



Benemérita Universidad Autónoma de Puebla

**Facultad de Ciencias de la
Computación**

**“Reconocimiento de patrones en
imágenes médicas por medio de
redes neuronales convolucionales”**

T E S I S

Para obtener el título de:

**Maestro en Ciencias de la
Computación**

Presenta:

Luis Ángel Santamaría Colula

Asesor de Tesis:

Dr. César Bautista Ramos

Co-asesor de Tesis:

Dr. Rafael Lemuz López

Puebla, Pue.

Noviembre 2019

Dedicatorias y agradecimientos

A mi padre Jorge Tomás Santamaría Alameda y mi madre Antonia Colula Rodríguez por su apoyo incondicional, sus esfuerzos en todos los momentos, paciencia, sacrificios y confiar en mí, las palabras no alcanzarían para reconocer cada momento que estuvieron a mi lado.

A mis hermanos Sandra y Jorge Arturo por su constancia y entrega, siendo fuentes de inspiración en cada momento difícil.

A cada una de las personas que me apoyaron: amigos, profesores, compañeros, familia y a mi novia Nancy que me acompañaron durante este periodo de mi vida, soy la conclusión de cada uno de los hechos en los cuales participaron, gracias.

Y sobre todo quiero agradecer a cada uno de mis profesores que fueron el cuerpo académico de mi vida: por su paciencia, dedicación, consejos, entrega, así como las ganas de seguir transmitiendo conocimiento y defender la ciencia a toda costa.

"Los autores agradecen al Laboratorio Nacional de Supercómputo del Sureste de México perteneciente a la red de laboratorios nacionales CONACYT, por los recursos computacionales, el apoyo y la asistencia técnica." Así como también al CONACYT por la beca de estudios que hizo todo esto posible.

“La única constante es el cambio”

Heráclito de Éfeso

Contenido

RESUMEN	5
INTRODUCCIÓN	6
1.1 IMÁGENES MÉDICAS	8
1.1.1 <i>Imágenes microscópicas</i>	8
1.1.2 <i>Imágenes macroscópicas</i>	8
1.1.3 <i>Extracción de las imágenes médicas</i>	9
1.1.4 <i>Formato de las imágenes</i>	10
1.2 MODELOS PARA EL PROCESAMIENTO DE IMÁGENES MÉDICAS.....	12
1.3 DIAGNÓSTICOS ASISTIDOS POR COMPUTADORA.....	12
1.4 ENFERMEDADES DETECTABLES EN IMÁGENES MÉDICAS	13
1.4.1 <i>Tumores cerebrales</i>	13
1.4.2 <i>Tuberculosis</i>	13
1.4.3 <i>Neumonía</i>	14
1.5 PROCESAMIENTO DIGITAL DE IMÁGENES.	15
CAPÍTULO 2. CLASIFICACIÓN Y REGISTRO DE IMÁGENES	16
2.1 ALGORITMOS DE CLASIFICACIÓN DE IMÁGENES	16
2.2 MÉTODOS DE ENTRENAMIENTO Y EVALUACIÓN.....	17
2.2.1 <i>Métricas de evaluación</i>	17
2.2.2 <i>Precisión y tasa de error</i>	18
2.3 FUNCIONES DE PÉRDIDA Y DE PUNTUACIÓN.....	18
2.3.1 <i>Función de pérdida</i>	19
2.3.2 <i>Optimización</i>	20
2.4 REDES NEURONALES.....	21
2.5 EL ALGORITMO DE PROPAGACIÓN DE ERRORES HACIA ATRÁS (BACKPROPAGATION).....	24
2.5.1 <i>Entropía cruzada</i>	24
2.6 REDES NEURONALES CONVOLUCIONALES	25
2.6.1 <i>Arquitectura</i>	25
2.6.2 <i>Implementación de modelos de redes neuronales convolucionales</i>	28
2.6.2.1 <i>Herramientas y librerías</i>	29
2.6.2.2 <i>Implementación de un modelo de red neuronal convolucional en Keras</i>	29
2.6.4 <i>Modelos pre entrenados</i>	36
2.6.4.1 <i>AlexNet</i>	36
2.6.4.2 <i>GoogleNet / Inception</i>	37
2.6.4.3 <i>VGG</i>	38
2.6.4.4 <i>MobileNet</i>	38
2.6.4.5 <i>Desempeño</i>	39
2.7 REGISTRO DE IMÁGENES	40
CAPÍTULO 3. CNNs EN PROBLEMAS DE CLASIFICACIÓN Y REGISTRO DE IMÁGENES MÉDICAS	43
3.1 RECONOCIMIENTO DE TEJIDO TUMORAL EN RESONANCIAS MAGNÉTICAS DEL CEREBRO ..	43
3.1.1 <i>Trabajos relacionados</i>	43
3.1.2 <i>Pre procesamiento</i>	44
3.1.3 <i>Procesamiento</i>	45
3.1.4 <i>Pruebas</i>	46
3.1.5 <i>Aplicación móvil</i>	48
3.1.6 <i>Conclusiones</i>	49
3.2 ENSAMBLE DE APRENDIZAJE PROFUNDO PARA CLASIFICAR TUBERCULOSIS.	49
3.2.1 <i>Trabajo relacionado</i>	50
3.2.2 <i>Método</i>	51

3.2.3 <i>Evaluación</i>	51
3.2.3.1 Tipos de tuberculosis evaluados	51
3.2.3.2 Información de la base de datos.	52
3.2.3.3 Evaluación de una red neuronal convolucional para clasificar	52
3.2.3.4 Evaluación de la red con clases balanceadas	53
3.2.3.5 Evaluación de un ensamble de clasificadores, una CNN por clase.....	53
3.2.4 <i>Conclusiones</i>	54
3.3 REGISTRO DE TÓRAX EN IMÁGENES DE RAYOS X UTILIZANDO CNN	55
3.3.1 <i>Introducción</i>	55
3.3.2 <i>Base de imágenes de rayos X de sujetos sanos y con neumonía</i>	56
3.3.3 <i>Trabajos relacionados</i>	56
3.3.3.1 Clasificación de neumonía	56
3.3.3.2 Registro en imágenes utilizando redes neuronales convolucionales.....	57
3.3.3.3 Registro de radiografías de tórax	57
3.3.4 <i>Algoritmo de registro propuesto</i>	58
3.3.5 <i>Evaluación de las CNNs enfocadas a la detección</i>	62
3.3.6 <i>Evaluación del algoritmo de registro</i>	64
3.3.7 <i>Conclusiones</i>	69
CONCLUSIÓN	70
TRABAJOS FUTUROS	71
ANEXOS	75
1.1 CARTA DE ACEPTACIÓN POR EL LABORATORIO NACIONAL DE SUPERCÓMPUTO DEL SURESTE DE MÉXICO.....	75
1.2 EL CÓDIGO CREADO Y UTILIZADO PARA LA TESIS	76
1.3 INFORMACIÓN SOBRE EL USO DE REDES NEURONALES CONVOLUCIONALES.....	76

Resumen

Las imágenes médicas digitalizadas permiten la detección, diagnóstico y tratamiento de enfermedades. Las imágenes médicas macroscópicas son aquellas imágenes obtenidas por medio de radiación ionizada o magnetismo, las cuales permiten la identificación de órganos, y estructuras del cuerpo.

En los últimos años la aplicación de diversas herramientas computacionales como: bases de datos, procesamiento distribuido, procesamiento digital de imágenes y reconocimiento de patrones en imágenes médicas digitales han contribuido en la creación de Diagnósticos Asistidos por Computadora o también conocidos como CAD (Computer Aided Diagnosis), los cuales sirven como una herramienta auxiliar en el cuidado de la salud. Esta tesis plantea utilizar diversas arquitecturas basadas en Redes Neuronales Convolucionales (CNN por sus siglas en inglés) para la identificación automática de enfermedades en imágenes médicas.

En este trabajo se describen tres arquitecturas de CNN para resolver igual número de problemas de clasificación o detección de patrones asociados con enfermedades, se utilizan diferentes tipos de imágenes médicas como: imágenes de resonancias magnéticas del cerebro para identificar tumores cerebrales, tomografías del tórax para la identificación de tipos de tuberculosis y radiografías del tórax para la detección de neumonía. Finalmente se propone un algoritmo de registro automático de las regiones que corresponden al tórax que identifica intrínsecamente la traslación, escala y rotación que alinean las regiones del tórax en imágenes de rayos X.

Introducción

Las imágenes médicas que contienen una representación del cuerpo humano o partes de él, permiten a través de su análisis la detección, diagnóstico y tratamiento de enfermedades. El uso de imágenes médicas puede permitir mejorar los diagnósticos que son realizadas por médicos especialistas altamente capacitados, aun cuando la confirmación es a través del historial clínico, signos vitales y exámenes de laboratorio. Desgraciadamente factores como la posición del paciente, la mala calidad de la imagen o la falta de pericia del médico encargado de capturar o interpretar la información, puede desembocar en un diagnóstico erróneo, provocando hasta la muerte del paciente.

En los últimos años las imágenes digitales han permitido, a través del reconocimiento de patrones, el desarrollo de sistemas para el diagnóstico asistido por computadora, los cuales sirven como una herramienta auxiliar en el área de la medicina, reduciendo de esta manera, costos y haciendo eficiente la detección de enfermedades. Existen sistemas capaces de detectar automáticamente la presencia de enfermedades específicas en las imágenes médicas, estos utilizan diferentes algoritmos sofisticados para extraer patrones significativos asociados a las enfermedades algunos de los algoritmos más comúnmente utilizados son los métodos de segmentación como el de Cuenca [28], el uso de técnicas de procesamiento de imágenes como el Laplaciano del Gaussian (LoG) y el patrón binario local (LBP) [29] por mencionar algunos. Uno de los enfoques más exitosos para el análisis de imágenes médicas en este momento es una clase de modelos de aprendizaje profundo llamados redes neuronales convolucionales (CNN). Esta tesis plantea la utilización de diversas arquitecturas de red neuronal convolucional enfocada a la identificación automática de enfermedades en imágenes médicas, también propone una arquitectura de pre procesamiento la cual realiza un proceso de registro de imágenes utilizando redes neuronales. Las CNN pueden alcanzar hasta un 96 por ciento de verosimilitud (por esta razón son utilizadas actualmente por Google y Facebook), de tal manera que entender una red neuronal convolucional puede contribuir a la automatización, reducción en tiempo y costo de procesamiento, así como también agregar etapas de pre procesamiento hacen que se pongan a la vanguardia los métodos de aprendizaje automático y los sistemas de diagnósticos asistidos por computadora.

El trabajo describe los componentes clásicos de una arquitectura de CNN y un análisis de cómo algunas variaciones en los parámetros inciden en el desempeño de la tarea de identificación de enfermedades. El uso de herramientas computacionales, así como la implementación de métodos matemáticos los cuales permiten el análisis, reconocimiento y clasificación de

patrones, de tal manera que al llevar estos métodos a la identificación de enfermedades por medio de un análisis de imágenes médicas macroscópicas se mejora la capacidad de detección. Esta tesis se divide en tres capítulos, el primero presenta una breve introducción a las imágenes médicas y al procesamiento digital de imágenes. El segundo capítulo revisa el funcionamiento y arquitectura de las redes neuronales. Por último, el tercer capítulo aplica dos modelos de arquitecturas de redes neuronales convolucionales para resolver un problema de clasificación de tumores en imágenes de resonancia magnética del cerebro, y otro modelo se aplica para la clasificación de tuberculosis utilizando tomografías de los pulmones. Además, se propone el uso de una arquitectura de CNN para realizar el registro en imágenes de rayos X de tórax. Finalmente se presentan las conclusiones del trabajo y algunas propuestas de trabajo futuro.

Capítulo 1. Imágenes médicas digitales

1.1 Imágenes médicas

Las imágenes médicas se pueden clasificar en dos tipos: microscópicas y macroscópicas. A continuación, la descripción de estas.

1.1.1 Imágenes microscópicas

Las imágenes microscópicas permiten la observación de células, organismo básico que contiene todo ser vivo. Estas imágenes son de interés para un diagnóstico más preciso, no se requiere el uso de radiación, y por lo general la adquisición de este tipo de imágenes se da por medio del uso de microscopios ópticos o electrónicos.

1.1.2 Imágenes macroscópicas

Las imágenes macroscópicas permiten la observación de órganos del cuerpo como lo son el corazón o el cerebro. Estas imágenes son adquiridas por medio de tomografías, radiación o resonancias magnéticas. La extracción de estas imágenes médicas permite la observación de la estructura del órgano; el medio como se adquiere permite mejorar la calidad de imagen, así como también una representación en 2 o 3 dimensiones. A continuación, los tipos de imágenes médicas y sus principales aplicaciones en la tabla 1.

Tabla 1. Tipos de imágenes médicas y ejemplos de sus principales aplicaciones.

Rayos X	pecho(mama), pulmón, huesos
MR (Resonancias magnéticas)	tejido suave, cerebro
US (Ultrasonido)	fetos, cambios patológicos, órganos internos.

Estas modalidades generan representaciones visuales del cuerpo humano, en forma de cortes o capturas generalizadas de los órganos [1].

Durante el desarrollo de esta tesis solo se trabaja con imágenes macroscópicas.

1.1.3 Extracción de las imágenes médicas

Las imágenes son generadas por aparatos especializados los cuales utilizan radiación o magnetismo. Dependiendo del órgano del cual se desea extraer información es el proceso que se realiza. En las dos secciones siguientes se describe a grandes rasgos el proceso de generación de imágenes de resonancia magnética y rayos X para la obtención de imágenes médicas macroscópicas.

1.1.3.1 Resonancia magnética

Una resonancia magnética (RM) del cerebro es un examen imagenológico que utiliza imanes y ondas de radio potentes para crear imágenes del cerebro y de los tejidos nerviosos circundantes (ver figura 1). Esta técnica no emplea radiación [2].

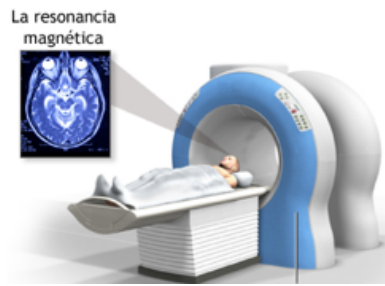


Fig. 1. El escáner de resonancia magnética.

Una RM del cerebro puede usarse para diagnosticar y vigilar enfermedades y trastornos que afectan dicho órgano, como:

- Anomalía congénita
- Sangrado en el cerebro (hemorragia subaracnoidea o intracraneal)
- Historia familiar de aneurismas
- Infección, como absceso cerebral
- Tumores (cancerígenos y no cancerígenos)
- Trastornos hormonales (tales como acromegalia, galactorrea y síndrome de Cushing)
- Esclerosis múltiple
- Accidente cerebrovascular

La RM genera imágenes de cortes bidimensionales del órgano, el corte se compone de elementos de volumen (3 mm aprox.). El conjunto de imágenes que se genera por paciente

varía desde 50 hasta 150 cortes, dependiendo de la distancia de corte, así como también de la edad del paciente [3].

1.1.3.2 Rayos X

Los rayos X son un tipo de radiación llamada ondas electromagnéticas. Las imágenes de rayos X muestran el interior del cuerpo en diferentes tonos de blanco y negro. Esto es debido a que diferentes tejidos absorben diferentes cantidades de radiación. El calcio en los huesos absorbe la mayoría de los rayos X, por lo que los huesos se ven blancos; la grasa y otros tejidos blandos absorben menos, y se ven de color gris; el aire absorbe la menor cantidad, por lo que los pulmones se ven negros [4].

La radiografía de tórax se realiza para evaluar los pulmones, el corazón y la pared torácica.

Los médicos utilizan el examen para ayudar a diagnosticar o controlar el tratamiento de condiciones tales como:

- neumonía
- insuficiencia cardíaca u otros problemas cardíacos
- enfisema
- cáncer de pulmón
- colocación de aparatos médicos
- acumulación de líquido o aire alrededor de los pulmones
- otras enfermedades clínicas [5].

1.1.4 Formato de las imágenes

La imagen digital es una representación bidimensional de una imagen que utiliza hardware y software especializado. A partir de las imágenes digitales se pueden solucionar diversos problemas, sobre estas es posible aplicar compresión, optimización, filtrado, entre otros procesos con la finalidad de mejorar y extraer información. Existen dos modos principales para manipular la información que integra una imagen digital: imágenes de mapa de bits y las imágenes vectoriales. Esta tesis utiliza imágenes en mapas de bits que se encuentran formadas por una rejilla de celdas, cada una de estas celdas se le denomina píxel, el cual tiene asignado un valor de color y luminancia propios.

Se han desarrollado diferentes técnicas de compresión debido a que los archivos de imagen pueden ocupar mucho espacio. Estas técnicas tratan de reducir, mediante algoritmos

matemáticos, el volumen del archivo para disminuir los recursos de cómputo y reducir el tiempo transferencia. Existen dos técnicas de compresión: sin pérdida y con pérdida; lo que se refiere a la información de la imagen que se pierde al ser comprimida. La compresión con pérdida es utilizada para reducir las cadenas del código, desechando información redundante de la imagen. Así, los archivos comprimidos con este método pierden parte de los datos de la imagen, pero se vuelve más eficiente el procesamiento de una gran cantidad de imágenes [6]. A continuación, se describen los formatos de las imágenes con las que trabaja esta tesis, así como la técnica de compresión con la cuentan.

TIF El formato TIF (Tag Image File Format) se utiliza para imágenes de mapa de bits y es admitido prácticamente por todas las aplicaciones de autoedición y tratamiento de imágenes. Este formato fue desarrollado por Aldus Corporation. Su uso es de los más extendidos en la industria gráfica por la calidad de imagen y de impresión que presenta.

DICOM La imagen digital y las comunicaciones en medicina (DICOM, por sus siglas en inglés) es el estándar para la comunicación y el manejo de la información de imágenes médicas y los datos relacionados. DICOM se usa más comúnmente para almacenar y transmitir imágenes médicas, lo que permite la integración de dispositivos de imágenes médicas, como escáneres, servidores, estaciones de trabajo, impresoras, hardware de red y sistemas de comunicación y archivo de imágenes (PACS) de varios fabricantes [6].

Minc El formato de archivo Minc se desarrolló en el Instituto Neurológico de Montreal (MNI) a partir de 1992 para proporcionar un formato de datos flexible para imágenes médicas. El formato es utilizado principalmente por las herramientas de software desarrolladas por el MNI Brain Imaging Center, es decir, un visor y una biblioteca de software de procesamiento de imágenes [7].

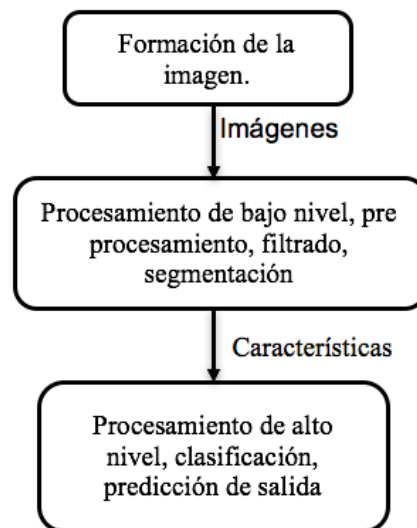
JPG o JPEG Se utiliza usualmente para almacenar fotografías y otras imágenes de tono continuo. Utiliza un sistema de compresión que de forma eficiente reduce el tamaño de los archivos. Compensan esta pérdida con técnicas que suavizan los bordes y áreas que tienen un color similar, haciendo que la disminución de información sea invisible a simple vista [6].

Las imágenes digitales con las que se trabaja esta tesis son generadas por aparatos especializados descritos en la sección anterior, estas imágenes vienen en formatos como

BMPEI, DICOM, o TIF, la tesis convierte los formatos anteriormente mencionados a imágenes en formato JPG, debido al tamaño de la base de imágenes utilizada.

1.2 Modelos para el procesamiento de imágenes médicas.

El análisis e interpretación de las imágenes médicas requiere una serie de pasos que permita la extracción de características, para posteriormente clasificarlas.



La descripción de cada uno de los pasos es la siguiente:

Formación de la imagen. - Se refiere al método que se utiliza para la creación y extracción de la imagen médica.

Procesamiento de bajo nivel. - Esta parte se refiere al procesamiento digital de imágenes, para resaltar las características.

Procesamiento de alto nivel. - Se utilizan algoritmos que trabajen con las características de las imágenes para clasificarlas y posteriormente predecir a qué grupo pertenecen.

1.3 Diagnósticos asistidos por computadora.

Los sistemas de diagnósticos asistidos por computadora se pueden implementar tanto en hardware como en software, las características más importantes que deben contener son:

- Colección de datos provenientes de sensores biológicos.
- Extracción de las características relevantes para la clasificación.
- Pre procesamiento de los datos.
- Selección relevante de las características.
- Agrupación de las características y propiedades en grupos basados en sus similares.

- Interpretación de los resultados obtenidos basados en ideas o conceptos [1].

1.4 Enfermedades detectables en imágenes médicas

Las enfermedades o tumores pueden ser percibidas por la existencia de anomalías o rasgos no comunes en las imágenes médicas, en el caso de esta tesis, se trabajan tres tipos de enfermedades: los tumores cerebrales, la tuberculosis y la neumonía.

1.4.1 Tumores cerebrales

Un tumor cerebral es un crecimiento de células anormales en el tejido del cerebro. Los tumores pueden ser benignos (sin células de cáncer) o malignos (con células cancerígenas que crecen muy rápido). Algunos son primarios, o sea, que comienzan en el cerebro. Otros son metastásicos, es decir, que comenzaron en alguna otra parte del cuerpo y llegan al cerebro [7]. A continuación, se observan imágenes de resonancias magnéticas con y sin presencia de tejido tumoral.

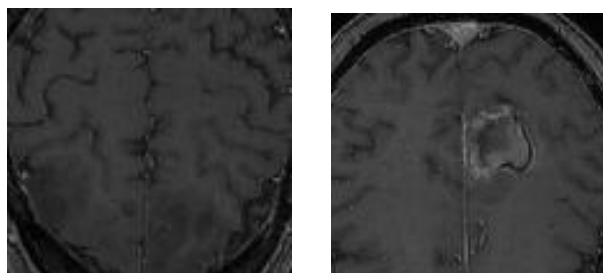


Fig. 2. Ejemplo de resonancias magnéticas utilizada para la tesis.

