



Acta de Correcciones al Proyecto de Grado Ingeniería de Sistemas y Computación

Fecha: 25 de Febrero 2022

Autores: Andrés Felipe Muñoz Díaz

Nombre del Proyecto de Grado: Identificación de radiografías patológicas y no patológicas mediante procesamiento digital de imágenes y técnicas de aprendizaje de máquina.

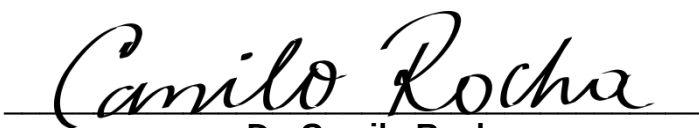
Director: Hernan Darío Vargas Cardona

Como indica el artículo 2.27 de las Directrices de Trabajo de Grado, he verificado que los estudiantes indicados arriba han implementado todas las correcciones que los Jurados del Proyecto de Grado definieron que se efectuaran, como consta en el Acta de Calificación correspondiente.

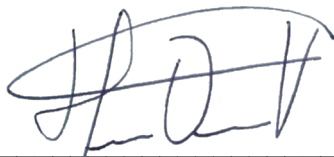
Firma de Director(a) del Proyecto de Grado

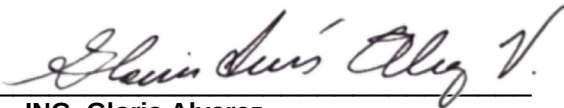
Nota de Aceptación

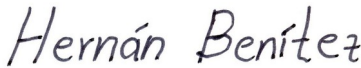
Aprobado por el Comité de Trabajo de Grado
en cumplimiento de los requisitos exigidos por la
Pontificia Universidad Javeriana para optar el
título de Ingeniero de Sistemas y computación.


Dr. Camilo Rocha
Decano de la Facultad de Ingeniería


ING. Gerardo Sarria
Director Carrera Ingeniería Electronica.


ING. Hernán Darío Vargas
Director(a) Trabajo


ING. Gloria Alvarez
Jurado 1


ING. Hernan Benitez
Jurado 2

Pontificia Universidad Javeriana Cali
Facultad de Ingeniería.
Ingeniería de Sistemas y Computación.
Anteproyecto de Grado.

Identificación de radiografías de tórax patológicas y no patológicas mediante procesamiento digital de imágenes y técnicas de aprendizaje de máquina

Andrés Felipe Muñoz Díaz

Director: Dr. Hernán Darío Vargas Cardona

25 de Febrero de 2022



Santiago de Cali, 25 de Febrero de 2022.

Señores

Pontificia Universidad Javeriana Cali.

Dr. Gerardo Mauricio Sarria

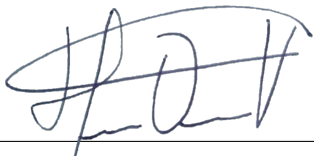
Director Carrera de Ingeniería de Sistemas y Computación.

Cali.

Cordial Saludo.

Por medio de la presente me permito informarle que el estudiante de Ingeniería de Sistemas y Computación Andrés Felipe Muñoz Díaz (cod: 73112) trabaja bajo mi dirección en el proyecto de grado titulado “Identificación de radiografías de tórax patológicas y no patológicas mediante procesamiento digital de imágenes y técnicas de aprendizaje de máquina”.

Atentamente,

A handwritten signature in blue ink, appearing to read 'H. D. V.', is positioned above a horizontal line.

Dr. Hernán Darío Vargas Cardona

Santiago de Cali, 25 de Febrero de 2022.

Señores

Pontificia Universidad Javeriana Cali.

Dr. Gerardo Mauricio Sarria

Director Carrera de Ingeniería de Sistemas y Computación.

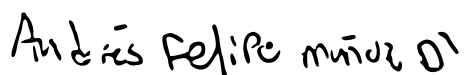
Cali.

Cordial Saludo.

Me permito presentar a su consideración el proyecto de grado titulado “Identificación de radiografías de tórax patológicas y no patológicas mediante procesamiento digital de imágenes y técnicas de aprendizaje de máquina” con el fin de cumplir con los requisitos exigidos por la Universidad para llevar a cabo el proyecto de grado y posteriormente optar al título de Ingeniero de Sistemas y Computación.

Al firmar aquí, doy fe que entiendo y conozco las directrices para la presentación de trabajos de grado de la Facultad de Ingeniería aprobadas el 26 de Noviembre de 2009, donde se establecen los plazos y normas para el desarrollo del anteproyecto y del trabajo de grado.

Atentamente,



Andrés Felipe Muñoz Díaz

Código: 73112

Abstract

In the world, a large number of people suffer from respiratory diseases, which may or may not be diagnosed. Due to the bad or late diagnosis, there are a large number of deaths per year, so it is important to find a viable way of detecting pathologies prematurely in order to be able to give timely treatment and not in advanced stages of the diseases. Given the above, this work proposes a methodology in Python programming language which makes it possible for identifying pathological and non-pathological chest xrays. This is done from 3 stages, the first one of image pre-processing, in which diagnostic images are cleaned and the noise is eliminated. The second a stage of extraction of discriminative characteristics and a third stage where different machine learning algorithms are used to carry out the proposed classification.

Keywords:

- Machine learning
- Image processing
- Featuring extraction
- Pathology
- Xray - Diagnostic image

Resumen

En el mundo, gran cantidad de personas padecen enfermedades respiratorias, las cuales pueden ser o no diagnosticadas. A causa del mal o tardío diagnóstico, se presentan gran cantidad de muertes al año, por lo cual es importante encontrar una manera viable que permita detectar las patologías prematuramente para así mismo poder dar tratamiento oportuno y no en etapas avanzadas de las enfermedades. Dado esto, este trabajo propone una metodología en lenguaje Python la cual permite identificar radiografías de tórax patológicas y no patológicas. Lo anterior se hace a partir de 3 etapas, la primera de pre-procesamiento de imágenes, en la que se realiza una limpieza de las imágenes diagnósticas y eliminación de ruido. La segunda una etapa de extracción de características discriminativas y una tercera etapa donde a partir de técnicas de aprendizaje de máquina para realizar la clasificación propuesta desde diferentes algoritmos.

Palabras Clave:

- Aprendizaje de máquina
- Procesamiento de imágenes
- Extracción de características
- Patología
- Radiografía - Imagen diagnóstica

Índice general

1. Descripción del Problema	13
1.1. Planteamiento del Problema	13
1.1.1. Formulación	14
1.1.2. Sistemización	14
1.2. Objetivos	14
1.2.1. Objetivo General	14
1.2.2. Objetivos Específicos	14
1.3. Justificación	15
1.3.1. Pertinencia	15
1.3.2. Viabilidad	16
1.3.3. Impactos	16
1.4. Marco de Referencia	16
1.4.1. Áreas Temáticas	16
1.4.2. Marco Teórico	17
1.4.3. Trabajos Relacionados	19
1.5. Metodología	20
1.5.1. Tipo de Estudio	20
1.5.2. Actividades	20
2. Desarrollo del Proyecto	23
2.1. Selección y pre-procesamiento del conjunto de datos	23
2.1.1. Identificación del conjunto de datos	23
2.1.2. Pre-procesamiento del conjunto de datos	24
2.2. Extracción de características	26
2.2.1. Extracción Manual de características	26
2.2.2. Extracción de características por transfer learning	28
2.3. Modelo de aprendizaje de máquina para la clasificación de radiografías de tórax entre patológicas y no patológicas	29
2.3.1. Máquina de Soporte Vectorial (SVM)	29
2.3.2. Deep Learning - Transfer learning	30
2.3.3. Red Neuronal Convolutacional (CNN)	30
3. Resultados y discusión	33
3.1. Pre-procesamiento de las imágenes del conjunto de datos	33
3.2. Extracción de características	35
3.2.1. Extracción manual de Características.	35
3.2.2. Extracción de Características por transfer learning	37

3.3. Modelo de aprendizaje de máquina para la clasificación de radiografías de tórax entre patológicas y no patológicas	39
3.3.1. Máquina de soporte vectorial (SVM):	39
3.3.2. Deep Learning - Transfer Learning:	42
3.3.3. Red Neuronal Convolutacional (CNN) :	45
4. Conclusiones	49
4.1. Conclusiones:	49
4.2. Trabajos futuros:	49
Bibliografía	51

Introducción

Las enfermedades respiratorias son consideradas como una de las principales causas de muerte y discapacidad en el mundo, debido a la vulnerabilidad de los pulmones causada por las partículas, químicos, infecciones e impurezas en el aire. Aunque el deterioro respiratorio se genera en todas las regiones del mundo y en todas las clases sociales, la pobreza, el hacinamiento, las exposiciones ambientales y las malas condiciones de vida aumentan la vulnerabilidad de estos trastornos. Además, las enfermedades respiratorias causan el 10 % de todos los años de vida perdidos ajustados por discapacidad (AVAD) [1].

Según la Organización Mundial de la Salud (OMS) cientos de millones de personas padecen de alguna enfermedad respiratoria crónica (ERC) y a menudo muchas no llegan a diagnosticarse [2]. Las enfermedades en el sistema respiratorio causan numerosas muertes al año, entre las más letales se encuentran la enfermedad pulmonar obstructiva crónica (EPOC), la tuberculosis y el cáncer de pulmón. En el 2016 el EPOC causó tres millones de defunciones en el mundo, mientras que el cáncer de pulmón produjo 1,7 millones de muertes y la tuberculosis 1,3 fallecimientos, siendo estas, tres de las 10 principales causas de muerte en el mundo [3].

Con el fin de diagnosticar y tratar oportunamente las enfermedades es necesario utilizar herramientas o tecnologías como la inteligencia artificial (IA), la cual es una rama de la computación capaz de analizar conjuntos de datos médicos complejos [4]. Los métodos de IA se han convertido en una herramienta popular para los investigadores médicos, dado que estas técnicas pueden descubrir e identificar patrones [5], ayudando a predecir si un paciente tiene o no una enfermedad.

Debido a las elevadas cifras de personas que padecen enfermedades respiratorias en el mundo, diagnosticadas o no diagnosticadas y a la cantidad de muertes al año, es importante encontrar una manera efectiva, económica, fácil y rápida que permita detectar las patologías prematuramente para así mismo poder dar tratamiento oportuno y no en etapas avanzadas de las enfermedades. Dado esto, en el desarrollo de este proyecto se usaron técnicas de inteligencia artificial para dar una solución que cumpla con lo descrito anteriormente y que permita identificar pacientes enfermos y no enfermos a partir del análisis de radiografías de tórax.

En este documento se encontrará en el capítulo 1, el planteamiento detallado del problema, los objetivos y justificación del proyecto, el marco de referencia utilizado y la metodología llevada a cabo. En el capítulo 2 se detalla el desarrollo de cada uno de los objetivos del proyecto, incluyendo cómo se hicieron, que algoritmos se usaron y como se llevaron a cabo. El capítulo 3 contiene los resultados obtenidos en cada uno de los objetivos y presentación de los modelos obtenidos a partir de los diferentes algoritmos utilizados. Finalmente, en el capítulo 4 se presentan las conclusiones del proyecto y recomendaciones para trabajos futuros.

Descripción del Problema

1.1. Planteamiento del Problema

El principal reto que deben afrontar las personas con una enfermedad pulmonar es la dificultad que se presenta al momento de ser detectadas, debido a que varias de estas enfermedades, como es el caso del cáncer de pulmón no presenta síntomas precozmente y los diagnósticos se hacen en etapas avanzadas de la enfermedad [6]. La oportuna detección de las enfermedades de pulmón aumenta la probabilidad de vida del paciente considerablemente, como es en el caso del cáncer de pulmón microcítico en el cual la tasa relativa de supervivencia a 5 años baja del 29 % si se detecta cuando aún no se ha propagado al 3 % cuando se ha extendido a otras partes del cuerpo [7].

