse. Para conseguir esto, se programa una cadena de procesamiento automatizada que realice esta transformación de los datos de entrada. En primer lugar se controla la uniformidad de frecuencias de muestreo aplicando las transformaciones de aumentación o decimado cuando sean requeridas. Se decide trabajar con una frecuencia de muestreo de 16000 muestras por segundo, considerando que se tratan con señales de voz que concentran su información por debajo del límite de representación frecuencial de $8000Hz$ impuesto por esta decisión. Luego, los audios de voz se convolucionan con las respuestas al impulso para formar pares de señales con y sin reverberación. El resultado de la convolución se recorta para descartar el retardo generado por la convolución, haciendo que los pares de señales sean sincrónicas. Luego, se to-

man ventanas rectangulares de 32640 muestras, lo que equivale a segmentos de audio de $2,04$ segundos para la frecuencia de muestreo utilizada. Lo siguiente es aplicar la transformada de corto término de Fourier tanto a la señal limpia como a la señal convolucionada. La transformada se aplica con una ventana de 512 muestras y un salto de 128 muestras lo cual equivale a un solapamiento del $75\,\%$. Esto permite la correcta reconstrucción de la señal al antitrasformar. Se obtienen espectrogramas complejos, a los cuales se les calcula la magnitud, descartando la información de fase. Además, se aplica una normalización para acotar el dominio en valores que sean convenientes para el algoritmo de aprendizaje posterior.

Finalmente, las instancias finales de este proceso son en el espectro de magnitud de la señal con reverberación (que corresponde a la variable de entrada de la red neuronal) y el espectro de magnitud de la señal sin reverberación (que corresponde a lo que se compara con la salida de la red en la función de costo, es decir, el objetivo que el modelo busca estimar). Ambas instancias tienen las mismas dimensiones, que corresponden a 256 cuadros temporales y 257 valores posibles de frecuencia (se conserva solo la parte positiva del espectro frecuencial simétrico). Por último, se descartan los puntos correspondientes al valor máximo de frecuencia. Esto se realiza para obtener dimensiones finales de $256x256$ lo cual facilita el proceso de compresión y expansión de los espectros al ser dimensiones múltiplos de 2. Se descarta la frecuencia más alta ya que por la característica de la fuente no contendrá información crucial para la representación.

Cabe destacar que debido a este preprocesamiento aplicado, a la hora de evaluar el modelo se deberán aplicar una serie de procesos previos sobre el audio a procesar. Más precisamente, se deberá segmentar el audio y obtener espectros de magnitud de la STFT respetando los mismos parámetros que en el preprocesamiento. Luego, como la salida de la red es una máscara de amplitud comprimida, se debe descomprimir esta máscara, aplicarla sobre el espectro reverberado y luego combinar el espectro de amplitud modificado resultante con la fase original de la señal para poder finalmente obtener la información de audio de salida (ya sea a través de la aplicación del algoritmo de Griffin-Lim o bien simplemente combinando la fase reverberante con la magnitud dereverberada).

4.4 MODELO PROPUESTO

El modelo propuesto se basa en una arquitectura de red neuronal completamente convolucional tipo 'autoencoder' inspirada en el trabajo de Ernst et. al.[28]. Más precisamente, un autoencoder es una estructura que tiene como objetivo aprender niveles de representación de la información de entrada, para luego poder reconstruir una instancia similar descartando la información no deseada o considerada ruido". En este caso, la señal no deseada corresponde a la reverberación. El esquema básico del algoritmo se puede observar en la figura 20, donde la variable x representa a las variables de entrada, e y representa la variable de salida.

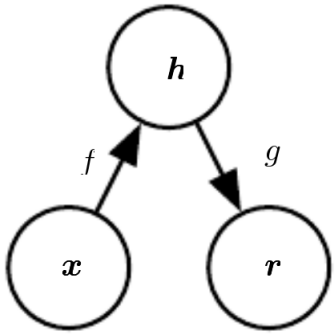


Figura 20: Estructura general de un autoencoder.

El esquema se compone de tres partes fundamentales:

- Una función de codificación f en donde las dimensiones de la variable de entrada se comprimen y las características más relevantes son aprendidas. Esta función realiza el mapeo de la variable de entrada al espacio latente.

- Un espacio latente h (o espacio de representación), en donde se concentran las representaciones internas aprendidas a partir de la compresión de la variable de entrada.

- Una función de decodificación en donde se aplica el proceso inverso que en la codificación, expandiendo las dimensiones tomadas del espacio latente para formar una representación que minimice el error de reconstrucción.

El sistema propuesto consiste en la estimación del espectro de la señal anecóica a partir del espectro de la señal reverberada. Para conseguirlo, en lugar de hacer un mapeo directo entre

ambos espectros, se opta por estimar una máscara de amplitud. Se decidió trabajar con máscaras ya que estudios previos demostraron que con este método se obtienen mejores resultados que realizando estimaciones de mapeos directos entre dos espectros [60]. Para trabajar con espectros, las señales de entradas se transforman al dominio temporal-frecuencial a partir de la transformada de Fourier de corto término. Se utilizó una ventana temporal de 512 muestras, con un solapamiento del 75 %.

La estructura de red neuronal utilizada consiste en una U-NET con conexiones de saltos, inspirada inicialmente en [28]. Este tipo de estructuras consiste en tomar mapas bidimensionales de entrada y a partir de la aplicación sucesiva de capas convolucionales con valores de salto mayor a 1, reducir la dimensionalidad del mismo e ir aumentando el número de filtros utilizados por la capa convolutiva. Un esquema básico de está estructura se puede ver en la figura 21, en donde se puede ver que las dimensiones de las capas siguen una forma de 'U', lo cual le da el nombre a estas estructuras.

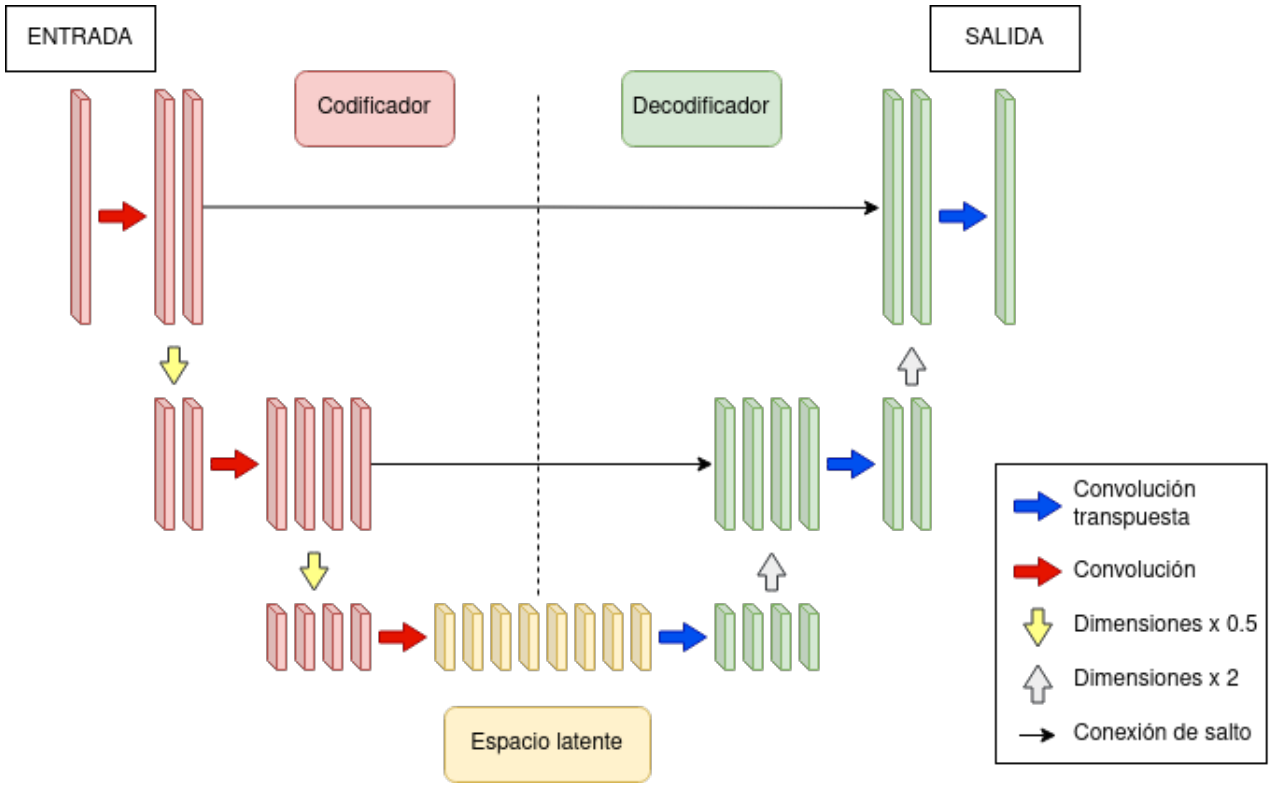


Figura 21: Esquema básico de una red tipo 'U-NET' con conexiones de salto.

A medida que se avanza en el modelo, el tamaño del tensor bidimensional de entrada va disminuyendo y la cantidad de filtros utilizados va aumentando. Esta primera parte se puede

pensar como un codificador, ya que el sistema está tomando información de manera jerárquica. Este proceso se realiza sucesivamente hasta que el tensor alcanza una dimensión de 1x1. Luego, prosigue una etapa de decodificación en la cual se aplica el proceso inverso. Esto es, la dimensión del tensor se va aumentando con capas de convolución transpuesta configuradas con tamaños de saltos mayores a 1, y la cantidad de filtros utilizados va disminuyendo. Esto se repite hasta que las dimensiones del tensor sean las mismas que tenía a la entrada del codificador. Mediante este esquema de U-NET y el efecto del cuello de botella de las dimensiones, se consigue que la estimación de cada punto del tensor resultante esté condicionado por todos los puntos que componen el tensor de entrada. Se puede decir entonces que la estimación de cada punto del espectro final depende de todo el espectro de entrada, y no solo de una región determinada. Para poder pasar información de manera más directa desde el decodificador hacia el codificador, se implementan conexiones de salto. La conexión de salto consiste en concatenar la salida de una capa del codificador con una capa del decodificador. Para poder hacerlo, la dimensión de concatenación (en este caso, las dimensiones del espectrograma) deben ser las mismas. De esta manera se logran decodificaciones más precisas, solucionando problemas como el desvanecimiento de gradiente [27].

Una representación gráfica del modelo final implementado se puede apreciar en la figura 22. En cada capa se indican tres valores, donde el primero representa la dimensión temporal, el segundo la dimensión frecuencial y el tercero el número de canales (equivalente a la cantidad de filtros). En las primeras capas, las dimensiones se reducen a la mitad en cada instancia debido al uso de un desplazamiento de paso 2 en el cálculo de los filtros convolucionales, lo que realiza la compresión de la información. En las capas subsiguientes, las dimensiones sufren el efecto contrario hasta volver a obtener las dimensiones originales. Este tipo de estructura tiende a perder información importante de bajo nivel durante el proceso de compresión. Como por lo general las variables de entrada y de salida comparten información estructural, se puede mejorar el funcionamiento de estas estructuras implementando conexiones entre las capas del codificador y el decodificador. Esto quiere decir que los mapas de características de las capas que conforman el codificador se van a concatenar directamente con los mapas de características en el decodificador, es decir, la salida de la capa i se concatena con la salida de la capa $N - i$ donde N es el número de capas. Estos saltos evitan que las activaciones pasen por el cuello

de botella permitiendo la propagación de esta información estructural que estas arquitecturas tienden a perder.