Las imágenes que se utilizan para esta tesis se encuentran en formato MINC, se realiza una conversión para extraer las resonancias en imágenes con extensión JPG, las imágenes cuentan con una dimensión de 86×136 píxeles y en escala de gris. La presencia del tejido tumoral se observa en la imagen de la derecha la cual es una anomalía comparada con la imagen de la izquierda.

1.4.2 Tuberculosis

La tuberculosis es una enfermedad infecciosa causada por una bacteria llamada *Bacillus microbacterium tuberculosis*. Con una alta tasa de mortalidad en el mundo, esta enfermedad fue una de las diez principales causas de muerte en el mundo en 2015. Diagnosticar esta enfermedad de manera rápida y precisa es un objetivo vital que limitaría su invasión y daño. Uno de los principales problemas de esta enfermedad es que las pruebas tradicionales producen resultados imprecisos o demasiado tardados. Por estas razones, los investigadores han estado interesados

en el diagnóstico de esta enfermedad [8]. Las imágenes digitales que se utilizan para el análisis de la detección de tuberculosis tienen un corte axial. En la figura 3 se observa como es este tipo de captura.

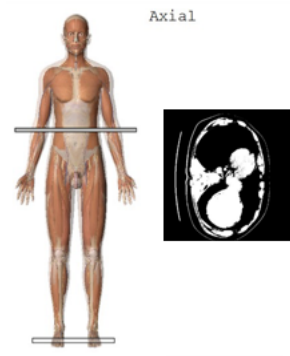


Fig. 3. Muestra la imagen típica de un corte axial en una resonancia magnética.

Las imágenes que se utilizaron para esta tesis se encuentran en formato JPG y cuentan con una dimensión de 512×1512 píxeles y en escala de gris. La presencia de tuberculosis se puede observar como una anomalía en las imágenes capturadas.

1.4.3 Neumonía

Diagnosticar con precisión la neumonía es una tarea difícil, requiere la revisión de una radiografía de tórax (CXR) por especialistas altamente capacitados y la confirmación a través de la historia clínica, los signos vitales y los exámenes de laboratorio. La neumonía generalmente se manifiesta como un área o áreas de mayor opacidad [9] en CXR. Sin embargo, el diagnóstico de neumonía en CXR es complicado debido a otras afecciones en los pulmones, como sobrecarga de líquidos (edema pulmonar), hemorragia, pérdida de volumen (atelectasia o colapso), cáncer de pulmón o cambios posteriores a la radiación o quirúrgicos. Fuera de los pulmones, el líquido en el espacio pleural (derrame pleural) también aparece como una mayor opacidad en CXR. Cuando está disponible, la comparación de CXR tomada en diferentes puntos de tiempo y la correlación con los síntomas clínicos y la historia son útiles para hacer el diagnóstico. Como se menciona anteriormente en una imagen digital la neumonía se puede apreciar como un área con mayor opacidad, en la imagen de la figura 4 se muestra imágenes típicas con presencia de neumonía.

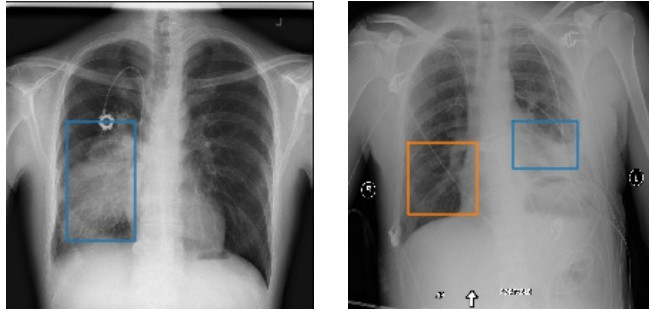


Fig. 4. Muestra de imágenes con neumonía.

Las imágenes originalmente se encontraban en el formato DICOM, se realiza una conversión al formato JPG, obteniendo una resolución de 1024 x1024 píxeles, en RGB.

1.5 Procesamiento digital de imágenes.

El procesamiento de imágenes tiene como objetivo mejorar el aspecto de las imágenes y hacer más evidentes ciertos detalles que se desean hacer notar. La imagen puede haber sido generada de muchas maneras, por ejemplo, fotográficamente, o electrónicamente, por medio de dispositivos especializados. El procesamiento de las imágenes se puede hacer por medio de métodos ópticos, o bien por medio de métodos digitales, en una computadora [10]. El procesamiento digital en el caso de las imágenes médicas permite quitar ruido, cambiar el formato en el que se encuentran las imágenes o redimensionar.

Capítulo 2. Clasificación y registro de imágenes

Este capítulo se centra en explicar desde los fundamentos de los clasificadores, hasta las arquitecturas de redes neuronales convolucionales y los modelos pre-entrenados. Las redes neuronales convolucionales son clasificadores que se centran en agrupar y por ende es necesario explicar parámetros que se utilizan como métricas para evaluar y mejorar su aprendizaje.

2.1 Algoritmos de clasificación de imágenes

La clasificación de imágenes es la tarea de asignar una etiqueta a una imagen de un conjunto fijo de categorías, este es un problema central de la visión por computadora, porque a partir de la solución de este problema se pueden solucionar situaciones como detección y segmentación de objetos. La tarea de conocer un objeto visualmente es trivial para el ser humano, pero desde la perspectiva de un algoritmo computacional existe una lista compleja de desafíos que deben ser solucionados como lo son: variación del punto de vista, deformaciones, oclusiones, iluminación, fondo, etc.

Los algoritmos que pretendan resolver el problema de clasificación deben ser invariantes, es decir que puedan solucionar el problema independientemente de los casos anteriormente descritos. Un enfoque que permite la solución de este tipo de problemas es aquel que se basa en los datos, es decir, a partir de un conjunto este reconoce patrones y se entrena para poder clasificar correctamente las imágenes.

Existen diversos algoritmos los cuales permiten la clasificación y aunque son poco utilizados en la práctica nos permiten tener una visualización de qué tan compleja es la tarea de clasificación y detección de patrones en imágenes, estos algoritmos son: El clasificador del vecino más cercano (KNN por sus siglas en ingles), clasificador lineal (LR,NN), máquinas de soporte vectorial, etc. Para realizar el proceso de segmentación que tiene la finalidad de reducir el área de interés en el análisis de imágenes médicas comúnmente se utilizan los siguientes algoritmos: el algoritmo de Cuenca [28] o procesamiento de imágenes como el Laplaciano de Gaussian (LoG) y el patrón binario local (LBP) [29]. La desventaja de utilizar este tipo de algoritmos se centra en el costo computacional, pues la extracción y clasificación de los patrones varía con el tipo de algoritmo y la dificultad adicional de encontrar los parámetros apropiados.

2.2 Métodos de entrenamiento y evaluación.

Independientemente de la técnica de aprendizaje que se empleé para resolver problema de clasificación, todas ellas tienen algo en común. El resultado del proceso de aprendizaje será un modelo que permite aproximarse a la solución.

Para obtener el modelo es necesario disponer de una gran cantidad de datos (suficientes casos representativos que permitan la generalización a partir de ejemplos concretos), se utilizan estos datos para construir un modelo mediante un proceso de entrenamiento, dando como resultado un modelo aplicable a datos diferentes de los utilizados en su entrenamiento. Si las conclusiones derivadas del modelo aprendido no vienen avaladas por ejemplos suficientes, entonces es inevitable la existencia de errores y ruido en los datos de entrenamiento, el cual no resultaría fiable en la práctica.

Una vez construido un modelo a partir de un conjunto de datos de entrenamiento, es necesario evaluar su calidad. Para ello, en la literatura se estudian dos aspectos complementarios:

Las métricas de evaluación: Cómo medir la calidad de un modelo obtenido mediante el uso de técnicas de aprendizaje automático.

Los métodos de evaluación: Cómo realizar una estimación fiable de las métricas que indica la calidad de un modelo.

2.2.1 Métricas de evaluación

Se toma como ejemplo el caso más habitual y sencillo: un problema de clasificación binaria. Al construir un modelo de clasificación que permita discriminar entre dos clases diferentes. A la primera de ellas, que corresponde a los ejemplos que se desea identificar, se denomina clase positiva (P). Los demás ejemplos serán la clase negativa (N). Al aplicar un modelo de clasificación a un conjunto de datos previamente etiquetado, se observa cómo etiqueta los diferentes ejemplos el clasificador. Sólo existen cuatro posibilidades, que podemos representar en una matriz de contingencia, también llamada matriz de confusión:

Tabla. 2. Matriz de confusión para un problema de clasificación binaria.

	P	N
P	TP	FN
N	FP	TN

Verdaderos positivos [*TP: True Positive*]: Los ejemplos de la clase positiva que el clasificador es capaz de clasificar correctamente.

Falsos positivos [*FP: False Positive*]: Los ejemplos que, aun no siendo de la clase positiva, nuestro clasificador predice que sí lo son.

Falsos negativos [*FN: False Negative*]: Los errores que comete el clasificador en sentido contrario, diciéndonos que no son de la clase positiva cuando en realidad sí lo son.

Verdaderos negativos [*TN: True Negative*]: Los ejemplos de la clase negativa que el clasificador clasifica correctamente.

En un problema de clasificación con múltiples clases, la matriz de confusión tendría una dimensión de $K \times K$, donde K es el número de clases del problema. Las filas y columnas de esa matriz permiten ver dónde acierta y dónde se equivoca el modelo de clasificación, lo que indica posibles problemas y sugiere lo que se debe cambiar. El modelo de clasificación permite disponer de medidas numéricas que resumen el contenido de la matriz de confusión y comparar modelos de clasificación alternativos. Estas métricas se definirán a continuación.

2.2.2 Precisión y tasa de error

La métrica utilizada para resumir el rendimiento de un modelo de aprendizaje supervisado es, la precisión [*accuracy*]. La precisión indica la proporción de ejemplos que un clasificador es capaz de clasificar correctamente, indicada habitualmente en forma de tanto por ciento.

$$precisión = accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

De forma alternativa, se pueden contabilizar los fallos del clasificador y resumir su rendimiento mediante una simple tasa de error [50].:

$$error = \frac{FP + FN}{n} = \frac{\text{errores}}{n}$$

Estos conceptos serán utilizados posteriormente en la evaluación y pruebas del capítulo tres de esta tesis.

2.3 Funciones de pérdida y de puntuación

Como antes se mencionó la tarea de clasificación consiste en asignar una etiqueta a una imagen de un conjunto fijo de categorías, para que un clasificador resuelva correctamente esta tarea se encontrará basado en dos componentes principales: una **función de puntuación** que asigna los

datos en bruto a las puntuaciones de la clase, y una **función de pérdida** que cuantifica el acuerdo entre las puntuaciones pronosticadas y las etiquetas reales, mostrando que tan certera es la clasificación.

Para explicar a detalle la función de puntuación se pone de ejemplo un clasificador lineal:

$$f(x_i, W, b) = W x_i + b$$

Donde:

x_i	Es una imagen la cual será evaluada.
W	Una matriz de tamaño $K \times D$, donde K son las categorías para clasificar y D son las dimensiones de las imágenes.
B	Es un vector de tamaño $K \times 1$ que representa el desplazamiento de los hiperplanos con respecto al origen.

Para entender esta notación colocamos la siguiente imagen:

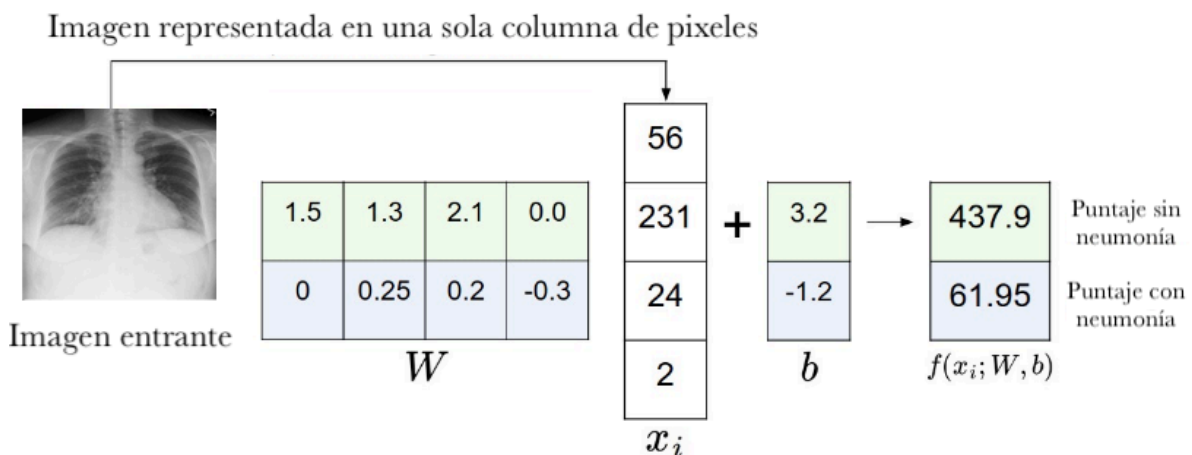


Fig. 5. Visualización del proceso de evaluación en un clasificador lineal.

Como se puede observar en esta imagen, la función de puntuación, asigna un valor el cual nos muestra que la clasificación es correcta por el algoritmo de entrenamiento. En esta imagen los valores de W fueron obtenidos por un entrenamiento previo.

2.3.1 Función de pérdida

Como anteriormente se definió contamos con una función de valores de los píxeles a las puntuaciones de clases que fue parametrizada por un conjunto de pesos W , los valores de esta

matriz pueden ser modificados de tal manera que se clasifiquen correctamente los objetos a las clases a las que pertenecen. Para modificar los valores de la puntuación es necesario medir las diferencias entre el mejor puntaje y el puntaje obtenido, este proceso se lleva a cabo en el entrenamiento. La función de pérdida también es conocida como función de costos u objetivo. Existen diferentes algoritmos para definir la función de pérdida e ir reajustando W , como lo son Softmax o Máquinas de Soporte Vectorial multiclase (SVM). El clasificador Softmax proporciona "probabilidades" para cada clase. A diferencia de la SVM que calcula puntuaciones no calibradas y no fáciles de interpretar para todas las clases, el clasificador Softmax nos permite calcular las "probabilidades" para todas las etiquetas.

2.3.2 Optimización

La optimización es el proceso de encontrar el conjunto de parámetros W que minimizan la función de pérdida.

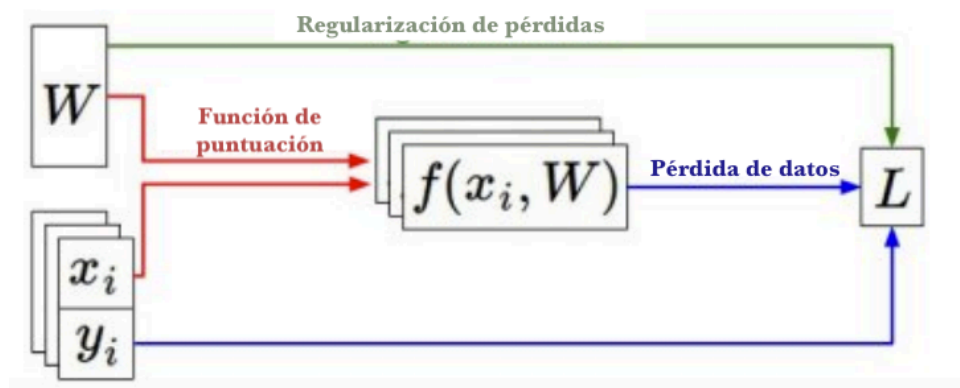


Fig. 6. Visualización del flujo de la función de optimización.

El conjunto de datos de pares de (x, y) se conoce y es fijo. Los pesos comienzan como números aleatorios y pueden cambiar. Durante el paso hacia adelante, la función de puntuación calcula las puntuaciones de clase, almacenadas en el vector f . La función de pérdida contiene dos componentes: la pérdida de datos calcula la compatibilidad entre las puntuaciones f y las etiquetas y . La regularización de la pérdida es sólo una función de pesos. Durante el gradiente descendente se calculan los pesos (y opcionalmente en los datos si se desea) y se usa para realizar una actualización de parámetros [11].

La función de puntuación y la regularización de las pérdidas son fundamentales durante el entrenamiento supervisado de las redes neuronales y las redes neuronales convolucionales.

2.4 Redes neuronales

Para explicar el funcionamiento de las redes neuronales convolucionales primero es necesario analizar el funcionamiento de una neurona. Haciendo una analogía con las neuronas del cerebro, se observa que la estructura básica está compuesta de 4 partes: dendritas, cuerpo de la neurona, axón y terminaciones nerviosas, como se observa en la figura 7.

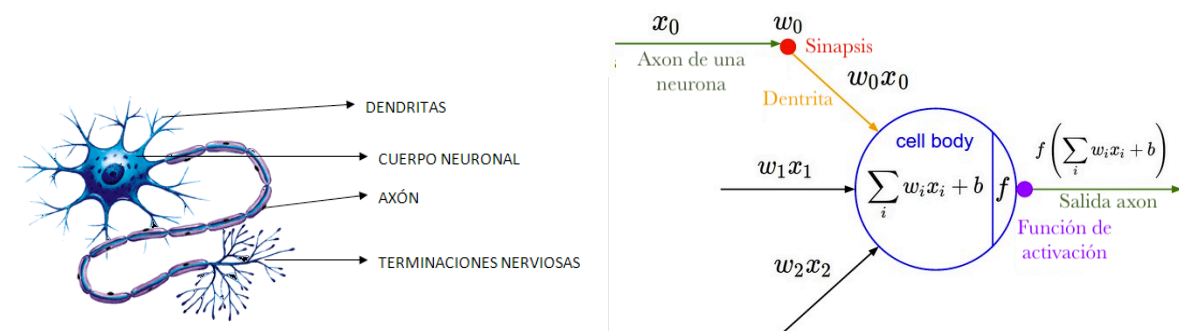


Fig. 7. Representación básica de una neurona.

Cada neurona se encuentra conectada entre sí por medio de las dendritas, por las cuales se intercambia información, entre más interconexiones más inteligentes. Cada neurona básicamente acumula la energía que entra por las dendritas, y esta es ponderada. Esta ponderación es lo que provoca que una red neuronal se encuentre entrenada. Haciendo una analogía simple, un ser humano toma decisiones escuchando consejos de personas las cuales considera importantes, e ignora consejos de personas que no conoce, esto es ponderar las entradas, y el resultado es tomar decisiones “correctas” gracias a los consejos. Cuando se acumula toda la energía en la neurona, se suma todo y pasa a lo que se conoce como una “función de activación”. Esta función de activación es nula si se encuentra por debajo de la función del umbral, pero si es superior se “dispara”.

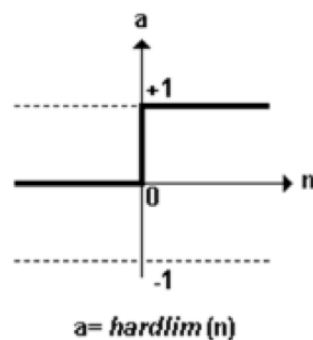


Fig. 8. Función de transferencia escalón.

En la figura 6 podemos observar la función escalón que no es derivable por lo cual normalmente se utiliza la función sigmoide, que se puede ver en la figura 9.

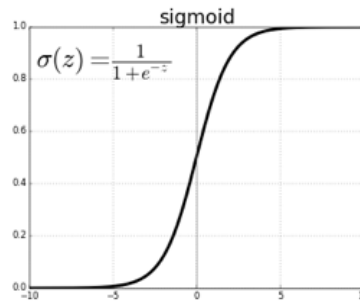


Fig. 9. Función de transferencia sigmoide.

La interpretación geométrica de una neurona es la de una línea que separa el espacio en dos clases. Cuando entrenamos una neurona el objetivo es lograr que esa línea separe correctamente los grupos. Tomando como ejemplo la compuerta lógica OR la cual ejecuta la suma lógica de las variables de entrada, dando como resultado una variable de salida como se observa en la tabla 3.

Tabla 3. Compuerta lógica OR

Entrada		Salida
A	B	
0	0	0
0	1	1
1	0	1
1	1	1

Visualmente el clasificador de la compuerta OR se puede representar de la siguiente manera:

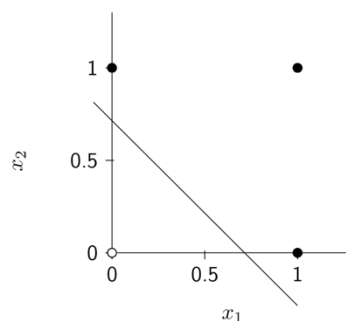


Fig. 10. Ejemplo de un clasificador lineal basado en la compuerta OR

Donde la línea que separa los dos grupos (puntos negros y punto blanco) es la representación gráfica de una neurona entrenada.

Una sola neurona tiene la capacidad de ser entrenada para poder resolver el problema de clasificar bits y representa las funciones de compuertas lógicas como AND y OR (ver figura 10). Sin embargo, si consideramos la compuerta XOR la cual ejecuta una suma lógica de variables de entrada dando como resultado una variable de salida como se muestra en la tabla 4.

Tabla 4. Compuerta lógica XOR

Entrada		Salida
A	B	
0	0	1
0	1	0
1	0	0
1	1	1

El problema se complica, pues como se muestra en la figura 11, la clasificación no se puede resolver correctamente con una sola neurona.

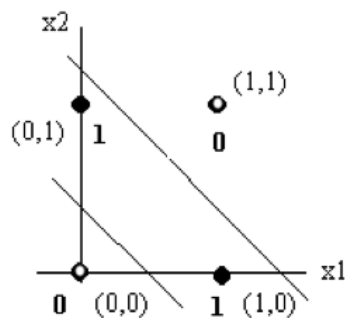


Fig.11. Compuerta lógica XOR

Para resolver el problema es necesario colocar otra neurona, lo que provoca que exista una segunda línea que logra clasificar en dos grupos (puntos negros y puntos blancos).

Con este ejemplo se puede observar que, al agregar más neuronas, se crean redes neuronales propiciando la solución de problemas no solo lineales, sino también de problemas de orden cuadrático permitiendo que las neuronas solucionen el problema con elipses o circunferencias.

