UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN
INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y EN SISTEMAS

ESTIMACIÓN DEL TENSOR DE DIFUSIÓN EN IMÁGENES CEREBRALES UTILIZANDO TRANSFORMERS

T E S I S

QUE PARA OPTAR POR EL GRADO DE:

MAESTRO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

PRESENTA:
ING. DANIEL BANDALA ÁLVAREZ

TUTOR PRINCIPAL:
DR. JORGE LUIS PÉREZ GONZÁLEZ
IIMAS

Ciudad Universitaria, CDMX. Diciembre de 2023

Tesis presentada en diciembre de 2023 ante los sinodales:

Dr. Alfonso Gastélum Strozzi
Dr. Jorge Luis Pérez González
Dra. Nidiyare Hevia Montiel
Dra. Jimena Olveres Montiel
Dr. Paul Erick Méndez Monroy

Director de Tesis:

Dr. Jorge Luis Pérez González

«*God used beautiful mathematics in creating the world.*»

Paul Dirac

Para mi madre, mis seres queridos y amigos, todos y cada uno me han inspirado y han dejado una huella en mí de alguna manera.

Agradecimientos

En primer lugar, no hay palabras suficientes para agradecer a mi madre, Georgina Álvarez Pastrana, quien ha sido mi fuente de inspiración y apoyo incondicional en cada paso que he dado. Su amor, paciencia y sabiduría han sido mi guía y es gracias a ella que hoy puedo celebrar este logro. Mamá, tu fortaleza y dedicación no tienen comparación, estoy eternamente agradecido por todo lo que has hecho por mí.

A mis profesores, les estoy profundamente agradecido por su guía y dedicación. Ustedes han sido más que mentores para mí; han sido amigos que han creído en mí y han invertido su tiempo y esfuerzo para ayudarme a alcanzar mi máximo potencial. Su influencia ha sido clave en mi formación y desarrollo y les estoy eternamente agradecido.

A mis amigos, gracias por estar siempre en las buenas y en las malas. Su apoyo y amistad han sido fundamentales para superar los desafíos y celebrar los éxitos. Ustedes han añadido alegría y color a mi vida y estoy agradecido por cada momento compartido.

Quiero expresar mi agradecimiento al CONAHCYT, por brindarme las oportunidades y recursos necesarios para avanzar en mi carrera y formación profesional. Me siento honrado de haber sido parte de su programa de apoyos.

Investigación realizada gracias al Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica (PAPIIT) de la UNAM IA104622 e IT101422. Agradezco a la DGAPA-UNAM la beca recibida.

Finalmente, agradezco al Dr. Abelardo Ávila-Curiel, Psic. Marsela Alvarez-Izazaga, Dr. Juan Fernandez Ruíz y Dr. Israel Vaca Palomares por compartir las imágenes para realizar las pruebas adicionales de niños con desnutrición.

Resumen

La presente investigación aborda el problema de la estimación del tensor de difusión a partir de imágenes cerebrales de resonancia magnética. Este tensor de difusión es utilizado para estudiar la microestructura de los tractos en la materia blanca cerebral y representa un elemento esencial para comprender la integridad de las fibras nerviosas y la conectividad cerebral. Por otra parte, los métodos convencionales utilizados se basan en una aproximación del comportamiento real de la difusión de las moléculas de agua en el tejido cerebral, por lo que se requiere una gran cantidad de muestras para realizar el ajuste de este tensor. En este contexto, se propone la integración de módulos basados en autoatención, los denominados *Transformers*, en redes neuronales profundas para procesar imágenes de resonancia magnética ponderadas por difusión y estimar tres mapas derivados del tensor de difusión: la anisotropía fraccional, la difusividad media y las orientaciones de difusión. Esta estimación se basa en seis señales de difusión y una señal sin dirección de difusión, lo que optimiza la eficiencia computacional de la estimación de los mapas y reduce el tiempo necesario para obtener el volumen completo del cerebro de un paciente. Entonces, al utilizar un modelo de redes neuronales profundas, es posible ajustar las estimaciones de los mapas mediante un enfoque de aprendizaje supervisado, por lo que es necesario contar con una base de información correctamente etiquetada. Para ello, se han utilizado las imágenes de 35 sujetos sin patologías del Proyecto de Conectoma Humano e imágenes de 30 pacientes sanos de la Iniciativa de Neuroimágenes para la Enfermedad de Alzheimer. Esta información se ha procesado a través de la herramienta de software FSL para obtener los tres mapas antes mencionados para cada paciente. El modelo implementado es una variante del modelo *UT-Net*, del cual se ha validado su robustez en distintos contextos clínicos. El entrenamiento y los resultados que se han obtenido muestran el potencial de los mecanismos de atención en el procesamiento de imágenes por resonancia magnética, resaltando su eficiencia computacional y escalabilidad, particularmente útiles para aplicaciones en tiempo real y procesamiento de grandes volúmenes de datos. Finalmente, se valida la capacidad de generalización del modelo implementado mediante la inferencia de los tres mapas derivados del tensor de difusión sobre un conjunto de pacientes mexicanos con deterioro cognitivo leve y afectaciones causadas por desnutrición durante la infancia.

Abstract

The present investigation addresses the problem of diffusion tensor estimation from magnetic resonance brain imaging. This diffusion tensor is used to study the microstructure of tracts in the cerebral white matter and represents an essential element to understand the integrity of nerve fibers and brain connectivity. On the other hand, the conventional methods used are based on an approximation of the real behavior of the diffusion of water molecules in brain tissue, thus requiring a large number of samples to perform the fitting of this tensor. In this context, the integration of self-attention based modules, the so-called *Transformers*, into deep neural networks is proposed to process diffusion-weighted magnetic resonance images and estimate three maps derived from the diffusion tensor: the fractional anisotropy, the mean diffusivity and the diffusion orientations. This estimation is based on six diffusion signals and a signal with no diffusion direction, which optimizes the computational efficiency of map estimation and reduces the time required to obtain the full volume of a patient's brain. Then, using a deep neural network model, one can adjust map estimates through a supervised learning approach, necessitating a correctly labeled information base. For this purpose, images of 35 subjects without pathologies from the Human Connectome Project and images of 30 healthy patients from the Alzheimer's Disease Neuroimaging Initiative have been used. This information has been processed through the FSL software tool, to obtain the three maps for each patient. The implemented model is a variant of the *UT-Net* model, which has been validated in different clinical contexts. The training and the results obtained show the potential of the attention mechanisms in magnetic resonance image processing, highlighting their computational efficiency and scalability, particularly useful for real-time applications and processing of large volumes of data. Finally, the generalizability of the implemented model is validated by inferring the three maps derived from the diffusion tensor on a set of Mexican patients with mild cognitive impairment and affectations caused by malnutrition during childhood.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

El campo de la visión artificial se enfoca en el estudio y diseño de algoritmos que le permiten a las computadores y procesadores digitales interpretar el mundo de una manera similar al como lo hacen los seres humanos a través de la visión: mediante la adquisición, análisis y comprensión de imágenes digitales. Esta tecnología, que combina técnicas de procesamiento de imágenes y señales digitales, óptica, aprendizaje automático, entre otras, tiene aplicaciones que van desde la detección automática de objetos hasta la interpretación de escenas complejas e inferencia de nuevo conocimiento [51, 114]. Dentro de este amplio espectro de aplicaciones, una de las áreas de mayor potencial y rápido crecimiento es el procesamiento de imágenes médicas. Esto último se refiere al uso de técnicas computacionales para mejorar, analizar e interpretar imágenes biomédicas como radiografías, resonancias magnéticas, tomografías computarizadas y ultrasonidos. Estas imágenes son fundamentales para el diagnóstico, seguimiento y tratamiento de una amplia variedad de afecciones médicas. Sin embargo, debido a la complejidad y sutileza de las estructuras anatómicas, los comportamientos químicos en patologías, así como a las variaciones naturales entre individuos, la interpretación manual de estas imágenes puede ser lenta, tediosa y dispuesta a errores [93, 117, 24]. Entonces, el uso de la visión computacional en el ámbito médico promete revolucionar el campo de la medicina diagnóstica y la investigación clínica al automatizar y asistir en tareas como la segmentación de órganos, la detección de tumores, la identificación de patologías o la reconstrucción tridimensional de estructuras anatómicas del cuerpo humano. Con ello, los profesionales de la salud pueden obtener diagnósticos con una mayor precisión en menor tiempo, lo que se traduce en tratamientos con mayor efectividad y un mejor pronóstico para el paciente [127, 143]. Así, más allá de la mera interpretación de imágenes, el procesamiento de imágenes médicas está impulsando la investigación y el desarrollo de nuevas modalidades de imagen, protocolos de adquisición y herramientas de visualización. Por ello, la adquisición y almacenamiento de grandes volúmenes de datos médicos, la computación de alto rendimiento y algoritmos cada vez más sofisticados están llevando a la medicina hacia una era en donde el diagnóstico asistido por computadora será la norma y no la excepción. En concreto, la intersección de la visión computacional con el procesamiento de imágenes médicas representa una de las fusiones más potentes de la tecnología y la

atención médica en el siglo XXI, prometiendo transformaciones profundas en cómo se detectan, diagnostican y se tratan enfermedades.

En particular, la tecnología de Imágenes por Resonancia Magnética (IRM o MRI, por sus siglas en inglés) ha revolucionado la medicina moderna, desde su invención en la década de 1970, ofreciendo imágenes detalladas del interior del cuerpo humano sin la necesidad de radiación ionizante. Estas imágenes, que pueden mostrar tejidos blandos con un nivel de detalle en el orden de micras, han sido esenciales en el diagnóstico y tratamiento de innumerables condiciones médicas. Aun así, con la creciente complejidad de las técnicas de imágenes de IRM y la necesidad de interpretaciones precisas durante el diagnóstico, ha surgido la motivación de explorar alternativas para mejorar la eficiencia y precisión del análisis de estas imágenes. Además, el procesamiento de imágenes de IRM es inherentemente complejo debido a la naturaleza de las mismas leyes físicas que rigen su comportamiento y este tipo de imágenes no solo contienen información estructural, sino también funcional, con variantes como las IRM funcionales (fMRI) que capturan y mapean la actividad cerebral en tiempo real. Esta abundancia de datos puede ser abrumadora para el ojo humano, lo que lleva a posibles errores de interpretación o a la omisión de detalles cruciales dentro de las imágenes. Aquí es donde los algoritmos de inteligencia artificial (IA) y capacidades de aprendizaje automático entran en acción. Al implementar estas técnicas y modelos en el análisis de imágenes de IRM, se abre la puerta a la automatización de tareas como la detección de patrones y patologías [120, 39]. Para ello, los sistemas de IA, como las redes neuronales convolucionales, se entrenan utilizando vastos conjuntos de datos, aprendiendo a reconocer características específicas en las imágenes y a distinguir entre condiciones normales y patológicas. Estos sistemas pueden identificar con rapidez y precisión áreas de interés, reduciendo el tiempo de diagnóstico y aumentando la confiabilidad de los resultados. Así, la combinación de la experiencia humana con la capacidad analítica de la IA puede resultar en un diagnóstico más preciso, permitiendo intervenciones tempranas y tratamientos más efectivos. Entonces, el potencial de la IA en el ámbito de las imágenes de IRM es claro y actualmente esta tecnología se encuentra en las primeras etapas de desarrollo, donde todavía existe un vasto campo de exploración en cuanto a sus capacidades en el contexto clínico. Por tanto, a medida que esta tecnología madure, se espera que se presenten avances aún más significativos, solidificando el papel de la IA como una herramienta indispensable en la medicina de diagnóstico por imagen.

En este trabajo de investigación se propone un modelo de IA basado en redes neuronales profundas para estimar tres mapas derivados del tensor de difusión a partir de imágenes cerebrales de IRM. Este modelo aprovecha las ventajas de los mecanismos de autoatención y operadores convolucionales para extraer las características más significativas que permitan modelar la relación no lineal entre las señales obtenidas por experimentos de resonancia magnética y el tensor de difusión.

1.1. Estado del arte

En esta sección se describen de manera breve las aportaciones y trabajos que han aportado en la comprensión detallada del sistema de visión natural y que ha inspirado uno de los campos en la ciencia e ingeniería donde convergen múltiples disciplinas: la visión computacional. Asimismo, se abordan las tecnologías y aplicaciones que han emergido en las últimas décadas gracias a los avances teóricos en esta área y un incremento considerable en la capacidad de procesamiento. Se presenta una perspectiva general de las técnicas actuales de extracción de imágenes estructurales y funcionales del cerebro por medio de resonancia magnética, se describen las características principales de las imágenes por difusión y los parámetros estándares que se pueden inferir a partir de estas. Y, finalmente, se presentan los trabajos y avances recientes que se han propuesto para el procesamiento y estimación del tensor de difusión a partir de estas imágenes médicas utilizando algoritmos de aprendizaje automático.

1.1.1. Visión computacional

Los primeros estudios del sistema de visión humano y el fenómeno de percepción de profundidad se remonta a los antiguos griegos con Euclides y sus aportaciones en óptica, geometría y la naturaleza de la visión [11]. Para el siglo $XVII$, los trabajos de Alhazen, Kepler y Descartes realizaron avances importantes en la comprensión de la función de la retina y el nervio óptico en el sistema de visión natural. Poco después, Sir Isaac Newton publica su trabajo *Opticks* [64], en donde, de manera acertada, describe la forma en la que la información pasa de los ojos al cerebro.

El sistema de visión humano comienza en los ojos, la luz incidente pasa a través de la pupila que controla la cantidad de luz que pasa a través de la retina del ojo y el tamaño de la pupila es controlado por el esfínter pupilar del iris. Cuanto mayor sea esta apertura, mayor será la apertura esférica y menor la profundidad del enfoque del ojo. De manera análoga, las cámaras intentan simular el mecanismo de visión natural y el modelo más simple de una cámara comprende una apertura circular y un plano de imagen. Esta apertura se encuentra entre el plano y la escena tridimensional observada, de modo que toda la luz emitida o reflejada en la escena se encuentra obligada a pasar a través de la apertura circular antes de llegar al plano de la imagen. Por tanto, existe una correspondencia directa entre cada área de la imagen bidimensional en el plano y el área en la escena tridimensional, como es observado a través de la apertura [35]. En cambio, a diferencia del sistema de visión natural de los seres humanos, es también posible reconstruir imágenes a partir de señales de radiofrecuencia, como es en el caso de la tecnología IRM. En este tipo de imágenes se utiliza un campo electromagnético para alinear y excitar los núcleos de hidrógeno en el cuerpo con radiofrecuencias. Estos núcleos emiten señales de radiofrecuencia de regreso que se adquieren en un espacio en el dominio de frecuencia. Entonces, para transformar estos datos en una imagen visual,

se utiliza la Transformada de Fourier, convirtiendo la información de frecuencia en una representación espacial [157]. Finalmente, se puede procesar esta imagen para mejorar su claridad y utilidad en el diagnóstico y análisis.

Debido a esto, recientemente la visión computacional se ha convertido en una tecnología clave en muchos campos, desde sistemas de asistencia al conductor para automóviles hasta interacciones de usuario en videojuegos. En la automatización industrial, la visión artificial es utilizada frecuentemente para el control de procesos y calidad. Así, un campo que ha encontrado aplicaciones importantes y que ha impulsado toda una nueva área en la ingeniería es el procesamiento de imágenes médicas, ya sea para limpiar el ruido blanco o componentes indeseadas en las señales obtenidas o para extraer e inferir conocimiento nuevo a partir de estas, con lo que se han obtenido exitosos resultados. Ejemplos como la detección temprana de cáncer de mama [145, 138], el diagnóstico exacto de retinopatía diabética [167], la localización de tumores [152], el diagnóstico y comprensión de trastornos neurodegenerativos, entre muchos otros, respaldan la opción de utilizar modelos y algoritmos de aprendizaje computacional para la asistencia e investigación médica.

1.1.2. Imágenes cerebrales ponderadas por difusión

Las neuroimágenes se han convertido en una herramienta estándar en el estudio clínico de pacientes con patologías neurodegenerativas o en el análisis exploratorio prequirúrgico. Asimismo, en el área de la neurociencia se han realizado avances contundentes debido al desarrollo de técnicas de extracción y procesamiento de datos e imágenes estructurales y funcionales del cerebro. Es entonces que, gracias a distintas técnicas de imágenes por resonancia magnética y el desarrollo de nuevos métodos como la tomografía por emisión de positrones, se hace posible una comprensión de la microestructura y conectividad cerebral [193, 161, 20]. Una característica distintiva de la IRM es su capacidad para producir diferentes tipos de imágenes o secuencias, cada una optimizada para visualizar ciertas estructuras o patologías. Estas secuencias se logran variando los parámetros del escáner para la resonancia. Entre las secuencias más comunes se encuentran las imágenes T1 (T1W), estas imágenes son ideales para visualizar la anatomía detallada, especialmente del cerebro, y diferenciar entre sustancia gris y blanca como se muestra en la Figura 1.1; las imágenes T2 (T2W), las cuales suelen mostrar el líquido cefalorraquídeo en blanco, siendo especialmente útiles para detectar edemas o lesiones con alto contenido de agua; las imágenes FLAIR (Fluid Attenuated Inversion Recovery), las cuales son similares a las imágenes T2W, pero en donde se suprime la señal del líquido cefalorraquídeo; y las imágenes ponderadas por difusión (DWI), las cuales detectan la difusión de las moléculas de agua en los tejidos. La elección de la secuencia adecuada es esencial para obtener la información diagnóstica más relevante para cada caso clínico [135]. En particular, las imágenes DWI permiten estudiar la estructura del tejido de la materia blanca debido a su alta sensibilidad al desplazamiento de las moléculas de agua

con una precisión del orden de micras [164]. En un medio isotrópico, estas moléculas de agua se encuentran moviéndose constantemente en direcciones aleatorias gracias a su energía térmica en un proceso denominado autodifusión [110, 112], en donde el movimiento promedio de estas moléculas no tiene una dirección preferencial. Entonces, debido a que las estructuras celulares pueden impedir el movimiento del agua a una escala microscópica, las imágenes DWI actúan como evidencia de la microestructura del tejido interno cerebral. Esta técnica posee el potencial de revelar información, sin ningún método invasivo, sobre la organización y estructura del tejido de materia blanca cerebral a una escala celular.

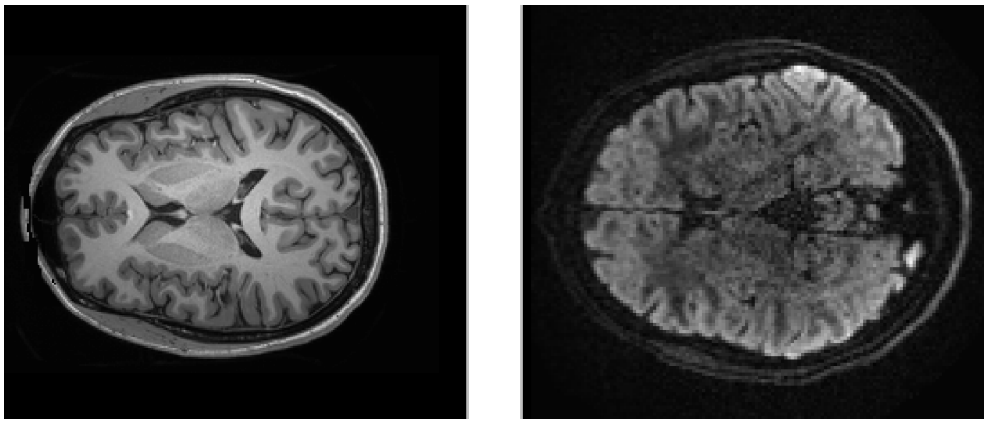

FIGURA 1.1: Protocolos distintos de adquisición de IRM: Imagen T1W (izquierda) e imagen DWI (derecha). Imágenes tomadas del Proyecto del Conectoma Humano [111, 57].

Existen varias metodologías para modelar la difusión en sistemas biológicos, cada una con distintas suposiciones y niveles de complejidad. El modelo más simple es asumir una difusión libre, caracterizada por un único coeficiente de difusión. Sin embargo, debido a que las medidas de la difusividad claramente dependen de los parámetros experimentales, se ha introducido el coeficiente de difusión aparente (ADC) [82, 39, 18, 79]. Por otra parte, la materia blanca cerebral se compone principalmente de axones, que son responsables de transmitir información eléctrica y química a través del cerebro. Estos son largas proyecciones celulares que se extienden desde las células nerviosas, o neuronas, y permiten que las señales eléctricas se propaguen entre terminales nerviosas, donde se comunican con otras neuronas o células del cuerpo. Estos axones cerebrales están organizados en fascículos o haces, que a menudo se denominan tractos y se encuentran recubiertos por una capa de mielina, un material graso que actúa como un aislante eléctrico que acelera la velocidad de transmisión de los impulsos nerviosos. Además, se ha demostrado que la difusión en la materia blanca cerebral es dependiente de la dirección [164, 102], una observación que evidentemente se relaciona con la orientación de las fibras de axones dentro de las regiones de materia blanca con conjuntos de fibras

coherentes. Esta anisotropía direccional, de las señales de difusión, presenta la única posibilidad de inferir las fibras de materia blanca dentro de un cerebro de un paciente vivo y de forma completamente no invasiva. Entonces, el modelo del tensor de difusión se ha introducido como una extensión del modelo de ADC con la capacidad de describir la difusión anisotrópica y del cual se pueden extraer mapas como la difusividad media (MD), la anisotropía fraccional (FA) y las direcciones principales de la difusión (DO) [126, 80, 137, 32]. Gracias a la simplicidad de esta técnica, esta se ha utilizado como un estándar desde hace un par de décadas, en especial en el área clínica. No obstante, presenta varias limitantes como la incapacidad de detectar el cruzamiento en distintas direcciones de grupos de tractos o fibras divergentes, por lo que la interpretación de las medidas derivadas de un tensor es inexacta en regiones donde la orientación de las fibras no es única. Esto ha motivado al desarrollo de modelos de orden superior para representar la microestructura y conectividad neuronal [176, 45].

