



**UNIVERSIDAD DE PAMPLONA  
FACULTAD DE INGENIERÍAS Y ARQUITECTURA  
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, SISTEMAS Y  
TELECOMUNICACIONES  
PROGRAMA DE INGENIERÍA ELECTRÓNICA**

**TÍTULO.**  
**DETECCIÓN Y CLASIFICACIÓN DE ENFERMEDADES RESPIRATORIAS  
MEDIANTE SONIDO PULMONAR APLICANDO INTELIGENCIA ARTIFICIAL**

**AUTOR:**  
**JUAN MANUEL ORTEGA CADRAZCO**

**DIRECTOR:**  
**MSc. YESID EUGENIO SANTAFE RAMON**  
**MSc. LUIS ENRIQUE MENDOZA**

**UNIVERSIDAD DE PAMPLONA**  
**FACULTAD DE INGENIERÍAS Y ARQUITECTURA**  
**DEPARTAMENTO DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA,**  
**SISTEMAS Y TELECOMUNICACIONES**  
**PROGRAMA DE INGENIERÍA ELECTRÓNICA**



**UNIVERSIDAD DE PAMPLONA**  
**PAMPLONA N. DE S. – COLOMBIA**  
**18 de noviembre de 2021**

**TÍTULO.**  
**DETECCIÓN Y CLASIFICACIÓN DE ENFERMEDADES RESPIRATORIAS  
MEDIANTE SONIDO PULMONAR APLICANDO INTELIGENCIA ARTIFICIAL**

**AUTOR:**  
**JUAN MANUEL ORTEGA CADRAZCO**

**TRABAJO DE GRADO PARA OPTAR POR EL TÍTULO DE INGENIERO  
ELECTRÓNICO**

**DIRECTOR:**  
**MSc. YESID EUGENIO SANTAFE RAMON**  
**MSc. LUIS ENRIQUE MENDOZA**

**UNIVERSIDAD DE PAMPLONA**  
**FACULTAD DE INGENIERÍAS Y ARQUITECTURA**  
**DEPARTAMENTO DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA,**  
**SISTEMAS Y TELECOMUNICACIONES**  
**PROGRAMA DE INGENIERÍA ELECTRÓNICA**  
**PAMPLONA N. DE S. – COLOMBIA**  
**18 de noviembre de 2021**

**UNIVERSIDAD DE PAMPLONA  
FACULTAD DE INGENIERÍAS Y ARQUITECTURA  
DEPARTAMENTO DE INGENIERÍAS ELÉCTRICA ELECTRÓNICA SISTEMAS  
Y TELECOMUNICACIONES  
PROGRAMA DE INGENIERÍA ELECTRÓNICA**

**AUTORIZACIÓN PARA SUSTENTAR  
TRABAJO PRESENTADO PARA OPTAR POR EL TÍTULO DE  
INGENIERO ELECTRÓNICO**

**TÍTULO**

**FECHA DE INICIO DEL TRABAJO:**

**FECHA DE TERMINACIÓN DEL TRABAJO:**

**NOMBRES Y FIRMAS DE AUTORIZACIÓN PARA SUSTENTAR**

**AUTOR:** \_\_\_\_\_

**DIRECTOR:** \_\_\_\_\_

**DIRECTOR DE PROGRAMA:** \_\_\_\_\_

**JURADO CALIFICADOR:**

**PRESIDENTE: ING.** \_\_\_\_\_

**OPONENTE: ING** \_\_\_\_\_

**SECRETARIO: ING.** \_\_\_\_\_

**PAMPLONA, COLOMBIA**

**FECHA**

## DEDICATORIA

*Dedicado a mis padres y hermanos, como también a familiares y amigos que estuvieron presentes en este proceso de superación personal y profesional para alcanzar nuevas metas y sumar peldaños que engrandecen la divinidad en nuestro paso por la vida.*

## **AGRADECIMIENTOS**

*Mis agradecimientos a Dios y a todas las personas e instituciones que de una u otra manera aportaron en este proceso de formación personal y profesional que hoy podemos alcanzar como un nuevo logro para continuar en el camino infinito del conocimiento.*

## CONTENIDO

	Pág.
1. RESUMEN .....	12
2. INTRODUCCIÓN .....	13
3. OBJETIVOS .....	60
3.1. OBJETIVO GENERAL .....	60
3.2. OBJETIVOS ESPECÍFICOS .....	60
4. PLANTEAMIENTO DEL PROBLEMA .....	61
4.1. DEFINICIÓN DEL PROBLEMA .....	61
4.2 JUSTIFICACIÓN DEL PROBLEMA .....	61
5. ESTADO DEL ARTE MARCO DE REFERENCIA .....	62
6. RESULTADOS .....	70
7. CONCLUSIONES Y RECOMENDACIONES .....	86
8. BIBLIOGRAFÍA .....	87
9. ANEXOS .....	91

## LISTA DE TABLAS

	Pág.
Tabla 1. Características y propiedades Wavelets .....	41
Tabla 2. Niveles banco de filtro wavelet .....	72
Tabla 3. Prueba filtro Wavelet .....	81
Tabla 4. Prueba espectrograma wavelet .....	82
Tabla 5. Prueba red neuronal convolucional .....	82

## LISTA DE FIGURAS

	Pág.
Figura 1. Tracto respiratorio superior .....	16
Figura 2. Tracto respiratorio medio superior .....	21
Figura 3. Tracto respiratorio inferior .....	26
Figura 4. Bronquitis aguda .....	27
Figura 5. Asma .....	28
Figura 6. Neumonía .....	29
Figura 7. Edema pulmonar .....	30
Figura 8. Función de escala (db4) asociada a la función wavelet (db4) .....	41
Figura 9. Estructura método de convolución .....	42
Figura 10. Esquema de descomposición multi resolución .....	43
Figura 11. Espectrograma Wavelet .....	44
Figura 12. Red neuronal con dos capas ocultas .....	48
Figura 13. Estructura Perceptrón .....	54
Figura 14. Estructura red convolucional .....	59
Figura 15. Descomposición frecuencial Wavelet .....	73
Figura 16. Asma por sibilancia .....	74
Figura 17. Bronquitis por crepitantes .....	74

Figura 18. EPOC por sibilancia .....	75
Figura 19. Neumonía por crepitantes .....	75
Figura 20. Murmullo pulmonar normal .....	76
Figura 21. Fibrosis pulmonar por crepitantes .....	76
Figura 22. Efusión pleural por frote pleural .....	77
Figura 23. Insuficiencia cardiaca por crepitantes .....	77
Figura 24. Espectrograma wavelet .....	78
Figura 25. Diagrama de bloques .....	80
Figura 26. Categorías entrenamiento .....	83
Figura 27. Gráfica de entrenamiento y validación base de datos .....	84
Figura 28. Gráfica de entrenamiento y validación muestras .....	85

## 1. RESUMEN

Las enfermedades pulmonares son consideradas actualmente un agravante en aumento a la salud humana a nivel mundial. Debido a la alerta de pandemia emitida por la Organización Mundial de la Salud (OMS), y el aumento de cuadros clínicos respiratorios a nivel nacional se hace evidente el atraso tecnológico en el campo médico nacional e internacional para el análisis y estudio de sonidos respiratorios en pacientes, teniendo como método principal a nivel nacional la auscultación de tórax por medio de estetoscopio análogo con un margen de análisis y diagnóstico que pone en riesgo la salud pública.

La inteligencia artificial es un método que puede generar mejores análisis y diagnóstico de enfermedades respiratorias por medio de sonido pulmonar, y para ello se hará uso de herramientas matemáticas de cómputo para estudiar una base de datos de sonidos respiratorios de acceso abierto para analizar y clasificar estos sonidos en enfermedades y obtener una base de datos computarizada apta para la identificación y clasificación de sonidos respiratorios.

Los sonidos respiratorios de la base de datos adquirida serán estudiados frecuentemente con herramientas matemáticas de cómputo como Wavelet para separar su espectro en niveles frecuenciales y buscar características de frecuencia y tiempo que evidencien una anomalía respiratoria como una sibilancia, roncus, crepitantes, frote pleural entre otros o la ausencia de estos.

Cuando se hallan identificado las anomalías frecuenciales y temporales en la forma de onda de los sonidos respiratorios se procederá a clasificarlos y etiquetarlos con las enfermedades asociadas a dichas características, para luego crear imágenes de tiempo y frecuencia conocidas como espectrogramas de wavelet e ingresarlas a una red neuronal convolucional (CNN) la cual generará una clasificación de categorías o clases de enfermedades respiratorias y dará una validación en un porcentaje cercano al 100% por medio del entrenamiento y aprendizaje de la red neuronal, y con ello tener una herramienta de identificación y clasificación de sonidos respiratorios computarizada para aportar al estudio y la modernización de los diagnósticos de las enfermedades pulmonares.

## 2. INTRODUCCIÓN

### SISTEMA RESPIRATORIO: ANATOMÍA

#### DEFINICIÓN DEL SISTEMA RESPIRATORIO

“El sistema respiratorio está formado por las estructuras que realizan el intercambio de gases entre la atmósfera y la sangre. El oxígeno (O<sub>2</sub>) es introducido dentro del cuerpo para su posterior distribución a los tejidos y el dióxido de carbono (CO<sub>2</sub>) producido por el metabolismo celular, es eliminado al exterior. Además, interviene en la regulación del pH corporal, en la protección contra los agentes patógenos y las sustancias irritantes que son inhalados y en la vocalización, ya que, al moverse el aire a través de las cuerdas vocales, produce vibraciones que son utilizadas para hablar, cantar, gritar.....

El proceso de intercambio de O<sub>2</sub> y CO<sub>2</sub> entre la sangre y la atmósfera, recibe el nombre de **respiración externa**.

El proceso de intercambio de gases entre la sangre de los capilares y las células de los tejidos en donde se localizan esos capilares se llama **respiración interna**”.

[1]

#### TRACTO RESPIRATORIO SUPERIOR

##### **Nariz y fosas nasales**

“La nariz es la parte superior del sistema respiratorio y varía en tamaño y forma en diferentes personas. Se proyecta hacia adelante desde la cara, a la que está unida su raíz, por debajo de la frente, y su dorso se extiende desde la raíz hasta el vértice o punta.

La parte superior de la nariz es ósea, se llama **punto de la nariz** y está compuesto por los huesos nasales, parte del maxilar superior y la parte nasal del hueso frontal.

La parte inferior de la nariz es cartilaginosa y se compone de cartílagos hialinos: 5 principales y otros más pequeños.

En el interior de la nariz se encuentra el **tabique nasal** que es parcialmente óseo y parcialmente cartilaginoso y divide a la cavidad nasal en dos partes llamadas las **fosas nasales**. La parte ósea del tabique está formada por parte del hueso

etmoides y por el vómer y se localiza en el plano medio de las fosas nasales hasta el 7º año de vida. Después suele abombarse hacia uno de los lados, generalmente el derecho. La parte cartilaginosa está formada por cartílago hialino y se llama cartílago septal.

Las fosas nasales se abren al exterior por dos aberturas llamadas los **orificios o ventanas nasales**, limitados por fuera por las alas de la nariz, y se comunican con la nasofaringe por dos orificios posteriores o **coanas**. En cada fosa nasal se distingue un techo, una pared medial, una pared lateral y un suelo. El techo es curvado y estrecho y está formado por 3 huesos: frontal, etmoidal y esfenoidal.

El suelo es más ancho que el techo y está formado por parte de los huesos maxilar y palatino”. [1]

“La pared interna está formada por el tabique nasal óseo y es lisa. La pared externa es rugosa debido a la presencia de 3 elevaciones óseas longitudinales: los **cornetes nasales** superior, medio e inferior que se proyectan hacia el interior de cada fosa nasal y se curvan hacia abajo formando canales de paso de aire que se llaman **meatos**. Debajo del cornete superior se encuentra el **meato superior** en donde desembocan los senos etmoidales. Debajo del cornete medio se encuentra el **meato medio** en donde desembocan los senos maxilar y frontal. Debajo del cornete inferior se encuentra el **meato inferior**, en donde desemboca el conducto lácrimo-nasal”. [1]

“Las fosas nasales en su parte más exterior están recubiertas por piel que contiene un cierto número de gruesos pelos cortos o vibrisas y en su parte restante, por una membrana mucosa con epitelio pseudoestratificado columnar ciliado. Las vibrisas atrapan las partículas más grandes suspendidas en el aire inspirado antes de que alcancen la mucosa nasal, mientras que el resto de partículas es atrapado por una fina capa de moco segregada por las glándulas mucosas del epitelio, que luego es propulsado por los cilios hacia la faringe para ser deglutido e inactivado en el estómago. Además, el aire inspirado al pasar por la mucosa nasal es humedecido y calentado antes de seguir su camino por las vías respiratorias.

El 1/3 superior de la mucosa nasal, situada en el techo y la zona superior de las paredes interna y externa de las fosas nasales, es la **mucosa olfatoria**, ya que contiene los receptores sensitivos olfatorios”. [1]

## **Senos paranasales**

“Los senos paranasales son cavidades llenas de aire, de diferente tamaño y forma según las personas, que se originan al introducirse la mucosa de la cavidad nasal en los huesos del cráneo contiguos y, por tanto, están tapizadas por mucosa nasal, aunque más delgada y con menos vasos sanguíneos que la que recubre las fosas nasales. Los huesos que poseen cavidades aéreas son el frontal, el

etmoides, el esfenoides y el maxilar superior. En el recién nacido, la mayoría de senos son rudimentarios o están ausentes y durante la infancia y la adolescencia crecen e invaden los huesos adyacentes. El crecimiento de los senos es importante porque altera el tamaño y la forma de la cara y da resonancia a la voz. El moco secretado por las glándulas de la mucosa que los tapiza, pasa a las fosas nasales a través de los meatos”. [1]

**Senos frontales.** “Se localizan entre las tablas interna y externa del hueso frontal, por detrás de los arcos superciliares y a partir de los 7 años ya pueden ser visualizados en radiografías. Aunque es posible encontrar numerosos senos frontales, lo habitual es que haya uno derecho y otro izquierdo, que rara vez son de igual tamaño en una misma persona ya que el tabique que los separa no suele encontrarse en el plano medio. El tamaño de los senos frontales varía desde unos 5 mm hasta grandes espacios que se extienden lateralmente. Cada seno frontal comunica con la fosa nasal correspondiente a través del meato medio”. [1]

**Senos etmoidales.** “El número de cavidades aéreas en el hueso etmoides varía de 3-18 y no suelen ser visibles radiológicamente hasta los 2 años de edad. Desembocan en las fosas nasales por los meatos superiores”. [1]

**Senos esfenoidales.** “Suelen ser 2, se sitúan en el hueso esfenoides, por detrás de la parte superior de las fosas nasales, están separados entre sí por un tabique óseo que habitualmente no se encuentra en el plano medio y están en relación con estructuras anatómicas importantes como son los nervios ópticos, el quiasma óptico, la hipófisis, las arterias carótidas internas y los senos cavernosos. A diferencia de los otros senos éstos desembocan en las fosas nasales por encima de los cornetes superiores”. [1]

**Senos maxilares.** “Son los senos paranasales más grandes y su techo es el suelo de la órbita. En el momento del nacimiento son muy pequeños, pero luego crecen lentamente hasta el momento en que salen los dientes permanentes. Desembocan en la fosa nasal correspondiente por el meato medio a través de un orificio situado en la parte superior-interna del seno, de modo que es imposible su drenaje cuando la cabeza está en posición vertical, motivo por el que se requieren maniobras especiales”. [1]

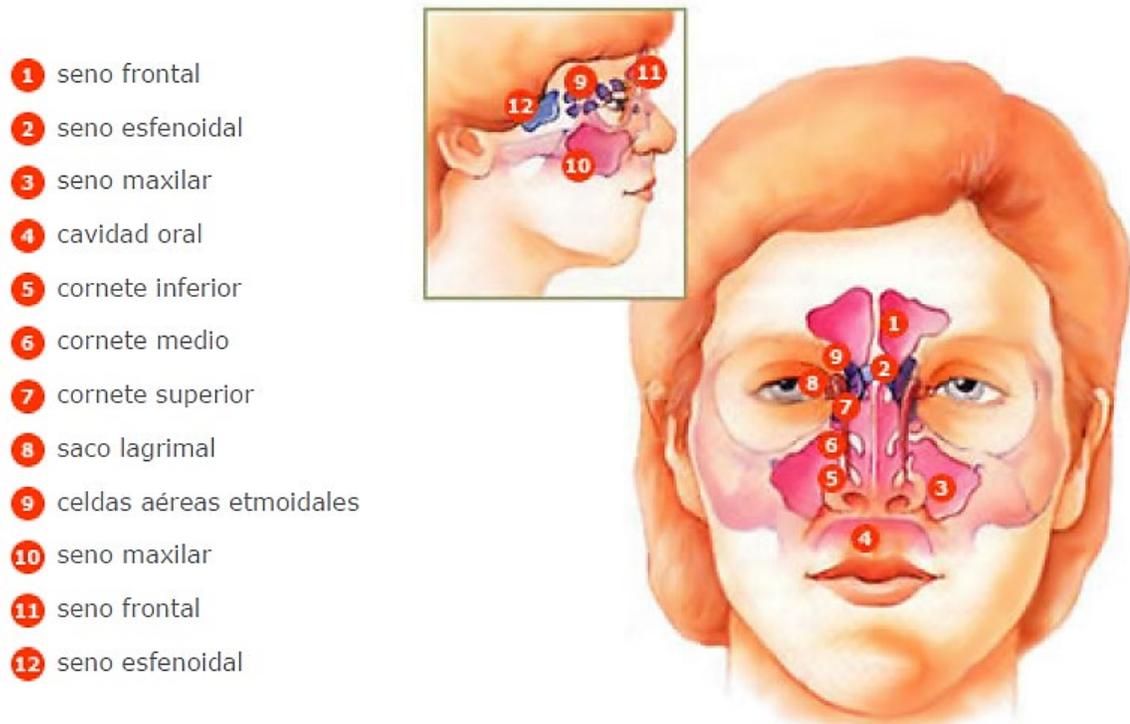


Figura 1. Tracto respiratorio superior [1]

## Boca

“La boca es la primera parte del tubo digestivo, aunque también se emplea para respirar. Está tapizada por una membrana mucosa, la mucosa oral, con epitelio estratificado escamoso no queratinizado y limitada por las mejillas y los labios. El espacio en forma de herradura situado entre los dientes y los labios, se llama **vestíbulo** y el espacio situado por detrás de los dientes es la **cavidad oral** propiamente dicha. El techo de la cavidad oral está formado por el **paladar** que consiste en dos partes: una ósea llamada **paladar duro**, formada por parte de los huesos maxilar superior y palatinos y otra, formada por músculos pares recubiertos de mucosa, llamada el **paladar blando o velo del paladar**, que se inserta por delante en el paladar duro y, por detrás es libre y presenta una proyección cónica en la línea media, la **úvula**”. [1]

“A cada lado del paladar blando hay dos músculos recubiertos de repliegues verticales de mucosa que constituyen los dos **pilares anteriores** y los dos **pilares posteriores** del paladar y forman el **istmo de las fauces** o puerta de comunicación de la cavidad oral con la parte oral de la faringe u orofaringe. Por su

parte anterior la cavidad oral se comunica con el exterior por la abertura de la boca”. [1]

## **Faringe**

“La faringe es un tubo que continúa a la boca y constituye el extremo superior común de los tubos respiratorio y digestivo. En su parte superior desembocan los orificios posteriores de las fosas nasales o coanas, en su parte media desemboca el istmo de las fauces o puerta de comunicación con la cavidad oral y por su parte inferior se continúa con el esófago, de modo que conduce alimentos hacia el esófago y aire hacia la laringe y los pulmones. Para una mejor descripción se divide en 3 partes: nasofaringe, situada por detrás de la nariz y por encima del paladar blando, orofaringe, situada por detrás de la boca, y laringofaringe, situada por detrás de la laringe. Debido a que la vía para los alimentos y el aire es común en la faringe, algunas veces la comida pasa a la laringe produciendo tos y sensación de ahogo y otras veces el aire entra en el tubo digestivo acumulándose gas en el estómago y provocando eructos”. [1]

**Nasofaringe.** “Se la considera la parte nasal de la faringe ya que es una extensión hacia atrás de las fosas nasales, está recubierta de una mucosa similar a la mucosa nasal y tiene una función respiratoria. Hay varias colecciones de tejido linfóide llamadas amígdalas, así, en su techo y pared posterior la **amígdala faríngea** (llamada popularmente vegetaciones o adenoides). En su pared externa, desemboca la **trompa de Eustaquio** que es la comunicación entre el oído medio y la nasofaringe y por detrás de cada uno de los orificios de desembocadura se encuentran las dos **amígdalas tubáricas**. La infección de unos adenoides puede diseminarse a una amígdala tubárica por proximidad, produciendo el cierre de la trompa correspondiente y una infección en la cavidad timpánica, lo que dará lugar a una otitis media con el peligro consiguiente de pérdida de audición temporal o permanente”. [1]

**Orofaringe.** “Es la parte oral de la faringe y tiene una función digestiva ya que es continuación de la boca a través del istmo de las fauces y está tapizada por una mucosa similar a la mucosa oral. La orofaringe está limitada por arriba por el paladar blando, por abajo por la base de la lengua, en donde se encuentra una colección de tejido linfóide llamada **amígdala lingual**, y por los lados por los pilares del paladar anteriores y posteriores, entre los cuales, en cada lado, se encuentra otra colección de tejido linfóide que constituye las **amígdalas palatinas**

(que cuando se infectan son llamadas popularmente anginas) cuya parte visible no es una guía exacta de su tamaño real porque una gran porción de ellas puede estar oculta por detrás de la lengua”. [1]

“Las amígdalas palatinas, lingual y faríngea constituyen una banda circular de tejido linfoide situada en el istmo de las fauces llamada **anillo amigdalino o anillo de Waldeyer** que tiene la misión fundamental de evitar la diseminación de las infecciones desde las cavidades nasal y oral hacia los tubos respiratorio y gastrointestinal”. [1]

**Laringofaringe.** “Es la parte laríngea de la faringe ya que se encuentra por detrás de la laringe. Está tapizada por una membrana mucosa con epitelio plano estratificado no queratinizado y se continúa con el esófago. Por su parte posterior se relaciona con los cuerpos de las vértebras cervicales 4ª a 6ª”. [1]

## **Laringe**

“Es un órgano especializado que se encarga de la **fonación** o emisión de sonidos con la ayuda de las cuerdas vocales, situadas en su interior. Está localizada entre la laringofaringe y la tráquea y es una parte esencial de las vías aéreas ya que actúa como una válvula que impide que los alimentos deglutidos y los cuerpos extraños entren en las vías respiratorias. Está tapizada por una membrana mucosa con epitelio estratificado escamoso no queratinizado y su esqueleto está formado por 9 cartílagos unidos entre sí por diversos ligamentos. Tres cartílagos son impares: el tiroides, el cricoides y la epiglotis y tres cartílagos son pares: los aritenoides, los corniculados y los cuneiformes”. [1]

**Cartílago tiroides.** “Es el más grande de los cartílagos laríngeos y está compuesto por 2 láminas cuadriláteras de cartílago hialino que se fusionan por delante en la línea media, formando la prominencia laríngea o nuez de Adán que es más marcada en los hombres porque el ángulo de unión de las láminas es mayor que en las mujeres. Por su borde superior se une al hueso hioides. El borde posterior de cada lámina se proyecta hacia arriba como cuerno superior y hacia abajo como cuerno inferior; los cuernos inferiores se articulan con el cartílago cricoides”. [1]

**Cartílago cricoides.** “Es el más inferior de los cartílagos laríngeos y tiene la forma de un anillo de sello con el sello dirigido hacia atrás. Está formado por

cartílago hialino y es más pequeño que el cartílago tiroides pero más grueso y fuerte. Su borde superior se articula con el cartílago tiroides y su borde inferior con el primer anillo de la tráquea”. [1]

**Cartílago epiglotis.** “Tiene forma de raqueta, está formado por cartílago elástico y situado por detrás de la raíz de la lengua y del hueso hioides y por delante del orificio de entrada a la laringe. Su borde superior es libre y forma el borde superior del orificio laríngeo y su borde inferior está unido al cartílago tiroides”. [1]

**Cartílagos aritenoides.** “Son 2, están formados por cartílago hialino y se articulan con el cartílago cricoides. En cada uno de ellos se inserta un ligamento que forma parte de una cuerda vocal”. [1]

**Cartílagos corniculados y cuneiformes.** “También son cartílagos pares y están formados por cartílago elástico. Los cartílagos corniculados están unidos a los vértices de las aritenoides y son como una prolongación de éstos y los cartílagos cuneiformes se encuentran en los pliegues de unión de las aritenoides y la epiglotis. Estos cartílagos se aproximan cuando se cierra el orificio de entrada a la laringe en el momento de deglutir”. [1]

### **Interior de la laringe**

“La cavidad o interior de la laringe se extiende desde el orificio de entrada a la laringe hasta el borde inferior del cartílago cricoides en donde se continúa con la tráquea, y queda dividida en 3 partes por dos pliegues superiores (o vestibulares o cuerdas vocales falsas) y dos pliegues inferiores (o cuerdas vocales verdaderas) que se proyectan hacia el interior de la laringe desde cada lado.

La parte de la cavidad laríngea situada por encima de los pliegues superiores se llama **vestíbulo laríngeo**, la situada entre los pliegues superiores y los inferiores se llama **ventrículo laríngeo** y la situada por debajo de los pliegues inferiores se llama **cavidad infraglótica**.

La mucosa laríngea está recubierta de epitelio estratificado escamoso no queratinizado hasta la cavidad infraglótica a partir de la cual se encuentra un epitelio pseudoestratificado columnar ciliado que ya se continúa con el de la mucosa de la tráquea”. [1]

“Los pliegues superiores o vestibulares o cuerdas vocales falsas están separados entre sí por la **hendidura vestibular** y los pliegues inferiores o cuerdas vocales verdaderas están separados entre sí por la **hendidura glótica**”. [1]

“La glotis incluye las cuerdas vocales verdaderas y la hendidura glótica y es, por tanto, la parte de la cavidad laríngea más directamente relacionada con la emisión de voz”. [1]

**Las cuerdas vocales falsas** “consisten en 2 espesos pliegues de mucosa que rodean a unos ligamentos y se extienden entre los cartílagos tiroideos y aritenoides. No tienen papel en la emisión de voz, sino que forman parte del mecanismo protector por el que la laringe se cierra en el momento de deglutir para evitar la entrada de alimentos u otros cuerpos extraños en las vías respiratorias.