El código ejemplo implementado en Python de una neurona es:

```
class Neurona(object):
    def adelante(self, inputs):
        """ asume que las entradas y los pesos son matrices numpy 1-D y el
        sesgo es un número """
        neurona_sum = np.sum(inputs * self.weights) + self.sesgo
        tasa_apren= 1.0 / (1.0 + math.exp(-neurona_sum))
        #Función de activación sigmoide
        return tasa_apren
```

2.5 El algoritmo de propagación de errores hacia atrás (backpropagation)

El entrenamiento de una red neuronal se realiza con algoritmos como *backpropagation*. Este algoritmo, de propagación de errores hacia atrás, utiliza el error observado en la salida de una red neuronal multicapa para ir ajustando paulatinamente los parámetros que definen la respuesta de la red ante una entrada dada, con la intención de reducir el error cometido por la red. Técnicamente, el término *backpropagation* se refiere al método mediante el cual se calcula el gradiente necesario para ajustar los parámetros de la red. Se aplica iterativamente un método de optimización basado en el gradiente descendente para ir ajustando los pesos de la red neuronal y conseguir entrenarla para cumplir una tarea. Para las neuronas de la capa de salida, ese gradiente es fácil de calcular. Sólo necesitamos definir una función de error, coste o pérdida y calcular su derivada con respecto a los pesos de las neuronas de salida para poder ajustar dichos pesos.

Para calcular el gradiente del error, primero se calcula la salida de la red a partir de un ejemplo del conjunto de entrenamiento. Por este motivo, el algoritmo de aprendizaje basado en *backpropagation* y gradiente descendente consta, de dos fases:

1. Una fase de propagación hacia adelante, en la que se suministra a la red un patrón de entrada y calcula la salida de la red para dicho patrón.
2. Una fase de propagación hacia atrás, de donde proviene el término *backpropagation*, en la que se evalúa el error cometido por la red y propaga dicho error hacia atrás, capa por capa, de forma que se pueda calcular eficientemente el gradiente del error para las neuronas ocultas de la red.

En definitiva, *backpropagation* se encarga de evaluar de forma eficiente la contribución de cada neurona oculta al error observado en la capa de salida. Para ello, se aprovecha el que la red neuronal está organizada por capas [50].

2.5.1 Entropía cruzada

La entropía cruzada indica la cantidad de información que falta, en promedio, para predecir correctamente los valores asociados a ejemplos reales dado el modelo predictivo. La entropía cruzada, indica el número de bits necesarios para codificar un dato cuando se utiliza una distribución equivocada en vez de la distribución real. Cuando se construye un modelo se

intenta que se aproxime lo máximo posible a la distribución real.

La entropía cruzada se puede ver como una medida de sorpresa. El modelo se construye a partir de un conjunto de datos de entrenamiento. Sin embargo, el modelo no es perfecto y sus predicciones no serán correctas para todos los datos con los que se encuentre. La entropía cruzada mide, en promedio, cómo de sorprendido quedará el modelo cuando descubra el valor correcto correspondiente a un dato real. Si el valor es el predicho por el modelo, la sorpresa será nula. Cuando el valor no sea el esperado, la sorpresa será alta [50].

2.6 Redes neuronales convolucionales

Las redes neuronales convolucionales son similares a las redes neuronales ordinarias; tienen pesos y sesgos que se pueden aprender. Cada neurona recibe algunas entradas, realiza un producto punto y opcionalmente contiene capas de no linealidad. Toda la red aún expresa una única función de puntuación diferenciable: desde los píxeles de la imagen sin procesar en un extremo hasta las puntuaciones de clase en el otro. Además, tienen una función de pérdida (por ejemplo, Softmax) en la última capa (totalmente conectada).

Las arquitecturas de redes neuronales convolucionales suponen explícitamente que las entradas son imágenes, lo que permite codificar ciertas propiedades en la arquitectura. Esto hace que la función de avance sea más eficiente de implementar y reduzca enormemente la cantidad de parámetros en la red.

2.6.1 Arquitectura

Las redes neuronales reciben una entrada y la transforman a través de una serie de *capas ocultas*. Cada capa oculta está formada por un conjunto de neuronas, donde cada neurona está completamente conectada a todas las neuronas en la capa anterior, y donde las neuronas en una sola capa funcionan de manera completamente independiente y no comparten ninguna conexión. La última capa totalmente conectada se llama "capa de salida" y en la configuración de clasificación representa las puntuaciones de clase.

Las redes neuronales regulares no se adaptan bien a las imágenes, debido al tamaño y la cantidad de datos que representa cada imagen en el manejo de datos debido a la información procesada por cada píxel que sugiere una conectividad total, la cual es un desperdicio y la gran cantidad de parámetros conduciría rápidamente a un sobreajuste.

Volúmenes 3D de neuronas. Las redes neuronales convolucionales aprovechan el hecho de que la entrada consiste en imágenes y restringen la arquitectura de una manera más sensible. En

particular, a diferencia de una red neuronal regular, las capas de una red de convección tienen neuronas dispuestas en 3 dimensiones: **ancho, alto, profundidad** (la *profundidad* se refiere a la tercera dimensión de un volumen de activación, no a la profundidad de una red neuronal completa, que puede referirse al número total de capas en una red). Por ejemplo, las imágenes de entrada cuentan con un volumen de activaciones, y el volumen tiene dimensiones. Las neuronas en una capa solo estarán conectadas a una pequeña región de la capa anterior, en lugar de a todas las neuronas de una manera totalmente conectada. La figura 12 muestra una arquitectura de red simple y una red convolucional:

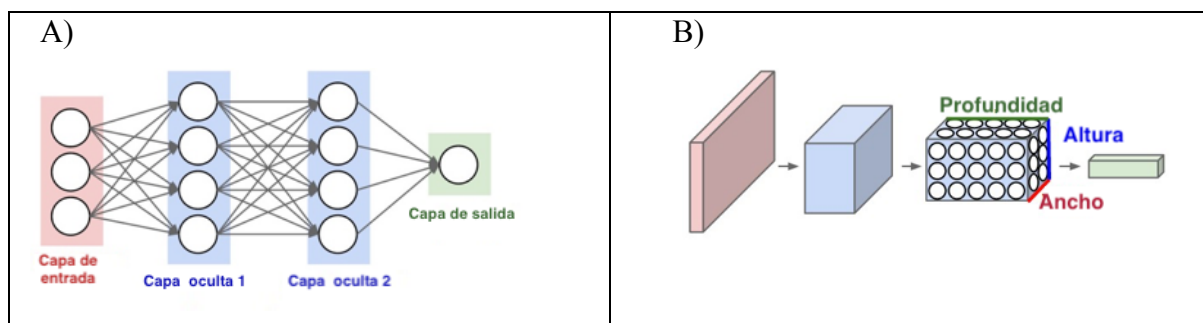


Fig. 12. Comparativa entre una red neuronal simple A) y una red neuronal convolucional B).

A) Una red neuronal regular multicapas. B) Una red neuronal convolucional organiza sus neuronas en tres dimensiones (ancho, alto, profundo), como se visualiza en una de las capas. Cada capa transforma el volumen de entrada 3D en un volumen de salida 3D de activaciones neuronales. En este ejemplo, la capa de entrada (primera capa) contiene la imagen, por lo que su anchura y altura serían las dimensiones de la imagen y la profundidad sería 3 (canal rojo, verde, azul) [12]. Las redes neuronales convolucionales son un modelo donde las neuronas corresponden a campos receptivos de una manera muy similar a las neuronas de la corteza visual primaria de un cerebro biológico. La red se compone de múltiples capas. En el principio se encuentra la fase de extracción de características compuesta de neuronas convolucionales y de reducción. Existen cuatro operaciones principales que conforman las redes convolucionales. A continuación, cada una de estas y sus funciones:

- **Capa convolucional:** Su principal propósito es extraer características de una imagen. Consiste de un conjunto de filtros entrenables que realizan la suma del producto punto con los valores de la capa precedente. En la práctica, los valores de los filtros son aprendidos para su **activación** al encontrar ciertas características. Al ser colocados en cascada se obtienen diferentes niveles de abstracción.

- **Unidad de Rectificador Lineal:** Es utilizado después de cada convolucional. Es una operación que reemplaza los valores negativos por cero y su propósito es agregar no linealidad al modelo, eliminando la relación proporcional entre la entrada y salida.
- **Pooling:** Algoritmo utilizado para reducir las dimensiones mediante submuestreo, con el objetivo de disminuir los tiempos de procesamiento reteniendo la información más importante.
- **Capa totalmente conectada (Dense Layer):** Realiza la clasificación basada en las características extraídas por las capas de convolución y las reducidas por pooling. En esta capa todos los nodos están conectados con la capa precedente. En la figura 13 se muestra el diagrama general de las etapas de una red neuronal convolucional.

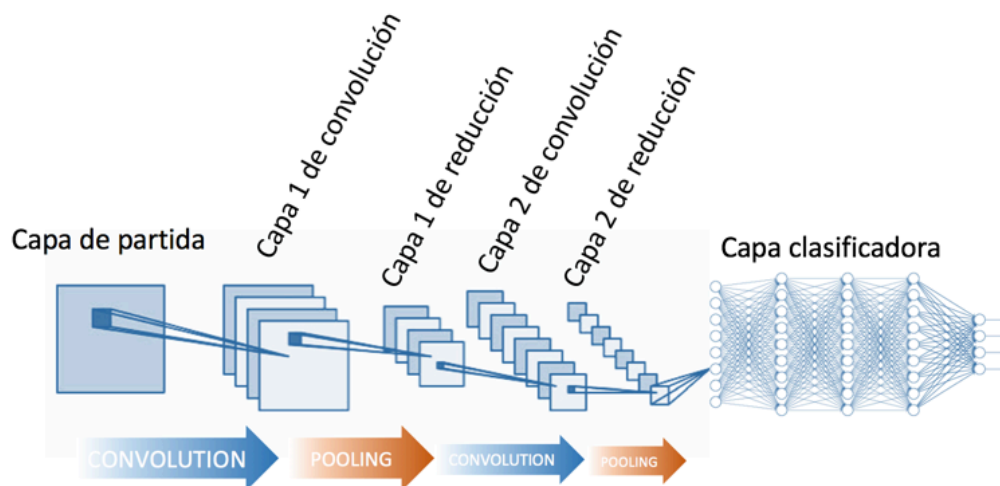


Fig. 13. Etapas de procesamiento en una red neuronal convolucional

Las capas de convolución y las de pooling se encargan de extraer características mientras que la capa totalmente conectada actúa como clasificador. Para el funcionamiento de este modelo debemos proceder al entrenamiento. Esto implica:

- 1) Inicializar todos los parámetros o pesos con valores aleatorios.
- 2) Utilizar una imagen de entrenamiento y utilizarla en el modelo.
- 3) Calcular el error total de las probabilidades resultantes del modelo.
- 4) Propagar hacia atrás para calcular el error de gradiente de todos los pesos en la red y utilizar gradiente descendente para actualizar estos valores y minimizar el error de salida [13].

Haciendo una analogía al cerebro, cada entrada en el volumen de salida 3D también puede interpretarse como una salida de una neurona que mira sólo una pequeña región en la entrada

y comparte parámetros con todas las neuronas de la izquierda y derecha espacial (ya que todos estos números resultan de aplicar el mismo filtro). Ahora analizamos los detalles de las conectividades neuronales, su disposición en el espacio y su esquema de intercambio de parámetros.

Conectividad local. Cuando se trata de entradas de alta dimensión, como imágenes, no es práctico conectar todas las neuronas en el volumen anterior. En su lugar, conectaremos cada neurona solo a una región local del volumen de entrada. La extensión espacial de esta conectividad es un hiperparámetro llamado **campo receptivo** de la neurona (equivalentemente, este es el tamaño del filtro). La extensión de la conectividad a lo largo del eje de profundidad es siempre igual a la profundidad del volumen de entrada.

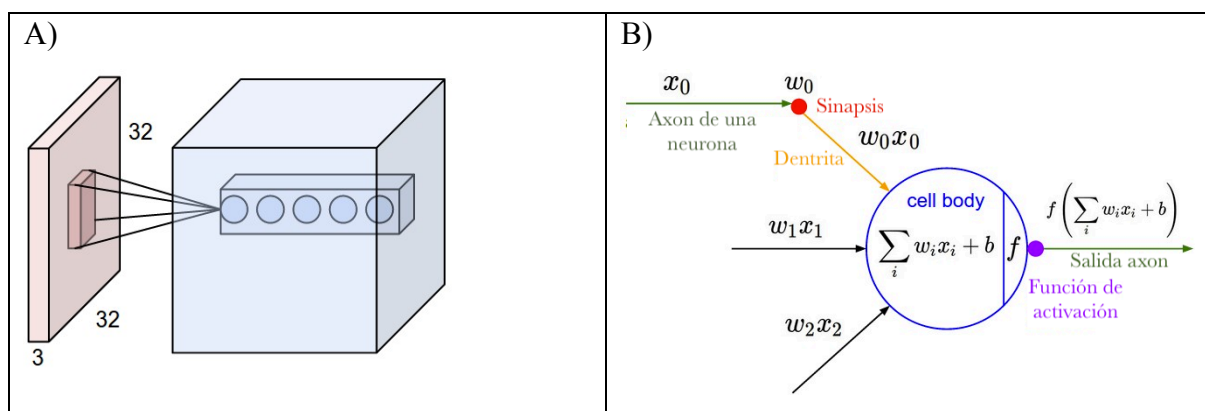


Fig. 14. Comparativa entre neuronas: A) una neuronal convolucional y B) una neurona.

A) un ejemplo de volumen de entrada en rojo (por ejemplo, una imagen de 32x32x3), y un ejemplo de volumen de neuronas en la primera capa convolucional. Cada neurona en la capa convolucional está conectada solo a una región local en el volumen de entrada espacialmente, pero a la profundidad completa (es decir, todos los canales de color). **B)** las neuronas de la red neuronal permanecen sin cambios: aún computan un producto puntual de sus pesos con la entrada seguida de una no linealidad, pero su conectividad ahora está restringida a ser local espacialmente [11].

2.6.2 Implementación de modelos de redes neuronales convolucionales.

Actualmente la implementación de modelos de redes neuronales convolucionales se da en lenguajes de programación como Python, Java o MatLab. Para la implementación de esta tesis se utilizó Python, por su gran cantidad de bibliotecas populares con las que cuenta. A continuación, se detallarán las herramientas.

2.6.2.1 Herramientas y librerías

Python es un gran lenguaje de programación de propósito general, pero con la ayuda de algunas librerías populares (numpy, scipy, matplotlib) se convierte en un entorno poderoso para la computación científica. A continuación, se describe brevemente librerías que se utilizan para la implementación de esta tesis.

Keras es una biblioteca de Redes Neuronales de Código Abierto escrita en Python. Es capaz de ejecutarse sobre TensorFlow, Microsoft Cognitive Toolkit o Theano. Está especialmente diseñada para posibilitar la experimentación en más o menos poco tiempo con redes de Aprendizaje Profundo. Sus fuertes se centran en ser amigable para el usuario, modular y extensible [51].

TensorFlow es una biblioteca de código abierto para aprendizaje automático a través de un rango de tareas, y desarrollado por Google para satisfacer sus necesidades de sistemas capaces de construir y entrenar redes neuronales para detectar y descifrar patrones y correlaciones, análogos al aprendizaje y razonamiento usados por los humanos [13].

2.6.2.2 Implementación de un modelo de red neuronal convolucional en Keras

La programación de una red neuronal convolucional de arquitectura simple se puede crear en Python con la librería de Keras. A continuación, una implementación simple de una arquitectura de red neuronal convolucional: consiste en una convolución seguida de un *max-pooling*. Aplicando 32 filtros usando una ventana de 5×5 para la capa convolucional y una ventana de 2×2 para la capa de *pooling* seguida de la función de activación ReLU. Procesa un tensor de entrada de tamaño (28, 28, 1) y lo especificamos mediante el valor del argumento `input_shape=(28, 28,1)` en nuestra primera capa:

```
from keras import layers
from keras import models

model = models.Sequential()
model.add(layers.Conv2D(32, (5,5), activation='relu', input_shape=(28,
28,1)))
model.add(layers.MaxPooling2D((2, 2)))

model.summary()
```

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 24, 24, 32)	832
max_pooling2d_1 (MaxPooling2D)	(None, 12, 12, 32)	0

```
=====  
Total params: 832  
Trainable params: 832  
Non-trainable params: 0
```

El número de parámetros de la capa conv2D corresponde a la matriz de pesos W de 5×5 y un sesgo b para cada uno de los filtros es 832 parámetros. El *max-pooling* no requiere parámetros puesto que es una operación matemática de encontrar el máximo.

La implementación simple de un modelo convolucional permite apilar varias capas como la construida anteriormente. El ejemplo, crea un segundo grupo de capas que tendrá 64 filtros con una ventana de 5×5 en la capa convolucional y una de 2×2 en la capa de *pooling*. En este caso, el número de canales de entrada tomará el valor de las 32 características obtenidas de la capa anterior, no hace falta especificarlo porque Keras lo deduce:

```
model = models.Sequential()  
model.add(layers.Conv2D(32, (5, 5), activation='relu', input_shape=(28, 28, 1))  
)  
model.add(layers.MaxPooling2D((2, 2)))  
model.add(layers.Conv2D(64, (5, 5), activation='relu'))  
model.add(layers.MaxPooling2D((2, 2)))
```

Si se muestra la arquitectura del modelo con:

```
model.summary()
```

Se puede ver:

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 24, 24, 32)	832
max_pooling2d_1 (MaxPooling2D)	(None, 12, 12, 32)	0
conv2d_2 (Conv2D)	(None, 8, 8, 64)	51264
max_pooling2d_2 (MaxPooling2D)	(None, 4, 4, 64)	0

```
=====  
Total params: 52,096  
Trainable params: 52,096  
Non-trainable params: 0
```

El tamaño de la segunda capa de convolución resultante es de 8×8 dado que parte de un espacio de entrada de $12 \times 12 \times 32$ y una ventana deslizante de 5×5 . El número de parámetros 51 264 corresponde a que la segunda capa tendrá 64 filtros, especificado en el argumento, con 801

parámetros cada uno (1 corresponde al sesgo, y luego tenemos la matriz W de 5×5 para cada una de las 32 entradas $((5 \times 5 \times 32) + 1) \times 64 = 51264$).

La salida de las capas *Conv2D* y *MaxPooling2D* es un tensor 3D de forma (*height*, *width*, *channels*). Las dimensiones *width* y *height* tienden a reducirse a medida que nos adentramos en las capas ocultas de la red neuronal. El número de filtros es controlado a través del primer argumento pasado a la capa *Conv2D* (habitualmente de tamaño 32 o 64).

El siguiente paso, ahora 64 filtros de 4×4 , consiste en añadir una capa densamente conectada (*densely connected layer*), que servirá para alimentar una capa final de *softmax* como la introducida en el capítulo 3 para hacer la clasificación:

```
model.add(layers.Dense(10, activation='softmax'))
```

En este ejemplo se ajustan los tensores a la entrada de la capa densa como la *softmax*, que es un tensor de 1D, mientras que la salida de la anterior es un tensor de 3D; por eso se tiene primero que aplanar el tensor de 3D a uno de 1D. Nuestra salida (4,4,64) se debe aplanar a un vector de (1024) antes de aplicar el *Softmax*.

En este caso, el número de parámetros de la capa *softmax* es $10 \times 1024 + 10$, con una salida de un vector:

```
model = models.Sequential()

model.add(layers.Conv2D(32, (5, 5), activation='relu',
input_shape=(28, 28, 1)))
model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Conv2D(64, (5, 5), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Flatten())
model.add(layers.Dense(10, activation='softmax'))
```

Con el método *summary()* podemos ver esta información sobre los parámetros de cada capa y formato de los tensores de salida de cada capa:

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 24, 24, 32)	832
max_pooling2d_1 (MaxPooling2D)	(None, 12, 12, 32)	0

conv2d_2 (Conv2D)	(None, 8, 8, 64)	51264
max_pooling2d_2 (MaxPooling2D)	(None, 4, 4, 64)	0
flatten_1 (Flatten)	(None, 1024)	0
dense_1 (Dense)	(None, 10)	10250
=====		
Total params:	62,346	
Trainable params:	62,346	
Non-trainable params:	0	

Observando este resumen, se aprecia que en las capas convolucionales es donde se requiere más memoria, y por ende más poder de cómputo para almacenar los datos. En cambio, en la capa densamente conectada de softmax se necesita poco espacio de memoria, pero en comparación se necesitan muchos parámetros para el modelo, que deberán ser aprendidos. Es importante ser consciente de los tamaños de los datos y de los parámetros porque cuando tenemos modelos basados en redes neuronales convolucionales, estos tienen muchas capas, como veremos más adelante, y estos valores pueden dispararse.

Una representación visual de la anterior información se muestra en la siguiente figura, donde observamos una representación gráfica de la forma de los tensores que se pasan entre capas y sus conexiones:

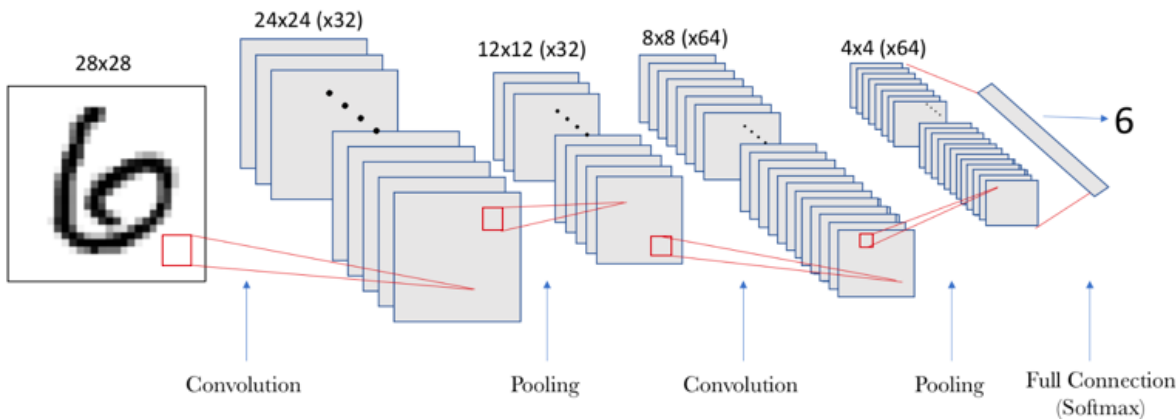


Fig. 15. Representación visual de los tensores entre las capas.