1.1.3. Tensor de difusión con inteligencia artificial

En general, la difusión de un grupo de moléculas en un ambiente no restringido se mueve libremente en todas las direcciones, por lo que el movimiento neto del conjunto de moléculas no tiene una dirección particular. Sin embargo, en el tejido cerebral, y especialmente en las fibras nerviosas, este movimiento se ve restringido y tiende a seguir la dirección de estas fibras. Las imágenes por tensor de difusión (DTI) aprovechan este comportamiento para obtener señales que reflejan la orientación y la integridad de las fibras nerviosas, ofreciendo así mapas tridimensionales de la conectividad cerebral. La técnica utilizada para extraer esta información se basa en el modelo de Stejskal-Tanner, propuesto en 1965 por Edward O. Stejskal y John E. Tanner [158], y este modelo describe cómo se puede medir la difusión utilizando secuencias de IRM. Esencialmente, se aplican dos gradientes de campo magnético, uno después del otro, y la señal resultante es afectada por el movimiento de las moléculas entre estos dos pulsos. Cambiando la magnitud y la duración de estos gradientes, se puede obtener información sobre el grado y la dirección de la difusión en cada punto de la imagen. Entonces, al recolectar datos de difusión en múltiples direcciones y aplicar el modelo de Stejskal-Tanner, es posible reconstruir el tensor de difusión para cada pixel o voxel de la imagen o volumen, respectivamente. Este tensor describe la forma y la orientación de la difusión en ese punto específico, permitiendo inferir la orientación de las fibras nerviosas y su integridad. Por tanto, a través de este modelo, se puede cuantificar y representar matemáticamente el movimiento de las moléculas de agua en todo el cerebro, proporcionando una ventana a la microestructura del tejido de materia blanca. Gracias a esto, esta técnica ha encontrado aplicaciones en la investigación de enfermedades neurodegenerativas, traumas, tumores y en el mapeo cerebral previo a cirugías [67, 113, 40, 94, 133, 201]. Sin embargo, es complicado encontrar una solución analítica al modelo de Stejskal-Tanner, por lo que se recurre a solucionar el problema de la estimación del tensor de difusión mediante

métodos numéricos. Estos métodos, para su óptimo funcionamiento, requieren una gran cantidad de datos, por lo que es común que en estudios de IRM se generen imágenes en más de 30 direcciones distintas de difusión, lo cual agrega una mayor complejidad en las señales, como el movimiento del paciente y distorsiones generadas por el propio escáner. Además, el modelo de Stejskal-Tanner asume en todo momento una distribución gaussiana del desplazamiento de las partículas que se difunden, por lo que puede llevar a una desviación respecto a la estimación real cuando las partículas no se difunden bajo este comportamiento. En consecuencia, el procesamiento de este tipo de información resulta tedioso y computacionalmente exhaustivo, puesto que se requiere estimar el tensor de difusión y descomponer este tensor en sus vectores y valores propios para cada uno de los voxeles del volumen de un paciente [19, 85]. Esto ha motivado buscar alternativas al problema de la estimación del tensor de difusión y los mapas derivados de este tensor mediante algoritmos de IA. De la misma forma, el seguimiento de los tractos o tractografía, el cual es el estudio del delineamiento de los conjuntos de fibras de materia blanca en el cerebro, es una de las áreas que ha ganado mayor atención debido a su potencial para estudiar la conectividad del cerebro en muestras de pacientes vivos [191, 10, 22, 68]. En general, los algoritmos para estimar una tractografía son simples, aunque sujetos a un número significante de limitaciones, por lo que resulta de interés el desarrollo de nuevos modelos o técnicas que permitan aumentar las capacidades y reducir la complejidad computacional de los métodos convencionales para el ajuste del tensor de difusión en la microestructura de tejidos biológicos [202, 93, 187, 186, 95, 203, 147].

En los últimos años, la aplicación de técnicas de aprendizaje automático, algoritmos de visión por computadora y redes neuronales profundas para el procesamiento de este conjunto de imágenes han dado resultados sorprendentes, particularmente en la automatización de tareas repetitivas, como la segmentación de imágenes [26, 205, 195, 192, 188], la medición de características para diagnóstico y la clasificación de patologías [103, 159, 4, 75, 189]. En [96] Li y col. presentan un modelo basado en redes neuronales convolucionales que ajusta la relación no lineal entre las imágenes DWI y distintos mapas derivados del tensor de difusión. Lo más sorprendente de este modelo es que únicamente requiere 6 señales de difusión en direcciones distintas del gradiente de magnetización para estas estimaciones. La diferencia en el flujo de información entre el modelo antes mencionado y los métodos convencionales se ilustran en la Figura 1.2. En específico, se utiliza una red *encoder-decoder* tipo *U-Net* [143] que, empleando capas convolucionales, aumenta la dimensionalidad del canal de las imágenes gradualmente hasta extraer un vector de características y con las capas de la parte del *decoder*, junto con conexiones residuales, se reconstruyen los principales mapas de parámetros cuantitativos del tensor de difusión, la anisotropía fraccional, la difusividad media y las direcciones principales de la difusión. Del mismo modo, en [179], Wasserthal y col. presentan una arquitectura similar que utiliza tres modelos de redes neuronales convolucionales para la estimación de la orientación principal de la difusión, la segmentación de los tractos de interés y la

segmentación de los puntos de inicio y finalización del seguimiento de la tractografía. Finalmente, combinan la información de estas tres redes para obtener los histogramas de una tractografía probabilística. Sin embargo, el modelo presenta una fuerte dependencia de los parámetros del método de adquisición de resonancia magnética, por lo que sería necesario reajustar los parámetros del modelo en caso de utilizar imágenes obtenidas con distintos protocolos.

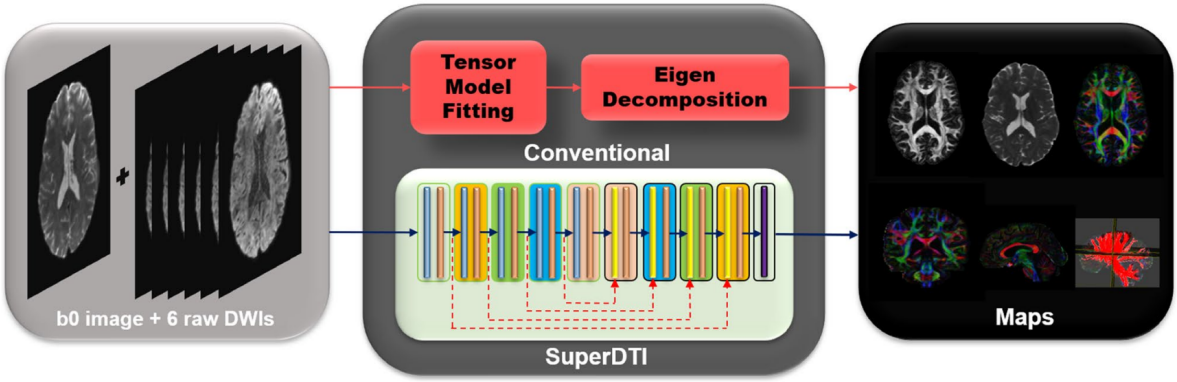

FIGURA 1.2: Comparación esquemática entre el ajuste convencional del modelo de tensor de difusión y el modelo de aprendizaje profundo para la generación de distintos mapas del tensor de difusión [96].

En [120] Nelkenbaum y col. proponen una red neuronal *AGYnet* tipo *autoencoder* para segmentar la materia blanca en el cerebro, esta toma como información de entrada imágenes estructurales ponderadas en tiempo T1 y los mapas de las direcciones principales de difusión. La arquitectura cuenta con compuertas de atención en la parte de *decoder* que permite combinar las características obtenidas de cada tipo de entrada. Esto permite mantener las características originales y resaltar aquellas que realicen una mayor contribución de información a la estimación del modelo [149]. Entonces, gracias a estos mecanismos de autoatención, ha surgido una alternativa a las redes neuronales convolucionales o recurrentes para el procesamiento de secuencias, los llamados *Transformers* [169]. Y, si bien en un principio este tipo de modelo tiene sus orígenes en el procesamiento de lenguaje natural y traducción automática, ha sido posible adaptar este tipo de arquitectura a tareas de visión computacional [42, 88, 98, 59, 128, 107, 183, 175]. Así, recientemente, en [85] Karimi y col. proponen una arquitectura basada únicamente en *Transformers* para la estimación de los coeficientes del tensor de difusión. Con esta arquitectura, mostrada en la Figura 1.3, demuestran que es posible modelar la relación no lineal entre las señales de difusión y el tensor de difusión de una región de pixeles y sus pixeles vecinos utilizando módulos basados en autoatención. Estos mecanismos de atención en redes neuronales profundas son técnicas avanzadas que permiten a los modelos ponderar diferentes partes de una entrada según su relevancia para una tarea

dada y se encuentran inspirados en la forma en que los seres humanos focalizan la atención a ciertas partes de un estímulo mientras ignoran otras. Estos mecanismos otorgan a las redes neuronales la capacidad de centrarse en porciones específicas de los datos de entrada, mejorando así la precisión y eficiencia en tareas como la traducción automática, generación de texto y el reconocimiento de imágenes [168].

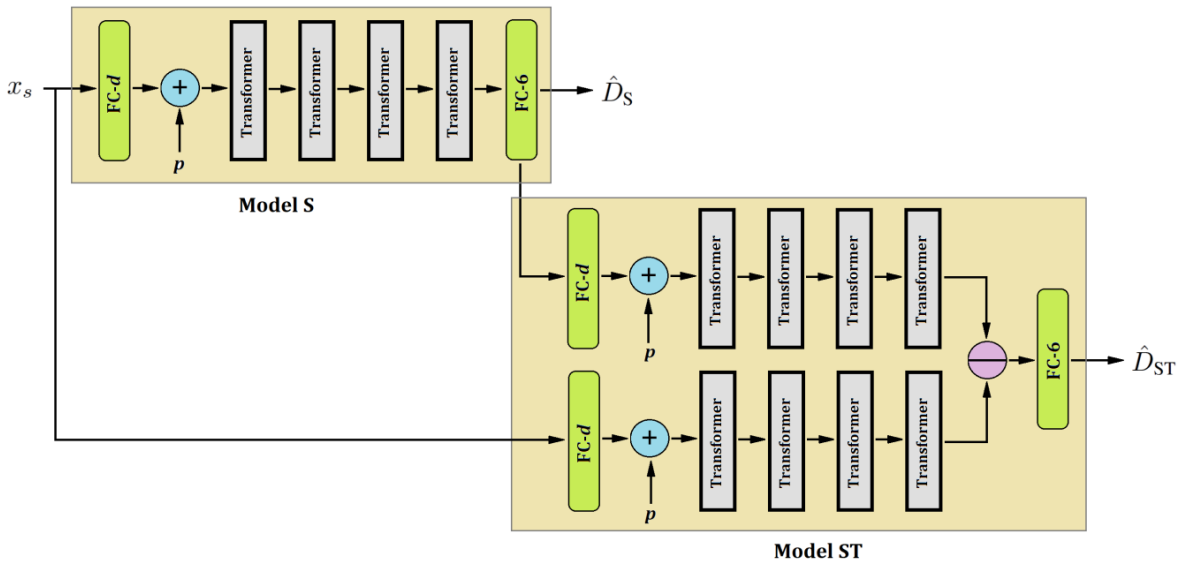

FIGURA 1.3: Arquitectura basada en dos modelos tipo *Transformers* para la estimación de los coeficientes del tensor de difusión [85].

Los primeros intentos de adaptar los mecanismos de atención en redes neuronales se remontan a finales de la década de 1980 [156]. Sin embargo, no fue hasta el año 2015 que se han realizado aportes contundentes en esta área y que, por consiguiente, impulsó la investigación en el desarrollos de estas técnicas [114, 106, 6, 105, 46]. En [29] Chen y col. presentan la arquitectura *TransUNet*, que utiliza mecanismos de autoatención desde la perspectiva de predicción secuencia a secuencia, en este caso pixel a pixel, en donde para compensar la pérdida de resolución en las características utilizan un esquema híbrido entre módulos convolucionales y *Transformers*, aprovechando las características espaciales de alta resolución y el contexto global para la segmentación de imágenes médicas. Sus resultados sugieren que una arquitectura basada tanto en *Transformers* como en bloques convolucionales presenta una mejor manera de aprovechar los mecanismos de atención que las arquitecturas basadas en un único principio.

1.2. Planteamiento del problema

Las imágenes DWI, como se ha visto, son actualmente las mejores herramientas no invasivas para estudiar la microestructura del tejido en el cerebro. No obstante, debido a la inherente naturaleza no lineal de los principios físicos utilizados por el escáner para la extracción de estas imágenes, se hacen presentes problemas como el ruido térmico en las señales, incompatibilidad con los vectores de difusión, desviación de las señales con cada muestra e inducción de corrientes de Eddy [163]. Además, cualquier mínimo movimiento del paciente durante la adquisición de las imágenes introduce distorsiones por lo que se convierte en todo un reto extraer imágenes cerebrales en infantes. En consecuencia, procesar y acondicionar las imágenes DWI para tareas de segmentación y extracción de características del tensor de difusión se vuelve un proceso tedioso y en la mayoría de los casos se requieren tiempos de procesamiento prolongados, aun con hardware especializado. Por otra parte, convencionalmente se utiliza el modelo de Stejskal–Tanner [85] para ajustar los coeficientes del tensor de difusión. En este procedimiento, se aplica una transformación a las señales de difusión para transformar los datos a un dominio logarítmico y, posteriormente, se utilizan distintos algoritmos de regresión lineal para obtener una solución a estos coeficientes. Alternativamente, es posible resolver la ecuación no lineal original, pero esto conduce a otros problemas como comportamientos caóticos y alto costo computacional. Además, en la práctica, se recomienda extraer la mayor cantidad posible de imágenes para aumentar la exactitud y robustez en la estimación del tensor, lo que sugiere que los métodos actuales no resultan óptimos. En particular, estas estimaciones clásicas se realizan por cada pixel o voxel, según sea el caso, de manera independiente y son incapaces de aprovechar la regularidad espacial de la microestructura del tejido en la materia blanca que se comparte entre pixeles o voxeles vecinos.

En este sentido, en el presente trabajo de investigación se propone un modelo de inteligencia artificial basado en redes neuronales híbrido con bloques tipo *Transformers* y bloques convolucionales para la estimación de los mapas característicos del tensor de difusión. En particular, se estiman los mapas de anisotropía fraccional, difusividad media y las orientaciones de difusión. Este modelo busca reducir los tiempos de procesamiento y el número de señales de difusión necesarias para el cálculo de los mapas, mientras que al mismo tiempo debe ser capaz de extraer una representación mejor adaptada que los métodos convencionales mencionados anteriormente, tomando en cuenta que la arquitectura propuesta es capaz de modelar la relación entre los elementos de una secuencia sin importar la distancia a la que se encuentran uno respecto del otro.

1.3. Justificación

Los mecanismos de atención se han convertido en una parte integral al momento de modelar secuencias de elementos en múltiples tareas, principalmente en traductores de texto como *GPT-3* [25] y modelos de visión computacional como el *Vision Transformer* (ViT) [42]. Esto se debe a su capacidad de modelar dependencias sin importar la distancia entre los elementos de esta secuencia [89]. Y, a pesar de su arquitectura sencilla, el codificador tipo *Transformer* ha demostrado un desempeño competitivo con las redes neuronales tipo convolucionales al capturar patrones repetibles en un conjunto de información. Por otro lado, el tejido cerebral es, por naturaleza, espacialmente regular en el sentido de que las propiedades microestructurales del cerebro, tal como la orientación de las fibras de materia blanca, no cambia aleatoriamente entre voxeles adyacentes [85]. Aún más, estas correlaciones espaciales en el tejido cerebral se comparten con gran medida entre distintas partes del cerebro puesto que, por ejemplo, existen conjuntos de fibras que cruzan de un hemisferio del cerebro al otro. Por este motivo, se propone utilizar una arquitectura de red neuronal basada en *Transformers* y operadores convolucionales para modelar las relaciones locales y espaciales entre las señales de difusión y los parámetros del tensor de difusión. Esta arquitectura debe permitir realizar un procesamiento de imágenes DWI con un preprocesamiento mínimo en un menor tiempo que los métodos convencionales basados en el modelo de Stejskal–Tanner.

1.4. Contribución

El análisis y procesamiento de imágenes médicas utilizando algoritmos de IA, en específico con redes neuronales profundas, ha dado resultados sobresalientes en la última década. Esto ha permitido aumentar la precisión mientras se disminuyen tiempos de procesamiento que favorecen tanto en el área clínica como en neurociencia, por lo que resulta importante la investigación y desarrollo de este tipo de modelos para el soporte en tareas de clasificación y estimación de parámetros en este tipo de imágenes. Por tanto, en el presente trabajo se propone una arquitectura basada en redes neuronales que integra mecanismos de atención y operadores convolucionales para la estimación del tensor de difusión en imágenes cerebrales por IRM y la principal ventaja de este modelo respecto al método convencional es que utiliza una menor cantidad de señales de difusión, demostrando que las redes neuronales profundas son capaces de extraer una mayor cantidad de información a partir de una menor cantidad de datos de entrada. De esta forma, el diseño y metodología de esta arquitectura se presenta como una solución alternativa a los métodos convencionales utilizados para extraer estos mapas.

1.5. Objetivos

En esta sección se presenta el objetivo general establecido para el presente trabajo de investigación y se determinan los objetivos específicos del mismo.

1.5.1. Objetivo general

Diseñar un modelo basado en redes neuronales profundas para estimar mapas derivados del tensor de difusión en imágenes cerebrales de resonancia magnética.

1.5.2. Objetivos específicos

- Organizar la base de datos de imágenes DWI que se utilizará para la construcción del modelo.
- Extraer las imágenes DWI con un único valor del gradiente de atenuación y realizar el preprocesamiento estándar para extracción de ruido, corrección de anillos de Gibbs, corrección de movimiento y corrección de corrientes de Eddy.
- Obtener el tensor de difusión mediante el modelo de Stejskal–Tanner y calcular los mapas de anisotropía fraccional, difusividad media y orientaciones de difusión para cada caso de la base de información.
- Diseñar e implementar una red neuronal profunda que integre mecanismos de atención para la estimación de los mapas del tensor de difusión.
- Optimizar el modelo mediante entrenamiento supervisado.
- Validar los resultados obtenidos a través de diversas métricas de similitud.

1.6. Alcances y limitaciones

De acuerdo con el planteamiento del problema de la presente investigación y los objetivos antes descritos, se definen los siguientes puntos para especificar los alcances y las limitaciones en el desarrollo de la metodología. En primer lugar, se debe considerar el tamaño y la cantidad de parámetros de la arquitectura que se ha de implementar y asegurar que el modelo sea realizable considerando el sistema físico con el que se dispone, en cuanto a memoria y capacidad de procesamiento. Es también por esto último que el diseño del modelo debe ser tal que el procesamiento de la información de IRM se realice por pixeles, dejando como trabajo futuro la posibilidad de extender el procesamiento a 3 dimensiones. Por otro lado, es importante mencionar que los datos que se generan para el

entrenamiento se extraen utilizando la herramienta de código libre FSL (*FMRIB Software Library*), por lo que los resultados y desempeño del modelo dependen directamente del uso eficiente y prestaciones de esta herramienta. Por último, el entrenamiento y ajuste del modelo se basa únicamente en un conjunto de datos en donde se han utilizado dos protocolos distintos para la extracción de las imágenes, por lo que la robustez y generalización del modelo al utilizar una base de datos externa para validación depende del grado de similitud en los protocolos y parámetros con los que se haya realizado la extracción de estas últimas imágenes.

1.7. Organización del documento de tesis

El presente documento de tesis se desarrolla en 5 capítulos. En el primer capítulo, se aborda la introducción y los antecedentes de la investigación. En el segundo capítulo se presenta el contenido teórico empleado en el diseño, caracterización y formulación matemática del modelo propuesto. En el tercer capítulo se muestra la metodología empleada para la elaboración de la investigación, así como la arquitectura propuesta para la estimación del tensor de difusión. Posteriormente, en el cuarto capítulo se abordan las especificaciones de implementación y los resultados que se han obtenido. Y, finalmente, en el quinto capítulo se presentan las conclusiones de la presente tesis y los trabajos futuros que se proponen para continuar con esta línea de investigación.

Capítulo 2

Marco Teórico

En este segundo capítulo se presenta el contenido teórico necesario para el desarrollo del presente trabajo de investigación. Para comenzar, se describen los principios físicos involucrados en la tecnología de la imagenología por resonancia magnética, así como la descripción de las secuencias de imágenes que se emplean para la estimación y cálculo de los mapas paramétricos del tensor de difusión en imágenes cerebrales. Luego, se describen los algoritmos y modelos utilizados para el procesamiento de imágenes mediante redes neuronales profundas, junto con sus respectivas ventajas y limitaciones. Y, por último, se muestran los fundamentos matemáticos concernientes a los módulos de autoatención y la manera en que se integran en arquitecturas modernas.

2.1. Imágenes por resonancia magnética

La tecnología de imágenes por IRM es una técnica que permite generar imágenes médicas anatómicas y fisiológicas en organismos o tejido. Esta técnica ha surgido gracias a los avances realizados en múltiples disciplinas tales como la física, medicina, ingeniería, biología, entre muchas más. Particularmente, las aportaciones que determinaron el nacimiento de este método de extracción de imágenes no invasivas son las investigaciones en las propiedades de los materiales a bajas temperaturas, referente a la superfluidez y superconductividad; el desarrollo de una teoría completa del átomo; el descubrimiento del momento magnético en las partículas fundamentales; el desarrollo de nuevos métodos para obtener medidas de precesión en la propiedades magnéticas en partículas subatómicas; y las contribuciones en el desarrollo en espectroscopia de resonancia magnética nuclear (NMR) de alta resolución.