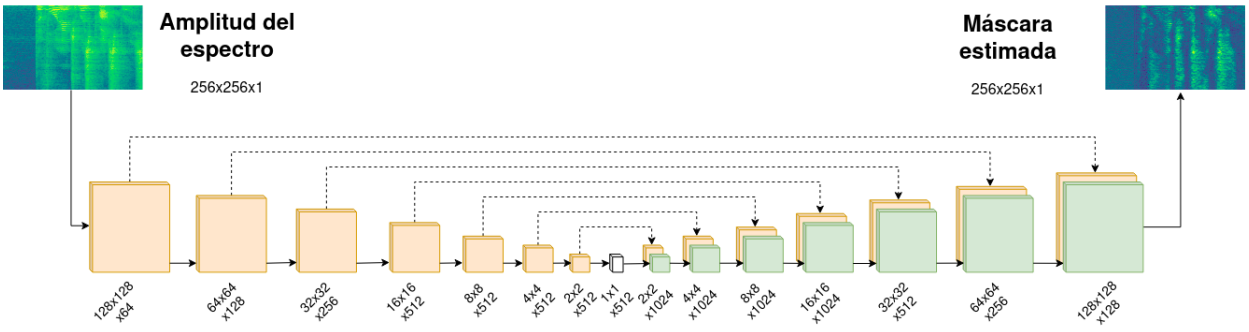


Figura 22: Modelo de red neuronal convolucional implementado

De esta forma, el modelo realiza una compresión de los datos de entrada, buscando minimizar el error en la reconstrucción al comparar con las instancias que se le presentan como objetivo. Para este caso particular, las entradas del modelo son espectrogramas, es decir, valores en el dominio tiempo-frecuencia. Las dimensiones del espectrograma de entrada son comprimidas hasta llegar a dimensiones de $1x1$ y luego son expandidas nuevamente, lo que produce un aumento de los campos perceptivos que permite propagar información global tanto en tiempo como en frecuencia. Esto significa que el cómputo de cada punto de salida se verá influenciado por la totalidad del espectrograma de entrada.

4.5 ESPECIFICACIONES DE LA ARQUITECTURA IMPLEMENTADA

Como se indicó anteriormente, esta arquitectura se basa en el entrenamiento de un modelo completamente convolucional. Sin embargo, la etapa de codificación y decodificación requieren evaluar ciertos detalles a la hora de su implementación.

En la etapa de codificación, la primera capa consiste en una capa convolucional utilizando una activación del tipo Leaky-Relu con una pendiente de $0,2$. Se escoge esta activación por sobre la rectificación lineal debido a que es favorable frente al problema de desvanecimiento de gradiente, lo cual puede ser un problema al trabajar con una arquitectura de red tan profunda. Luego, las seis capas subsiguientes son también capas convolucionales con la misma función de activación pero con el agregado de que implementan normalización por lotes. Finalmente, la

última capa de esta etapa es convolucional con normalización por lotes y función de activación ReLU.

En la etapa de decodificación es importante tener en cuenta que el tamaño de los campos perceptivos de las capas deben ser múltiplos enteros del tamaño de salto para que no se produzcan artefactos indeseados durante el proceso inverso al cuello de botella que ocurre al utilizar capas deconvolutivas con tamaño de salto mayor a 1. Sin embargo, aun teniendo esto en consideración, las capas de deconvolución pueden generar artefactos indeseados. En lugar de utilizar capas convolutivas se opta por implementar una combinación de dos capas consecutivas: en primer lugar una capa que aumente las dimensiones del espectrograma generando nuevos puntos a partir de una interpolación entre los valores más cercanos, y luego una capa convolutiva. De esta manera se obtiene el efecto del aumento de dimensiones junto con el análisis convolutivo. Esta deconvolución se combina con un drop-out del 50% y una función de activación ReLU en las primeras tres capas del decodificador. Luego, continuan 4 capas idénticas pero omitiendo el drop-out. Finalmente, la última capa del decodificador que a la vez es la capa de salida de la red consiste también en una deconvolución sin drop-out pero utilizando una función de activación ReLU que durante el entrenamiento no se limita, y luego del entrenamiento se limita en $1{,}0$ para hacer predicciones. En todas las capas convolucionales y deconvolucionales se utiliza un tamaño de filtro de $6x6$ y un tamaño de salto igual a 2. Finalmente, la función de costo utilizada para evaluar las predicciones realizadas por el modelo frente a las máscaras ideales en la salida es el error cuadrático medio (MSE) el cual se expresa en la ecuación 16 y para la optimización se utilizó el algoritmo de estimación adaptativa de momento (ADAM) [61] con un valor de tasa de aprendizaje de $0{,}001$.

$$L_{MSE} = \sum_{i=1}^{N-1} \overline{(M_i(t,f) - \hat{M}_i(t,f))^2} \tag{16}$$

La arquitectura de red neuronal se implementó utilizando el lenguaje de programación Python (versión 3.7), particularmente haciendo uso de las bibliotecas Keras[2] y Tensorflow[3].

[2]Página oficial de Keras: https://keras.io/

[3]Página oficial de Tensorflow: https://www.tensorflow.org/

4.6 EVALUACIÓN DEL MODELO

En este trabajo se ponen a prueba cuestiones relativas al manejo, generación y aumentación de datos utilizados para el entrenamiento del modelo. Las evaluaciones se diferencian entre combinaciones entre datos simulados o reales, y en el ordenamiento de la complejidad de los datos durante el entrenamiento.

4.6.1 Combinaciones de bases de datos

En primer lugar, como se cuenta con respuestas al impulso reales, simuladas o generadas, y aumentadas, se prueban combinaciones de estos datos a la hora de formar los diferentes conjuntos de entrenamiento y validación. Es decir, se entrena el modelo utilizando un determinado conjunto, y luego se evalúa su funcionamiento sobre el total de los conjuntos. Esto desemboca en 3 pruebas, en donde los conjuntos de entrenamiento y evaluación quedan determinados según la tabla 1.

Tabla 1: Configuración del primer conjunto de pruebas.

	Prueba 1	**Prueba 2**	**Prueba 3**
Conjunto de entrenamiento	Reales	Generadas	Aumentadas
Conjuntos de evaluación	Reales Generadas Aumentadas	Reales Generadas Aumentadas	Reales Generadas Aumentadas

Luego, se evalúa el rendimiento del modelo propuesto al utilizar combinaciones de conjuntos en la etapa de entrenamiento. Como el objetivo principal es lograr extender la variedad presente en las respuestas al impulso reales, las combinaciones propuestas consisten en combinar las respuestas reales con las generadas y aumentadas como se indica en la tabla 2.

Tabla 2: Configuración del segundo conjunto de pruebas.

	Prueba 1	**Prueba 2**	**Prueba 3**
Conjunto de entrenamiento	Reales + Aumentadas	Reales + Generadas	Reales + Aumentadas + Generadas
Conjuntos de evaluación	Reales Generadas Aumentadas	Reales Generadas Aumentadas	Reales Generadas Aumentadas

4.6.2 Ordenamiento de los datos durante el entrenamiento

Por otro lado, se evalúa la influencia del orden de complejidad con el que las instancias de entrenamiento se le presentan a la red, siguiendo los lineamientos de la técnica de aprendizaje por curriculum. En este caso se considera que una instancia de audio es más compleja de de-reverberar cuando posee un mayor tiempo de reverberación, debido a que esto equivale a una mayor distorsión de la señal original que se quiere recuperar. Entonces, se decide comparar el rendimiento del sistema final al ser entrenado con instancias en las cuales los tiempos de reverberación esten ordenados en forma creciente (curriculum), ordenados en forma decreciente (anti-curriculum) y ordenados aleatoriamente.

Para esta evaluación del orden de los datos en el entrenamiento, es necesario generar un conjunto de datos anecóicos-reverberados de manera controlada, asegurando una distribución homogénea de los tiempos de reverberación presentes en el conjunto final. Para conseguir esto se decide conformar el conjunto de datos a partir de respuestas al impulso aumentadas, ya que estas se pueden generar controlando paramétricamente el tiempo de reverberación medio y la relación directo-reverberado. Se busca cubrir el rango de tiempo de reverberación medio desde $0{,}1\ s$ hasta $3{,}5\ s$ con variaciones de relación directo-reverberado desde $-10\ dB$ a $10\ dB$.

CAPÍTULO 5: RESULTADOS Y DISCUSIONES

El código desarrollado a lo largo de este trabajo se encuentra disponible en un repositorio público en línea junto con su correspondiente documentación [62].

5.1 BASES DE DATOS DE RESPUESTAS AL IMPULSO

Para tener una medida de la variedad de reverberación presente en los conjuntos de respuestas al impulso reales, generadas y aumentadas, se utilizaron los parámetros TR_mid y DRR. En las figuras 23, 24, y 25 se muestran los parámetros acústicos anteriormente mencionados para cada conjunto de respuestas al impulso utilizado.

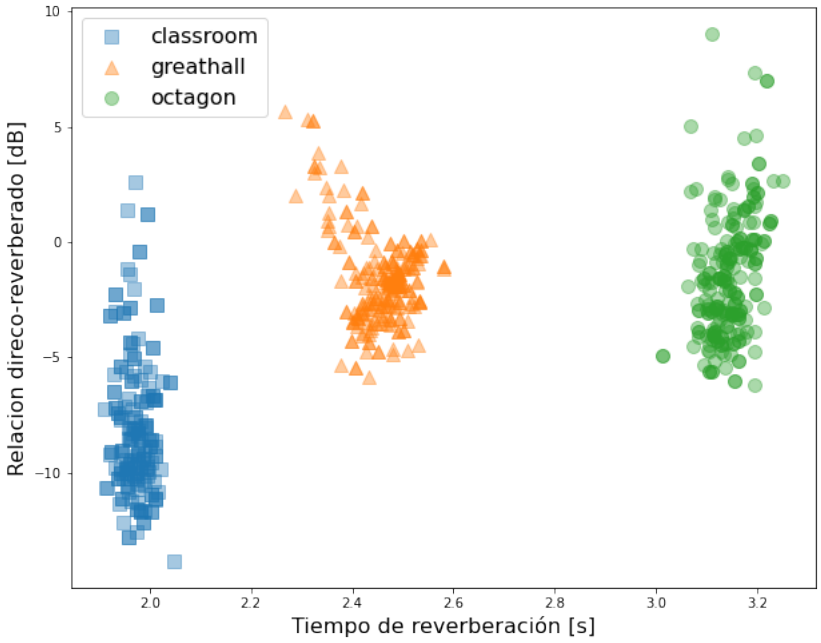


Figura 23: Conjunto de respuestas al impulso reales.

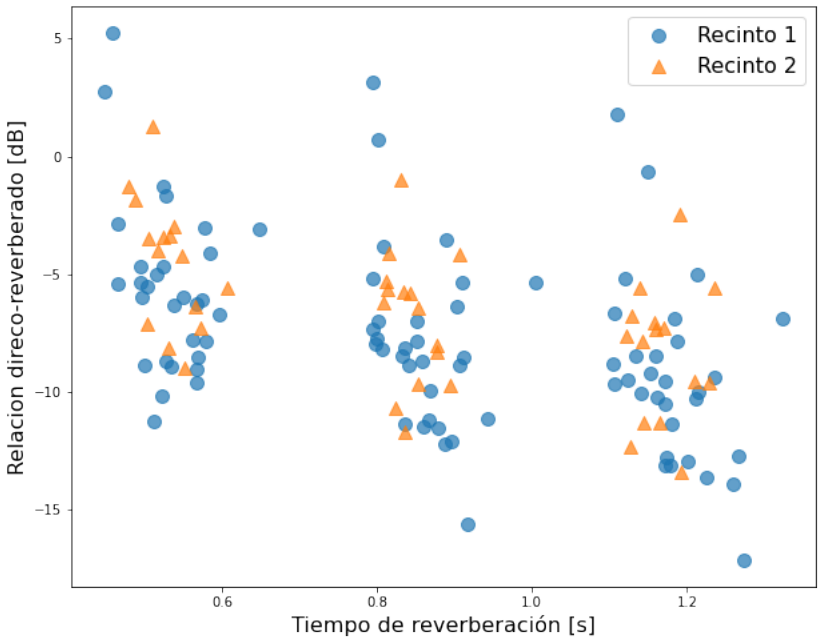


Figura 24: Conjunto de respuestas al impulso generadas.