Una vez definido el modelo de la red neuronal ya en disposición de pasar a entrenar el modelo, es decir, ajustar los parámetros de todas las capas convolucionales. El código ya presentado anteriormente:

```

from keras.datasets import mnist
from keras.utils import to_categorical

(train_images, train_labels), (test_images, test_labels) =
mnist.load_data()

train_images = train_images.reshape((60000, 28, 28, 1))
train_images = train_images.astype('float32') / 255

test_images = test_images.reshape((10000, 28, 28, 1))
test_images = test_images.astype('float32') / 255

train_labels = to_categorical(train_labels)
test_labels = to_categorical(test_labels)

model.compile(loss='categorical_crossentropy',
              optimizer='sgd',
              metrics=['accuracy'])

model.fit(train_images, train_labels,
          batch_size=100,
          epochs=5,
          verbose=1)

test_loss, test_acc = model.evaluate(test_images, test_labels)

print('Test accuracy:', test_acc)

Test accuracy: 0.9704

```

Con la implementación del modelo anterior se pueden reconocer dígitos en imágenes utilizando una arquitectura simple de una red neuronal convolucional. También con este ejemplo se observan la cantidad de datos que se usan, así como el poder de computo necesario.

2.6.3 Clasificación y detección

Las redes neuronales convolucionales cumplen la tarea de clasificar imágenes, agregándoles una etiqueta. La arquitectura e implementación se puede observar en las secciones previas, a partir de este principio se han creado herramientas las cuales detectan y localizan en una imagen ciertos objetos de interés. En la figura 16 se muestran las diferentes tareas.

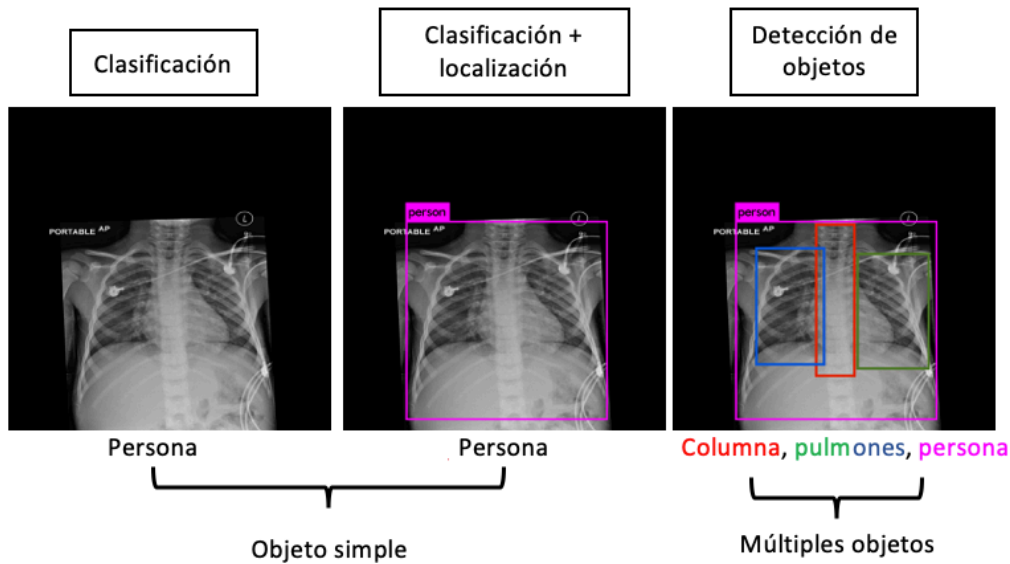


Fig. 16. Tareas que permiten hacer las redes neuronales convolucionales.

El principio que permite la localización de objetos es la regresión, esta consiste en devolver 4 números en lugar de una clase (x_0 , y_0 , ancho, alto), que están relacionados con un cuadro delimitador. Se entrenan estos modelos con imágenes y cuadros delimitadores donde se localiza el objeto, y utiliza la distancia L2 para calcular la pérdida entre el cuadro delimitador previsto y el recuadro delimitador real durante el entrenamiento.

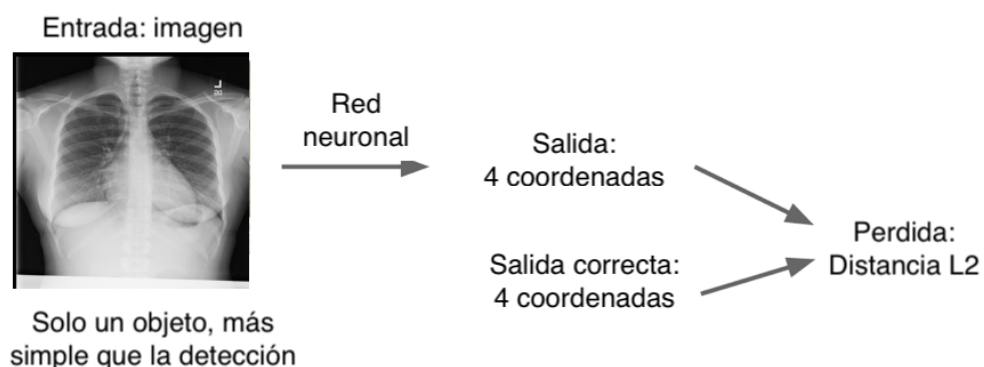


Fig. 17. Evaluación del procesamiento para la detección.

Normalmente, lo que se hace es unir otra capa completamente conectada en la última capa de convolución.

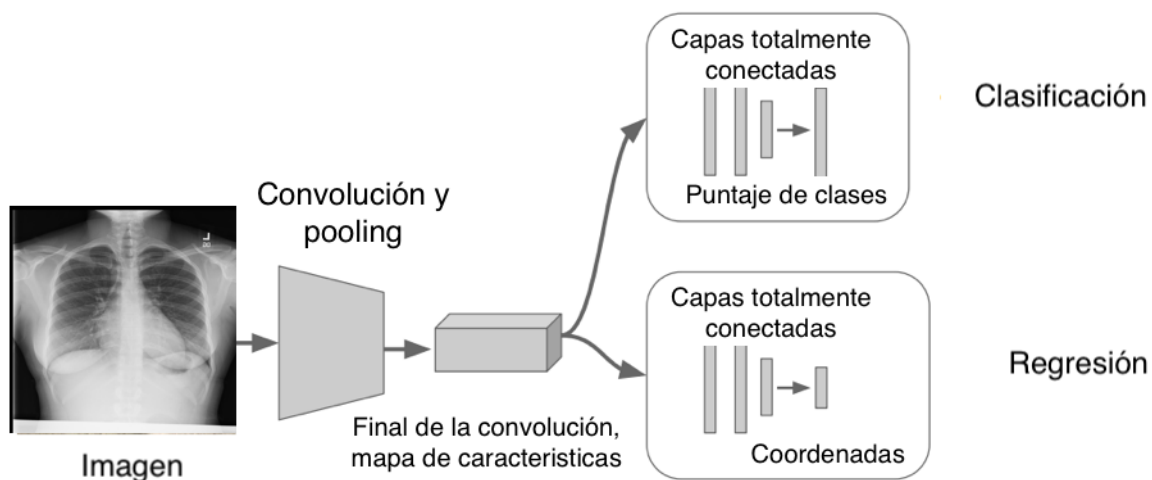


Fig. 18. Arquitectura de una red neuronal convolucional especializada en la detección.

Dependiendo de la arquitectura es como se realiza el procesamiento de detección, en algunos casos se puede realizar adjuntando al final de la CNN una capa completamente conectada. Otra forma de realizar el análisis de imágenes para la detección de objetos es introduciendo una imagen para que esta se parta en una cuadrícula, posteriormente se realiza el procesamiento de cada cuadrícula y se genera un mapa de probabilidades por clase, como se puede observar en la figura 19.

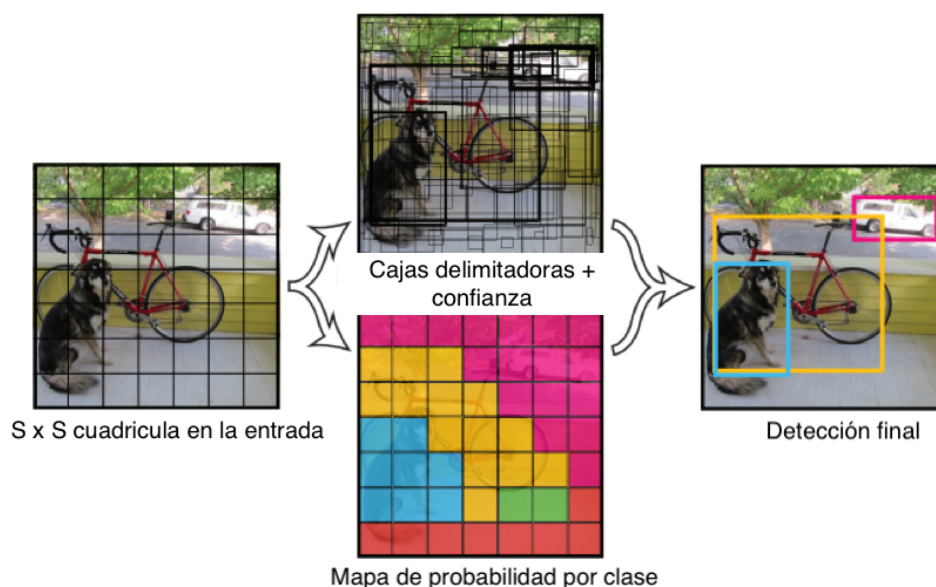


Fig. 19. Procesamiento para la detección de varios objetos.

Para evaluar la precisión en la detección es necesario comparar la intersección sobre la unión (IoU) entre la predicción y la verdad fundamental es más grande que algún umbral ($ex > 0.5$)



Fig. 20. Visualización de la evaluación de los detectores.

2.6.4 Modelos pre entrenados

El contenido previo muestra que el diseño de una red neuronal convolucional no es tarea sencilla, requiere la elección de un número de capas, la elección del tipo de capa, y encontrar cuál de estas se adapta mejor a las imágenes con las que se realizan las pruebas. Lo que se recomienda es hacer uso de modelos ya entrenados los cuales demuestran un alto grado de asertividad, así como también proponen la tarea de detección y clasificación de imágenes. Muchos de los modelos pre entrenados mencionados a continuación surgieron en ILSVRC¹ concurso orientado a la clasificación y detección de objetos por medio de modelos de visión computacional que se realiza cada año.

2.6.4.1 AlexNet

AlexNet es un modelo de CNN el cual fue uno de los pioneros en la implementación de este tipo de redes, Alex Krizhevsky, Ilya Sutskever y Geoffrey Hinton crearon una "gran red neuronal convolucional profunda" que se usó para ganar el ILSVRC en el 2012. Utilizaron un diseño relativamente simple, en comparación con las arquitecturas modernas. La red estaba formada por 5 capas convolucionales, capas de agrupación máxima, capas omitidas y 3 capas totalmente conectadas. La red que diseñaron se utilizó para la clasificación con 1000 categorías posibles [14].

¹ <http://www.image-net.org/challenges/LSVRC/>

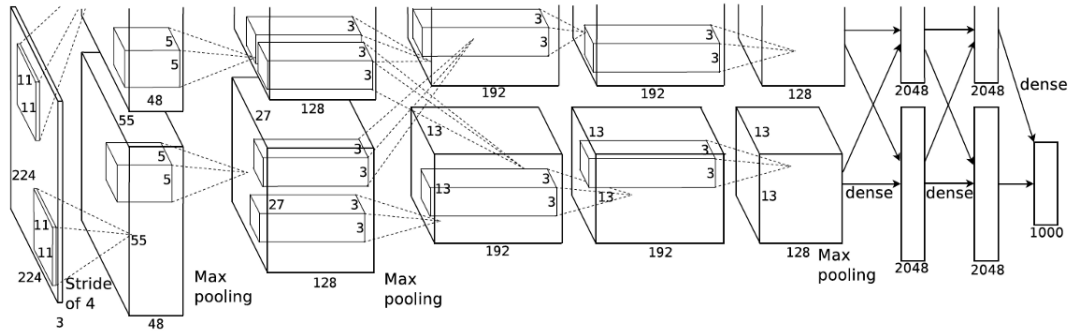


Fig. 21. Arquitectura del modelo de CNN AlexNet.

2.6.4.2 GoogleNet / Inception

Arquitectura de CNN profunda cuyo nombre en código es Inception logra transformar la tarea de clasificación y detección en el desafío de reconocimiento visual a gran escala de ImageNet 2014 (ILSVRC2014). El distintivo principal de esta arquitectura es la utilización mejorada de los recursos informáticos dentro de la red. Mediante un diseño cuidadosamente realizado, aumenta la profundidad y el ancho de la red a la vez que mantiene el rendimiento computacional constante. Para optimizar la calidad, las decisiones arquitectónicas se basaron en el principio de Hebbian y la intuición del procesamiento de múltiples escalas. Una encarnación particular de esta arquitectura, GoogLeNet, una red de 22 capas de profundidad, se utilizó para evaluar su calidad en el contexto de la detección y clasificación de objetos [15]. Esta red se diseñó teniendo en cuenta la eficiencia computacional, especialmente buscando la cantidad mínima de RAM para su entrenamiento y uso. Introducen los módulos Inception, los cuales sustituyen a una capa de convolución por varias capas de convoluciones más pequeñas y una última capa de filtro que se encarga de agrupar los resultados. Estos módulos producen mejores resultados que las convoluciones más grandes, adicionalmente mejoran la velocidad en el cálculo. El número total de parámetros es de unos 4 millones [14].

Precisión de entrenamiento en Inception v3

El modelo Inception v3 divide los datos para entrenar la red neuronal (como se menciona en la sección 2.2 de esta tesis) de la siguiente manera.

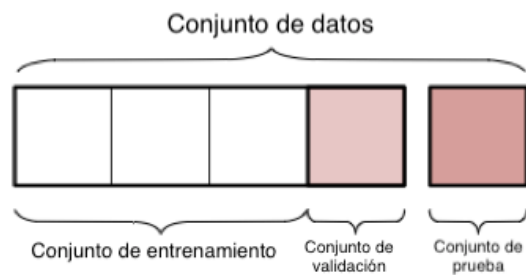


Fig. 22. División de los datos para el entrenamiento del modelo Inception v3.

La precisión inicial del entrenamiento muestra qué porcentaje de las imágenes utilizadas en el lote de entrenamiento actual se etiquetaron con la clase correcta. La precisión de validación es la precisión en un grupo de imágenes seleccionadas al azar de un conjunto diferente. Una medida real del rendimiento de la red es medir su desempeño en un conjunto de datos no incluidos en los datos de entrenamiento; esto se mide con la exactitud de la validación. Si la precisión del entrenamiento es alta, pero la exactitud en los ejemplos de validación sigue siendo baja, eso significa que la red está demasiado ajustada y se memorizan características particulares en las imágenes de entrenamiento que no son útiles en general. La entropía cruzada es una función de pérdida que da una idea de qué tan bien está progresando el proceso de aprendizaje. El objetivo del entrenamiento es hacer que la pérdida sea lo más pequeña posible, para que pueda saber si el aprendizaje funciona, observando si la pérdida continúa, ignorando el ruido a corto plazo.

2.6.4.3 VGG

La red tiene 13 capas de convolución, 5 de ellas con maxpool, y tres capas de redes profundas, dos de ellas de 4.096 y la última de 1.000. Por último, se tiene una capa de softmax para obtener los resultados de las clases. Esta red quedó en segunda posición en la competición ILSVRC del año 2014, detrás de GoogLeNet. No obstante, se ha incluido en el proyecto debido a que es considerada la mejor red actual en cuanto a extracción de características en imágenes. El número de parámetros aproximado es de 138 millones [16].

2.6.4.4 MobileNet

Modelos eficientes llamados MobileNets para aplicaciones de visión móviles e integradas. Las redes móviles se basan en una arquitectura optimizada que utiliza convoluciones separables en profundidad para construir redes neuronales ligeras y profundas, dos simples parámetros globales que intercambian eficientemente la latencia y la precisión. Estos hiperparámetros permiten que el creador de modelos elija el modelo del tamaño correcto para su aplicación en

función de las restricciones del problema. MobileNets en una amplia gama de aplicaciones y casos de uso que incluyen detección de objetos, clasificación, atributos de cara y geolocalización a gran escala [17].

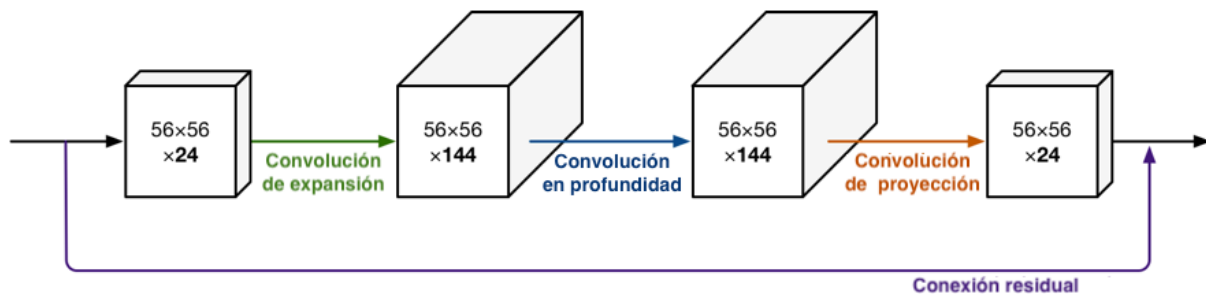


Fig. 23. Arquitectura del modelo de CNN MobileNet.

2.6.4.5 Desempeño

Los modelos pre entrenados previamente mencionados, permiten realizar una comparativa de qué tan exactos pueden llegar a ser, por lo que funcionan bastante bien con alta velocidad. La velocidad y el consumo de energía de la red son proporcionales al número de MAC (Acumulación múltiple), que es una medida del número de operaciones de multiplicación y adición fusionadas. A continuación, un gráfico que muestra la eficiencia de los modelos de redes neuronales convolucionales previamente mencionados.

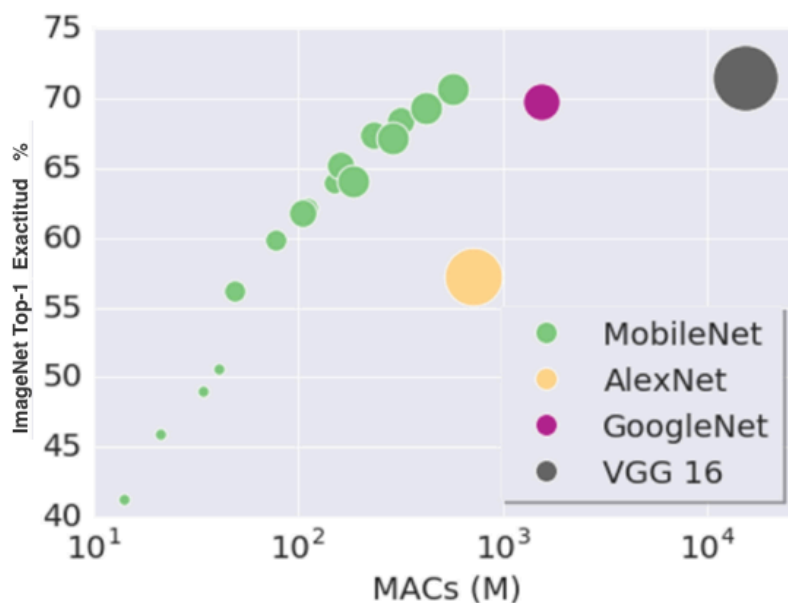


Fig. 24. Gráfico comparativo que muestra la eficiencia de los modelos previamente mencionados [18].

2.7 Registro de imágenes

El registro es un proceso que consiste en encontrar la transformación geométrica que alinea dos o más imágenes dadas en la mejor correspondencia posible. Este proceso en imágenes se puede aplicar a una gran gama de áreas, incluyendo el de la salud, por ejemplo, supongamos que tenemos un conjunto de imágenes de resonancias magnéticas del cerebro, para realizar de mejor manera la detección automatizada de alguna anomalía, sería adecuado contar con una resonancia del cerebro sano, contra una del cerebro con alguna anomalía. Para registrarlas será necesario realizar operaciones en las imágenes (como corrección de intensidad, reducción de tamaño de la imagen, etc) de tal manera que se encuentren lo mejor correspondientes. En la figura 25 se puede observar el ejemplo anteriormente descrito.

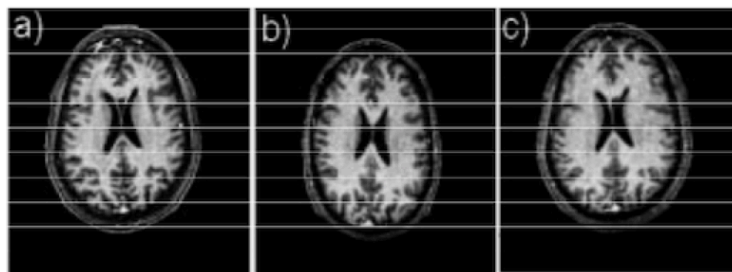


Fig 25. Ejemplo de registro de imágenes de resonancia magnética. a) Imagen destino, b) Imagen fuente, c) Imagen Fuente transformada.

Este proceso permite segmentar fácilmente cada una de las partes que integran una imagen, y existen diversas propuestas para realizar el registro como lo son el registro paramétrico en el cual por medio de unos pocos parámetros se logra una deformación de toda la imagen en una transformación rígida (el ejemplo anterior se puede considerar como una transformación rígida). También existe un modelo no paramétrico en el cual tenemos un campo de deformaciones que se aplican de forma independiente a cada punto de la imagen [19].

2.7.1 Redes de transformadores espaciales

En el año 2015 un grupo de Google Deepmind² propone la introducción de un módulo de transformador espacial. La idea básica es que este módulo transforme la imagen de entrada de manera que las capas posteriores tengan un tiempo más fácil para hacer una clasificación. En lugar de realizar cambios en la arquitectura CNN principal en sí, a los autores les preocupa hacer cambios en la imagen *antes de* que se introduzca en la capa convolución específica. Las

² <https://deepmind.com/>

dos cosas que este módulo espera corregir son la normalización de la postura (escenarios en los que el objeto está inclinado o escalado) y la atención espacial (llamar la atención sobre el objeto correcto en una imagen abarrotada). Para las CNN tradicionales, si se quisiera que el modelo fuera invariante a las imágenes con diferentes escalas y rotaciones, se necesita una gran cantidad de ejemplos de entrenamiento para que el modelo aprenda correctamente.