**Las cuerdas vocales verdaderas** tienen forma de cuña con un vértice que se proyecta hacia el interior de la cavidad laríngea y una base que se apoya en el cartílago tiroideo. Cada cuerda vocal verdadera está compuesta por un ligamento, por una membrana elástica y por fibras de músculo estriado. Todo ello tapizado por una membrana mucosa con epitelio estratificado escamoso no queratinizado”. [1]

“La forma de la hendidura glótica variará según la posición de las cuerdas vocales. Mientras se respira tranquilamente la hendidura glótica se estrecha y presenta forma de cuña y, en cambio, se ensancha en la inspiración intensa. Al hablar, las cuerdas vocales se aproximan mucho de modo que la hendidura glótica aparece como una línea. Los cambios en el tono de voz se deben a variaciones en la tensión y en la longitud de las cuerdas vocales, en el ancho de la hendidura glótica y en la intensidad de los esfuerzos respiratorios, así, por ejemplo, los tonos bajos de la voz de los hombres se deben a la mayor longitud de sus cuerdas vocales”. [1]

## **Tráquea**

“Es un ancho tubo que continúa a la laringe y está tapizado por una mucosa con epitelio pseudoestratificado columnar ciliado. La luz o cavidad del tubo se mantiene abierta por medio de una serie de cartílagos hialinos (16-20) en forma de C con la parte abierta hacia atrás. Los extremos abiertos de los anillos cartilagosos quedan estabilizados por fibras musculares lisas y tejido conjuntivo elástico formando una superficie posterior plana en contacto directo con el esófago, por

delante del cual desciende, lo que permite acomodar dentro de la tráquea las expansiones del esófago producidas al tragar. Termina a nivel del ángulo esternal y de la apófisis espinosa de la 4ª vértebra torácica, al dividirse en los bronquios principales derecho e izquierdo. El arco o cayado de la aorta en un principio es anterior a la tráquea y luego se coloca en su lado izquierdo”. [1]

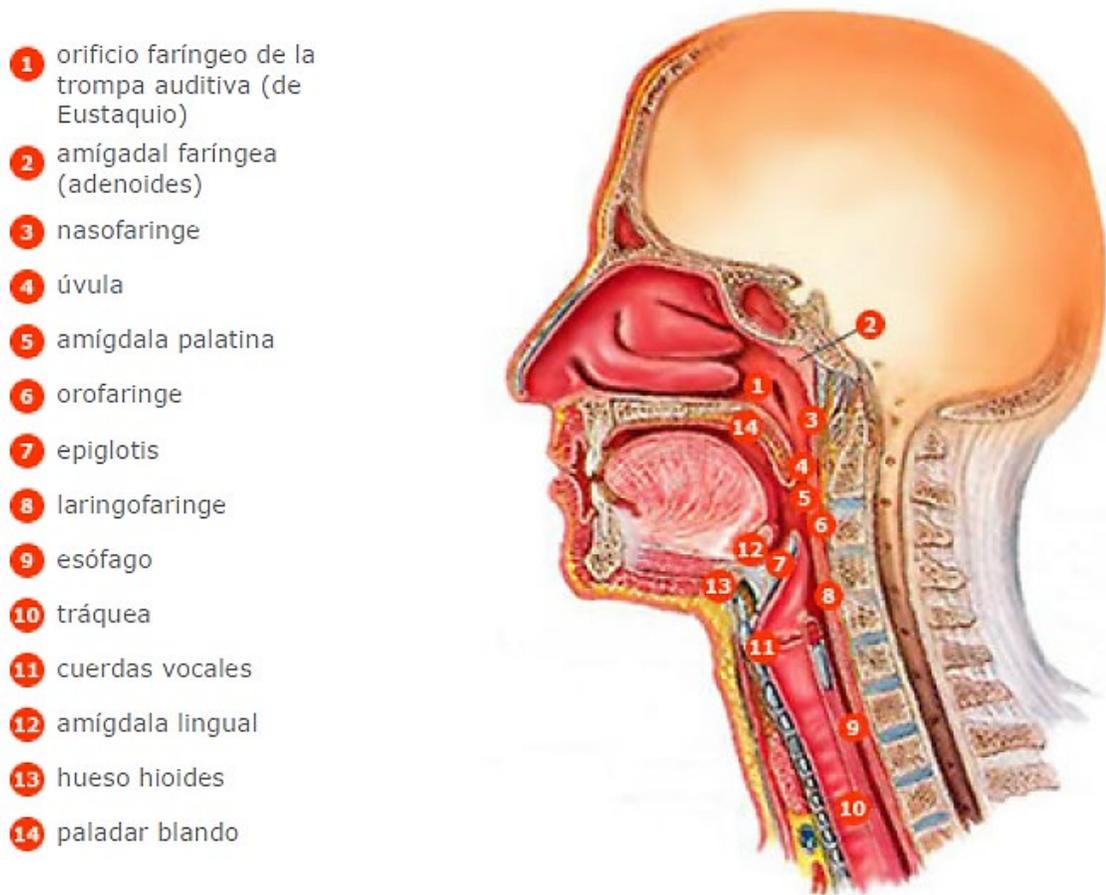


Figura 2. Tracto respiratorio medio superior [1]

## TRACTO RESPIRATORIO INFERIOR

### Bronquios

“Los **bronquios principales** son dos tubos formados por anillos completos de cartílago hialino, uno para cada pulmón, y se dirigen hacia abajo y afuera desde el final de la tráquea hasta los hilios pulmonares por donde penetran en los

pulmones. El bronquio principal derecho es más vertical, corto y ancho que el izquierdo lo que explica que sea más probable que un objeto aspirado entre en el bronquio principal derecho. Una vez dentro de los pulmones, los bronquios se dividen continuamente, de modo que cada rama corresponde a un sector definido del pulmón”. [1]

“Cada bronquio principal se divide en **bronquios lobulares** que son 2 en el lado izquierdo y 3 en el lado derecho, cada uno correspondiente a un lóbulo del pulmón. Cada bronquio lobular se divide, a su vez, en **bronquios segmentarios** que corresponden a los llamados segmentos pulmonares, cada uno de los cuales tiene sus propios bronquio, arteria y vena segmentarios. Los bronquios segmentarios, a su vez, se dividen en bronquios más pequeños o **bronquíolos** que se ramifican en tubos más pequeños, de un modo repetido hasta formar los **bronquíolos terminales**. Toda esta ramificación bronquial se parece a un árbol invertido y por ello se llama árbol bronquial”. [1]

“A medida que se produce la ramificación bronquial, el epitelio de la mucosa va cambiando. En los bronquios principales, lobulares y segmentarios la mucosa tiene epitelio pseudoestratificado columnar ciliado. En los bronquiolos más grandes pasa a tener epitelio columnar simple ciliado, en los bronquiolos más pequeños, epitelio cuboidal simple ciliado y en los bronquiolos terminales, epitelio cuboidal simple no ciliado. Además, los anillos cartilaginosos van desapareciendo y las fibras musculares lisas van aumentando, hasta que ya no hay cartílago y solo músculo liso en la pared de los bronquiolos más pequeños, de modo que la contracción muscular puede cerrar la cavidad de estos bronquiolos, impidiendo la entrada de aire en los alvéolos, como sucede por ejemplo en una crisis asmática, lo que puede ser una situación amenazadora para la vida”. [1]

## **Pulmones**

“Los pulmones son los órganos esenciales de la respiración. Son ligeros, blandos, esponjosos y muy elásticos y pueden reducirse a la 1/3 parte de su tamaño cuando se abre la cavidad torácica. Durante la primera etapa de la vida son de color rosado, pero al final son oscuros y moteados debido al acúmulo de partículas de polvo inhalado que queda atrapado en los fagocitos (macrófagos) de los pulmones a lo largo de los años”. [1]

“Cada pulmón tiene la forma de un semicono, está contenido dentro de su propio saco pleural en la cavidad torácica, y está separado uno del otro por el corazón y otras estructuras del mediastino. El pulmón derecho es mayor y más pesado que el izquierdo y su diámetro vertical es menor porque la cúpula derecha del diafragma es más alta, en cambio es más ancho que el izquierdo porque el corazón se abomba más hacia el lado izquierdo. El pulmón izquierdo está dividido en un lóbulo superior, que presenta la escotadura cardíaca en donde se sitúa el corazón, y un lóbulo inferior. El pulmón derecho está dividido en tres lóbulos: superior, medio e inferior”. [1]

“Cada pulmón presenta un vértice, una base y dos caras. El vértice es el polo superior redondeado de cada pulmón y se extiende a través de la abertura superior del tórax, por encima de la 1ª costilla. La base o cara diafragmática es cóncava y en forma de semiluna y se apoya en la superficie convexa del diafragma que separa al pulmón derecho del hígado y al pulmón izquierdo del hígado, estómago y bazo. La cara costal es grande, lisa y convexa y se adapta a la pared torácica y la cara interna tiene una parte vertebral que ocupa el canal a cada lado de la columna vertebral y otra mediastínica que presenta depresiones debido al corazón y los grandes vasos”. [1]

“El **hilio** de cada pulmón se encuentra cerca del centro de la cara interna, está rodeado por pleura y es la zona por donde pasan las estructuras que entran y salen de cada pulmón (arterias, venas, bronquios, nervios, vasos y ganglios linfáticos) formando los **pedículos pulmonares** que también están rodeados por pleura. De este modo los pedículos unen la cara interna de cada pulmón al corazón y la tráquea”. [1]

“Las ramas de la arteria pulmonar distribuyen sangre venosa en los pulmones para que éstos la puedan oxigenar. Acompañan a los bronquios de tal modo que hay una rama para cada lóbulo, cada segmento bronco-pulmonar y cada área funcional del pulmón. Las ramas terminales de las arterias pulmonares se ramifican en capilares que se encuentran recubriendo las paredes de los alvéolos. Por su parte, las arterias bronquiales son pequeñas y transportan sangre oxigenada para irrigar los bronquios en todas sus ramificaciones”. [1]

“Las venas pulmonares recogen la sangre oxigenada desde los pulmones y la transportan a la aurícula izquierda del corazón.

Por su parte, las venas bronquiales recogen la sangre venosa procedente de los bronquios y la llevan a la vena ácigos (la derecha) y la vena hemiacigos (la izquierda)". [1]

## **Unidad respiratoria**

"Los bronquios se dividen una y otra vez hasta que su diámetro es inferior a 1 mm, después de lo cual se conocen como bronquiolos y ya no tienen en sus paredes ni glándulas mucosas ni cartílagos. Los bronquiolos se subdividen a su vez en bronquiolos terminales. Estos se subdividen hasta formar los **bronquiolos respiratorios** que se caracterizan porque en parte tienen estructura de bronquiolos, pero en parte ya tienen alvéolos en su pared que se abren directamente en su cavidad". [1]

"La unidad respiratoria es la zona del pulmón que está aireada por un bronquiolo respiratorio. Cada bronquiolo respiratorio se divide en varias vías llamadas conductos alveolares que, a su vez, se abren a numerosos sacos alveolares y alvéolos. Cada saco alveolar está formado por varios alvéolos y cada alvéolo es una bolsa redondeada, abierta, por un lado, con un diámetro medio de unas 300 micras, que tiene una pared extremadamente delicada formada por epitelio plano simple. En los 2 pulmones hay alrededor de unos 300 millones de alvéolos". [1]

## **ESTRUCTURAS ACCESORIAS**

### **Pleuras**

"Son membranas serosas, es decir que tapizan una cavidad corporal que no está abierta al exterior y recubren los órganos que se encuentran en su interior que, en este caso, son los pulmones. Una serosa consiste en una fina capa de tejido conjuntivo laxo cubierta por una capa de **epitelio escamoso simple** y como el tipo de epitelio es siempre el mismo en todas las serosas, se le da el nombre genérico de **mesotelio** al epitelio de una serosa". [1]

"Hay 2 pleuras en cada lado. Cada pulmón está cubierto completa e íntimamente por una membrana serosa, lisa y brillante llamada **pleura visceral**. La cavidad

torácica está cubierta por otra membrana serosa llamada **pleura parietal**. El espacio virtual que hay entre ambas pleuras se llama **cavidad pleural**. Las cavidades pleurales de cada lado son 2 espacios no comunicados entre sí y cerrados herméticamente en los que existe una capa muy fina de líquido seroso lubricante secretado por el mesotelio, el **líquido pleural**, cuya misión es reducir el roce entre las capas parietal y visceral de cada lado para que no haya interferencias con los movimientos respiratorios”. [1]

“La pleura parietal recubre las diferentes partes de la cavidad torácica y, con propósitos descriptivos, recibe un nombre según la zona que recubre: la pleura costal es la porción más fuerte de la pleura parietal y cubre las superficies internas de la caja torácica. La pleura mediastínica cubre el mediastino, la pleura diafragmática es delgada y cubre la superficie superior del diafragma y, por último, la cúpula pleural cubre el vértice del pulmón”. [1]

“Durante la respiración tranquila existen 3 zonas de las cavidades pleurales que no son ocupadas por los pulmones y en donde dos partes de pleura parietal contactan una con la otra por sus superficies internas. Estas zonas se llaman **senos pleurales** y se llenan en una inspiración profunda. Los **senos costodiafragmáticos derecho e izquierdo** están situados entre las pleuras costal y diafragmática a cada lado y se acortan y se agrandan alternativamente a medida que los pulmones se mueven dentro y fuera de ellos durante la inspiración y la espiración y el **seno costomediastínico** se encuentra a nivel de la escotadura cardíaca, en donde se ponen en contacto las partes costal y mediastínica de la pleura parietal izquierda”. [1]

## PARED TORÁCICA

### **Mediastino**

“La cavidad torácica presenta 3 divisiones principales que son las cavidades pleurales derecha e izquierda y el mediastino que es la estrecha parte media y, por tanto, está entre las dos cavidades pleurales. Se extiende desde el orificio superior del tórax hasta el diafragma y desde el esternón y los cartílagos costales hasta la superficie anterior de las 12 vértebras torácicas. Contiene el **corazón** y los **grandes vasos**, la **tráquea** y los **bronquios**, el **timo**, el **esófago**, los **nervios frénicos** y los **nervios vagos** (X par craneal), el **conducto torácico** y **ganglios**

**linfáticos.** Todas estas estructuras están rodeadas por tejido conectivo laxo y tejido adiposo cuya laxitud junto con la elasticidad de los pulmones permite al mediastino acomodarse al movimiento y cambios de volumen de la cavidad torácica”. [1]

**El timo** “es una masa de tejido linfoide de forma aplanada y lobular que se encuentra por detrás del manubrio esternal. En los recién nacidos puede extenderse a través de la abertura torácica superior hacia el cuello debido a su gran tamaño, pero a medida que el niño crece va disminuyendo hasta casi desaparecer en el adulto”. [1]

**El conducto torácico** “es el conducto linfático principal del organismo, con unos 45 cm de longitud, y transporta la mayor parte de linfa del cuerpo hasta desembocar en el sistema venoso, en la vena braquiocefálica izquierda”. [1]

- 1 laringe (cartílago tiroides y cartílago cricoides)
- 2 aorta
- 3 bronquio principal izquierdo
- 4 mediastino
- 5 cavidad pleural izquierda
- 6 cavidad pleural derecha
- 7 pulmón izquierdo
- 8 espacio pleural
- 9 diafragma
- 10 esófago
- 11 pleura visceral
- 12 pleura parietal
- 13 pulmón derecho
- 14 bronquio principal derecho
- 15 tráquea

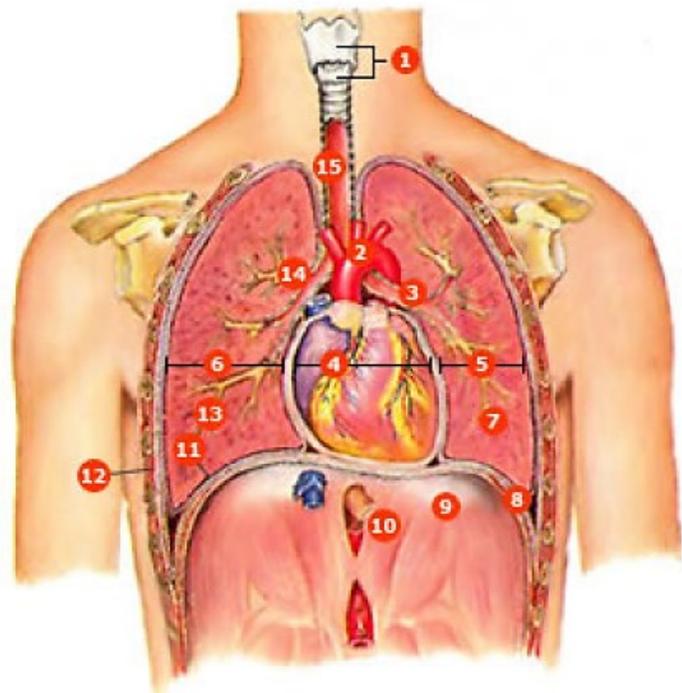


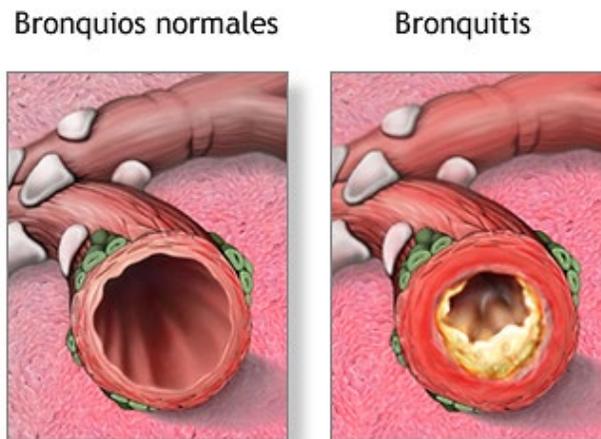
Figura 3. Tracto respiratorio inferior [1]

## ENFERMEDADES RELACIONADAS A LOS SONIDOS RESPIRATORIOS

Las causas de los ruidos respiratorios anormales pueden incluir:

### **Bronquitis aguda**

“Es la hinchazón e inflamación de las vías aéreas principales que llevan aire hacia los pulmones. Esta hinchazón estrecha las vías respiratorias, lo cual dificulta la respiración. Otro síntoma de bronquitis es tos y tos con flemas. Aguda significa que los síntomas solo han estado presentes por un período corto”. [2]



 ADAM.

Figura 4. Bronquitis aguda [2]

“Cuando se produce la bronquitis aguda, casi siempre viene después de haber tenido un resfriado o una enfermedad seudogripal. La infección de bronquitis es causada por un virus. En primer lugar, afecta a la nariz, los senos paranasales y la garganta. Luego, se propaga a las vías respiratorias que llevan a los pulmones.

Algunas veces, las bacterias también infectan las vías respiratorias. Esto es más común en personas con enfermedad pulmonar obstructiva crónica (EPOC)". [2]

## Asma

"Es una enfermedad crónica que provoca que las vías respiratorias de los pulmones se hinchen y se estrechen. Esto hace que se presente dificultad para respirar como sibilancias, falta de aliento, opresión en el pecho y tos". [2]

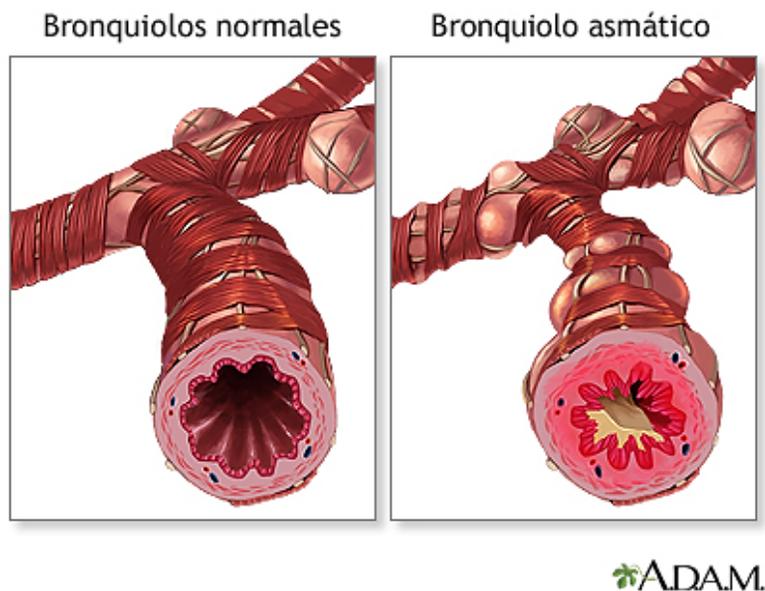


Figura 5. Asma [2]

"El asma es causada por hinchazón (inflamación) de las vías respiratorias. Cuando se presenta un ataque de asma, el recubrimiento de las vías respiratorias se inflama y los músculos que las rodean se tensionan. Esto reduce la cantidad de aire que puede pasar por estas.

Los síntomas de asma pueden ser provocados por la inhalación de sustancias llamadas alérgenos o desencadenantes, o por otras causas". [2]

## Enfermedad pulmonar intersticial

"La enfermedad pulmonar intersticial (EPI) es un grupo de trastornos pulmonares en los cuales los tejidos pulmonares resultan inflamados y luego dañados.

Los pulmones contienen diminutos sacos de aire (alvéolos), que es donde se absorbe el oxígeno. Estos sacos de aire se expanden con cada respiración”. [2]

“El tejido alrededor de estos sacos de aire se denomina intersticio. En personas con enfermedad pulmonar intersticial, este tejido se vuelve rígido o cicatrizado, y los sacos de aire no pueden expandirse tanto. En consecuencia, una cantidad menor de oxígeno puede llegar al cuerpo.

La EPI puede ocurrir sin una causa conocida. Esto se denomina EPI idiopática. La fibrosis pulmonar idiopática (FPI) es la enfermedad más común de este tipo”. [2]

## Neumonía

“Es una afección respiratoria en la cual hay una infección del pulmón.

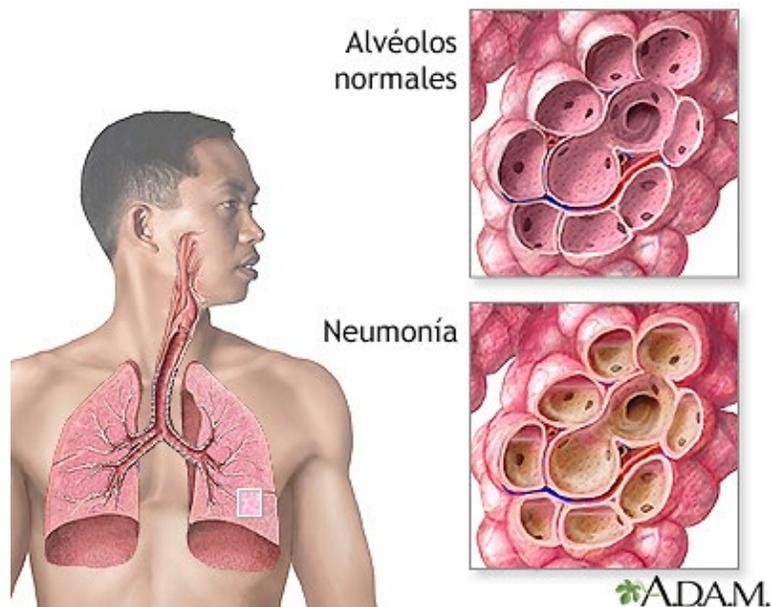


Figura 6. Neumonía [2]

La neumonía extrahospitalaria se encuentra en personas que no hayan estado recientemente en el hospital u otro centro de atención médica, como un hogar de ancianos o un centro de rehabilitación. La neumonía que afecta a las personas en centros de atención médica, tales como hospitales, se denomina neumonía intrahospitalaria (o neumonía asociada a la atención médica)”. [2]

“La neumonía es una enfermedad común que afecta a millones de personas en los Estados Unidos todos los años. Pueden causarla microbios llamados

bacterias, virus y hongos. En los adultos, las bacterias son la causa más común de neumonía”. [2]

### **Edema pulmonar**

“Es una acumulación anormal de líquido en los pulmones que lleva a que se presente dificultad para respirar.

El edema pulmonar a menudo es causado por insuficiencia cardíaca congestiva. Cuando el corazón no es capaz de bombear sangre de manera eficiente, esta se puede represar en las venas que llevan sangre a través de los pulmones”. [2]

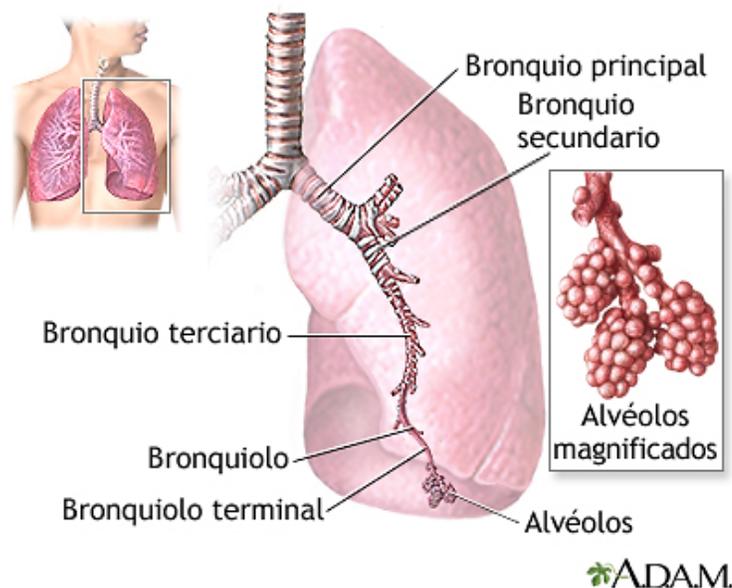


Figura 7. Edema pulmonar [2]

“A medida que la presión en estos vasos sanguíneos se incrementa, el líquido es empujado hacia los espacios de aire (alvéolos) en los pulmones. Este líquido reduce el movimiento normal del oxígeno a través de los pulmones. Estos dos factores se combinan para causar dificultad para respirar”. [2]

### **enfermedad pulmonar obstructiva crónica (EPOC)**

“La enfermedad de obstrucción pulmonar crónica o enfermedad pulmonar obstructiva crónica (EPOC) es un grupo de enfermedades pulmonares que dificultan la respiración y empeoran con el tiempo.

Normalmente, las vías respiratorias y los alvéolos pulmonares son elásticos o se estiran. Al inhalar, las vías respiratorias llevan aire a estos saquitos. Estos se llenan de aire, como un pequeño globo. Al exhalar, los saquitos se desinflan y sale el aire. Si tiene EPOC, fluye menos aire dentro y fuera de sus vías respiratorias debido a uno o más problemas”. [2]

La EPOC tiene dos tipos principales:

**Enfisema:** “Afecta a los alvéolos pulmonares, así como las paredes entre ellos. Resultan dañados y son menos elásticos

**Bronquitis crónica:** El revestimiento de las vías respiratorias se irrita e inflama constantemente, lo que produce inflamación y mucosidad”. [2]

### **Cáncer de pulmón**

“El cáncer de pulmón es un cáncer que se forma en los tejidos del pulmón, generalmente en las células que recubren los conductos de aire. Es la principal causa de muerte por cáncer tanto en hombres como mujeres.

Hay dos tipos principales, cáncer de pulmón de células pequeñas y cáncer de pulmón de células no pequeñas. Estos dos tipos crecen de manera diferente y se tratan de manera diferente. El cáncer de pulmón de células no pequeñas es el tipo más común”. [2]

### **Fibrosis pulmonar idiopática**

“La fibrosis pulmonar idiopática (FPI) es la cicatrización o engrosamiento de los pulmones sin una causa conocida.