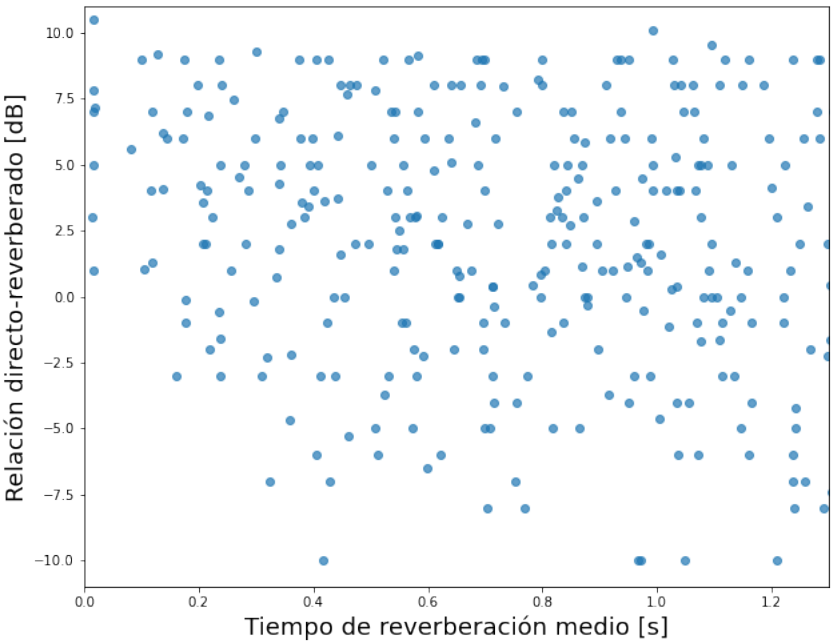


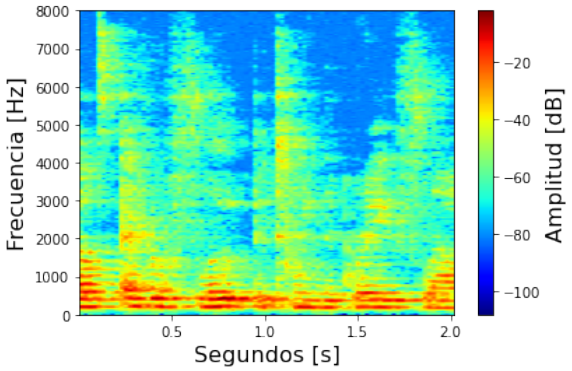
Figura 25: Conjunto de respuestas al impulso aumentadas.

Del análisis de estos conjuntos utilizando como medida la relación directo reverberado y el tiempo de reverberación medio se pueden observar algunas particularidades. Respecto a las respuestas al impulso reales, en la figura 23 se pueden distinguir tres grandes grupos de puntos de acuerdo a los tres recintos de los cuales fueron obtenidas dichas respuestas. Las variaciones de ambos parámetros acústicos ocurren debido a las diferentes posiciones de micrófono que han sido utilizadas en cada recinto, produciendo variaciones de DRR en un rango de $10dB$ y de tiempo de reverberación de $2\ s$ aproximadamente. Pese a estas variaciones, la distribución de los puntos no es uniforme dentro del rango, sin mencionar que no existen respuestas correspondientes a tiempos de reverberación menores a $1{,}8\ s$. Algo similar ocurre con las respuestas al impulso generadas que se muestran en la figura 24. Si bien en este caso se tiene control sobre los puntos centrales de los conjuntos de puntos (se generaron para tiempos de reverberación de $0{,}5\ s$, $0{,}75\ s$ y $1{,}0\ s$) ocurre el mismo fenómeno que con las respuestas al impulso reales, en donde se forman grupos de puntos que no se dispersan uniformemente en el plano. Esto cambia para el tercer conjunto que corresponde a las respuestas al impulso generadas a partir del proceso de aumentación. La dispersión de estas respuestas se observa en la figura 25. A primera vista se observa una mayor uniformidad de los puntos en el plano, ya que no se aprecian conjuntos separados sino más bien una aleatoriedad uniforme a lo largo del rango generado. La uniformidad de la dispersión la podemos atribuir al control que se tiene sobre estos parámetros a la hora de generar las respuestas aumentadas, y la aleatoriedad entre los puntos se debe al hecho de que siempre se parte de una respuesta al impulso real diferente para realizar la aumentación, lo cual produce que el margen entre los parámetros deseados y los obtenidos sea variable. Por otro lado, parece haber una menor cantidad de puntos en la esquina inferior izquierda del grafico, es decir, tiempos de reverberación bajos con relaciones directo-reverberado bajos. Esta es una limitación tanto propia del algoritmo de aumentación como también de la naturaleza de las respuestas al impulso reales, en donde para tiempos de reverberación bajos la energía de la parte tardía de la respuesta es de por sí baja.

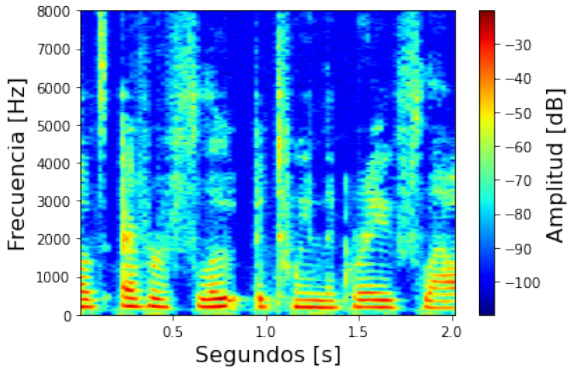
5.2 FUNCIONAMIENTO DEL SISTEMA

En la figura 26 se muestra una instancia de ejemplo del funcionamiento del algoritmo de dereverberación implementado. Durante el entrenamiento, el espectrograma reverberado in-

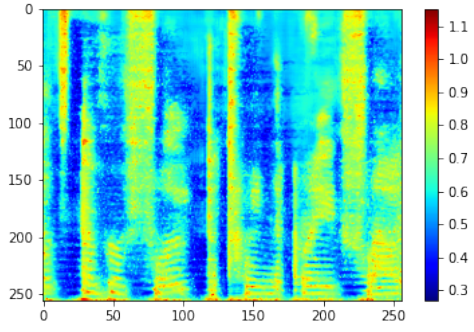
gresa a la red neuronal para procesarse y generar una máscara de amplitud. En la salida, esta máscara se aplica sobre el mismo espectrograma reverberado de entrada para generar el espectrograma dereverberado. Esta es la salida de la red, la cual se compara contra el espectrograma anecoico dentro de la función de costo para poder propagar el error a lo largo de los pesos sinápticos de la red neuronal. Luego, a la hora de hacer predicciones, solo se necesita ingresar un espectrograma reverberado para que la red estime una máscara de amplitud con la cual pueda generarse el espectrograma dereverberado. Cabe aclarar que a lo largo de estos procesos se trabaja únicamente sobre la magnitud de la STFT, a lo que se hace referencia como espectrograma.



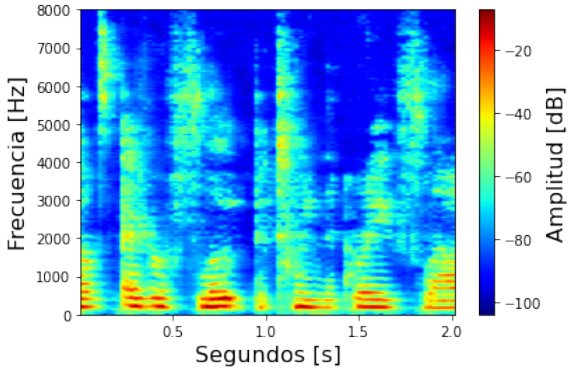
(a) ***Espectrograma reverberado***



(b) ***Espectrograma anecoico***



(c) ***Mascara estimada por la red***



(d) ***Espectrograma dereverberado***

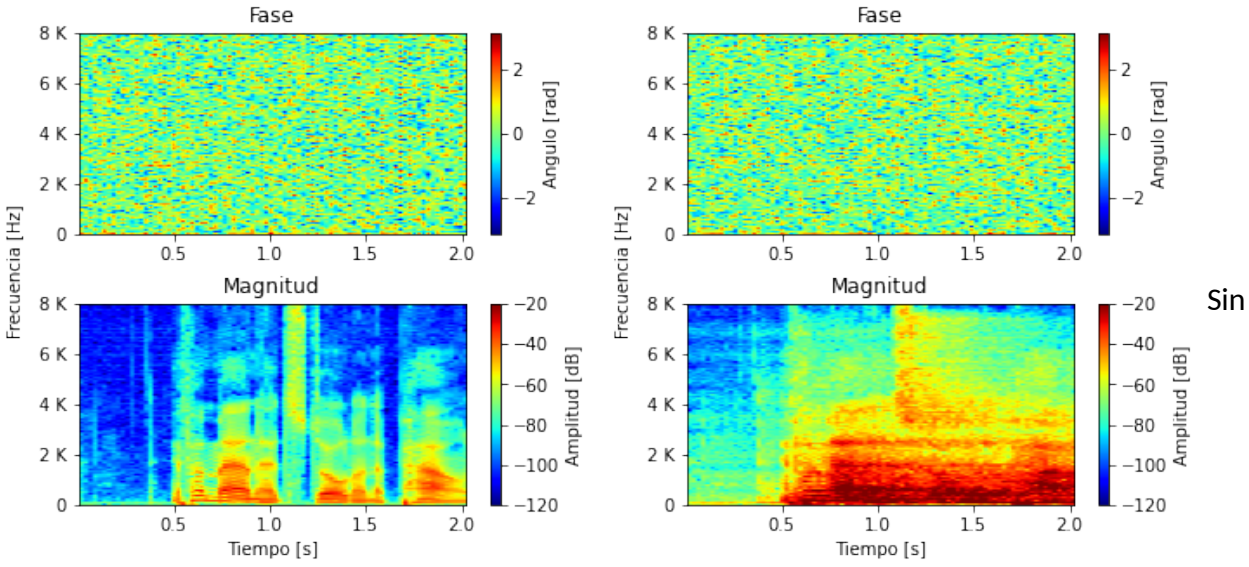
Figura 26: Ejemplo de procesamiento de audio reverberado

Se puede observar que el aprendizaje de la red neuronal desemboca en la generación de máscaras de amplitud que mantienen la ganancia de las componentes propias del habla anecoica y atenúan las componentes introducidas por el fenómeno de la reverberación. La atenuación

de las componentes reverberantes ocurre con mayor precisión para frecuencias bajas, en donde hay más energía. A pesar de esto, se pueden observar rasgos del espectro reverberado aun presentes en el espectro dereverberado, lo que es de esperarse debido a que el proceso únicamente esta aplicando un filtro de amplitud por sobre la magnitud del espectro reverberado.

5.3 RECONSTRUCCIÓN DE AUDIO DEREVERBERADO

El proceso de dereverberación de los audios sucede sobre la magnitud de los espectros STFT de los audios con reverberación. Una vez estimada la magnitud del espectro dereverberado, es necesario combinar esta magnitud con información de fase, para poder conformar un espectrograma complejo apto para antitranformarse y pasar del dominio temporal-frecuencial al dominio temporal (información de audio). Un ejemplo de los espectros de magnitud y fase para un audio con reverberación y sin reverberación se muestra en la figura 27.



(a) Audio sin reverberación

(b) Audio con reverberación

Sin embargo

Figura 27: Espectrogramas de magnitud y fase de los audios para entrenamiento

Se puede apreciar que los espectros de fase parecen contener poca información estructural a comparación de los espectros de magnitud. Tanto la fase del audio con reverberación como la fase del audio sin reverberación parecen contener información aleatoria, y a simple vista son similares entre sí. Es por esto que el proceso de dereverberación se realiza solo sobre la

magnitud de la STFT.

Para determinar la fase del nuevo espectro de magnitud estimado por la red (dereverberado), se consideraron dos alternativas: utilizar directamente la fase del espectro con reverberación o bien utilizar el método iterativo de Griffin-Lim para estimar la fase a partir de la magnitud dereverberada. Este último método iterativo puede inicializarse con una fase determinada (como la fase del audio reverberado) para aprovechar información existente de manera de mejorar la estimación o puede inicializarse de manera aleatoria. Para determinar el número necesario de iteraciones a utilizar en el algoritmo de Griffin-Lim se evaluó la evolución de las métricas utilizadas en este trabajo (SDR, SRMR y ESTOI) en relación al número de iteraciones en la obtención de la fase. En la figura 28 se pueden observar estas relaciones para cada métrica, teniendo en cuenta que se utilizó el algoritmo inicializado desde una fase aleatoria. Se puede apreciar que los valores se estabilizan al aproximarse a 100 iteraciones, siendo este el número de iteraciones que se utilizó para las pruebas subsiguientes.