Las radiografías frontales y laterales son habitualmente el examen inicial que se realiza para identificar enfermedades pulmonares. Frecuentemente enfermedades ya desarrolladas pasan desapercibidas en etapas tempranas, pues características puntuales en las imágenes como lesiones nodulares menores a dos centímetros de diámetro u opacidades focales de baja frecuencias pueden llegar a no ser identificadas. Otra dificultad que existe a la hora de diagnosticar una enfermedad de pulmón son los posibles errores en la interpretación radiológica. Esto puede suceder tanto por fallos de interpretación por parte del radiólogo como errores de percepción y ambigüedades en el informe o por fallos relacionados con el sistema, es decir, problemas técnicos al adquirir la imagen o simplemente información médica incorrecta [8].

Los errores encontrados en el diagnóstico de imágenes médicas se pueden dividir principalmente en dos grupos, el primero consta de todo lo relevante a la calidad y percepción de la imagen y el segundo consiste en todos los errores que pueden ocurrir por parte de la interpretación humana [9]. Las investigaciones se han enfocado principalmente en el método con el que se toman las imágenes, pasando de la radiografía tradicional a métodos más precisos como la tomografía computarizada, obteniendo así gran capacidad de resolución espacial e inclusive imágenes multiplanares y tridimensionales [6].

Dado que la interpretación médica está ligada a la condición humana, este tipo de error suele presentarse frecuentemente, inclusive llegando a ser la tercera causa de mortalidad en los Estados Unidos [10]. El error humano puede presentarse por mala percepción, falta de conocimiento o mal juicio, siendo causal del 40 % al 54 % de las demandas por negligencia médica en casos que involucran radiología [9]. Aunque eliminar el error humano está lejos de la realidad, existen maneras para reducir la frecuencia y mitigar las consecuencias, como es el uso de tecnología como apoyo en la toma de decisiones.

En los últimos años, se ha aumentado el número de investigaciones y proyectos tecnológicos que buscan ayudar en el análisis y diagnóstico de muestras médicas de manera fácil y acertada. La

inteligencia artificial ha sido herramienta común en este tipo de investigaciones, principalmente algoritmos de Deep Learning como lo son las redes neuronales profundas (convolucionales, adversarias, autoencoders, etc), arrojando innumerables aplicaciones en el campo del análisis de imágenes médicas [11]. La inteligencia artificial no busca reemplazar la labor del médico, por el contrario, apunta a ayudar al radiólogo a priorizar casos de manera temprana y oportuna mediante la identificación de pacientes con alguna enfermedad o sospecha de poder tenerla [12].

El uso de inteligencia artificial presenta retos y limitantes, pues el rendimiento de muchos algoritmos automatizados y semiautomatizados no es tan alto, lo que ha obstaculizado su utilidad debido a que casi siempre se necesitan lectores humanos para verificar su precisión. El desempeño de un algoritmo de inteligencia artificial se debe en gran parte a la calidad y cantidad de datos previos que se tienen para el entrenamiento, por lo cual surgen complicaciones al tratar con enfermedades poco comunes, pues solo un número limitado de personas cuentan con la experiencia necesaria para verificar estas enfermedades, limitando así la cantidad de datos clasificados para el entrenamiento [11].

1.1.1. Formulación

¿Cómo identificar radiografías de tórax patológicas y no patológicas mediante procesamiento digital de imágenes y técnicas de aprendizaje de máquina?

1.1.2. Sistematización

¿Cómo reducir el ruido de las imágenes para así mejorar la calidad de las radiografías de tórax patológicas y no patológicas?

¿Cómo lograr la extracción de características discriminativas de las radiografías?

¿Cómo identificar si las radiografías son patológicas o no patológicas?

1.2. Objetivos

1.2.1. Objetivo General

Aplicación de una metodología para la identificación de radiografías de tórax patológicas y no patológicas mediante procesamiento digital de imágenes y técnicas de aprendizaje de máquina.

1.2.2. Objetivos Específicos

1. Implementar una etapa de preprocesamiento de las radiografías digitales que incluye filtrado, eliminación de artefactos y segmentación de la región de interés.
2. Desarrollar una fase de extracción de características discriminativas de las radiografías a partir de descriptores geométricos, estadísticos, de textura y técnicas basadas en transformadas.

3. Identificar radiografías patológicas y no patológicas empleando técnicas de aprendizaje de máquina

1.3. Justificación

1.3.1. Pertinencia

Como se mencionó anteriormente las enfermedades respiratorias se encuentran dentro de las principales causas de muerte en el mundo debido a las elevadas cifras de personas que las padecen. Ejemplo de esto es la tuberculosis, la cual se considera una epidemia en muchos países en vía de desarrollo, los cuales suelen tener un difícil acceso a la atención médica, por lo que en muchas ocasiones no es detectada. Casos como este evidencian la necesidad de encontrar una manera accesible, rápida y fácil de identificar las enfermedades, principalmente en países en vía de desarrollo donde se dificulta el diagnóstico [13].

El mal diagnóstico es uno de los principales problemas que se presentan al momento de identificar enfermedades pulmonares, ya sea porque se pasan por alto o simplemente se dice tenerlas cuando el paciente se encuentra sano. Frente a esto, se hace necesario no reemplazar la labor del médico, pero sí brindar herramientas de apoyo que permitan tener mayor precisión en la identificación de las enfermedades. Los crecientes desarrollos en inteligencia artificial, abren la posibilidad de su uso en ambientes clínicos reales, pues en estudios se han mejorado la cantidad de diagnósticos correctos de enfermedades y disminuido los falsos positivos en comparación a especialistas del tema [14].

La rapidez y precisión al dar un diagnóstico impacta directamente el desarrollo de una posible enfermedad y la recuperación del paciente, por esto es viable el uso de inteligencia artificial, pues al realizar automáticamente medidas estandarizadas en las imágenes, reduce considerablemente tanto el tiempo requerido por el radiólogo para la interpretación, como la necesidad de realizar tareas complejas que proporcionan poco valor. La inteligencia artificial no solo aporta al médico un rápido diagnóstico, si no también toma en consideración la historia clínica del paciente y un entrenamiento previo para definir qué exámenes son o no pertinentes de realizar, eliminando así procesos inoficiosos o redundantes y enfocando los esfuerzos en los estudios que más aportan en la identificación de una posible enfermedad [15].

La eficacia del tratamiento de una enfermedad pulmonar y la probabilidad del paciente de sobrevivir yace en gran parte en que tan oportuno sea diagnosticada la enfermedad. Por ejemplo, en el cáncer de pulmón la detección y tratamiento temprano de la enfermedad puede reducir entre el 14 % y el 20 % de las muertes a causa de esta enfermedad en personas de alto riesgo. La inteligencia artificial, puede ser ese soporte que requieren los médicos especialistas para identificar el riesgo de padecer una enfermedad para así realizar los respectivos exámenes y tratamientos que se requieran de manera oportuna [16]. El buen uso de la tecnología en aspectos médicos tiene impacto directo en la salud y en la vida de las personas, dando cabida a las investigaciones y proyectos los cuales como en el caso de la inteligencia artificial han tenido resultados prometedores.

1.3.2. Viabilidad

Anteriores trabajos de investigación han demostrado que es posible identificar patologías pulmonares mediante el procesamiento de imágenes diagnósticas en conjunto con técnicas de aprendizaje de máquina[17][18][19]. El uso de estas tecnologías en el campo médico es viable desde el punto de vista económico, debido a que solo se requiere del uso de un computador para examinar las radiografías, siendo así un apoyo de bajo costo para los médicos y además presenta viabilidad desde el punto de vista de los conocimientos adquiridos en la carrera pues se utiliza y profundiza en métodos numéricos, computación gráfica e inteligencia artificial.

1.3.3. Impactos

- **En el campo del conocimiento:** El presente proyecto evidencia un importante campo de estudio, el cual por la viabilidad y pertinencia del mismo deja un punto de partida para futuros trabajos de investigación en el uso de inteligencia artificial y procesamiento de imágenes no solo en estudios radiológicos sino en el campo médico en general.
- **Sociales y Económicos:** En lo económico, este trabajo abre la posibilidad de utilizar métodos tecnológicos de bajo costo como la inteligencia artificial para el diagnóstico de enfermedades, sin necesidad de incurrir en exámenes especializados y de costo elevado a no ser que sea estrictamente necesario. De esta manera se puede brindar la solución a poblaciones vulnerables y de bajos recursos. Además, no solo beneficia a las personas, al disminuir el error humano los centros médicos y profesionales de la salud disminuirán los gastos legales que acarrearán pues errores de diagnósticos, pues entre el 40 % y el 54 % de las demandas médicas son ocasionadas por diagnosticar mal a los pacientes[9].
- **Ambientales:** El Impacto ambiental es mínimo, al ser software no tiene contaminación directa del medio ambiente y su impacto radica en el consumo energético del computador que lo use.

1.4. Marco de Referencia

En este apartado se describen las áreas temáticas, la base teórica y trabajos pertinentes en el estado del arte los cuales sirven de base para el desarrollo de este proyecto.

1.4.1. Áreas Temáticas

- Temas Sociales y Profesionales.
 - Política Informática / Tecnológica
 - Política de Información Médica
- Computación Aplicada.
 - Vida y Ciencias Médicas

- Matemática de la Computación
 - Análisis Matemático
 - Cálculo
- Metodologías Computacionales.
 - Inteligencia Artificial
 - Aprendizaje Automático

1.4.2. Marco Teórico

- **Radiología:** La radiología, también conocida como diagnóstico por imágenes, es una serie de diferentes pruebas a partir de la toma de imágenes en varias partes del cuerpo. Muchas de estas pruebas son únicas ya que permiten a los médicos ver dentro del cuerpo. Se pueden usar varios exámenes de imágenes diferentes para proporcionar esta vista, incluyendo rayos X, MRI, ultrasonido, tomografía computarizada y PET [20].
- **Radiología de tórax:** Las radiografías torácicas producen imágenes del corazón, los pulmones, los vasos sanguíneos, las vías respiratorias, los huesos del tórax y de la columna vertebral. Estas radiografías también pueden revelar la presencia de líquido o aire en los pulmones o alrededor de ellos. Además, es una forma frecuente de diagnosticar enfermedades, pero también se utiliza para determinar si un tratamiento específico está dando resultado[21].
- **Patologías pulmonares:** Las patologías pulmonares se refiere a enfermedades o trastornos que impiden que los pulmones trabajen apropiadamente. Estos trastornos se dividen en tres categorías: los que afectan las vías respiratorias, los tejidos pulmonares y enfermedades de la circulación pulmonar [22].
- **Inteligencia Artificial:** La Inteligencia Artificial (IA) es un término general que implica el uso de una computadora para modelar y/o replicar el comportamiento inteligente. La investigación en IA se centra en el desarrollo y análisis de algoritmos que aprenden y/o realizan un comportamiento inteligente con una mínima intervención humana [23].
- **Machine Learning:** El machine learning o aprendizaje automático es una rama de la inteligencia artificial que tiene como objetivo enseñar a las computadoras cómo aprender y actuar sin ser programadas explícitamente. Más específicamente, el aprendizaje automático es un enfoque para el análisis de datos que implica construir y adaptar modelos, que permiten a los programas “aprender” a través de un entrenamiento [24].
- **Máquina de Soporte Vectorial (SVM):** Las máquinas de soporte vectorial son algoritmos supervisados de aprendizaje automático utilizados para clasificación, regresión y detección de valores atípicos. Las SVM son efectivas en espacios con alta dimensionalidad e inclusive cuando se cuenta con más atributos que datos, presentando un alto dinamismo y adaptabilidad a los problemas por los diferentes kernel con los que se puede utilizar [25].

- **Deep Learning:** El Deep learning es un subconjunto del aprendizaje automático donde las redes neuronales artificiales, algoritmos inspirados en el cerebro humano, aprenden de grandes cantidades de datos. De manera similar a cómo aprendemos de la experiencia, el algoritmo de deep learning realizaría una tarea repetidamente, donde cada vez aprende más del resultado[26].
- **Red Neuronal Convolutiva (CNN):** Una red neuronal convolutiva, es un algoritmo supervisado de deep learning el cual realiza el aprendizaje directamente de los datos sin necesidad de proporcionar características manualmente. Las CNN son utilizadas en reconocimiento y clasificación de imágenes, audio, series de tiempo y señales[27]. La ventaja de las CNN es que son altamente precisas, extrae características de los datos y puede ser reutilizada para otros problemas, pero tiene el limitante que puede requerir alto consumo de recursos de hardware[27].