Los primeros experimentos realizados con esta técnica fueron efectuados en 1973 por los investigadores Lauterbur y Mansfield, quienes obtuvieron el premio nobel en química en el año 2003 gracias a sus aportaciones concernientes a la imagenología por resonancia magnética [49]. Para ese entonces, era bien sabido que el momento angular intrínseco, o *spin*, de un núcleo de hidrógeno inmerso dentro de un campo magnético oscila sobre el eje de dirección de este campo con una frecuencia que depende linealmente de la magnitud de intensidad de dicho campo [24]. Entonces, si un objeto es atravesado por un campo

magnético variable en el espacio, las frecuencias inducidas también serán espacialmente variables. A partir de esto, se demostró que es posible capturar y separar las componentes de frecuencias de estas señales de forma que sea factible obtener información espacial del objeto inmerso en el campo magnético [135, 117]. Esto último puede interpretarse como una codificación espacial de la estructura del objeto.

2.1.1. Construcción y principios físicos

La idea principal de la técnica de imágenes por resonancia magnética se basa en la interacción de un *spin* nuclear con un campo magnético externo. En concreto, a partir de la interacción de un protón de una molécula de hidrógeno con un campo externo resulta en un proceso de precesión de este protón alrededor del eje de dirección del campo, como se muestra en la Figura 2.1. De esta forma, la molécula de hidrógeno actúa como un pequeño imán, puesto que contiene únicamente una partícula con carga positiva, el protón, y una partícula con carga negativa, el electrón. Así que, gracias a que el cuerpo humano está constituido en mayor parte por agua, que incluye dos moléculas de hidrógeno y una de oxígeno, el potencial de la señal de hidrógeno es mayor que la de cualquier otro químico en el cuerpo [134, 135]. Entonces, cuando un paciente es posicionado dentro de un escáner, la mayoría de las partículas de hidrógeno del paciente se alinean con la polaridad del mismo escáner. Esto se conoce como el vector de magnetización.

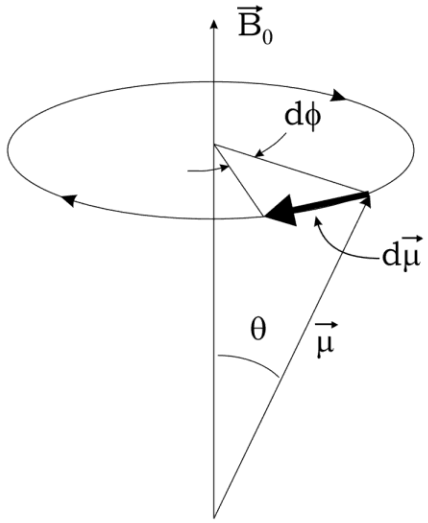

FIGURA 2.1: Movimiento circular $d\phi$ causado por un torque inducido $d\vec{\mu}$ por precesión. La interacción del spin de un protón $\vec{\mu}$ y el campo magnético produce un torque alrededor del eje de dirección del campo $\vec{B}_0$ [24].

De manera general, la tecnología por resonancia magnética se basa en magnetizar las moléculas de hidrógeno del objeto del cual se requiere extraer su estructura espacial interna y aplicar potentes campos magnéticos en distintas direcciones para forzar un movimiento oscilatorio de estas partículas. Luego, debido al movimiento de cargas, captar el movimiento de estas mismas mediante las corrientes que se inducen en los devanados para la reconstrucción de las imágenes. En este contexto, el objetivo es correlacionar una serie de señales de radiofrecuencia medidas con la localización espacial de cada señal. Esto último es posible gracias al hecho de que la adición de campos magnéticos variables espacialmente a lo largo del objeto producen señales con componentes de frecuencias que varían respecto al espacio [194]. En particular, la frecuencia angular del vector de momento magnético depende linealmente de la magnitud del campo aplicado y se obtiene a partir de la frecuencia de *Larmor* $w_L(x)$ dada por la siguiente expresión

$$w_L(x) = \gamma B(x) \tag{2.1}$$

en donde γ es la relación giromagnética y x denota las coordenadas espaciales en la dirección del gradiente del campo magnético. Esto significa que los componentes del espectro de las señales captadas por el escáner ahora representan información espacial, lo que permite que la estructura del objeto pueda ser reconstruida.

En este sentido, el vector de magnetización neto, en imágenes de resonancia magnética, es la suma vectorial de los momentos magnéticos de cada uno de los núcleos de hidrógeno individuales de la muestra. Así, en ausencia de un campo magnético, los momentos magnéticos individuales se encuentran orientados de manera aleatoria y el vector de magnetización resultante es cero, puesto que no hay una dirección particular a la que la mayoría de estos vectores apunten. De forma que, cuando un campo magnético con gradiente variable es aplicado sobre la muestra, el vector de magnetización de la muestra se alinea con el eje de dirección del campo magnético aplicado [117, 37]. De este modo, el procedimiento estándar consiste en aplicar un campo magnético fijo que polarice al vector de magnetización de la muestra y, a continuación, se aplican secuencias de pulsos magnéticos adicionales con distintos gradientes y de mayor potencia, provocando que el vector de magnetización neto rote y cambie de dirección. Posteriormente, después de cada pulso de la secuencia, se captura el proceso de relajamiento natural del vector de magnetización hacia el estado original mediante los devanados del sensor del escáner, lo que permitirá una reconstrucción de la estructura espacial interna de la muestra. En específico, una secuencia de pulsos de resonancia magnética es un conjunto programado de gradientes magnéticos cambiantes. Cada una de esta secuencia se especifica mediante una serie de parámetros y varias secuencias agrupadas en un protocolo de resonancia magnética. En general, estos parámetros son el tiempo de eco (TE), referente al tiempo entre la aplicación del pulso de excitación de radiofrecuencia y el pico de la señal inducida en la bobina; el tiempo de repetición (TR), que hace referencia al tiempo desde la aplicación de un pulso de excitación hasta la aplicación del siguiente pulso y

determina cuánta magnetización longitudinal se recupera entre cada pulso; el ángulo de giro, referente al fenómeno por el cual el eje del protón de hidrógeno se desplaza desde su eje Z del plano longitudinal a su eje XY del plano transversal; las secuencias de pulsos de recuperación de inversión, que se utilizan para anular selectivamente la señal de ciertos tejidos como la grasa o líquido; y los parámetros espaciales de adquisición del espacio-k (*k-space*), el cual es un arreglo que representa las frecuencias espaciales de las imágenes capturadas y en donde se almacena la información digitalizada antes de ser reconstruida, por lo que además define la transformación de Fourier necesaria para realizar esta reconstrucción [24, 81, 115]. En consecuencia, diferentes combinaciones de estos parámetros afectan el contraste del tejido y la resolución espacial permitiendo obtener distintos tipos de imágenes. En los siguientes apartados se describen las modalidades de imágenes por resonancia magnética con las que se trabaja en esta investigación.

2.1.2. Imágenes anatómicas

Las imágenes por resonancia magnética más empleadas en la práctica son las imágenes ponderadas en tiempo T1W y T2W, las cuales revelan la estructura y anatomía del tejido blando y adiposo. Estas se pueden caracterizar por dos tiempos de relajación diferentes. El tiempo T1 corresponde al tiempo de relajación longitudinal y es la constante de tiempo que determina la velocidad a la que los protones excitados regresan al estado de equilibrio. Además, es una medida del tiempo que tardan los protones giratorios en alinearse nuevamente con el campo magnético externo inicial. El tiempo T2 es el tiempo de relajación transversal y corresponde a la constante de tiempo que determina la velocidad a la que los protones excitados alcanzan el equilibrio o se desfasan entre sí. Este último se puede interpretar como una medida del tiempo que demoran los protones giratorios en perder la coherencia de fase entre los núcleos que giran perpendiculares al campo magnético principal [3, 151, 150]. En este contexto, se habla de ponderación de imágenes puesto que las señales de imágenes por resonancia magnética no son cuantitativas de manera directa y cada imagen individual puede contener contribuciones de señales de múltiples mecanismos. Así, se pretende maximizar las características deseadas mientras se minimizan las características no deseadas.

	TR (msec)	TE (msec)
Imágenes T1w	500	14
Imágenes T2w	4,000	90

TABLA 2.1: Tiempos aproximados de eco y repetición para imágenes anatómicas T1 y T2.

Las imágenes ponderadas en T1 se producen utilizando tiempos cortos de TE y TR. El contraste y el brillo de la imagen están determinados principalmente por las propiedades T1 del tejido. En cambio, las imágenes ponderadas en T2 se producen utilizando tiempos

de TE y TR más largos, como se muestra en la Tabla 2.1. En estas imágenes, el contraste y el brillo están determinados predominantemente por las propiedades T2 del tejido [41, 58]. En general, las imágenes ponderadas en T1 y T2 se pueden diferenciar fácilmente observando el líquido cefalorraquídeo (LCR), el cual se muestra oscuro en las imágenes ponderadas en T1 y brillante en las imágenes ponderadas en T2.

2.1.3. Imágenes ponderadas por difusión

Las imágenes ponderadas por difusión detectan el movimiento aleatorio de los protones de las moléculas de agua y estas moléculas se difunden con relativa libertad en el espacio extracelular, lo que se conoce como movimiento Browniano [118]. Por el contrario, este movimiento se encuentra significativamente restringido en el espacio intracelular y, particularmente, en el tejido interno de materia blanca cerebral que se compone principalmente de conexiones nerviosas [8, 7, 63]. Entonces, en un medio isotrópico, la difusión de un material descrito por un campo escalar $\phi(\mathbf{r}, t)$, se encuentra caracterizado por la siguiente ecuación

$$\frac{\partial \phi}{\partial t} = D\nabla^2\phi \tag{2.2}$$

en donde la constante de difusión D tiene un valor de $0.0023mm^2/s$ para la autodifusión del agua a condiciones ambiente. En una dimensión, esta ecuación también se conoce como la segunda ley de Fick [109, 178]. Sin embargo, si el coeficiente de difusión D no es constante y depende directamente de la posición o del campo ϕ, es decir para un medio no homogéneo, la ecuación anterior se remplaza por una de mayor complejidad

$$\frac{\partial \phi}{\partial t} = \nabla[D(\phi, \mathbf{r})\nabla\phi]. \tag{2.3}$$

La tecnología que permitió el uso práctico de las imágenes ponderadas por difusión fue el desarrollo de métodos para imágenes ecoplanares robustas (EPI). Esto gracias a que la fase acumulada asociada al movimiento involuntario del paciente puede exceder considerablemente la difusión inducida por los cambios de fase y, en consecuencia, es necesario utilizar técnicas de imagenología lo suficientemente rápidas que permitan despreciar el movimiento de la muestra. Es por ello que, para sensibilizar las imágenes de resonancia magnética a la difusión, la intensidad del campo magnético varía linealmente mediante un gradiente de campo por pulsos. Luego, dado que la precesión es proporcional a la fuerza del imán, los protones comienzan a preceder a diferentes velocidades, lo que da como resultado la dispersión de la fase y la pérdida de señal. Se aplica otro pulso de gradiente de la misma magnitud, pero con dirección opuesta para cambiar la fase de los espines. Este alineamiento no es paralelo de manera perfecta para los protones que se han movido durante el intervalo de tiempo entre los pulsos y, por

consiguiente, la intensidad de la señal medida por el escáner de resonancia magnética se reduce. Este método de pulso de gradiente de campo ha sido propuesto inicialmente por Stejskal y Tanner [158], quienes derivaron la reducción de la señal obtenida mediante la aplicación de un gradiente de pulso relacionado con la cantidad de difusión que se ha generado. Así, con el fin de localizar esta atenuación de la señal para obtener imágenes de difusión, se deben combinar los pulsos de gradiente de campo magnético utilizados con pulsos de gradiente de sondeo de movimiento, según el método de Stejskal y Tanner. Es importante mencionar que esta combinación no es trivial, ya que surgen términos cruzados entre todos los pulsos de gradiente. Por tanto, la ecuación establecida por Stejskal y Tanner se vuelve imprecisa y la atenuación de la señal debe calcularse, ya sea analítica o numéricamente, integrando todos los pulsos de gradiente presentes en la secuencia de resonancia magnética y sus interacciones. El cálculo se vuelve muy complejo dados los muchos pulsos presentes en la secuencia de resonancia magnética y, como simplificación, Le Bihan [13] sugirió reunir todos los términos de gradiente controlados por un factor b, que depende solo de los parámetros de adquisición [154, 86, 104, 181, 123]. De esta forma, la atenuación de la señal S_b en presencia de difusión $D(mm^2/s)$ está dada por la expresión

$$S_b = S_0 e^{-bD} \tag{2.4}$$

donde S_0 es la señal sin ningún gradiente de difusión aplicado, por ejemplo $b = 0$, y S_b es la señal con gradiente de difusión $b\ s/mm^2$. Un valor clínico común de b es $1,000 s/mm^2$. Después, debido a que la difusión se encuentra restringida en tejido biológico, se hace referencia a la difusión D como un coeficiente de difusión aparente ADC. Este último se puede calcular a partir de

$$ADC = -\frac{1}{b} log(S_b/S_0). \tag{2.5}$$

El valor de b es una función de las amplitudes de los pulsos de gradientes aplicados a la muestra, los periodos y los intervalos utilizados en el experimento

$$b = \gamma_p^2 G^2 \delta^2 (\Delta - \frac{1}{3}\delta) \tag{2.6}$$

en donde γ_p es la relación giromagnética, G es la amplitud del gradiente, δ es la duración de cada pulso de gradiente y Δ es el intervalo entre cada pulso [3]. A partir de estas señales, es posible estimar el tensor de difusión como una extensión simple del modelo ADC, capaz de describir la anisotropía de difusión de las moléculas de agua [190, 21, 17, 31].

2.1.4. Tensor de difusión

En algunos tejidos biológicos la difusión es de carácter anisotrópico, como lo es en la materia blanca en el cerebro. Particularmente, la difusión se restringe por membranas celulares por lo que existe una difusión preferencial a lo largo de los tractos de materia blanca [63, 50, 173, 43, 127]. En consecuencia, es de interés el utilizar las imágenes ponderadas por difusión para visualizar la anisotropía en estos tractos. Esta anisotropía considera que las señales ponderadas por difusión dependen de la dirección de los gradientes aplicados. De esta forma, en un medio anisotrópico, la ecuación de difusión 2.2 se convierte a

$$\frac{\partial \phi}{\partial t} = \sum_{i=1}^{3} \sum_{j=1}^{3} D_{ij} \frac{\partial^2 \phi}{\partial x_i \partial x_j} \tag{2.7}$$

donde $\mathbf{D} = [D]_{ij}$ es un tensor simétrico conocido como tensor de difusión. Y, con respecto a una marco de referencia con ejes cartesianos, se puede expresar como

$$\mathbf{D} = \begin{bmatrix} D_{xx} & D_{xy} & D_{xz} \\ D_{yx} & D_{yy} & D_{yz} \\ D_{zx} & D_{zy} & D_{zz} \end{bmatrix} \tag{2.8}$$

en donde $D_{xy} = D_{yx}, D_{yz} = D_{zy}, D_{xz} = D_{zx}$ y los elementos de la diagonal principal representan la difusión a lo largo de las direcciones principales del gradiente x, y y z, mientras que el resto de los términos son elementos covariantes. Entonces, es de utilidad definir cantidades invariantes ante rotaciones e independientes de las direcciones en las que los gradientes se han aplicado, debido a que los ejes principales de difusión antes mencionados normalmente no se encuentran alineados con las características anatómicas. La métrica invariante ante rotaciones más simple es la huella o rastro de $\mathbf{D}$, el cual únicamente requiere tres señales con el gradiente de difusión aplicado de manera separada a lo largo de x, y y z [160, 116, 165]. Así, esta métrica se calcula a partir de

$$Trace[\mathbf{D}] = D_{xx} + D_{yy} + D_{zz} \tag{2.9}$$

y el componente de difusión promedio se puede expresar como

$$ADC = \frac{1}{3} Trace[\mathbf{D}]. \tag{2.10}$$

En el tejido isotrópico, los valores de los componentes de difusión promedio contienen valores similares a los de las imágenes ponderadas por difusión. Sin embargo, en tejido anisotrópico, una adquisición de imágenes por tensor de difusión permite calcular el tensor de difusión junto con parámetros estándares para medir y visualizar la anisotropía en la materia blanca cerebral [3]. Esto último requiere capturar múltiples señales con el gradiente de difusión aplicado en al menos 6 direcciones distintas.

Luego, debido a que el tensor **D** es simétrico este debe contener tres vectores propios ortogonales $\mathbf{e}_1$, $\mathbf{e}_2$ y $\mathbf{e}_3$ con constantes de difusión en estas tres direcciones dados por sus correspondientes valores propios λ_1, λ_2 y λ_3. De este modo, las imágenes pueden ser reconstruidas para representar la anisotropía del tejido. Además, estos valores propios pueden ser convencionalmente ordenados tal que $\lambda_1 \geq \lambda_2 \geq \lambda_3$; y el tensor puede ser representado gráficamente como un elipsoide con los ejes mayores dados por los valores propios, y las direcciones de estos ejes dados por los vectores propios. En el tejido de materia blanca en el cerebro, $\mathbf{e}_1$ se encuentra alineado con la dirección local del tracto de fibra y $\lambda_1 > \lambda_2 \approx \lambda_3$ ya que estas fibras son aproximadamente cilíndricas [5, 172, 100, 30, 12]. Por tanto, una de las medidas más empleadas en la práctica es la anisotropía fraccional (FA), la cual se puede expresar en múltiples formas

$$FA = \sqrt{1 - \frac{\lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_1\lambda_3}{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}} \tag{2.11}$$

$$FA = \sqrt{\frac{3}{2}\frac{(\lambda_1 - \bar{\lambda})^2 + (\lambda_2 - \bar{\lambda})^2 + (\lambda_3 - \bar{\lambda})^2}{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}} \tag{2.12}$$

en donde $\bar{\lambda} = (\lambda_1 + \lambda_2 + \lambda_3)/3 = Trace[\mathbf{D}]/3$ es el promedio de los valores propios del tensor **D**, también nombrado como difusividad media. Entonces, resulta evidente que para valores de $FA = 0$ se presenta el medio isotrópico en la muestra, mientras que valores de $FA = 1$ representan la anisotropía máxima e implica que $\lambda_1 \neq 0$ y $\lambda_2 = \lambda_3 = 0$ puesto que las señales de difusión contienen únicamente una sola dirección [9, 91, 97, 48]. En la Figura 2.2 se muestran dos imágenes de vista superior de diferentes tipos de mapas obtenidos por tensor de difusión.

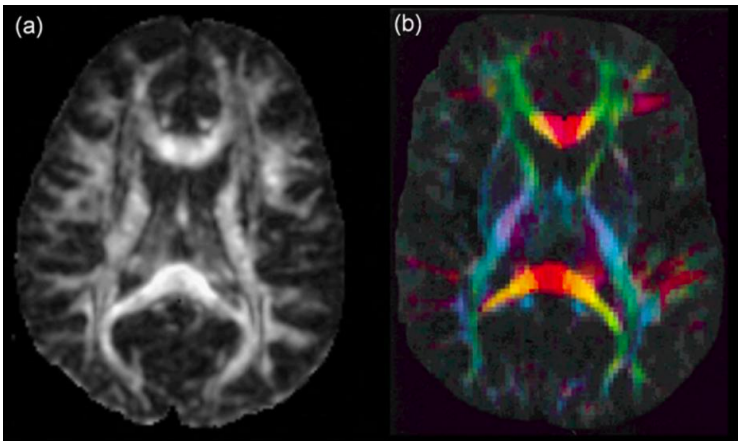

FIGURA 2.2: Imágenes DTI: a) Anisotropía fraccional y b) mapa de direcciones de difusión [3].

La información obtenida a partir de imágenes por tensor de difusión permite reconstruir mapas tridimensionales de los tractos de materia blanca en ciertas regiones del cerebro, o en todo el tejido de materia blanca. Esto último se conoce como tractografía, y se realiza mediante la conexión de los vectores propios principales en voxeles adyacentes para formar caminos continuos. Por otra parte, los mapas de anisotropía fraccional resultan útiles en el contexto clínico, ya que ayudan a diagnosticar daños cerebrales por lesiones o golpes [132, 84, 141].

2.2. Visión computacional con redes neuronales

El aprendizaje automático es un área de investigación dedicado a desarrollar métodos, algoritmos y modelos que emulen la capacidad de los seres humanos de aprender, es decir, métodos que aprovechan los datos para mejorar el rendimiento en algún conjunto de tareas y forman parte del campo de la inteligencia artificial. Estos algoritmos de aprendizaje automático construyen un modelo basado en datos de muestra, conocidos como datos de entrenamiento, para hacer predicciones o tomar decisiones sin estar programados explícitamente para hacerlo. Se utilizan en una amplia variedad de aplicaciones como en medicina, clasificación de grandes conjuntos de datos y visión computacional, donde es difícil desarrollar algoritmos robustos para efectuar las tareas necesarias [73]. Los problemas de aprendizaje automático, generalmente, se pueden dividir en tres categorías: el aprendizaje supervisado, no supervisado y por refuerzo. Los algoritmos de aprendizaje supervisado exploran la información recolectada de experimentos y se mapean las entradas con las salidas esperadas del modelo. En el aprendizaje por refuerzo se le asigna al modelo una métrica para evaluar su desempeño y, posteriormente, se obliga al modelo a maximizar esta métrica al modificar aleatoriamente su genotipo o estructura en un ambiente controlado [51]. En cambio, en el aprendizaje no supervisado no se dispone de las salidas o resultados esperados del modelo, sino que este mismo es el encargado de encontrar patrones e inferir el resultado esperado a partir de una colección de datos de entrada. Las redes neuronales son un tipo de algoritmo que se puede optimizar con cualquiera de los tres tipos de aprendizaje mencionados. En este apartado se describen de manera general la teoría y fundamentos utilizados en el aprendizaje computacional mediante redes neuronales basadas en aprendizaje supervisado por gradiente.