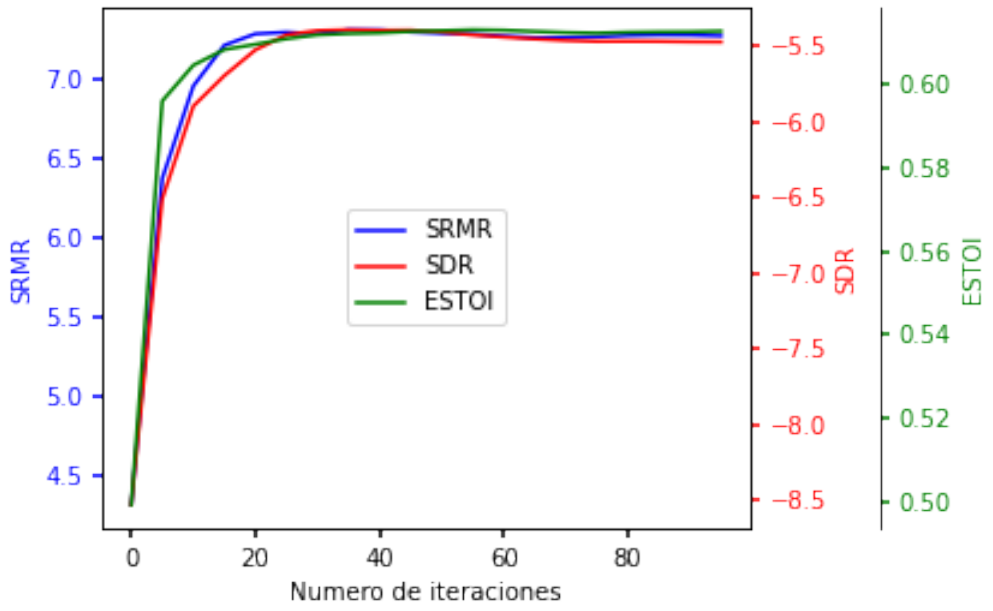


Figura 28: Influencia del número de iteraciones del algoritmo de Griffin-Lim.

En la tabla 3 se muestra el resultado de la comparación entre las distintas alternativas para la determinación de la fase del espectro dereverberado. Para hacer esta comparación se utilizó audio reverberado como referencia, sobre el cual se calculan las métricas objetivas SDR, SRMR

y ESTOI. Luego, se implementó cada alternativa de obtención de fase para el espectro de magnitud dereverberado con el fin de generar información de audio y calcular estas mismas métricas. En la tabla se expresan las variaciones de las métricas para cada alternativa con respecto al audio reverberado. Se puede observar que la principal diferencia ocurre sobre el parámetro SDR. La reconstrucción de fase utilizando el algoritmo de Griffin-Lim inicializado de manera aleatoria empeora el resultado de esta métrica, lo cual es un efecto contrario al deseado. Sin embargo, utilizando este algoritmo inicializado con la fase reverberada genera una mejora. Finalmente, la dereverberación utilizando directamente la fase reverberada produce una mejora mucho mayor que los metodos iterativos previamente mencionados. Para las otras dos métricas, SRMR y ESTOI, los métodos iterativos producen mejores resultados que la utilización directa de la fase reverberada, pero las diferencias entre las alternativas son menores.

Tabla 3: Comparación de métodos de reconstrucción de espéctrograma complejo para generar audio

	SDR	**SRMR**	**ESTOI**
Audio reverberado (referencia)	*- 3.11*	*1.73*	*0.29*
Δ Dereverberación con fase reverberada	+4.27	+4.53	+0.31
Δ Dereverberación Griffin-Lim iniciado con fase reverberada	+1.38	+5.13	+0.33
Δ Dereverberación Griffin-Lim iniciado con fase aleatoria	-2.92	+5.24	+0.33

De los resultados obtenidos sobre las distintas alternativas estudiadas para obtener el espectro de fase que complemente a la magnitud dereverberada generada se puede remarcar en primer lugar que las métricas utilizadas no siempre reflejan con fidelidad lo que ocurre en la percepción auditiva de los resultados. Durante la realización de estas pruebas, ocurrió que el método que arrojaba mejores resultados sobre las métricas no era el que mejor se percibia auditivamente. También, ocurrió que sobre determinados ejemplos ambas alternativas generaban distorsiones subjetivamente similares pero producían resultados notoriamente distantes sobre las métricas. De igual manera, la utilización de la fase reverberada de manera directa resultó ser el método más robusto frente a las métricas. Esto, sumado a su utilización en otros trabajos del estado del arte AGREGAR REFERENCIA hizo que esta alternativa sea la escogida a lo largo de este trabajo. Sin embargo, este análisis de fase deja en evidencia la falta de correlación de ciertas métricas como el SDR con la percepción auditiva de los resultados. Por cuestiones como

esta existen nuevas variantes de estas métricas que buscan generar una mayor robustez frente a estas cuestiones de reconstrucción de audio [63].

5.4 DEREVERBERACIÓN DEL HABLA Y MANEJO DE DATOS

Para las evaluaciones se tuvieron en cuenta tres conjuntos de datos de acuerdo al tipo de respuestas al impulso utilizadas para generar la reverberación: reales, generadas y aumentadas. Para medir el desempeño de la tarea de dereverberación, las métricas se evaluan sobre los conjuntos reverberados y luego sobre sus correspondientes resultados dereverberados. Los resultados de estas métricas para los conjuntos reverberados se puede observar en la tabla 5.

Tabla 4: Resultados de las métricas sobre los conjuntos reverberados

Conjunto	**SDR**	**SRMR**	**ESTOI**
Reales	-3.94	1.22	0.28
Generadas	2.89	2.53	0.46
Aumentadas	8.09	3.19	0.64

Los resultados obtenidos para el primer conjunto de pruebas definidos en la tabla 1 se expresan en las figuras 29, 30 y 31 para las variaciones de las métricas SDR, SRMR y ESTOI respectivamente.

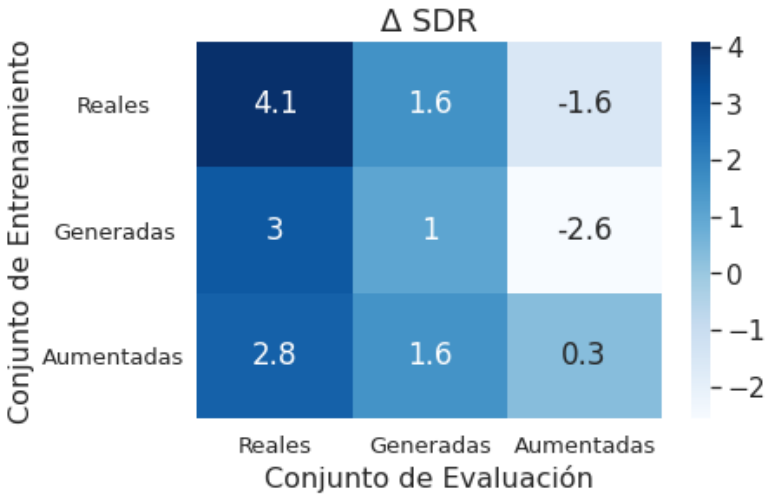


Figura 29: Variaciones de SDR para el primer conjunto de pruebas.

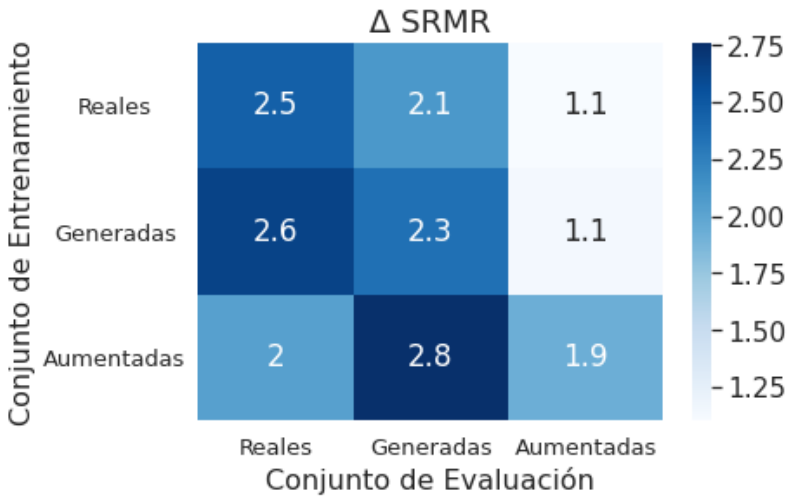


Figura 30: Variaciones de SRMR para el primer conjunto de pruebas.

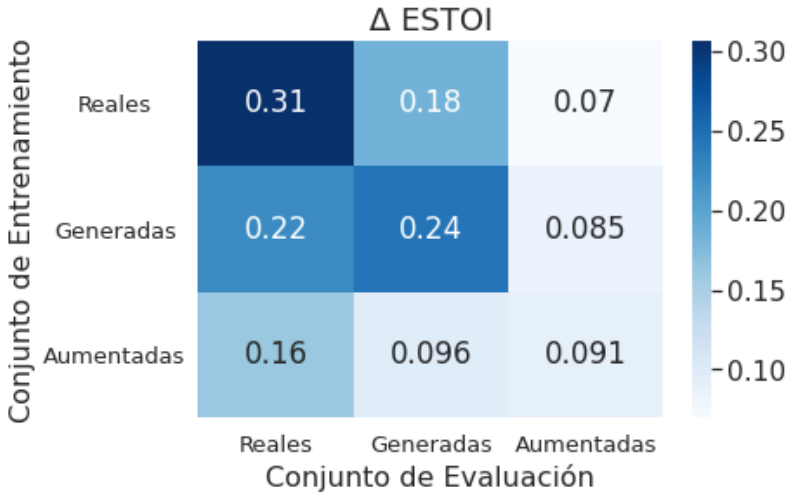


Figura 31: Variaciones de ESTOI para el primer conjunto de pruebas.

Del primer conjunto de pruebas se esperaba obtener los mejores resultados para aquellos casos en los que el conjunto de entrenamiento y el conjunto de evaluación coinciden. Esto ocurrió para la evaluación con la métrica ESTOI. Para las otras métricas, el comportamiento esperado ocurrió en general para los conjuntos formados con respuestas al impulso reales y aumentadas, pero no para las generadas. Particularmente, utilizar respuestas al impulso aumentadas durante el entrenamiento produjo mejores resultados al evaluar sobre respuestas al impulso generadas que usando respuestas al impulso generadas durante el entrenamiento. Esto puede deberse al hecho de que, si bien ambos conjuntos contienen tiempos de reverberación

del mismo rango, las respuestas al impulso aumentadas tienen una distribución más uniforme a lo largo de este rango.

Los resultados correspondientes al segundo conjunto de pruebas definido en la tabla 2 se muestran en las figuras 32, 33 y 34 para las variaciones de las métricas SDR, SRMR y ESTOI respectivamente.

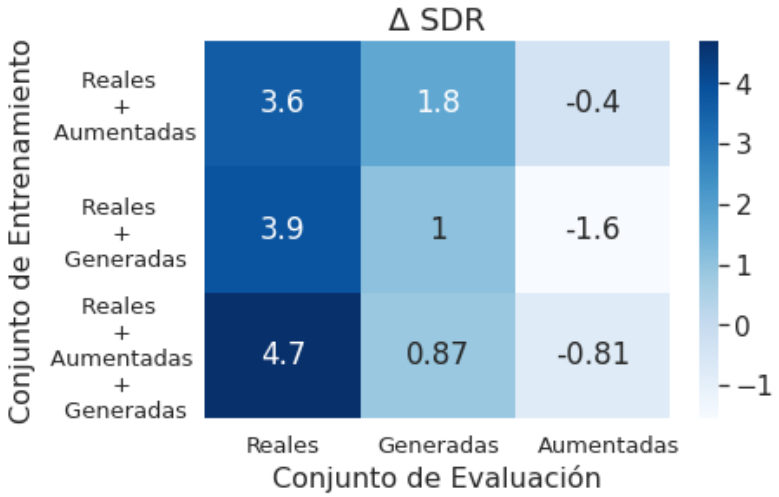


Figura 32: Variaciones de SDR para el segundo conjunto de pruebas.