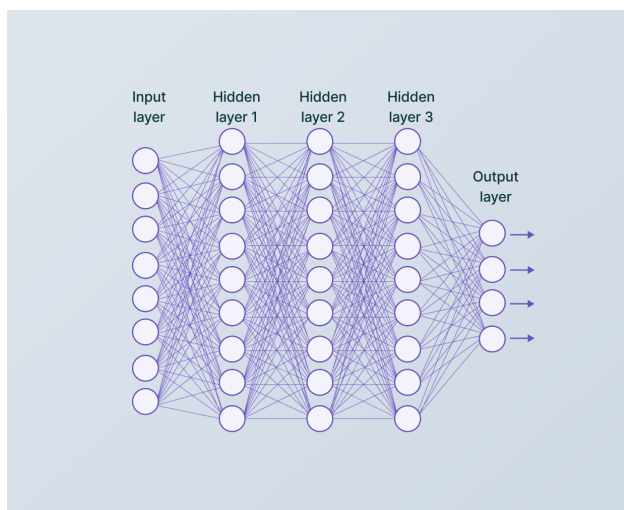


Figura 1.1: Representación de una CNN. Imagen tomada de [28]

- **Extracción de Características:** La extracción de características es un proceso en el cual un conjunto inicial de datos sin procesar se reduce a grupos más manejables para el procesamiento. Un conjunto grande conjuntos de datos es aquel que cuenta con una gran cantidad de variables y requieren elevados recursos de máquina para ser procesados. La extracción de características es el nombre de los métodos que seleccionan y/o combinan variables en características, reduciendo efectivamente la cantidad de datos que deben procesarse, mientras que aún describen de manera precisa y completa el conjunto de datos original [29].
- **Transfer Learning:** Transfer learning es un método de Deep Learning, el cual a utiliza un modelo previamente existente y lo entrena nuevamente para un problema nuevo relacionado con el original. Hay numerosas aplicaciones en las que resaltan transferencia de características o del balanceo de los nodos, en las cuales se mantienen ciertas capas de la red y se entrenan nuevamente las otras. [30].

1.4.3. Trabajos Relacionados

Dentro de los trabajos que se han enfocado en el procesamiento digital de radiografías de tórax se encuentra el artículo “Multi-scale Morphological Image Enhancement of Chest Radiographs by a Hybrid Scheme” desarrollado por Fatemeh Shahsavari Alavijeh y Homayoun Mahdavi-Nasab en el cual mediante el uso de una transformada discreta de wavelet y métodos bayesianos de eliminación de ruido, buscan mejorar la visibilidad de los detalles mejorando la calidad de las imágenes diagnósticas. Este método consiste principalmente en realizar una descomposición en coeficientes utilizando la transformada de wavelet, a los cuales se les aplica métodos bayesianos para eliminar tanto ruido sea posible. Posteriormente, se reconstruye la imagen mediante la transformada inversa de wavelet, se normaliza y se le aplican filtros para suprimir de las radiografías detalles no necesarios y que no aportan valor [31].

Para la identificación de patologías a partir de radiografías de tórax se han propuesto soluciones basadas en deep learning como es el caso del trabajo investigativo “Deep learning in chest radiography: Detection of findings and presence of change” en donde los autores buscan evaluar la exactitud de un algoritmo de deep learning para la detección de anomalías y evaluación de la estabilidad o el cambio en los resultados de imágenes diagnósticas de tórax. La investigación inició con la clasificación por parte de radiólogos expertos de 874 radiografías de 727 pacientes adultos, dichos resultados fueron utilizados como estándar de referencia. Posteriormente, se utilizó el algoritmo para clasificar las radiografías y se asignaron radiólogos que desconocían el diagnóstico de las imágenes para realizar la lectura de estas. Finalmente, se realizó un análisis de los resultados obtenidos por el algoritmo y se compararon con los de los radiólogos expertos permitiendo concluir que no existe diferencia estadística entre los resultados obtenidos por el algoritmo y el estándar de referencia pero que aún así no podía reemplazar al radiólogo en su labor [32].

Patil y Ubuti en su artículo “Chest X-ray features extraction for lung cancer classification” presenta un algoritmo para la detección de nódulos en radiografías de tórax. Se utilizaron un total de 50 imágenes, a las cuales se les hizo un etapa de preprocesamiento para aumentar el detalle y extraer partes que no tuvieran valor. Posteriormente, se realizó una segmentación de las regiones de interés, las cuales eran donde hubiera presencia de nódulos y áreas cancerígenas. Finalmente, se realizó una etapa de extracción de características geométricas y de contraste, dando como resultado datos estadísticos (media, varianza, covarianza, etc.) relevantes en el proceso de clasificación [33].

El procesamiento de imágenes, la extracción de características y el uso de algoritmos de clasificación han sido usados para la identificación de patologías y ha sido motivo de trabajos de investigación como es el caso de “Feature Extraction and Classification on Esophageal X-Ray Images of Xinjiang Kazak Nationality” en donde los autores toman 300 radiografías y realizan una segmentación de la región de interés, donde se realizan filtros de media y posteriormente se usa el método de ecualización de histograma. Luego del procesamiento de las imágenes, se tomaron 37 características entre textura, frecuencia y complejidad. De las características extraídas, se tomaron las que fueran discriminadoras para ser usadas en el proceso de clasificación. Finalmente, se utilizaron máquinas de vectores de soporte y el algoritmo de los k vecinos más cercanos para realizar la clasificación de las imágenes diagnósticas en cancerígenas y no cancerígenas [34].

1.5. Metodología

1.5.1. Tipo de Estudio

El presente trabajo expondrá un estudio de tipo exploratorio, en donde se busca conocer y presentar el potencial del uso de inteligencia artificial en el campo médico especialmente en la detección de patologías a partir del análisis de imágenes diagnósticas. Este trabajo no experimentará con personas, sino que utilizará datos debidamente etiquetados por un equipo de expertos.

1.5.2. Actividades

Para cumplir con los objetivos específicos del proyecto fue necesario llevar a cabo las siguientes actividades:

- **Objetivo 1 :** Implementar una etapa de preprocesamiento de las radiografías digitales que incluye filtrado, eliminación de artefactos, y segmentación de la región de interés
 - Identificar un conjunto de datos público que contenga radiografías de tórax de pacientes y sus respectivos diagnósticos.
 - Depurar y organizar las radiografías del conjunto de datos a utilizar, dejando como resultado un grupo de imágenes con mayor calidad.
 - Identificar técnicas o algoritmos de preprocesamiento de imágenes digitales.
 - Utilizar algoritmos previamente identificados para preprocesar las imágenes diagnósticas presentes en el conjunto de datos.
- **Objetivo 2 :** Desarrollar una fase de extracción de características discriminativas de las radiografías a partir de descriptores geométricos, estadísticos o de textura y técnicas basadas en transformadas.
 - Estudiar el concepto de extracción de características e identificar técnicas utilizadas para el desarrollo de este.
 - Identificar transformadas matemáticas utilizadas como base para realizar extracción de características.
 - Aplicar las técnicas de extracción de características en las imágenes del conjunto de datos previamente definido.
- **Objetivo 3 :** Identificar radiografías patológicas y no patológicas empleando técnicas de aprendizaje de máquina.
 - Desarrollar un modelo de aprendizaje de máquina que permita identificar radiografías patológicas y no patológicas.

- Entrenar el modelo previamente desarrollado con imágenes provenientes del conjunto de datos seleccionado.
- Evaluar el modelo y validar la precisión de los resultados obtenidos.
- Realizar la matriz de confusión con los resultados obtenidos en las pruebas.

Desarrollo del Proyecto

2.1. Selección y pre-procesamiento del conjunto de datos

A continuación se describe el proceso llevado a cabo para la selección del conjunto de datos y el pre-procesamiento de las imágenes del mismo.

2.1.1. Identificación del conjunto de datos

Con el fin de utilizar técnicas de aprendizaje de máquina y desarrollar una metodología que permita la identificación de radiografías de tórax patológicas y no patológicas es necesario la identificación de al menos un conjunto de datos adecuado. Este conjunto debe contener numerosas radiografías de tórax, cuyas imágenes deben ser lo más claras y limpias posibles, a su vez cada una de ellas debe contener la respectiva etiqueta de si pertenece a un paciente saludable o si por el contrario tiene alguna patología

Para la identificación del conjunto de datos adecuado, fue necesario una investigación exhaustiva, esto debido a que las imágenes diagnósticas son normalmente confidenciales y por ende de difícil acceso. La búsqueda se basó en la identificación de los datos utilizados por otras investigaciones, de las cuales resaltan bases de datos públicas de sitios como Kaggle, hospitales con enfoque investigativo cómo es el caso del hospital de la Universidad de Stanford o el NIH Clinical Center de Estados Unidos, el cual es el hospital más grande del país dedicado totalmente a la investigación clínica [35].

De las bases de datos encontradas, se hizo un barrido aleatorio por los datos dentro de estas y se seleccionó la proveniente del NIH pues a consideración del autor es muy completa y las imágenes son mayormente uniformes y de buena calidad. La base de datos consistía de carpetas con 45000 radiografías de tórax y un archivo de excel con nueve columnas de información de los datos y la clasificación de los mismos. Finalmente para adecuar las imágenes con las que se contaban, se depuró dejando solamente las radiografías frontales y se ajustó el Excel para solo tener el nombre del archivo y si era o no de una persona enferma. Al finalizar la selección y depuración de las imágenes del proyecto, se contó con un conjunto de datos balanceado el cual contiene 5190 imágenes de las cuales 2550 pertenecían a personas sanas y 2640 a radiografías patológicas.

A continuación se muestra un ejemplo de una radiografía de una persona con patología y otra de una persona sana. Ambas imágenes fueron tomadas del conjunto de datos seleccionado.

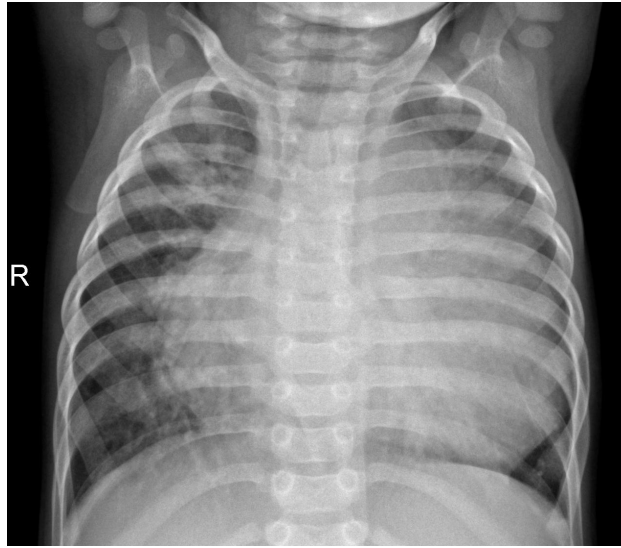


Figura 2.1: Imágen paciente enfermo de neumonía. imagen tomada del NIH

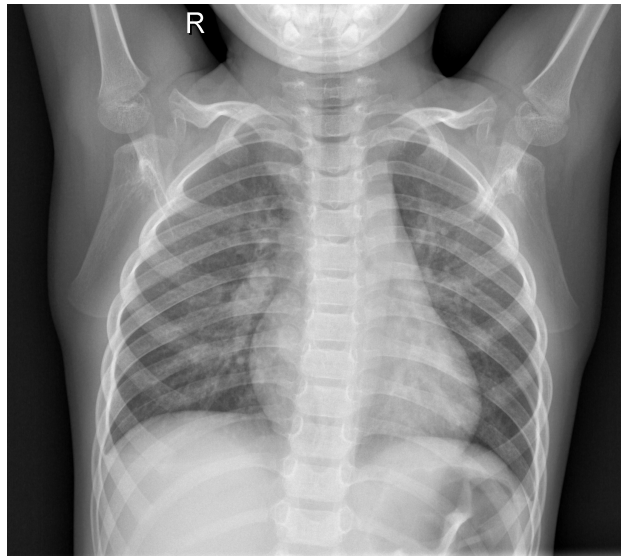


Figura 2.2: Imágen paciente sano. imagen tomada del NIH

2.1.2. Pre-procesamiento del conjunto de datos

Existen múltiples factores que pueden afectar la calidad de las radiografías de tórax, como el aire o los fluidos presentes en los pulmones, tejidos blandos que se encuentran sobre la región de interés, problemas con los equipos o un mal diseño del lugar donde se toman las imágenes [36]. Al

presentarse alguno de los problemas previamente descritos, se evidencia en la imagen problemas de contraste, regiones borrosas o con ruido que dificultan la lectura y posterior diagnóstico de los pacientes.