2.2.1. Perceptrón multicapa

Una red neuronal artificial (RNA) es un modelo de inteligencia artificial que permite a las computadoras procesar información de una manera inspirada biológicamente en el cerebro humano. En donde, gracias al descubrimiento de la neurona y sus funciones, ha sido posible modelar de manera funcional el comportamiento emergente de las redes neuronales de los seres humanos. Este se basa en el procesamiento interconectado entre

nodos que operan sobre una variable de entrada y la transmisión direccional de estas variables. Las redes neuronales artificiales intentan resolver problemas complicados, como el reconocimiento de patrones o diagnosticar enfermedades. En los primeros días de la inteligencia artificial, el campo desarrolló soluciones para problemas que son naturalmente difíciles para los seres humanos, pero relativamente fáciles para las computadoras como problemas que pueden describirse mediante una lista de reglas matemáticas. El verdadero desafío para la inteligencia artificial ha sido el resolver las tareas que son fáciles de realizar para las personas pero difíciles de describir formalmente, es decir, problemas que se resuelven intuitivamente o de forma automática, como reconocer palabras o rostros en imágenes [60]. Entonces, estos nodos se modelan mediante el denominado perceptrón, propuesto por Rosenblatt [144, 139, 162] y se compone de un clasificador binario que toma un vector como entrada $\mathbf{x}$, realiza una suma ponderada de los elementos de este vector con una función no lineal fija para obtener un vector de características $\phi(\mathbf{x})$ y se utiliza este vector para generar un modelo lineal generalizado dado por

$$y(\mathbf{x}) = g(\mathbf{x}^T \phi(\mathbf{x})) \tag{2.13}$$

donde la función no lineal $g(\cdot)$ es comúnmente la función escalón, la función sigmoide, función rampa o funciones trigonométricas hiperbólicas. Luego, el algoritmo utilizado para determinar los parámetros $\mathbf{w}$ del perceptrón puede ser traducido a un problema de optimizado mediante la minimización de la función de error. En este sentido, los modelos lineales para regresión y clasificación se basan en combinaciones lineales de funciones de base no lineales fijas $\phi_j(x)$ y toman la forma

$$y(\mathbf{x}, \mathbf{w}) = g(\sum_{j=1}^{M} w_j \phi_j(\mathbf{x})) \tag{2.14}$$

donde $g(\cdot)$ es una función no lineal de activación en el caso de clasificación y es la identidad en el caso de regresión. La idea es ampliar este modelo haciendo que las funciones de base $\phi_j(x)$ dependan de los parámetros y, posteriormente, permitir que estos parámetros se ajusten junto con los coeficientes $\{w_j\}$ durante el entrenamiento. Existen múltiples formas de construir funciones de base no lineales paramétricas. En este sentido, cada función base es en sí misma una función no lineal de una combinación lineal de las entradas, donde los coeficientes en la combinación lineal son parámetros adaptativos [14, 1, 65, 99]. Entonces, para construir una red neuronal se deben generar M combinaciones lineales de las variables de entrada $\{x_1, x_2, ..., x_d\}$ para N capas totalmente conectadas, de la forma

$$a_j = \sum_{i=1}^{d} w_{ji}^{(n)} x_i + w_{j0}^{(n)} \tag{2.15}$$

donde $j = 1, 2, ..., M$ es el número del perceptrón en la capa y $n = 1, 2, ..., N$ es el número de la capa. Los parámetros w_{ji}^n son los pesos o ganancias y los términos w_{j0} son los valores base. Las cantidades a_j se conocen como activaciones, cada una de estas se transforma utilizando una función no lineal diferenciable $h(\cdot)$ y este resultado corresponde a las salidas de cada una de las neuronas tanto en las capas ocultas como en las capas de entrada y de salida. Por otro lado, una elección natural de la función de error sería el número total de patrones mal clasificados. En la siguiente Figura 2.3 se muestra una representación gráfica del flujo de información en un perceptrón.

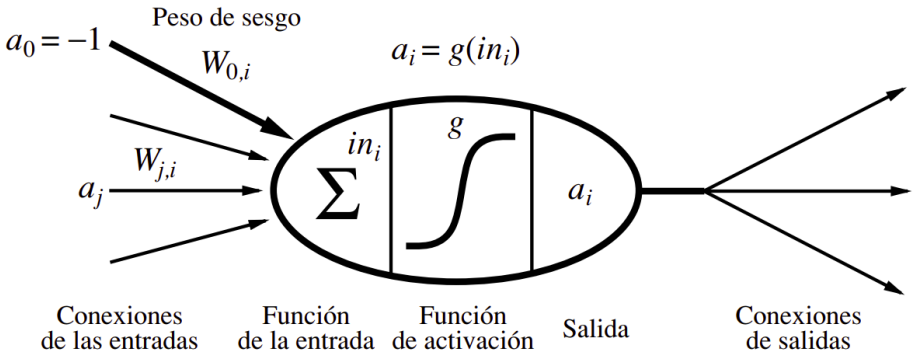

FIGURA 2.3: Modelo gráfico de un perceptrón o neurona artificial [146].

Sin embargo, esto no conduce a un algoritmo de aprendizaje simple porque el error es una función constante por partes de **w**, con discontinuidades dondequiera que un cambio en **w** haga que el límite de decisión se mueva a través de uno de los puntos de datos [197, 122, 121]. En estos casos, los métodos basados en cambiar **w** usando el gradiente de la función de error no se pueden aplicar porque el gradiente es cero en casi todo el espacio de exploración.

2.2.2. Aprendizaje basado en gradiente

Las redes neuronales artificiales formadas a partir de perceptrones constituyen una clase general de funciones no lineales paramétricas de un vector **x** de variables o características de entrada a un vector **y** de variables de salida. Como se ha mencionado, se realiza un mapeo entre estos dos vectores. En tal caso, una manera simple de abordar el problema de determinar los parámetros de una red artificial es minimizar la suma de los errores cuadráticos de forma que, dado un vector de entrada $\{x_n\}$ donde $n = 1, 2, ..., N$, y su correspondiente conjunto de vectores objetivo $\{t_n\}$, se minimiza la función de error

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \|\mathbf{y}(x_n, \mathbf{w}) - t_n\|^2. \tag{2.16}$$

Sin embargo, la mayor diferencia entre los modelos lineales clásicos y las redes neuronales es que la no linealidad de una red neuronal hace que la mayoría de las funciones de pérdida no sean convexas. Esto significa que las redes neuronales se suelen entrenar utilizando optimizadores iterativos basados en el gradiente que se limitan a llevar la función de coste a un valor muy bajo, en lugar de los estimadores de ecuaciones lineales utilizados para entrenar modelos de regresión lineal o los algoritmos de optimización convexa con garantías de convergencia global, utilizados para entrenar la regresión logística o máquinas de soporte vectorial [60]. Esta optimización convexa converge a partir de cualquier parámetro inicial y el descenso de gradiente estocástico aplicado a funciones de pérdida no convexas no garantiza la convergencia debido a que es sensible a los valores de los parámetros iniciales. Entonces, en el caso de las redes neuronales de procesamiento hacia adelante, es importante inicializar todos los pesos con pequeños valores aleatorios. Los sesgos pueden ser inicializados a cero o a pequeños valores positivos. En la mayoría de los casos, el modelo paramétrico se define mediante una distribución $p(\mathbf{y}|\mathbf{x};\theta)$ y es posible emplear simplemente el principio de máxima verosimilitud, lo que significa que se utiliza la entropía cruzada entre los datos de entrenamiento y las predicciones del modelo como función de coste [51]. En otras ocasiones, se adopta un enfoque en el que en lugar de predecir una distribución de probabilidad completa de $\mathbf{y}$, el modelo se limita a predecir alguna estadística de $\mathbf{y}$ condicionada en $\mathbf{x}$. Así, funciones de pérdida especializadas permiten entrenar un predictor de estas estimaciones.

En general, la tarea es encontrar el vector de pesos $\mathbf{w}$ que minimiza la función $E(\mathbf{w})$. Para ello, es importante tener una imagen geométrica de la función de error, puesto que se puede representar como una superficie en el espacio de los pesos o ganancias de la red neuronal. En primer lugar, hay que considerar que si se realiza un pequeño paso en el espacio de pesos desde $\mathbf{w}$ hasta $\mathbf{w} + \delta\mathbf{w}$, el cambio en la función de error es $\delta E \simeq \delta\mathbf{w}^T \nabla E(\mathbf{w})$, donde el vector $\nabla E(\mathbf{w})$ apunta en la dirección de mayor tasa de incremento de la función de error. Dado que el error $E(\mathbf{w})$ es una función continua suave de $\mathbf{w}$, su valor más pequeño se producirá en un punto del espacio de pesos tal que el gradiente de la función de error se desvanece

$$\nabla E(\mathbf{w}) = 0 \tag{2.17}$$

ya que de lo contrario se podría obtener un pequeño paso en la dirección de $-\nabla E(\mathbf{w})$ y así reducir aún más el error. Los puntos en los que el gradiente desaparece se denominan puntos estacionarios y pueden clasificarse en mínimos, máximos y puntos de equilibrio. En concreto, el objetivo es encontrar un vector $\mathbf{w}$ tal que $E(\mathbf{w})$ tome su valor más pequeño posible. Sin embargo, la función de error suele tener una alta dependencia no lineal de los pesos y los parámetros de sesgo, por lo que habrá muchos puntos en el espacio de pesos en los que el gradiente desaparece o es numéricamente muy pequeño. En la práctica, es extremadamente difícil y exhaustivo encontrar una solución analítica a la ecuación $\nabla E(\mathbf{w}) = 0$, por lo que se recurre a procedimientos numéricos iterativos. La

optimización de funciones continuas no lineales es un problema ampliamente estudiado y la mayoría de las técnicas implican la elección de algún valor inicial $\mathbf{w}(0)$ para el vector de pesos y, como resultado, el desplazamiento a través del espacio de pesos en una sucesión de pasos es de la forma

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} + \Delta\mathbf{w}^{(\tau)} \tag{2.18}$$

donde τ representa el paso de iteración. Diferentes algoritmos implican diferentes opciones para la actualización del vector de pesos $\Delta\mathbf{w}^{(\tau)}$. Estos algoritmos suelen hacer uso de la información de gradiente y por lo tanto requieren que, después de cada actualización, el valor de $\nabla E(\mathbf{w})$ se evalúe en el nuevo vector de pesos $\mathbf{w}^{(\tau+1)}$. El enfoque más sencillo y práctico para utilizar la información del gradiente es elegir la actualización del peso para incluir un pequeño paso en la dirección contraria del gradiente del error, de modo que

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta\nabla E(\mathbf{w}^{(\tau)}) \tag{2.19}$$

donde el parámetro $\eta > 0$ se conoce como la tasa de aprendizaje. Después de cada actualización, el gradiente se evalúa para el nuevo vector de pesos y se repite el proceso [14]. Para encontrar un mínimo suficientemente óptimo, puede ser necesario ejecutar un algoritmo basado en el gradiente varias veces, utilizando cada vez un punto de partida diferente elegido aleatoriamente, y comparando el rendimiento resultante en un conjunto de validación independiente.

2.2.3. Reconocimiento en visión computacional

La arquitectura de un sistema de visión artificial depende en gran medida de la aplicación, por lo que algunos sistemas son aplicaciones independientes que resuelven un problema específico de medición o detección, mientras que otros constituyen un subsistema de un diseño más grande que, por ejemplo, también contiene subsistemas para el control de actuadores mecánicos, planificación, bases de datos de información, gestión de interfaces de máquina, entre otros [47, 124]. No obstante, hay etapas comunes que se encuentran en casi todos los sistemas de visión artificial. La primera de ellas comprende la fase de previa al procesamiento de las imágenes, en donde se realiza la estandarización de los valores de estas imágenes y se aplica una transformación espacial. La etapa de extracción de características se centra en la minería y la identificación de características distintivas presentes en cada imagen que aporten la mayor cantidad de información relevante para el problema en cuestión. Y, por último, se diseña un modelo de inferencia, con el cual se lleva a cabo un aprendizaje basado en las características seleccionadas con el objetivo de lograr un reconocimiento visual.

Las imágenes digitales en su formato original no siempre resultan apropiadas para su procesamiento, por lo que se debe efectuar un procesamiento previo de la información

que contienen estas imágenes como aplicación de filtros o extracción de regiones de interés. Esto último, por sí solo, puede definir el éxito del modelo. Después, se continúa con la extracción y minería de características de estos datos. El objetivo es encontrar un conjunto de características representativas que describan adecuadamente cada entrada. Tal conjunto de características debe maximizar la probabilidad de mapear correctamente cada entrada con su respectiva salida y, al mismo tiempo, debe minimizar la probabilidad de asignar una entrada a una salida incorrecta. Como última etapa, se utiliza un algoritmo de inteligencia artificial para efectuar el objetivo principal del sistema como segmentación, detección, localización, seguimiento o clasificación. Generalmente, el desempeño del modelo depende directamente en la selección de las mejores características [51, 140]. Existen distintos tipos de características que se pueden utilizar para describir una imagen. Las características locales son aquellos descriptores que describen ciertas regiones concretas de la imagen, como la detección de bordes, puntos clave, textura, área o momentos inerciales. Por otro lado, se pueden extraer características globales que describen completamente una imagen, no solo una región específica, como histogramas de color RGB o HSV, conteo de pixeles o espectros en el dominio de la frecuencia.

En los sistemas modernos de reconocimiento visual se utilizan arquitecturas de redes neuronales profundas que combinan y relacionan el procesamiento de las últimas dos etapas descritas anteriormente. Esto significa que la selección de características se ajusta durante la optimización del modelo, lo que optimiza los resultados en tareas de reconocimiento y regresión [52]. En el siguiente apartado se describe un modelo basado en un tipo de redes neuronales profundas, que se ha utilizado en las últimas décadas con excelentes resultados para clasificación, segmentación y detección de objetos [159].

2.2.4. Redes neuronales convolucionales

Cuando se construye una red neuronal artificial con al menos una capa oculta se le conoce como una red neuronal profunda. El caso más simple supone una única capa oculta, como se ilustra en la Figura 2.4. La ventaja de añadir capas ocultas es la posibilidad de expandir el espacio de solución que puede representar la red. Cada unidad oculta, modelado por un perceptrón, se puede interpretar como un elemento que representa una función de umbral suave en el espacio de entradas. Entonces, como se ha mencionado, una unidad de salida es una combinación lineal con umbral suave de varias de estas funciones y, en general, con una única capa oculta suficientemente grande es posible representar cualquier función continua de las entradas con una precisión arbitraria y con dos capas es posible representar funciones discontinuas [146]. El algoritmo de aprendizaje para redes multicapa es similar al algoritmo de aprendizaje del perceptrón mostrado anteriormente. Una notable diferencia es que es posible tener varias salidas, así que se cuenta con un vector de salida $\mathbf{h}_W(x)$ en vez de un único valor, y cada ejemplo tiene un vector de salida $\mathbf{y}$. La mayor diferencia es que, mientras que el error $\mathbf{y} - \mathbf{h}_W$ en la capa de salida es claro, el error en las capas ocultas es desconocido, porque los datos

de entrenamiento no indican explícitamente cuál debe ser el valor que han de tomar los nodos ocultos. Sin embargo, el error se puede propagar hacia atrás desde la capa de salida hacia las capas ocultas y la capa de entrada.

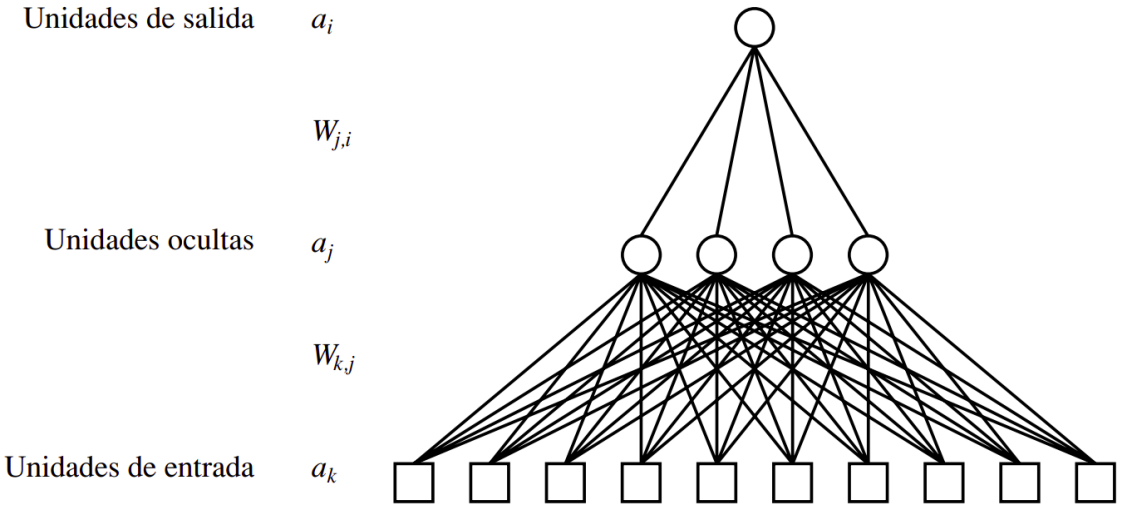

FIGURA 2.4: Una red neuronal multicapa con una capa oculta, múltiples entradas y una salida [146].

Las redes neuronales convolucionales (CNN) son una clase particular de red neuronal para el procesamiento de datos que tienen una topología similar a una cuadrícula. Este tipo de redes son análogas a las redes neuronales artificiales tradicionales en el sentido de que están compuestas por neuronas que se optimizan durante el aprendizaje. Cada neurona sigue recibiendo una entrada y realizando una operación. Desde los vectores de imágenes en bruto de entrada hasta la salida final de la puntuación de la clase, toda la red seguirá expresando una única función de puntuación perceptiva, los pesos de cada neurona. La última capa contendrá funciones de pérdida asociadas a las clases, y todas las técnicas de regularización y aprendizaje desarrollados para las redes neuronales tradicionales son, de igual manera, aplicables. La principal diferencia entre las redes neuronales convolucionales y las redes neuronales basadas en el perceptrón multicapa (MLP) es que las primeras se utilizan principalmente en el campo del reconocimiento de patrones en imágenes [204, 90, 148]. Esto permite codificar características específicas de la imagen en la arquitectura, al tiempo que se reducen aún más los parámetros necesarios para configurar el modelo.

El término convolucional indica que la red se basa en una operación matemática llamada convolución y constituye un tipo especializado de operación lineal. Las redes convolucionales son simplemente redes neuronales que utilizan la convolución en lugar de la multiplicación general de matrices en al menos una de sus capas. En su forma más general, la convolución es una operación sobre dos funciones de un argumento de valor real. En concreto, este operador matemático toma dos señales como entrada y genera

una tercera señal que indica como la primera de estas es transformada o deformada por la segunda. Este operador se encuentra definido por la siguiente integral

$$s(t) = \int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau = (f * h)(t). \tag{2.20}$$

En el procesamiento digital de señales, el primer argumento de la convolución suele denominarse entrada y el segundo argumento como núcleo, filtro o kernel [60]. El resultado de esta operación se denomina mapa de características. Ahora, si se consideran datos de carácter discreto, la ecuación anterior toma la forma

$$s[t] = \sum_{\tau=-\infty}^{\infty} f[\tau]h[t-\tau]. \tag{2.21}$$

De modo que, en las aplicaciones de aprendizaje automático en visión computacional, la entrada suele ser una matriz multidimensional de datos y el núcleo suele ser una matriz multidimensional de parámetros que son adaptados por el algoritmo de aprendizaje y estas matrices multidimensionales se denominan tensores.

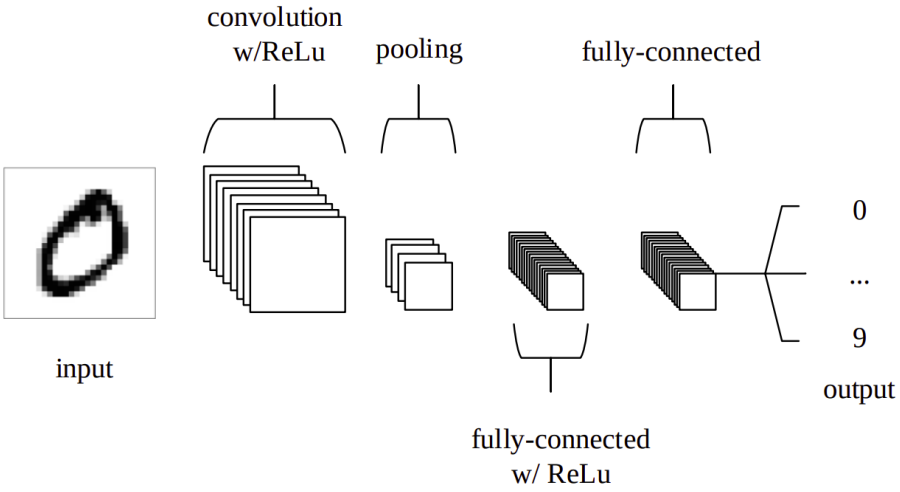

FIGURA 2.5: Diagrama de arquitectura de red neuronal convolucional [125].

En este contexto, se aplican operaciones convolucionales sobre más de un eje a la vez. En particular, para el procesamiento de una imagen bidimensional I como entrada y un kernel bidimensional K, el operador de convolución es de la forma

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(i-m, j-n)K(m,n). \tag{2.22}$$

Es importante mencionar que una de las características clave en este tipo de redes es que las neuronas están organizadas en tres dimensiones, la dimensionalidad espacial de

la entrada: altura y ancho, y la profundidad. Esta última no se refiere al número total de capas dentro de la red neuronal, sino a la tercera dimensión de un volumen de activación. A diferencia de las redes neuronales estándar, las neuronas de cualquier capa sólo se conectan a una pequeña región de la capa que la precede.