La entidad en los modelos CNN tradicionales que se ocupaban de la invariancia espacial era la capa de agrupación máxima. El razonamiento intuitivo detrás de esta capa fue que una vez que sabemos que una característica específica se encuentra en el volumen de entrada original (donde haya valores de activación altos), su ubicación exacta no es tan importante como su ubicación relativa a otras características. Este nuevo transformador espacial es dinámico de manera que producirá diferentes comportamientos (diferentes distorsiones /transformaciones) para cada imagen de entrada. No es tan simple y predefinido como un maxpool tradicional. A continuación cómo funciona este módulo transformador. El módulo consta de:

- Una red de localización que toma el volumen de entrada y genera los parámetros de la transformación espacial que se debe aplicar. Los parámetros (θ) pueden ser de 6 dimensiones para una transformación afín.
- La creación de una cuadrícula de muestreo que es el resultado de deformar la cuadrícula regular con la transformación afín (θ) creada en la red de localización.
- Un muestreador cuyo propósito es realizar una distorsión del mapa de características de entrada.

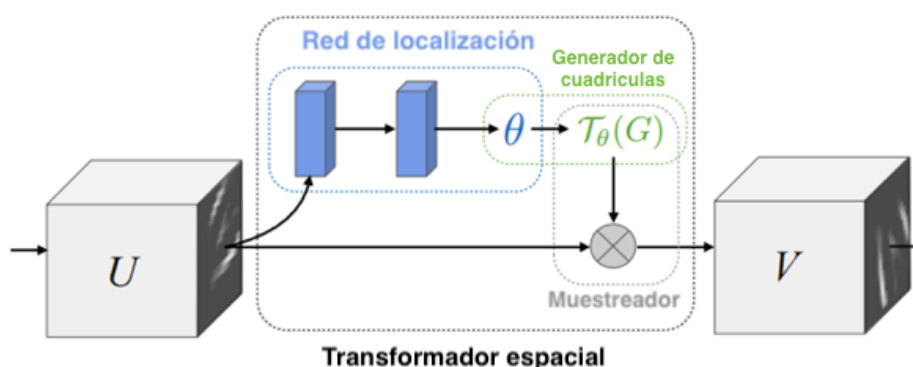


Fig. 26. Arquitectura del transformador espacial.

Este módulo se puede colocar en una CNN en cualquier momento y, básicamente, ayuda a la red a aprender cómo transformar los mapas de características de manera que minimice la función de costo durante el entrenamiento.

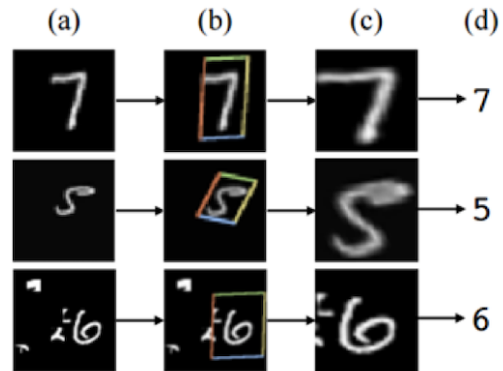


Fig. 27. Resultado obtenido al utilizar redes de transformadores espaciales, aplicada en el reconocimiento de dígitos.

Capítulo 3. CNNs en problemas de clasificación y registro de imágenes médicas

Con la información previamente descrita se puede apreciar que las redes neuronales convolucionales son una herramienta que permite la clasificación e identificación de patrones. En este capítulo se implementan arquitecturas pre entrenadas de redes neuronales convolucionales para: clasificar imágenes de resonancia magnética del cerebro, analizar tomografías de tórax para detectar tuberculosis creando un ensamble de clasificadores que mejoran la clasificación de tipos de tuberculosis y finalmente se propone una arquitectura de procesamiento basado en CNNs de radiografías del tórax para realizar el registro en imágenes de rayos X.

3.1 Reconocimiento de tejido tumoral en resonancias magnéticas del cerebro

Un tumor cerebral es un crecimiento de células anormales en el tejido del cerebro. Los tumores pueden ser benignos (sin células de cáncer) o malignos (con células cancerígenas que crecen muy rápido). Algunos son primarios, o sea, que comienzan en el cerebro. Otros son metastásicos, o sea, que comenzaron en alguna otra parte del cuerpo y llegan al cerebro [20]. Se combina la información de diferentes resonancias magnéticas (RM) para entrenar un modelo pre entrenado de red neuronal convolucional llamado Inception v3. La base de datos con la que se trabajó esta sección del proyecto fue tomada de 13 pacientes con tumores cerebrales en el Instituto Tecnológico de Montreal en 2010. Se consideran imágenes de resonancia magnética preoperatorias y postoperatorias. Cada paciente tiene una RM preoperatoria y postoperatoria después de la resección³.

Este grupo incluye imágenes preoperatorias de RM y post resección de ultra sonido (US). Los datos en formato MINC y mat (MATLAB) se proporcionan para todos los pacientes. Los datos de RM preoperatorios contienen el tumor, que se reemplaza por la cavidad de resección en las imágenes intraoperatorias de US. Dado que el tumor en la RM preoperatoria no existe en las imágenes posteriores a la resección de los US [21].

3.1.1 Trabajos relacionados

En [22] se describe un algoritmo de clasificación de tumores cerebrales que aplica, la segmentación Fuzzy C-Means (FCM) para separar el tumor y la región no tumoral del cerebro. Además, imágenes wavelet se extraen mediante el uso de Transformada wavelet discreta (DWT) multinivel. Finalmente, se incorporan redes neuronales profundas para la clasificación

³ Operación quirúrgica que consiste en separar total o parcialmente uno o varios órganos o tejidos del cuerpo.

de tumores cerebrales con alta precisión. Esta técnica se compara con los métodos de clasificación del vecino más cercano (KNN), análisis del discriminante lineal y optimización mínima secuencial. Este método arroja una tasa de precisión del 96.97% en el análisis de la clasificación de tumores cerebrales basada en redes neuronales, pero la complejidad es muy alta y el rendimiento es muy bajo.

En [23] se presenta la propuesta de modelado multiescala para análisis de imágenes de estudios de tumores cerebrales, se describe un modelo de crecimiento de tumores bio-fisiomecánico novedoso para analizar el crecimiento tumoral paso a paso de los pacientes. Se aplicó para gliomas y tumores sólidos con márgenes individuales para incautar el efecto significativo de la masa tumoral. Los métodos discretos y continuos se combinan para hacer un modelo de crecimiento tumoral. El esquema propuesto proporciona la posibilidad de segmentar explícitamente las imágenes cerebrales basadas en el tumor y en el registro. Esta técnica se utiliza principalmente para la segmentación del tejido cerebral, pero el tiempo de cálculo es alto.

En el artículo [24] se propone un algoritmo para la segmentación de la imagen del tumor cerebral multimodal (BRATS), se introduce una nueva segmentación del tumor cerebral, que también se denomina esquema de segmentación multimodal del tumor cerebral. También se combinan diferentes algoritmos de segmentación para lograr un alto rendimiento que el método existente, pero la complejidad es alta.

En una revisión sobre los métodos de segmentación de tumores cerebrales basados en resonancia magnética [25], se presenta la revisión de los algoritmos de segmentación tumoral cerebral, se discute acerca de los diversos métodos de segmentación, tales como la segmentación basada en regiones, la segmentación basada en umbrales, la segmentación difusa, la segmentación basada en atlas, la segmentación de campo aleatorio Markov (MRF), el modelo deformable, el modelo geométrico deformable, la precisión, la solidez y la validez que se analiza para estos métodos.

3.1.2 Pre procesamiento

Los datos de cada usuario se descargan en formato MINC, se realizó una conversión para extraer las resonancias en imágenes con extensión JPG como se mencionó en el primer capítulo.

A continuación (ver figura 28) se muestran algunas imágenes resultado de la RM

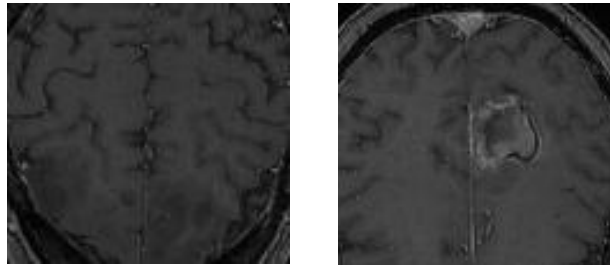


Fig. 28. Imagen de RM utilizada para el entrenamiento.

A cada paciente se le realizaron de 66 a 122 resonancias, las cuales contenían tejido tumoral, se seleccionaron 9 pacientes para crear una carpeta total de tumores cerebrales, y otra donde no se cuenta con presencia de tejido tumoral. El total de imágenes de la primera carpeta es 346, la segunda carpeta cuenta con 493 imágenes. A partir de esta clasificación de imágenes se entrenó la red neuronal.

3.1.3 Procesamiento

Con las imágenes organizadas en dos carpetas una por clase, se realizó la evaluación preliminar en la clasificación utilizando el modelo de Inception v3. Para la implementación se utilizaron librerías de Python como los son: Tensorflow y Keras anteriormente descritas, las cuales permiten realizar las operaciones matemáticas para el procesamiento de datos y entrenamiento. El número de iteraciones para el entrenamiento de la red neuronal fue de 4000.

Los resultados del entrenamiento son los siguientes, ver Tabla 5:

Tabla 5. Resultados de la evaluación del problema de clasificación de tumores cerebrales.

Entropía cruzada	0.023125
Exactitud de la validación	92.0% (N=100)
Exactitud final de la prueba	98.8% (N=81)

Como se puede observar en la tabla 5 la entropía cruzada es de 0.023125 lo que significa que mide, en promedio, cómo de sorprendido queda el modelo cuando descubre el valor correcto correspondiente a un dato real, para mayor información consulte la sección 2.5.1 de esta tesis. Se mencionó en la sección 2.6.4.2 que al entrenar el modelo Inception v3 este divide los datos en un conjunto de entrenamiento, validación y prueba, respecto a la exactitud de la validación se interpreta como a partir de 100 imágenes (lo que significa N), 92 fueron clasificadas correctamente. Por último, la exactitud final de la prueba indica el resultado que de 81 imágenes el 98.8% fueron clasificadas correctamente.

3.1.4 Pruebas

Con un total de trece pacientes tomados de la base de datos antes mencionada, tres de ellos fueron utilizados para las pruebas. De los cuales se muestran los siguientes resultados.

Paciente uno

El paciente uno, cuenta con un total de 85 resonancias magnéticas, la red neuronal convolucional no fue entrenada con ninguna de estas imágenes y al hacer un análisis de cada resonancia magnética con la red, muestra la existencia de tejido tumoral a partir de la resonancia número 50, esto se puede observar en la figura 29.

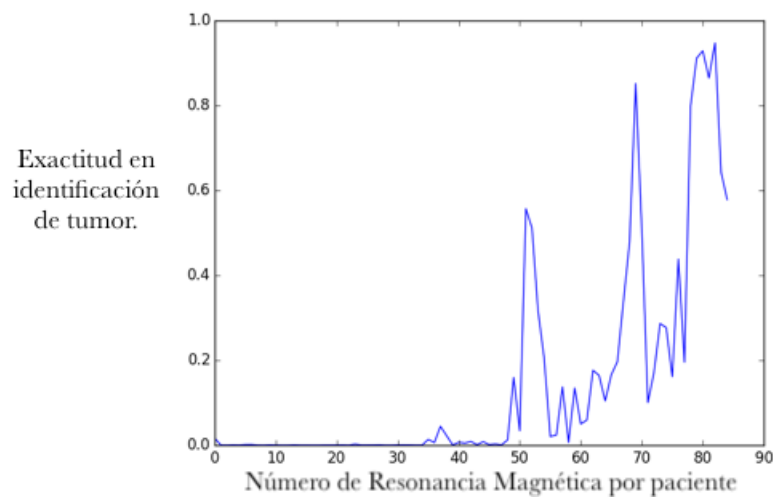


Fig. 29. Gráfica del paciente uno evaluando sus 85 resonancias magnéticas.

El resultado de la gráfica 29 muestra la evaluación del paciente uno en cada una de sus resonancias magnéticas que se ven representadas por el eje de las x , el eje de las y es la predicción la red neuronal que va de cero a uno, donde cero significa que no existe presencia de tejido tumoral y uno significa que existe tejido tumoral. En esta grafica varía en la exactitud para identificar la presencia de tejido tumoral.

Paciente dos

El paciente dos, cuenta con un total de 85 resonancias magnéticas, y al ser analizadas cada resonancia magnética con la CNN, nos muestra la existencia de tejido tumoral, con una probabilidad de hasta 99 por ciento, como se muestra en la figura 30.

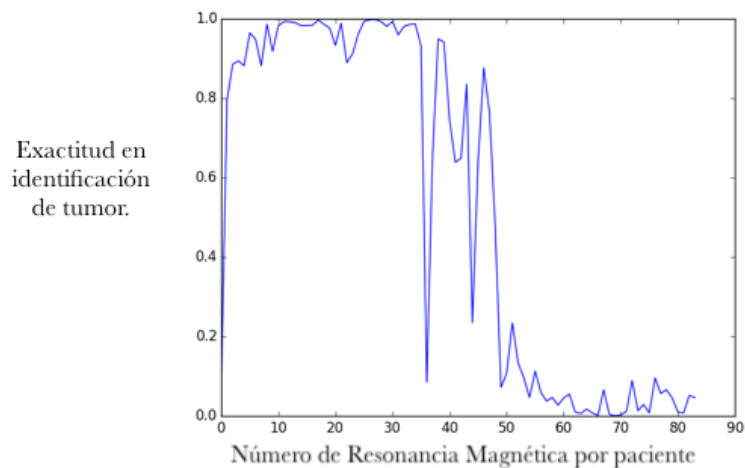


Fig. 30. Gráfica del paciente dos evaluando sus 85 resonancias magnéticas.

El resultado de la gráfica 30 muestra la evaluación del paciente dos en cada una de sus resonancias magnéticas que se ven representadas por el eje de las x, el eje de las y es la predicción la red neuronal que va de cero a uno, donde cero significa que no existe presencia de tejido tumoral y uno significa que existe tejido tumoral. En esta grafica la red neuronal muestra una exactitud variable en este rango de resonancias.

Paciente tres

El paciente tres, cuenta con un total de 85 resonancias magnéticas y al ser analizadas cada una de estas con la CNN, nos muestra la existencia de tejido tumoral, a partir de la resonancia 50, se puede observar en la figura 31.

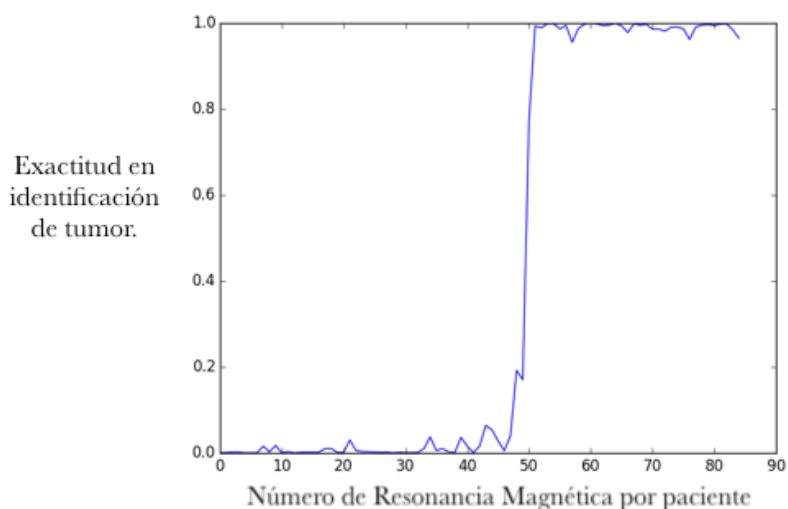


Fig. 31. Gráfica del paciente tres de sus 85 resonancias magnéticas.

El resultado de la gráfica 31 muestra la evaluación del paciente tres en cada una de sus resonancias magnéticas que se ven representadas por el eje de las x, el eje de las y es la

predicción la red neuronal que va de cero a uno, donde cero significa que no existe presencia de tejido tumoral y uno significa que existe tejido tumoral. En esta grafica la red neuronal encuentra con un alto grado de exactitud constante la presencia de tejido tumoral en este paciente.

En los tres pacientes anteriormente analizados existía la presencia de tejido tumoral encontrada por la red neuronal. Lo anterior sugiere que la red neuronal convolucional fue entrenada adecuadamente y que la red tiene la capacidad de detectar la presencia de tejido tumoral.

3.1.5 Aplicación móvil

Para mostrar la posibilidad de aplicar estas herramientas en dispositivos móviles, se crea una aplicación móvil implementada en Android, la cual tiene la capacidad de utilizar la cámara del celular y sobre cada imagen capturada muestra la etiqueta de tumor o sin tumor. La aplicación se realiza con la misma base de datos, el entrenamiento de una CNN basada en el modelo MobileNet para la clasificación de imágenes. El número de imágenes para el entrenamiento del modelo fue de: 492 imágenes sin tumor cerebral y 395 imágenes con tumor cerebral. Los resultados del entrenamiento son los que se muestran en la tabla 6 después de entrenarse con 2000 pasos.

Tabla 6. Resultados de la evaluación de tumores cerebrales de la aplicación móvil.

Entropía cruzada	0.023113
Exactitud de la validación	98.0%(N=100)
Exactitud final de la prueba	97.1% (N=206)

En la tabla 6 se observa la entropía cruzada baja, lo que significa que el modelo se encuentra poco sorprendido cuando descubre el valor correcto correspondiente a un dato real, para mayor información consulte la sección 2.5.1 de esta tesis. Se mencionó en la sección 2.6.4.2 que al entrenar el modelo Inception v3 este divide los datos en conjuntos, este proceso también lo realiza MobileNet al ser reentrenado. Respecto a la exactitud de la validación se interpreta como a partir de 100 imágenes (lo que significa N), 98 fueron clasificadas correctamente. Por último, la exactitud final de la prueba nos da los resultados de 206 imágenes el 97.1% fueron clasificadas correctamente.

La clasificación basada en TensorFlow implementada en Android, utiliza el modelo Google Inception v3 para clasificar los marcos de la cámara en tiempo real, mostrando los mejores resultados en una superposición en la imagen de la cámara. Como se puede ver a continuación.

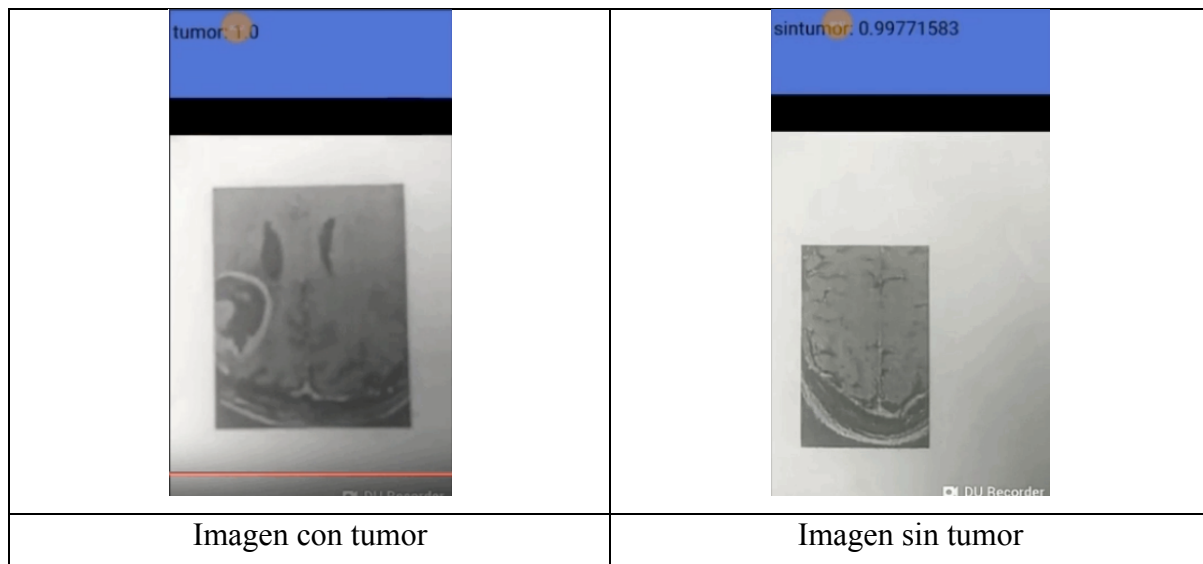


Fig. 32. Capturas de pantalla de la aplicación móvil para detectar tumores cerebrales en imágenes médicas.

3.1.6 Conclusiones

El objetivo principal de esta sección de la tesis es mostrar los resultados alcanzados por el entrenamiento de los dos modelos de CNN basados en el modelo de Inception-v3 y MobileNet. Estos muestran la posibilidad de aplicar el proceso de transferencia de conocimiento alcanzando resultados de hasta 97.1% de precisión, aplicados en el reconocimiento de tejido tumoral, sin necesidad de equipos de cómputo especializados. Comparando este método con los métodos descritos en el estado del arte, se obtiene una mejora marginal en los resultados. Respecto al rendimiento del algoritmo de clasificación el implementar redes neuronales convolucionales en Python tiene el potencial de utilizar estas herramientas en cualquier dispositivo móvil.

3.2 Ensamble de aprendizaje profundo para clasificar tuberculosis.

En esta sección se utiliza una CNN para clasificar las tomografías computarizadas de tórax con presencia de tuberculosis en cinco tipos (infiltrativa, focal, tuberculoma, miliar y fibrocavernosa). El trabajo muestra el desempeño de la red bajo diferentes esquemas de entrenamiento y describe un modelo de aprendizaje por transferencia de conocimiento que combinado con un ensamble de CNNs puede mejorar la precisión de la clasificación.