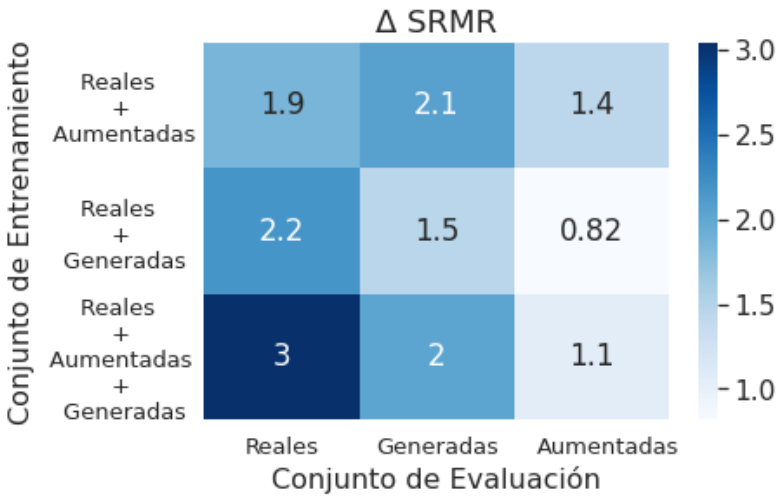


Figura 33: Variaciones de SRMR para el segundo conjunto de pruebas.

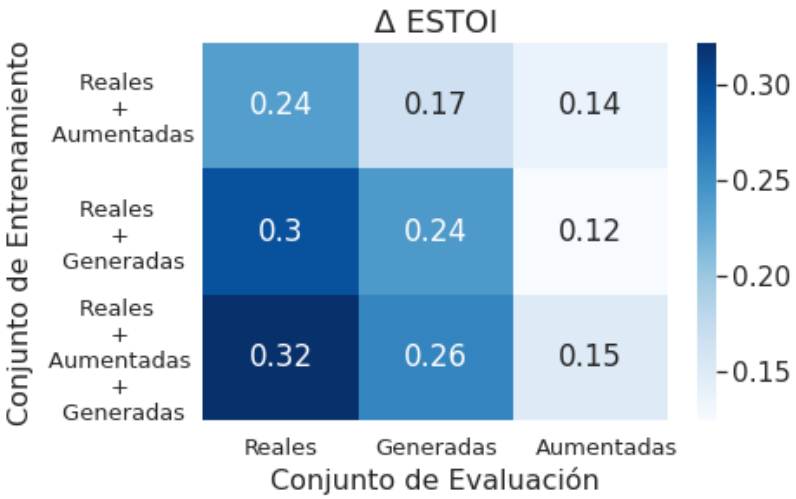


Figura 34: Variaciones de ESTOI para el segundo conjunto de pruebas.

Para el segundo conjunto de pruebas se combinaron tipos de respuestas al impulso en la conformación de los datos de entrenamiento y se volvió a evaluar en los mismos conjuntos de la primera prueba, asegurándose que el número de instancias de entrenamiento se mantenga fijo en todas las pruebas. Se debe tener en consideración que es de mayor importancia para éste trabajo evaluar el rendimiento al evaluar sobre respuestas al impulso reales, ya que es el objetivo principal del sistema implementado. En esta prueba, para todas las métricas los mejores resultados se obtuvieron al combinar los tres tipos de datos en la conformación del conjunto de entrenamiento. Es decir, una mayor diversidad de impulsos presentes a la hora de generar los datos de entrenamiento desemboca en una mejora en el rendimiento del sistema. Además, la combinación de respuestas al impulso reales-generadas arrojó mejores resultados que la combinación reales-aumentadas para todas las métricas. Esto puede deberse al hecho de que las respuestas al impulso aumentadas si bien varían la pendiente de caída de la cola reverberante, mantienen el mismo perfil frecuencial que las respuestas al impulso reales de las que provienen. En este aspecto, las respuestas al impulso generadas pueden enriquecer en mayor medida la diversidad de los datos utilizados aportando nuevos perfiles de tiempo de reverberación, lo cual puede llevar a mejores resultados generales.

5.5 APRENDIZAJE POR CURRICULUM

Para evaluar la influencia del ordenamiento de los datos en el proceso de entrenamiento en primer lugar se generó una base de datos de respuestas al impulso asegurando una adecuada dispersión de los parámetros acústicos de relación directo-reverberado y tiempo de reverberación medio. Para poder conseguir esto, se partió de la base de datos de respuestas al impulso reales C4DM y se aplicó el método de aumentación. Esta vez se generaron tiempos de reverberación medio desde $0{,}1\ s$ a $3{,}5\ s$ y relaciones directo-reverberado de $-10\ dB$ a $10\ dB$. En la figura 35 se puede observar la dispersión de los parámetros acústicos mencionados en el conjunto de respuestas al impulso conformado.

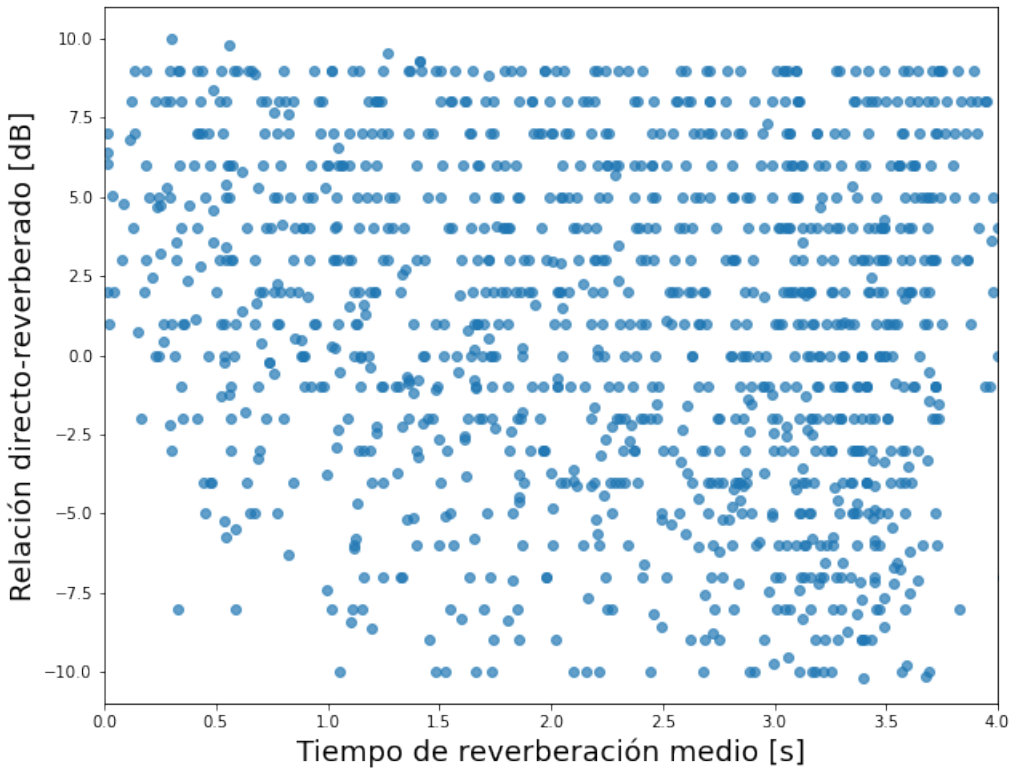


Figura 35: Respuestas al impulso generadas por aumentación.

Partiendo de la información del tiempo de reverberación medio de cada respuesta al impulso, se generaron pares de audios anecoicos-reverberados junto con un registro que indicaba

que tiempo de reverberación medio correspondía con cada audio reverberado. Este registro se utilizó para conformar los esquemas de entrenamiento que fueron evaluados. Entonces, se organizaron los datos de entrenamiento de tres maneras: con tiempos de reverberación crecientes, decrecientes y aleatorios. Cabe aclarar que la red se entrenó por una sola época sobre estos conjuntos de datos. Una vez realizado el entrenamiento, se utilizó el modelo entrenado para hacer predicciones sobre audios reververados y se calcularon las métricas objetivas sobre los resultados de cada variante. En las figuras 36, 37, y 38 se muestran los resultados obtenidos para las métricas SDR, SRMR y ESTOI respectivamente.

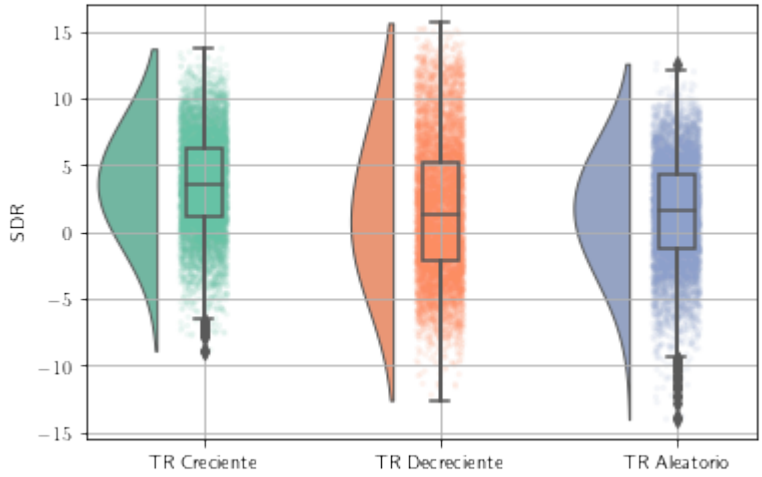


Figura 36: Comparación de SDR entre tipos de ordenamiento de datos durante el entrenamiento.

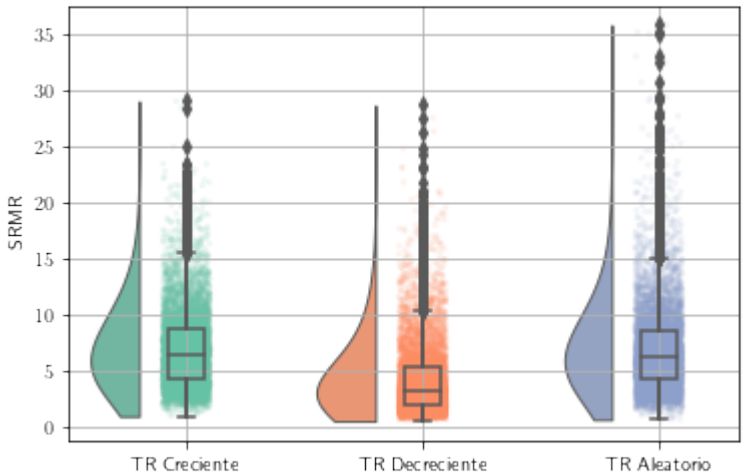


Figura 37: Comparación de SRMR entre tipos de ordenamiento de datos durante el entrenamiento.

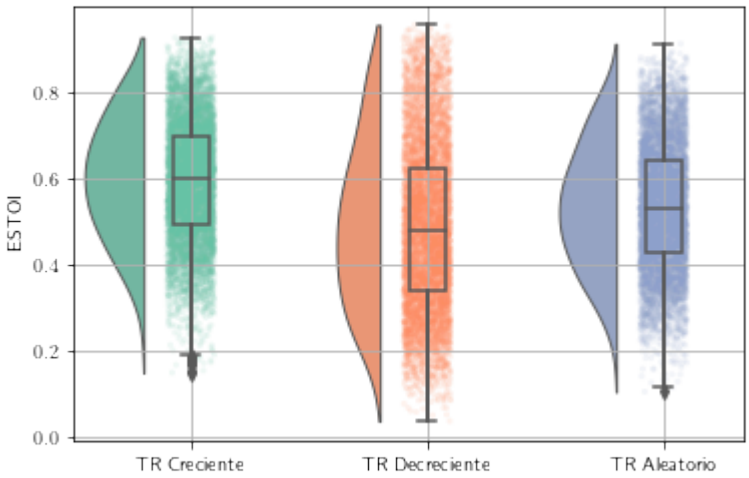


Figura 38: Comparación de ESTOI entre tipos de ordenamiento de datos durante el entrenamiento.

A primera vista se puede percibir que para todas las métricas el resultado obtenido para el entrenamiento con tiempos de reverberación crecientes es el mejor, el resultado obtenido para el entrenamiento con tiempos de reverberacón decrecientes es el peor, y el resultado para el entrenamiento con tiempos de reverberación aleatorios esta en un punto medio entre los dos anteriores, en ocasiones más cerca del mejor y en ocasiones más cerca del peor. Para ilustrar mejor este comportamiento, en la tabla 5 se muestran las medianas de las métricas obtenidas para cada esquema de entrenamiento.