Los proveedores de atención médica no conocen qué causa la FPI ni por qué razón algunas personas la desarrollan. Idiopática significa que la causa se desconoce. La afección puede deberse a que los pulmones responden a una lesión o sustancia desconocida. Los genes pueden jugar un papel en el desarrollo de esta enfermedad. La enfermedad se presenta con más frecuencia en personas entre 60 y 70 años de edad. La FPI es más frecuente en hombres que en mujeres”. [2]

## AUSCULTACIÓN DEL SISTEMA RESPIRATORIO

“La auscultación del sistema respiratorio es un método económico, no invasivo, seguro, fácil de realizar y una de las técnicas de diagnóstico más antiguas utilizadas por los médicos para diagnosticar diversas enfermedades pulmonares. La anamnesis y un examen físico detallado, incluida la secuencia tradicional de inspección, palpación, percusión y auscultación, deben considerarse una parte esencial del examen clínico, incluso en el siglo 21 con un avance explosivo en la tecnología relacionada con las ciencias de la salud. El avance tecnológico ha llevado a la erosión de la enseñanza de cabecera debido a la excesiva dependencia de las pruebas de laboratorio; por lo tanto, la relevancia clínica de la auscultación ha retrocedido significativamente en los últimos años. Fue Hipócrates quien inició el concepto de auscultación aplicando el oído al tórax del paciente para escuchar los sonidos respiratorios transmitidos y llamó a este procedimiento "auscultación inmediata". Describió esto como un método de auscultación directa. Sin embargo, con la invención del estetoscopio por Rene Theophile Hyac en el Laënnec en 1816; el arte de la auscultación no solo se hizo popular en todo el mundo, sino que también resultó cómodo para pacientes y médicos. Laënnec publicó su obra fundamental en 1819 en su obra maestra, "Tratado sobre las enfermedades del tórax". Inicialmente; usó un cono de papel enrollado y luego un tubo de madera. El estetoscopio moderno había sufrido varias modificaciones antes de ser moldeado a la forma actual. La auscultación de los pulmones incluye sonidos respiratorios: su carácter e intensidad, resonancia vocal y sonidos adventicios. Discutiremos los diversos tipos de sonidos respiratorios, sonidos adventicios y resonancia vocal, su importancia clínica y patogenia”. [3]

### FÍSICA DE LOS SONIDOS RESPIRATORIOS

“El sonido de la respiración tiene tres caracteres; frecuencia, intensidad y timbre o calidad; lo que nos ayuda a diferenciar dos sonidos similares”. [3]

#### **Frecuencia y tono**

“La frecuencia mide el número de ondas sonoras o vibraciones por segundo y se mide objetivamente. Se mide en hercios (Hz). La frecuencia depende del número de longitudes de onda por segundo. La longitud de onda es la distancia desde el pico de una onda de presión hasta la siguiente onda de presión y comúnmente se

designa con la letra griega lambda ( $\lambda$ ). La longitud de onda depende de la velocidad de las ondas sonoras, el medio a través del cual atraviesan las ondas sonoras y la temperatura del medio. Cuando las longitudes de onda son más cortas, hay un mayor número de ondas sonoras por segundo y las frecuencias serán más altas. Por otro lado, con longitudes de onda más largas, las frecuencias son más bajas. El tono es la percepción subjetiva de la frecuencia del sonido. El tono depende de la frecuencia y suele estar dentro de los 5 Hz de la frecuencia. El oído humano puede percibir ondas sonoras en una amplia gama de frecuencias, que van de 20 a 20.000 Hz". [3]

### **Amplitud o sonoridad**

"La amplitud está relacionada con la energía de las ondas sonoras y se mide por la altura de las ondas sonoras desde la posición media. La sonoridad es la percepción subjetiva de amplitud. El rango de amplitud es extremadamente amplio, por lo que se mide en una escala logarítmica y se representa en decibelios (dB). El sonido medido a 10 dB tiene un aumento en la intensidad del sonido de 10 veces". [3]

### **Calidad o timbre**

"La calidad o el timbre es una propiedad importante del sonido que diferencia dos sonidos con el mismo tono y volumen. El sonido se compone de varias frecuencias. La frecuencia fundamental o frecuencia primaria es la frecuencia más baja de una onda de sonido y determina el tono del sonido. Las frecuencias más altas que las frecuencias fundamentales se denominan armónicos. Los armónicos son sobretonos cuyas frecuencias son múltiplos enteros de la frecuencia fundamental". [3]

## **RUIDOS RESPIRATORIOS VESICULARES O PULMONARES NORMALES**

"El sonido vesicular de la respiración es un nombre inapropiado ya que vesículas significa alvéolos, y esto da la impresión de que el sonido de la respiración se origina en el nivel alveolar. Sin embargo, los ruidos respiratorios no se pueden generar a nivel alveolar ya que el flujo de aire es laminar dentro de los alvéolos. El sonido espiratorio es audible solo en la fase inicial. La fase espiratoria corta se debe a la naturaleza pasiva de la espiración que genera un flujo de aire menos turbulento. El origen de ambas fases de la respiración son también sitios indiferentes. El componente inspiratorio se origina en las vías respiratorias lobulares y segmentarias, mientras que el componente espiratorio surge de las

vías respiratorias más centrales. Por lo tanto, la turbulencia generada durante la espiración se aleja de la pared torácica y se vuelve más débil. Los sonidos pulmonares normalmente alcanzan su punto máximo en frecuencias por debajo de 100 Hz, con una fuerte caída de la energía del sonido que se produce entre 100 y 200 Hz, pero aún puede detectarse a 800 Hz o más con micrófonos sensibles. Hay variaciones regionales en la intensidad del sonido de la respiración. En el ápice, la intensidad disminuye con la progresión de la inspiración realizada a partir del volumen residual mientras que, en la base, inicialmente el sonido es menos intenso y con la progresión de la inspiración, la intensidad aumenta gradualmente”. [3]

### **Sonido respiratorio vesicular normal con espiración prolongada**

“Puede ocurrir en enfermedades obstructivas de las vías respiratorias como asma y bronquitis crónica. A veces, la inspiración se vuelve dura en calidad”. [3]

### **Sonido de respiración bronquial**

“Normalmente se escucha en la parte anterior sobre el manubrio y posteriormente entre las vértebras C7 y T3. Los ruidos respiratorios bronquiales contienen componentes de frecuencia mucho más alta que los sonidos respiratorios normales debido a la alteración de la función de filtrado de paso bajo de los alvéolos, como ocurre en la consolidación”. [3]

### **Sonido traqueal normal**

“Es un sonido áspero, muy fuerte y agudo que se escucha sobre la tráquea. La frecuencia típica del sonido de la respiración traqueal varía de 100 a 1500 Hz, con una caída en la potencia por encima de una frecuencia de corte de aproximadamente 800 Hz. El sonido de la respiración traqueal tiene ambas fases del ciclo respiratorio iguales con un espacio distinto entre ellas. El rango de frecuencia del sonido traqueal es mucho más amplio que el sonido pulmonar normal con frecuencias que van de 100 a 5.000 Hz, con una fuerte caída de energía a una frecuencia de aproximadamente 800 Hz”. [3]

### **Sonido respiratorio broncovesicular**

“Es intermedio entre la respiración bronquial y vesicular. Tiene una intensidad y un tono intermedios con la misma duración de la fase inspiratoria y espiratoria. Se oye normalmente en sentido anterior sobre el primer y segundo espacio intercostal y entre las escápulas posteriores. Es anormal en otros lugares”. [3]

## SONIDOS PULMONARES ADVENTRICOS O ANORMALES

“Los sonidos adventicios son sonidos respiratorios adicionales superpuestos a los sonidos respiratorios normales. Ya en 1957, Robertson y Coope propusieron una clasificación simplificada de los sonidos pulmonares adventicios en dos categorías principales; Sonidos continuos e interrumpidos. Los sonidos continuos se clasificaron además en sibilancias de tono alto y bajo, y los sonidos interrumpidos se dividieron en tres categorías: crepitantes gruesos, medios y finos. En 1976, la International Lung Sound Association simplificó aún más la terminología: sonido discontinuo en crepitaciones finas y gruesas y sonido continuo en sibilancias y roncus”. [4]

“El sonido adventicio continuo dura más de 250 ms. Las sibilancias y los roncus son sonidos pulmonares musicales continuos. El Comité de nomenclatura pulmonar de la American Thoracic Society (ATS) define las sibilancias como sonidos continuos de tono alto con una frecuencia dominante de 400 Hz o más, y los roncus como sonidos musicales continuos de tono bajo con una frecuencia dominante de aproximadamente 200 Hz o menos. Aunque la definición ATS de sonido continuo incluye una duración superior a 250 ms, las sibilancias no necesariamente deben extenderse más allá de 250 ms y, por lo general, dura más de 80-100 ms. Las sibilancias suelen ser más fuertes que los ruidos respiratorios subyacentes y, a menudo, son audibles con la boca abierta del paciente o mediante auscultación sobre la tráquea y, en ocasiones, a cierta distancia del paciente. El roncus, al ser de tono bajo, se escucha mejor sobre la pared torácica. Los ronquidos tienen una calidad de ronquido ya que el tono suele estar cerca de 150 Hz. Tanto las sibilancias como los ronquidos se caracterizan por formas de onda sinusoidales. Con mucha frecuencia, las sibilancias son de naturaleza espiratoria, pero también pueden ser inspiratorias o bifásicas”. [4]

### **Las alteraciones en los sonidos pulmonares pueden clasificarse como:**

**Crepitaciones finas:** “se caracterizan por ser un sonido inspiratorio discontinuo de tono alto, con una deflexión de onda rápidamente amortiguada de frecuencia aproximada de 650Hz y una duración de aproximada de 5ms”. [24]

**Crepitaciones gruesas:** “se caracterizan por ser un sonido inspiratorio discontinuo de tono bajo, con una deflexión de onda rápidamente amortiguada de frecuencia aproximada de 350Hz y una duración de aproximada de 15ms”. [24]

**Sibilancias:** “se caracterizan por ser un sonido espiratorio continuo de tono alto, con una frecuencia sinusoide desde los 100 a los 5000Hz y una duración mayor a 80ms”. [24]

**Roncus:** “se caracterizan por ser un sonido espiratorio continuo de tono bajo, con una frecuencia sinusoidal aproximada de 150Hz y una duración mayor a 80ms”. [24]

**Estridor:** “se caracterizan por ser un sonido inspiratorio continuo de tono alto, con una frecuencia sinusoidal mayor a 500Hz y una duración de aproximada de 5ms”. [24]

**Frote pleural:** “se caracterizan por ser un sonido inspiratorio y espiratorio continuo de tono bajo, con una sucesión rítmica de sonidos cortos de frecuencia menor a 350Hz y una duración mayor a 15ms”. [24]

## CARACTERISTICAS Y ENFERMEDADES ASOCIADAS

**Las sibilancias** “describen sonidos respiratorios que son de naturaleza más musical que los sonidos respiratorios normales: respiración bronquial y vesicular. Puede ser agudo o grave”. [25]

“Una sibilancia de tono alto puede ser indicativo de patología dentro de las vías respiratorias más pequeñas como los bronquiolos, mientras que una sibilancia de tono bajo puede deberse a una patología dentro de las vías respiratorias más grandes como los bronquios principales. Sin embargo, es posible que el sitio de la patología no siempre cause estas diferencias en el tono.

Las sibilancias indican que las vías respiratorias se estrechan, ya sea por una masa sólida, tapón de moco, broncoespasmo o hinchazón de la pared bronquial. El estrechamiento puede ser más prominente al espirar (exhalar) que al inspirar”. [25]

“Las causas más comunes de sibilancias son el **asma** y la **EPOC** (enfisema o **bronquitis** crónica). Estas condiciones dan como resultado episodios de sibilancias con sonidos de tono y volumen variables. Las causas de las sibilancias, como **el cáncer de pulmón**, suelen dar como resultado una sibilancia que es continua y no difiere significativamente en el tono, ya que la obstrucción es fija”. [25]

“Un **estridor** es un silbido monótono que es agudo y a menudo se confunde con un silbido. Puede variar desde un tono musical hasta áspero y áspero y, en la mayoría de los casos, es más pronunciado en la inspiración (inhalación).

Un estridor indica una obstrucción parcial generalmente en las vías respiratorias superiores (faringe, laringe) y la parte superior de la tráquea. Debido a la resonancia vocal, también se puede escuchar en el pecho, sin embargo, es más fuerte en la garganta y a menudo es audible sin un estetoscopio”. [25]

“La causa más común de estridor es la inflamación del tracto respiratorio superior ( **faringitis** , **laringitis** , **traqueítis** ) generalmente debido a una infección. Un **objeto extraño** alojado en el tracto respiratorio superior o un **absceso** también pueden causar estridor. Las causas menos comunes incluyen **parálisis de las cuerdas vocales** o **laringoespasma**”. [25]

**Los crepitantes** “son sonidos fuertes, de estallidos o chasquidos que son de naturaleza casi explosiva y más prominentes al inspirar (inhalar). No es musical y discontinuo a menudo interrumpido por la respiración normal o incluso por otros ruidos respiratorios anormales”. [25]

“Los crepitantes suelen indicar un colapso de las vías respiratorias distales que se abren rápidamente cuando el aire entra en el tracto respiratorio al inspirar, lo que provoca un crujido. Otra causa de estertores es cuando el aire ingresa al pulmón y burbujea en las secreciones, ya sea líquido en los pulmones o moco. Tiene un sonido más burbujeante en estos casos”. [25]

“La causa de los crepitantes se puede identificar en parte por el momento en que se escuchan durante la inspiración. Algunas de las causas incluyen:

- **Bronquiolititis:** se escucha al principio de la inspiración.
- **Bronquiectasia:** crepitantes inspiratorios y espiratorios.
- **Edema pulmonar:** crepitantes de naturaleza burbujeante más prominentes a mitad de la inspiración.
- **Fibrosis pulmonar:** se escucha al final de la inspiración (hacia el final de la inhalación)
- **Tuberculosis (TB)** – sonido burbujeante
- **Asma:** sonido burbujeante con sibilancias
- **Bronquitis:** sonido burbujeante con sibilancias ocasionales
- **Neumonía:** ruido burbujeante o crepitante

- **EPOC:** ruido áspero, a veces burbujeante
- **Absceso pulmonar:** ruido rechinante, a menudo con sibilancias o estridor según la ubicación”. [25]

“Un **roce pleural** es un chirrido o crujido como un objeto rígido que cruje bajo estrés. Es el resultado de que las dos pleuras, que recubren la cavidad pulmonar, se frotan entre sí cuando se inflaman. Un frote pleural a menudo se acompaña de dolor y está ausente si hay líquido alrededor de los pulmones entre las dos pleuras (derrame pleural). Otro término asociado es un roce pleuropericárdico que indica que la pleura también roza contra el revestimiento del pericardio (revestimiento del corazón)”. [25]

“Causas de un roce por fricción pleural:

- **Pleuritis** (pleuresía)
- **Fibrosis pleural**
- **Mesotelioma**
- **Absceso pulmonar**
- **Neumonía**
- **Infarto de pulmón”** [25]

## MATLAB

**MATLAB** (abreviatura de *MATrix LABoratory*, «laboratorio de matrices») “es un sistema de cómputo numérico que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M). Está disponible para las plataformas Unix, Windows, macOS y GNU/Linux”. [5]

“Entre sus prestaciones básicas se hallan la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos *hardware*. El paquete MATLAB dispone de dos herramientas adicionales que expanden sus prestaciones, a saber, Simulink (plataforma de simulación multidominio) y GUIDE (editor de interfaces de usuario - GUI). Además, se pueden ampliar las capacidades de MATLAB con las *cajas de herramientas (toolboxes)*; y las de Simulink con los *paquetes de bloques (blocksets)*”. [5]

“Es un *software* muy usado en universidades y centros de investigación y desarrollo. En los últimos años ha aumentado el número de prestaciones, como la de programar directamente procesadores digitales de señal o crear código VHDL.

En 2004, se estimaba que MATLAB era empleado por más de un millón de personas en ámbitos académicos y empresariales”. [5]

“Las aplicaciones de MATLAB se desarrollan en un lenguaje de programación propio. Este lenguaje es interpretado, y puede ejecutarse tanto en el entorno interactivo, como a través de un archivo de script (archivos \*.m). Este lenguaje permite operaciones de vectores y matrices, funciones, cálculo lambda, y programación orientada a objetos”. [5]

## TRANSFORMADA WAVELET

“La Transformada Wavelet (TW) es una técnica de procesamiento de señales no estacionarias, encargada de mapear una señal transitoria en una representación de tiempo-frecuencia. Este método permite el filtrado y extracción de características fundamentales de una señal (como son tiempo, frecuencia, amplitud, etc.) por medio de un análisis multi-resolución, el cual permite analizar con detalle las características de interés”. [6]

“La TW representa una señal en una secuencia de coeficientes de ondas finitas, y puede estar definida por la función wavelet  $\psi(t)$  (también llamada wavelet madre ( $\psi$ )), y la función de escala  $\phi(t)$  también conocida como la wavelet padre”. [6]

“Cuando se refiere a una “wavelet madre”, se está indicando el hecho de que las funciones usadas, derivan de una función principal, es decir, la wavelet madre es el modelo o prototipo, a partir del cual se generan el resto de funciones.

De manera muy general, la TW de una función  $f(t)$  es la descomposición de  $f(t)$  en un conjunto de funciones  $\psi_s, \tau(t)$ , que forman una base y son llamadas las Wavelets”, [6]

$$Wf(a, b) = \int f(t)\psi_{a, b}(t)dt$$

“Las Wavelets son generadas a partir de la traslación y cambio de escala de una misma función wavelet  $\psi(t)$ :

1. **Cambio de escala:** Por la que es comprimida o dilatada mediante el parámetro escala “a”.
2. **Traslación:** Con la que la señal o Wavelet es trasladada por el parámetro traslación “b” a lo largo del eje de tiempos.

De esta forma, cada familia wavelet está definida mediante la siguiente expresión”, [6]

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right), a > 0$$

“La TW se puede clasificar en dos grupos de acuerdo a la forma en que los parámetros de desplazamiento y escala son discretizados. Estos dos grupos son conocidos como la TW continua (TWC) y la TW discreta (TWD).

Dentro de la TWC y TWD existen sub grupos de TW, conocidas como las wavelets madres que son descriptas a continuación”. [6]

### **Clases de wavelet**

“Las clases de TW son definidas por las propiedades que cada una de ellas poseen, entre las más importantes, se tiene: la función de escala, el número de momentos de desvanecimiento, el soporte compacto, la regularidad y la simetría. Estas propiedades serán descritas a continuación”. [6]

Propiedades	Haar	Daubechies	Symlets	Morlet	Mexican Hat	Meyer	Gaussiana
Nombre	haar	dbN	symN	morl	mexh	meyr	gausN
<b>Soporte Compacto</b>	*	*	*				
<b>Ortogonal</b>							
<b>Soporte Compacto Biortogonal</b>							
<b>Análisis Ortogonal</b>	*	*	*			*	
<b>Análisis Biortogonal</b>	*	*	*			*	
<b>DTW</b>	*	*	*				
<b>CTW</b>	*	*	*	*	*	*	*
<b>Momentos de desvanecimiento</b>		*	*				
<b>Simetría</b>	*			*	*	*	*
<b>Antisimetría</b>		*					
<b>Cerca simetría</b>			*				
<b>Función de escala</b>	*	*	*			*	
<b>Regularidad</b>		*	*				
<b>Infinitamente regular</b>				*	*	*	*
<b>Reconstrucción exacta</b>	*	*	*		*	*	*
<b>Algoritmo Rápido</b>	*	*	*				*
<b>Filtros FIR</b>	*	*	*				

Tabla 1. Características y propiedades Wavelets. [6]

“**Función de escala o wavelet padre  $\varphi(t)$** : Es una función que está asociada a las transformaciones wavelet. Definición basada en,

$$\varphi_{j,k}(t) = 2^{-j} \psi(2^{-j}t - K), a, b \in \mathbb{Z}$$

Donde  $K$  varía desde 1 al número de coeficientes en el componente específico de una descomposición.

La función de escala ( $\varphi$ ) se relaciona con las aproximaciones de la descomposición wavelet, es decir,  $\varphi$  realiza aproximaciones de la señal y a pesar de que esta es parte de la definición de la TW, sin embargo, no es una propiedad intrínseca asociada a cualquier wavelet madre”. [6]

En este proyecto se utilizará la wavelet Daubechies (Db) descrita a continuación.

### Wavelet Daubechies.

“Las Daubechies (Db), propuestas por Ingrid Daubechies, son una familia de wavelets ortogonales, las cuales definen una transformación discreta y están caracterizados por un número máximo de momentos de desvanecimiento en cierto soporte dado. Con cada tipo de wavelet de esta clase, hay una función de escala asociada que genera un análisis de múltiples resoluciones ortogonales. A

continuación, se observa la función de escala db4 asociada a la función wavelet db4”. [6]

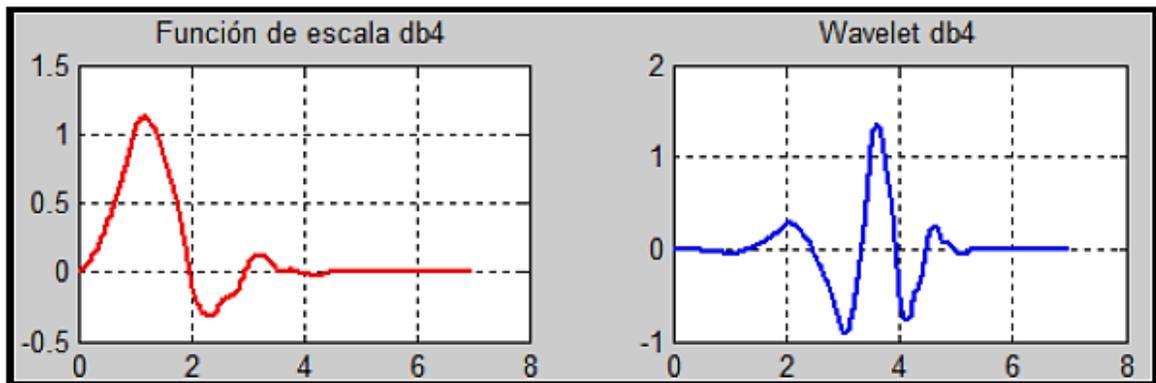


Figura 8. Función de escala (db4) asociada a la función wavelet (db4). [6]

### Métodos para implementar la TW

“Existen diversos métodos para la representación de la TW entre ellos se analizan los más reconocidos, el método convolucional y el esquema lifting; cada método posee un procedimiento distinto para poder encontrar los coeficientes detallados y aproximados, sin embargo, el resultado de ambos métodos es el mismo, en este proyecto se usará el método convolucional”. [6]

### Método Convolucional

“El método por convolución consta de filtrar una señal de entrada de forma independiente a través de filtros de descomposición pasa altas y pasa bajas. Las salidas provenientes de los filtros son sub-muestreadas por dos, es decir, se selecciona a partir de cada dos muestras para producir coeficientes de aproximación  $A(n)$  y coeficientes de detalles  $D(n)$  como se muestra en la Figura. Los filtros son conocidos como bancos de filtros de descomposición”. [6]

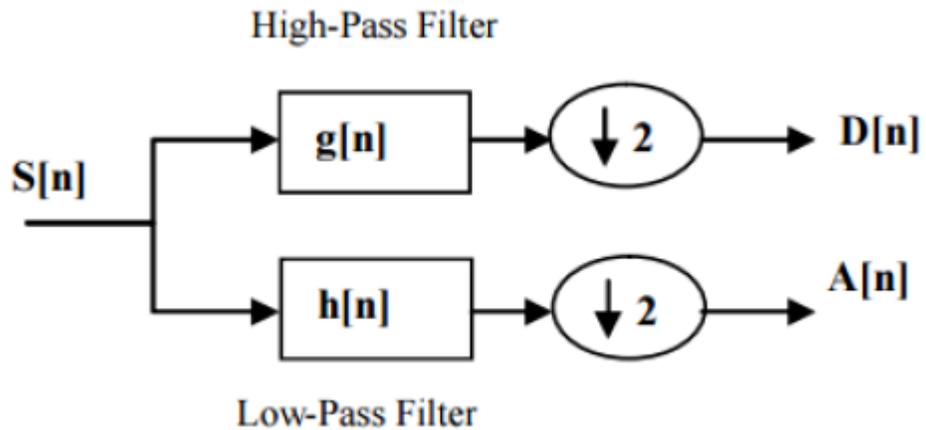


Figura 9. Estructura método de convolución. [6]

### Selección del nivel N

“La selección del nivel N significa hasta cuantos niveles se realizará la descomposición de la señal mediante los coeficientes de los filtros”. [6]

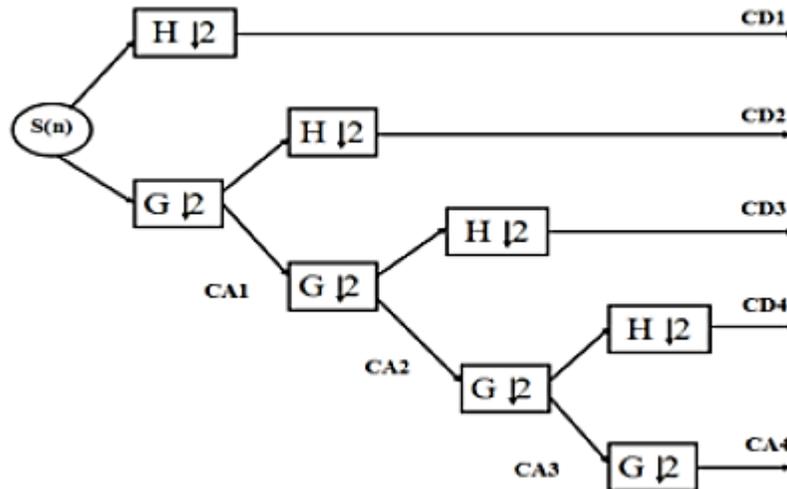


Figura 10. Esquema de descomposición multi resolución. [6]

“La separación en niveles consiste en un sub muestreo sucesivo en potencias de base 2, se debe tener en cuenta que la máxima cantidad de niveles está limitada por el número de muestras dada la siguiente relación:

$$\text{Maxniveles} = \log_2 \text{Numerodemuestras}$$

Por ejemplo, al tener un total de 1024 muestras en la señal, se permite un máximo de 10 niveles, los coeficientes que salen de los filtros pasa-altas son conocidos como detallados, de la misma manera, los coeficientes obtenidos por los filtros pasa-bajos son conocidos como coeficientes aproximados, donde  $s(n)$  representa la señal original,  $H$  representa un filtro pasa alto,  $G$  un filtro pasa bajo (los valores de  $H$  y  $G$  dependen del tipo de wavelet implementada),  $\downarrow 2$  es un sub muestro que divide en 2 la frecuencia actual de la señal,  $CA$  son los coeficientes aproximados y  $CD$  son los coeficientes detallados”. [6]

## ANALISIS WAVELET

“El análisis wavelet se está convirtiendo en una herramienta matemática habitual en el estudio de señales no estacionarias como lo son la mayoría de los sonidos. Esta herramienta realiza el análisis de una señal usando versiones escaladas y trasladadas de una función base llamada función wavelet madre  $\psi(t)$ . Hay dos diferentes tipos de transformada wavelet: la Transformada Wavelet Discreta (DWT) y la Transformada Wavelet Continua (CWT). La principal diferencia entre ambas transformadas es la forma en la cual el parámetro de escalamiento es discretizado. La CWT discretiza más fielmente que la DWT. La diferencia es que mientras que en la CWT normalmente se determina alguna base que es una potencia fraccionaria de dos, es decir  $2^{j/v}$  con  $j=1, 2, 3, \dots, n$  donde el parámetro  $v$  es conocido como el número de voces por octava, ya que para poder incrementar la escala en una octava se necesitan  $v$  escalas intermedias, mientras que en la DWT el parámetro de escalamiento siempre se discretiza con potencias enteras de dos, esto es  $2^j$  con  $j=1, 2, 3, \dots, n$  por lo que el número de voces por octava es siempre 1, por lo tanto la CWT dependiendo del valor de  $v$  nos proporciona una mayor resolución en frecuencia de la señal en análisis. Sin embargo, esto también aumenta la cantidad de cálculo requerido. La CWT se puede considerar como un banco de filtros que tienen sub bandas de frecuencia espaciadas de manera logarítmica similar al sistema auditivo humano, por lo cual provee una representación tiempo frecuencia con una resolución logarítmica en el eje de la frecuencia, lo que proporciona una mejor representación para la inspección visual, como se muestra en la figura”. [8]

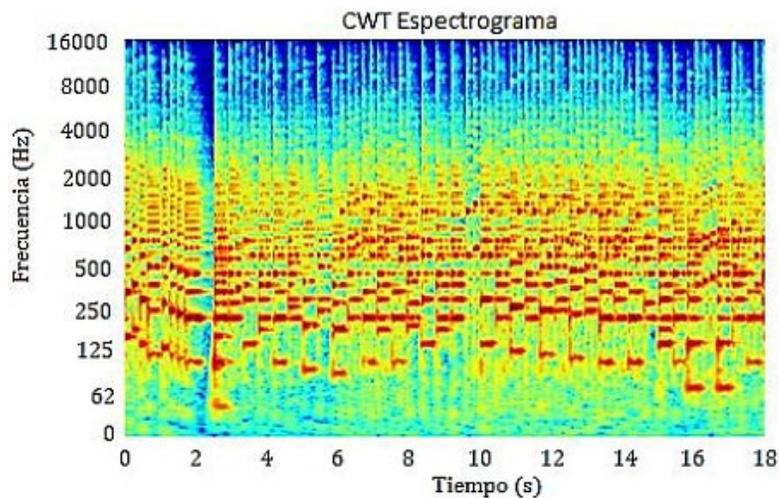


Figura 11. Espectrograma Wavelet [8]

“Recientemente los sistemas de clasificación de sonidos basados ya sea en la DWT o la CWT y las redes neuronales de aprendizaje profundo (CNN) han comprobado su utilidad en las tareas de ESC, donde han mostrado ser útiles para capturar la modulación de energía de los espectrogramas obtenidos de las señales de audio. En este trabajo se explora el poder de la CWT para generar espectrogramas representativos para clasificar sonidos ambientales usando redes CNN pre entrenadas”. [8]

## REDES NEURONALES

“Las redes neuronales son modelos matemáticos que se incluyen en la familia de algoritmos conocida como deep learning (aprendizaje profundo en castellano), donde una red neuronal es un modelo computacional, paralelo, compuesto de unidades procesadoras adaptativas (llamadas neuronas) con una alta interconexión entre ellas. Visto como un proceso de reconocimiento de patrones, las redes neuronales son una extensión de los métodos estadísticos. Se trata de un proceso de machine learning (o aprendizaje automático), el cual es usado para dos posibles objetivos, clasificación y predicción”: [9]

**Clasificación:** “una red neuronal puede ser usada para diferenciar unos elementos de otros, puede ir desde ofrecer la solución de funciones lógicas (como

la función or, not o &) a la clasificación de imágenes (como realizar distinciones entre tipos de animales o reconocimiento facial)". [9]

**Predicción:** "a partir de resultados previos se realiza una aproximación numérica a situaciones diferentes de las cuales se desconoce su resultado. Aunque se realizaron estudios anteriores, se considera el primer modelo, el presentado por McCulloch y Pitts, quienes en 1943 presentaron un elemento conocido como neurona, el cual tomaba  $n$  valores de entrada y generaba una salida mediante una función, que mediante unos valores que ajustan los valores de entrada y un sesgo podían realizar ciertas clasificaciones, la forma general de esta función (siguiendo lo sugerido por Zhang, L. y Zhang, B.): [9]

"donde: 
$$y = \text{sgn}(\omega x - \vartheta),$$

$$\text{sgn}(v) = \begin{cases} -1 & \text{si } v \leq 0 \\ 1 & \text{si } v > 0 \end{cases}$$

$\omega = (\omega_1, \omega_2, \dots, \omega_n)$  valores que ajustan la entrada o pesos sinápticos.