En específico, las CNN se componen de tres tipos de capas: las capas convolucionales, capas de agrupación (*pooling*) y capas totalmente conectadas. Cuando estas capas se apilan, se forma una arquitectura de una red neuronal convolucional, como se muestra en la Figura 2.5. Al igual que en otras formas de RNA, la capa de entrada procesa los valores de los pixeles de la imagen. La capa convolucional determina la salida de las neuronas que están conectadas a regiones locales de la entrada mediante el cálculo del producto escalar entre sus pesos y la región conectada al volumen de entrada. La unidad lineal rectificada (ReLu) tiene como objetivo aplicar una función de activación a cada elemento, como la función sigmoide, a la salida de la activación producida por la capa anterior. La capa de agrupación simplemente realiza un muestreo descendente a lo largo de la dimensionalidad espacial de la entrada dada, reduciendo aún más el número de parámetros dentro de esa activación. Y, finalmente, las capas totalmente conectadas realizan las mismas tareas que las RNA estándar y producen puntuaciones de clase a partir de las activaciones que se utilizan para la clasificación [125, 148]. En algunos casos, es recomendable utilizar la función ReLu entre estas capas, para mejorar el rendimiento. Mediante este sencillo método de transformación, las CNN son capaces de transformar la entrada original capa por capa utilizando técnicas convolucionales y de muestreo para producir puntuaciones de clase con fines de clasificación y regresión.

2.3. Redes neuronales con autoatención: *Transformers*

En esta sección se presentan los fundamentos teóricos concernientes a la arquitectura de redes neuronales basadas en autoatención, denominadas *Transformers*, para el procesamiento de datos secuenciales. Estos tipos de datos surgen a través de la medición de series temporales, por ejemplo, los valores diarios de un tipo de cambio de moneda o las características acústicas en marcos temporales sucesivos utilizados para el reconocimiento del habla. Los datos secuenciales también pueden surgir en contextos distintos a las series temporales, por ejemplo, la secuencia de pares de bases de nucleótidos a lo largo de una cadena de ADN o la secuencia de caracteres de una frase en inglés [14]. A continuación, se describe el tipo de información que es posible modelar mediante secuencias de datos y sus implicaciones al aplicar estos modelos en el campo del aprendizaje automático. Se abordan los mecanismos de atención y sus particularidades utilizadas en redes neuronales y se presenta el codificador *Transformer* para la extracción de características.

2.3.1. Modelado de secuencias

Una forma de tratar los datos secuenciales es simplemente ignorar los aspectos secuenciales y tratar las observaciones como variables aleatorias independientes e idénticamente distribuidas. Sin embargo, este enfoque no permite explotar los patrones secuenciales de los datos, como las correlaciones entre las observaciones que están próximos en la secuencia. Por otro lado, un grafo computacional es una forma de formalizar la estructura de un conjunto de cálculos, como los implicados en la asignación de entradas y parámetros a salidas y pérdidas. Para expresar estos efectos en un modelo probabilístico es necesario modificar la suposición de variables aleatorias independientes y una de las formas más sencillas de hacerlo es considerar un modelo de Markov [15]. En primer lugar, se observa que, sin pérdida de generalidad, es posible utilizar la regla del producto para expresar la distribución conjunta de una secuencia de observaciones de la forma

$$p(x_1, ..., x_N) = \prod_{n=1}^{N} p(x_n|x_1, ..., x_{n-1}). \tag{2.23}$$

Ahora, asumiendo que cada una de las distribuciones condicionales del lado derecho en la Figura 2.6 es independiente de todas las observaciones anteriores excepto la más reciente, obtenemos la cadena de Markov de primer orden. La distribución conjunta para una secuencia de N observaciones bajo este modelo viene dada por

$$p(x_1, ..., x_N) = p(x_1) \prod_{n=2}^{N} p(x_n|x_{n-1}). \tag{2.24}$$

No obstante, de manera general el modelo de independencia sigue siendo altamente restrictivo. Para muchas observaciones secuenciales, se espera que las tendencias de los datos a lo largo de varias observaciones sucesivas proporcionen información importante para predecir el siguiente valor. Una forma de permitir que las observaciones anteriores tengan una influencia es pasar a cadenas de Markov de orden superior. Luego, es de interés un modelo para secuencias que no estén limitadas por las suposiciones de una cadena de Markov de un orden arbitrario y que, sin embargo, pueda ser determinado utilizando un número limitado de parámetros libres. Esto es posible introduciendo variables latentes adicionales que permitan construir una clase de modelos a partir de componentes simples. En este caso, para cada observación x_t, se incorpora una variable latente correspondiente h_t, que puede ser de distinto tipo o dimensionalidad que la variable observada [14]. De modo que, son las variables latentes las que forman una cadena de Markov, dando lugar a la estructura gráfica conocida como modelo de espacio de estados, que se muestra en la Figura 2.6. Esto satisface la propiedad clave de independencia condicional de que h_{t-1} y h_{n+1} son independientes dado t_n. La

distribución conjunta de este modelo está dada por

$$p(x_1, ..., x_N, h_1, ..., h_N) = p(h_1)[\prod_{n=2}^{N} p(h_n|h_{n-1})] \prod_{n=1}^{N} p(x_n|h_n). \tag{2.25}$$

Nótese que siempre hay un camino que conecta dos variables observadas x_n y x_1 a través de las variables latentes, y este camino nunca está bloqueado. Esta característica es la base en la construcción de redes neuronales recurrentes (RNN).

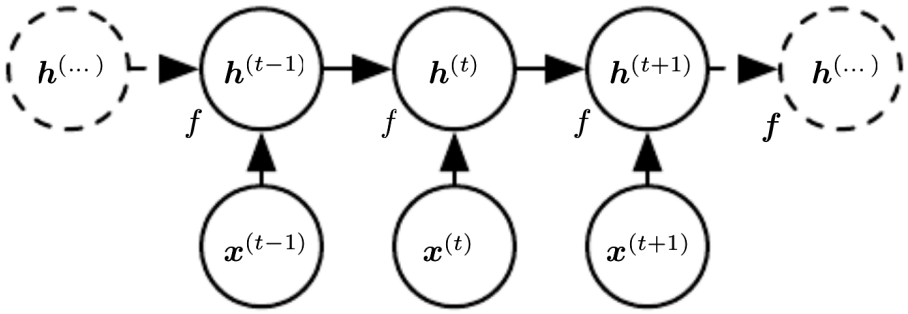

FIGURA 2.6: Cadena de Markov con variables latentes [60].

Por lo tanto, la distribución predictiva $p(x_{n+1}|x_1, ..., x_n)$ para la observación x_{n+1} dadas todas las observaciones anteriores no presenta ninguna propiedad de independencia condicional y, por lo tanto, las predicciones para x_{n+1} dependen de todas las observaciones anteriores.

2.3.2. Mecanismos de atención

La mayor parte de los modelos que se han desarrollado en el campo de la inteligencia artificial, si no es que todos, buscan imitar los procesos cognitivos de los seres humanos para resolver tareas complejas. Uno de estos procesos en particular ha sido el mecanismo de atención al ejecutar distintas tareas como la interpretación del habla o reconocimiento de objetos, al procesar los estímulos externos con atención focalizada o selectiva. Por medio de estos mecanismos es posible seleccionar y centrar la atención en un solo estímulo descartando otros irrelevantes que pueden interferir en el proceso. Los mecanismos de atención se han incorporado a las redes neuronales desde hace un par de décadas con el objetivo aumentar le eficiencia en los resultados del modelo, mientras se reduce la complejidad de este. Existen distintos tipos de mecanismos de atención y un modelo puede utilizar diferentes combinaciones de estas técnicas. Las tres categorías principales se distinguen por el tipo específicos de vectores de características, tipos específicos de consultas de modelos, o si es simplemente un mecanismo general que no está relacionado ni con el modelo de características, ni con el modelo de consultas [23]. El hecho de que

la atención no dependa de la organización de los vectores de características permite aplicarla a varios problemas que utilizan datos con estructuras diferentes. En general, la atención puede aplicarse a cualquier problema para el que pueda definirse o extraerse un conjunto de vectores de características.

Para implementar un bloque de atención general, en primer lugar, se plantea una arquitectura como un bloque para una aplicación específica y este simplemente toma una entrada, lleva a cabo un proceso específico y produce una salida deseada. Este bloque se compone de cuatro elementos: el modelo de características, el modelo de consulta, el modelo de atención y el modelo de salida [38]. Entonces, el bloque de atención toma como entrada la matriz $\mathbf{X} \in \mathbb{R}^{d_x \times n_x}$, donde d_x representa el tamaño de los vectores de entrada y n_x representa la cantidad de vectores de entrada. Las columnas de esta matriz pueden representar las palabras de una frase, los pixeles de una imagen o cualquier otra colección de datos. El modelo de características se emplea para extraer los n_f vectores de características $\mathbf{F} = f_1, f_2, ..., f_{n_f} \in \mathbb{R}^{d_f}$ de $\mathbf{X}$, donde d_f representa el tamaño de los vectores de características. Este último puede ser una red neuronal recurrente, una red neuronal convolucional, una transformación lineal de los datos o los datos mismos originales. Básicamente, el modelo de características consiste en todos los pasos que transforman la entrada original $\mathbf{X}$ en los vectores de características $f_1, f_2, ..., f_{n_f}$ que el modelo de atención procesa. Después, para determinar qué vectores de características atender, el modelo de atención requiere la consulta $\mathbf{q} \in \mathbb{R}^{d_q}$, donde d_q indica el tamaño del vector de consulta. Esta consulta es extraída a partir del modelo y, generalmente, se diseña en función del tipo de resultado que se desea obtener del modelo. Esta consulta puede interpretarse como una búsqueda o una pregunta. En este sentido, la consulta solicita la información necesaria de los vectores de características basándose en el contexto de predicción actual [23]. Posteriormente, los vectores de características y la consulta se utilizan como entrada para el modelo de atención. Este modelo consiste en un único módulo de atención general, como se muestra en la Figura 2.7. La entrada del módulo de atención es la consulta $\mathbf{q} \in \mathbb{R}^{d_q}$ y la matriz de vectores de características $\mathbf{F} = [f_1, f_2, ..., f_{n_f}] \in \mathbb{R}^{d_f \times n_f}$. De la matriz $\mathbf{F}$ se extraen dos matrices distintas: la matriz de llaves $\mathbf{K} = [k_1, k_2, ..., k_{n_f}] \in \mathbb{R}^{d_k \times n_f}$ y la matriz de valores $\mathbf{V} = [v_1, v_2, ..., v_{n_f}] \in \mathbb{R}^{d_v \times n_f}$, donde d_k y d_v indican las dimensiones de los vectores llave o columnas de $\mathbf{K}$ y de los vectores valor columnas de $\mathbf{V}$, respectivamente. La forma general de obtener estas matrices es mediante una transformación lineal de $\mathbf{F}$ utilizando las matrices de pesos $W_K \in \mathbb{R}^{d_k \times d_f}$ y $W_V \in \mathbb{R}^{d_k \times d_f}$, para $\mathbf{K}$ y $\mathbf{V}$, respectivamente. De esta manera, las matrices $\mathbf{K}$ y $\mathbf{V}$ se obtienen a partir de

$$\mathbf{K} = W_K \times \mathbf{F}, \qquad \mathbf{V} = W_V \times \mathbf{F} \tag{2.26}$$

en donde los elementos de ambas matrices de pesos pueden ser ajustados durante el entrenamiento o inducidas por el modelo [169]. Por tanto, el objetivo principal del

módulo de atención es producir una media ponderada de los vectores de valores en **V**.

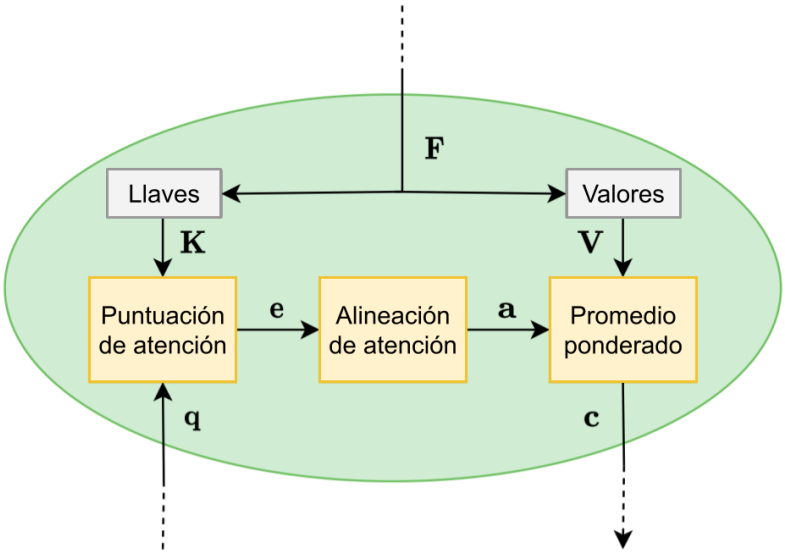

FIGURA 2.7: Mecanismo interno de un módulo de atención [23].

Las ponderaciones utilizadas para producir este resultado se obtienen mediante una puntuación y alineación de la atención [62]. La consulta **q** y la matriz de llaves **K** se utilizan para calcular el vector de puntuaciones de atención $\mathbf{e} = [e_1, e_2, ..., e_{n_f}] \in \mathbb{R}^{n_f}$. Esta puntuación representa la importancia de la información contenida en el vector llave k_l según la consulta. Así, si las dimensiones de la consulta y de los vectores llave son las mismas, un ejemplo de función de puntuación es el producto punto vectorial. Esta puntuación es de la forma

$$e_l = score(\mathbf{q}, k_l) \tag{2.27}$$

y las puntuaciones de atención se procesan a través de una capa de alineación. Las puntuaciones pueden tener generalmente un amplio rango fuera de $[0, 1]$. No obstante, el objetivo es producir una media ponderada, entonces las puntuaciones se redistribuyen a través de una función de alineación $align(\cdot)$ de la forma

$$a_l = align(e_l, \mathbf{e}) \tag{2.28}$$

donde $a_l \in \mathbb{R}$ es el peso de atención correspondiente al l-ésimo vector de valores. Una posible elección de función de alineación es la función *softmax*, debido a que la función está acotada entre $[0, 1]$, es una combinación de funciones exponenciales y polinómicas y es infinitamente diferenciable en todo el conjunto de los numero reales. Así, cada peso es una representación directa de la importancia de cada vector de características en relación con los demás para el problema en particular. Y el vector de pesos de atención $\mathbf{a} = [a_1, a_2, ..., a_{n_f}] \in \mathbb{R}^{n_f}$ se utiliza para producir el vector de contexto $\mathbf{c} \in \mathbb{R}^{d_v}$ calculando

una media ponderada de las columnas de la matriz de valores **V**, a partir de

$$\mathbf{c} = \sum_{l=1}^{n_f} a_l \times v_l. \tag{2.29}$$

Finalmente, el vector de contexto se utiliza para generar la salida $\hat{y}$. Este modelo de salida traduce el vector de contexto en una predicción de salida y esta podría ser una simple capa *softmax* que toma como entrada el vector de contexto **c**, de la forma

$$\hat{y} = softmax(W_c \times \mathbf{c} + \mathbf{b}_c) \tag{2.30}$$

en donde $\hat{y} \in \mathbb{R}^{d_y}$, d_y es el número de clases de salida y tanto $W_c \in \mathbb{R}^{d_y \times d_v}$ como $b_c \in \mathbb{R}^{d_y}$ son parámetros ajustables durante el entrenamiento. Por otra parte, las consultas son un aspecto importante de cualquier modelo de atención, ya que determinan directamente qué información se extrae de los vectores de características. Existen consultas básicas, que son consultas que suelen ser sencillas de definir en función de los datos y el modelo. Algunos mecanismos de atención, como la co-atención, la atención rotatoria y la atención cruzada utilizan consultas especializadas. En particular, se tiene mecanismos que calculan la atención basándose exclusivamente en los vectores de características. Esta idea da lugar a un tipo de atención denominado autoatención o intraatención [101]. Esta técnica, en específico, se utiliza a menudo en el modelo para crear representaciones mejoradas de los vectores de características y ha dado lugar a modelos *Transformer* para el procesamiento del lenguaje [169], y modelos *Transformer* para el procesamiento de imágenes [42], en donde ambas arquitecturas utilizan múltiples módulos de autoatención multi-cabeza para mejorar la representación de los vectores de características. Las relaciones captadas por este mecanismo de autoatención se incorporan a nuevas representaciones.

Cabe mencionar que, en el contexto de visión computacional, los mecanismos de atención se han convertido en una técnica indispensable en las arquitecturas de aprendizaje profundo [62]. De modo que los principales métodos de atención son la atención por canal de la imagen, que genera una máscara de atención en el dominio del canal y la utiliza para seleccionar los canales de mayor relevancia para el objetivo del problema [72, 174, 198, 54]. Esta dimensión de canal puede ser una representación del color de la imagen, tal como *RGB* o *HSV*; la atención espacial, que genera una máscara de atención a través de los dominios espaciales para seleccionar regiones espaciales importantes o predecir directamente la posición espacial más relevante [114, 27, 196, 76]; y la atención temporal [185, 199, 28], que genera una máscara de atención en la dimensión temporal para seleccionar los fotogramas o elementos de una secuencia que son clave para la aplicación.

2.3.3. Codificador *Transformer*

Los mecanismos de atención incorporados en redes neuronales profundas han permitido construir arquitecturas basadas únicamente en estos mecanismos, en particular en la autoatención. Por una parte, se ha demostrado que la transmisión de información a través de una larga serie de conexiones recurrentes en un modelo secuencial conduce a una pérdida de información relevante y, en consecuencia, a dificultades en el entrenamiento. Además, la naturaleza inherentemente secuencial de las redes recurrentes inhibe el uso de recursos computacionales paralelos [83]. Estas consideraciones motivaron al desarrollo de los *Transformers* [169] para el procesamiento de lenguaje natural. Este es un enfoque para el procesamiento de secuencias que elimina las conexiones recurrentes y emplea modelos basados en redes totalmente conectadas. Los *Transformers* entonces mapean secuencias de vectores de entrada $\{x_1, x_2, ..., x_n\}$ a secuencias de vectores de salida $\{y_1, y_2, ..., y_n\}$ de la misma longitud. Estos modelos están formados por pilas de capas de red que consisten en capas lineales simples, redes de propagación hacia adelante y conexiones residuales. Además de estos componentes estándar, la innovación clave de este tipo de módulos es el uso de capas de autoatención, como se muestra en la Figura 2.8. En esencia, la función de atención puede considerarse como un mapeo entre una consulta y un conjunto de pares llave-valor a una salida.

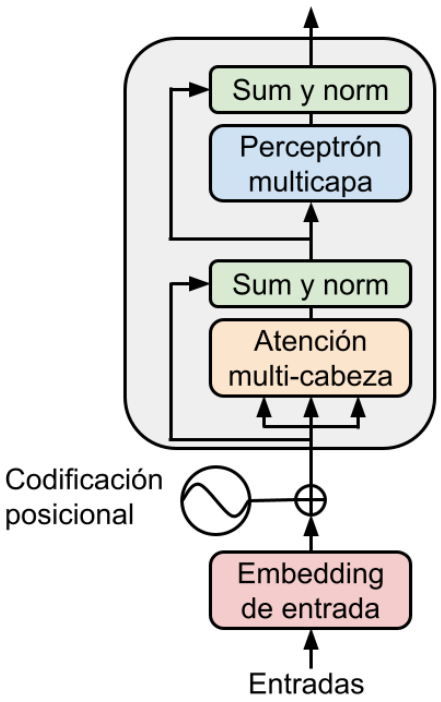

FIGURA 2.8: Codificador tipo *Transformer* [169].

La autoatención permite que una red extraiga y utilice directamente la información de contexto arbitrariamente grande sin necesidad de pasar por conexiones recurrentes intermedias como es el caso en una red neuronal recurrente. Al procesar cada elemento de la entrada, el modelo tiene acceso a todas las entradas hasta la que se está considerando, pero no tiene acceso a información sobre las entradas más allá de la actual. Además, el cálculo realizado para cada elemento es independiente de todos los demás, lo que permite paralelizar fácilmente tanto la inferencia hacia delante como el entrenamiento de dichos modelos [168]. Entonces, en la arquitectura original [169], el vector de entradas $\{x_1, x_2, ..., x_n\}$ se introduce a un módulo de transformación lineal, se agrega una codificación posicional y se ingresan estos vectores resultantes al codificador tipo *Transformer*, como se ilustra en la Figura 2.8. Este último contiene 4 submódulos: un módulo de atención de múltiples cabezas, un módulo de conexión residual y normalización, una capa totalmente conectada y nuevamente una capa de conexión residual y normalización. Más adelante se abordan los detalles concernientes a esta codificación posicional.

La forma más simple de comparación entre dos vectores o tensores en una capa de autoatención es el operador de producto punto. No obstante, el resultado del producto punto puede ser un valor arbitrariamente grande, ya sea positivo o negativo. Esto puede dar lugar a problemas numéricos y a una pérdida efectiva del gradiente durante el entrenamiento. Para evitar esto, el producto punto debe ser escalado de forma adecuada. Un enfoque de producto punto escalado divide el resultado del producto punto por un factor relacionado con el tamaño de las transformaciones de las entradas antes de aplicarles la función *softmax*. En el artículo original [169] Vaswani y col. proponen dividir el producto punto por la raíz cuadrada de la dimensión de los vectores de consulta y llave. De esta manera, se calcula la función de atención sobre un conjunto de consultas simultáneas, representadas en una matriz $Q = W_q \times \mathbf{F}$, y las llaves y valores contenidas en las matrices K y V, definidas por las ecuaciones 2.26. Y, por tanto, la matriz de salida del modelo de autoatención con producto punto escalado, representado en la Figura 2.9, es de la forma

$$attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V. \tag{2.31}$$

El factor de escala $\sqrt{d_k}$ se introduce para contrarrestar el efecto concerniente a que el producto punto crezca en magnitud para valores grandes de la dimensión del tensor de entrada. Además, cuando se aplica la función *softmax* a valores de magnitud muy grandes, el resultado en el gradiente del error puede arrojar valores numéricos muy pequeños. Esto ocasionaría un problema con el desvanecimiento de los gradientes durante el entrenamiento.