Tabla 5: Medianas correspondientes a cada esquema de entrenamiento.

	$\tilde{X}_{SDR}$	$\tilde{X}_{SRMR}$	$\tilde{X}_{ESTOI}$
TR Creciente	3.54	6.37	0.59
TR Decreciente	1.24	3.25	0.47
TR Aleatorio	1.61	6.24	0.53

Finalmente, con estos resultados se pudo corroborar el impacto del ordenamiento de los datos de entrenamiento en el rendimiento final del sistema. En este caso, la complejidad de los datos se asoció al tiempo de reverberación de cada instancia. Con esto, al entrenar con datos ordenados por tiempo de reverberación de manera creciente, es decir, iniciando con tiempos de reverberación bajos y aumentando progresivamente el tiempo de reverberación se obtuvieron mejores resultados para todas las métricas. También se comprobó que el ordenamiento inverso

produce el peor resultado de los tres, y el orden aleatorio cae en un punto medio entre estos dos casos. Esto es consistente con la teoría de la técnica de aprendizaje por curriculum. En primera instancia, para tiempos de reverberación bajos, es sencillo para la red neuronal mantener el sonido directo que es predominante en la señal debido a la escasa reverberación (la máscara ideal se asemeja a una máscara unitaria). Esta información aprendida un primer momento guía en cierta medida al algoritmo para ir aprendiendo de instancias progresivamente más complejas en donde la energía proveniente de la reverberación es cada vez mayor.

CAPÍTULO 6: CONCLUSIONES

A lo largo de este trabajo se implementó un algoritmo de aprendizaje profundo para la tarea de dereverberación de señales de habla. Se utilizó una estructura de tipo autoencoder con conexiones de salto, la cual se entrenó para estimar máscaras de amplitud que al aplicarse sobre un espectro de magnitud reverberado generen un espectro de magnitud dereverberado, que luego de combinarse con el espectro de fase reverberado pueda desembocar en información de audio dereverberado.

Particularmente se estudió la influencia del manejo de datos para el entrenamiento de este algoritmo, manipulando y generando respuestas al impulso de diversas maneras.

Se generaron datos de entrenamiento y evaluación a partir de respuestas al impulso reales, generadas y aumentadas. Se pudo comprobar que una mayor diversidad de respuestas al impulso para la generación de los datos de entrenamiento genera mejores resultados, y que tanto las respuestas al impulso generadas como las aumentadas son útiles como método de aumentación de datos de entrenamiento.

Por otro lado, del análisis del ordenamiento de datos de entrenamiento se comprobó que el tiempo de reverberación medio se puede relacionar de manera directa con el nivel de dificultad del proceso de dereverberación, y que el ordenamiento de datos de entrenamiento en orden de dificultad creciente guia el entrenamiento y genera mejores resultados finales, acorde a la técnica de aprendizaje por curriculum.

Estas conclusiones generan un aporte importante para el estudio de las tareas de dereverberación de audio, para las cuales las bases de datos de respuestas al impulso existentes son escasas, poco abarcativas y en ocasiones difíciles de conseguir o muy costosas.

CAPÍTULO 7: LINEAS FUTURAS DE INVESTIGACIÓN

La dereverberación del habla a partir de la estimación de máscaras de amplitud demostró ser un proceso eficaz pero limitado. Pueden obtenerse mejoras expandiendo el procesamiento para que se consideren también las componentes complejas de la STFT, es decir, realizar la dereverberación en amplitud y fase. Esto se podría conseguir estimando máscaras complejas o bien estimando máscaras independientes para magnitud y fase. Por otro lado, en este enfoque se procesa de la misma manera la totalidad del espectro. Sabiendo que la reverberación aporta más energía en bajas frecuencias, podría segmentarse el espectro en bandas y procesar cada banda con configuraciones diferentes. De manera más general, también podrían aprovecharse dependencias temporales presentes en la reverberación introduciendo capas recurrentes dentro de la estructura de capas convolucionales.

Otro aspecto a considerar es el referido a las métricas utilizadas tanto para el entrenamiento como para la evaluación de los modelos. Se deben analizar y proponer nuevas métricas que se correlacionen de manera directa con la percepción auditiva de la tarea de dereverberación [64]. De esta forma el entrenamiento se realizará en pos de una mejora en la percepción auditiva de los resultados, y los resultados de las evaluaciones podrán ser utilizados para tomar decisiones que mejoren la calidad del proceso de dereverberación de manera consistente y objetiva.

Respecto al proceso de aumentación, es de interés tener un cierto control sobre el perfil frecuencial de tiempo de reverberación con el que se generan las respuestas al impulso aumentadas. En este trabajo, estas respuestas al impulso siguen las mismas relaciones interfrecuenciales que las respuestas al impulso de las que parten, generando el mismo perfil de tiempo de reverberación pero desplazado. Una mayor diversidad de respuestas podría conseguirse si se logra controlar también la forma general de la curva de tiempo de reverberación.

Perfeccionar las técnicas de aumentación de respuestas al impulso podría llevar en un futuro a la creación de un corpus de datos de dominio libre destinado específicamente al desarrollo de sistemas de dereverberación del habla, estandarizando y dando acceso a los mismos datos de entrenamiento y evaluación para todos los estudios de dereverberación. Generar esta base de datos de acceso libre puede asegurar que exista una correcta comparación entre las soluciones propuestas por diferentes trabajos, de forma tal que cada resultado logre contribuir de manera

objetiva a la mejora general de las técnicas de dereverberación.

Bibliografía

[1] L. Deng, G. Hinton y B. Kingsbury. “New types of deep neural network learning for speech recognition and related applications: An overview”. En: *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process (ICASSP)* (2013).

[2] H. Chung y col. “Noise-adaptive deep neural network for single-channel speech enhancement”. En: *Proc. Int. Workshop on Machine Learning for Signal Process. (MLSP)* (2018).

[3] Pearson J. y col. “Robust distant-talking speech recognition”. En: *IEEE International Conference on. Vol. 1* (1996). Acoustics, Speech, and Signal Processing.

[4] Castellano Pierre J., S. Sradharan y David Cole. “Speaker recognition in reverberant enclosures”. En: *IEEE International Conference on. Vol. 1* (1996). Acoustics, Speech, and Signal Processing.

[5] DiBiase Joseph H., Harvey F. Silverman y Michael S. Brandstein. “Robust localization in reverberant rooms”. En: *Springer Berlin Heidelberg* (2001). Microphone Arrays.

[6] Philipos C. Loizou. *SPEECH ENHANCEMENT, Theory and Practice*. CRCR Press, 2013.

[7] Masato Miyoshi y Yutaka Kaneda. “Inverse Filtering of Room Acoustics”. En: *IEEE Transactions on Acoustics, Speech and Signal Processing* (1988).

[8] Stephen T. Neely y Jont B. Allen. “Invertibility of a room impulse response”. En: *Acoustics Research Department, Bell Laboratories, Murray Hill, New Jerse* (1997).

[9] Laurence R. Rabiner y W. Schafer Ronald. *Digital Speech Processing*. Prentice- Hall, 1975.

[10] Yegnanarayana B. y Satyanarayana P. “Enhancement of Reverberant Speech Using LP Residual Signal”. En: *IEEE Transactions on Speech and Audio Processing* (2000).

[11] Gannot S. y Moonen M. “Subspace Methods for Multimicrophone Speech Dereverberation”. En: *EURASIP journal on advances in signal processing* (2003).

[12] N. Roman y D. L. Wang. “Pitch-based monaural segregation of reverberant speech”. En: *Journal of Acoustical Society of America* (2006).

[13] M. Avendano y H. Hermansky. “Study on the dereverberation of speech based on temporal envelope filtering”. En: *Proc. of ICSLP* (1996).

[14] K. Lebart y J. M. Boucher. “A New Method Based on Spectral Subtraction for SpeechDereverberation”. En: *Acta Acustica united with Acustica* (2001).

[15] K. Lebart, J.-M. Boucher y P. Denbigh. “A new method based on spectral subtraction for speech dereverberation”. En: *Acta Acustica united with Acustica* (2001).

[16] M. Wu y D. L. Wang. “A two-stage algorithm for one-microphone reverberant speech enhancement”. En: *IEEE Transactions on Acoustics, Speech and Signal Processing* (2006).

[17] D. Wang. *On ideal binary mask as the computational goal of auditory scene analysis*. Kluwer, 2005, págs. 181-197.

[18] A. S. Bregman. “Auditory Scene Analysis”. En: *Cambridge, MA: MIT Press* (1990).

[19] Nicoleta Roman y John Woodruff. “Intelligibility of reverberant noisy speech with ideal binary masking”. En: *Journal of Acoustical Society of America* (2011).

[20] Nicoleta Roman y John Woodruff. “Speech intelligibility in reverberation with ideal binary masking: Effects of early reflections and signal-to-noise ratio threshold”. En: *Journal of Acoustical Society of America* (2013).

[21] O. Hazrati, J. Lee y P. C. Loizou. “Blind binary masking for reverberation suppression in cochlear implants”. En: *Journal of Acoustical Society of America* (2013).

[22] Z. Jin y D. L. Wang. “A supervised learning approach to monaural segregation of reverberant speech”. En: *IEEE Transactions on Acoustics, Speech and Signal Processing* (2009).

[23] D. S. Williamson y D. Wang. “Time-frequency masking in the complex domain for speech dereverberation and denoising”. En: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2017).

[24] I. Kodrasi y H. Bourlard. “Single-channel late reverberation power spectral density estimation using denoising autoencoders”. En: *Proc. Interspeech* (2018).

[25] C. Li y col. “Single-channel speech dereverberation via generative adversarial training”. En: *Proc. Interspeech* (2018).

[26] Kun Han, Yuxuan Wang y DeLiang Wang. “Learning spectral mapping for speech dereverberation”. En: *IEEE International Conference on Acoustic, Speech and Signal Processing* (2014).

[27] Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'reilly media, 2019.

[28] Ori Ernst y col. "Speech Dereverberation Using Fully Convolutional Networks". En: *22nd International Conference on Digital Signal Processing* (2017).

[29] F. Weninger y col. "Deep recurrent de-noising auto-encoder and blind de-reverberation for reverberated speech recognition". En: *IEEE International Conference on Acoustics, Speech and Signal Processing* (2014).

[30] João Felipe Santos y Tiago H. Falk. "Speech Dereverberation with Context-aware Recurrent Neural Networks". En: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2017).

[31] Daniel W. Griffin y Jaa S. Lim. "Signal Estimation for Modified Short-Time Fourier Transform". En: *IEEE Trans. Acoust. Speech Signal Process* (1984).

[32] D. S. Wang, Y. X. Zou y W. Shi. "A Deep Convolutional Encoder-Decoder Model for Robust Speech Dereverberation". En: *22nd International Conference on Digital Signal Processing* (2017).

[33] Jiaqi Su, Zeyu Jin y Adam Finkelstein. "HiFi-GAN: High-Fidelity Denoising and Dereverberation Based on Speech Deep Features in Adversarial Networks". En: *Proc. Interspeech* (2020).

[34] Leo L. Beranek. *Acústica*. Segunda Edición, 1969.

[35] Alan V. Oppenheim, Ronald W. Shafer y John R. Buck. *Discrete-Time Signal Processing*. Prentice- Hall, 1989.

[36] E. Oran Brigham. *The Fast Fourier Transform And Its Aplications*. Prentice- Hall, 1988.

[37] *Documentación oficial de Mathworks del módulo "DSP"de Matlab*. https://la.mathworks.com/help/dsp/ref/dsp.stft.html. Extraido el 15 de Octubre de 2021.

[38] Meinard Müller. *Fundamentals of Music Processing*. Springer, 2021.

[39] Udo Zölzer. *Digital Audio Signal Processing*. O'Reilly, 2017.

[40] Farina Angelo. "Simultaneous measurement of impulse response and distortion with a swept sine technique". En: *108th AES Convention* (2000).

[41] M. R. Schroeder. “Digital Simulation of Sound Transmission in Reverberant Spaces”. En: *Journal of the Acoustical Society of America* (1970).

[42] J. B. Allen y D. A. Berkeley. “Image Method for Efficient Simulating Small RoomAcoustics”. En: *Journal of the Acoustical Society of America* (1979).