$x = (x_1, x_2, \dots, x_n)$  valores de entrada.

$\vartheta$  es un sesgo introducido para modificar la salida". [9]

"En 1949, Donald Hebb desarrolla una de las de las reglas de aprendizaje, método de reducción del error, más usada, The Hebbian rule aparece publicada en su libro [3]. En 1958 Rosenblatt presenta el perceptrón inspirado en el trabajo de McCulloch y Pitts, una red neuronal que toma varias entradas binarias y produce una única salida binaria. En 1960 Widrow y Hooff presentan el ADALINE, un modelo que ofrecía buenos resultados cuando el resto de modelos se habían estancado. El perceptrón multicapa creado en 1965 también supo resolver los problemas existentes. Años después en 1989 Yann LeCun crea la primera red neuronal convolucional, LeNet, la cual abrió un gran abanico de posibilidades en el reconocimiento de datos con más de una dimensión (reconocimiento de imágenes o de voz) y en 2012 con la creación de la red AlexNet por Hinton, Sutskever y Krizhevsky se observó todo el potencial que pueden llegar a ofrecer este tipo de redes neuronales". [9]

## Ventajas de las redes neuronales

“Para poder comprender el gran avance que suponen las redes neuronales debemos comprender las ventajas que ofrecen:

- Son sistemas distribuidos no lineales, una neurona al operar de forma no lineal implica que la red neuronal no opera de forma lineal, permitiendo operar con sistemas no lineales o caóticos.
- Son tolerantes a fallos, es decir que el fallo en una neurona no obteniendo la salida optima no tiene por qué afectar al buen funcionamiento de nuestra red.
- Adaptabilidad, el sistema ofrece cierta capacidad a los cambios en el entorno de trabajo, presencia de ruido, modificaciones en la entrada..., aunque tampoco ofrece una gran capacidad de adaptabilidad ya que esto afectaría a la convergencia de nuestro modelo.
- Tiene la capacidad de establecer relaciones complejas entre los datos de entrada.
- Aunque sus objetivos sean los mismos que los de la estadística clásica, puede haber variación en los resultados obtenidos siendo mejor las redes en la clasificación y actuando de forma similar en la predicción, aunque las redes neuronales convolucionales presentan un coste mayor de programación.

Todas estas propiedades generan la gran capacidad que poseen las redes neuronales, su buen funcionamiento y su alta aplicabilidad en el mundo moderno”.  
[9]

### **Componentes de una red neuronal**

“Dada una función,  $f$ , la cual realiza una partición del espacio  $R^n$ , buscar una aproximación de  $f$ :

$$f: [-1,1]^n \rightarrow \{-1,1\}^r,$$

de la cual conocemos la salida deseada,  $d \in \{-1,1\}^r$  aplicada a un patrón o entrada  $x \in [-1,1]^n$ , donde  $r$  representa el número de clases en las que

deseamos separar nuestro patrón de entrada. Análogamente podremos definir la entrada en  $[0,1]^n$  y la salida en  $\{0,1\}^r$ . La información que introduciremos en la red serán  $p$  patrones de entrenamiento": [9]

$$\{x_1, d_1\}, \{x_2, d_2\} \dots \{x_p, d_p\}.$$

“Estas entradas se irán modificando dentro de la red en cada neurona hasta formar una salida, la cual al principio del entrenamiento producirá una salida que no coincidirá con la salida deseada, pero a través de la modificación de los valores correspondientes a las conexiones, o pesos (denotados por  $w_{ab}$  el valor de la de la neurona  $a$  con la neurona  $b$ ), obtendremos la salida deseada.

Para conseguir que los datos entren en el espacio  $x \in [-1,1]^n$  realizaremos un proceso de normalización, y si todos los datos son positivos entonces, dividiendo por el máximo  $x_i$  viable, la entrada será  $x \in [0,1]^n$ ". [9]

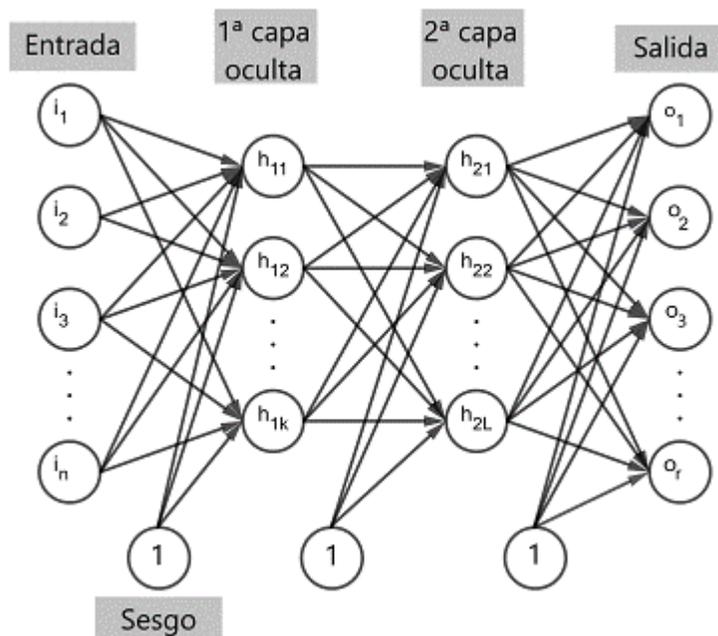


Figura 12. Red neuronal con dos capas ocultas [9]

### Unidades de procesamiento

“Una neurona o unidad de procesamiento es el elemento básico de una red neuronal. Una neurona se encarga de forjar un valor de salida,  $y_j$ , a partir de las

otras neuronas que enlazan, es decir tienen conexiones, con ella e información obtenida del exterior, sesgo. Podemos discernir tres tipos de neuronas:

1. Neuronas de entradas,  $i_k$ : su función es procesar la información procedente del exterior e introducirla en el sistema, en particular introduce la componente  $k$ -ésima del vector  $x_j$ .
2. Neuronas ocultas,  $h_{jk}$ : son neuronas que se encuentran en el centro de nuestra red, tienen una conexión  $\theta_{jk}$  llamada sesgo, o bias en inglés, y está conectada con otras neuronas de capas previas.
3. Neuronas de salida,  $o_k$ : son las neuronas que se encargan de mostrar la predicción o clasificación realizada por nuestra red, las cuales ofrecerán valores de salida en  $\{-1, 1\}$  o en  $\{0, 1\}$ . [9]

### Conexiones entre unidades

“Ofrecer una conexión entre un grupo de neuronas  $a_1, a_2, \dots, a_k$  en una neurona  $b$  equivale a dar una regla de propagación de las neuronas  $a_1, a_2, \dots, a_k$  a la neurona  $b$ , donde una regla de propagación es una función que relaciona la salida de las neuronas  $a_1, a_2, \dots, a_k$  y la entrada de la neurona  $b$ .

En la mayoría de los casos asumimos que cada unidad ofrece una contribución aditiva a la entrada de las unidades a las cual está conectada. La regla de propagación usual es aquella en la cual la entrada total en la unidad  $b$  es simplemente la suma de las salidas de las diferentes neuronas a las cuales está conectada, ponderada por los pesos de dichas componentes junto con la entrada externa, o sesgo  $\theta_b$ ” [9]

$$s_b(t) = \sum_{j=1}^k \omega_{tj} y_j + \theta_b$$

“Si la contribución  $\omega_{jb}$  es positiva se denomina excitación, en caso contrario se denomina inhibición. Existen otras reglas de propagación más complejas y menos frecuentes, por ejemplo, la introducida por Feldman y Ballard, conocida como la regla sigma-pi unit”: [9]

$$s_b(t) = \sum_{j=1}^k \omega_{tj} \prod_{i=1}^k y_i + \theta_b$$

## Reglas de activación y salida

“A lo largo largo del proceso de aprendizaje de una red neuronal se deben aplicar otros factores como las reglas de activación y desactivación de las neuronas, se trata de una regla que da el efecto de las entradas en la activación de la unidad. Para una neurona cualquiera  $k$  del modelo la función,  $F_k$ , que toma todas las entradas  $s_{t k}$  y el estado actual de la neurona  $y_{t k}$  produciendo un nuevo estado de activación de esta”: [9]

$$y_{t+1 k} = F_k(y_{t k}, s_k(t)).$$

“Suele ser una función no decreciente del total de entradas en la unidad:

$$y_{t+1 k} = F_k(s_k(t)) = F_k(\sum_j \omega_{t j k} y_{t j} + \theta_{t k}).$$

Algunas de las funciones de activación más comunes son las siguientes:

### la función ReLU (Rectified Lineal Unit):

Es una función que anula los valores negativos y deja los positivos tal como entran.

$$y_{t+1 k} = F(s_k(t)) = \max\{0, s_k(t)\}.$$

La cual ha demostrado un buen desempeño en tipos de problemas como el reconocimiento de imágenes (tal y como se desarrollará en el capítulo 2), aunque puede dejar muchas neuronas desactivadas y no está acotada”. [9]

### la función lineal:

$$y_{t+1 k} = a \cdot s_k(t),$$

“donde  $a$  es una constante real. Esta función destaca por tener cierto parecido con la ReLU, aunque destaca los valores negativos. Puede ser utilizada para exagerar las características que más destacan ( $a > 1$ ) o suavizarlas ( $a < 1$ )”. [9]

### la función sigmoideal:

$$y_{t+1}^k = F(s_k(t)) = \frac{1}{1 + e^{-s_k(t)}}$$

“Una de las bondades de esta función reside en la posibilidad de escribir su derivada en términos de ella misma, lo cual será muy importante para optimizar el coste computacional, lo cual explicamos a continuación en mayor detalle”. [9]

$$F(x) = \frac{1}{1 + e^{-x}},$$

$$F'(x) = F(x)(1 - F(x)).$$

En nuestro caso la función F dependerá de diversas variables  $\theta, \omega_1, \omega_2, \dots, \omega_m$ ,

$$F = F(\theta + \omega_1 y_1 + \omega_2 y_2 + \dots + \omega_m y_m)$$

Por lo cual, aplicando la regla de la cadena:

$$\delta F / \delta \theta = F(1-F),$$

$$\delta F / \delta \omega_i = y_i F(1-F) \quad i = 1, \dots, m.$$

### Reglas de aprendizaje

“Las reglas de aprendizaje son los algoritmos utilizados para la modificación de los pesos y el umbral con la intención de disminuir el error cometido por nuestra red. Las funciones de error, E, son aquellas que ofrecen una medida de la diferencia entre la salida de nuestra red y la salida deseada, algunas de las funciones que podemos encontrar son”: [9]

- Error cuadrático:

$$E = \frac{1}{2} \sum_{j=1}^p \sum_{k=1}^r (d_{kj} - y_{kj})^2.$$

- Entropía cruzada:

$$E = - \sum_{j=1}^p \sum_{k=1}^r [d_{kj} \log(y_{kj}) + (1 - d_{kj}) \log(1 - y_{kj})].$$

## Regla de aprendizaje del perceptrón simple

“Es el método de aprendizaje usado por el perceptrón simple, un tipo de red que explicaremos en profundidad más adelante. La regla es expresada de la siguiente manera”: [9]

$$\omega_{t+1 j} = \omega_{t j} + \Delta\omega_{t j},$$

siendo

$$\Delta\omega_{t j} = \mu(t)[d_t - y_t]x_{t j},$$

“lo cual indica que el peso es proporcional al error producido multiplicado por el valor de la entrada.  $\mu(t) \in (0,1)$  es un parámetro positivo, llamado tasa de aprendizaje, que limita la magnitud del cambio de los pesos sinápticos”. [9]

## Regla del método delta

“Primero definimos  $W_1$  como la matriz de pesos de la capa de entrada a la primera capa oculta, incluyendo el sesgo, de la misma manera definimos  $W_2$ , la matriz que representa los pesos de la segunda capa oculta con la tercera, siguiendo sucesivamente denotamos  $W_k$  los pesos que relacionan la última capa oculta y la capa de salida, denominamos por  $\Theta$  el tensor formado por  $(W_1, \dots, W_k)$ .

Si en la capa  $i$  disponemos de la función de activación  $F_i$ , podemos definir la salida ofrecida por la red de la siguiente manera” [9]

“Debido a este hecho podemos expresar la función de error, como una función  $E(\Theta, x)$ , puesto que la salida deseada,  $d$ , depende de la entrada,  $x$ , por tanto, la regla del método delta se expresa de la siguiente manera:

$$\omega_{t+1 j} = \omega_{t j} - \mu \delta E(\Theta, x) \delta\omega_j(\omega_{t j}, x_t),$$

Por ser supervisada compara si el vector de salida coincide con la respuesta deseada. Si la diferencia fuera cero podemos observar que no habrá modificación de los pesos, en caso contrario, ajusta los pesos para reducir su diferencia”. [9]

## Método del gradiente estocástico

“El método del gradiente estocástico es un proceso que permite hallar mínimos en funciones  $J: \mathbb{R}^d \rightarrow \mathbb{R}$  diferenciables.

Se puede formular el método del gradiente estocástico de la siguiente manera: [9]

1. Se toma  $u_0 \in \mathbb{R}^d$ , arbitrario.

2. Conocido  $u_t$ , se calcula  $u_{t+1}$  de la siguiente manera:

$$u_{t+1} = u_t - \mu \nabla J(u_t),$$

donde  $\mu$  es la tasa de aprendizaje.

Por otro lado se tiene que para toda función elíptica el método del gradiente estocástico converge en un número finito de pasos, donde una función,  $J: \mathbb{R}^d \rightarrow \mathbb{R}$ , se dice elíptica si es continua y diferenciable y existe  $\alpha > 0$  tal que”: [9]

$$(\nabla J(v) - \nabla J(u))(v - u) \geq \alpha \|v - u\|^2.$$

### **Modificación de los pesos**

“Por ser una regla de aprendizaje debemos tener en cuenta como se modifican los pesos. Siguiendo la notación expresada en (1.2). Podemos aplicar el método del gradiente estocástico de la siguiente manera”: [9]

$$\Theta_{t+1} = \Theta_t - \mu \nabla \Theta E(\Theta_t, x_t).$$

### **Método AdaGrad**

“El gradiente adaptativo, AdaGrad, es igual que la regla del aprendizaje del gradiente estocástico, salvo que la tasa de aprendizaje,  $\mu_t$  se ajusta después de algunas iteraciones”. [9]

$$\Theta_{t+1} = \Theta_t - \mu_t \nabla \Theta E(\Theta_t, x_t).$$

### **Topologías de redes neuronales**

“Una vez conocidas las distintas componentes de una red neuronal, procedemos a explicar los distintos tipos de redes las cuales se clasifican según su topología: La topología de una red viene dada por cómo están definidas sus funciones de activación, la función de propagación, su regla de aprendizaje y su número de capas. Según sus conexiones podemos definir dos tipos de redes neuronales:

- Redes Feed-forward propagation (o simplemente forward), donde son estrictamente hacia delante. Los ejemplos clásicos de este tipo de red son el perceptrón y el Adaline los cuales presentaremos en mayor detalle.
- Redes recurrentes (Feed back-forward propagation o back-forward), se permiten conexiones entre redes de la misma capa, incluida la misma neurona y hacia capas anteriores. En este tipo de redes se pueden alcanzar estados de estabilidad para ciertas neuronas, donde la activación de esta neurona no cambiará”. [9]

## Perceptrón

“El perceptrón es la red neuronal más básica y antigua, es usado para la separación de patrones linealmente separables. Consiste en una única capa neuronal, capa de salida, con pesos sinápticos, o simplemente pesos, y un umbral ajustable”. [9]

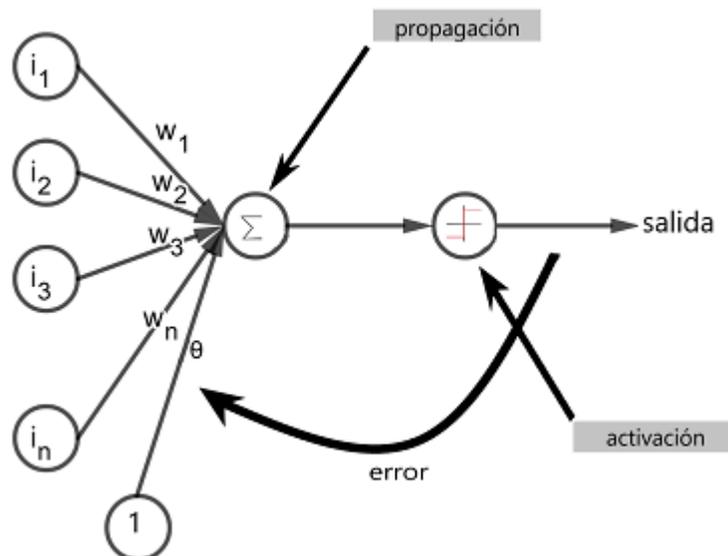


Figura 13. Estructura Perceptrón [9]

## Entrenamiento en redes neuronales

“Las redes neuronales cuentan con dos fases, entrenamiento y predicción. Para el ajuste de los pesos de las conexiones (fase de entrenamiento), se necesitan dos conjuntos de datos. Un conjunto de entrenamiento y otro de validación.

Al comienzo de nuestra red contendrá unos pesos puestos al azar. Mediante el conjunto de datos de entrenamiento se procederá al ajuste de los datos minimizando el error producido por estos. Los datos de entrenamiento se pueden introducir varias veces llamando a cada reiteración **época**”. {9}

“Una vez realizas todas las épocas deseadas, habiendo ajustado los pesos de forma que nuestros datos de entrenamiento son predichos de forma correcta, se procede a la introducción de los datos de validación. Si la predicción de los datos de validación es correcta, podemos dar por finalizado la fase de entrenamiento. En caso contrario si hemos escogido una topología de red adecuada, la falta de precisión en la validación se puede deber a dos causas principales, una mala elección en los datos de entrenamiento o un sobreajuste”

La mala elección de los datos de entrenamiento puede ser debida a que el conjunto es poco representativo, los datos no suficientes o no se ajustan de manera equilibrada a todas las clases”. [9]

“El sobreajuste se produce cuando en nuestra red se ha producido un sobreentrenamiento. La red se centra en detalles poco relevantes, los cuales impiden distinguir casos que estén fuera de los datos de entrenamiento, dicho de otra manera, es incapaz de generalizar”. [9]

## Disminución del error

“Durante la fase de entrenamiento podemos diferenciar dos tipos de disminución de error:

- Pattern mode: el error se modifica después del procesamiento de cada dato de entrenamiento. También conocido como entrenamiento online.
- Batch mode: el error se modifica después del procesamiento de un conjunto de datos, a veces incluso tras cada época. También conocido como entrenamiento off line.

En la práctica se prefiere el batch mode puesto que, aunque requiere un mayor uso de memoria reduce el coste computacional en la modificación de los pesos, además si los datos son presentados de una forma aleatoria entonces se reduce la probabilidad de caer en un mínimo local de la función de error, en vez de en un mínimo global del error el algoritmo se estaciona en un mínimo local. El batch mode permite una aproximación más precisa del vector gradiente. El análisis de la efectividad de cada método solo se puede observar una vez resuelto el problema de ambas formas.

Introduciendo en la red los datos de validación se observa el error cometido por la red y se utiliza como estimación de la calidad de nuestra red.

Una red se dice que generaliza bien cuando incluimos datos en nuestra red, no usados en el entrenamiento, y se obtiene la respuesta deseada”. [9]

### **Selección del modelo**

“Para la selección de las variables de entradas, se suele realizar un estudio previo de la importancia de las variables mediante métodos estadísticos, manteniendo las que posean una mayor relevancia.

Otra complejidad se haya en la selección del modelo es el número de capas ocultas que debemos introducir, no es necesario el uso de más de dos capas. Siendo más específicos, se puede observar que se debe usar el menor número de capas ocultas posibles para la resolución del problema. El uso de dos capas ocultas se suele utilizar para la aproximación de funciones continuas con algunas discontinuidades, aunque el uso de dos capas ocultas ya aumenta las posibilidades de obtener un mínimo local del error en vez del global. En caso de usar varias capas deberemos inicializar el proceso con distintos pesos aleatorios o realizar distintos algoritmos que disminuyan las posibilidades de acabar convergiendo en mínimos locales del error”. [9]

## **REDES NEURONALES CONVOLUCIONALES**

“Las redes neuronales convolucionales (convolutional neural networks, CNN o ConvNets) son una herramienta reciente de la tecnología. Son un tipo de red neuronal, que surge como variación del perceptrón multicapa, la diferencia se encuentra en el uso de capas de convolución y de subsampling, ideas que desarrollaremos más adelante”. [9]”

Están especialmente diseñadas para el procesamiento de datos de dos dimensiones, sobre todo reconocimiento de imágenes y señales de voz en espectrogramas, también pueden ser modificadas para la realización de tareas en una dimensión o varias dimensiones.

La idea comenzó a ser desarrollada con Kunihiko Fukushima, quien en 1982 desarrolló el neocognitron una red neuronal de tipo backpropagation que imita el proceso del cortex visual. En 1998 Yann LeCun entrenó la red denominada LeNet, la cual emplea una estructura de red neuronal convolucional y consiguiendo clasificar imágenes de dígitos escritos a mano con una precisión del 99,3%. Pero fue en 2012, con la creación de la red AlexNet (Geoffrey Hinton, Ilya Sutskever y Alex Krizhevsky) para el concurso anual ImageNet Large Scale Visual Recognition Challenge (ILSVRC) cuando se observó la potencia de las redes neuronales para la clasificación de imágenes”. [9]

### Capas de convolución

“Primero procederemos a introducir la convolución y la adaptaremos al tipo de convolución utilizada en las redes neuronales, para lo cual nos basaremos en la convolución es un tipo de operación lineal, en su forma más general es la operación de dos funciones(t) y w(t), denotada por  $(x * w)(t)$  o  $s(t)$ ”: [9]

$$s(t) = (x * w)(t) = \int x(a)w(t - a)da.$$

“Si enlazamos con la nomenclatura de las CNN entonces  $x(t)$  sería la entrada y  $w(t)$  el núcleo o kernel y  $s(t)$  la salida producida por la convolución la denominaremos mapa de características.

Por el hecho de discretizar los datos si  $t$  toma valores enteros y  $w$  solo está definido en valores enteros, entonces”: [9]

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t - a).$$

“Una cosa importante que debemos tener en cuenta en las CNN es la dimensión de entrada que estamos trabajando, por lo cual deberemos adaptar el núcleo a la dimensión necesaria (tres en el caso de imágenes a color, dos en imágenes en blanco y negro). En el caso de las bidimensionales la fórmula de convolución, con entrada  $I$  y con núcleo  $K$ , es”: [9]

$$s(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m,n)K(i-m, j -n),$$

La cual es conmutativa:

$$s(i, j) = \sum_m \sum_n I(i-m, j-n)K(m,n).$$

### **Pooling.**

“Una vez realizada la convolución nos interesa realizar una reducción del volumen de datos, de esta tarea se encarga el submuestreo o pooling, en cierta manera interactúa dentro de la red del mismo modo que una capa convolucional, haciendo operaciones en pequeñas regiones de la matriz de entrada, acumulando los elementos de las capas de características”. [9]

“Dada una matriz  $A_{axa}$  podemos definir el proceso de Max-Pooling con una amplitud  $k$  y un stride  $p$  como la matriz  $P(i, j)$  tal que:

$$P(i, j) = \max_{n,m=1,\dots,k} A[(i-1)p+m,(j-1)p+n].$$

Si fueran necesarias más filas o columnas para el desarrollo del algoritmo se pueden incluir más filas y columnas mediante el método de zero-padding.

El algoritmo de Max-Pooling usualmente utilizado es el aquel con una amplitud y stride 2”. [9]

### **Average-Pooling**

“Es similar al método de Max-Pooling, salvo que en vez de ser la salida el máximo se trata de la media aritmética”. [9]

$$P(i, j) = \frac{1}{k^2} \sum_{n,m=1,\dots,k} A[(i-1)p+m,(j-1)p+n].$$

### **Estructura de una red convolucional.**

“La arquitectura de una red neuronal convolucional empieza aplicando una capa convolucional y una capa de pooling, siguiendo este proceso repetidamente hasta obtener un conjunto de matrices, tal que al poner todos los elementos como un

vector (proceso de flattening), tenga una cantidad computacionalmente viable de elementos como para ser introducido como entrada de una red neuronal". [9]

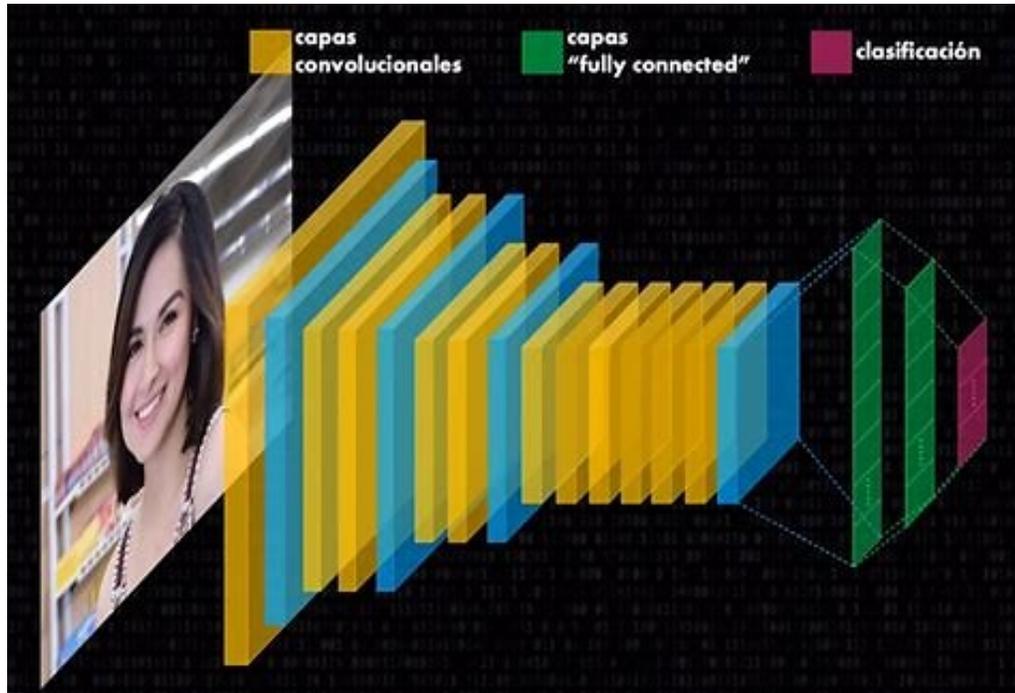


Figura 14. Estructura red convolucional [10]

### **3. OBJETIVOS**

#### **3.1 OBJETIVO GENERAL**

Diagnosticar enfermedades respiratorias mediante sonido pulmonar aplicando inteligencia artificial.

#### **3.2 OBJETIVOS ESPECÍFICOS**

1. Adquirir una base de datos de sonidos pulmonares de diferentes enfermedades respiratorias.
2. Diseñar un algoritmo usando técnicas de inteligencia artificial para el procesamiento digital de las señales de audio adquiridas.
3. Analizar las características de las señales de sonido pulmonar utilizando el algoritmo diseñado.
4. Comparar y clasificar los resultados obtenidos para diagnosticar enfermedades respiratorias.

## **4. PLANTEAMIENTO DEL PROBLEMA**

### **4.1 DEFINICIÓN DEL PROBLEMA**

“En Colombia el Ministerio de Salud y Protección Social en cabeza del del Instituto Nacional de Salud (INS) por medio de sus boletines epidemiológicos en 2019 confirma un aumento de los casos de infecciones respiratorias a nivel nacional, se estima en total de 1392 casos, lo que pone en riesgo la salud y plantea nuevas estrategias para contrarrestarlo”. [11]

La salud pública a nivel nacional presenta un aumento de alertas en cuanto al aumento de cuadros clínicos respiratorios, esto sumado a la actual estado de pandemia emitido por la organización mundial de la salud (OMS) ponen entredicho los métodos o mecanismos tradicionales o rudimentarios de diagnóstico de enfermedades respiratorias como es el caso de la auscultación del tórax por parte del médico para escuchar y diagnosticar los sonidos respiratorios del paciente por medio de un estetoscopio análogo el cual genera un campo limitado de escucha y análisis a simple oído, lo que puede llevar a un mal diagnóstico y complicar la salud de los pacientes.