Por la posibilidad de presentar alguno de los problemas descritos y con el propósito de contar con los mejores datos posibles en esta investigación, fue necesario la implementación de una etapa de pre-procesamiento de cada una de las imágenes. Para esto, se buscó en el estado del arte y se identificaron unas técnicas basadas en transformadas para la reducción de ruido y mejora de contraste que se ve reflejado en imágenes de mejor calidad.

Primero y como parte del pre-procesamiento de las imágenes del conjunto de datos, se buscó reducir el ruido a partir de la transformada wavelet utilizando la función Daubechies. En este proyecto, se utilizó la descomposición ortogonal wavelet de nivel dos y posteriormente a este, se realizó un proceso de umbralización (thresholding) de los coeficientes de la imagen descompuesta, mediante el método de umbralización por reducción Bayesiana. Finalmente se realiza la transformada inversa de wavelet y se reconstruye la imagen, esta vez con la respectiva reducción de ruido. El proceso descrito se implementó en Python mediante el uso de la librería skimage la cual cuenta con funciones dedicadas a wavelet y al procesamiento de imágenes a partir de estas.

Una vez se obtuvo la reducción de ruido mediante el uso de la transformada wavelet, se identificó que para mejorar la calidad de imágenes frecuentemente es utilizado el algoritmo de ecualización de histograma (HE), el cual mejora el contraste aplanando el histograma de la imagen. El problema con este algoritmo, es que toma la imagen entera uniformemente, transformando por igual regiones que pueden ser significativamente más oscuras o claras por igual, generando poca mejora o inclusive empeorando en la calidad [37].

Finalmente, se encuentra el algoritmo AHE con contraste limitado (CLAHE), el cual limita la mejora de contraste para evitar los problemas de amplificación de ruido que presenta AHE. CLAHE en esencia funciona igual que AHE, pero este suaviza los histogramas según un límite llamado clip limit, aumentando el contraste y los detalles de la imagen sin generar ruido[37].

Dado lo anterior, en este proyecto se utilizó el algoritmo de CLAHE sobre toda la base de datos de las radiografías mejorando así notablemente la calidad mediante un aumento del detalle y reducción del ruido en la región torácica, permitiendo así un mejor análisis de cada una de las imágenes.

Cómo último paso en la etapa de pre-procesamiento de las imágenes se identifica que todas estas deben tener el mismo tamaño, para conseguir una base de datos homogénea que permita mejores resultados en el entrenamiento del modelo de clasificación. Con el fin de reducir la capacidad de procesamiento necesaria, se reduce el tamaño de todas las radiografías del conjunto de datos a 200 píxeles x 200 píxeles, esto se hizo mediante la librería PIL.Image la cual utiliza interpolación proximal para hacerlo.

2.2. Extracción de características

El conjunto de datos está compuesto por múltiples imágenes diagnósticas (radiografías de Tórax), por ende, requiere elevados recursos de máquina para ser procesado en su totalidad. Dado lo anterior fue necesario la implementación de una etapa de extracción de características, con la cual se buscó reducir la dimensionalidad de cada una de las imágenes y representarlas en un vector lineal de características que describen el conjunto de datos original. En esta etapa, se investigaron e identificaron diferentes maneras para realizar una extracción de características, de las cuales en el proyecto se implementaron dos, la primera seleccionar características de manera manual y la otra mediante el uso de transfer learning para generar un extractor de características a partir de un modelo de clasificación ya existente.

2.2.1. Extracción Manual de características

Para la extracción manual de características, se buscó cuáles eran utilizadas a menudo y cuáles técnicas se aplican para la extracción de las mismas. Lo primero que se hizo, fue la implementación de un patrón binario local(LBP), el cual es un operador de textura que transforma cada píxel en un número binario. El valor resultante se genera comparando el pixel con cada uno de los lo que lo rodean, siendo uno si es mayor y cero si es menor, como se muestra en el siguiente ejemplo:

			218 < 157 ...? 0	218 < 255 ...? 1	
157	178	220	218 < 178 ...? 0	218 < 219 ...? 1	
219	218	255	218 < 220 ...? 1	218 < 215 ...? 0	
215	219	255	218 < 255 ...? 1	218 < 219 ...? 1	LBP (1,1) = 00111101 = 61

Figura 2.3: Ejemplo Algoritmo LBP. imagen tomada del [38]

Una vez se obtiene el LBP, se busca extraer las primeras dos características que son la energía y la entropía de Shannon. Para empezar, se normaliza la matriz LBP mediante la división de todos sus valores por el valor máximo presente. Luego, se utilizó la librería matemática Numpy para generar el histograma de la matriz normalizada, dado que esta librería cuenta con un conjunto de funciones que lo facilitaba. Una vez se tiene el histograma se generan las probabilidades al dividirlo por la sumatoria de sus valores y retirando los valores en cero. La energía, primera característica extraída, se obtiene mediante la sumatoria de los cuadrados de los valores presentes en la matriz de probabilidades y la entropía de Shannon se obtuvo mediante la sumatoria de la multiplicación de cada valor de la matriz de probabilidades por el logaritmo base dos del mismo.

Para las siguientes características, se utilizó la Matriz de co-ocurrencia de nivel de gris (GLCM),

el cual es un método para extraer detalles de textura y discriminación de patrones. Con la ayuda de la librería skimage se utiliza el algoritmo GLCM para calcular cinco características de textura, las cuales fueron contraste, disimilitud, homogeneidad, energía y correlación. Como entrada del algoritmo GLMC, es necesario indicar el número de píxeles vecinos que se van a evaluar y el ángulo en grados de la orientación en la cual se tomarán los píxeles. En este proyecto se utilizó una distancia de dos píxeles como valor fijo y se generaron las 5 características previamente descritas para las orientaciones 0° , 90° , 180° y 270° , generando así 20 características a partir del GLMC.

Para las siguientes dos características se extrajo la energía y la entropía pero esta vez a partir del filtro de Gabor que es una técnica utilizada para realizar análisis de textura. Para empezar, se genera el filtro Gabor mediante el uso de la librería skimage, Luego, fue necesario generar el histograma de la matriz Gabor, que al igual que para el LBP se hizo mediante el uso de la librería matemática Numpy. Una vez se tiene el histograma se generan las probabilidades al dividir el histograma por la sumatoria de sus valores y retirando las probabilidades en cero. La energía de Gabor, se extrae mediante la sumatoria de los cuadrados de los valores presentes en la matriz de probabilidades y la Entropía Gabor se obtuvo mediante la sumatoria de la multiplicación de cada valor de la matriz de probabilidades por el logaritmo base dos del mismo.

Una vez extraídas las 24 características a cada una de las imágenes se almacenan en una lista de listas (matriz), en la cual las listas interiores representan las imágenes del conjunto de datos. Luego se realizó un análisis para evaluar cuáles de las características eran relevantes y podrían ayudar en la posterior etapa de entrenamiento y generación del modelo de clasificación de radiografías patológicas y no patológicas. Para lo anterior, se normalizó la matriz, se calculó la varianza de cada característica y se removieron aquellas cuya varianza era menor a 0,001, dejando como resultado 19 características. Esto se hizo debido a que en el proceso de entrenamiento de un modelo de inteligencia artificial los valores con variación muy pequeña implica que son muy similares, lo cual puede impactar en la calidad y capacidad de aprender del modelo. A continuación se presenta un diagrama con la descripción del proceso previamente descrito.

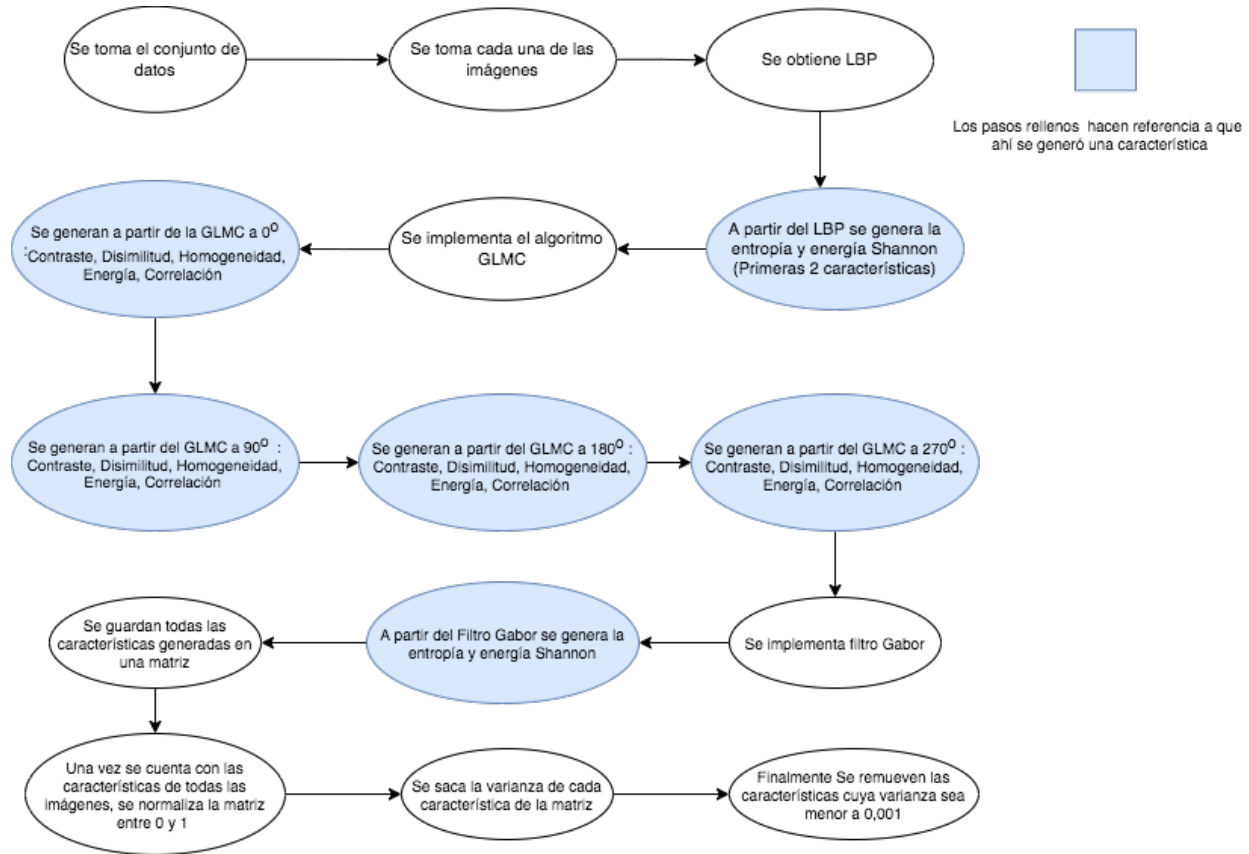


Figura 2.4: diagrama del proceso de extracción de características

2.2.2. Extracción de características por transfer learning

Como segundo método de extracción de características se decidió utilizar transfer learning, es decir, partir de un modelo de inteligencia artificial que ya existía y había sido entrenado previamente. Para que el modelo funcionara, su proceso de entrenamiento previo debía ser en un problema similar al que plantea este proyecto, es decir de clasificación de imágenes. Con el fin de cumplir con lo anterior, se utilizó la librería Keras, la cual cuenta con un conjunto de modelos de deep learning utilizados para clasificación, extracción de características y afinación de parámetros [39]

Luego de leer la documentación de los diferentes modelos presentes en Keras se escogió el modelo VGG19 el cual cuenta con 26 capas y 143667240 parámetros. Como todo modelo, el VGG19 tiene la posibilidad de ser entrenado desde cero o tomar los pesos de un entrenamiento previo. Dado que en este caso se buscó crear un extractor de características se tomó el entrenamiento que trae el modelo y que fue hecho a partir de la base de datos de imágenes Imagenet, la cual es una base de datos libre utilizada en investigaciones y cuenta con cientos de miles de imágenes previamente clasificadas [40].