[43] Tiago H. Falk y Wai-Yip Chan. “A Non-intrusive Quality Measure of Dereverberated Speech”. En: *Department of Electrical and Computer Engineering* (2008). Queen’s University, Kingston, Ontario, Canada.

[44] Tiago H. Falk y Wai-Yip Chan. “A Non-intrusive Quality Measure of Dereverberated Speech”. En: *Department of Electrical and Computer Engineering* (2008). Queen’s University, Kingston, Ontario, Canada.

[45] Emmanuel Vincent, Rémi Gribonval y Cédric Févotte. “Performance Measurement in Blind Audio Source Separation”. En: *IEEE Transactions on Audio, Speech and Language Processing* (2006).

[46] I.A. Basheer y M. Hajmeer. “Artificial neural networks: fundamentals, computing, design, and application”. En: *Journal of Microbiological Methods* (2000).

[47] Chunlei Liu, Longbiao Wang y Jianwu Dang. “A Logical Calculus of Ideas Immanent in Nervous Activity”. En: *Bulletin of Mathematical Biophysics* (1943).

[48] Josh Patterson y Adam Gibson. *Deep Learning: A Practitioner’s Approach*. Wiley, 2008.

[49] Francois Chollet. *Deep Learning with Python*. Manning Publications, 2017.

[50] Y. Bengio y col. “Curriculum Learning”. En: *Proceedings of the 26thInternational Conference on Machine Learning* (2009).

[51] D. L. Wang y J. S. Lim. “The unimportance of phase in speechenhancement”. En: *IEEE Trans. Acoust. Speech Signal Process* (1982).

[52] Y. Ephraim y D. Malah. “Speech enhancement using a minimummean-square error short-time spectral amplitude estimator”. En: *IEEE Trans. Acoust. Speech Signal Process* (1984).

[53] R. Stewart y M. Sander. “Database of omnidirectional and B-format impulse responses”. En: *Proc. IEEE Int. Conf. Acous-tics, Speech, and Signal Process. (ICASSP)* (2010).

[54] R. Scheibler, E. Bezzam e I. Dokmanić. “Pyroomacoustics: A Python package for audio room simulations and array processing algorithms”. En: *Proc. IEEE ICASSP* (2018).

[55] J. B. Allen y D. A. Berkley. “Image method for efficiently simulating small-room acoustics”. En: *J. Acoust. Soc. Am., vol. 65* (1979).

[56] Nicholas J. Bryan. “Impulse response data augmentation and deep neural networks for blind room acoustic parameter estimation”. En: *Adobe Research* (2019). San Francisco, CA, USA.

[57] A. Lundeby, T. E. Vigran H. Bietz y M. Vorlander. “Uncertainties of Measurements in Room Acoustics”. En: *Acta Acustica united with Acustica*, (1995).

[58] Matthew G. Blevins y col. “Quantifying the just noticeable difference of reverberation time with band-limited noise centered around 1000 Hz using a transformed up-down adaptive method”. En: *International Symposium on Room Acoustics* (2013). Toronto, Canada.

[59] Panayotov V. y col. “Libris-peech: an asr corpus based on public domain audio books”. En: *In Acoustics,Speech and Signal Processing (ICASSP)* (2015).

[60] Yuxuan Wang, Arun Narayanan y De Liang Wang. “On Training Targets for Supervised Speech Separation”. En: *IEEE/ACM Trans Audio Speech Lang Process* (2014).

[61] Kingma D. P. y Ba J. “Adam: A Method for Stochastic Optimization”. En: *ICLR* (2014).

[62] *Martin Bernardo Meza, Repositorio del trabajo ”Dereverberación del habla a partir de algoritmos de aprendizaje profundo”* . https://github.com/martinBmeza/deep-dereverb. Extraido el 6 de Octubre de 2021.

[63] Jonathan Le Roux y col. “SDR - half-baked or well done?” En: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2018).

[64] Pranay Manocha y col. “CDPAM: Constrastive learning for perceptual audio similarity”. En: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2021).

ANEXO A: AUMENTACIÓN DE TIEMPO DE REVERBERACIÓN

El proceso de aumentación de respuestas al impulso puede realizarse tomando como parámetros del proceso ciertos descriptores acústicos como el tiempo de reverberación TR_{60}. Esto es, partiendo de una respuesta al impulso con un TR_{60} determinado, se busca generar nuevas respuestas al impulso cuyo TR_{60} pueda ser controlado paramétricamente para ocupar de manera homogénea y balanceada un rango de valores de interés. El TR_{60} se relaciona directamente con la forma de la envolvente de caída de nivel exponencial presente en la parte tardía de las respuestas al impulso. El proceso de aumentación equivale a modificar esta pendiente de caída, multiplicando la señal original por una nueva envolvente que produzca el efecto deseado en la envolvente resultante. Los pasos a seguir para realizar este proceso son:

- Acondicionamiento de la respuesta al impulso de entrada.
- Filtrado por bandas de octava o bandas de tercio de octava.
- Estimación de piso de ruido.
- Estimación de envolvente de caída.
- Sintetizar una señal aplicando la envolvente estimada con piso de ruido cero a una señal de ruido Gaussiano.
- Realizar el cross-fade entre la señal sintetizada y la señal original en el punto inicial del piso de ruido.
- Aumentación de la envolvente de caída multiplicando la señal por la correspondiente envolvente exponencial creciente/decreciente.
- Suma de las sub-bandas para obtener la señal resultante en su espectro completo.
- Integración de la parte tardía aumentada con la parte temprana de la respuesta al impulso inicial.

A continuación se explica cada paso del algoritmo con mayor profundidad. En primer lugar, se define una determinada frecuencia de muestreo y profundidad de bits para trabajar con la señal de entrada, en este caso una respuesta al impulso real. Una vez asegurada la homogeneidad de estas características, la señal se normaliza para trabajar en un rango de amplitud acotado en el intervalo $[-1, 1]$ y luego se separa la parte temprana de la parte tardía de la respuesta al impulso. Esto último se realiza utilizando una ventana de tolerancia de $t_0 = 2{,}5ms$. Para los pasos siguientes se trabaja únicamente modificando la parte tardía, y la parte temprana se almacena para ser utilizada en el paso final a la hora de reconstruir la respuesta completa. En la figura 39 se muestra la respuesta al impulso desde la que se parte, distinguiendo la descomposición temporal de la misma y luego la parte tardía de la respuesta al impulso aislada con la que se va a trabajar durante el proceso.

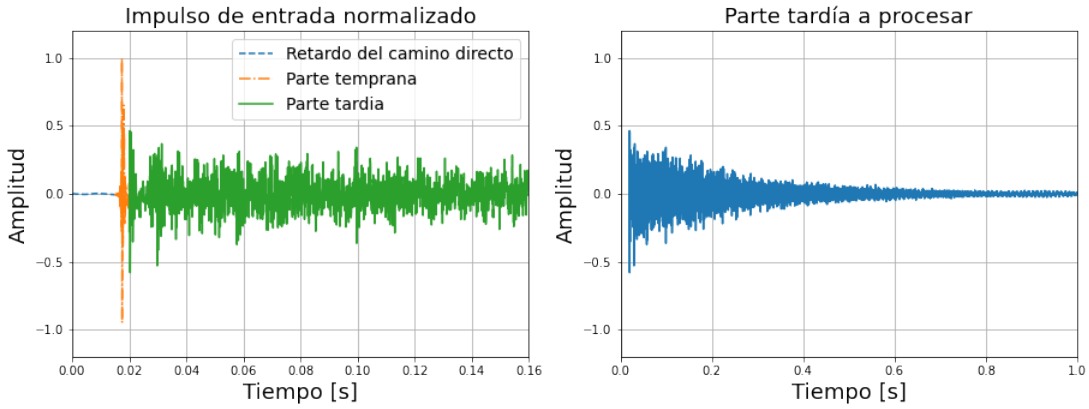


Figura 39: Descomposición de la respuesta al impulso a procesar durante la aumentación temporal

Luego, el siguiente paso consiste en descomponer la señal en bandas de octava o tercios de octava. En esta demostración se trabaja con bandas de octava desde $125\ Hz$ hasta $4000\ Hz$ teniendo en cuenta que se trabaja con una frecuencia de muestreo de 16000 muestras por segundo. Es necesario trabajar en sub-bandas frecuenciales para contemplar la dependencia del tiempo de reverberación con la frecuencia, y mantener esa característica en las señales a generar en este proceso. Para conseguir esta descomposición en sub-bandas frecuenciales se crea un banco de filtros. El mismo se compone de filtros Butterworth que van siendo creados a partir de las frecuencias centrales que se quiera obtener en cada banda. El proceso consiste en crear filtros pasa-banda para generar una banda de paso alta y una banda de paso baja. Luego,

se toma la banda de paso alta y se la vuelve a dividir en banda de paso alta y baja aplicando un nuevo par de filtros pasa-banda. Esto se repite hasta completar todas las frecuencias de corte necesarias. De esta forma se obtiene el prototipo IIR de cada filtro que compone el banco de filtros. Una vez obtenido esto, se crean respuestas de tipo FIR para cada filtro a través de pasar un impulso ideal por cada filtro. En la figura 40 se puede observar la respuesta en frecuencia del banco de filtros. Un detalle importante a considerar es que la suma del efecto de todos los filtros es unitaria para todo el rango frecuencial analizado, siendo esta una característica necesaria para poder descomponer la señal en bandas y luego re-componer la señal sumando las bandas sin generar ningún tipo de distorsión en el proceso. Para lograr esto, los coeficientes de los filtros deben ser elevados al cuadrado (lo que equivale poner dos filtros en cascada) para lograr que en las intersecciones entre los filtros la suma de amplitud sea unitaria (en la frecuencia de corte se genera una caída de 6 dB en lugar de los 3 dB que presentaria un filtro Butterworth simple).

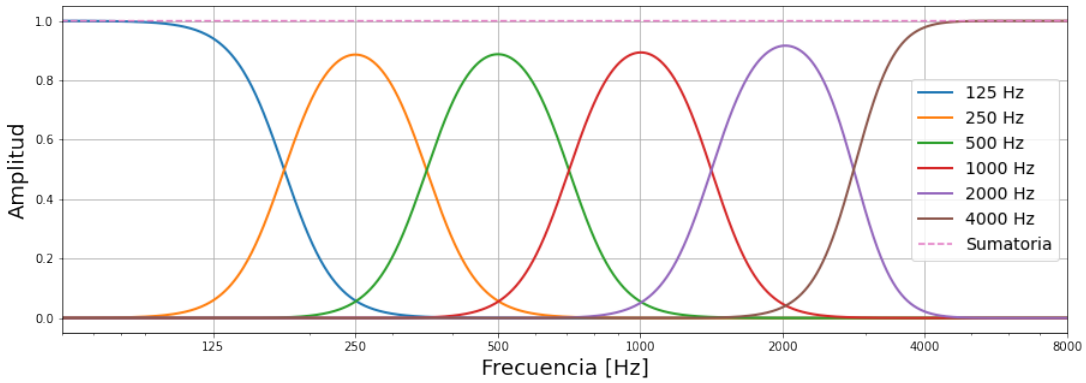


Figura 40: Banco de filtros Butterworth

A partir de aplicar este banco de filtros, la señal de entrada se descompone en 6 sub-bandas como se muestra en la figura 41. En este gráfico se puede apreciar como la pendiente de caída varia según la banda de frecuencia que se observa. Todos los procesos subsiguientes se aplican de manera independiente para cada banda, y luego al final las bandas se suman para volver a tener una señal correspondiente al espectro completo original. De ahora en adelante, por una cuestion de simplicidad se muestran los gráficos pertenecientes a la banda de $1000Hz$ de manera ilustrativa.