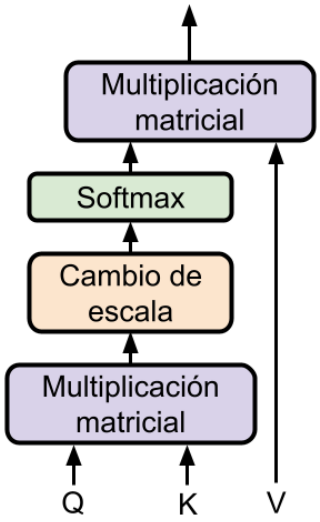

FIGURA 2.9: Atención basada en producto punto escalado [169].

Luego, los distintos elementos en una secuencia pueden relacionarse entre sí de diferentes maneras de forma simultánea. En este sentido, es difícil que un solo bloque *Transformer* aprenda a modelar todos los tipos de relaciones paralelas entre sus entradas. Aun así, los *Transformers* solucionan este problema empleando capas de autoatención de múltiples cabezas. Se trata de conjuntos de capas de autoatención, llamados cabezas, que se apilan en capas paralelas a la misma profundidad en un modelo y cada una con su propio conjunto de parámetros [83]. Dados estos distintos conjuntos de parámetros, cada cabeza puede aprender diferentes aspectos de las relaciones que existen entre las entradas en el mismo nivel de abstracción.

Para aplicar esta noción de múltiples representaciones en la arquitectura, cada cabeza i, en una capa de autoatención está provista de su propio conjunto de matrices de llaves, consultas y valores: W_i^K , W_i^Q y W_i^V . Estas matrices se utilizan para proyectar las entradas de la capa x_i por separado para cada cabeza, mientras que el resto del cálculo de autoatención permanece sin cambios. La salida de una capa de varias cabezas con h cabezas consiste en h vectores de la misma longitud, como se presenta en la Figura 2.10. Para utilizar estos vectores en el procesamiento posterior, se combinan y se reducen a la dimensión de entrada original d_m. Para ello, se concatenan las salidas de cada cabeza y se utiliza otra proyección lineal para reducirlas a la dimensión de salida original

$$MultiHead(Q, K, V) = head_1 \oplus head_2 \oplus ... \oplus head_h \tag{2.32}$$

en donde cada cabeza estima un tensor de autoatención distinto de manera independiente, a partir de la ecuación 2.31, de modo que $head_i = attention(Q, K, V)$. De manera general, el mecanismo de atención de múltiples cabezas proyecta linealmente las consultas, las llaves y los valores h veces, utilizando cada vez una proyección aprendida

diferente. El mecanismo de autoatención se aplica entonces a cada una de estas proyecciones en paralelo, para producir salidas, que a su vez se concatenan y se proyectan de nuevo para producir un resultado final. La idea que subyace a este módulo es permitir que la función de atención extraiga información de diferentes subespacios de representación, lo que, de otro modo, no sería posible con un único bloque de atención.

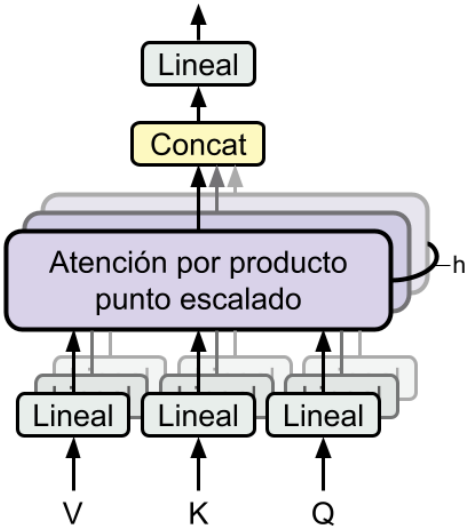

FIGURA 2.10: Bloque de atención multi-cabeza [169].

Por último, la información de salida del bloque de atención multi-cabeza pasa a través de una capa red neuronal completamente conectada, que se aplica a cada posición de la secuencia por separado e idénticamente. Esto significa que la misma red neuronal se utiliza para cada posición de la secuencia, pero cada posición se procesa de manera independiente. La estructura de esta capa en un *Transformer* consta de dos capas lineales con una función de activación no lineal aplicada entre ellas y su propósito es introducir no linealidades en el modelo, lo que es crucial para que el modelo sea capaz de aprender representaciones complejas [51]. Aunque las capas de atención en el Transformer capturan las dependencias globales entre los elementos de una secuencia, la capa totalmente conectada permite al modelo transformar estas representaciones de manera no lineal y mejorar su capacidad de representación.

2.3.4. Decodificador *Transformer*

En general, el codificador *Transformer* se encarga de procesar los vectores de características de una secuencia para extraer características relevantes y crear una representación del contexto. Mientras que el decodificador *Transformer* se utiliza para generar la salida del modelo. Este recibe como entrada la representación contextualizada generada por el codificador, para generar los vectores de llaves y valores, y un vector de consultas con la

información de la secuencia que se está estimando [29]. Este decodificador también se compone de un conjunto de capas de atención y capas de redes totalmente conectadas.

En la arquitectura original presentada por Vaswani y col. [169], el decodificador *Transformer* utiliza la información del contexto del codificador y su propia salida generada previamente para calcular los pesos de atención y generar los elementos de una secuencia de salida. Este proceso se repite hasta que se genera toda la secuencia deseada. Entonces, en resumen, el codificador *Transformer* procesa la entrada para obtener una representación contextualizada, mientras que el decodificador *Transformer* utiliza la representación del contexto del codificador y la salida generada previamente para generar la salida final.

2.3.5. Información posicional

El modelo tipo *Transformer* no contiene bloques de recurrencia ni convolución, entonces, para que el modelo haga uso del orden de la secuencia, se debe incluir alguna información sobre la posición relativa o absoluta de los vectores de características en la secuencia. En el caso de las redes neuronales recurrentes, la información sobre el orden de las entradas está integrada en la naturaleza misma de los modelos. En cambio, esto no ocurre con los *Transformers*, no hay nada que aporte información posicional a estos modelos sobre el orden relativo o absoluto de los elementos de una secuencia de entrada. Esto se puede apreciar a partir del hecho de que, si se revuelve el orden de las entradas en el cálculo de atención ilustrado anteriormente, se obtiene exactamente la misma respuesta. Para resolver este problema, las entradas del *Transformer* se combinan con una codificación posicional específica para cada elemento en una secuencia de entrada. Un enfoque simple y eficaz es comenzar con incrustaciones inicializadas al azar correspondientes a cada posición de entrada posible hasta cierta longitud máxima. En otro caso, es posible ajustar los elementos de la codificación posicional junto con otros parámetros durante el entrenamiento [56]. Para producir una codificación de entrada que capture la información posicional, simplemente se agrega esta codificación del elemento para cada entrada a su correspondiente elemento posicional. Este nuevo vector sirve de entrada para el procesamiento posterior.

Un problema potencial de este enfoque es que existirán más instancias de entrenamiento para las posiciones iniciales en las entradas y menos en los límites de la secuencia, en donde estos últimos elementos pueden estar mal entrenados y no generalizar de manera óptima durante las pruebas [168]. Un enfoque alternativo a la codificación posicional es elegir una función estática que mapee las entradas de números enteros a vectores de valor real de manera que capture las relaciones inherentes entre las posiciones. En el trabajo original de *Transformer* se propone una combinación de funciones seno y coseno con frecuencias diferentes de la forma

$$PE(pos, 2i_d) = sen(\frac{pos}{1000^{2i_d/d_{model}}}) \tag{2.33}$$

$$PE(pos, 2i_d + 1) = cos(\frac{pos}{1000^{2i_d/d_{model}}}) \tag{2.34}$$

en donde pos es la posición, i_d es la dimensión y d_{model} es la dimensión del modelo. Esto es, cada dimensión de la codificación posicional corresponde a una función sinusoidal, asegurando que las posiciones cercanas tengan codificaciones similares mientras se mantienen diferenciables. Otro enfoque es la codificación posicional aprendida, en la cual el modelo entrena parámetros específicos para cada posición durante el proceso de aprendizaje, permitiendo que el modelo se adapte mejor a los patrones específicos del conjunto de datos.

Ahora bien, a lo largo de este capítulo se ha realizado una descripción extensa del marco teórico utilizado en el desarrollo del presente trabajo, proporcionando así una base sólida de conceptos clave, teorías relevantes y estudios previos vinculados a la temática de investigación. De esta forma, las preguntas de investigación han sido formuladas guiadas por el conocimiento adquirido sobre el tema. Con base en esto, es posible continuar con la siguiente etapa del desarrollo de este estudio, donde la metodología y los materiales a ser utilizados en la investigación son expuestos de manera detallada. Así, en el siguiente capítulo se realiza un cambio de enfoque desde el análisis reflexivo de las ideas preexistentes hacia la aplicación sistemática y concreta de estrategias y técnicas para la óptima implementación del modelo para la estimación de los mapas del tensor de difusión. En este nuevo capítulo, cada aspecto del enfoque metodológico es delineado, asegurando que la investigación se lleve a cabo de manera rigurosa.

Capítulo 3

Metodología

En este capítulo se presentan cada una de las fases que constituyen la metodología de desarrollo para el modelo que se propone en el presente trabajo de investigación. Se muestran los requerimientos y especificaciones de diseño, así como las características de la arquitectura del modelo y las etapas principales que lo componen. En este sentido, se aborda el diseño e implementación de un modelo basado en *Transformers* y operadores convolucionales para la estimación de los mapas del tensor de difusión. De la misma manera, se describe la base de información utilizada para la construcción de este modelo, se presenta el procesamiento que se ha realizado para obtener las imágenes de referencia correspondiente a los mapas y el preprocesamiento utilizado para acondicionar esta información para la capa de entrada. Por último, se muestran las métricas empleadas para la validación del desempeño de la arquitectura.

En este sentido, a partir de los objetivos descritos en el primer capítulo, se utiliza el siguiente esquema metodológico y se especifican los resultados tangibles o medibles que se esperan en cada fase del procedimiento.

Etapa 1. Base de datos: en esta primera fase se definen los requerimientos, especificaciones y herramientas necesarias para diseñar e implementar el modelo propuesto, se selecciona la base de datos de imágenes DWI que se utilizará para el entrenamiento de este modelo y se realiza el preprocesamiento necesario a cada elemento de esta base. Por tanto, de esta etapa se debe obtener como resultado una base de datos organizada y estandarizada en el formato y características que requiere la arquitectura del modelo.

Etapa 2. Procesamiento con FMRIB Software Library: Construcción del modelo de referencia utilizando las señales de difusión con un valor de atenuación de $b = 1,000 s/mm^2$ de las bases de datos obtenidas. Se realiza el preprocesamiento necesario y el ajuste del tensor de difusión para obtener los mapas de referencia mediante las herramientas de la librería FSL [182].

Etapa 3. Modelo para la estimación de los mapas del tensor de difusión: en esta fase se realiza el diseño e implementación del modelo propuesto para la estimación

de los parámetros del tensor de difusión. Se delimita cada parte del software en las que se ha dividido el sistema en las fases previas. Se realizan pruebas de entrenamiento y funcionales del modelo.

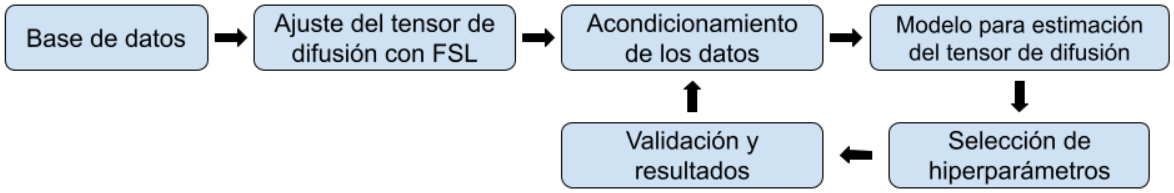

FIGURA 3.1: Diagrama de la metodología empleada para el desarrollo de la investigación.

Etapa 4. Validación y resultados: durante esta fase se detallan los aspectos finales de las pruebas del modelo, se obtienen las métricas de desempeño entre el modelo implementado y el método convencional para validar y evaluar las estimaciones generadas por el modelo propuesto y verificar el cumplimiento de las especificaciones iniciales. La culminación de esta investigación se lleva a cabo al obtener todos los resultados. A partir de esto, se abre la discusión para plantear las conclusiones de la presente investigación.

3.1. Base de datos

Las imágenes DWI utilizadas para la construcción del modelo se obtienen de dos proyectos incluidos en el Laboratorio de Neuroimágenes en la Universidad del Sur de California (USC). El primero es el Proyecto de Conectoma Humano (*Human Connectome Project*, HCP), el cual es un proyecto patrocinado por dos consorcios de institutos de investigación y miembros de los Institutos Nacionales de Salud (NIH) de EUA, para construir y entender una mejor relación entre la conectividad cerebral y su funcionamiento [61]. A partir de este proyecto, se obtienen muestras de 35 sujetos control, es decir, pacientes sin patologías cerebrales, con un rango de edades entre 20 y 89 años. Del mismo modo, se utiliza la base de datos del proyecto Iniciativa de Neuroimágenes de la Enfermedad de Alzheimer (*Alzheimer's Disease Neuroimaging Initiative*, ADNI), el cual es patrocinado por expertos de instituciones públicas y farmacéuticas privadas. De este base de información se obtienen estudios de 30 sujetos control con un rango de edades entre 63 y 87 años, por lo que en total se utilizan las muestras de 65 sujetos control para el entrenamiento y validación de los parámetros del modelo. Asimismo, la distribución de genero en estos dos conjuntos de imágenes es equitativa. Es importante mencionar que los datos de ambas bases de información son de libre acceso al público e investigadores.

El conjunto de imágenes DWI de la base de información ADNI ha sido previamente procesado utilizando el paquete de comandos *MRtrix3* 3.0v [166] y la librería *FMRIB Software Library* 6.0v [78], esto incluye corrección de corrientes inducidas de Eddy, corrección de distorsión EPI, corrección de movimiento, extracción de anillos de Gibbs y corrección de campo de sesgo. Posteriormente, se extrae el tejido no cerebral y se calculan los valores propios del tensor de difusión para obtener las imágenes de anisotropía fraccional, difusividad media y orientación de difusión. La implementación y detalle de este procesamiento se describe en [131]. Es importante mencionar que los archivos de ambas bases de información se encuentran en formato *NIfTI*. A continuación, se describen los detalles de los repositorios utilizados en el desarrollo de la presente investigación.

- Proyecto de Conectoma Humano (HCP), el cual consiste en un proyecto patrocinado por dos consorcios de institutos de investigación y miembros de los Institutos Nacionales de Salud de Estados Unidos. De esta base se obtienen imágenes de 35 sujetos control en un rango de edad entre 20 y 89 años. Estas imágenes DWI se han adquirido en cortes axiales oblicuos y con gradientes de difusión monopolares. El tiempo de eco y los tiempos de difusión se han optimizado para la capa de valor $b = 10,000 s/mm^2$, y se fijaron para para todas las señales de valores b inferiores. Cada caso de difusión se compone de 5 conjuntos de señales para valores del gradiente de difusión de $1,000$, $3,000$, $5,000$ y $10,000$ (s/mm^2). Además, se extrae una imagen no ponderada por difusión $b = 0$ cada 14 volúmenes de imagen, dando lugar a 552 volúmenes en total con dimensiones de $140x140x96$. La dirección de codificación de la fase fue de anterior a posterior (AP) y la adquisición multibanda no se utilizó en este conjunto de datos [2, 61]. Tanto los datos de difusión como los estructurales se proporcionan en espacio nativo del escáner y la no linealidad del gradiente en las imágenes DWI y T1w/T2w han sido corregidas. Las características de esta base de datos definen los parámetros que se utilizan para la construcción y entrenamiento del modelo implementado. La información se contiene en tensores de dimensión $[140, 140, 96, 552]$.

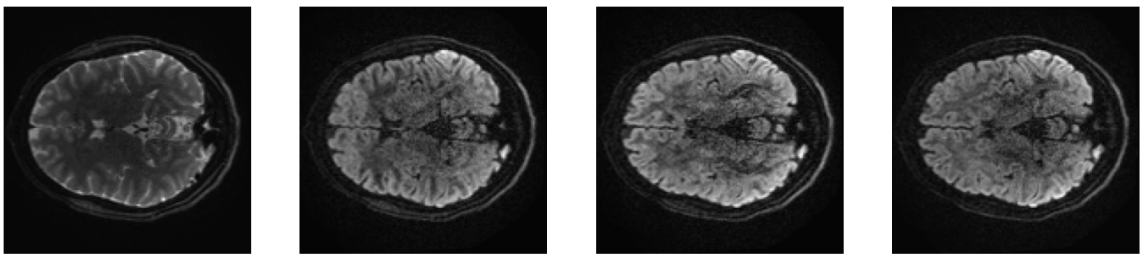

FIGURA 3.2: Imagen sin gradiente de magnetización (izquierda) y 3 imágenes con distintas direcciones del gradiente de magnetización de la base de datos del Proyecto de Conectoma Humano.

En particular, las imágenes de 35 sujetos control proporciona un total de 231,840 rebanadas 2D, lo que representa una amplia gama de datos que pueden ser utilizados para investigar cómo cambia la conectividad cerebral a lo largo del envejecimiento. Esta gran cantidad de imágenes permite un análisis detallado de la evolución de la conectividad en diferentes regiones del cerebro y su relación con el progreso de patologías neurodegenerativas. Además, es importante destacar que los tiempos de difusión se han optimizado específicamente para la capa de valor $b = 10,000 s/mm^2$. Así, al mantener tiempos de difusión fijos para todos los valores de b inferiores, se obtiene una imagen más clara de la integridad de la materia blanca en el cerebro.

- Iniciativa de Neuroimágenes para la Enfermedad de Alzheimer (ADNI), que cuenta con más de 3,000 imágenes cerebrales de IRM, DTI, Tomografía Computarizada y Tomografía por Emisión de Positrones. De este conjunto de información se han recopilado imágenes de 30 sujetos control en un rango de edad entre 63 y 87 años. Se han extraído imágenes en 38 direcciones distintas del vector de magnetización y cada volumen contiene 80 imágenes de $256x256$ pixeles. La información se contiene en tensores de dimensión $[256, 256, 80, 30]$.

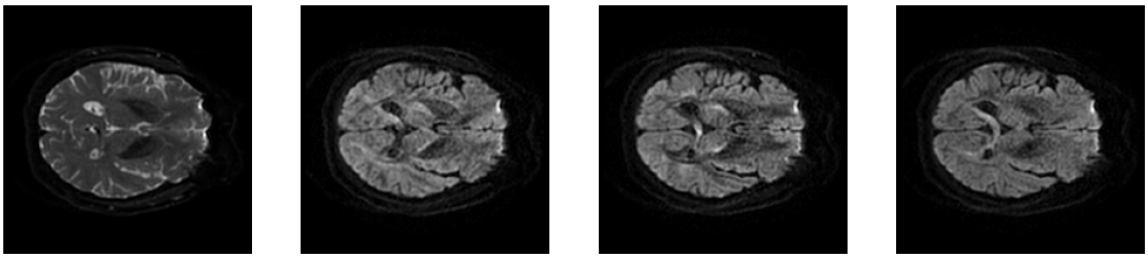

FIGURA 3.3: Imagen sin gradiente de magnetización (izquierda) y 3 imágenes con distintas direcciones del gradiente de magnetización de la base de datos de la Iniciativa de Neuroimágenes para la Enfermedad de Alzheimer.

Esta información ha permitido a los investigadores obtener una comprensión más profunda de los cambios en el cerebro que ocurren durante el desarrollo de la enfermedad de Alzheimer. En particular, las imágenes de 30 sujetos control con 38 imágenes con distintas direcciones del vector de magnetización es una muestra valiosa que puede ser utilizada para comparar con muestras de pacientes con enfermedad de Alzheimer y detectar diferencias en la difusión del agua en el cerebro. La difusión del agua se utiliza como un marcador indirecto de la integridad de la materia blanca en el cerebro, que puede estar afectada en pacientes con enfermedad de Alzheimer. Además, el hecho de que cada volumen contenga 80 imágenes de 256x256 pixeles, con una sensibilidad de atenuación de $b = 1,000 s/mm^2$, proporciona una gran cantidad de datos de imagen que pueden ser procesados y analizados con técnicas de procesamiento de imágenes y aprendizaje automático para detectar patrones y cambios en el cerebro que pueden ser indicativos de esta enfermedad.

3.2. Ajuste del tensor de difusión con FSL

Un procesamiento estándar de las imágenes DWI obtenidas por un escáner de resonancia magnética se realiza utilizando las herramientas de la librería FSL 6.0v [182, 78, 180, 44]. Este procedimiento consiste en extraer el conjunto de imágenes que han sido obtenidas con un único valor de atenuación b, extraer artificios generados por el propio escáner y limpiar las imágenes de estas señales. Finalmente, se ajusta el tensor de difusión mediante un método de mínimos cuadrados con restricciones.