### **4.2 JUSTIFICACIÓN**

En esta era la tecnología es el mejor aliado de la medicina moderna, y para el caso de los sonidos respiratorios un análisis por computadora sería el mejor método que actualmente podemos implementar para garantizar un buen diagnóstico en pro de la salud de los pacientes.

Esto hace necesario la implementación de nuevas tecnologías como la inteligencia artificial para el análisis de sonido pulmonar que conlleve a mejorar y complementar el diagnóstico y exactitud en la prevención y mejoramiento en la calidad de vida de la población.

Con este proyecto se presente estudiar una base de datos de sonidos respiratorios por medio de inteligencia artificial y herramientas de cómputo para realizar una clasificación de varias enfermedades respiratorias presentes o detectables por medio de auscultación y generar una base de datos computarizada para el análisis y estudio e identificación de enfermedades presentes en los sonidos respiratorios.

## 5. ESTADO DEL ARTE MARCO DE REFERENCIA

- “Interpretación de los sonidos pulmonares utilizando características estadísticas basadas en espectrogramas”. [12]

“Los sonidos pulmonares transmiten información útil relacionada con la patología pulmonar. La auscultación es la práctica clínica más utilizada. La auscultación se considera una herramienta de diagnóstico inferior para diferenciar los sonidos pulmonares adventicios de los normales, ya que depende de la capacidad y la experiencia auditiva del médico. La Organización Mundial de la Salud ha declarado que aproximadamente 3 millones de personas mueren cada año a causa de enfermedades pulmonares crónicas. La auscultación computarizada junto con los métodos de procesamiento de señales ha mejorado las capacidades de diagnóstico del sonido pulmonar. Los cambios anatómicos en el pulmón se reflejan en el atributo de dominio de tiempo y frecuencia del sonido pulmonar. El espectrograma es una representación gráfica de la señal en el dominio de la frecuencia del tiempo. Este trabajo es un intento de categorizar el sonido pulmonar como normal, crepitante y sibilante utilizando el espectrograma de los sonidos pulmonares. La textura y el análisis estadístico del espectrograma se calcula con el propósito de demarcación. Los parámetros estadísticos como la media, la varianza, la asimetría, la curtosis y la entropía de las imágenes del espectrograma se calculan y grafican. Los resultados demuestran que la varianza de las características y la curtosis extraídas del espectrograma delimitan los sonidos pulmonares normales del crujido y las sibilancias con mayor eficacia”. [11]

- “Identificar las características basadas en gráficos para la auscultación del sonido pulmonar”. [13]

“La auscultación es un método históricamente significativo y actualmente relevante empleado en el diagnóstico médico. El avance tecnológico unido a los principios matemáticos condujo al desarrollo de sofisticados instrumentos biomédicos. El artículo propone un novedoso método de auscultación basado en gráficos, que incorpora los principios de las técnicas de aprendizaje automático (MLT), para satisfacer los requisitos de la auscultación remota, que es muy relevante durante el brote de la pandemia COVID-19. La técnica propuesta, delineada mediante el análisis de la compleja red construida a partir de las sesenta señales de sonido respiratorio bronquial y estridor (BS y SS), revela la condición de la tráquea y del tracto respiratorio superior a través de la dinámica

del flujo de aire durante la producción de las señales. Los análisis en el dominio del tiempo y la frecuencia de las señales de series de tiempo desenvuelven la naturaleza discontinua y musical de la señal BS y SS, respectivamente. Las características del gráfico extraídas de la red compleja, construida utilizando un enfoque de mapeo de correlación, arroja luz sobre las condiciones de la tráquea. El mayor valor del número de aristas, densidad del gráfico y transitividad en BS, con componentes de baja frecuencia y alta correlación entre los nodos, explica la dinámica del flujo de aire a través de la tráquea de diámetro uniforme y mayor. La disminución de estas características y la mejora de la centralidad de grado, la centralidad de proximidad y la entropía en SS, con componentes de mayor frecuencia y baja correlación entre los nodos, refleja la condición patológica de las constricciones en la tráquea. Los MLT, análisis de componentes principales y discriminantes lineales, podrían clasificar de manera eficaz y predecir con precisión las señales utilizando las características del gráfico, lo que sugiere su potencial en la auscultación”. [13]

- “Bancos de filtros y paquetes interpolatorios de wavelets”. [14]

“En el muestreo clásico de wavelets, la investigación generalmente se centra en bases interpolatorias en espacios individuales. Rara vez se tienen en cuenta las relaciones entre las diferentes bases interpolatorias, incluidas las potencialmente relacionadas por los bancos de filtros. En este trabajo, mostramos que cuando existe una base interpolatoria en un espacio con un análisis multiresolución, este espacio puede descomponerse ortogonalmente en dos subespacios en los que también existen bases interpolatorias. Además, de manera similar a las bases ortogonales estándar, estas bases interpolatorias pueden estar relacionadas por pares de filtros, llamados bancos de filtros interpolatorios. Para tales bancos de filtros interpolatorios, se demuestra que la existencia de funciones de escalado interpolatorio generalmente conduce a la de las wavelets interpolatorias y los paquetes de wavelets interpolatorios. Con este resultado, proponemos además nuevos algoritmos para construir tales wavelets interpolatorios/paquetes wavelets a partir de las funciones de escala correspondientes. En ejemplos, nuestros algoritmos se aplican a algunos espacios de wavelet típicos, demostrando nuestros teoremas para wavelets interpolatorios, paquetes de wavelets interpolatorios y bancos de filtros interpolatorios”. [14]

- “Capítulo 6 - Bancos de filtros y compresión de wavelets”. [15]

“Las wavelets son funciones matemáticas que se pueden utilizar como alternativa a transformaciones como la transformada discreta del coseno (DCT) para dividir una señal en diferentes bandas de frecuencia (subbandas) antes del procesamiento y la cuantización. Las wavelets generalmente se implementan como bancos de filtros que permiten analizar cada subbanda a una resolución que coincida con su escala. Esto, como veremos más adelante, normalmente se logra combinando las operaciones de filtrado con el submuestreo, que también tiene el beneficio de mantener una frecuencia de muestreo general constante en todas las etapas de la descomposición. Este enfoque tiene ventajas significativas sobre los métodos basados en Fourier, especialmente cuando las señales no están relacionadas armónicamente, son de corta duración con discontinuidades o no son estacionarias. También tiene beneficios en términos de cuantización del coeficiente de subbanda, ya que los artefactos producidos tienden a ser perceptualmente menos molestos que los artefactos de bloqueo asociados con el DCT. Este capítulo examina en primer lugar la estructura básica de dos canales del banco de filtros y luego investiga las propiedades del filtro que permiten una reconstrucción perfecta en ausencia de cuantización. Luego pasamos a considerar el caso más general del filtrado multirrapa, mostrando cómo, a través de la combinación adecuada de filtros y operaciones de muestreo hacia arriba y hacia abajo, podemos lograr un muestreo crítico. A continuación, examinamos casos específicos de filtros útiles de subbanda y wavelet y luego extendemos la arquitectura básica al caso de una descomposición multietapa (multiescala) y a dos dimensiones según sea necesario para las aplicaciones de compresión de imágenes. Finalmente, examinamos las estrategias de asignación de bits y la escalabilidad en el contexto del estándar de codificación de imágenes fijas JPEG2000”. [15]

- “SIGNOS DE ENFERMEDAD RESPIRATORIA | Sonidos pulmonares”. [16]

“El sonido tiene una serie de características vitales que incluyen frecuencia, intensidad, duración y calidad. Estos componentes determinan la gran cantidad de sonidos pulmonares que pueden ser escuchados por el oído humano. Históricamente, los sonidos pulmonares se detectaban colocando el oído en la pared torácica del paciente. Sin embargo, con el descubrimiento del estetoscopio, esta práctica fue abandonada. Aunque se puede dilucidar mucho sobre la patología torácica subyacente escuchando a un paciente respirar mientras está de pie al final de la cama, auscultación ha añadido mucho al examen clínico de pacientes con enfermedades respiratorias. Los sonidos pulmonares se pueden dividir ampliamente en sonidos respiratorios y sonidos de voz, y pueden ser normales y anormales. Los sonidos pulmonares anormales pueden transmitirse anormalmente por la respiración o la voz, o pueden ser adventiciales (accidentales). En este artículo, se discutirá el rango de varios sonidos normales y

anormales que produce el pulmón y se resaltará su pertinencia a la enfermedad respiratoria”. [16]

- “Aprendizaje de imágenes hiperespectrales a partir de imágenes RGB a través de una CNN gruesa a fina”. [17]

“La teledetección hiperespectral es conocida por su extraordinaria capacidad de distinción espectral para discriminar diferentes materiales. Sin embargo, el costo de la adquisición de imágenes hiperespectrales (HSI) es mucho más alto en comparación con las imágenes RGB tradicionales. Además, las resoluciones espaciales y temporales se sacrifican para obtener una resolución espectral muy alta debido a las limitaciones de las tecnologías de sensores. Por lo tanto, en este artículo, los HSI se reconstruyen utilizando imágenes RGB de fácil adquisición y una red neuronal convolucional (CNN). Como resultado, las imágenes RGB de alta resolución espacial y temporal se pueden heredar a los HSI. Específicamente, una CNN de dos etapas, conocida como red de superresolución espectral (SSR-Net), está diseñada para aprender el modelo de transformación entre imágenes RGB y HSI a partir de datos de entrenamiento, incluida una red de predicción de banda (BP-Net) para estimar bandas hiperespectrales a partir de imágenes RGB y una red de refinamiento (RF-Net) para reducir aún más la distorsión espectral en el paso de predicción de banda. Como resultado, las características conjuntas aprendidas en la SSR-Net propuesta pueden predecir directamente los HSI a partir de sus escenas correspondientes en imágenes RGB sin conocimiento previo. Los resultados experimentales obtenidos en varios conjuntos de datos de referencia demuestran que el SSR-Net propuesto supera a varios métodos de vanguardia al garantizar una mayor calidad en la reconstrucción HSI y mejora significativamente el rendimiento de las imágenes RGB tradicionales en la clasificación”. [17]

- “Verificación de la estructura de redes neuronales profundas en tiempo de compilación”. [18]

“Este artículo presenta *TensorSafe*, una biblioteca Haskell que hace posible la definición y validación estructural de arquitecturas de redes neuronales profundas. Hoy en día, el proceso de desarrollo de modelos de aprendizaje profundo se ha simplificado notablemente debido a la disponibilidad de herramientas sofisticadas en la industria. Sin embargo, la mayoría de estas herramientas no proporcionan ningún control de seguridad en el momento de la compilación, lo que hace que los desarrolladores se ocupen de errores e incertidumbres inesperados en tiempo de ejecución. En particular, validar la

estructura de las redes neuronales profundas en el momento de la compilación es un tema complejo que implica la validación matemática de todas las operaciones que realizará un modelo de aprendizaje profundo. Además, esta comprobación estructural requiere un uso avanzado de teorías de sistemas de tipos para manipular definiciones de tipos abstractas capaces de modelar construcciones de redes neuronales. Muchas técnicas de programación diferentes estuvieron involucradas en la especificación de TensorSafe. Principalmente, la aplicación del paradigma de programación funcional y el uso de programación a nivel de tipo fueron de gran importancia para el proceso de desarrollo y para sondear la corrección de los modelos de redes neuronales. La evaluación experimental mostró que mediante el uso de TensorSafe es posible crear correctamente arquitecturas de redes neuronales profundas bien conocidas, tales como AlexNet o ResNet50”. [18]

- “Redes neuronales profundas de Kronecker: un marco general para redes neuronales con funciones de activación adaptativa”. [19]

“Proponemos un nuevo tipo de redes neuronales, las redes neuronales de Kronecker (KNN), que forman un marco general para las redes neuronales con funciones de activación adaptativa. Los KNN emplean el producto Kronecker, que proporciona una forma eficiente de construir una red muy amplia manteniendo bajo el número de parámetros. Nuestro análisis teórico revela que, en condiciones adecuadas, los KNN inducen una desintegración más rápida de la pérdida que la de las redes de alimentación. Esto también se verifica empíricamente a través de un conjunto de ejemplos computacionales. Además, bajo ciertos supuestos técnicos, establecemos la convergencia global de descenso de gradiente para KNNs. Como caso concreto, proponemos el *Rowdy* función de activación que está diseñada para deshacerse de cualquier región de saturación mediante la inyección de fluctuaciones sinusoidales, que incluyen parámetros entrenables. La función de activación de *Rowdy* propuesta se puede emplear en cualquier arquitectura de red neuronal como redes neuronales de avance, redes neuronales recurrentes, redes neuronales convolucionales, etc. La efectividad de los KNN con activación de *Rowdy* se demuestra a través de varios experimentos computacionales que incluyen la aproximación de funciones utilizando redes neuronales de avance, la inferencia de soluciones de ecuaciones diferenciales parciales utilizando las redes neuronales informadas por la física y los problemas de referencia de aprendizaje profundo estándar utilizando redes neuronales convolucionales y totalmente conectadas”. [19]

- “Selección de características dispersas de bajo rango para la clasificación de imágenes”. [20]

“Hay mucha redundancia en las imágenes en bruto de alta dimensión, lo que no solo aumenta en gran medida la carga computacional del proceso de clasificación de imágenes, sino que también degrada inevitablemente el rendimiento de clasificación del modelo. Los algoritmos de reducción de dimensionalidad de alto rendimiento necesitan un desarrollo urgente. Para resolver este problema, desarrollamos un modelo de selección de características novedoso para la reducción de dimensiones. Reduce en gran medida las características redundantes y selecciona las características más representativas para su clasificación. Además, también diseñamos una versión novedosa de la red neuronal convolucional ligera (newCNN). El nuevo CNN puede mejorar el rendimiento de clasificación del sistema. Para mejorar la precisión de la clasificación, creamos un modelo de clasificación híbrida (HC) con el nuevo CNN y Support Vector Machines (SVM). Este modelo no solo resuelve el problema del sobreajuste en el proceso de entrenamiento, sino que también tiene una excelente capacidad de generalización y robustez. Los resultados experimentales verifican la efectividad de nuestros métodos propuestos”. [20]

- “Un análisis de redes neuronales convolucionales para la clasificación de imágenes”. [21]

“Este artículo presenta un análisis empírico del desempeño de las redes neuronales convolucionales (CNN) populares para identificar objetos en transmisiones de video en tiempo real. Las redes neuronales de convolución más populares para la detección de objetos y la clasificación de categorías de objetos a partir de imágenes son Alex Nets, GoogLeNet y ResNet50. Hay una variedad de conjuntos de datos de imágenes disponibles para probar el desempeño de diferentes tipos de CNN. Los conjuntos de datos de referencia que se encuentran comúnmente para evaluar el rendimiento de una red neuronal convolucional son un conjunto de datos ImageNet y conjuntos de datos de imágenes CIFAR10, CIFAR100 y MNIST. Este estudio se centra en analizar el rendimiento de tres redes populares: Alex Net, GoogLeNet y ResNet50. Hemos tomado tres conjuntos de datos más populares ImageNet, CIFAR10 y CIFAR100 para nuestro estudio, ya que probar el rendimiento de una red en un solo conjunto de datos no revela su verdadera capacidad y limitaciones. Debe tenerse en cuenta que los videos no se utilizan como un conjunto de datos de entrenamiento, se utilizan como conjuntos de datos de prueba. Nuestro análisis muestra que GoogLeNet y ResNet50 pueden reconocer objetos con mayor precisión en comparación con Alex Net. Además, el rendimiento de las CNN capacitadas varía sustancialmente entre las

diferentes categorías de objetos y, por lo tanto, discutiremos las posibles razones de esto”. [21]

- “ADCNN: Hacia el aprendizaje de la dilatación adaptativa para redes neuronales convolucionales”. [22]

“Los núcleos de convolución dilatados están limitados por su dilatación compartida, lo que les impide ser conscientes de los diversos contenidos espaciales en diferentes ubicaciones. Abordamos tales limitaciones formulando la dilatación como pesos entrenables con respecto a posiciones individuales. Proponemos Adaptive Dilation Convolutional Neural Networks (ADCNN), una extensión ligera que permite a los núcleos convolucionales ajustar su valor de dilatación en función de diferentes contenidos a nivel de píxel. A diferencia de los modelos de contenido adaptativo anteriores, ADCNN infiere dinámicamente la dilatación por píxeles a través del modelado de patrones de retroalimentación directa, lo que proporciona una nueva perspectiva para desarrollar estructuras de red adaptativas distintas del muestreo de espacios del núcleo. Los resultados de nuestra evaluación indican que las ADCNN se pueden integrar fácilmente en varias redes troncales y superar constantemente a sus contrapartes habituales en diversas tareas visuales”. [22]

- “Nuevas perspectivas en la evaluación automatizada de los sonidos respiratorios”. [23]

“Nuestra capacidad auditiva no nos permite detectar cambios menores a 3dB a frecuencias bajas (donde se encuentra contenido el sonido respiratorio < 500Hz). Con el análisis acústico es fácil detectar cambios significativos en la intensidad del sonido y, detectar obstrucciones leves del 10% con la disminución de la intensidad del sonido respiratorio a bajas frecuencias en inspiración y el incremento de las altas frecuencias en espiración. Los trabajos de investigación están orientados a desarrollar mejores sistemas de adquisición y procesamiento de la señal acústica, así como también entender los mecanismos involucrados en la génesis de diversas patologías respiratorias (estenosis traqueal, asma, enfermedad pulmonar obstructiva crónica, síndrome de apnea obstructiva del sueño, fibrosis pulmonar, entre otras) y en la evaluación del análisis acústico del sonido respiratorio como herramienta diagnóstica y de seguimiento. En los laboratorios de acústica respiratoria se retoma el arte de la auscultación para incorporar los avances científicos y tecnológicos logrados desde la época de Laënnec. A largo plazo, el objetivo es lograr que el estetoscopio se utilice con

mayor eficiencia, motivo por el cual es importante reflexionar sobre lo que hacemos, lo que sabemos y la metodología que utilizamos como clínicos al recabar un signo a la exploración”. [23]

## 6. RESULTADOS

**Actividad 1.** Adquisición de cada uno de los sonidos pulmonares como base de datos.

El primer paso para la realización del proyecto es adquirir una base de datos de sonidos respiratorios, esta es la materia prima para la obtención de información necesaria en el análisis de los parámetros o características de las distintas enfermedades respiratorias que pueden ser detectadas a través de sonido pulmonar.

Dentro de la base de datos de sonido respiratorio adquirida "Un conjunto de datos de sonidos pulmonares registrados en la pared torácica con un estetoscopio electrónico" encontramos varias grabaciones donde se evidencia en su sonoridad la presencia de distintas alteraciones respiratorias en los pacientes sometidos a las mediciones.

"Archivos de audio con anotaciones de los sonidos pulmonares grabados desde varios puntos de vista y la ubicación en la pared torácica de donde se tomó la grabación (Posterior: P Inferior: L Izquierda: L Derecha R, Superior: U, Anterior: A, Medio: M). al igual que el Tipo de filtro; B: campana 20-200Hz, D: diafragma 100-500 Hz, E: rango extendido 50-500 Hz. y número de paciente: P1-P112". [7]

"Los sonidos pulmonares se adquirieron mediante un estetoscopio electrónico colocado en varios puntos estratégicos de la pared torácica. La grabación se realizó usando el estetoscopio electrónico Littmann de 3M modelo 3200 y se transmitió a una computadora usando el adaptador Bluetooth provisto". [7]

**Actividad 2.** Selección de las técnicas de inteligencia artificial y extracción de patrones.

Las técnicas de procesamiento digital de señales e inteligencia artificial se seleccionaron de acuerdo a las características y cualidades que estas poseen para desarrollar este proyecto frente a técnicas similares a ellas que no se ajustan a los requerimientos mínimos de los cálculos y diseños propuestos.

Para la identificación de los sonidos pulmonares y clasificación de las enfermedades respiratorias se hará uso de la herramienta matemática Wavelet, que, frente a otras como Fourier, ofrece un análisis espectral de frecuencia contra tiempo y opciones de diversos tipos de visualización y resolución en la dilatación o compresión de las señales para una mejor identificación de patrones o formas de onda deseados.

Debido a estas cualidades, Wavelet también será seleccionada para la creación de imágenes de espectrogramas de las señales de audio que serán procesadas para la clasificación de una base de datos computarizada basada en inteligencia artificial.

El proceso de inteligencia artificial de clasificación de enfermedades basado en los espectrogramas de las señales de audio para la creación de nuestra base de datos computarizada será basado en redes neuronales convolucionales (CNN), ya que estas por ser parte del aprendizaje profundo o Deep Learning ofrecen un nivel de procesamiento superior necesario en este proyecto, frente a otras redes neuronales de aprendizaje superficial o Machine Learning.

La selección del tipo de red neuronal convolucional dependerá del porcentaje de validación de las pruebas de entrenamiento y aprendizaje que se realicen en cada una de ellas, debido a que, por su estructura, profundidad y capacidad, estas tendrán resultados diferentes.

**Actividad 3.** Desarrollo de sistema de código para el procesamiento digital de señales.

Se procede a implementar la Transformada Wavelet tipo Daubechies (Db10), para realizar el análisis en frecuencia y tiempo de los sonidos respiratorios haciendo uso del banco de filtros wavelet para separar las frecuencias a analizar y obtener los patrones característicos de las señales, como su forma de onda, longitud, amplitud, simetría entre otras, para determinar a qué tipo de sonido pulmonar es y que enfermedad respiratoria representa.

Con una frecuencia de muestreo ( $F_s$ ) de 4000Hz propia de los audios obtenidos y limitando los datos a 5 segundos (seg) obtenemos una longitud máxima ( $L_m$ ) de 20000 muestras (pts). Y con esta hallamos el nivel máximo de descomposición frecuencial que es de 14 niveles.

Fs (hz) = 4000		t (seg) = 5		Lm (pts) = 20000		Niv.Max. = 14.29	
NIVEL	RANGO DE FRECUENCIA (hz)					N° MUESTRAS (pts)	
1	1000	2000				10000	
2	500	1000				5000	
3	250	500				2500	
4	125	250				1250	
5	62.5	125				625	
6	31.25	62.5				312.5	
7	15.625	31.25				156.25	
8	7.8125	15.625				78.125	
9	3.90625	7.8125				39.0625	
10	1.953125	3.90625				19.53125	
11	0.9765625	1.953125				9.765625	
12	0.4882813	0.9765625				4.8828125	
13	0.2441406	0.48828125				2.44140625	
14	0.1220703	0.24414063				1.22070313	

Tabla 2. Niveles banco de filtro wavelet.

Al calcular el banco de filtros wavelet se tiene exactamente que rangos de frecuencia y numero de muestras o puntos tiene cada nivel. Con esto se procede a realizar la descomposición frecuencial de wavelet tipo Daubechies (Db10) a 5 niveles para cada una de las señales de audio como se muestra a continuación.

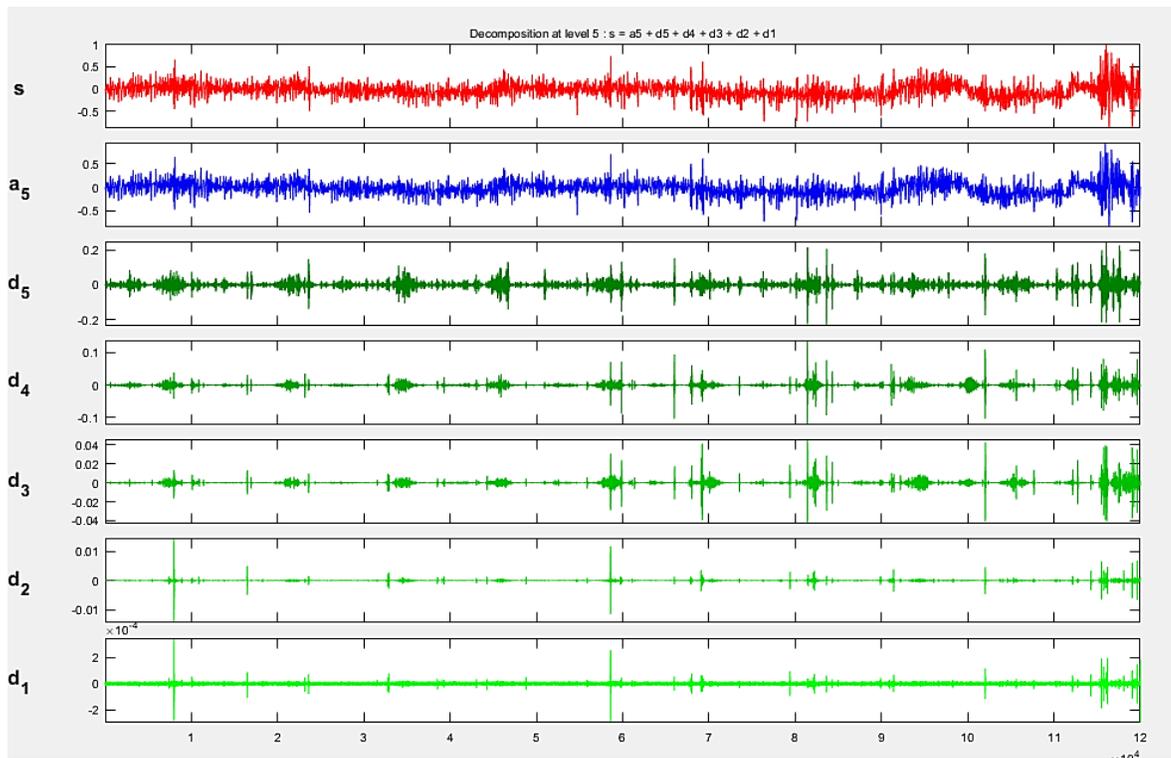


Figura 15. Descomposición frecuencial Wavelet.

Con la descomposición frecuencial wavelet realizada se puede analizar cada nivel o detalle de descomposición por rango de frecuencia en busca de las formas de onda características de los sonidos pulmonares. Al encontrar estas se puede identificar a que tipo de sonido pulmonar pertenece.

Teniendo identificado el sonido pulmonar al que pertenece la señal se procede a identificar sus características en tiempo, frecuencia, amplitud, simetría, repetitividad, entre otras, para clasificar o etiquetar el tipo de enfermedad respiratoria a la cual pertenece.

Al finalizar del proceso anterior se encontraron en las señales analizadas patrones de forma de onda característicos de sonidos pulmonares como sibilancias, crepitantes, frote pleural y murmullo pulmonar normal. Los cuales se identificaron con enfermedades como asma, bronquitis, efusión pleural, EPOC, fibrosis pulmonar, insuficiencia cardiaca, neumonía y Murmullo normal pulmonar.

A continuación, se muestran algunos de los patrones obtenidos:

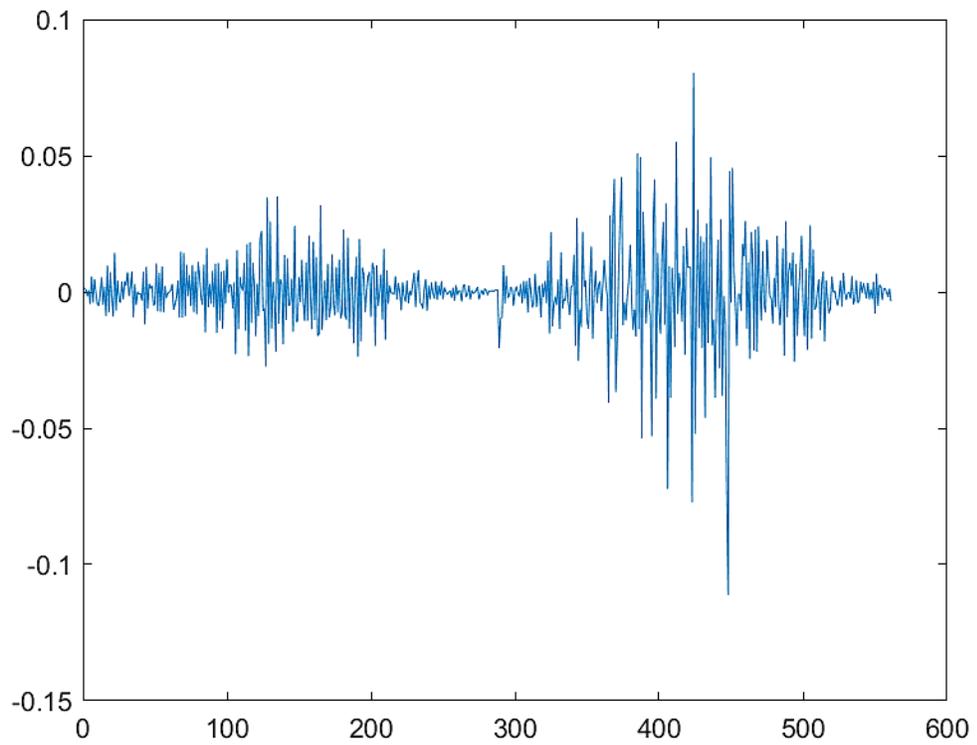


Figura 16. Asma por sibilancia.