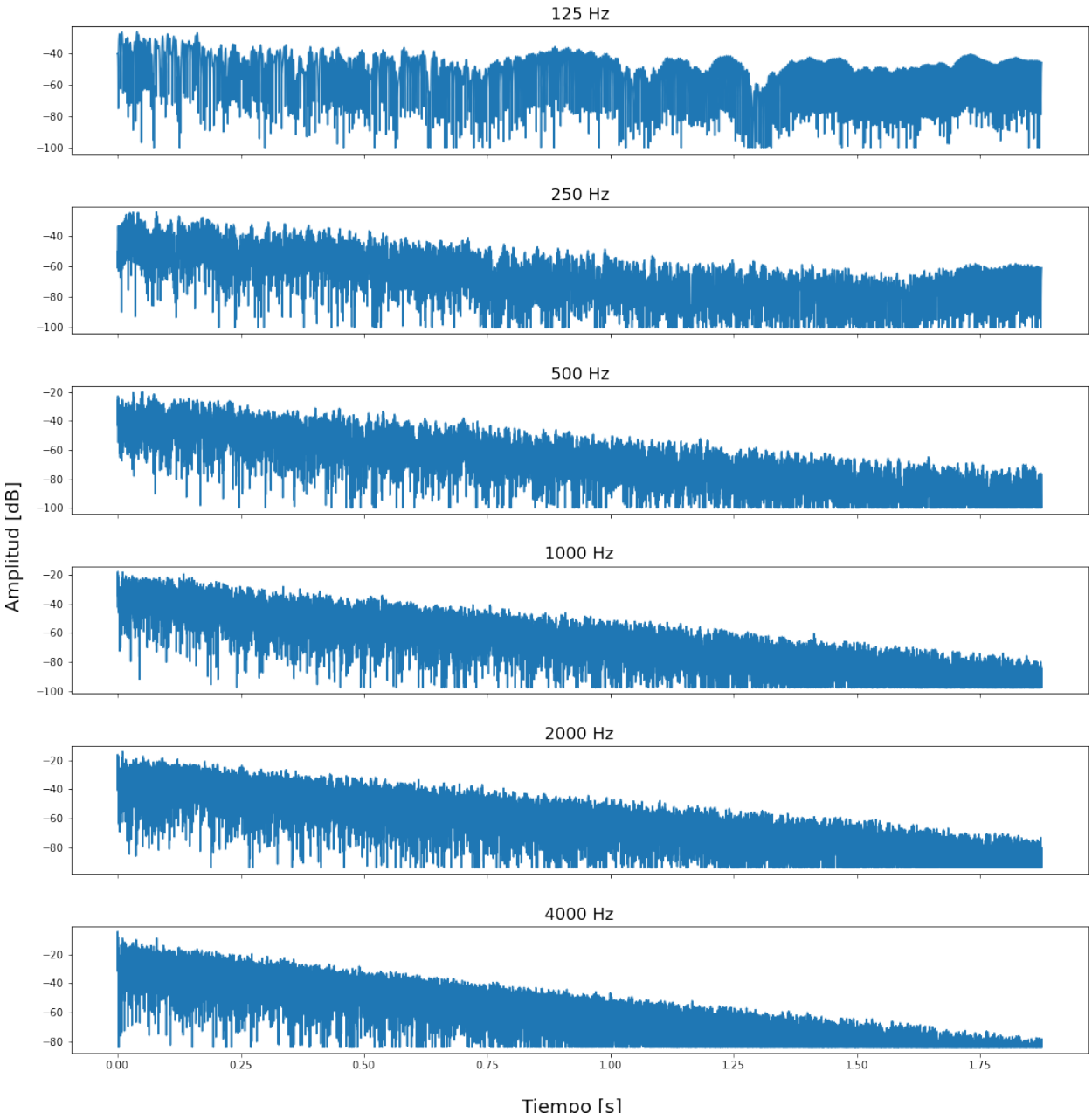


Figura 41: Sub-bandas obtenidas luego de aplicar el banco de filtros

El paso siguiente consiste en determinar el piso de ruido de la señal. Detectar el piso de ruido permite asegurar que el método de aumentación no amplifique ruido cuando se busca obtener un tiempo de reverberación mayor al inicial propio de la respuesta al impulso de entrada. Para determinar el punto donde predomina el ruido en la respuesta al impulso se utiliza el método iterativo de Lundeby [57]. El mismo consta de los siguientes pasos:

1. La respuesta al impulso al cuadrado es promediada en intervalos de tiempo locales de entre $10\ ms$ y $50\ ms$ para obtener una curva 'suavizada', es decir, disminuir las variaciones instantaneas sin perder las pendientes cortas.

2. Se hace una primera estimacion del piso de ruido. Para hacerlo se toma el segmento correspondiente al último $10\,\%$ de la respuesta al impulso.

3. La pendiente de caída se estima aplicando una regresión lineal sobre el intervalo de tiempo que contiene la respuesta entre el pico de $0\ dB$ y el primer intervalo $5 - 10\ dB$ por encima del ruido de fondo.

4. Se determina un punto de cruce provisorio en la intersección entre la pendiente de caída estimada y el nivel de piso de ruido.

5. Se obtiene un nuevo intervalo de tiempo de acuerdo a la pendiente calculada, de manera que haya entre 3 y 10 intervalos por cada $10\ dB$ de caída

6. Se vuelve a promediar localmente el impulso al cuadrado de acuerdo al nuevo intervalo temporal calculado previamente

7. Se estima el ruido de fondo nuevamente. El segmento a evaluar debe corresponder a $5 - 10\ dB$ luego del punto de cruce (siguiendo la curva estimada previamente), o bien, un minimo del $10\,\%$ de la señal total (en el caso de tener que optar por el $10\,\%$ de nuevo, el resultado seria el mismo que antes, y el punto encontrado previamente seria el definitivo).

8. Se estima la pendiente de caída para un rango dinámico de entre $20\ dB$ y $10\ dB$, empezando desde un punto $5 - 10\ dB$ por encima del nivel de ruido.

9. Se encuentra un nuevo punto de cruce.

10. Los pasos 7-9 se repiten hasta que el valor del piso de ruido converja, tolerando un maximo de 6 iteraciones.

El paso siguiente es estimar la pendiente paramétrica que mejor se aproxime a la pendiente de caída real. La estimación se basa en el modelo de la ecuación 13. Por lo tanto, los parámetros que se busca estimar son la amplitud inicial, la tasa de caída y el nivel de piso de ruido. La estimación se realiza aplicando un algoritmo de ajuste no lineal por cuadrados mínimos. El resultado de la estimación para la banda de $1000Hz$ se muestra en la figura 42.

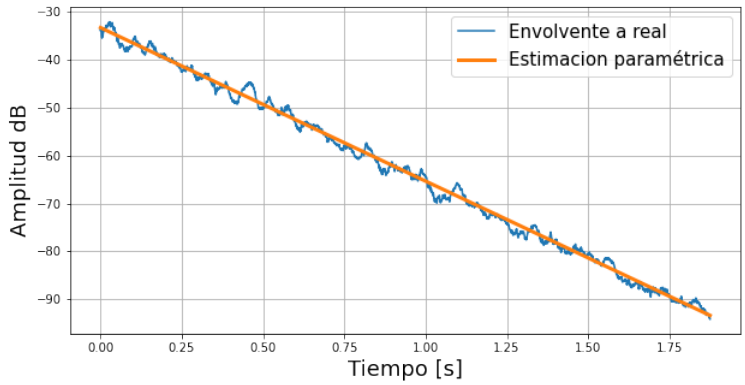


Figura 42: Estimación paramétrica de la pendiente de caída.

Con estos parámetros estimados se genera una nueva envolvente de caída pero llevando el nivel de piso de ruido a cero, y se aplica esta envolvente sobre una señal de ruido Gaussiano de media cero y desvío estándar unitario. Con esta señal sintética y la señal original se hace una transición cruzada en el punto estimado del piso de ruido. De esta manera se elimina el ruido de la señal original, reemplazádolo por la caída exponencial determinada por la envolvente paramétrica. En la figura 43 se puede observar como se extiende la respuesta luego del punto de inicio del piso de ruido de la señal original.

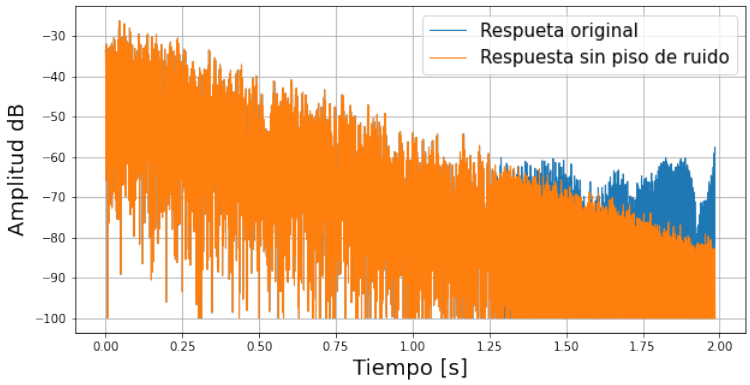


Figura 43: Respuesta original y extendida sin piso de ruido.

Finalmente, teniendo las bandas extendidas y habiéndose eliminado el piso de ruido, se prosigue con el proceso de aumentación del tiempo de reverberación para cada sub-banda, multiplicando cada respuesta por la correspondiente envolvente exponencial creciente o decreciente según corresponda, para obtener la envolvente de caída necesaria para generar el tiempo de reverberación deseado, como se indica en la ecuación 15. Un ejemplo del resultado de la aumentación sobre una sub-banda de frecuencia se muestra en la figura 44, en donde el tiempo de reverberación objetivo es menor que el tiempo de reverberación de la respuesta original, por lo cual la pendiente aumentada resulta mas atenuada que la pendiente original.

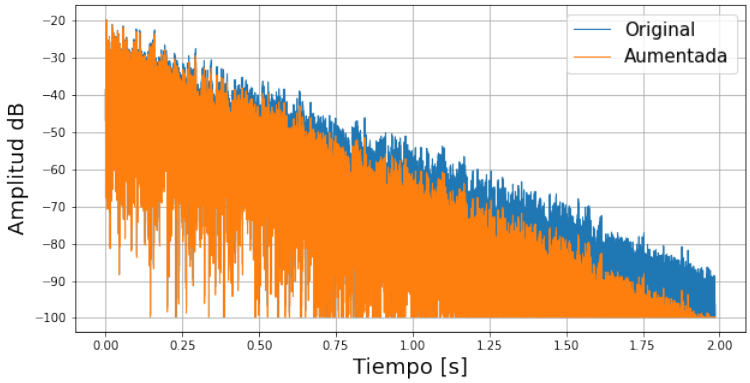


Figura 44: Aumentación del tiempo de reverberación alterando la envolvente de caída original.

Una vez realizada la alteración de la envolvente de caída para todas las bandas frecuenciales, estas se suman para conformar nuevamente la parte tardía de la respuesta al impulso de espectro completo. El paso final consiste en concatenar los segmentos de la respuesta al impul-

so original que no fueron alterados, es decir, el delay del camino directo y la parte temprana de la respuesta. Con esto, el proceso de aumentación termina, y se obtiene como resultado una nueva respuesta al impulso con el tiempo de reverberación deseado. En la figura 45 se muestra la respuesta al impulso original comparada con la nueva respuesta generada a partir del proceso anteriormente descrito.

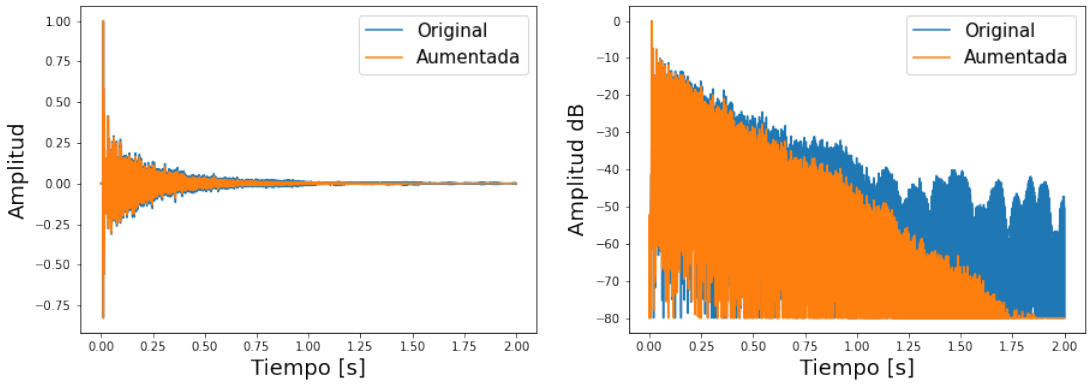


Figura 45: Resultado del proceso de aumentación del tiempo de reverberación de una respuesta al impulso