Finalmente, y con el fin de transformar el modelo de clasificación VGG19 en un extractor de características, se remueven las capas finales de la red, las cuales son las encargadas de clasificación,

generando así un modelo que tiene como salida una capa interior del VGG19, la cual cuenta con nodos y valores ya afinados que serán utilizados como las características de las imágenes a clasificar. Esto se hizo dado que la red fue entrenada en una gran cantidad de imágenes de diferentes tipos, donde aprendió en su procesamiento interno a analizar características o particularidades comunes entre las diferentes imágenes. Estas pueden ser utilizadas para cualquier representación de una imagen como un vector de características. Para concluir esta etapa y al igual que en el proceso manual, se pasó todo el conjunto de datos por el extractor de características y se guardaron los resultados en una matriz que fue usada posteriormente en la etapa de entrenamiento y clasificación.

2.3. Modelo de aprendizaje de máquina para la clasificación de radiografías de tórax entre patológicas y no patológicas

Con el fin de desarrollar un modelo que presentara una buena tasa de acierto, se utilizaron diferentes enfoques y algoritmos de machine learning y deep learning para así poder evaluar cual generaba mejores resultados. En esta sección se describen los algoritmos utilizados y cómo fueron implementados en ese proyecto.

2.3.1. Máquina de Soporte Vectorial (SVM)

El primer algoritmo que se utilizó fue una máquina de soporte vectorial, para este se tomaron los vectores de características resultantes de la etapa anterior y se hizo el mismo proceso de entrenamiento y validación para ambos, esto con el fin de sacar resultados concluyentes del uso del SVM en el vector de características extraído manualmente o el resultante del uso de transfer learning.

Este proceso de entrenamiento consistió en tomar el vector de características y sus respectivas etiquetas y dividirlo en dos partes. La primera parte contaba con el 80 % de los datos y se utilizó para entrenar el modelo, el 20 % restante fue utilizado para evaluar el desempeño del modelo entrenado. Luego, se implementó la SVM, para esto se utilizó la librería sklearn y su conjunto de funciones dedicadas para este algoritmo de aprendizaje de máquina. Sklearn pedía como entrada el tipo de kernel a utilizar por la SVM y unos hiperparámetros que varían según el kernel seleccionado. Para esta investigación se utilizaron dos SVM, una con kernel lineal e hiperparámetro C y la otra con kernel RBF e hiperparámetros C y gamma.

Finalmente, fue necesario afinar los hiperparámetros para ver que valores entregaban mejores resultados. Para esto, se corrió el proceso de entrenamiento múltiples veces, variando los valores de los hiperparámetros en cada una de las ejecuciones. Para el caso de la SVM lineal se varió C entre todos los valores del conjunto 0.001,0.01,0.1,0.5,1.0,1.1,1.5,2.0,4.0,6,8,10,50,100. Para el caso del kernel RBF se alternó C entre los valores del conjunto 0.1,0.5,1.0,2.0,4.0,6,10 y gamma tomó cada uno de los valores del conjunto 0.0001,0.001,0.1,0.5,1.0,2.0,6,10,50,100. Para RBF se realizó una cantidad combinatoria de pruebas fijando cada valor de C y probando con todos los de gamma. Finalmente se realizó en cada ejecución de ambos kernel la respectiva matriz de confusión para poder visualizar cuál de todas las combinaciones de valores presentaba mejores resultados.

2.3.2. Deep Learning - Transfer learning

como segundo método, se buscó implementar un modelo que desde otro enfoque fuera capaz de aprender de la totalidad del conjunto de datos y no solo de las características extraídas. Para lograrlo, se utilizó transfer learning, donde al igual que en la etapa de extracción de características se toma una red neuronal existente, pero con la diferencia de que no se remueven capas y se utiliza la red en su totalidad. La red neuronal sirvió para evaluar su desempeño en el conjunto de datos y ver desde el enfoque deep learning los resultados que se podían llegar a alcanzar.

Al igual que en la etapa anterior se utilizó la librería Keras y con esta se importó la red neuronal VGG19. Una vez se contaba con el modelo, se definió mantener los valores del entrenamiento a partir de la base de datos Imagenet. Dado que se buscaba usar el modelo con otro propósito al que fue desarrollado, se mantuvieron los valores del entrenamiento previo en todas las capas menos en la capa de clasificación final. Esta capa sería entrenada nuevamente con la base de datos del proyecto para así poder clasificar radiografías entre patológicas y no patológicas.

Antes de iniciar el entrenamiento y evaluación del modelo, se tomó la base de datos y sus respectivas etiquetas y se pasaron por un proceso de aumento de datos (data augmentation), este proceso consistió en usar la librería Keras y con sus funciones generar nuevas imágenes a partir de las originales del modelo, para así contar con un conjunto de datos más completo sin necesidad de buscar nuevas imágenes.

Con el conjunto de datos listo, se implementó una función de Callback para el proceso de aprendizaje del modelo. La función utilizada fue ReduceLROnPlateau la cual consiste en monitorear la precisión del modelo y si después de 5 iteraciones no tiene una mejora se reduce la tasa de aprendizaje, para así evitar que el modelo se quede estancado en un óptimo local y pueda buscar una mejor solución. La reducción de la tasa de aprendizaje se dará en un 10 % cada vez y se detendrá en el hipotético caso que el valor tienda a 0 o termine el entrenamiento del modelo.

Dado que el objetivo del proyecto es solo clasificar las imágenes diagnósticas en patológicas y no patológicas, el modelo fue compilado con una función de pérdida binary crossentropy, la cual es utilizada cuando se clasifica solamente entre dos clases que son representadas en cero y uno. La métrica utilizada en la compilación fue la precisión, es decir que el modelo en su proceso de aprendizaje, buscará mejorar este valor en cada epoch que realice.

Para finalizar, se utilizó la función model.fit_generator para entrenar el modelo, esta función recibe como entrada el conjunto de datos de entrenamiento, el conjunto de datos de validación o prueba, la cantidad epochs que a realizar y los pasos por cada epoch, cuantos pasos de validación se harán y finalmente la función de Call back. Esta etapa es la que más recursos de máquina consumió y más tiempo tomó, dejando como resultado las tasas de acierto del entrenamiento y validación del modelo.

2.3.3. Red Neuronal Convolutiva (CNN)

Para finalizar, se implementó una Red Neuronal Convolutiva con el fin de que el modelo pudiera aprender de la totalidad del conjunto de datos y no solo de las características extraídas. Para la construcción de este algoritmo, fue necesario utilizar la librería Keras con la cual se inicializó

el modelo y se definieron cada una de las capas que lo componen. Para este caso, se definió una capa de entrada a la red que recibiera datos de 200 píxeles x 200 píxeles pues estas eran las dimensiones de las imágenes de la base de datos. Luego de construir el modelo, la CNN contó con 14 capas y 27784670 parámetros los cuales todos iban a ser entrenados. La capa final tendría como salida un valor binario el cual representaría si la radiografía de tórax corresponde a una persona sana o si por lo contrario a una que cuente con alguna patología.

Al igual que en el modelo desarrollado por transfer learning, se utilizó la librería keras para llevar a cabo un proceso de aumento de datos , es decir de incrementar las imágenes de la base de datos sin afectar la calidad del entrenamiento ni requerir buscar nuevas imágenes. Luego se utilizó la misma función de Callback que se implementó en el método anterior es decir la función ReduceLROnPlateau

Finalmente, el modelo fue compilado con una función de pérdida binary crossentropy y se utilizó la función model.fit_generator para entrenar el modelo, esta función recibe como entrada el conjunto de datos de entrenamiento, el conjunto de datos de validación o prueba , la cantidad epochs a realizar y los pasos por cada epoch, cuantos pasos de validación se harán y finalmente la función de Callback. Esta etapa es la que más recursos de máquina consumió y más tiempo tomó, dejando como resultado las tasas de acierto del entrenamiento y validación del modelo.

Resultados y discusión

En este capítulo se mostrarán los resultados obtenidos en el proyecto en las etapas de pre-procesamiento de imágenes, extracción de características y el desempeño de los modelos de aprendizaje implementados.

3.1. Pre-procesamiento de las imágenes del conjunto de datos

Una vez se contaba con el conjunto de datos con el que se iba a evaluar el modelo, se creó una estructura de carpetas exactamente igual a la que traían las imágenes inicialmente. El propósito de estas carpetas fue almacenar las imágenes una vez estuviera completada la etapa de pre-procesamiento, para así mantener intactas las imágenes originales por si eventualmente se requería volver a utilizarlas.

Lo primero que se hizo, fue generar dos vectores, el primero contenía las rutas y nombres de cada una de las imágenes del conjunto de datos y el segundo las rutas de donde debían ser dejadas las imágenes ajustadas. Estos vectores se utilizaron como entrada de una función desarrollada, la cual mediante un ciclo recorre el vector de ubicaciones original y las abre una a una, las transforma en escala de grises y les ajusta el tamaño a 200 píxeles x 200 píxeles. Una vez tenemos las imágenes con las dimensiones ajustadas, se pasan por otra función, la cual está construida por la transformada Wavelet y el algoritmo Clahe. Al final de cada iteración del ciclo que recorre las ubicaciones, contamos con una imagen nueva, con mayor detalle y mucho menos ruido, la cual se almacena bajo el mismo nombre en la ruta referente a la nueva estructura de carpetas.

A continuación se presenta la misma imagen en su estado original y luego se muestra tras pasar por el proceso de pre-procesamiento.

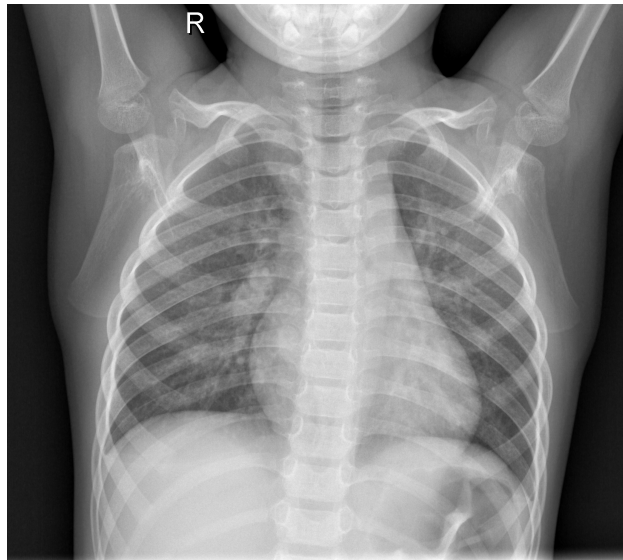


Figura 3.1: Imagen original tomada del conjunto de datos NIH



Figura 3.2: Imagen del conjunto NIH Luego de la etapa de pre-procesamiento

Con estas imágenes se evidencia la mejora en la calidad, en donde después de pasar por esta etapa de pre-procesamiento se aumenta el contraste y se reduce el ruido, dejando como resultado un conjunto de datos con mayor detalle, el cual fue utilizado como insumo del resto del proyecto.

3.2. Extracción de características

A continuación, se mostrarán los resultados obtenidos en la etapa de extracción de características tanto en el proceso manual como en el que involucra Transfer Learning.

3.2.1. Extracción manual de Características.

En esta etapa se tomaron cada una de las imágenes pre-procesadas y se pasaron por un conjunto de 24 funciones que retornaban igual número de características. Estos 24 resultados fueron almacenados en un vector que representa la imagen y que luego se guardaría en una matriz que contendría información de todo el conjunto de datos.