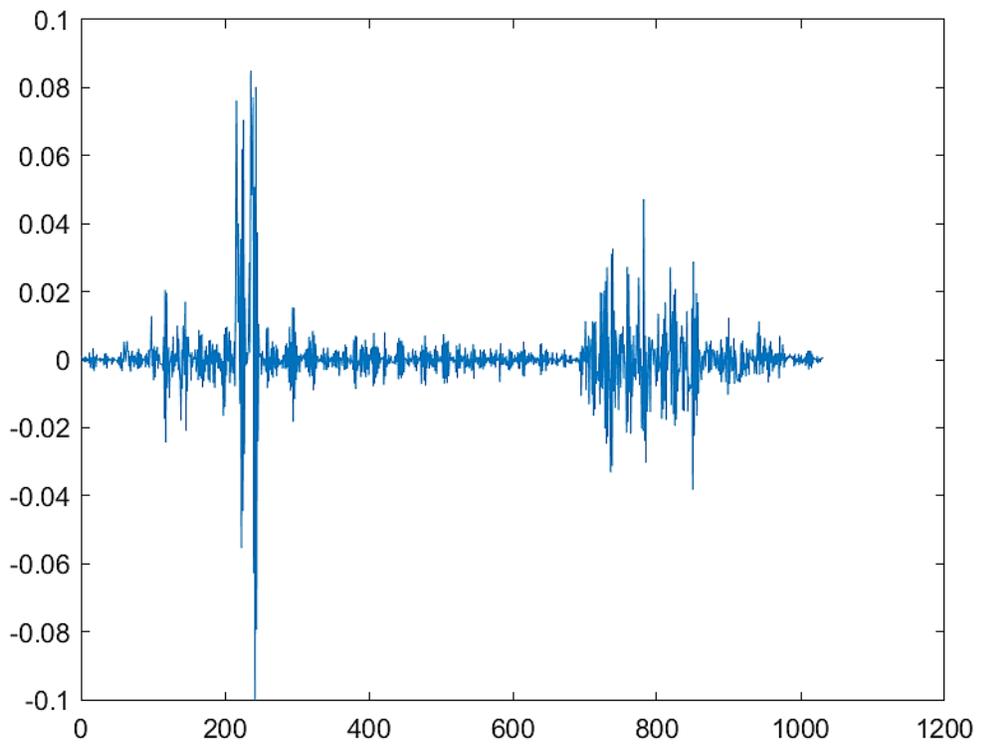


Figura 17. Bronquitis por crepitantes.

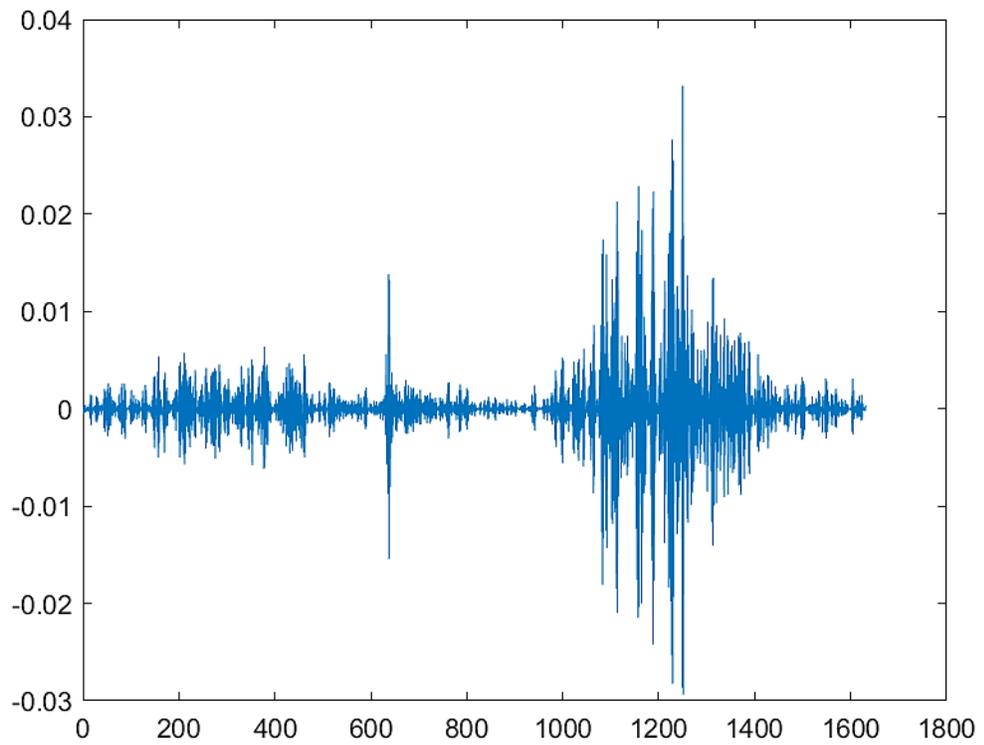


Figura 18. EPOC por sibilancia.

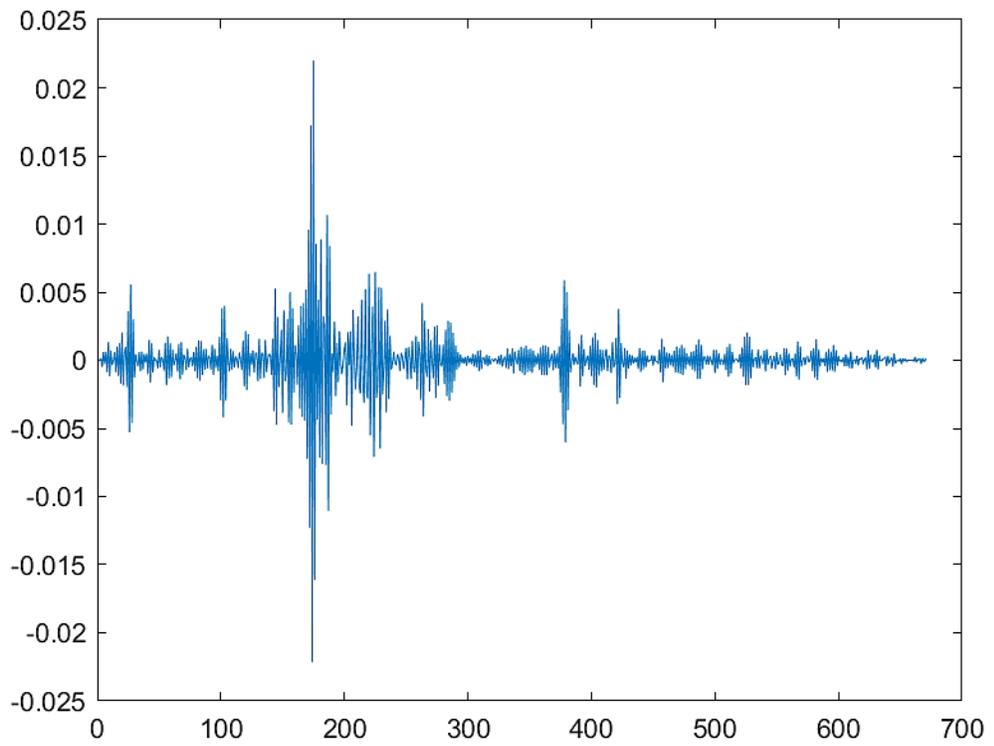


Figura 19. Neumonía por crepitantes.

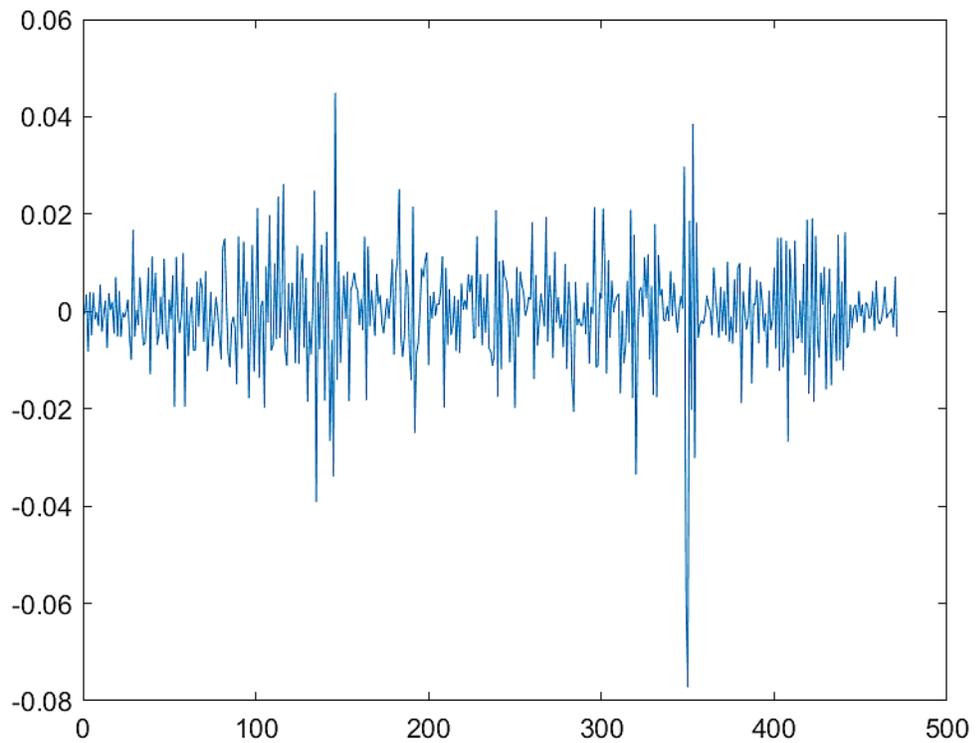


Figura 20. Murmullo pulmonar normal.

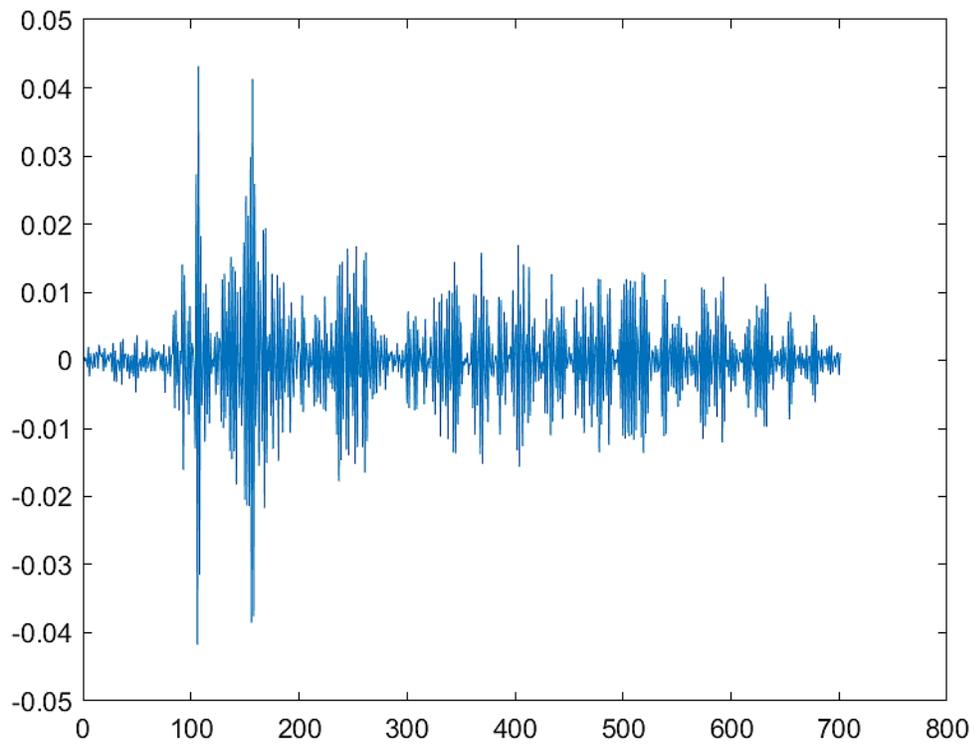


Figura 21. Fibrosis pulmonar por crepitantes.

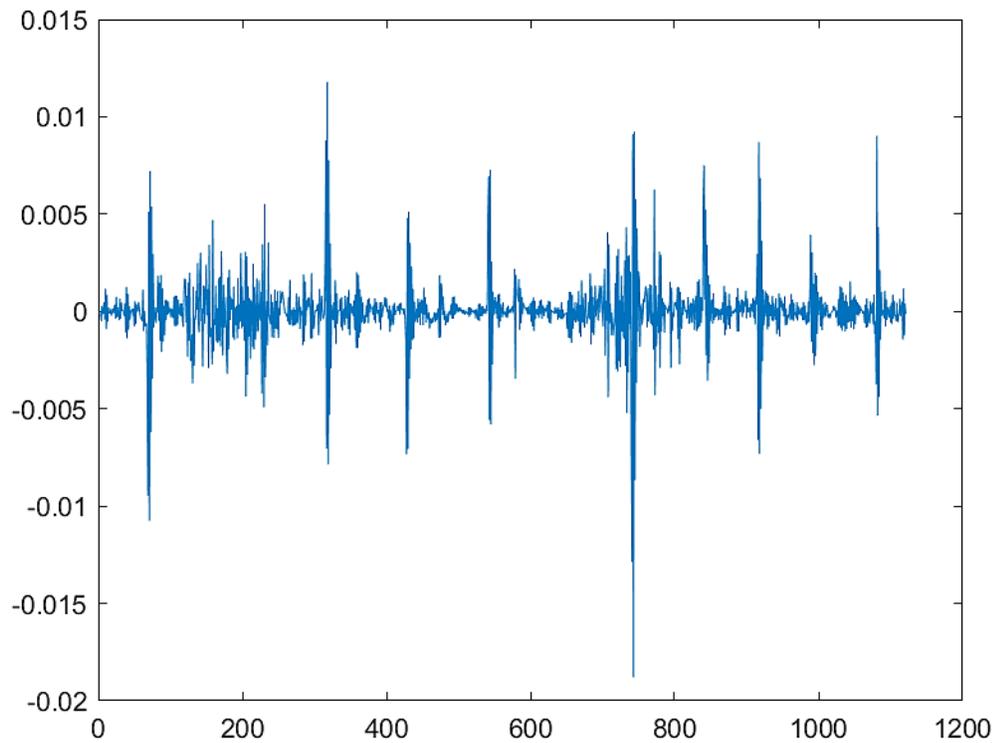


Figura 22. Efusión pleural por frote pleural.

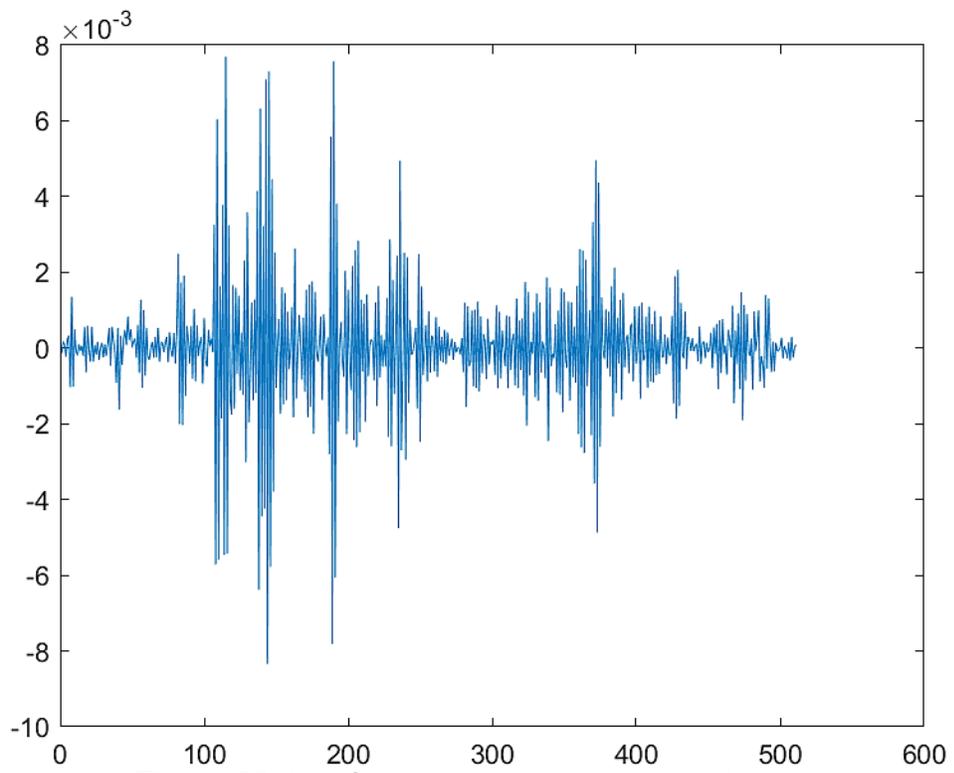


Figura 23. Insuficiencia cardiaca por crepitantes.

Teniendo identificadas las señales audio y categorizadas por tipo de sonido pulmonar y enfermedad que presentan, se escogieron 5 de cada una para obtener las 8 categorías para crear nuestra base de datos computarizada.

Para ello se hará uso de wavelet, creando imágenes de espectrogramas de 5 segundos de audio cada una de las señales de categoría.

Los parámetros wavelet escogidos por su mejor rendimiento fueron Wavelet Morlet (morl), con Sampling period "1", Step by step mode "1", desde "1" hasta "1024" en resolución, basada en "coeficientes" en escala "frecuencial", con mapa de color "jet" a nivel de color "128". Como se muestra a continuación:

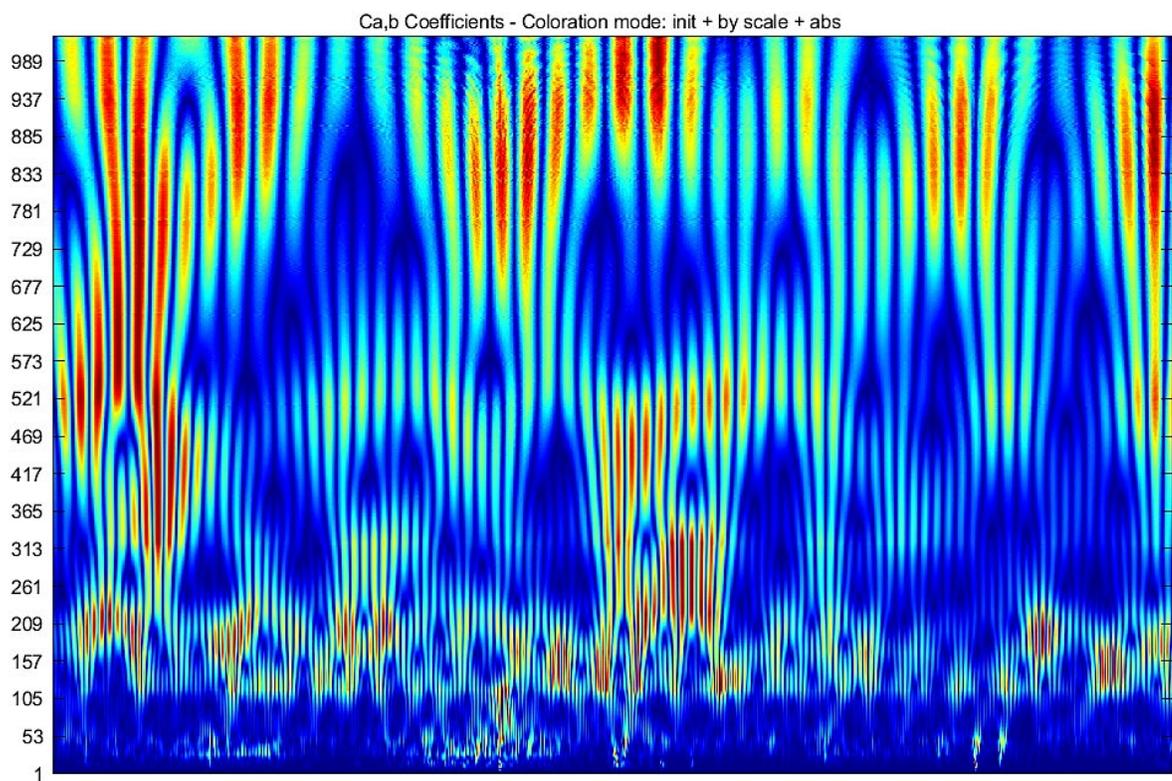


Figura 24. Espectrograma wavelet.

Al finalizar este proceso se obtuvieron 40 espectrogramas creando así una base de datos de 8 categorías cada una con 5 imágenes:

**Base de datos:**

- Asma, sibilancia ..... (5)
- Bronquitis, crepitantes ..... (5)
- Efusión pleural, frote pleural ..... (5)

- EPOC, sibilancia ..... (5)
- Fibrosis pulmonar, crepitantes ..... (5)
- Insuficiencia cardiaca, crepitantes ..... (5)
- Neumonía, crepitantes ..... (5)
- Normal ..... (5)

Teniendo la base de datos de espectrogramas de sonidos respiratorios, se hará uso de la inteligencia artificial para la clasificación y validación de los datos obtenidos y algunas muestras externas al proceso anterior.

La herramienta que se seleccionó por su desempeño y profundidad de aprendizaje fue la estructura de la red neuronal convolucional ResNet-50 modificando sus parámetros internos para adaptarla los requerimientos del proyecto. Esta frente a otras demostró en las pruebas y entrenamiento tener mejores resultados y mayor robustez requerida para el aprendizaje y la validación.

A continuación, se mostrará el diagrama de bloques del proyecto incluida la estructura convolucional de la ResNet-50.

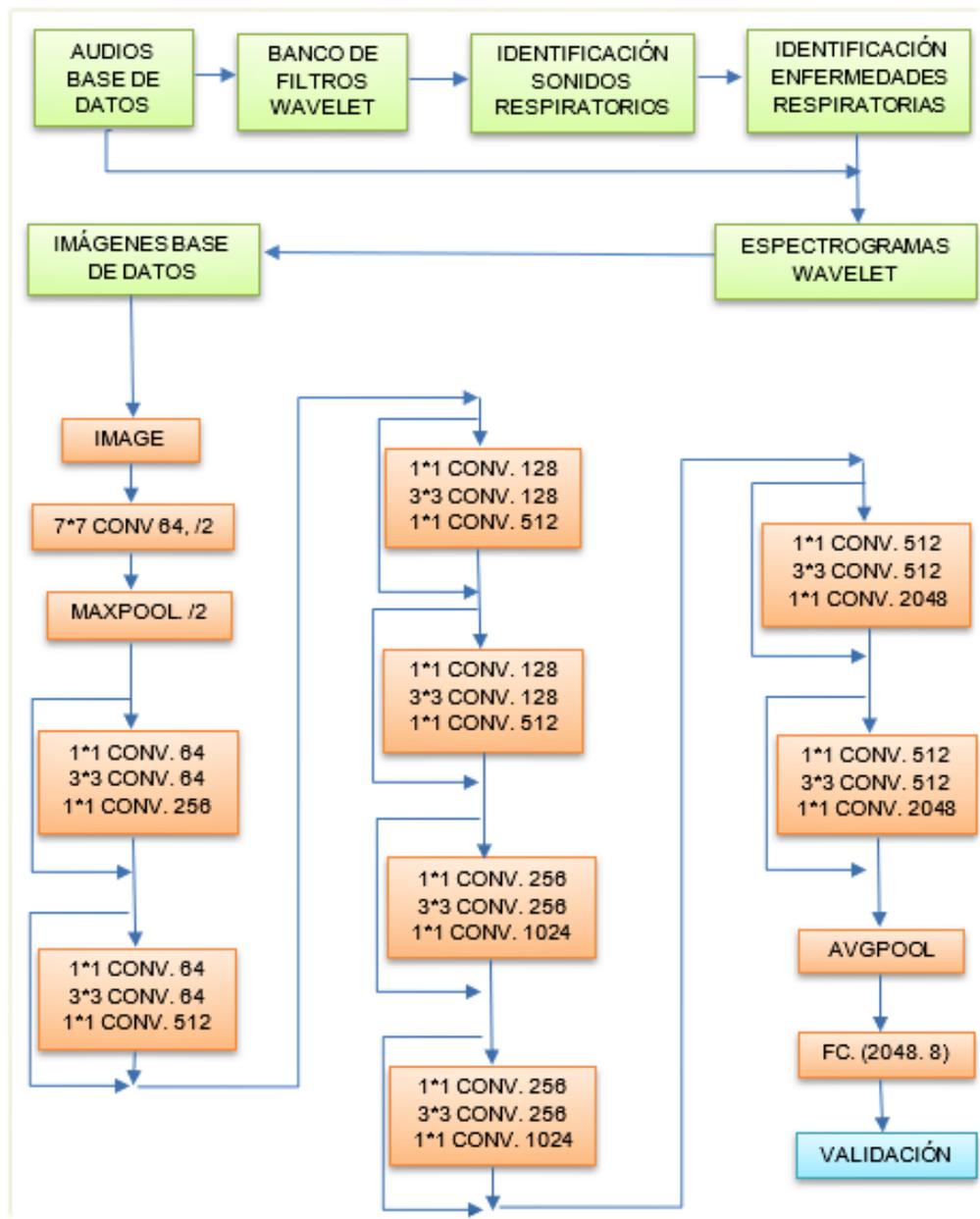


Figura 25. Diagrama de bloques.

**Actividad 4.** Desarrollo de pruebas para verificar su funcionamiento.

Para el desarrollo del proyecto se llevaron a cabo pruebas entre varios métodos y herramientas para mejorar y alcanzar el objetivo final.

En el desarrollo de las pruebas wavelet se analizaron los distintos tipos de wavelets madre y su configuración para realizar la descomposición frecuencial de las señales de audio.

WAVELE	ESCALA	NIVEL	%
db	10	5	95
coif	6	5	90
fk	8	5	83
haar	no	5	77
dmey	no	5	74
rbio	5.5	5	71
sym	8	5	70

Tabla 3. Prueba filtro Wavelet.

También se pretendían encontrar la mejor manera de transmitir y visualizar el espectro frecuencial wavelet en las imágenes de la base de datos, para esto se evaluaron distintos tipos de wavelets con distintas configuraciones:

WAVELE	ESCALA	RESOLUCIO	COLORMA	%
morl	10	1024	jet	96
db	10	1024	jet	94
sym	8	1024	jet	88
meyr	no	1024	hot	85
dmey	no	512	spring	84
mexh	no	1024	hot	81
coif	5	512	bone	76

Tabla 4. Prueba espectrograma wavelet.

Para el proceso de inteligencia artificial se efectuaron pruebas a varias configuraciones de redes neuronales convolucionales para identificar su rendimiento en el desarrollo de entrenamiento y aprendizaje.

CNN	% ENTRENAMIENTO	% VALIDACIÓN
resnet-50	96	98
shufflenet	84	85
eficientnet-b0	76	74
darknet-53	62	64
googlenet	60	63
alexnet	54	50

Tabla 5. Prueba red neuronal convolucional.

#### Actividad 5. Corrección de errores.

Para la corrección de errores se modificaron los parámetros de salida de la red neuronal convolucional para adaptarla a las 8 categorías requeridas para la clasificación y validación en este proyecto.

También se ajustaron los parámetros de tiempo de aprendizaje, número de etapas, frecuencia de validación y número de iteraciones por época.

**Actividad 6.** Detección y clasificación de posibles enfermedades respiratorias.

Para la clasificación y validación se ingresaron las 40 imágenes, 5 de ellas en cada una de las 8 categorías etiquetadas anteriormente.

La base de datos de imágenes se dividió en dos grupos, 70% para entrenamiento y 30% para validación. Dado esto se tienen 4 imágenes por categoría para entrenamiento, para un total de 32 imágenes para entrenamiento. Y 1 imagen por categoría para validación, para un total de 8 imágenes para validación, como se muestra a continuación.