La tuberculosis (TB) es causada por bacterias (*Mycobacterium tuberculosis*) que afectan con mayor frecuencia a los pulmones. Después de 130 años del descubrimiento de esta bacteria, la enfermedad sigue siendo una amenaza persistente y una de las principales causas de muerte en todo el mundo. El mayor desastre que puede ocurrirle a un paciente con TB es que los organismos se vuelven resistentes a dos o más de los medicamentos estándar. A diferencia de la tuberculosis sensible a los medicamentos, su forma resistente a múltiples medicamentos es mucho más difícil y costosa de recuperar. Por lo tanto, la detección temprana del estado de resistencia al fármaco es de gran importancia para un tratamiento eficaz. Los métodos de detección más utilizados son costosos o llevan demasiado tiempo (hasta varios meses). Por lo tanto, existe la necesidad de métodos rápidos de detección. Uno de los posibles enfoques para esta tarea se basa en el análisis de imágenes por tomografía computarizada (TC). Otra tarea difícil es la detección automática de los tipos de TB utilizando volúmenes de TC [26]. Las últimas pruebas moleculares son rápidas y precisas, sin embargo, su disponibilidad y costo son una gran preocupación. La radiografía de tórax (CXR, por sus siglas en inglés) también se usa para propósitos de detección que es el método recomendado por la Organización Mundial de la Salud para la detección temprana de la tuberculosis. Los sistemas de diagnóstico asistido por computadora se pueden usar para acelerar el proceso de búsqueda activa de casos en áreas que carecen de un radiólogo experto al ayudarlos [27].

3.2.1 Trabajo relacionado

En los últimos años, se han publicado varios artículos revolucionarios sobre el diagnóstico asistido por computadora (CAD) en CXR, a pesar de que se necesita más investigación en este campo para cumplir con los requisitos prácticos de rendimiento. En el 2013 el Instituto Nacional de Salud Bethesda [28] propuso un método para el diagnóstico utilizando la segmentación pulmonar basada en el algoritmo de la Cuenca. Se utiliza una pequeña ventana de exploración en el área pulmonar segmentada para encontrar el valor de intensidad promedio y máximo de los píxeles. La clasificación del centro de la ventana se realiza comparando cada píxel de la ventana con un valor de umbral seleccionado entre el valor de intensidad promedio y máximo. Leibstein y Nel [29] propusieron un método que utiliza técnicas de procesamiento de imágenes como el Laplaciano de Gaussian (LoG) y el patrón binario local (LBP), para detectar manualmente los nódulos de TB al resaltarlos. Su artículo también se centra en el procesamiento previo de segmentar los pulmones y suprimir las costillas utilizando energía dual. El método propuesto para el diagnóstico de TB por Ratnasari [30], [31] utiliza la característica estadística para el análisis de textura. En este método, el vector de características

se crea mediante el cálculo de características estadísticas, particularmente la media, la desviación estándar, la asimetría y la entropía. Luego, la dimensionalidad de la característica se realiza para la reducción hasta una y dos dimensiones utilizando el método de análisis de componentes principales (PCA) y la clasificación final se realiza utilizando el clasificador de distancia mínima. Las comparaciones de los resultados generados al seleccionar una salida de PCA unidimensional y bidimensional también se presentan en el documento.

3.2.2 Método

El modelo de red neuronal convolucional empleado para clasificar las imágenes es el de Inception v3 que se ejecuta en el Laboratorio Nacional de Supercómputo del sureste de México (LNS), que cuenta con varios segmentos de cómputo científico [32]. Las imágenes de CT en el conjunto de datos son de diferente tamaño y las imágenes tienen un canal de color, escala de grises. La base de datos fue proporcionada por ImageClef [33] la cual tiene como objetivo mejorar la clasificación de imágenes con presencia de tuberculosis.

3.2.3 Evaluación

Inception-v3 es una de las herramientas diseñadas para clasificar imágenes. Por otro lado, TensorFlow es una herramienta para el aprendizaje automático. Aunque contiene una amplia gama de funcionalidades, TensorFlow está diseñado principalmente para modelos de redes neuronales profundas. Los modelos modernos de reconocimiento de imágenes tienen millones de parámetros; entrenarlos desde cero requiere una gran cantidad de datos de entrenamiento etiquetados y una gran cantidad de potencia de cómputo (cientos de horas de GPU o más). El aprendizaje por transferencia es una técnica que reduce gran parte de esto tomando una pieza de un modelo que ya ha sido entrenado en una tarea relacionada y reutilizándolo en un nuevo modelo.

3.2.3.1 Tipos de tuberculosis evaluados

Existen varios tipos de tuberculosis, cada uno de los cuales se comporta de manera diferente cuando se expande en los pulmones, como se puede observar en la figura 33.

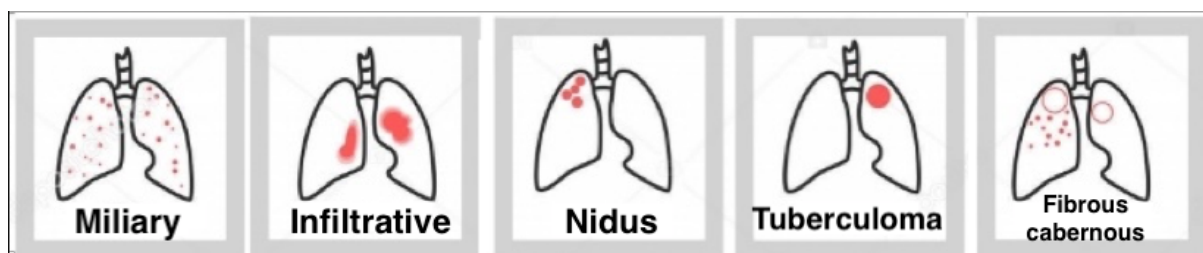


Fig. 33. Tipos de tuberculosis proporcionados por la base de datos.

3.2.3.2 Información de la base de datos.

La base de datos que se utilizó para realizar las pruebas tiene un total de 5 carpetas, con imágenes en formato JPG. Cada imagen tiene una dimensión de 512 x 512 píxeles. En la tabla 7 se muestra el número de imágenes por tipo de tuberculosis utilizadas en el proyecto.

Tabla 7. Resultados de la clasificación de tipo de tuberculosis.

Carpeta	Número de imágenes.
Infiltrative tuberculosis	1,493
Focal (Nidus) tuberculosis	1,397
Tuberculoma	1,348
Miliary tuberculosis	1,623
Fibrous cabernous	1,310
Total:	7,171

A continuación, algunos ejemplos de imágenes con las que se entrenó la CNN:

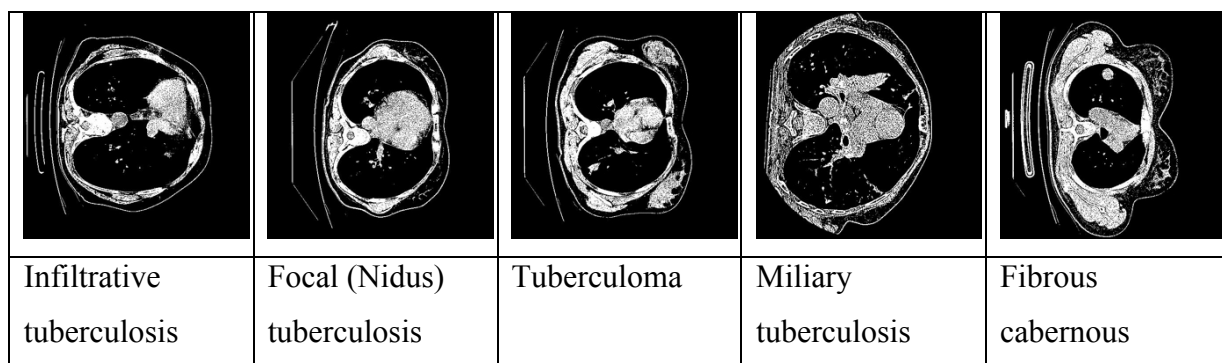


Fig. 34. Ejemplos de las imágenes encontradas en la base de datos.

3.2.3.3 Evaluación de una red neuronal convolucional para clasificar

Como se muestra en la tabla anterior, la base de datos tiene un total de 7,171 imágenes, estas imágenes divididas en 5 carpetas dieron el siguiente resultado al entrenar el modelo Inception V3.

Tabla 8. Resultados de la clasificación de tipo de tuberculosis con un solo clasificador.

Prueba	Exactitud final	Entropía cruzada	Descripción
1	78.7%(N=757)	0.507203	Se introducen las 5 carpetas, sin modificar su contenido.

La exactitud de la prueba final de la red neuronal convolucional entrenada s muestra un 78.7 por ciento de éxito en un total de 757 imágenes de prueba, la entropía cruzada puede ser alta, y como se mencionó antes en la sección 2.5.1 de esta tesis, la entropía cruzada es una función de pérdida que da una idea de lo bien que avanza el proceso de aprendizaje.

3.2.3.4 Evaluación de la red con clases balanceadas

Se reenvía el mismo modelo inicial, pero con la diferencia de que todas las carpetas tienen el mismo número de imágenes (cada carpeta tiene un total de 1310 imágenes seleccionadas al azar), ver tabla 9.

Tabla 9. Resultados de la clasificación de tipo de tuberculosis balanceando las clases

Prueba	Exactitud final	Entropía cruzada	Descripción
2	80.2% (N=660)	0.498975	Todas las carpetas tienen un total de 1310 imágenes seleccionadas al azar.

La exactitud de la prueba final de la red neuronal convolucional entrenada nos muestra un 80.2 por ciento de éxito en un total de 660 imágenes de prueba, la entropía cruzada ha disminuido marginalmente.

3.2.3.5 Evaluación de un ensamble de clasificadores, una CNN por clase

En las siguientes evaluaciones, se crearon dos carpetas para cada prueba, la primera carpeta: una clase específica, y la segunda carpeta contiene las otras 4 clases, ver tabla 10.

Tabla 10. Resultados de la clasificación de tipo de tuberculosis por ensamble de CNNs

Prueba	Exactitud final	Entropía cruzada	Descripción
3	85.6%	0.316147	Infiltrative tuberculosis vs otros
4	83.2%	0.186951	Focal (Nidus) tuberculosis vs otros
5	91.6%	0.209800	Tuberculoma vs otros
6	89.2%	0.204614	Miliary tuberculosis vs otros
7	91.1%	0.159215	Fibrous cabernous vs otros

En esta evaluación se observa que la precisión mejora, en comparación con las dos pruebas anteriores. También que la entropía cruzada es baja, por lo que el objetivo del entrenamiento es hacer que la pérdida sea lo más pequeña posible, para que pueda saber si el aprendizaje funciona, observando si la pérdida continúa, ignorando el ruido a corto plazo.

En estas pruebas se constató que al clasificar todas las imágenes usando los cinco clasificadores, la salida del clasificador con menor entropía se comportó de manera óptima.

3.2.4 Conclusiones

En esta sección, se entrenan varias CNN para clasificar imágenes de tomografías computarizadas. Se muestra que, al usar clasificadores especializados en cada tipo de tuberculosis, el enfoque propuesto puede alcanzar hasta el 88.1% de exactitud, así como reducir la entropía a 0.21 si se considera el promedio de los cinco clasificadores especializados, revela una mejora importante con respecto al considerar una sola red neuronal para los cinco tipos de tuberculosis. Comparando este método con los métodos descritos en el estado del arte, la clasificación de imágenes con los diferentes tipos de tuberculosis mejora cuando se realiza la clasificación por medio de ensambles de clasificadores en 8%. En trabajos futuros se buscará realizar una comparación más rigurosa.

3.3 Registro de tórax en imágenes de rayos x utilizando CNN

3.3.1 Introducción

El registro de imágenes es el proceso que busca la transformación geométrica para alinear dos imágenes en la mejor correspondencia posible. Este proceso puede aplicarse en imágenes médicas para mejorar los algoritmos de clasificación o la detección automática de patrones asociados con enfermedades. En esta sección se propone el uso de redes neuronales convolucionales y el análisis de componentes principales (*PCA: Principal Component Analysis*) para registrar imágenes de tórax. Luego se comparan los resultados de detección de neumonía en imágenes no registradas frente a imágenes registradas. La evaluación muestra una mejora marginal en la precisión de clasificación cuando las imágenes se alinean por el método propuesto.

Las imágenes médicas permiten observar patrones que dependen de ciertos aspectos, como el ángulo en el que se tomó la imagen, la edad del paciente y la calidad de la imagen. Por este motivo se realiza el proceso de registro en imágenes de rayos X del tórax. El resultado de un proceso de registro es obtener los parámetros de las transformaciones de traslación, cambio de escala y rotación que alinean las imágenes.

Las redes neuronales convolucionales cuentan con capas de agrupación máxima local que permiten que una red sea invariante espacialmente a la posición de las características. Sin embargo, debido al soporte espacial típicamente pequeño para la agrupación máxima (p. Ej., 2×2 píxeles), esta invariancia espacial sólo se realiza a través de una jerarquía profunda de agrupación máxima y contribuciones, y los mapas de características intermedias (activaciones de la capa convolucional). Sin embargo, las CNNs no son invariantes a grandes transformaciones de los datos de entrada [34, 35]. Esta limitación de las CNN se debe a que sólo tiene un mecanismo de agrupación predefinido y limitado para tratar las variaciones en el espacio de los datos.

Por este motivo se plantea realizar el proceso de registro en imágenes médicas, especializándonos en rayos X del tórax, para posteriormente entrenar el modelo de red neuronal convolucional Inception V3 [36] especializada en la clasificación de imágenes y ver como varían los resultados de imágenes sin procesar respecto a imágenes procesadas con nuestra propuesta.

Para realizar este proceso se propone la utilización de procesamiento digital de imágenes, entrenamiento de redes neuronales convolucionales especializadas en detección (modelo

MobileNet que ha sido descrito en el capítulo dos de esta tesis) y el método de PCA (que será descrito posteriormente en esta sección) que permite ajustar la orientación de las imágenes.

3.3.2 Base de imágenes de rayos X de sujetos sanos y con neumonía

La Sociedad Radiológica de América del Norte (RSNA) es una sociedad internacional de radiólogos, médicos y otros profesionales con más de 54,000 miembros de 146 países de todo el mundo. Este trabajo utiliza una base de datos proporcionada por la RSNA [37], que contiene un conjunto de imágenes que muestran la presencia de neumonía, así como características que muestran pulmones con anomalías. La acción del registro está condicionado a muestras de dichas imágenes, para la realización de esta sección.

3.3.3 Trabajos relacionados

Actualmente existe un interés en aplicar técnicas de aprendizaje profundo en imágenes médicas debido al gran éxito y los resultados prometedores obtenidos al clasificar las imágenes. En especial las redes neuronales convolucionales [38] tienen la capacidad de clasificar e identificar ciertos patrones en imágenes, aunque no es un invento reciente, ahora vemos una gran cantidad de modelos basados en CNN que logran resultados de vanguardia en clasificación [39, 40, 41], localización [42], segmentación [43] y reconocimiento de acciones tareas [44], entre otras.

A continuación, se presentan trabajos relacionados con la tarea de clasificación en imágenes médicas para detectar neumonía, posteriormente se muestran también trabajos sobre el registro en imágenes utilizando redes neuronales convolucionales y finalmente sobre registro de radiografías del tórax.

3.3.3.1 Clasificación de neumonía

La investigación mundial sobre el análisis de imágenes médicas por medio de redes neuronales convolucionales es una propuesta que muestra resultados de alto impacto. Stanford en 2017 presenta una red neuronal convolucional CheXNet: Detección de neumonía en radiografías de tórax con aprendizaje profundo [45], analizando imágenes y diagnósticos de los pacientes, esto se traduce en una precisión del 85% utilizando una comparación entre la herramienta y un médico. La investigación centrada en la detección muestra que las herramientas actuales son de gran ayuda, pero aún están por debajo de la detección automática óptima (reducir el error en las detecciones).

3.3.3.2 Registro en imágenes utilizando redes neuronales convolucionales

En el año 2015 un grupo de Google Deepmind propone la introducción de un módulo de transformadores espaciales para complementar el procesamiento de las redes convolucionales. La idea básica es que este módulo transforme la imagen de entrada de manera que las capas posteriores reduzcan el tiempo de entrenamiento para lograr clasificar los ejemplos. En lugar de realizar cambios en la arquitectura CNN principal en sí, los autores proponen hacer cambios en la imagen antes de que se introduzca en la capa de convolución específica. Los dos aspectos que este módulo espera corregir son la normalización de la postura (escenarios en los que el objeto está inclinado o escalado) y la atención espacial (llamar la atención sobre el objeto correcto en una imagen de múltiples objetos). Para las CNN tradicionales, si se quisiera que su modelo fuera invariante a las imágenes con diferentes escalas y rotaciones, necesitaría una gran cantidad de ejemplos de entrenamiento para que el modelo aprenda correctamente [46] para mayor información ver la sección 2.7.1 de esta tesis.

3.3.3.3 Registro de radiografías de tórax

En el artículo una evaluación de los métodos de registro de imágenes para radiografías del tórax [47] se estudia y evalúa el rendimiento de diferentes algoritmos de registro basados en diversos métodos de extracción y comparación de características en imágenes radiográficas de tórax. En particular se estudia, la combinación de tres descriptores de puntos de interés (SIFT (algoritmo usado en visión artificial para extraer características relevantes de las imágenes que posteriormente pueden usarse en reconocimiento de objetos, detección de movimiento, registro de la imagen y otras tareas), SURF (algoritmo capaz de obtener una representación visual de una imagen y extraer una información detallada y específica del contenido) y ORB (detector local rápido y robusto de características)). Estos se usaron como detectores de características, para la búsqueda de correspondencias, los métodos SIFT y SURF también se emplearon individualmente. Las pruebas se realizaron en imágenes de rayos X de tórax, utilizando datos reales de pacientes tomados en diferentes momentos. La mayor precisión de registro se logra cuando los descriptores SIFT y SURF se usan juntos para la extracción de puntos de interés, y el algoritmo SIFT se usa para la comparación de características.

En la detección automática de la rotación en las radiografías de tórax utilizando la medida principal de orientación de las costillas para el control de calidad [48], se presenta un método para detectar pulmones rotados en radiografías de tórax para el control de calidad y aumentar la detección automatizada de anomalías. El método calcula una medida principal de orientación

de las costillas utilizando una técnica de histograma de línea generalizada para el control de calidad y, por lo tanto, aumenta la detección automatizada de anomalías.

En el artículo extracción automática de puntos de control para imágenes de rayos X de tórax y registro elástico [49] se presenta un algoritmo para la detección de puntos de control y la extracción de la región de interés (ROI) para las imágenes de rayos X del tórax. Una estrategia de búsqueda para obtener los puntos de control y la distribución espacial de la intensidad en imágenes de rayos X del tórax, utilizando: la media, la varianza y la diferencia.

3.3.4 Algoritmo de registro propuesto

Esta sección describe la formulación del proceso de registro especializado en imágenes de rayos X del tórax. El mecanismo del proceso de registro se divide en siete etapas, como se muestra en la figura 35.

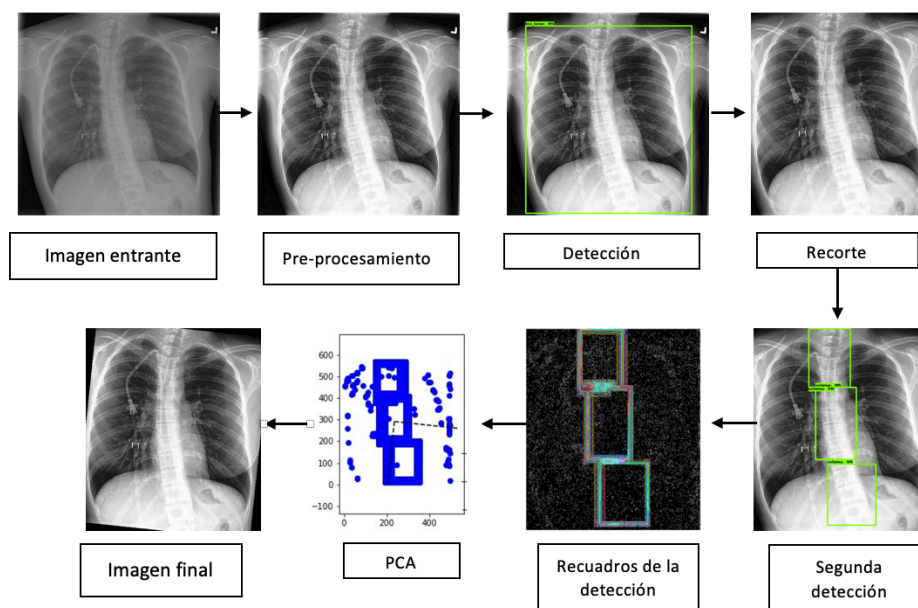


Fig. 35. Proceso de registro de tórax en imágenes de rayos x utilizando CNN

La primera etapa es el pre-procesamiento de las imágenes, este incluye la ecualización de las imágenes y reajuste del tamaño.

Posteriormente se observa en el diagrama que existen dos procesos de detección: uno de torax y otro de columna vertebral, estos procesos se llevan a cabo por medio del modelo red convolucional MobileNet 1.0. La primera detección se enfoca en la caja torácica, dicho modelo

fue entrenado con 140 imágenes, de estas 42 imágenes fueron para realizar pruebas y 98 para entrenar. Los resultados del entrenamiento fueron como se muestran en la tabla 11:

Tabla 11. Resultados de la detección de torax utilizando MobileNet 1.0

Modelo	Pérdida	Tiempo de la detección	Número de pasos del entrenamiento
MobileNet 1.0	0.6694	20.258 seg	4573

Los resultados de reentrenamiento de la red neuronal convolucional se pueden observar a continuación en la figura 36.

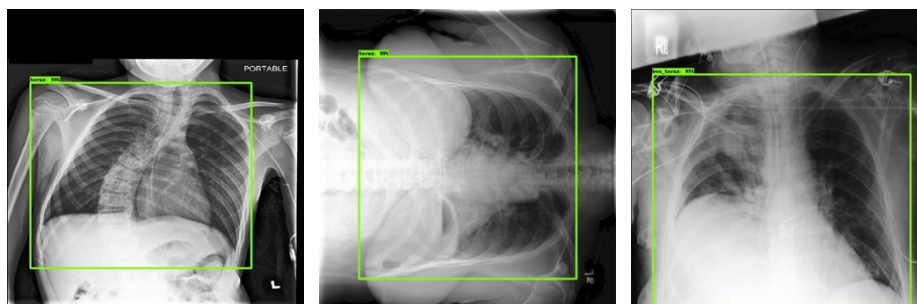


Fig. 36. Resultado de la detección de la caja del tórax utilizando CNN

Con los resultados de la detección se realiza una operación de recorte reajustando las dimensiones de las imágenes lo que equivale a encontrar los parámetros de traslación y cambio de escala de algunos modelos paramétricos de registro basado en características (así el modelo propuesto es parcialmente invariante al tamaño de la imagen o si se encuentra trasladada).