Una vez se tenía la matriz completa, se realizó un proceso de normalización de las características entre cero y uno. Para esto, se tomó cada característica por separado, se identificó el mayor valor entre todas las imágenes y se dividió cada valor por este número. Al final de este proceso, se obtuvo un valor de uno en el mayor número de cada características y el resto fueron valores pertenecientes al intervalo $[0,1]$. Una vez normalizada la matriz se calculó la varianza de las características por medio de una función de Python, los resultados se pueden ver en la siguiente tabla:

Energy_LBP	0.00017
Entropy_LBP	0.002742
GLCM_Contrast 0	0.01151
GLCM_Dissimilarity 0	0.005917
GLCM_Homogeneity 0	0.010541
GLCM_Energy 0	0.010309
GLCM_Correlation 0	0.000979
GLCM_Contrast 90	0.014644
GLCM_Dissimilarity 90	0.008252
GLCM_Homogeneity 90	0.009961
GLCM_Energy 90	0.014977
GLCM_Correlation 90	0.000935
GLCM_Contrast 180	0.01511
GLCM_Dissimilarity 180	0.008422
GLCM_Homogeneity 180	0.009709
GLCM_Energy 180	0.014951
GLCM_Correlation 180	0.000912
GLCM_Contrast 270	0.01151
GLCM_Dissimilarity 270	0.005917
GLCM_Homogeneity 270	0.010541
GLCM_Energy 270	0.010309
GLCM_Correlation 270	0.000979
Energy_Gabor	0.006338
Entropy_Gabor	0.01101

Figura 3.3: Tabla de varianzas de las características del conjunto de datos

Dado que entre menor sea la varianza significa que los valores se encuentran cercanos unos de otros, lo cual en modelos de aprendizaje puede sesgar el clasificador aprendiendo de valores no influyentes. Por lo anterior, se retiraron la correlación GLCM a 0°, la correlación GLCM a 90°, la correlación GLCM a 180°, la correlación GLCM a 270° y la Energía LBP, pues en estas la varianza era menor a 0.001. Al final de este proceso, se contó con 19 características que serían posteriormente

utilizadas en el proceso de entrenamiento del modelo de clasificación.

3.2.2. Extracción de Características por transfer learning

Como segundo método de Extracción de características, se implementó un extractor a partir de un modelo ya existente y entrenado para clasificación de imágenes. En este caso se utilizó el modelo VGG19, para el cual se definió que los parámetros no fueran entrenables, es decir, que mantuvieran los valores adquiridos al aprender de la base de datos Imagenet. El modelo se inicializó para que reciba de entrada imágenes con dimensiones 200 píxeles x 200 píxeles que es el tamaño que tienen las radiografías del conjunto de datos.

Una vez se cuenta con el modelo cargado se le retira la última capa de la red, la cual es la encargada de realizar la calificación. A continuación se evidencia el resumen de la red y algunas de las características de las capas que la componen.

Model: "vgg19"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 200, 200, 3)	0
block1_conv1 (Conv2D)	(None, 200, 200, 64)	1792
block1_conv2 (Conv2D)	(None, 200, 200, 64)	36928
block1_pool (MaxPooling2D)	(None, 100, 100, 64)	0
block2_conv1 (Conv2D)	(None, 100, 100, 128)	73856
block2_conv2 (Conv2D)	(None, 100, 100, 128)	147584
block2_pool (MaxPooling2D)	(None, 50, 50, 128)	0
block3_conv1 (Conv2D)	(None, 50, 50, 256)	295168
block3_conv2 (Conv2D)	(None, 50, 50, 256)	590080
block3_conv3 (Conv2D)	(None, 50, 50, 256)	590080
block3_conv4 (Conv2D)	(None, 50, 50, 256)	590080
block3_pool (MaxPooling2D)	(None, 25, 25, 256)	0
block4_conv1 (Conv2D)	(None, 25, 25, 512)	1180160
block4_conv2 (Conv2D)	(None, 25, 25, 512)	2359808
block4_conv3 (Conv2D)	(None, 25, 25, 512)	2359808
block4_conv4 (Conv2D)	(None, 25, 25, 512)	2359808
block4_pool (MaxPooling2D)	(None, 12, 12, 512)	0
block5_conv1 (Conv2D)	(None, 12, 12, 512)	2359808
block5_conv2 (Conv2D)	(None, 12, 12, 512)	2359808
block5_conv3 (Conv2D)	(None, 12, 12, 512)	2359808
block5_conv4 (Conv2D)	(None, 12, 12, 512)	2359808
block5_pool (MaxPooling2D)	(None, 6, 6, 512)	0
Total params: 20,024,384		
Trainable params: 0		
Non-trainable params: 20,024,384		

Figura 3.4: Detalle de la red VGG19

Una vez se retiró la capa final del modelo, se pasaron una a una las imágenes del conjunto de datos y se almacenaron en un vector. Al final de este proceso el vector cuenta con tantas posiciones como imágenes del conjunto de datos y cada posición equivale a una imagen, la cual es representada por las 18432 características obtenidas. Al finalizar esta etapa, el vector de características fue utilizado en el proceso de entrenamiento del modelo de clasificación.

3.3. Modelo de aprendizaje de máquina para la clasificación de radiografías de tórax entre patológicas y no patológicas

A continuación se describe el uso de los diferentes modelos de aprendizaje de máquina implementados y los resultados obtenidos luego de pasar el conjunto de datos por ellos para su entrenamiento y posterior validación.

3.3.1. Máquina de soporte vectorial (SVM):

Como primer método se implementó el algoritmo SVM, para el cual se evaluó el desempeño utilizando tanto un Kernel Lineal como uno RBF. Ambos tipos de kernel cuentan con unos hiperparámetros los cuales dependiendo del valor que se les asigne pueden verse reflejados en una SVM con mejores o peores resultados. A continuación se describe el proceso de selección de valores para cada kernel y los resultados obtenidos.

3.3.1.1. SVM kernel lineal

Para el kernel lineal, se utilizó tanto el conjunto de características extraídas manualmente como las obtenidas mediante transfer learning. Para ambos casos se utilizaron las mismas imágenes para entrenamiento y validación. A continuación se presenta una tabla con la precisión obtenida en el SVM con kernel lineal para cada valor de C con el conjunto de características extraídas manualmente:

C	Precisión
0.001	62.00%
0.01	62.00%
0.1	73.00%
0.5	71.00%
1	71.00%
1.1	71.00%
1.5	72.00%
2	72.00%
4	72.00%
7	72.00%
10	72.00%
50	72.00%
100	72.00%

Figura 3.5: Tabla de valores de C y precisión obtenida

De la tabla anterior se evidencia que el valor de C que presentó la mayor precisión fue 0.1 el cual tuvo una precisión del 73 %. A continuación evidenciamos la respectiva matriz de confusión :

C=0.1	0	1
0	144	90
1	80	310

Figura 3.6: Matriz de confusión para C=0.1 y el conjunto de características extraído manualmente

En el caso de las características obtenidas por transfer learning, para el SVM de kernel lineal todas las imágenes fueron clasificadas como patológicas independientemente del valor de C. a continuación se muestra la matriz de confusión obtenida en cada una de las iteraciones.

3.3. Modelo de aprendizaje de máquina para la clasificación de radiografías de tórax entre patológicas y no patológicas

41

Para todo C	0	1
0	0	234
1	0	390

Figura 3.7: Matriz de confusión para todo valor de C en el conjunto de características extraído por transfer learning

Para el kernel RBF, se utilizó tanto el conjunto de características extraídas manualmente como las obtenidas mediante transfer learning. Para ambos casos se utilizaron las mismas imágenes para entrenamiento y validación obteniendo los siguientes resultados:

C	Gamma	Precisión	Gamma	Precisión	Gamma	Precisión	Gamma	Precisión	Gamma	Precisión
0.1	0.0001	62.00%	0.001	62.00%	0.1	68.00%	0.5	73.00%	1	72.00%
	2	71.00%	6	72.00%	10	72.00%	50	68.00%	100	64.00%
0.5	0.0001	62.00%	0.001	62.00%	0.1	73.00%	0.5	73.00%	1	71.00%
	2	72.00%	6	72.00%	10	73.00%	50	73.00%	100	70.00%
1	0.0001	62.00%	0.001	62.00%	0.1	72.00%	0.5	71.00%	1	71.00%
	2	71.00%	6	73.00%	10	73.00%	50	74.00%	100	72.00%
2	0.0001	62.00%	0.001	62.00%	0.1	72.00%	0.5	71.00%	1	72.00%
	2	71.00%	6	74.00%	10	74.00%	50	73.00%	100	71.00%
4	0.0001	62.00%	0.001	63.00%	0.1	71.00%	0.5	72.00%	1	72.00%
	2	73.00%	6	75.00%	10	75.00%	50	72.00%	100	71.00%
6	0.0001	62.00%	0.001	62.00%	0.1	71.00%	0.5	72.00%	1	72.00%
	2	73.00%	6	75.00%	10	76.00%	50	72.00%	100	70.00%
10	0.0001	62.00%	0.001	68.00%	0.1	71.00%	0.5	72.00%	1	72.00%
	2	74.00%	6	75.00%	10	76.00%	50	72.00%	100	69.00%

Figura 3.8: Tabla de precisiones de SVM con Kernel RBF.

En la tabla anterior se muestra que los valores de los hiper-parámetros que presentaron mejores resultados fueron $C=6$ y $\gamma=10$ y $C=10$ y $\gamma=10$, que para ambos casos presentó una precisión del 76 %. A continuación evidenciamos las respectivas matrices de confusión :

C=6 Gamma=10	0	1
0	138	96
1	55	335

Figura 3.9: Matriz de confusión para $C=6$ y $\gamma=6$ en el conjunto de características extraído manualmente.

C=10 Gamma=6	0	1
0	134	100
1	51	339

Figura 3.10: Matriz de confusión para C=10 y Gamma=6 en el conjunto de características extraído manualmente..

En el caso de las características obtenidas por transfer learning y al igual que para el SVM de kernel lineal, en el SVM con kernel RBF todas las imágenes fueron clasificadas como patológicas independientemente de los valores de C o Gamma. A continuación se muestra la matriz de confusión obtenida en cada una de las iteraciones.

Para todo C y Gamma	0	1
0	0	234
1	0	390

Figura 3.11: Matriz de confusión para todo valor de C en el conjunto de características extraído por transfer learning

3.3.2. Deep Learning - Transfer Learning:

Como segundo método se utilizó nuevamente Transfer learning y del mismo modo que para el extractor de características se hizo a partir del modelo VGG19. Al igual que antes se mantuvo la red con lo aprendido en el entrenamiento a partir de la base de datos Imagenet, pero contó con la diferencia de incluir la capa final de la red para que fuera entrenada y aprendiera de la base de datos del proyecto, pues esta es la encargada de hacer la clasificación. El modelo se inicializó para que reciba de entrada imágenes con dimensiones 200 píxeles x 200 píxeles, pues este es el tamaño que tienen las radiografías del conjunto de datos. A continuación se evidencia el resumen de la red y algunas características de las capas que la componen.

Model: "model"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv4 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv4 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv4 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 2)	50178
Total params: 20,074,562		
Trainable params: 50,178		
Non-trainable params: 20,024,384		

Figura 3.13: Detalle de la red VGG19 con todas sus capas.

Una vez se contaba con el modelo, se compiló con una función de pérdida Binary_Crossentropy, un optimizador Adam y la métrica utilizada fue la precisión. Luego, se utilizó la librería ImageDataGenerator para aumentar los datos, este proceso consiste en copiar los datos y hacer pequeñas

modificaciones sobre las copias con el fin de hacer crecer la base de datos sin afectar la precisión del modelo. El proceso de aumento de datos es frecuentemente utilizado en bases de datos que no cuentan con cantidades grandes de elementos.