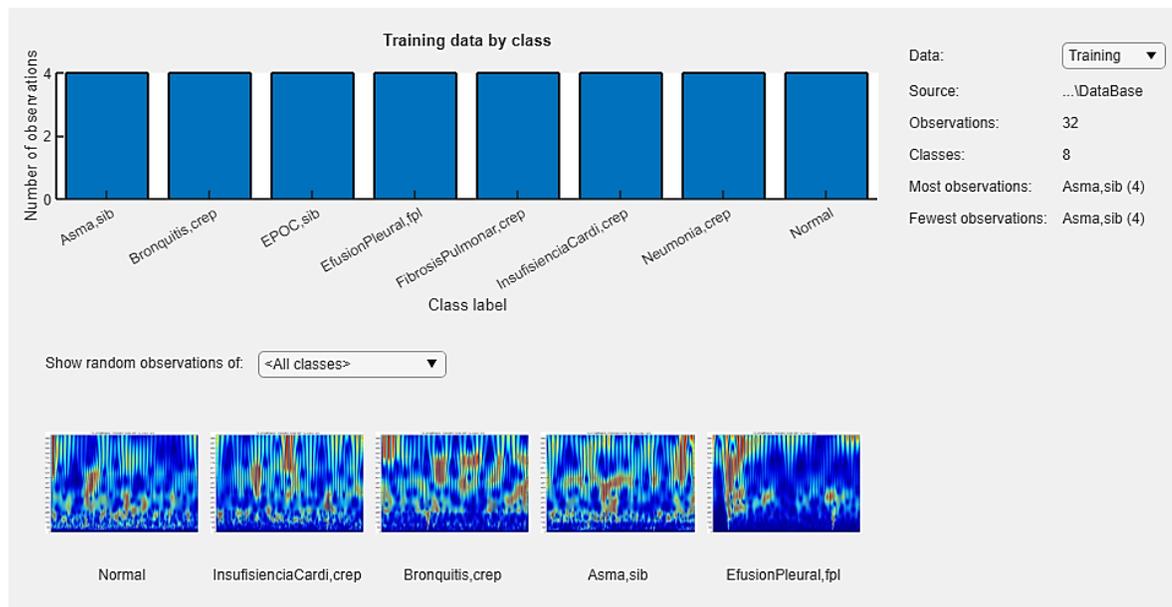


Figura 26. Categorías entrenamiento.

Aplicando el entrenamiento y validación se obtuvieron los siguientes resultados, donde la línea azul es el entrenamiento y la línea negra punteada es la validación.

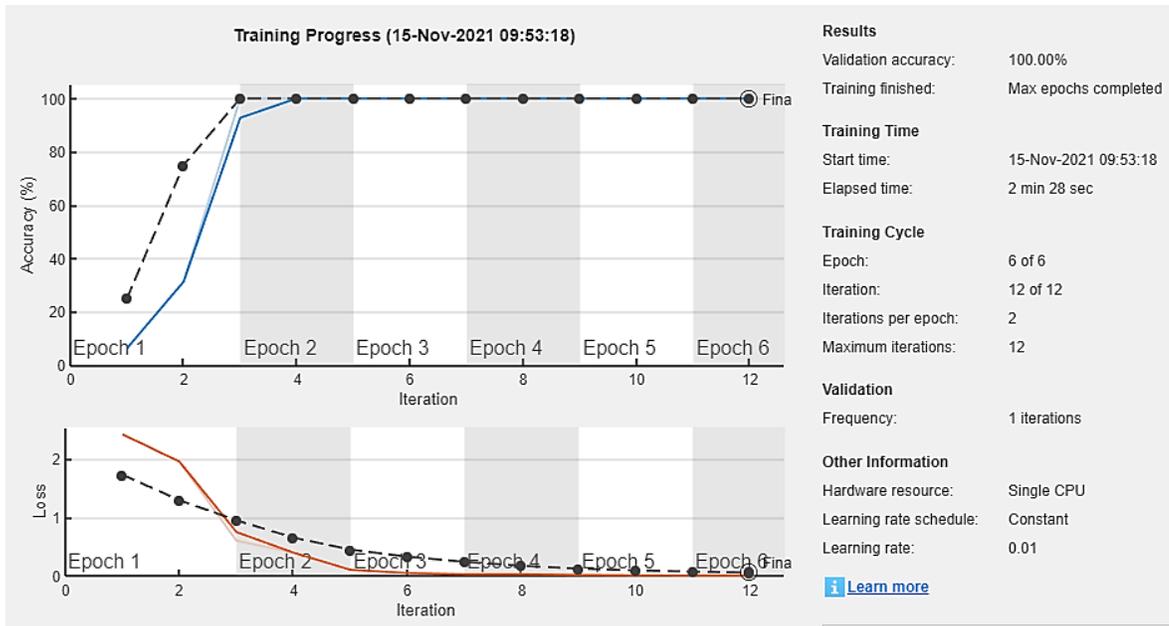


Figura 27. Gráfica de entrenamiento y validación base de datos.

Obteniendo un resultado de entrenamiento y validación del 100% de la base de datos.

Para la detección y validación de nuevas muestras se ingresaron las 40 imágenes de la base de datos, 5 de ellas en cada una de las 8 categorías, y una nueva carpeta con 8 imágenes de espectrogramas de audios no ingresados en los procesos anteriores.

En este caso el entrenamiento y la validación se dividieron en dos grupos, para el 70% del entrenamiento se usó el 100% de la base de datos correspondiente a 40 imágenes. y para el 30% validación se usó la carpeta con muestras nuevas. Dado esto se tienen 5 imágenes por categoría para entrenamiento, para un total de 40 imágenes para entrenamiento. Y 1 imagen por categoría para validación, para un total de 8 imágenes para validación, como se muestra a continuación.

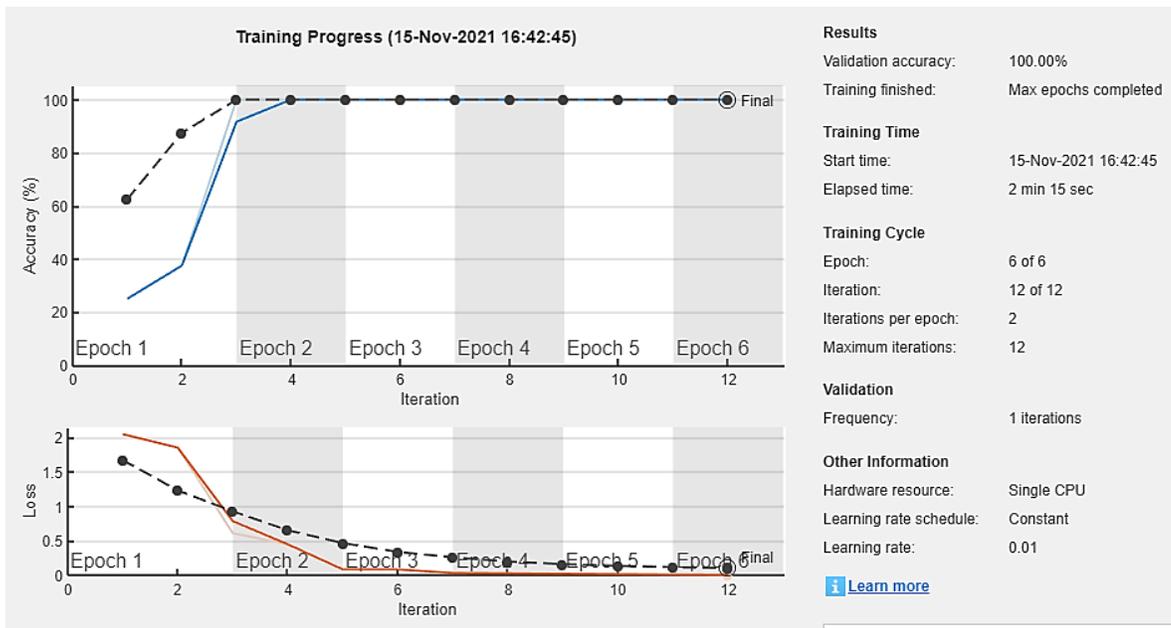


Figura 28. Gráfica de entrenamiento y validación muestras.

Obteniendo un resultado de entrenamiento y validación del 100% de las nuevas muestras ingresadas.

## 7. CONCLUSIONES Y RECOMENDACIONES

El proyecto realizado presenta avances significativos en la implementación de nuevas tecnologías en el campo de la salud, facilitando herramientas matemáticas computacionales que mejoran la clasificación e identificación de enfermedades respiratorias, dado como valor agregado un diagnóstico más amplio y preciso que beneficia directamente la salud de pacientes con enfermedades pulmonares y aporta una base de la cual se puede avanzar en procesos de investigación y desarrollo para generar nuevos o mejorados métodos que impactaran en la salud de todos.

Se demostró la importancia y capacidad de herramientas matemáticas como wavelet y redes neuronales convolucionales entre otras, que, aunque se crea que solo tienen una utilidad en áreas específicas como la edición de audio o reconocimiento facial, pueden ser implementadas en avances científicos e investigaciones que aporten a la mejora de la salud mundial.

Lastimosamente el enfoque o desarrollo que se le ha dado a las redes neuronales convolucionales las limita a trabajar directamente con imágenes, no existen procesos actualmente que involucren el Deep learning directamente con señales de audio, a excepción de algunas señales secuenciales, solo se pueden usar estos procesos imprimiendo las características acústicas del sonido en imágenes.

Al finalizar los procesos implementados se evidenció la clasificación e identificación de los grupos o categorías encontrados en la base de datos de sonidos respiratorios adquirida validando con un porcentaje alto y satisfactorio los procesos computacionales implementados que dejan como aporte una base de datos computarizada de identificación de 8 categorías de sonido pulmonar que puede ser usada en nuevos diagnósticos de enfermedades respiratorias en pacientes.

## 8. BIBLIOGRAFÍA

[1] Reiriz Palacios, Julia (2015), “Sistema respiratorio: Anatomía”, *Infermera Virtual*, Col·legi Oficial Infermeres I Infermers Barcelona, <https://www.infermeravirtual.com/files/media/file/97/Sistema%20respiratorio.pdf?1358605430>

[2] *Revista Digital*, MedlinePlus, “Enfermedad pulmonar”, <https://medlineplus.gov/spanish/ency/article/000066.htm>

[3] Sarkar, M., Madabhavi, I., Niranjana, N., & Dogra, M. (2015). Auscultation of the respiratory system. *Annals of thoracic medicine*, 10(3), 158–168. <https://doi.org/10.4103/1817-1737.160831>  
<https://www.thoracicmedicine.org/text.asp?2015/10/3/158/160831>

[4] *Revista Digital*, *Empendium*, “Ruidos respiratorios”, <https://empendium.com/manualmibe/compendio/chapter/B34.I.1.31>.

[5] *Enciclopedia Digital*, *Wikipedia*, “MATLAB”, <https://es.wikipedia.org/wiki/MATLAB>

[6] Ruiz Salazar, Jeferson; Orejuela Caicedo, David, A. (2016), “Implementación de la transformada wavelet sobre un sistema embebido para el pre-procesamiento de señales unidimensionales no estacionarias”, Universidad de san buenaventura, Cali, [http://bibliotecadigital.usbcali.edu.co/bitstream/10819/4278/1/Implementacion\\_transformada\\_wavelet\\_ruiz\\_2016.pdf](http://bibliotecadigital.usbcali.edu.co/bitstream/10819/4278/1/Implementacion_transformada_wavelet_ruiz_2016.pdf)

[7] Fraiwan, Mohammad; Fraiwan, Luay; Khassawneh, Basheer; Ibnian, Ali (2021), “A dataset of lung sounds recorded from the chest wall using an electronic stethoscope”, *Mendeley Data*, V3, doi: 10.17632/jwyy9np4gv.3, <https://data.mendeley.com/datasets/jwyy9np4gv/3>

[8] Mondragón, Francisco J., Pérez-Meana, Héctor M., Calderón, Gustavo, & Jiménez, Jonathan. (2021). Clasificación de sonidos ambientales usando la transformada wavelet continua y redes neuronales convolucionales. *Información tecnológica*, 32(2), 61-78. <https://dx.doi.org/10.4067/S0718-07642021000200061>

[9] García, Sánchez, Eugenio. (2019). "Introducción a las redes neuronales de convolución. Aplicación a la visión por ordenador". Universidad Zaragoza. <https://zagan.unizar.es/record/87398/files/TAZ-TFG-2019-3085.pdf>

[10] Revista Digital. Codificando Bits. (2021). "Reconocimiento Facial con Machine Learning: FaceNet y one-shot learning". <https://www.codificandobits.com/blog/reconocimiento-facial-machine-learning/>

[11] Revista digital, Opinión & Salud. Edición número 3509. ISSN: 2619-2144. [info@opinionysalud.com](mailto:info@opinionysalud.com). <https://www.opinionysalud.com/2019/04/12/aumentan-enfermedades-respiratorias-en-colombia/>

[12] Shanthakumari, G., & Priya, E. (2022). Interpretation of lung sounds using spectrogram-based statistical features doi:10.1007/978-981-16-4625-6\_81 Retrieved from [www.scopus.com](http://www.scopus.com)

[13] Sankararaman, S. (2022). Untangling the graph based features for lung sound auscultation. *Biomedical Signal Processing and Control*, 71 doi:10.1016/j.bspc.2021.103215

[14] Zhiguo Zhang, Mark A. Kon, Interpolatory filter banks and interpolatory wavelet packets, *Journal of Computational and Applied Mathematics*, Volume 374, 2020, 112755, ISSN 0377-0427, <https://doi.org/10.1016/j.cam.2020.112755>.

[15] David R. Bull, Fan Zhang, Chapter 6 - Filter-banks and wavelet compression, Editor(s): David R. Bull, Fan Zhang, Intelligent Image and Video Compression (Second Edition), Academic Press, 2021, Pages 183-224, ISBN 9780128203538, <https://doi.org/10.1016/B978-0-12-820353-8.00015-3>.

[16] D.C.J. Howell, SIGNS OF RESPIRATORY DISEASE | Lung Sounds, Editor(s): Geoffrey J. Laurent, Steven D. Shapiro, Encyclopedia of Respiratory Medicine, Academic Press, 2006, Pages 35-41, ISBN 9780123708793, <https://doi.org/10.1016/B0-12-370879-6/00355-0>.

[17] Mei, S., Geng, Y., Hou, J., & Du, Q. (2022). Learning hyperspectral images from RGB images via a coarse-to-fine CNN. Science China Information Sciences, 65(5) doi:10.1007/s11432-020-3102-9

[18] Leonardo Piñeyro, Alberto Pardo, Marcos Viera, Structure verification of deep neural networks at compilation time, Journal of Computer Languages, Volume 67, 2021, 101074, ISSN 2590-1184, <https://doi.org/10.1016/j.cola.2021.101074>.

[19] Ameya D. Jagtap, Yeonjong Shin, Kenji Kawaguchi, George Em Karniadakis, Deep Kronecker neural networks: A general framework for neural networks with adaptive activation functions, Neurocomputing, Volume 468, 2022, Pages 165-180, ISSN 0925-2312, <https://doi.org/10.1016/j.neucom.2021.10.036>.

[20] Wang, W., Ma, J., Xu, C., Zhang, Y., Ding, Y., Yu, S., . . . Liu, Y. (2022). Low-rank sparse feature selection for image classification. Expert Systems with Applications, 189 doi:10.1016/j.eswa.2021.115685

[21] Neha Sharma, Vibhor Jain, Anju Mishra, An Analysis Of Convolutional Neural Networks For Image Classification, Procedia Computer Science, Volume 132, 2018, Pages 377-384, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2018.05.198>.

[22] Yao, J., Wang, D., Hu, H., Xing, W., & Wang, L. (2022). ADCNN: Towards learning adaptive dilation for convolutional neural networks. *Pattern Recognition*, 123 doi:10.1016/j.patcog.2021.108369

[23] Revista del Instituto Nacional de Enfermedades Respiratorias. (2021). "Nuevas perspectivas en la evaluación automatizada de los sonidos respiratorios". Volumen 14. Número 4. Instituto Nacional de Enfermedades Respiratorias.

[24] Kim, Y., Hyon, Y., Jung, S.S. (2021). Respiratory sound classification for crackles, wheezes, and rhonchi in the clinical field using deep learning. *Sci Rep* 11, 17186. <https://doi.org/10.1038/s41598-021-96724-7>

[25] Revista Digital de Salud, El Útil. (2021). "Sonidos respiratorios anormales: Tipos y causas". <https://elutil.com/salud/sonidos-respiratorios-anormales-tipos-y-causas/>

## 9. ANEXOS

```
%%ingresar audio%%
[s fs] = audioread('EP8_P L L,70,M.wav');
sn = normalizar(s);
save(cd6.mat)
%% normalizar%%
function sonidoN=normalizar(sonido)
    maximo = max(abs(sonido));
    n= length(sonido);
    for i = 1:1:n
        sonidoN(i) = sonido(i)/maximo;
    end
end
%%guardar audio%%
load cd6.mat
audiowrite('cd6.wav',cd6,4000)
%%filtro wavlwt%%
[s fs] = audioread('EP8_P L L,70,M.wav');
sn = normalizar(s);
ss = sn(1:20000);
[c, l] = wavedec(ss, 6, 'db10');
approx = appcoef(c, l, 'db10');
[cd1,cd2,cd3,cd4,cd5,cd6] = detcoef(c,l,[1,2,3,4,5,6]);
subplot(8,1,1)
plot(ss)
title('señal original');
subplot(8,1,2)
plot(approx)
title('coeficientes de aproximacion');
subplot(8,1,3)
plot(cd6)
title('coeficientes de detalle - nivel 6');
subplot(8,1,4)
plot(cd5)
title('coeficientes de detalle - nivel 5');
subplot(8,1,5)
plot(cd4)
title('coeficientes de detalle - nivel 4');
subplot(8,1,6)
plot(cd3)
title('coeficientes de detalle - nivel 3');
subplot(8,1,7)
plot(cd2)
title('coeficientes de detalle - nivel 2');
subplot(8,1,8)
plot(cd1)
title('coeficientes de detalle - nivel 1');
%%recortar audio%%
DP8_P = audio(1:20000);
plot(DP8_P)
%%espectrograma%%
load handel
nfft = 512;
```

```

noverlap = 128;
win = hamming(nfft);
spectrogram(BP14_cref, win, noverlap, nfft, Fs, 'yaxis')
colormap('jet')
wname = 'morl'; % Choosing the wavelet
scales = 1:1:128; % Defining scales
coefs = cwt(BP2_Asthma, scales, wname); % Get wavelet coefficients
wscalogram('image', coefs, 'scales', scales, 'ydata', BP2_Asthma); % Get the scalogram, together
with time domain signal overlaid
colormap(jet) % Set the colormap
%% Script for creating and training a deep learning network with the following properties%%
Number of layers: 177
Number of connections: 192
Training setup file: D:\DOCUMENTOS\archivos\TRATAMIENTO\Trabajo de
grado\params_2021_11_11_14_06_41.mat
%% Load parameters for network initialization. For transfer learning, the network initialization
parameters are the parameters of the initial pretrained network%%
trainingSetup = load("D:\DOCUMENTOS\archivos\TRATAMIENTO\Trabajo de
grado\params_2021_11_11_14_06_41.mat");
%% Import training and validation data%%
imdsTrain = imageDatastore("D:\DOCUMENTOS\archivos\TRATAMIENTO\Trabajo de
grado\DataBase", "IncludeSubfolders", true, "LabelSource", "folderNames");
[imdsTrain, imdsValidation] = splitEachLabel(imdsTrain, 0.7);
% Resize the images to match the network input layer.
augimdsTrain = augmentedImageDatastore([224 224 3], imdsTrain);
augimdsValidation = augmentedImageDatastore([224 224 3], imdsValidation);
%% Specify options to use when training%%
opts = trainingOptions("sgdm", ...
    "ExecutionEnvironment", "auto", ...
    "InitialLearnRate", 0.01, ...
    "MaxEpochs", 6, ...
    "MiniBatchSize", 12, ...
    "Shuffle", "every-epoch", ...
    "ValidationFrequency", 1, ...
    "Plots", "training-progress", ...
    "ValidationData", augimdsValidation);
%% Create the layer graph variable to contain the network layers%%
lgraph = layerGraph();
%% Add the branches of the network to the layer graph. Each branch is a linear array of layers%%
tempLayers = [
    imageInputLayer([224 224 3], "Name", "input_1", "Mean", trainingSetup.input_1.Mean)
    convolution2dLayer([7 7], 64, "Name", "conv1", "Padding", [3 3 3 3], "Stride", [2
2], "Bias", trainingSetup.conv1.Bias, "Weights", trainingSetup.conv1.Weights)

    batchNormalizationLayer("Name", "bn_conv1", "Epsilon", 0.001, "Offset", trainingSetup.bn_conv1.Offset, "S
cale", trainingSetup.bn_conv1.Scale, "TrainedMean", trainingSetup.bn_conv1.TrainedMean, "TrainedVarianc
e", trainingSetup.bn_conv1.TrainedVariance)
    reluLayer("Name", "activation_1_relu")
    maxPooling2dLayer([3 3], "Name", "max_pooling2d_1", "Padding", [1 1 1 1], "Stride", [2 2]);
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1
1], 256, "Name", "res2a_branch1", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res2a_branch1.Bias, "Weig
hts", trainingSetup.res2a_branch1.Weights)

    batchNormalizationLayer("Name", "bn2a_branch1", "Epsilon", 0.001, "Offset", trainingSetup.bn2a_branch1.0

```

```

ffset,"Scale",trainingSetup.bn2a_branch1.Scale,"TrainedMean",trainingSetup.bn2a_branch1.TrainedMean
,"TrainedVariance",trainingSetup.bn2a_branch1.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    convolution2dLayer([1
1],64,"Name","res2a_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res2a_branch2a.Bias,"Wei
ghts",trainingSetup.res2a_branch2a.Weights)

    batchNormalizationLayer("Name","bn2a_branch2a","Epsilon",0.001,"Offset",trainingSetup.bn2a_branch2a
.Offset,"Scale",trainingSetup.bn2a_branch2a.Scale,"TrainedMean",trainingSetup.bn2a_branch2a.Trained
Mean,"TrainedVariance",trainingSetup.bn2a_branch2a.TrainedVariance)
    reluLayer("Name","activation_2_relu")
    convolution2dLayer([3
3],64,"Name","res2a_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res2a_b
ranch2b.Bias,"Weights",trainingSetup.res2a_branch2b.Weights)

    batchNormalizationLayer("Name","bn2a_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn2a_branch2b
.Offset,"Scale",trainingSetup.bn2a_branch2b.Scale,"TrainedMean",trainingSetup.bn2a_branch2b.Trained
Mean,"TrainedVariance",trainingSetup.bn2a_branch2b.TrainedVariance)
    reluLayer("Name","activation_3_relu")
    convolution2dLayer([1
1],256,"Name","res2a_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res2a_branch2c.Bias,"Wei
ghts",trainingSetup.res2a_branch2c.Weights)

    batchNormalizationLayer("Name","bn2a_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn2a_branch2c
.Offset,"Scale",trainingSetup.bn2a_branch2c.Scale,"TrainedMean",trainingSetup.bn2a_branch2c.Trained
Mean,"TrainedVariance",trainingSetup.bn2a_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    additionLayer(2,"Name","add_1")
    reluLayer("Name","activation_4_relu")];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    convolution2dLayer([1
1],64,"Name","res2b_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res2b_branch2a.Bias,"Wei
ghts",trainingSetup.res2b_branch2a.Weights)

    batchNormalizationLayer("Name","bn2b_branch2a","Epsilon",0.001,"Offset",trainingSetup.bn2b_branch2a
.Offset,"Scale",trainingSetup.bn2b_branch2a.Scale,"TrainedMean",trainingSetup.bn2b_branch2a.Trained
Mean,"TrainedVariance",trainingSetup.bn2b_branch2a.TrainedVariance)
    reluLayer("Name","activation_5_relu")
    convolution2dLayer([3
3],64,"Name","res2b_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res2b_b
ranch2b.Bias,"Weights",trainingSetup.res2b_branch2b.Weights)

    batchNormalizationLayer("Name","bn2b_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn2b_branch2b
.Offset,"Scale",trainingSetup.bn2b_branch2b.Scale,"TrainedMean",trainingSetup.bn2b_branch2b.Trained
Mean,"TrainedVariance",trainingSetup.bn2b_branch2b.TrainedVariance)
    reluLayer("Name","activation_6_relu")
    convolution2dLayer([1
1],256,"Name","res2b_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res2b_branch2c.Bias,"Wei
ghts",trainingSetup.res2b_branch2c.Weights)

    batchNormalizationLayer("Name","bn2b_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn2b_branch2c

```

```

.Offset, "Scale", trainingSetup.bn2b_branch2c.Scale, "TrainedMean", trainingSetup.bn2b_branch2c.Trained
Mean, "TrainedVariance", trainingSetup.bn2b_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    additionLayer(2, "Name", "add_2")
    reluLayer("Name", "activation_7_relu")];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1
1], 64, "Name", "res2c_branch2a", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res2c_branch2a.Bias, "Wei
ghts", trainingSetup.res2c_branch2a.Weights)

batchNormalizationLayer("Name", "bn2c_branch2a", "Epsilon", 0.001, "Offset", trainingSetup.bn2c_branch2a
.Offset, "Scale", trainingSetup.bn2c_branch2a.Scale, "TrainedMean", trainingSetup.bn2c_branch2a.Trained
Mean, "TrainedVariance", trainingSetup.bn2c_branch2a.TrainedVariance)
    reluLayer("Name", "activation_8_relu")
    convolution2dLayer([3
3], 64, "Name", "res2c_branch2b", "BiasLearnRateFactor", 0, "Padding", "same", "Bias", trainingSetup.res2c_b
ranch2b.Bias, "Weights", trainingSetup.res2c_branch2b.Weights)

batchNormalizationLayer("Name", "bn2c_branch2b", "Epsilon", 0.001, "Offset", trainingSetup.bn2c_branch2b
.Offset, "Scale", trainingSetup.bn2c_branch2b.Scale, "TrainedMean", trainingSetup.bn2c_branch2b.Trained
Mean, "TrainedVariance", trainingSetup.bn2c_branch2b.TrainedVariance)
    reluLayer("Name", "activation_9_relu")
    convolution2dLayer([1
1], 256, "Name", "res2c_branch2c", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res2c_branch2c.Bias, "We
ights", trainingSetup.res2c_branch2c.Weights)

batchNormalizationLayer("Name", "bn2c_branch2c", "Epsilon", 0.001, "Offset", trainingSetup.bn2c_branch2c
.Offset, "Scale", trainingSetup.bn2c_branch2c.Scale, "TrainedMean", trainingSetup.bn2c_branch2c.Trained
Mean, "TrainedVariance", trainingSetup.bn2c_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    additionLayer(2, "Name", "add_3")
    reluLayer("Name", "activation_10_relu")];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1 1], 128, "Name", "res3a_branch2a", "BiasLearnRateFactor", 0, "Stride", [2
2], "Bias", trainingSetup.res3a_branch2a.Bias, "Weights", trainingSetup.res3a_branch2a.Weights)

batchNormalizationLayer("Name", "bn3a_branch2a", "Epsilon", 0.001, "Offset", trainingSetup.bn3a_branch2a
.Offset, "Scale", trainingSetup.bn3a_branch2a.Scale, "TrainedMean", trainingSetup.bn3a_branch2a.Trained
Mean, "TrainedVariance", trainingSetup.bn3a_branch2a.TrainedVariance)
    reluLayer("Name", "activation_11_relu")
    convolution2dLayer([3
3], 128, "Name", "res3a_branch2b", "BiasLearnRateFactor", 0, "Padding", "same", "Bias", trainingSetup.res3a_
branch2b.Bias, "Weights", trainingSetup.res3a_branch2b.Weights)

batchNormalizationLayer("Name", "bn3a_branch2b", "Epsilon", 0.001, "Offset", trainingSetup.bn3a_branch2b
.Offset, "Scale", trainingSetup.bn3a_branch2b.Scale, "TrainedMean", trainingSetup.bn3a_branch2b.Trained
Mean, "TrainedVariance", trainingSetup.bn3a_branch2b.TrainedVariance)
    reluLayer("Name", "activation_12_relu")