3.2.1. Extracción de imágenes de difusión

Inicialmente se deben extraer las imágenes de las señales para un único valor de gradiente, es decir, para cada valor de b existe un tensor de difusión que se puede ajustar mediante distintas técnicas. Por ello, se toman las primeras 69 señales de difusión correspondiente a las señales con valor de gradiente $b = 1,000 s/mm^2$, debido a las características de las imágenes DWI de las bases de datos que se han obtenido. A continuación, se muestra el comando utilizado.

```
pf=preproc
ini_file=$pf/DWI.mif
# convert DWI images to MIF format
mrconvert -fslgrad bvecs.txt bvals.txt DWI.nii.gz $ini_file -force
# extract shell
mrconvert $ini_file $pf/DWI_single.mif -coord 3 0:1:68 -force
```

3.2.2. Extracción de ruido

Después, para evitar errores en la estimación del tensor, es necesario extraer el ruido gaussiano en las señales de difusión aprovechando la redundancia de datos en el espacio de las componentes principales de los datos, y utilizando el conocimiento a priori de que todo el conjunto de valores propios, o eigenspectrum, de las matrices de covarianza aleatorias está descrito por la distribución universal Marchenko-Pastur [170, 171]. Esto se realiza mediante el comando *dwidenoise* de la librería *MRtrix3* 3.0.4v [166], como se muestra a continuación, en donde se utiliza el algoritmo optimizado presentado en [34].

```
# extract noise from DWI images
dwidenoise $ini_file $pf/DWI_denoised.mif -noise $pf/DWI_noise.mif -force
    -estimator Exp2
```

A partir de las señales finales que se han obtenido y las señales originales es posible calcular el ruido total que se ha eliminado utilizando el algoritmo implementado en la librería antes mencionada. Este paso resulta útil para evaluar visual y cuantitativamente el desempeño del procedimiento.

3.2.3. Corrección de anillos de Gibbs

El efecto de anillos de Gibbs suele aparecer como múltiples líneas finas paralelas inmediatamente adyacentes a las interfaces de alto contraste en imágenes. Los artefactos de Gibbs se producen como consecuencia del uso de la transformación de Fourier para reconstruir las señales del escáner en imágenes de resonancia magnética [16]. Sin embargo, es posible corregir parte de esta pérdida debido a estas transformaciones utilizando el método de desplazamiento local de subvoxeles propuesto por Kellner [87]. Este método está diseñado para trabajar con imágenes adquiridas con una cobertura completa del espacio de adquisición o espacio de radiofrecuencias.

```
# remove gibbs rings
mrdegibbs $pf/DWI_denoised.mif $pf/DWI_no_rgibbs.mif
```

3.2.4. Corrección de movimiento y corrientes de Eddy

Luego, se realiza una corrección de movimiento entre las imágenes y la inducción de corrientes de Eddy en estas señales. Para ello, se convierten los datos de formato *NIfTI* a *mir* y se utiliza el comando *dwifslpreproc* que encapsula algunos de los pasos de procesamiento de datos y metadatos en las estrategias de adquisición de DWI más utilizadas. A continuación, se muestra el uso de este comando.

```
# eddy current and motion correction
dwifslpreproc $pf/DWI_no_rgibbs.mif $pf/DWI_preproc.mif -rpe_none -pe_dir
    $phase_direction -force
```

3.2.5. Estimación del tensor de difusión

Por último, una vez que se ha realizado el preprocesamiento, se convierten los datos nuevamente a formato *NIfTI*, se extrae una máscara binaria para descartar todo el tejido no cerebral de las imágenes y se ajusta el tensor de difusión para cada voxel mediante el comando *dtifit*, como se muestra a continuación.

```
# fit diffusion tensor
dtifit --data=$pf/DATA.nii.gz --out=dti/DTI --mask=$pf/BET.nii.gz_mask.
   nii.gz --bvecs=$pf/bvecs --bvals=$pf/bvals --save_tensor
```

Este ajuste se realiza mediante una transformación logarítmica de los datos originales procesados y se aplica un algoritmo de mínimos cuadrados con restricciones. De esta forma, una vez completado el procesamiento de los datos, se obtiene un volumen que contiene los 6 coeficientes independientes del tensor de difusión y, además, se generan otros mapas correspondientes a distintos parámetros que se pueden obtener a partir de los valores y vectores característicos del tensor, como la anisotropía fraccional, la difusividad media, la orientación principal de la difusión, entre otros.

3.3. Acondicionamiento de datos

De acuerdo con la arquitectura del modelo que se ha implementado, es necesario realizar el acondicionamiento de la información de la base de datos, de modo que sea adecuada para el entrenamiento y validación del modelo. Para empezar, se realiza un reordenamimiento de las dimensiones de los volúmenes de las señales de difusión obtenidas por el escáner de resonancia magnética, los cuales consisten originalmente en un tensor de 4 dimensiones $[h_s, w_s, d_s, c_s]$, en donde h_s es la altura de las imágenes, w_s es el ancho de las imágenes, d_s es la profundidad o rebanadas (*slices*) del volumen y c_s es el número de canales o señales de difusión. Este reordenamiento se efectúa de manera que el tensor resultante tiene las dimensiones $[d_s, c_s, h_s, w_s]$. En los siguientes apartados se describe la selección de las direcciones de las señales c_s de difusión utilizadas en la construcción del modelo y la normalización realizada sobre el conjunto de datos de cada paciente.

3.3.1. Direcciones de difusión

El tensor de difusión tiene solo 6 grados de libertad, por lo que teóricamente es necesario únicamente 6 señales de difusión para calcular los coeficientes del tensor en cada punto o voxel del volumen obtenido por el escáner. Es importante que estas señales capturen la mayor información posible de las direcciones perpendiculares y paralelas a las fibras en la región de interés dentro del cerebro, como se ilustra en la Figura 3.4, tales como las fibras corticoespinales, las fibras de asociación o las fibras comisurales.

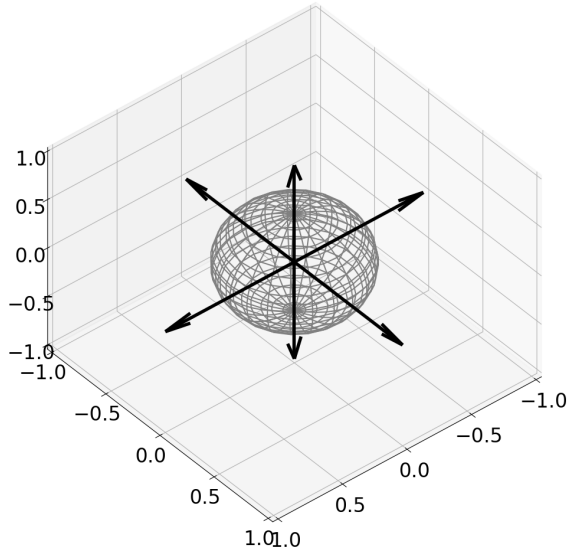

FIGURA 3.4: Direcciones de difusión utilizadas como entrada para el modelo implementado.

El interés en este trabajo de investigación es reconstruir el tensor de difusión en toda la región de materia blanca del cerebro. Por tanto, se deben utilizar el mínimo número de señales de difusión que capturen la información de los tractos neuronales en toda la materia blanca. Entonces, basándose en un sistema de referencia euclidiano de 3 dimensiones, se utilizan las direcciones positiva y negativa a lo largo de estos ejes, como se muestra en la Figura 3.4. De esta forma, considerando la imagen sin dirección de difusión o imagen estructural, se toman un total de 7 señales de difusión para la estimación de los mapas del tensor de difusión, lo cual implica una reducción de casi 10 veces el número de imágenes DWI utilizadas para el cálculo de los mapas del tensor, en el caso de la base de datos del Proyecto de Conectoma Humano; y una reducción de hasta 5 veces el número de imágenes necesarias para la base de datos de la Iniciativa de Neuroimágenes para la Enfermedad de Alzheimer.

3.3.2. Normalización de imágenes

Por otra parte, para evitar que los parámetros del modelo se disparen a valores infinitos es necesario limitar el rango de valores de los pixeles de las imágenes de entrada y de salida mediante una normalización. Este proceso tiene como objetivo estandarizar la intensidad de los pixeles en todas las imágenes de entrada, de tal manera que las imágenes tengan una distribución de intensidad de pixeles similar. Asimismo, esto ayuda a reducir las diferencias en la intensidad de pixeles entre las diferentes imágenes para homogeneizar el conjunto de datos de entrenamiento y de validación. Esto último permite que la red neuronal aprenda patrones de mayor relevancia y reduce el impacto de las variaciones en el contraste e iluminación en las imágenes, lo que puede mejorar la precisión de la segmentación y la clasificación. En específico, se ha utilizado una normalización respecto al máximo valor de intensidad de voxeles del conjunto de datos de cada paciente, lo que estandariza los valores de entrada entre cero y la unidad.

3.3.3. Tratamiento de impurezas

Durante el procesamiento de las imágenes ponderadas por difusión utilizando las herramientas de *FSL*, se ha realizado una extracción de cráneo de la región del cerebro con el comando *BET*, en donde se ha utilizado un valor de 0.2 para el umbral de la segmentación. De este modo, se evita omitir cualquier parte de la región de materia blanca en todo el cerebro. Sin embargo, esto mismo conduce a que se generen datos erróneos o valores atípicos sobre la región de los bordes del cerebro en los distintos mapas del tensor de difusión. Es evidente que estas señales impactan el entrenamiento y el desempeño del modelo durante la optimización. Por tanto, se lleva a cabo un procesamiento extra de los mapas de anisotropía fraccional, difusividad media y orientación de difusión, como se ilustra en las Figuras 3.5 y 3.6. En este procesamiento se realiza una segunda extracción de la región del cerebro, pero esta vez efectuando dicha extracción mediante el método

de segmentación de *Otsu* utilizando la librería *Dipy* [55, 136], la cual es una herramienta para el análisis de datos de difusión de imágenes de resonancia magnética.

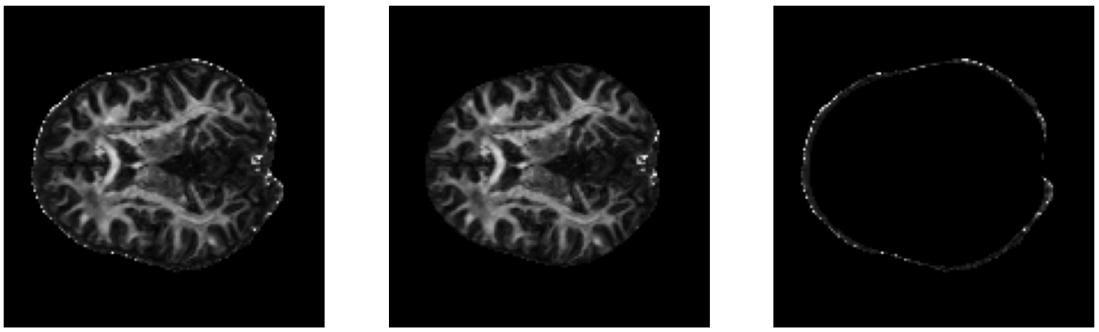

FIGURA 3.5: Tratamiento de valores atípicos en los mapas del tensor de difusión, ejemplo con mapa de anisotropía fraccional. Mapa original (izquierda), mapa sin impurezas (centro) y mapa de impurezas (derecha).

A partir de las primeras 15 señales de difusión de cada caso de los conjuntos de datos, *HCP* y *ADNI*, se genera una máscara binaria del cerebro mediante una umbralización por el método propuesto por *Otsu* [136] y, posteriormente, se aplica esta máscara a los mapas del tensor de difusión obtenidos con la herramienta de *FSL*. De esta forma, se cancelan los valores atípicos que se han producido al ajustar el tensor de difusión en regiones de la imagen que no pertenecen al cerebro o al volumen de materia blanca.

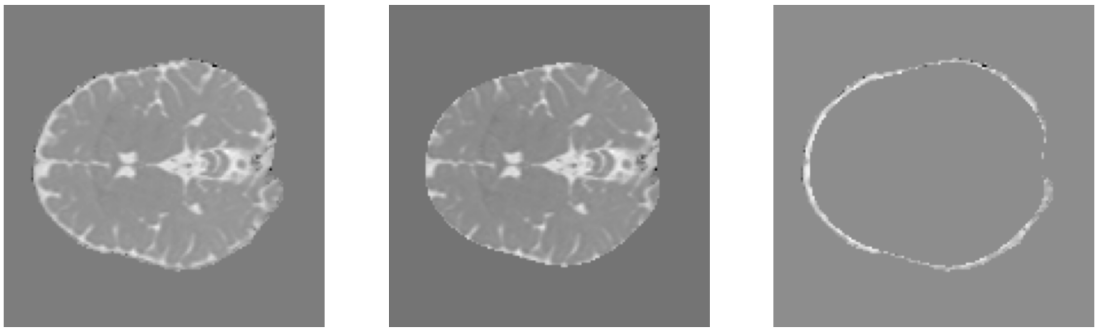

FIGURA 3.6: Tratamiento de valores atípicos en los mapas del tensor de difusión, ejemplo con mapa de difusividad media. Mapa original (izquierda), mapa sin impurezas (centro) y mapa de impurezas (derecha).

La limpieza de errores en los datos antes de entrenar el modelo puede tener un impacto significativo en su rendimiento y precisión. Al eliminar los valores atípicos, los datos se vuelven más homogéneos y coherentes, lo que ayuda al modelo a aprender patrones más relevantes para el objetivo del modelo. Además, los valores atípicos pueden ser ruido o información errónea que perturba al modelo, lo que puede llevar a una menor precisión y una mayor tasa de errores en las predicciones.

3.4. Arquitectura implementada

El diseño del modelo propuesto para la estimación de los mapas del tensor de difusión en la región de materia blanca de imágenes cerebrales se basa en la arquitectura *UT-Net* para segmentación, propuesta por Gao y col. en [53], la cual se compone de bloques convolucionales, módulos *Transformer*, bloques de submuestreo y capas totalmente conectadas, como se muestra en la Figura 3.7. El modelo se ha implementado utilizando la biblioteca de aprendizaje automático *PyTorch* [129], que se basa en la librería *Torch* [33] y es parte de los proyectos de *Linux Foundation*. Este modelo se compone de una parte de contracción o compresión, que se utiliza para la extracción de características y reducción de dimensiones espaciales; y una fase de expansión, empleada para la reconstrucción de las imágenes de referencia a sus dimensiones espaciales originales.

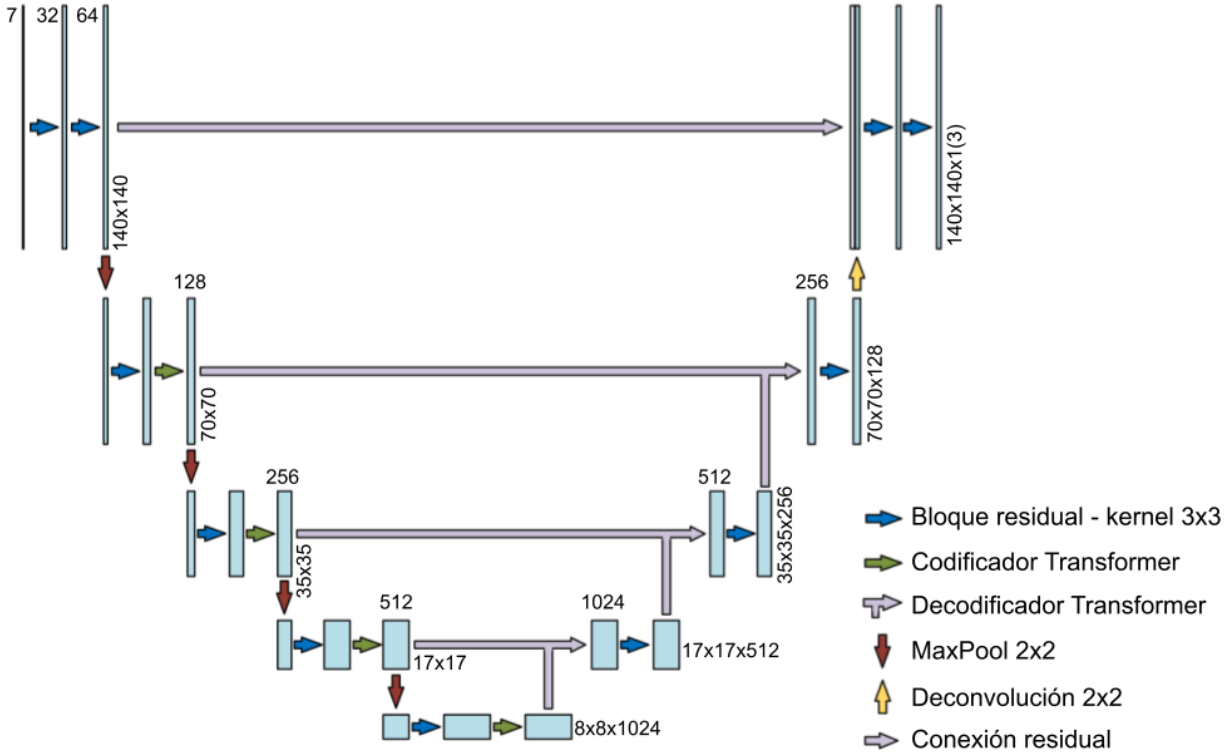

FIGURA 3.7: Arquitectura UT-Net implementada para la estimación de los mapas del tensor de difusión [53].

En la etapa de codificación de las imágenes se encuentran inicialmente dos bloques residuales que extraen las primeras características locales de las señales de difusión y aumentan en un factor de 4 el número de canales de entrada de las imágenes. Posteriormente, se extraen estas últimas características con un submuestreo del máximo valor utilizando una ventana de tamaño 2x2, por lo que las dimensiones espaciales de las

imágenes se reducen en un factor de 4. Posteriormente, se introducen estos tensores de características en el primer codificador tipo *Transformer*, que toma las imágenes de todos los canales y las proyecta sobre una dimensión, obteniendo así una secuencia de características que capturan las principales características de las imágenes. Se repite este proceso desde el submuestreo hasta el codificador *Transformer* por 3 niveles. En este punto, las dimensiones espaciales de las imágenes se han reducido en un factor de 16, mientras que con el mismo factor ha aumentado el número de canales o características. Entonces, a partir del espacio latente a la salida de la etapa de codificación del modelo se comienza la reconstrucción de las imágenes de salida utilizando un decodificador tipo *Transformer*, el cual toma dos tensores de entrada, correspondientes a las características del espacio latente de la etapa de contracción y las características de mayor resolución mediante una conexión de residual para proyectar el conjunto de características de las imágenes aumentando de dimensión espacial. De esta forma, se utilizan 3 decodificadores tipo *Transformer* y finalmente se utiliza un par de bloques residuales para reducir el número de canales de las imágenes de salida al número de canales necesarios para cada mapa.

Es importante considerar que el coste de procesamiento de un *Transformer* es exponencial con el tamaño de la secuencia y, en el caso de las imágenes, la unidad básica de análisis es el pixel, por lo que calcular las relaciones de cada par de pixeles en una imagen resulta costoso computacionalmente. Por esta razón, se han utilizado operadores convolucionales para la extracción de características locales y bloques *Transformer* para modelar dependencias de largo alcance entre las características de los pixeles de cada imagen. Por tanto, en lugar de integrar el módulo de autoatención sobre los mapas de características que se han extraído de manera automática o manual como se plantea en la arquitectura *ViT* [42] y *TransUNet* [29], se aplica el módulo *Transformer* a cada nivel del codificador y decodificador para modelar la dependencia de largo alcance en múltiples escalas. Sin embargo, se ha omitido un bloque *Transformer* en la capa de entrada, puesto que añadir uno de estos módulos en las capas poco profundas de la red no genera una mejora significativa en el desempeño [53], únicamente introduce cómputo adicional. Una posible razón es que las capas poco profundas de la red se centran más en las texturas detalladas, donde la recopilación del contexto global puede no ser informativa y los bloques convolucionales se encargan de procesar estas características espaciales. En las siguientes secciones se describen a detalle cada uno de los módulos principales utilizados en la implementación de la arquitectura propuesta para este problema.

3.4.1. Bloque residual

Un problema importante en las arquitecturas de redes neuronales profundas es la degradación o desvanecimiento del gradiente durante el entrenamiento, es decir, un optimizador puede tener dificultades para aproximarse a los estimadores de identidad mediante múltiples capas no lineales. Entonces, para solucionar este problema se utilizan bloques residuales que conectan la información de entrada de una secuencia de bloques

de normalización y operadores convolucionales con su salida, como se ilustra en la Figura 3.8. En este caso, cuando un mapeo de identidad es óptimo, el optimizador puede simplemente obligar que los pesos de las capas no lineales múltiples se vuelvan cero para aproximarse a un mapeo de identidad ideal.

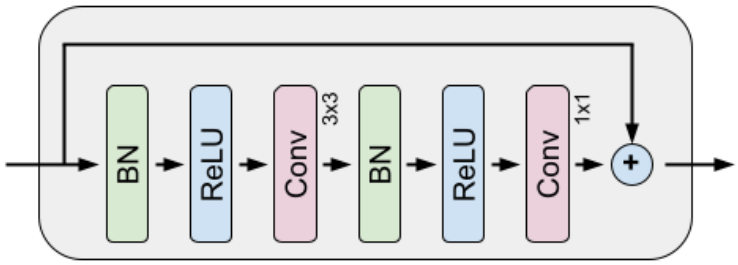

FIGURA 3.8: Bloque residual con operadores convolucionales.

En la práctica, es poco probable que las relaciones de identidad sean óptimas. Sin embargo, si la función óptima está más cerca de un mapeo de identidad que de un mapeo cero, deberá ser más fácil para el optimizador encontrar las perturbaciones con referencia a un mapeo de identidad, que aprender la función como una combinación no lineal de mayor complejidad [69]. Formalmente, se considera un bloque residual de construcción como

$$y_{br} = F(x_f, W_i) + x_f \tag{3.1}$$

en donde x_f e y_{br} son los vectores de entrada y salida de las capas consideradas. La función $F(x_f, W_i)$ representa el mapeo residual que hay que aprender. Esta última se puede componer de distintos bloques con operadores convolucionales y funciones de activación no lineales, como se muestra en la Figura 3.8. La operación $F + x_f$ se realiza mediante una conexión de acceso directo y una suma de elementos. Estas conexiones de acceso directo no introducen ningún parámetro adicional ni complejidad de cálculo.