A las imágenes resultantes se les aplicará un nuevo proceso de detección con un modelo de CNN entrenado para la detección de columna vertebral, dicho modelo fue entrenado con 140 imágenes. Los resultados del entrenamiento fueron los que se presentan en la tabla 12.

Tabla 12. Resultados de la detección de columna vertebral utilizando MobileNet 1.0

Modelo	Pérdida	Tiempo de la detección	Número de pasos del entrenamiento
MobileNet 1.0	1.0178	13.568 Seg	33732

Los resultados de reentrenamiento de la red neuronal convolucional se pueden apreciar a continuación en la figura 37.

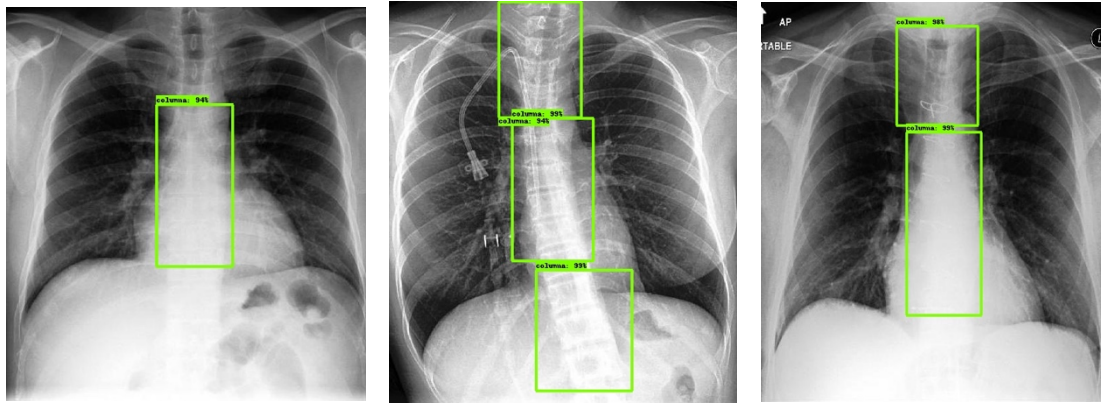


Figura 37. Resultado de la detección de la columna utilizando CNN.

La información que provee la herramienta de detección de columna vertebral son recuadros rectangulares con regiones de la columna vertebral en la imagen evaluada. Un proceso de selección de color permite encontrar la nube de puntos del perímetro de cada rectángulo. Así a partir de las coordenadas de la nube de puntos se realiza el análisis de componentes principales (PCA) para ubicar el eigenvector con el mayor eigenvalor asociado para calcular el ángulo de rotación de la columna en la caja torácica.

El análisis de componentes principales es una de las herramientas clave del análisis estadístico de datos, lo que hace es rotar los ejes sobre los que se definen las dimensiones con la que se describen datos. Los ejes se seleccionan de tal forma que se minimiza la varianza de los residuos (la dispersión de los puntos correspondientes a los datos con respecto a los ejes definidos por los componentes principales). De esa forma, los datos originales se proyectan en sub espacios en los que su varianza se maximiza. Normalmente, quedan sólo con los primeros componentes principales, suficientes para representar la variabilidad que existe en los datos, lo que nos permitirá reducir la dimensionalidad [50].

Sea $\{x_i, y_i\}$, la matriz de coordenadas de la nube de puntos del perímetro de las regiones detectadas con porciones de columna vertebral, para $i = 1, 2, \dots, N$ puntos detectados.

A continuación, presentamos el algoritmo PCA para detectar el eje correspondiente a la columna vertebral. El PCA se calcula de la siguiente manera.

Puntos medios:

$$\begin{bmatrix} m_x \\ m_y \end{bmatrix} = \sum_{i=1}^N \left(\begin{bmatrix} x_i \\ y_i \end{bmatrix} \right) / N$$

Matriz de covarianza:

$$C = \frac{1}{N} \sum_{i=0}^N \begin{bmatrix} (x_i - m_x)(x_i - m_x) & (x_i - m_x)(y_i - m_y) \\ (x_i - m_x)(y_i - m_y) & (y_i - m_x)(y_i - m_y) \end{bmatrix}$$

Descomposición de valores propios:

$$C \begin{bmatrix} dx_i \\ dy_i \end{bmatrix} = \lambda_i \begin{bmatrix} dx_i \\ dy_i \end{bmatrix}_{i=1,2}.$$

λ_i es un valor propio y $[dx_i \ dy_i]^T$ es el correspondiente vector propio. El valor propio mide la riqueza de información en la dirección ortogonal. Suponiendo que $\lambda_1 \geq \lambda_2$ entonces $[dx_i \ dy_i]^T$ es la dirección principal.

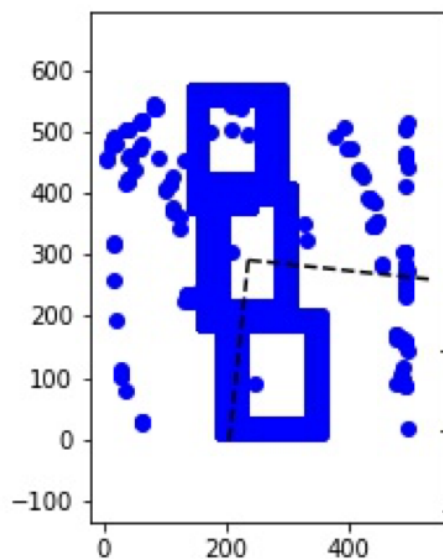


Fig. 38. Los recuadros formados y puntos son el conjunto de pixeles utilizados para calcular los eigenvectores, en líneas punteadas negras los eigenvectores obtenidos.

Una vez obtenida la dirección principal, la imagen se rota de forma que el principal eigenvector coincida con el eje cartesiano y , obteniendo como resultado la alineación de la columna vertebral después de procesar toda la base de imágenes.

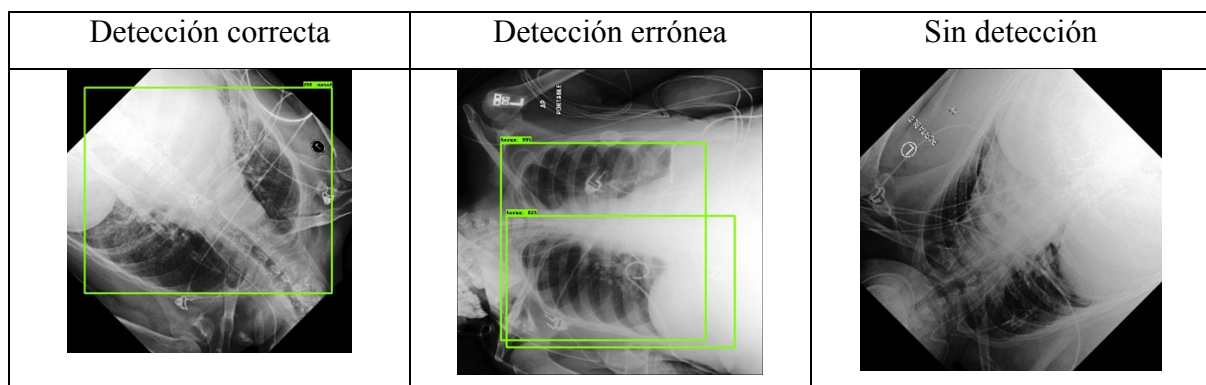
3.3.5 Evaluación de las CNNs enfocadas a la detección

La evaluación de las redes neuronales convolucionales enfocadas a la detección se realiza de dos maneras, la primera centrada en la detección evaluando 20 imágenes, 10 con neumonía y 10 sin neumonía, estas imágenes fueron rotadas en 45°, 90°, 135°, 180°, 225°, 270°, 315° grados. El total de imágenes producidas fue de 160, a continuación, en la tabla 13 se muestran los resultados de las redes neuronales enfocadas a la detección de la caja del tórax.

Tabla 13. Resultados de la detección de tórax en imágenes rotadas utilizando MobileNet 1.0

Tipo de imagen	Detección correcta	Detección errónea	Sin detección
Con neumonía	72	8	0
Sin neumonía	73	3	4
Total de imágenes	145	11	4

Se muestran imágenes de ejemplo, que ilustran los resultados de la tabla anterior:

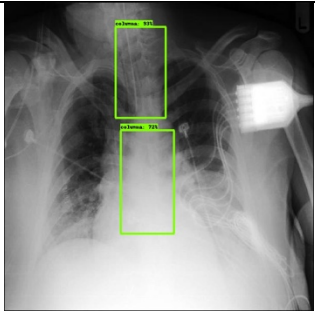



Se realizó la misma prueba para detectar la columna vertebral con las mismas imágenes de la prueba anterior los resultados son los que se muestran en la tabla 14.

Tabla 14. Resultados de la detección de columna vertebral en imágenes rotadas utilizando MobileNet 1.0

Tipo de imagen	Detección correcta	Detección errónea	Sin detección
Con neumonía	10	0	70
Sin neumonía	10	0	70
Total de imágenes	20	0	140

Se muestran imágenes de ejemplo, que ilustran los resultados de la tabla anterior:

Detección correcta	Detección errónea	Sin detección
	<p>Sin imágenes erróneas</p>	

Como se puede ver en esta tabla no existen imágenes erróneas. En esta evaluación se observa que la primera red neuronal orientada a la detección de caja del tórax es invariante a la rotación de las imágenes, por esta razón los resultados son de 145 imágenes con detección correcta de 160. En cambio, la red neuronal para detectar la columna vertebral solo consigue detectar 20 imágenes de 160, lo que significa que la red neuronal convolucional tiene problemas para detectar la columna vertebral, esto se debe a que cuentan con nubosidad las imágenes.

La segunda evaluación, se realiza con un total de 1000 imágenes; 500 imágenes que presentan neumonía y 500 imágenes sin neumonía. Se muestra en la tabla 15 los resultados de la evaluación de los modelos de CNNs re entrenados para la detección de la caja del tórax y la columna vertebral con dichas imágenes.

Tabla 15. Resultados de la detección de tórax y columna vertebral en 1000 imágenes utilizando MobileNet 1.0

Imágenes	Detección correcta para la caja del tórax.	Detección correcta para la columna vertebral.
Con neumonía (500 imágenes)	100%	75.2% (376 imágenes)
Sin neumonía (500 imágenes)	100%	88.8%(444 imágenes)
Porcentaje correcto general de las 1000 imágenes	100%	82% (820 imágenes)

Como se puede ver en la tabla anterior la red neuronal no marca ningún error para detectar la caja del tórax, en cambio la red neuronal para detectar la columna vertebral muestra un 82% por ciento de acierto.

3.3.6 Evaluación del algoritmo de registro

La evaluación del método propuesto, se realiza con un total de 1000 imágenes; 500 imágenes de radiografías de tórax que presentan neumonía y 500 imágenes sin neumonía. Al evaluar los modelos de CNNs MobileNet 1.0 entrenados para la detección se observa lo siguiente:

En las imágenes sin neumonía los detectores de tórax y columna tienen un índice de detección en el 100% de las imágenes. Sin embargo, solo en las imágenes con neumonía las redes neuronales no realizaron la detección de la columna vertebral en varias imágenes, a continuación, se muestran los errores en la tabla 16.

Tabla 16. Resultados de la detección de tórax y columna vertebral en 1000 imágenes utilizando MobileNet 1.0

Imágenes	Sin detectar caja del tórax	Sin detectar columna vertebral.
Con neumonía (500 imágenes)	0%	24.8% (124 imágenes)
Sin neumonía (500 imágenes)	0%	11.2%(56 imágenes)
Total del error:	0%	18 % (180 imágenes)

Como se puede ver en la tabla anterior la red neuronal no marca ningún error para detectar la caja del tórax, en cambio para detectar la columna vertebral muestra un dieciocho por ciento de error cuando los pacientes tienen neumonía.

Con estas 1000 imágenes procesadas utilizando la propuesta de algoritmo para registrar radiografías de tórax se realizan tres evaluaciones: las dos primeras centradas en el entrenamiento de diferentes modelos pre entrenados que permiten la clasificación de imágenes, de esta manera medimos la mejora en el rendimiento de las CNNs, y la tercera evaluación analiza el resultado de la suma promedio de las imágenes sin y con proceso de registro. Las primeras dos evaluaciones comparan los resultados de entrenar modelos de CNNs, con

imágenes procesadas y no procesadas, mostrando la existencia de mejoras en el caso de que existieran. La tercera evaluación analiza las imágenes resultantes por medio de una suma de imágenes, de esta manera se comparan los histogramas de las imágenes sin procesar y procesadas.

Primera evaluación

La primera evaluación se centra en re-entrenar el modelo Inception v3 con el objetivo de clasificar imágenes con y sin presencia de neumonía. Este modelo se re-entrenó de tres maneras diferentes, la primera fue con imágenes que no son procesadas por nuestro método de registro, la segunda fue con las imágenes resultantes de nuestro método sin reajustar los tamaños (los cuales varían dependiendo el tamaño de la caja torácica) y el tercero el cual reajusta el tamaño de todas las imágenes resultantes por nuestro método a un tamaño de 320 x 400 pixeles. En la tabla 17 se muestran los resultados de dicha evaluación.

Tabla 17. Resultados de la clasificación de neumonía utilizando Inception v3

Imágenes	Número de pasos de entrenamiento	Entropía cruzada	Exactitud de la validación	Exactitud final de las pruebas
Sin procesar	4000	0.156805	83.0% (N=100)	74.6% (N=114)
Procesadas	4000	0.212796	78.0% (N=100)	77.0% (N=87)
Procesadas redimensionadas	4000	0.228533	76.0% (N=100)	73.1% (N=104)

Al re-entrenar el modelo Inception v3 los datos resultantes del entrenamiento son las columnas, la exactitud final de las pruebas es nuestra métrica primordial. Se puede observar en esta prueba una mejora marginal de 3% en las imágenes procesadas sin realizar un reajuste del tamaño.

Segunda evaluación

La segunda evaluación se centra en re-entrenar el modelo MobileNet v1 con el objetivo de clasificar imágenes con y sin presencia de neumonía, similar a la evaluación previa con la diferencia del modelo a re-entrenar. Este modelo se re-entrenó de tres maneras diferentes, la primera fue con imágenes las cuales no son procesadas por nuestro método de registro, la segunda fue con las imágenes resultantes de nuestro método sin reajustar los tamaños (los cuales varían dependiendo el tamaño de la caja torácica) y el tercero el cual reajusta el tamaño de todas las imágenes resultantes por nuestro método a un tamaño de 320 x 400 píxeles. En la tabla 18 se muestran los resultados de dicha evaluación.

Tabla 18. Resultados de la clasificación de neumonía utilizando MobileNet v1

Imágenes	Número de pasos de entrenamiento	Entropía cruzada	Exactitud de la validación	Exactitud final de las pruebas
Sin procesar	10000	0.026431	86.0% (N=100)	74.6% (N=114)
Procesadas	10000	0.105245	64.0% (N=100)	72.4% (N=87)
Procesadas redimensionadas	10000	0.238091	78.0% (N=100)	75.0% (N=104)

Al re-entrenar el modelo MobileNet 1.0 los datos resultantes del entrenamiento en las imágenes de prueba se muestran en la última columna, la exactitud final en las imágenes de prueba es la métrica más importante de las redes neuronales. Se puede observar en esta prueba también una ligera mejora de 0.4% con las imágenes procesadas redimensionadas.

Tercera evaluación

Esta evaluación analiza el resultado de la suma promedio de 500 imágenes que no fueron alineadas con nuestro método y 500 imágenes que, si fueron registradas, las imágenes que fueron evaluadas contaban con neumonía. Como se puede observar en la figura 39 del lado izquierdo se muestra la suma promedio de las imágenes originales sin el proceso de registro.

Del lado derecho se muestra que al registrar las imágenes el tórax ocupa mayor área en la imagen promedio eliminando la información innecesaria lo que permite mejorar la detección además las áreas oscuras que corresponden a los pulmones son más evidentes.

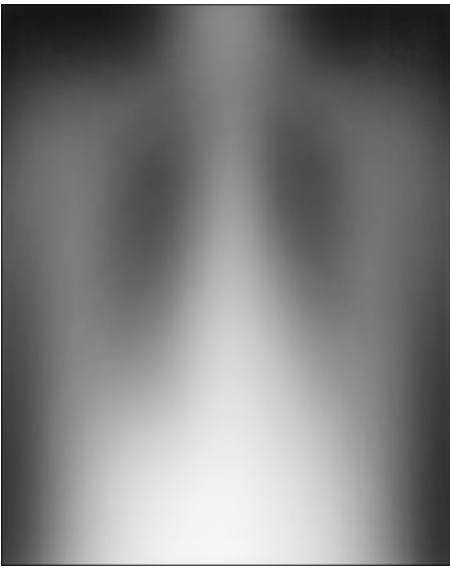

	
<p>Imagen promedio de 500 muestras sin procesar</p>	<p>Imagen promedio del tórax registrando las muestras por el método propuesto utilizando CNNs</p>

Fig. 39. Imágenes promedio de 500 imágenes con neumonía, a la izquierda sin registro, a la derecha con registro.

En el siguiente histograma se muestra la frecuencia en pixeles con las que aparecen los distintos niveles de intensidad en las imágenes anteriores, el nivel de intensidad está en el rango de 0 a 255, en donde el valor 0 representa el color negro y 255 el color blanco, observando el histograma se aprecia como la intensidad de iluminación se encuentra proporcionada de una manera más equitativa en la imagen promedio de las imágenes procesadas por el algoritmo de registrado .

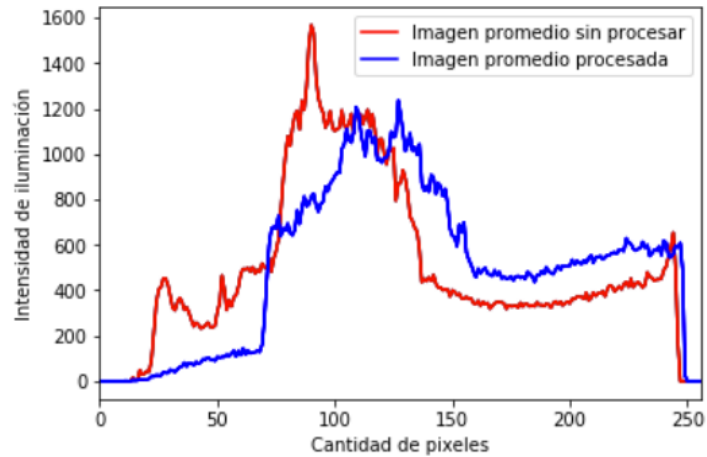


Fig. 40. Histograma comparativo entre las imágenes promedio con neumonía.

Una prueba similar se realiza con 500 imágenes, pero ahora sin neumonía, nótese que en la imagen promedio obtenida registrando las imágenes también mejora la definición de los rasgos, por lo que se ve menos borrosa en la sección de los pulmones.

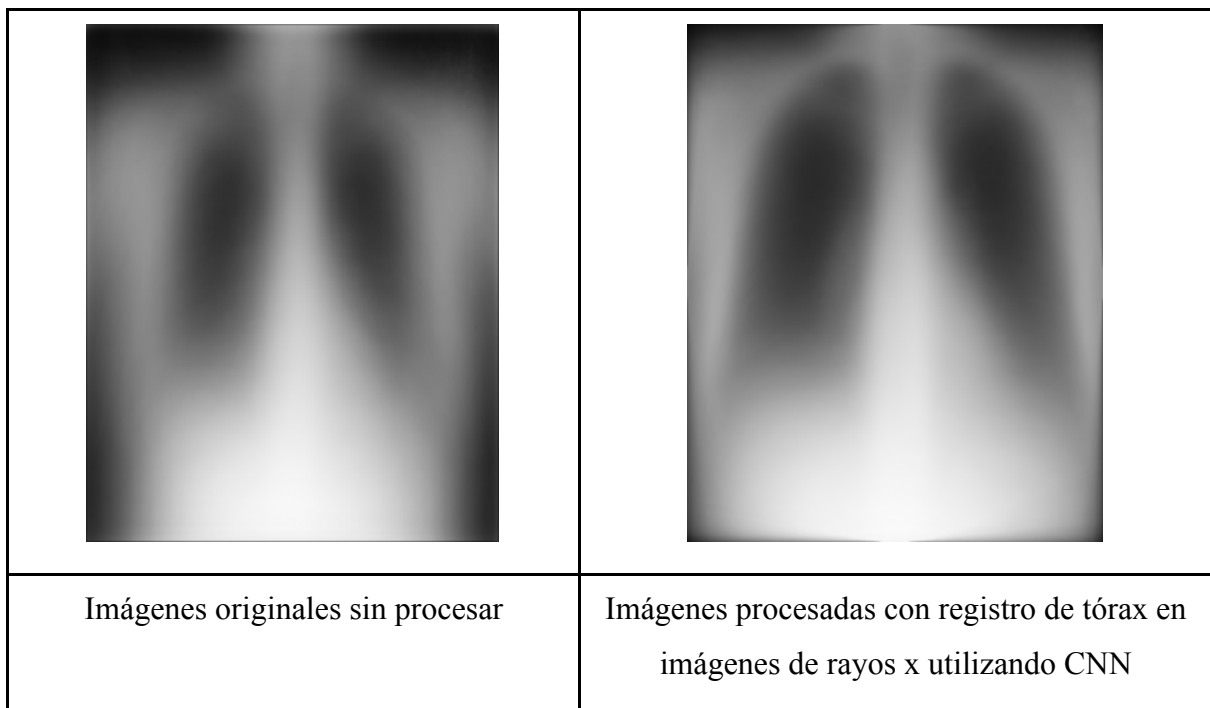


Fig. 41. Imágenes promedio de 500 imágenes sin neumonía, a la izquierda sin registro, a la derecha con registro.