```

```

        convolution2dLayer([1
1],512,"Name","res3a_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res3a_branch2c.Bias,"Weights",trainingSetup.res3a_branch2c.Weights)

batchNormalizationLayer("Name","bn3a_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn3a_branch2c
.Offset,"Scale",trainingSetup.bn3a_branch2c.Scale,"TrainedMean",trainingSetup.bn3a_branch2c.TrainedMean,"TrainedVariance",trainingSetup.bn3a_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    convolution2dLayer([1 1],512,"Name","res3a_branch1","BiasLearnRateFactor",0,"Stride",[2
2],"Bias",trainingSetup.res3a_branch1.Bias,"Weights",trainingSetup.res3a_branch1.Weights)

batchNormalizationLayer("Name","bn3a_branch1","Epsilon",0.001,"Offset",trainingSetup.bn3a_branch1.O
ffset,"Scale",trainingSetup.bn3a_branch1.Scale,"TrainedMean",trainingSetup.bn3a_branch1.TrainedMean,"TrainedVariance",trainingSetup.bn3a_branch1.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    additionLayer(2,"Name","add_4")
    reluLayer("Name","activation_13_relu)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    convolution2dLayer([1
1],128,"Name","res3b_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res3b_branch2a.Bias,"Weights",trainingSetup.res3b_branch2a.Weights)

batchNormalizationLayer("Name","bn3b_branch2a","Epsilon",0.001,"Offset",trainingSetup.bn3b_branch2a
.Offset,"Scale",trainingSetup.bn3b_branch2a.Scale,"TrainedMean",trainingSetup.bn3b_branch2a.TrainedMean,"TrainedVariance",trainingSetup.bn3b_branch2a.TrainedVariance)
    reluLayer("Name","activation_14_relu")
    convolution2dLayer([3
3],128,"Name","res3b_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res3b_
branch2b.Bias,"Weights",trainingSetup.res3b_branch2b.Weights)

batchNormalizationLayer("Name","bn3b_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn3b_branch2b
.Offset,"Scale",trainingSetup.bn3b_branch2b.Scale,"TrainedMean",trainingSetup.bn3b_branch2b.TrainedMean,"TrainedVariance",trainingSetup.bn3b_branch2b.TrainedVariance)
    reluLayer("Name","activation_15_relu")
    convolution2dLayer([1
1],512,"Name","res3b_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res3b_branch2c.Bias,"Weights",trainingSetup.res3b_branch2c.Weights)

batchNormalizationLayer("Name","bn3b_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn3b_branch2c
.Offset,"Scale",trainingSetup.bn3b_branch2c.Scale,"TrainedMean",trainingSetup.bn3b_branch2c.TrainedMean,"TrainedVariance",trainingSetup.bn3b_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    additionLayer(2,"Name","add_5")
    reluLayer("Name","activation_16_relu)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [

```

```

convolution2dLayer([1
1],128,"Name","res3c_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res3c_branch2a.Bias,"Weights",trainingSetup.res3c_branch2a.Weights)

batchNormalizationLayer("Name","bn3c_branch2a","Epsilon",0.001,"Offset",trainingSetup.bn3c_branch2a
.Offset,"Scale",trainingSetup.bn3c_branch2a.Scale,"TrainedMean",trainingSetup.bn3c_branch2a.Trained
Mean,"TrainedVariance",trainingSetup.bn3c_branch2a.TrainedVariance)
reluLayer("Name","activation_17_relu")
convolution2dLayer([3
3],128,"Name","res3c_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res3c_
branch2b.Bias,"Weights",trainingSetup.res3c_branch2b.Weights)

batchNormalizationLayer("Name","bn3c_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn3c_branch2b
.Offset,"Scale",trainingSetup.bn3c_branch2b.Scale,"TrainedMean",trainingSetup.bn3c_branch2b.Trained
Mean,"TrainedVariance",trainingSetup.bn3c_branch2b.TrainedVariance)
reluLayer("Name","activation_18_relu")
convolution2dLayer([1
1],512,"Name","res3c_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res3c_branch2c.Bias,"We
ights",trainingSetup.res3c_branch2c.Weights)

batchNormalizationLayer("Name","bn3c_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn3c_branch2c
.Offset,"Scale",trainingSetup.bn3c_branch2c.Scale,"TrainedMean",trainingSetup.bn3c_branch2c.Trained
Mean,"TrainedVariance",trainingSetup.bn3c_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
additionLayer(2,"Name","add_6")
reluLayer("Name","activation_19_relu");
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
convolution2dLayer([1
1],128,"Name","res3d_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res3d_branch2a.Bias,"We
ights",trainingSetup.res3d_branch2a.Weights)

batchNormalizationLayer("Name","bn3d_branch2a","Epsilon",0.001,"Offset",trainingSetup.bn3d_branch2a
.Offset,"Scale",trainingSetup.bn3d_branch2a.Scale,"TrainedMean",trainingSetup.bn3d_branch2a.Trained
Mean,"TrainedVariance",trainingSetup.bn3d_branch2a.TrainedVariance)
reluLayer("Name","activation_20_relu")
convolution2dLayer([3
3],128,"Name","res3d_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res3d_
branch2b.Bias,"Weights",trainingSetup.res3d_branch2b.Weights)

batchNormalizationLayer("Name","bn3d_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn3d_branch2b
.Offset,"Scale",trainingSetup.bn3d_branch2b.Scale,"TrainedMean",trainingSetup.bn3d_branch2b.Trained
Mean,"TrainedVariance",trainingSetup.bn3d_branch2b.TrainedVariance)
reluLayer("Name","activation_21_relu")
convolution2dLayer([1
1],512,"Name","res3d_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res3d_branch2c.Bias,"We
ights",trainingSetup.res3d_branch2c.Weights)

batchNormalizationLayer("Name","bn3d_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn3d_branch2c
.Offset,"Scale",trainingSetup.bn3d_branch2c.Scale,"TrainedMean",trainingSetup.bn3d_branch2c.Trained
Mean,"TrainedVariance",trainingSetup.bn3d_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [

```

```

        additionLayer(2, "Name", "add_7")
        reluLayer("Name", "activation_22_relu");
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1 1], 256, "Name", "res4a_branch2a", "BiasLearnRateFactor", 0, "Stride", [2
2], "Bias", trainingSetup.res4a_branch2a.Bias, "Weights", trainingSetup.res4a_branch2a.Weights)

    batchNormalizationLayer("Name", "bn4a_branch2a", "Epsilon", 0.001, "Offset", trainingSetup.bn4a_branch2a
.Offset, "Scale", trainingSetup.bn4a_branch2a.Scale, "TrainedMean", trainingSetup.bn4a_branch2a.Trained
Mean, "TrainedVariance", trainingSetup.bn4a_branch2a.TrainedVariance)
    reluLayer("Name", "activation_23_relu")
    convolution2dLayer([3
3], 256, "Name", "res4a_branch2b", "BiasLearnRateFactor", 0, "Padding", "same", "Bias", trainingSetup.res4a_
branch2b.Bias, "Weights", trainingSetup.res4a_branch2b.Weights)

    batchNormalizationLayer("Name", "bn4a_branch2b", "Epsilon", 0.001, "Offset", trainingSetup.bn4a_branch2b
.Offset, "Scale", trainingSetup.bn4a_branch2b.Scale, "TrainedMean", trainingSetup.bn4a_branch2b.Trained
Mean, "TrainedVariance", trainingSetup.bn4a_branch2b.TrainedVariance)
    reluLayer("Name", "activation_24_relu")
    convolution2dLayer([1
1], 1024, "Name", "res4a_branch2c", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res4a_branch2c.Bias, "W
eights", trainingSetup.res4a_branch2c.Weights)

    batchNormalizationLayer("Name", "bn4a_branch2c", "Epsilon", 0.001, "Offset", trainingSetup.bn4a_branch2c
.Offset, "Scale", trainingSetup.bn4a_branch2c.Scale, "TrainedMean", trainingSetup.bn4a_branch2c.Trained
Mean, "TrainedVariance", trainingSetup.bn4a_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1 1], 1024, "Name", "res4a_branch1", "BiasLearnRateFactor", 0, "Stride", [2
2], "Bias", trainingSetup.res4a_branch1.Bias, "Weights", trainingSetup.res4a_branch1.Weights)

    batchNormalizationLayer("Name", "bn4a_branch1", "Epsilon", 0.001, "Offset", trainingSetup.bn4a_branch1.O
ffset, "Scale", trainingSetup.bn4a_branch1.Scale, "TrainedMean", trainingSetup.bn4a_branch1.TrainedMean
, "TrainedVariance", trainingSetup.bn4a_branch1.TrainedVariance)];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    additionLayer(2, "Name", "add_8")
    reluLayer("Name", "activation_25_relu");
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1
1], 256, "Name", "res4b_branch2a", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res4b_branch2a.Bias, "We
ights", trainingSetup.res4b_branch2a.Weights)

    batchNormalizationLayer("Name", "bn4b_branch2a", "Epsilon", 0.001, "Offset", trainingSetup.bn4b_branch2a
.Offset, "Scale", trainingSetup.bn4b_branch2a.Scale, "TrainedMean", trainingSetup.bn4b_branch2a.Trained
Mean, "TrainedVariance", trainingSetup.bn4b_branch2a.TrainedVariance)
    reluLayer("Name", "activation_26_relu")
    convolution2dLayer([3
3], 256, "Name", "res4b_branch2b", "BiasLearnRateFactor", 0, "Padding", "same", "Bias", trainingSetup.res4b_
branch2b.Bias, "Weights", trainingSetup.res4b_branch2b.Weights)

    batchNormalizationLayer("Name", "bn4b_branch2b", "Epsilon", 0.001, "Offset", trainingSetup.bn4b_branch2b

```

```

.Offset, "Scale", trainingSetup.bn4b_branch2b.Scale, "TrainedMean", trainingSetup.bn4b_branch2b.Trained
Mean, "TrainedVariance", trainingSetup.bn4b_branch2b.TrainedVariance)
    reluLayer("Name", "activation_27_relu")
    convolution2dLayer([1
1], 1024, "Name", "res4b_branch2c", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res4b_branch2c.Bias, "W
eights", trainingSetup.res4b_branch2c.Weights)

batchNormalizationLayer("Name", "bn4b_branch2c", "Epsilon", 0.001, "Offset", trainingSetup.bn4b_branch2c
.Offset, "Scale", trainingSetup.bn4b_branch2c.Scale, "TrainedMean", trainingSetup.bn4b_branch2c.Trained
Mean, "TrainedVariance", trainingSetup.bn4b_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    additionLayer(2, "Name", "add_9")
    reluLayer("Name", "activation_28_relu")];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1
1], 256, "Name", "res4c_branch2a", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res4c_branch2a.Bias, "We
ights", trainingSetup.res4c_branch2a.Weights)

batchNormalizationLayer("Name", "bn4c_branch2a", "Epsilon", 0.001, "Offset", trainingSetup.bn4c_branch2a
.Offset, "Scale", trainingSetup.bn4c_branch2a.Scale, "TrainedMean", trainingSetup.bn4c_branch2a.Trained
Mean, "TrainedVariance", trainingSetup.bn4c_branch2a.TrainedVariance)
    reluLayer("Name", "activation_29_relu")
    convolution2dLayer([3
3], 256, "Name", "res4c_branch2b", "BiasLearnRateFactor", 0, "Padding", "same", "Bias", trainingSetup.res4c_
branch2b.Bias, "Weights", trainingSetup.res4c_branch2b.Weights)

batchNormalizationLayer("Name", "bn4c_branch2b", "Epsilon", 0.001, "Offset", trainingSetup.bn4c_branch2b
.Offset, "Scale", trainingSetup.bn4c_branch2b.Scale, "TrainedMean", trainingSetup.bn4c_branch2b.Trained
Mean, "TrainedVariance", trainingSetup.bn4c_branch2b.TrainedVariance)
    reluLayer("Name", "activation_30_relu")
    convolution2dLayer([1
1], 1024, "Name", "res4c_branch2c", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res4c_branch2c.Bias, "W
eights", trainingSetup.res4c_branch2c.Weights)

batchNormalizationLayer("Name", "bn4c_branch2c", "Epsilon", 0.001, "Offset", trainingSetup.bn4c_branch2c
.Offset, "Scale", trainingSetup.bn4c_branch2c.Scale, "TrainedMean", trainingSetup.bn4c_branch2c.Trained
Mean, "TrainedVariance", trainingSetup.bn4c_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    additionLayer(2, "Name", "add_10")
    reluLayer("Name", "activation_31_relu")];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1
1], 256, "Name", "res4d_branch2a", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res4d_branch2a.Bias, "We
ights", trainingSetup.res4d_branch2a.Weights)

batchNormalizationLayer("Name", "bn4d_branch2a", "Epsilon", 0.001, "Offset", trainingSetup.bn4d_branch2a
.Offset, "Scale", trainingSetup.bn4d_branch2a.Scale, "TrainedMean", trainingSetup.bn4d_branch2a.Trained
Mean, "TrainedVariance", trainingSetup.bn4d_branch2a.TrainedVariance)
    reluLayer("Name", "activation_32_relu")

```

```

convolution2dLayer([3
3],256,"Name","res4d_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res4d_
branch2b.Bias,"Weights",trainingSetup.res4d_branch2b.Weights)

batchNormalizationLayer("Name","bn4d_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn4d_branch2b
.Offset,"Scale",trainingSetup.bn4d_branch2b.Scale,"TrainedMean",trainingSetup.bn4d_branch2b.Trained
Mean,"TrainedVariance",trainingSetup.bn4d_branch2b.TrainedVariance)
reluLayer("Name","activation_33_relu")
convolution2dLayer([1
1],1024,"Name","res4d_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res4d_branch2c.Bias,"W
eights",trainingSetup.res4d_branch2c.Weights)

batchNormalizationLayer("Name","bn4d_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn4d_branch2c
.Offset,"Scale",trainingSetup.bn4d_branch2c.Scale,"TrainedMean",trainingSetup.bn4d_branch2c.Trained
Mean,"TrainedVariance",trainingSetup.bn4d_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
additionLayer(2,"Name","add_11")
reluLayer("Name","activation_34_relu")];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
convolution2dLayer([1
1],256,"Name","res4e_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res4e_branch2a.Bias,"We
ights",trainingSetup.res4e_branch2a.Weights)

batchNormalizationLayer("Name","bn4e_branch2a","Epsilon",0.001,"Offset",trainingSetup.bn4e_branch2a
.Offset,"Scale",trainingSetup.bn4e_branch2a.Scale,"TrainedMean",trainingSetup.bn4e_branch2a.Trained
Mean,"TrainedVariance",trainingSetup.bn4e_branch2a.TrainedVariance)
reluLayer("Name","activation_35_relu")
convolution2dLayer([3
3],256,"Name","res4e_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res4e_
branch2b.Bias,"Weights",trainingSetup.res4e_branch2b.Weights)

batchNormalizationLayer("Name","bn4e_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn4e_branch2b
.Offset,"Scale",trainingSetup.bn4e_branch2b.Scale,"TrainedMean",trainingSetup.bn4e_branch2b.Trained
Mean,"TrainedVariance",trainingSetup.bn4e_branch2b.TrainedVariance)
reluLayer("Name","activation_36_relu")
convolution2dLayer([1
1],1024,"Name","res4e_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res4e_branch2c.Bias,"W
eights",trainingSetup.res4e_branch2c.Weights)

batchNormalizationLayer("Name","bn4e_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn4e_branch2c
.Offset,"Scale",trainingSetup.bn4e_branch2c.Scale,"TrainedMean",trainingSetup.bn4e_branch2c.Trained
Mean,"TrainedVariance",trainingSetup.bn4e_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
additionLayer(2,"Name","add_12")
reluLayer("Name","activation_37_relu")];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
convolution2dLayer([1
1],256,"Name","res4f_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res4f_branch2a.Bias,"We
ights",trainingSetup.res4f_branch2a.Weights)

```

```

batchNormalizationLayer("Name", "bn4f_branch2a", "Epsilon", 0.001, "Offset", trainingSetup.bn4f_branch2a
.Offset, "Scale", trainingSetup.bn4f_branch2a.Scale, "TrainedMean", trainingSetup.bn4f_branch2a.Trained
Mean, "TrainedVariance", trainingSetup.bn4f_branch2a.TrainedVariance)
    reluLayer("Name", "activation_38_relu")
    convolution2dLayer([3
3], 256, "Name", "res4f_branch2b", "BiasLearnRateFactor", 0, "Padding", "same", "Bias", trainingSetup.res4f_
branch2b.Bias, "Weights", trainingSetup.res4f_branch2b.Weights)

batchNormalizationLayer("Name", "bn4f_branch2b", "Epsilon", 0.001, "Offset", trainingSetup.bn4f_branch2b
.Offset, "Scale", trainingSetup.bn4f_branch2b.Scale, "TrainedMean", trainingSetup.bn4f_branch2b.Trained
Mean, "TrainedVariance", trainingSetup.bn4f_branch2b.TrainedVariance)
    reluLayer("Name", "activation_39_relu")
    convolution2dLayer([1
1], 1024, "Name", "res4f_branch2c", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res4f_branch2c.Bias, "W
eights", trainingSetup.res4f_branch2c.Weights)

batchNormalizationLayer("Name", "bn4f_branch2c", "Epsilon", 0.001, "Offset", trainingSetup.bn4f_branch2c
.Offset, "Scale", trainingSetup.bn4f_branch2c.Scale, "TrainedMean", trainingSetup.bn4f_branch2c.Trained
Mean, "TrainedVariance", trainingSetup.bn4f_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    additionLayer(2, "Name", "add_13")
    reluLayer("Name", "activation_40_relu")];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1 1], 512, "Name", "res5a_branch2a", "BiasLearnRateFactor", 0, "Stride", [2
2], "Bias", trainingSetup.res5a_branch2a.Bias, "Weights", trainingSetup.res5a_branch2a.Weights)

batchNormalizationLayer("Name", "bn5a_branch2a", "Epsilon", 0.001, "Offset", trainingSetup.bn5a_branch2a
.Offset, "Scale", trainingSetup.bn5a_branch2a.Scale, "TrainedMean", trainingSetup.bn5a_branch2a.Trained
Mean, "TrainedVariance", trainingSetup.bn5a_branch2a.TrainedVariance)
    reluLayer("Name", "activation_41_relu")
    convolution2dLayer([3
3], 512, "Name", "res5a_branch2b", "BiasLearnRateFactor", 0, "Padding", "same", "Bias", trainingSetup.res5a_
branch2b.Bias, "Weights", trainingSetup.res5a_branch2b.Weights)

batchNormalizationLayer("Name", "bn5a_branch2b", "Epsilon", 0.001, "Offset", trainingSetup.bn5a_branch2b
.Offset, "Scale", trainingSetup.bn5a_branch2b.Scale, "TrainedMean", trainingSetup.bn5a_branch2b.Trained
Mean, "TrainedVariance", trainingSetup.bn5a_branch2b.TrainedVariance)
    reluLayer("Name", "activation_42_relu")
    convolution2dLayer([1
1], 2048, "Name", "res5a_branch2c", "BiasLearnRateFactor", 0, "Bias", trainingSetup.res5a_branch2c.Bias, "W
eights", trainingSetup.res5a_branch2c.Weights)

batchNormalizationLayer("Name", "bn5a_branch2c", "Epsilon", 0.001, "Offset", trainingSetup.bn5a_branch2c
.Offset, "Scale", trainingSetup.bn5a_branch2c.Scale, "TrainedMean", trainingSetup.bn5a_branch2c.Trained
Mean, "TrainedVariance", trainingSetup.bn5a_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph, tempLayers);

tempLayers = [
    convolution2dLayer([1 1], 2048, "Name", "res5a_branch1", "BiasLearnRateFactor", 0, "Stride", [2
2], "Bias", trainingSetup.res5a_branch1.Bias, "Weights", trainingSetup.res5a_branch1.Weights)

batchNormalizationLayer("Name", "bn5a_branch1", "Epsilon", 0.001, "Offset", trainingSetup.bn5a_branch1.0

```

```

ffset,"Scale",trainingSetup.bn5a_branch1.Scale,"TrainedMean",trainingSetup.bn5a_branch1.TrainedMean
,"TrainedVariance",trainingSetup.bn5a_branch1.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    additionLayer(2,"Name","add_14")
    reluLayer("Name","activation_43_relu");
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    convolution2dLayer([1
1],512,"Name","res5b_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res5b_branch2a.Bias,"We
ights",trainingSetup.res5b_branch2a.Weights)

batchNormalizationLayer("Name","bn5b_branch2a","Epsilon",0.001,"Offset",trainingSetup.bn5b_branch2a
.Offset,"Scale",trainingSetup.bn5b_branch2a.Scale,"TrainedMean",trainingSetup.bn5b_branch2a.Trained
Mean,"TrainedVariance",trainingSetup.bn5b_branch2a.TrainedVariance)
    reluLayer("Name","activation_44_relu")
    convolution2dLayer([3
3],512,"Name","res5b_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res5b_
branch2b.Bias,"Weights",trainingSetup.res5b_branch2b.Weights)

batchNormalizationLayer("Name","bn5b_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn5b_branch2b
.Offset,"Scale",trainingSetup.bn5b_branch2b.Scale,"TrainedMean",trainingSetup.bn5b_branch2b.Trained
Mean,"TrainedVariance",trainingSetup.bn5b_branch2b.TrainedVariance)
    reluLayer("Name","activation_45_relu")
    convolution2dLayer([1
1],2048,"Name","res5b_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res5b_branch2c.Bias,"W
eights",trainingSetup.res5b_branch2c.Weights)

batchNormalizationLayer("Name","bn5b_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn5b_branch2c
.Offset,"Scale",trainingSetup.bn5b_branch2c.Scale,"TrainedMean",trainingSetup.bn5b_branch2c.Trained
Mean,"TrainedVariance",trainingSetup.bn5b_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    additionLayer(2,"Name","add_15")
    reluLayer("Name","activation_46_relu");
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    convolution2dLayer([1
1],512,"Name","res5c_branch2a","BiasLearnRateFactor",0,"Bias",trainingSetup.res5c_branch2a.Bias,"We
ights",trainingSetup.res5c_branch2a.Weights)

batchNormalizationLayer("Name","bn5c_branch2a","Epsilon",0.001,"Offset",trainingSetup.bn5c_branch2a
.Offset,"Scale",trainingSetup.bn5c_branch2a.Scale,"TrainedMean",trainingSetup.bn5c_branch2a.Trained
Mean,"TrainedVariance",trainingSetup.bn5c_branch2a.TrainedVariance)
    reluLayer("Name","activation_47_relu")
    convolution2dLayer([3
3],512,"Name","res5c_branch2b","BiasLearnRateFactor",0,"Padding","same","Bias",trainingSetup.res5c_
branch2b.Bias,"Weights",trainingSetup.res5c_branch2b.Weights)

batchNormalizationLayer("Name","bn5c_branch2b","Epsilon",0.001,"Offset",trainingSetup.bn5c_branch2b
.Offset,"Scale",trainingSetup.bn5c_branch2b.Scale,"TrainedMean",trainingSetup.bn5c_branch2b.Trained
Mean,"TrainedVariance",trainingSetup.bn5c_branch2b.TrainedVariance)
    reluLayer("Name","activation_48_relu")

```

```

convolution2dLayer([1
1],2048,"Name","res5c_branch2c","BiasLearnRateFactor",0,"Bias",trainingSetup.res5c_branch2c.Bias,"Weights",trainingSetup.res5c_branch2c.Weights)

batchNormalizationLayer("Name","bn5c_branch2c","Epsilon",0.001,"Offset",trainingSetup.bn5c_branch2c.
.Offset,"Scale",trainingSetup.bn5c_branch2c.Scale,"TrainedMean",trainingSetup.bn5c_branch2c.Trained
Mean,"TrainedVariance",trainingSetup.bn5c_branch2c.TrainedVariance)];
lgraph = addLayers(lgraph,tempLayers);

tempLayers = [
    additionLayer(2,"Name","add_16")
    reluLayer("Name","activation_49_relu")
    globalAveragePooling2dLayer("Name","avg_pool")
    fullyConnectedLayer(8,"Name","fc")
    softmaxLayer("Name","fc1000_softmax")
    classificationLayer("Name","classoutput")];
lgraph = addLayers(lgraph,tempLayers);

% clean up helper variable
clear tempLayers;
%% Connect all the branches of the network to create the network graph%%
lgraph = connectLayers(lgraph,"max_pooling2d_1","res2a_branch1");
lgraph = connectLayers(lgraph,"max_pooling2d_1","res2a_branch2a");
lgraph = connectLayers(lgraph,"bn2a_branch1","add_1/in2");
lgraph = connectLayers(lgraph,"bn2a_branch2c","add_1/in1");
lgraph = connectLayers(lgraph,"activation_4_relu","res2b_branch2a");
lgraph = connectLayers(lgraph,"activation_4_relu","add_2/in2");
lgraph = connectLayers(lgraph,"bn2b_branch2c","add_2/in1");
lgraph = connectLayers(lgraph,"activation_7_relu","res2c_branch2a");
lgraph = connectLayers(lgraph,"activation_7_relu","add_3/in2");
lgraph = connectLayers(lgraph,"bn2c_branch2c","add_3/in1");
lgraph = connectLayers(lgraph,"activation_10_relu","res3a_branch2a");
lgraph = connectLayers(lgraph,"activation_10_relu","res3a_branch1");
lgraph = connectLayers(lgraph,"bn3a_branch1","add_4/in2");
lgraph = connectLayers(lgraph,"bn3a_branch2c","add_4/in1");
lgraph = connectLayers(lgraph,"activation_13_relu","res3b_branch2a");
lgraph = connectLayers(lgraph,"activation_13_relu","add_5/in2");
lgraph = connectLayers(lgraph,"bn3b_branch2c","add_5/in1");
lgraph = connectLayers(lgraph,"activation_16_relu","res3c_branch2a");
lgraph = connectLayers(lgraph,"activation_16_relu","add_6/in2");
lgraph = connectLayers(lgraph,"bn3c_branch2c","add_6/in1");
lgraph = connectLayers(lgraph,"activation_19_relu","res3d_branch2a");
lgraph = connectLayers(lgraph,"activation_19_relu","add_7/in2");
lgraph = connectLayers(lgraph,"bn3d_branch2c","add_7/in1");
lgraph = connectLayers(lgraph,"activation_22_relu","res4a_branch2a");
lgraph = connectLayers(lgraph,"activation_22_relu","res4a_branch1");
lgraph = connectLayers(lgraph,"bn4a_branch1","add_8/in2");
lgraph = connectLayers(lgraph,"bn4a_branch2c","add_8/in1");
lgraph = connectLayers(lgraph,"activation_25_relu","res4b_branch2a");
lgraph = connectLayers(lgraph,"activation_25_relu","add_9/in2");
lgraph = connectLayers(lgraph,"bn4b_branch2c","add_9/in1");
lgraph = connectLayers(lgraph,"activation_28_relu","res4c_branch2a");
lgraph = connectLayers(lgraph,"activation_28_relu","add_10/in2");
lgraph = connectLayers(lgraph,"bn4c_branch2c","add_10/in1");
lgraph = connectLayers(lgraph,"activation_31_relu","res4d_branch2a");
lgraph = connectLayers(lgraph,"activation_31_relu","add_11/in2");
lgraph = connectLayers(lgraph,"bn4d_branch2c","add_11/in1");

```

```

lgraph = connectLayers(lgraph,"activation_34_relu","res4e_branch2a");
lgraph = connectLayers(lgraph,"activation_34_relu","add_12/in2");
lgraph = connectLayers(lgraph,"bn4e_branch2c","add_12/in1");
lgraph = connectLayers(lgraph,"activation_37_relu","res4f_branch2a");
lgraph = connectLayers(lgraph,"activation_37_relu","add_13/in2");
lgraph = connectLayers(lgraph,"bn4f_branch2c","add_13/in1");
lgraph = connectLayers(lgraph,"activation_40_relu","res5a_branch2a");
lgraph = connectLayers(lgraph,"activation_40_relu","res5a_branch1");
lgraph = connectLayers(lgraph,"bn5a_branch2c","add_14/in1");
lgraph = connectLayers(lgraph,"bn5a_branch1","add_14/in2");
lgraph = connectLayers(lgraph,"activation_43_relu","res5b_branch2a");
lgraph = connectLayers(lgraph,"activation_43_relu","add_15/in2");
lgraph = connectLayers(lgraph,"bn5b_branch2c","add_15/in1");
lgraph = connectLayers(lgraph,"activation_46_relu","res5c_branch2a");
lgraph = connectLayers(lgraph,"activation_46_relu","add_16/in2");
lgraph = connectLayers(lgraph,"bn5c_branch2c","add_16/in1");
%% Train the network using the specified options and training data%%
[net, traininfo] = trainNetwork(augimdsTrain,lgraph,opts);

```