Finalmente se implementó la función callback ReduceLROnPlateau la cual hará seguimiento al parámetro validation loss y en caso de no mejorar reducirá la tasa de aprendizaje para evitar que se quede estancado en óptimos locales. Luego de esto, se entrenó el modelo y se validó en 10 epochs obteniendo los siguientes resultados.

```
Epoch 1/10
163/163 [=====] - 826s 5s/step - loss: 0.3850 - accuracy: 0.8420 - val_loss: 0.5249 - val_accuracy: 0.7885
Epoch 2/10
163/163 [=====] - 794s 5s/step - loss: 0.1582 - accuracy: 0.9352 - val_loss: 0.2685 - val_accuracy: 0.8974
Epoch 3/10
163/163 [=====] - 794s 5s/step - loss: 0.1153 - accuracy: 0.9580 - val_loss: 1.0369 - val_accuracy: 0.6987
Epoch 4/10
163/163 [=====] - 794s 5s/step - loss: 0.1233 - accuracy: 0.9485 - val_loss: 0.4746 - val_accuracy: 0.8413
Epoch 5/10
163/163 [=====] - 794s 5s/step - loss: 0.1069 - accuracy: 0.9583 - val_loss: 0.5023 - val_accuracy: 0.8349
Epoch 6/10
163/163 [=====] - 794s 5s/step - loss: 0.0870 - accuracy: 0.9684 - val_loss: 0.4260 - val_accuracy: 0.8494
Epoch 7/10
163/163 [=====] - 795s 5s/step - loss: 0.0676 - accuracy: 0.9739 - val_loss: 0.3792 - val_accuracy: 0.8670
Epoch 8/10
163/163 [=====] - 794s 5s/step - loss: 0.0779 - accuracy: 0.9750 - val_loss: 0.4441 - val_accuracy: 0.8381
Epoch 9/10
163/163 [=====] - 794s 5s/step - loss: 0.0755 - accuracy: 0.9717 - val_loss: 0.4495 - val_accuracy: 0.8462
Epoch 10/10
163/163 [=====] - 795s 5s/step - loss: 0.0828 - accuracy: 0.9718 - val_loss: 0.4833 - val_accuracy: 0.8413
```

Figura 3.14: Resultados del entrenamiento y validación de la red

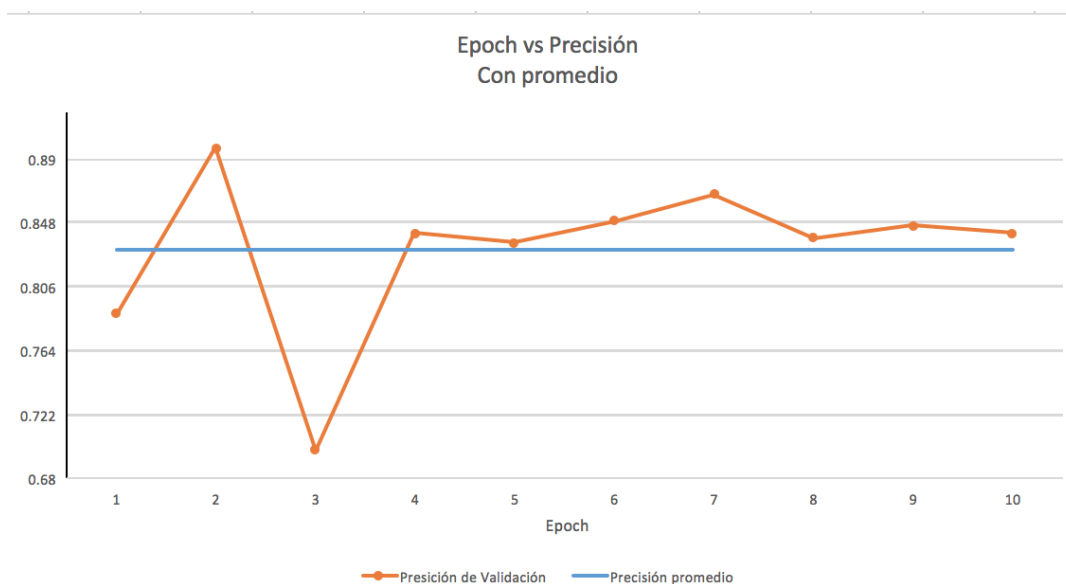


Figura 3.15: Gráfica de los resultados de la validación del modelo.

La figura 3.14 evidencia para cada epoch la pérdida en el entrenamiento (loss), la precisión en el entrenamiento (accuracy) , la pérdida en el proceso de validación o prueba (val_loss) y la precisión en el proceso de validación o prueba (val_accuracy). El principal valor que se tuvo en cuenta al final de la ejecución fue la precisión de validación, la cual para los 10 epochs tuvo un valor promedio de 83.028 % y una desviación estándar de 5.36 %.

3.3.3. Red Neuronal Convolutiva (CNN) :

Para finalizar, se utilizó una Red Neuronal Convolutiva pero esta vez se implementó desde cero y no con transfer learning como en el método anterior. El modelo se inicializó para que reciba de entrada imágenes con dimensiones 200 píxeles x 200 píxeles, pues este es el tamaño que tienen las radiografías del conjunto de datos. A continuación se evidencia el resumen de la red y algunas características de las capas que la componen.

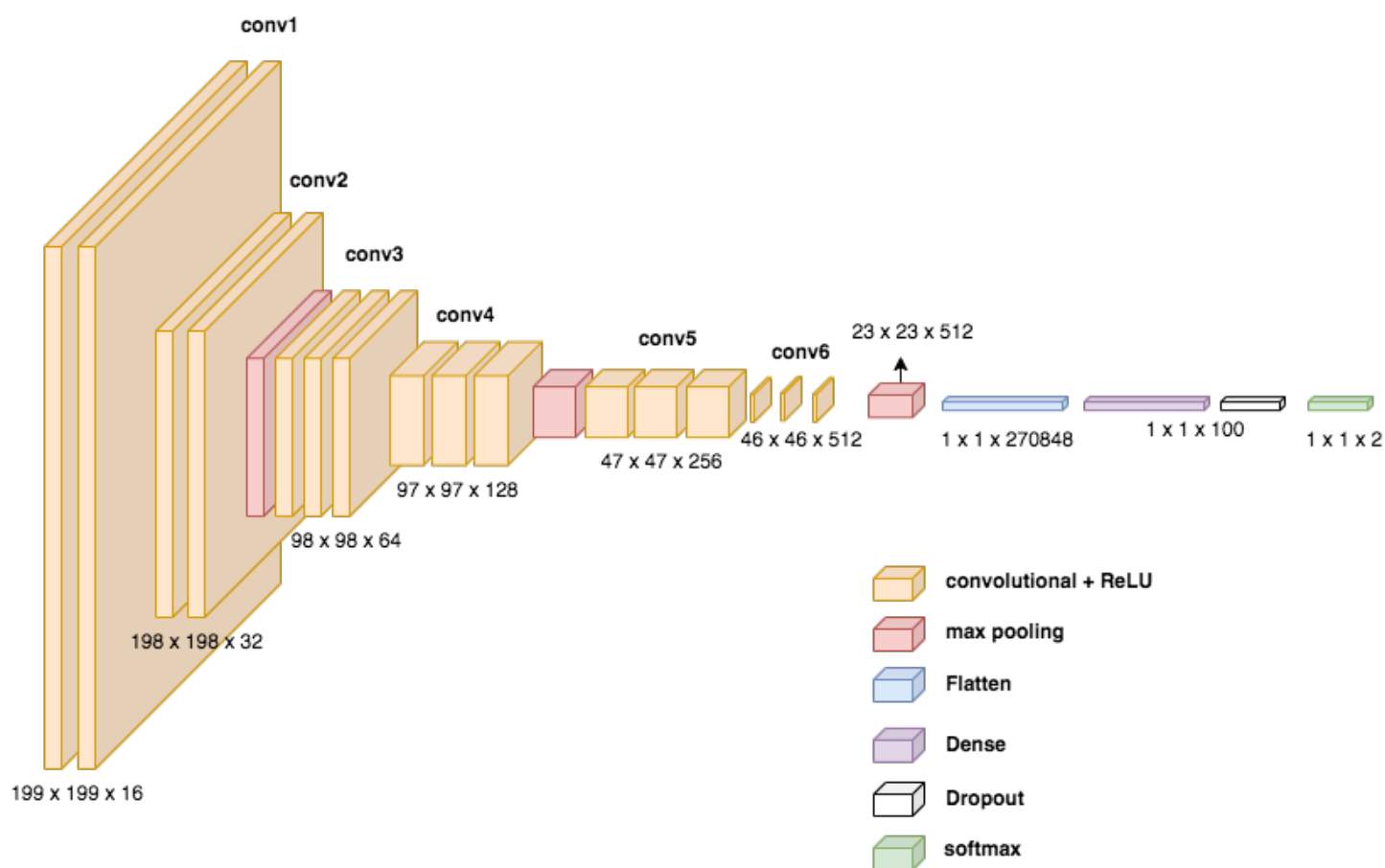


Figura 3.17: Arquitectura de la CNN

Model: "sequential_35"

Layer (type)	Output Shape	Param #
conv2d_94 (Conv2D)	(None, 199, 199, 16)	208
conv2d_95 (Conv2D)	(None, 198, 198, 32)	2080
max_pooling2d_43 (MaxPooling)	(None, 99, 99, 32)	0
conv2d_96 (Conv2D)	(None, 98, 98, 64)	8256
conv2d_97 (Conv2D)	(None, 97, 97, 128)	32896
max_pooling2d_44 (MaxPooling)	(None, 48, 48, 128)	0
conv2d_98 (Conv2D)	(None, 47, 47, 256)	131328
conv2d_99 (Conv2D)	(None, 46, 46, 512)	524800
max_pooling2d_45 (MaxPooling)	(None, 23, 23, 512)	0
flatten_11 (Flatten)	(None, 270848)	0
dense_20 (Dense)	(None, 100)	27084900
dropout_9 (Dropout)	(None, 100)	0
dense_21 (Dense)	(None, 2)	202

Total params: 27,784,670
 Trainable params: 27,784,670
 Non-trainable params: 0

Figura 3.16: Detalle de la CNN

Al igual que como se hizo para el modelo de transfer learning usado en el método anterior, se compiló con una función de pérdida Binary_Crossentropy, un optimizador Adam y la métrica utilizada fue la precisión. Luego, se utilizó la librería ImageDataGenerator para realizar un proceso de aumento de datos. Finalmente se implementó la función callback ReduceLROnPlateau y luego de esto, se entrenó el modelo y se validó en 10 epochs obteniendo los siguientes resultados.

```

Epoch 1/10
163/163 [=====] - 176s 1s/step - loss: 0.4244 - accuracy: 0.8048 - val_loss: 0.5098 - val_accuracy: 0.7676
Epoch 2/10
163/163 [=====] - 141s 863ms/step - loss: 0.2469 - accuracy: 0.8942 - val_loss: 0.4850 - val_accuracy: 0.8109
Epoch 3/10
163/163 [=====] - 140s 857ms/step - loss: 0.2052 - accuracy: 0.9192 - val_loss: 0.4110 - val_accuracy: 0.8494
Epoch 4/10
163/163 [=====] - 142s 871ms/step - loss: 0.2057 - accuracy: 0.9177 - val_loss: 0.4122 - val_accuracy: 0.8157
Epoch 5/10
163/163 [=====] - 139s 852ms/step - loss: 0.1825 - accuracy: 0.9287 - val_loss: 0.4835 - val_accuracy: 0.8237
Epoch 6/10
163/163 [=====] - 139s 853ms/step - loss: 0.1795 - accuracy: 0.9300 - val_loss: 0.7128 - val_accuracy: 0.7612
Epoch 7/10
163/163 [=====] - 140s 858ms/step - loss: 0.1520 - accuracy: 0.9494 - val_loss: 0.5237 - val_accuracy: 0.7853
Epoch 8/10
163/163 [=====] - 143s 873ms/step - loss: 0.1385 - accuracy: 0.9511 - val_loss: 0.6007 - val_accuracy: 0.7933
Epoch 9/10
163/163 [=====] - 143s 874ms/step - loss: 0.1450 - accuracy: 0.9475 - val_loss: 0.5954 - val_accuracy: 0.8093
Epoch 10/10
163/163 [=====] - 143s 873ms/step - loss: 0.1613 - accuracy: 0.9474 - val_loss: 0.6260 - val_accuracy: 0.8029

```

Figura 3.18: Resultados del entrenamiento y validación de la red

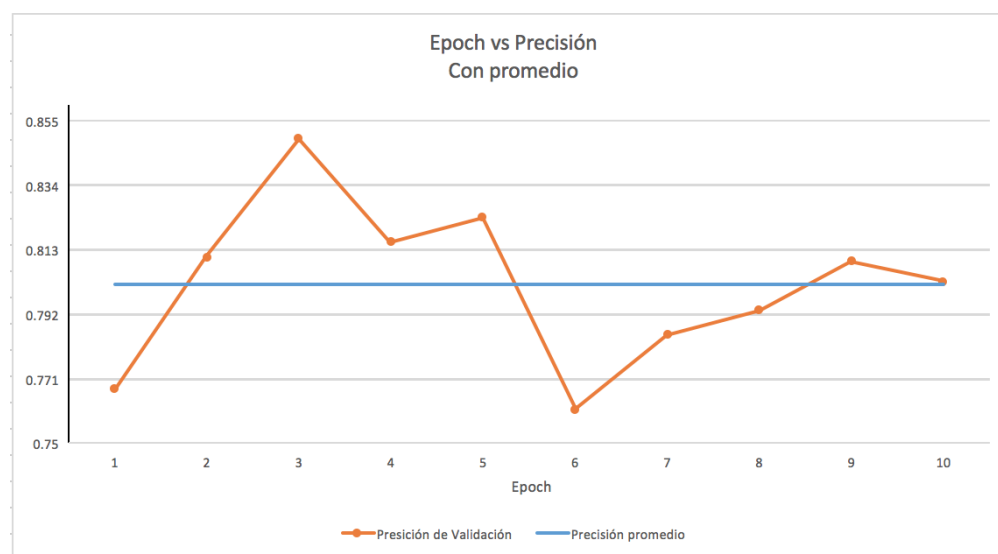


Figura 3.19: Gráfica de los resultados de la validación del modelo.