En el siguiente histograma se muestra la frecuencia en píxeles con las que aparecen los distintos niveles de intensidad en las imágenes anteriores, el nivel de intensidad está en el rango de 0 a 255, en donde el valor 0 representa el color negro y 255 el color blanco, observando el

histograma se aprecia como la intensidad de iluminación se encuentra proporcionada de una manera más equitativa en las imágenes promedio procesada.

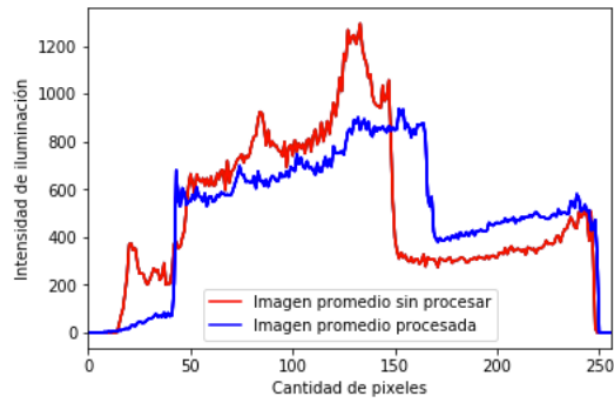


Fig. 42. Histograma comparativo entre las imágenes promedio sin neumonía.

En esta evaluación que se realiza calculando la suma promedio de 500 imágenes, muestra en el primer caso (imágenes con neumonía), que existe nubosidad en el área de los pulmones. En el segundo caso (imágenes sin neumonía), muestra que la nubosidad disminuye y se aprecian mejor los detalles en la imagen promedio. En ambos casos el método de registro de imágenes de rayos x de tórax utilizando CNN delimita las imágenes para aprovechar más la información de las imágenes.

3.3.7 Conclusiones

El registro de imágenes es el proceso que busca la transformación geométrica para alinear dos imágenes en la mejor correspondencia posible, esta sección se enfoca en una propuesta de registro utilizando CNNs y análisis de componentes principales. Los resultados se pueden apreciar visualmente en el apartado anterior. El método propuesto elimina información innecesaria y delimita mejor la región del tórax. En cuanto a la evaluación respecto a la clasificación de tipos de neumonía en imágenes de rayos X aunque la mejora es marginal pre procesando las imágenes con el método de registro los resultados sugieren que es necesario mejorar el preprocesamiento para revelar los detalles fisiológicos de los patrones asociados a cada tipo de neumonía.

Como trabajo futuro proponemos utilizar el PCA para obtener una descomposición de las imágenes de rayos X para formar imágenes de pseudo color previo a la capa de entrada de la CNN para detectar la columna vertebral.

Conclusión

En esta tesis se presentan tres propuestas para resolver problemas de clasificación de imágenes médicas utilizando redes neuronales convolucionales, se explica cómo funcionan, y con dicha herramienta se realizaron evaluaciones en bases de imágenes médicas conocidas. En la actualidad la inteligencia artificial y el aprendizaje supervisado está logrando automatizar procesos de decisión complejos, en el área de la salud el uso de estas herramientas sugiere la disminución de costos y la rapidez en los diagnósticos.

En esta tesis se trabajó con tres tipos de imágenes médicas: las primeras fueron tomografías del cerebro, donde fueron evaluadas imágenes con y sin presencia de tumores cerebrales, los resultados muestran que un modelo de red neuronal convolucional permite clasificar imágenes con presencia de tejido tumoral. El segundo tipo de imágenes con las que se trabajó fueron imágenes de resonancia magnética del tórax, en este caso el objetivo consistía en clasificar automáticamente cinco tipos de tuberculosis, se realizaron diferentes evaluaciones, la conclusión que se alcanza con este tipo de imágenes es que las redes neuronales convolucionales se comportan de manera más efectiva cuando es un clasificador de dos tipos. Por último, el tercer tipo de imágenes con las que se trabajó fueron imágenes de rayos x del tórax con y sin presencia de neumonía, con estas imágenes se realizaron varias pruebas, y se hizo una propuesta de un método de registro utilizando redes neuronales convolucionales especializadas en la detección de tórax, columna vertebral y el análisis de componentes principales. Así, después de registrar la base de imágenes se obtiene una mejora marginal al momento de clasificar las imágenes de rayos x de tórax con/sin neumonía.

Como aportaciones que se realizaron al elaborar esta tesis; el trabajo reportado en la sección 3.1 fue publicado en el IX Congreso Nacional de Tecnología Aplicada a Ciencias de la Salud que organizo el INAOE en el 2018. La sección 3.2 fue aceptada en 6th International Symposium on Language & Knowledge Engineering organizada por la BUAP.

Trabajos futuros

Con esta tesis se muestra el potencial que tienen las redes neuronales convolucionales para analizar imágenes médicas, así como el uso de las nuevas tecnologías que se encuentran disponibles para implementar soluciones. La implementación de estas herramientas en celulares, o dispositivos aptos para procesar imágenes (Raspberry con Movidius o Jetson Nano de NVIDIA) sugieren la creación de sistemas distribuidos los cuales pueden procesar imágenes médicas en los hospitales, para posteriormente almacenar la información en los expedientes clínicos electrónicos de los pacientes.

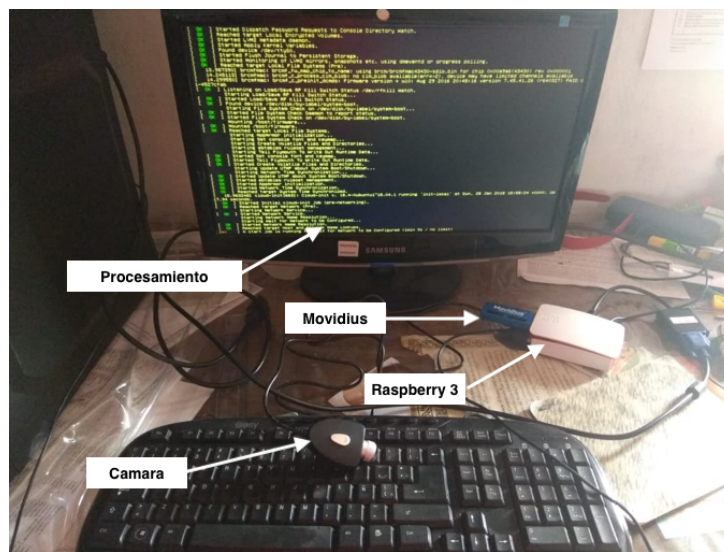


Fig. 43. Equipo requerido para poder implementar un sistema distribuido.

Como trabajo futuro proponemos la creación de esta propuesta, utilizando las herramientas aplicadas en esta tesis. También se propone utilizar el PCA para obtener una descomposición de las imágenes de rayos X para formar imágenes de pseudo color previo a la capa de entrada de la CNN para detectar la columna vertebral en la sección 3.3 que requiere una descomposición en las imágenes.

Bibliografía

1. Anke Meyer Bise. (2004). *Pattern Recognition for Medical Imaging*. United States of America: Elsevier Academic Press.
2. Aygun N, Zinreich SJ. Overview of diagnostic imaging of the head and neck. In: Flint PW, Haughey BH, Lund V, et al, eds. *Cummings Otolaryngology*. 6th ed. Philadelphia, PA: Elsevier Saunders; 2015:chap 8.
3. Laura Álvarez González, Diana María Elena Aldana, María Carmona Rosa . (2012). *PRINCIPIOS DE RESONANCIA MAGNÉTICA*. España: librossanitarios.
4. Institutos Nacionales de Salud. (2018). Rayos X. 26 de junio de 2018, de Health Illustrated Encyclopedia Sitio web: <https://medlineplus.gov/spanish/xrays.html>
5. RadiologyInfo. (2018). Rayos X del tórax (radiografía de tórax). 26 de junio de 2018, de RadiologyInfo.org Sitio web: <https://www.radiologyinfo.org/sp/pdf/chestrad.pdf>
6. Cristian Andrés Ordoñez Santiag. (2005). *FORMATOS DE IMAGEN DIGITA*. © Coordinación de Publicaciones Digitales. DGSCA-UNAMSe autoriza la reproducción total o parcial de este artículo, siempre y cuando se cite la fuente completa y su dirección electrónica. *Revista Digital Universitaria*, Volumen 5 Número 7 • ISSN: 1067-6079, 10.
7. U.S. National Library of Medicine. (2017). Tumores cerebrales . 24 de junio 2018, de Medline plus Sitio web: https://www.uam.es/personal_pdi/medicina/algvilla/fundamentos/nervioso/central.htm
8. Cid, Y.D., Kalinovsky, A., Liauchuk, V., Kovalev, V., Müller, H.: Overview of the imageclef 2017 tuberculosis task - predicting tuberculosis type and drug resistances. In: *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum*, Dublin, Ireland, September 11-14, 2017. (2017), <http://ceur-ws.org/Vol-1866/invitedpaper1.pdf>
9. Kelly B. The Chest Radiograph. *Ulster Med J* 2012;81(3):143-148
10. Briones Michel P, Lebrija Murillo O, Uribe Rioverde C. (2005). *PROCESAMIENTO DE IMÁGENES*. 24 de junio 2018, de Bibliotecadigital Sitio web: http://bibliotecadigital.ilce.edu.mx/sites/ciencia/volumen2/ciencia3/084/htm/sec_9.htm
11. Justin Johnson. (2017). Optimization. 30/08/2018, de stanford.edu Sitio web: <http://cs231n.github.io/optimization-1/>
12. Justin Johnson. (2017). Neural Networks. 30/10/2018, de stanford.edu Sitio web: <http://cs231n.github.io/neural-networks-1/>
13. Raul E. Lopez Briega. (2017). Redes neuronales convolucionales con TensorFlow. 15/11/2018, de relopezbriega Sitio web: <https://relopezbriega.github.io/blog/2016/08/02/redes-neuronales-convolucionales-con-tensorflow>
14. A. Krizhevsky, I. Sutskever, and G. E. Hinton. (2017).). ImageNet classification with deep convolutional neural networks. 10/02/2019, de . NIPS Conference Proceedings Sitio web: <https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>
15. Christian Szegedy and Wei Liu and Yangqing Jia and Pierre Sermanet and Scott Reed and Dragomir Anguelov and Dumitru Erhan and Vincent Vanhoucke and Andrew Rabinovich. (2015). *Computer Vision and Pattern Recognition (CVPR)*. 10/02/2019, de Google Sitio web: <http://arxiv.org/abs/1409.4842>

16. Simonyan, K. and Zisserman A., "Very Deep Convolutional Networks for Large-Scale Image Recognition", "International Conference on Learning Representations", "2015"
17. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. CoRR, abs/1704.04861.
18. "TensorFlow-Slim image classification model library" N. Silberman and S. Guadarrama, 2016 <https://codelabs.developers.google.com/codelabs/tensorflow-for-poets/#3>
19. S. Botello and J. L. Marroquín. (2003). EL METODO DE LOS ELEMENTOS FINITOS EN EL REGISTRO DE IMAGENES DE RESONANCIA MAGNETICA. Guanajuato, Gto. 36000, México: Comunicación del CIMAT.
20. U.S. National Library of Medicine. (2017). Tumores cerebrales . 24 de junio 2018, de Medline plus Sitio web: https://www.uam.es/personal_pdi/medicina/algvilla/fundamentos/nervioso/central.htm
21. Rivaz, H., Chen, S, Collins, DL., Automatic Deformable MR-Ultrasound Registration for Image-Guided Neurosurgery, IEEE Trans. Medical Imaging, 2015, 34(2): 366–380. Sitio web: http://nist.mni.mcgill.ca/?page_id=672
22. Heba Mohsen et al, "Classification using Deep Learning Neural Networks for Brain Tumors", Future Computing and Informatics, pp 1-4 (2017).
23. Stefan Bauer et al, "Multiscale Modeling for Image Analysis of Brain Tumor Studies", IEEE Transactions on Biomedical Engineering, **59**(1): (2012).
24. Bjoern H. Menze et al, "The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)", IEEE Transactions on Medical Imaging, (2014).
25. Jin Liu et al, "A Survey of MRI-Based Brain Tumor Segmentation Methods", TSINGHUA Science and Technology, **19**(6) (2011).
26. J. Pouchot, A. Grasland, C. Collet, J. Coste, J. M. Esdaile, and P. Vinceneux, "Reliability of tuberculin skin test measurement," Annals of internal medicine, vol. 126, no. 3, pp. 210–214, 1997.
27. B. van Ginneken, C. M. Schaefer-Prokop, and M. Prokop, "Computer-aided diagnosis: how to move from the laboratory to the clinic," Radiology, vol. 261, no. 3, pp. 719–732, 2011
28. K. Le, "Automated detection of early lung cancer and tuberculosis based on x-ray image analysis," in Proc. WSEAS International Conference on Signal, Speech and Image Processing, 2006, pp. 1–6.
29. J. M. Leibstein and A. L. Nel, "Detecting tuberculosis in chest radiographs using image processing techniques," University of Johannesburg, 2006.
30. R. N. Rohmah, A. Susanto, and I. Soesanti, "Lung tuberculosis identification based on statistical feature of thoracic x-ray," in QiR (Quality in Research), 2013 International Conference on. IEEE, 2013, pp. 19–26.
31. N. Ratnasari, A. Susanto, I. Soesanti et al., "Thoracic x-ray features extraction using thresholding-based roi template and pca-based features selection for lung tb classification purposes," in Instrumentation, Communications, Information Technology, and Biomedical Engineering (ICICI-BME), 2013 3rd International Conference on. IEEE, 2013, pp.65–69.
32. Laboratorio Nacional de Supercómputo del Sureste de México . (2019). Infraestructura. 01/09/2018, de Benemérita Universidad Autónoma de Puebla Sitio web: <http://lns.org.mx/?q=infraestructura%20>

33. Yashin Dicente Cid, Alexander Kalinovsky, Vitali Liauchuk, Vassili Kovalev, Henning Müller, Overview of ImageCLEFtuberculosis 2017 - Predicting Tuberculosis Type and Drug Resistances, CLEF working notes, CEUR, 2017.
34. T. S. Cohen and M. Welling. Transformation properties of learned visual representations. ICLR, 2015.
35. K. Lenc and A. Vedaldi. Understanding image representations by measuring their equivariance and equiv-alence. CVPR, 2015.
36. Christian Szegedy and Wei Liu and Yangqing Jia and Pierre Sermanet and Scott Reed and Dragomir Anguelov and Dumitru Erhan and Vincent Vanhoucke and Andrew Rabinovich. (2015). Computer Vision and Pattern Recognition (CVPR)}. 10/02/2019, de Google Sitio web: <http://arxiv.org/abs/1409.4842>.
37. Radiological Society of North America. (2018). RSNA Pneumonia Detection Challenge. 02/02/2019, de Kaggle Web site: <https://www.kaggle.com/c/rsna-pneumonia-detection-challenge/evaluation>
38. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.
39. M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. NIPS DLW, 2014
40. F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. arXiv:1503.03832, 2015
41. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. CVPR, 2015.
42. K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. ICLR, 2015.
43. J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015.
44. G. Gkioxari, R. Girshick, and J. Malik. Contextual action recognition with r* cnn. arXiv:1505.01197, 2015.
45. Stanford ML Group. (2017). CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. 1/02/2019, de Stanford Web site: <https://stanfordmlgroup.github.io/projects/chexnet/>
46. Jaderberg, Max, Karen Simonyan, and Andrew Zisserman. "Spatial transformer networks." In Advances in Neural Information Processing Systems, pp. 2008-2016. 2015.
47. M. Engin, H. Oğul, M. Ağildere and E. Sümer, "An evaluation of image registration methods for chest radiographs," 2015 SAI Intelligent Systems Conference (IntelliSys), London, 2015, pp. 822-827.
48. KC S, Candemir S, Jaeger S, Folio L, Karargyris A, Antani SK, Thoma GR. Automatically detecting rotation in chest radiographs using principal rib-orientation measure for quality control. Int. J. Patt. Recogn. Artif. Intell, Vol 29 No. 2.
49. S. Guo, X. Wu, and Z. Luo, "Automatic Extraction of Control Points for Chest X-ray Images and Elastic Registration," In 2nd International Conference on Bioinformatics and Biomedical Engineering (ICBEE 2008), Shanghai – China, pp. 2651-2654, 16-18 May, 2008.
50. FERNANDO BERZAL. (2018). REDES NEURONALES & DEEP LEARNING. Granada España : Editorial Universidad de Granada.
51. Keras team. (2019). Keras backends. 18/07/2019, de MkDocs Sitio web: <https://keras.io/backend/>

Anexos

1.1 Carta de aceptación por el laboratorio nacional de Supercómputo del sureste de México



Dr. Rafael Lemuz López
Buap
Presente

No. de Proyecto: 201803030C

El Laboratorio Nacional de Supercómputo del Sureste de México (LNS), de acuerdo a la convocatoria publicada el pasado 05 de Diciembre de 2017 hace de su conocimiento que el proyecto:

"Reconocimiento de patrones en imágenes médicas por medio de una red neuronal convolucional."

ha sido evaluado por parte del Comité Técnico Académico y Administrativo (CTAA) del LNS conforme a la información proporcionada, por lo que el resultado del dictamen es:

PROYECTO ACEPTADO

Para la asignación de los recursos de supercómputo correspondientes al desarrollo de este proyecto, es indispensable que lea, acepte y cumpla los términos y condiciones establecidos en el Convenio de Colaboración y Políticas de uso del LNS. Dichos recursos estarán sujetos a disponibilidad y cumplimiento de buenas prácticas de uso establecidas por este Laboratorio.

Para cualquier duda o aclaración le pedimos atentamente, nos contacte a través del correo electrónico lns@correo.buap.mx.

Reciba mi más cordial saludo,
ATENTAMENTE,
H. PUEBLA DE Z., 17 DE AGOSTO DE 2018.

\$2y\$10\$kv9j3QkX431XIE1yozv0E.HJKWkwp/NU7WOXiQwRHLJmrKFYd7DEa
Dr. Manuel Martín Ortiz
Director de Cómputo Académico e Investigación
Laboratorio Nacional de Supercómputo del Sureste de México



Calle de la Tecnología
Entre Calle de la Minerva y Calle Bibliotecas
Ciudad Universitaria, Puebla, Pue. C.P. 72592
01 (222) 229 55 00 Ext. 5119
lns@correo.buap.mx
www.lns.org.mx o www.lns.buap.mx

1.2 El código creado y utilizado para la tesis

En el siguiente link se puede observar el código utilizado para la realización de esta tesis.

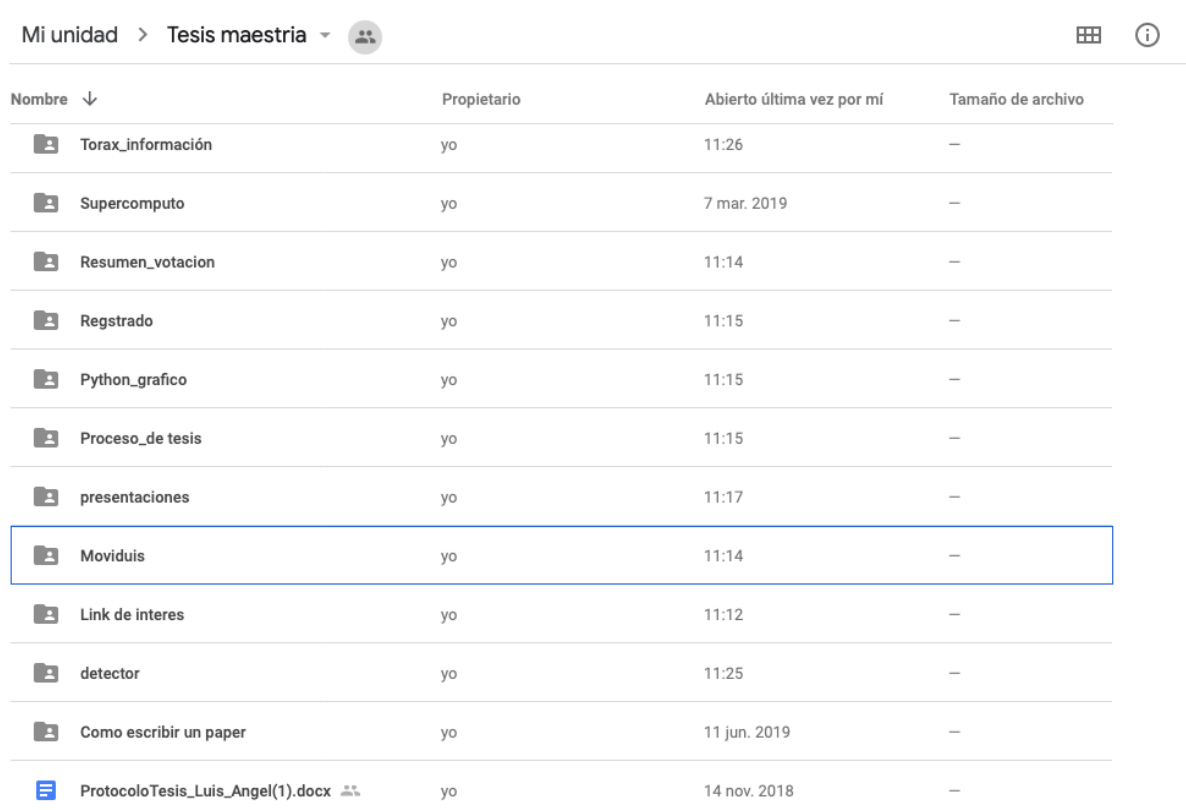
<https://github.com/lasc21>

1.3 Información sobre el uso de redes neuronales convolucionales

En el siguiente link se pueden descargar manuales de uso para implementar CNN, así como los artículos relevantes.

<https://drive.google.com/drive/folders/1ROfkYminNZ9HhCuMcPUS6a4Judx81i-e?usp=sharing>

Estructura de la información del link anterior.



Mi unidad > Tesis maestria

Nombre ↓	Propietario	Abierto última vez por mí	Tamaño de archivo
Torax_información	yo	11:26	—
Supercomputo	yo	7 mar. 2019	—
Resumen_votacion	yo	11:14	—
Registrado	yo	11:15	—
Python_grafico	yo	11:15	—
Proceso_de tesis	yo	11:15	—
presentaciones	yo	11:17	—
Moviduis	yo	11:14	—
Link de interes	yo	11:12	—
detector	yo	11:25	—
Como escribir un paper	yo	11 jun. 2019	—
ProtocoloTesis_Luis_Angel(1).docx	yo	14 nov. 2018	—