La figura 3.17 evidencia para cada epoch la pérdida en el entrenamiento (loss), la precisión en el entrenamiento (accuracy) , la pérdida en el proceso de validación o prueba (val_loss) y la precisión

de en el proceso de validación o prueba (`val_accuracy`). El principal valor que se tuvo en cuenta al final de la ejecución fue la precisión de validación, la cual para los 10 epochs tuvo un valor promedio de 80.193 % y una desviación estándar de 2.63 %.

Al terminar esta etapa, el modelo puede ser guardado y utilizado para la clasificación de nuevas imágenes en proyectos de investigación o directamente en el campo médico.

Conclusiones

En esta sección se detallaran las principales conclusiones del proyecto y algunas recomendaciones para trabajos futuros.

4.1. Conclusiones:

Con las altas cifras de pacientes que fallecen en el mundo y en especial en países con escasos recursos por malos o tardíos diagnósticos de las enfermedades se concluye que la implementación de este tipo de tecnologías y utilización en hospitales podrían ayudar a salvar muchas vidas mediante tratamientos en etapas tempranas de las patologías.

Como conclusión principal se identificó que es viable y pertinente la utilización de un modelo de aprendizaje de máquina como mecanismo de apoyo médico efectivo, económico y rápido para detectar patologías en el sistema respiratorio. En este proyecto se implementaron varios modelos de aprendizaje de máquina de los cuales se destacó el uso de Deep Learning con Transfer learning a partir del modelo pre-entrenado VGG19, el cual tuvo una precisión promedio de 83,028 % en la validación realizada, la cual es una cifra prometedora en un proyecto exploratorio como este.

Los resultados de esta investigación concluyen que para la base de datos con la que se cuenta, no es recomendable el uso de un extractor de características con transfer learning a partir del modelo VGG19 para usarlo como insumo de una SVM de kernel lineal o RBF, pues al ser tantas las características obtenidas, los clasificadores no lograron aprender de estas y clasificaron todo como si fueran enfermos.

4.2. Trabajos futuros:

Para futuros trabajos o ampliaciones de esta investigación, se recomienda en la manera de lo posible incrementar la cantidad de imágenes del conjunto de datos, incluyendo de esta manera otros tipos de enfermedades. Así se podrían obtener mejores resultados y no solamente detectar si la persona está enferma o no, sino también que pueda decir cual es la enfermedad que padece. Además, se recomienda contar con equipos que tengan recursos de Hardware (RAM y GPU) elevados para agilizar el proceso de entrenamiento y validación del modelo, pues a mayor cantidad de datos y complejidad del modelo, mayor el procesamiento que se requiere.

Para la utilización de transfer learning para la extracción de características, se recomienda utilizar una base de datos grande para la cual pueda aprender acertadamente y no sesgar el clasificador en una clase en específico.

Se recomienda utilizar inteligencia artificial en investigaciones en el campo médico que no solamente se relacionen con el sistema respiratorio o radiografías de tórax sino también a otro tipo de exámenes u órganos del cuerpo.

Bibliografía

- [1] World Health Organization. *Global Alliance against Chronic Respiratory Diseases*. https://www.who.int/gard/news_events/1-3.GARD-06-07-K1.pdf, 2000.
- [2] *The Global Impact of Respiratory Disease â Second Edition*. Forum of International Respiratory Societies. Sheffield, European Respiratory Society, 2017.
- [3] Organización Mundial de la Salud. *Las 10 principales causas de defunción*. OMS, 2018.
- [4] J. R. T. Monson P. J. Drew A. N. Ramesh, C. Kambhampati. Artificial intelligence in medicine. *Annals of the Royal College of Surgeons of England*, 2004.
- [5] Konstantinos P. Exarchos Michalis V. Karamouzis Dimitrios I. Fotiadis Konstantina Kourou, Themis P. Exarchos. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 2015.
- [6] Dr. Raúl Pefaur D. Current imaging in lung cancer. *Revista Médica Clínica Las Condes*, Volume 24, Issue 1,, 2013.
- [7] *The American Cancer Society medical and editorial content team*. Lung Cancer Survival Rates. *American Cancer Society*, 2019.
- [8] I. Vivas. Common errors in image interpretation in oncology. *SERAM Volume 60, Supplement 1*, 2018.
- [9] Luca Brunese Antonio Pinto. Spectrum of diagnostic errors in radiology. *World Journal of Radiology*, 2010.
- [10] María Sol Rodríguez Calvo Manuel Portela Romero, Rosendo Bugarín González. Human error, patient safety and medical training. *Educación Médica*, Volume 20, Supplement 1, Pages 169-174, 2019.
- [11] John Quackenbush Lawrence H. Schwartz Hugo J. W. L. Aerts Ahmed Hosny, Chintan Parmar. Artificial Intelligence in Radiology. *Nat Rev Cancer* 18, Pages 500â510, 2018.
- [12] Cinthia Cruz MD Quanzheng Li PhD Synho Do PhD Keith Dreyer DO PhD James Brink MD James H. Thrall MD, Xiang Li PhD. Artificial Intelligence and Machine Learning in Radiology: Opportunities, Challenges, Pitfalls, and Criteria for Success. *Journal of the American College of Radiology* , Volume 15, Issue 3, Part B, Pages 504-508, 2018.
- [13] Saurabh Jha Sagar Kulkarni. Artificial Intelligence, Radiology, and Tuberculosis: A Review. *Academic Radiology* Volume 27, Issue 1, Pages 71-75, 2020.
- [14] James Gallagher. Artificial intelligence diagnoses lung cancer. *BBC News*, 2019.

- [15] *French Radiology Community SFR-IA Group, CERF*. Artificial intelligence and medical imaging 2018: French Radiology Community white paper. *Diagnostic and Interventional Imaging Volume 99, Issue 11, Pages 727-742, 2018*.
- [16] *American Lung Association*. Lung Cancer Fact Sheet. *American Lung Association*, <https://www.lung.org/lung-health-diseases/lung-disease-lookup/lung-cancer/resource-library/lung-cancer-fact-sheet>, 2019.
- [17] *Rajaraman S Kim I Antani SK, Jaeger S*. Chest X-ray Analysis using Machine Intelligence Research for HIV/TB Screening. *National Library of Medicine*,.
- [18] *Bhargava Reddy Tarun Raj A Chiramal S Govil N Sinha Manjunath KS S Reddivari P Rao P Warier P Putha, M Tadepalli*. Can Artificial Intelligence Reliably Report Chest X-Rays? *Lab: Qure.ai and Columbia Asia Hospital*, 2018.
- [19] *Research Scientist Dave Steiner, MD and Google Health Shravya Shetty, Technical Lead*. Developing Deep Learning Models for Chest X-rays with Adjudicated Image Labels. *GOOGLE AI BLOG*, 2019.
- [20] *Ontario Association of Radiologists*. What is Radiology? *OARS* <https://oarinfo.ca/patient-info/what-radiology>, 2019.
- [21] *Mayo Clinic Editors*. Radiografías de tórax. *MayoClinic* <https://www.mayoclinic.org/es-es/tests-procedures/chest-x-rays/about/pac-20393494>, 2020.
- [22] *Schafer AI Goldman L. Kraft M*. *Approach to the patient with respiratory disease*. eds. Goldman-Cecil Medicine. 25th ed. Philadelphia, PA: Elsevier Saunders, chap 83., 2016.
- [23] *College of Science Engineering*. *Robotics and Artificial Intelligence*. University of Minnesota.
- [24] *Apife D'arcy John D. Kelleher, Brian Mac Namee*. *Fundamentals of Machine Learning for predictive data analytics*. MIT Press, 2015.
- [25] *Gandhi R*. *Support Vector Machine - Introduction to machine Learning Algorithms*. Towards Data Science, 2016.
- [26] *Bernard Marr*. *What Is Deep Learning AI? A Simple Guide With 8 Practical Examples*. Forbes, 2018.
- [27] *Inc The MathWorks*. *Redes Neuronales Convolucionales 3 cosas que necesitas saber*. The MathWorks, Inc: <https://la.mathworks.com/discovery/convolutional-neural-network-matlab.html>.
- [28] *Pragati B*. *A Comprehensive Guide to Convolutional Neural Networks*. <https://www.v7labs.com/blog/convolutional-neural-networks-guide>, 2021.
- [29] *Massoud Nikravesh Lofty Zadeh Isabelle Guyon, Steve Gunn*. *Feature Extraction Foundations and Applications*. Springer, Studies in Fuzziness and Soft Computing vol 207, 2006.

- [30] Jones T. *Clinical Center - Mission*. NIH:<https://www.nih.gov/about-nih/what-we-do/nih-almanac/clinical-center-cc>, 2018.
- [31] Homayoun MahdaviâNasab Fatemeh Shahsavari Alavijeh. *Multiâscale Morphological Image Enhancement of Chest Radiographs by a Hybrid Scheme*. Department of Electrical Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Isfahan, Iran, 2014.
- [32] C Nitiwarangkul J Patti F Homayounieh A Padole P Rao P Putha V Muse A Sharma S Digu-marthy R Singh, M Kalral. *Deep learning in chest radiography: Detection of findings and presence of change*. . PLoS ONE 13(10): e0204155. <https://doi.org/10.1371/journal.pone.0204155>, 2018.
- [33] S A Patil and V R Udipi. *Chest X-ray features extraction for lung cancer classification*. Journal of scientific and industrial research 69(4), 2010.
- [34] Chuan B. Yan Juan Yao Abdugheni Kutluk Xi M. Kong Sui X. Zhang BFang Yang, Murat Hamit. *Feature Extraction and Classification on Esophageal X-Ray Images of Xinjiang Kazak Nationality*. Hindawy, Journal of Healthcare Engineering, 2017.
- [35] NIH. *Transfer learning for deep learning*. IBM: <https://developer.ibm.com/articles/transfer-learning-for-deep-learning/>, 2019.
- [36] title = Reducing errors in portable chest radiography publisher = Applied Radiology: <https://appliedradiology.com/articles/reducing-errors-in-portable-chest-radiography> year = 2015 Jensen L, Meyer C.
- [37] publisher = Universidad Nacional de Asuncion: https://www.conacyt.gov.py/sites/default/files/Tesis_BECA0817.pdf year = 2017 More L, title = MEJORA DEL CONTRASTE DE IMAGENES A COLOR UTILIZAN
- [38] Medina R. *Obtención de patrones binarios locales en imágenes a color empleando álgebra de cuaterniones*. UCLV: <https://dspace.uclv.edu.cu/>, 2018.
- [39] Keras. *Keras Applications*. Keras:<https://keras.io/api/applications/>.
- [40] Imagenet. *About ImageNet*. Imagenet:<https://www.image-net.org/about.php>, 2021.