3.4.2. Atención eficiente

Dado que las imágenes son datos muy estructurados, la mayoría de los pixeles de los mapas de características de alta resolución dentro de una ventana local comparten características similares, excepto en el caso de las regiones limitantes. Por lo tanto, el cálculo de la atención por pares entre todos los pixeles es ineficiente y redundante. Desde una perspectiva teórica, la autoatención es esencialmente de corto alcance para las secuencias largas [174], lo que indica que la mayor parte de la información se concentra en los valores singulares más grandes. Por tanto, se utilizan los bloques de atención multi-cabeza eficientes, propuesto en [153], los cuales se basan en la idea de utilizar dos transformaciones para proyectar los vectores de llave y valor $K, V \in \mathbb{R}^{n \times d}$ en una codificación de menor dimensión $\overline{K}, \overline{V} \in \mathbb{R}^{k \times d}$, en donde $k = hw \ll n, h$ y w es el tamaño

espacial reducido del mapa de características después del submuestreo. Entonces, la autoatención eficiente presentada por Gao en [53] se formula como

$$EfficientAttention(Q, \overline{K}, \overline{V}) = softmax(\frac{Q\overline{K}}{\sqrt{d}})\overline{V} \tag{3.2}$$

De esta manera, la complejidad computacional del cálculo de los mapas de atención se reduce a $O(nkd)$. En particular, la proyección a una menor dimensión puede ser mediante cualquier operación de muestreo descendente, como la agrupación por promedio o valor máximo o utilizar un operador convolucional. En este caso, se emplea una convolución con tamaño de kernel 1x1 seguida de una interpolación bilineal para reducir la muestra del mapa de características. Este módulo se integra en la etapa de codificación como se muestra en la Figura 3.9. En el codificador, la autoatención se utiliza para capturar las relaciones dentro de la secuencia de entrada. Mientras que, en el decodificador, la autoatención también se utiliza para capturar las relaciones entre la posición actual en el decodificador y las posiciones anteriores en la secuencia de salida generada.

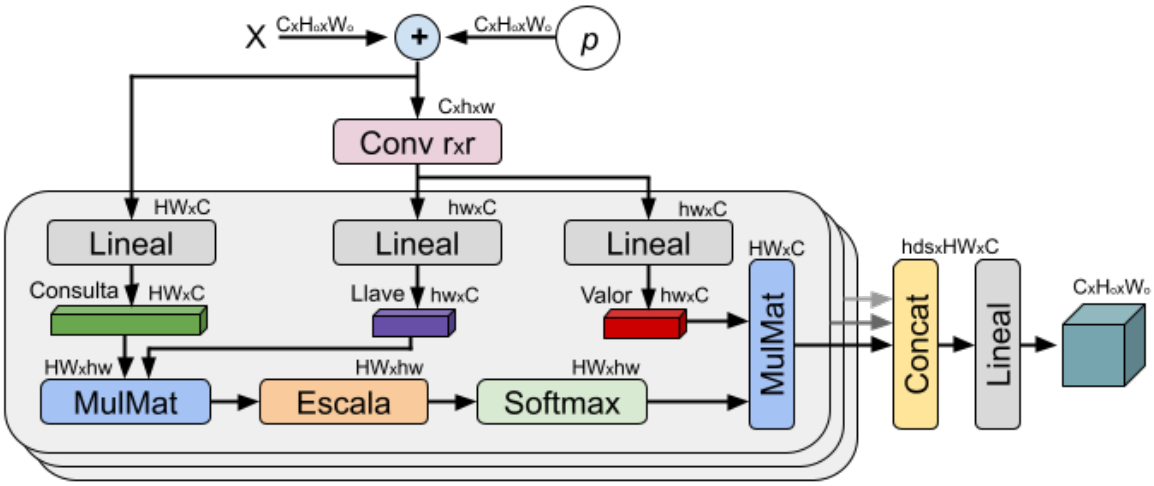

FIGURA 3.9: Bloque *E-MHSA* de multiatención eficiente para codificador *Transformer* [53].

Para la fase de decodificación se utiliza el módulo mostrado en la Figura 3.10, en donde se integran las características de alta resolución, obtenidas en la parte de contracción, como consultas del decodificador con valores y llaves obtenidas con características de baja resolución. Este decodificador recibe una secuencia de entrada y, además, utiliza información adicional proveniente de la salida del codificador para generar los siguientes elementos de una secuencia. En particular, el codificador y el decodificador *Transformer* tienen estructuras similares pero con algunas diferencias clave en su funcionamiento. El codificador se centra en capturar la información contextual de la secuencia de entrada,

mientras que el decodificador utiliza tanto la información contextual del codificador como la información generada previamente para generar una secuencia de salida en cada nivel de la arquitectura.

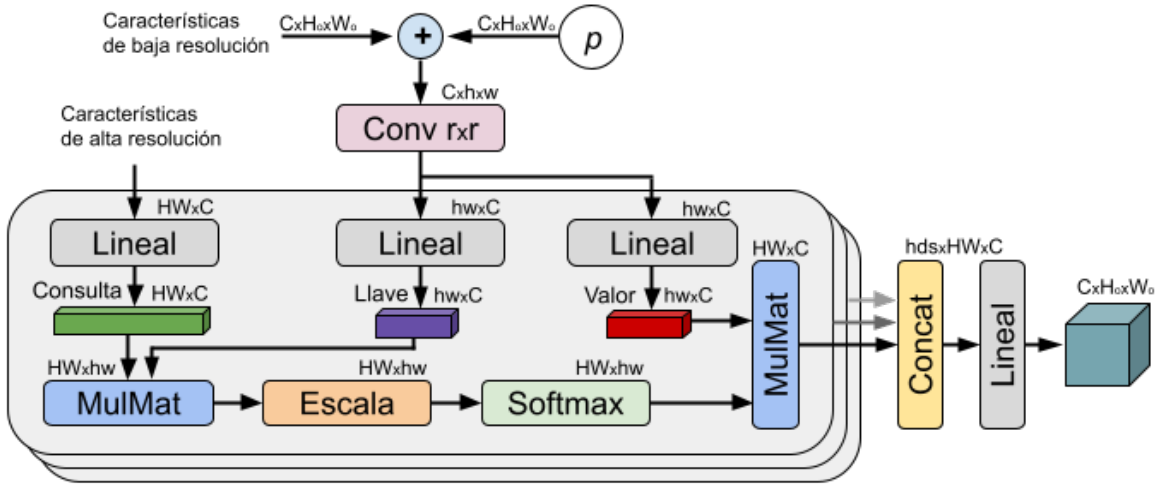

FIGURA 3.10: Bloque *E-MHSA* de multiatención eficiente para decodificador *Transformer* [53].

En el siguiente apartado se presenta la implementación del módulo de multiatención eficiente, el cual es utilizado tanto para el módulo *Transformer* codificador como el módulo decodificador. La clase *EfficientMultiHeadAttention* es una implementación de una capa de atención eficiente de múltiples cabezas empleada para capturar las relaciones entre diferentes partes de una secuencia o mapa de características.

```
class EfficientMultiHeadAttention(nn.Module):
    def __init__(self, embed_dim: int, num_heads: int = 8,
                 reduction_ratio: int = 1, reduced_size: int = 28):
        super().__init__()
        # positional information
        self.pos_embedding = nn.Parameter(torch.randn(embed_dim,
                                    reduced_size, reduced_size)*0.02)
        # reducer convolution
        self.reducer = nn.Sequential(
            nn.Conv2d(
                embed_dim, embed_dim, kernel_size=reduction_ratio,
                stride=reduction_ratio
            ),
            LayerNorm2d(embed_dim),
        )
        # multihead attention module
        self.att = nn.MultiheadAttention(
            embed_dim, num_heads=num_heads, batch_first=False
        )
```

```
20
21      def forward(self, x, q=None):
22          _, h, w = x.shape if q is None else q.shape
23          # add position embedding
24          x += self.pos_embedding
25          # reduce dimensions
26          reduced_x = self.reducer(x)
27          # attention needs tensor of shape (sequence_length, embed_dim)
28          reduced_x = rearrange(reduced_x, "c h w -> (h w) c")
29          x = rearrange(x, "c h w -> (h w) c")
30          # encoder-decoder condition
31          query = x if q is None else rearrange(q, "c h w -> (h w) c")
32          # get attention map
33          attn = self.att(query, reduced_x, reduced_x)
34          # reshape it back to (batch, embed_dim, height, width)
35          out = rearrange(attn[0], "(h w) c -> c h w", h=h, w=w)
36          return out
```

La función de procesamiento hacia adelante, o capa totalmente conectada, toma como argumentos dos vectores de características x y q, en donde el segundo es el vector de características del cual se proyectan las consultas al mapa de atención. En el caso del codificador, como se ha comentado, estas consultas se extraen del mismo vector x para modelar la autoatención. Es importante mencionar que se ha integrado el módulo *MultiheadAttention* de la librería de *PyTorch* para calcular los mapas de atención. Este último recibe como entrada una secuencia de elementos de dimensión fija.

3.4.3. Codificador *Transformer*

El bloque codificador tipo *Transformer* se ha implementado en base al artículo original de la arquitectura *UT-Net* propuesta por Gao en [53] y el módulo presentado por Xie en [184], en donde se han utilizado estos modelos para tareas de segmentación de imágenes médicas con resultados positivos. En este caso, para el modelo implementado en este trabajo de investigación las señales de entrada se normalizan y se introducen a un bloque de multiatención eficiente presentado en el apartado anterior, como se muestra en la Figura 3.11. Luego, se agrega una conexión residual con los tensores de entrada del bloque *Transformer* completo. El tensor resultante de la capa anterior se introduce a una capa de normalización y, a continuación, se procesa mediante un bloque con múltiples capas convolucionales y una función de activación no lineal, como presenta Xie en [184]. En la Figura 3.12 se ilustra un diagrama de la implementación de este bloque. Finalmente, para reducir la longitud efectiva del modelo durante el entrenamiento, se omiten aleatoriamente capas por completo mediante una capa de selección de profundidad estocástica.

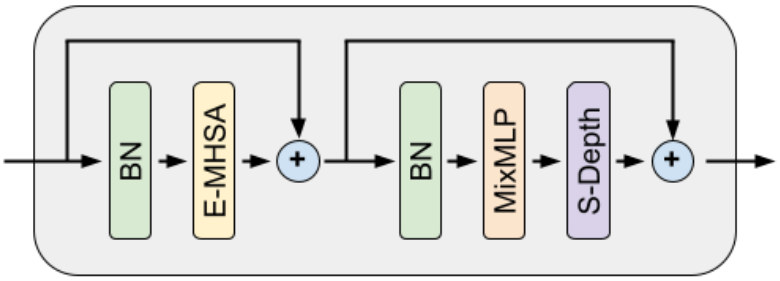

FIGURA 3.11: Codificador tipo *Transformer* para arquitectura implementada.

Esta última capa introduce conexiones de omisión de la misma manera que la arquitectura *ResNet* [69], pero el patrón de conexión se altera aleatoriamente para cada canal de las imágenes, que en este caso son las distintas direcciones del gradiente de magnetización. Se seleccionan aleatoriamente conjuntos de capas y se eliminan sus correspondientes funciones de transformación, manteniendo únicamente la conexión residual de identidad [74]. Este módulo tiene como objetivo reducir la profundidad de la red durante el entrenamiento, mientras que se mantiene inalterada durante la validación.

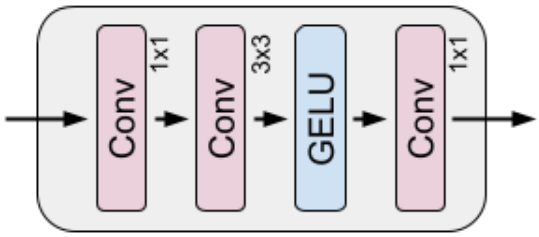

FIGURA 3.12: Bloque *MixMLP* con operadores convolucionales para clasificación en el codificador *Transformer* [184].

A continuación se presenta la implementación en *Python* que se ha elaborado del bloque tipo *Transformer* para la etapa de codificación en el desarrollo de la arquitectura implementada.

```
class TransformerEncoder(nn.Module):
        def __init__(self, dim, n_heads, red_ratio, mlp_expansion,
    depth_prob=0., img_size=28):
        super().__init__()
        self.transformer = nn.Sequential(
            # multi-head self attention
            ResidualAdd(
                nn.Sequential(
                    LayerNorm2d(dim),
                    EfficientMultiHeadAttention(dim, n_heads, red_ratio,
    img_size),
                )
            ),
```

```
            # feed-forward layer
            ResidualAdd(
                nn.Sequential(
                    LayerNorm2d(dim),
                    MixMLP(dim, expansion=mlp_expansion),
                    StochasticDepth(p=depth_prob, mode="batch")
                )
            )
        )
    def forward(self, x):
        # forward pass
        out = self.transformer(x)
        return out
```

3.4.4. Decodificador *Transformer*

En la etapa de expansión o decodificación de la arquitectura se utiliza en esencia un codificador *Transformer* pero que toma dos tensores de características de distintas dimensiones como entrada. Entonces, al igual que en el bloque *Transformer* para la etapa de contracción, se introducen los tensores de características de baja resolución a una capa de normalización. Después, se introduce el último resultado a un bloque de multiatención eficiente, como se muestra en la Figura 3.13, pero en este caso se utiliza este tensor para proyectar los tensores de llaves y valores K y V, mientras que las consultas correspondientes al tensor de consultas Q se proyectan a partir de las características de alta resolución obtenidas por el decodificador que se encuentra al mismo nivel de dimensiones espaciales en la arquitectura tipo U.

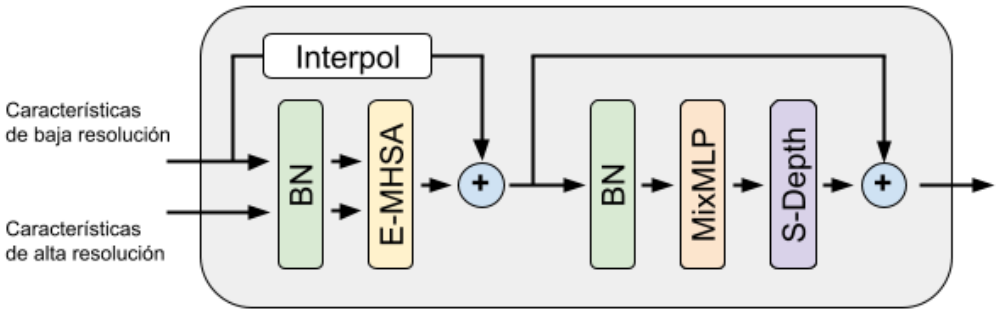

FIGURA 3.13: Decodificador *Transformer* para arquitectura implementada.

Con este bloque decodificador tipo *Transformer* se combinan las características de baja y alta resolución para incrementar las dimensiones espaciales a las dimensiones de alta resolución. En cambio, para sumar los tensores de alta y baja resolución es necesario realizar una interpolación sobre las características de baja resolución para igualar las dimensiones de ambos tensores. De la misma manera, se agrega una conexión directa

del último resultado con el tensor de características de entrada del bloque completo. Es importante notar que para que esta suma de elementos se realice de manera correcta el tensor de características de baja resolución se debe ajustar de forma que exista una concordancia con las dimensiones espaciales del vector de características de mayor resolución. Posteriormente, este resultado se introduce a un bloque residual en donde la función residual consiste en una capa de normalización, una capa de activación no lineal y una capa convolucional. La implementación elaborada para el bloque *Transformer* de la etapa de expansión se muestra a continuación.

```
class TransformerDecoder(nn.Module):
    def __init__(self, low_res_ch, high_res_ch, n_heads, reduction_ratio,
    mlp_expansion, depth_prob=0., img_size=28):
        super().__init__()
        self.embed = nn.Conv2d(low_res_ch, high_res_ch, kernel_size=1)
        # attention decoder
        self.bn_l = LayerNorm2d(low_res_ch)
        self.bn_h = LayerNorm2d(high_res_ch)
        self.attn = EfficientMultiHeadAttention(high_res_ch, n_heads,
                                        reduction_ratio, img_size)
        # residual addition and mixMLP
        self.mlp = nn.Sequential(
            ResidualAdd(
                nn.Sequential(
                    LayerNorm2d(high_res_ch),
                    MixMLP(high_res_ch, expansion=mlp_expansion),
                    StochasticDepth(p=depth_prob, mode="batch")
                )
            )
        )
    def forward(self, x1, x2):
        #x1: low-res, x2: high-res
        x1 = self.bn_l(x1)
        x2 = self.bn_h(x2)
        # embed dimensions
        x1 = self.embed(x1)
        # calculate residue
        x1_shape = x1.shape
        res = x1.expand(1,x1_shape[0],x1_shape[1],x1_shape[2])
        res = F.interpolate(res, size=x2.shape[-2:],
            mode='bilinear', align_corners=True)[0]
        # attention decoder
        out = self.attn(x1, x2)
        # add residue
        out = out + res
        # multilayer perceptron layer
        out = self.mlp(out)
        return out
```

En particular, esta clase depende de los canales de las imágenes, el número de cabezas que deben contener los bloques de multiatención, el umbral de omisión para el mapa de atención y de la proyección lineal, así como del tamaño y tipo de interpolación que se realiza en el bloque de multiatención eficiente.

3.4.5. Hiperparámetros del modelo

El modelo completo para la estimación de los mapas DTI, al integrar todos los módulos descritos en los apartados anteriores, cuenta con millones de parámetros ajustables durante el entrenamiento del modelo. En cambio, la sintonización de los hiperparámetros del modelo consiste en encontrar los valores óptimos para los parámetros que no se aprenden durante el entrenamiento de la red, como la tasa de aprendizaje, el tamaño del lote, la regularización, el número de capas, el número de neuronas por capa, entre otros. Estos parámetros pueden tener un gran impacto en el rendimiento de la red neuronal y encontrar los valores adecuados puede mejorar la precisión de la red. Por ello, es importante conocer la lista de los parámetros que se pueden ajustar de acuerdo con la arquitectura que se ha implementado. En la Tabla 3.1 se describen estos parámetros.

Hiperparámetro	**Descripción**
in_chans	Número de canales de entrada o número de señales de difusión
out_chans	Número de canales de salida de la red
img_size	Tamaño espacial de imágenes
embed_dim	Dimensión inicial del espacio latente en cada etapa del codificador-decodificador
n_heads	Número de cabezas de autoatención en los bloques Transformers
mlp_ratio	Relación de incremento de dimensionalidad en el perceptrón multicapa de cada bloque tipo Transformer.
reduction_ratio	Relación de reducción de dimensionalidad en el bloque de autoatención eficiente.
depth_prob	Probabilidad de dropout en la profundidad de los tensores de características
tanh_output	Uso de función de activación $Tanh()$ en la última capa de la arquitectura

TABLA 3.1: Hiperparámetros del modelo implementado.

En la etapa de entrenamiento y validación se deben probar diferentes combinaciones de estos valores para evaluar el rendimiento de la red neuronal. El objetivo es encontrar

los valores de los hiperparámetros que maximicen el rendimiento del modelo. Esto se puede hacer de forma empírica, probando diferentes combinaciones de valores, o utilizando técnicas de optimización automáticas, como la búsqueda en cuadrícula, la búsqueda aleatoria o la optimización bayesiana. En este caso, se realiza un ajuste empírico.

3.5. Métricas de validación

En el campo de visión computacional y aprendizaje automático, la mayoría de las técnicas de evaluación en las diferencias de la imagen se basan en la cuantificación de los errores entre una imagen de referencia y una de muestra [155]. Una métrica habitual consiste en cuantificar la diferencia absoluta de los valores de cada uno de los pixeles correspondientes entre la imagen de muestra y la de referencia utilizando el error cuadrático medio y la relación de similitud entre los objetos de las imágenes. Entonces, para validar el desempeño del modelo implementado se utilizan las siguientes medidas.

3.5.1. Error cuadrático medio normalizado

En modelación estadística, una manera común de medir la calidad en la aproximación de un modelo es la desviación cuadrática media (MSE), dada por la ecuación

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.3}$$

en donde y_i es la observación i-ésima del conjunto de referencia e $\hat{y}_i$ es el valor de y_i que ha aproximado el modelo [77]. Este indicador es bastante intuitivo, puesto que, si los valores predichos por el modelo están muy cerca de las soluciones reales, el error cuadrático medio será un valor pequeño. Si las respuestas predichas y las verdaderas difieren sustancialmente, al menos para algunas observaciones, el valor de MSE será grande. En un caso ideal, un valor de cero indicaría un ajuste perfecto de los datos. Dado que el MSE se mide en la misma escala, con las mismas unidades que y, se puede esperar que el 68 % de los valores de y estén dentro de un valor unitario, siempre y cuando los datos se distribuyen normalmente [14]. Por otra parte, la normalización del MSE facilita la comparación entre conjuntos de datos o modelos con diferentes escalas. Entonces, existen diferentes maneras de normalizar el error cuadrático medio, esto puede ser utilizando la media, la desviación estándar, la diferencia entre percentiles o la diferencia máxima entre las muestras [119]. En este trabajo de investigación se utiliza la media de la muestra y para la normalización de esta métrica, de modo que la ecuación

$$NMSE = \frac{MSE}{\overline{y}} \tag{3.4}$$

es una medida utilizada para evaluar el desempeño de la arquitectura implementada.

3.5.2. Índice de similitud estructural

La medida del índice de similitud estructural (SSIM) es un método para predecir la calidad percibida de las imágenes digitales. En concreto, este índice es una métrica perceptiva que cuantifica la degradación de la calidad de la imagen causada por el procesamiento, como la compresión de datos, o por las pérdidas en la transmisión de información. En este sentido, el índice de similitud estructural se basa en la comparación de tres medidas que caracterizan un objeto en una imagen: la luminancia, el contraste y la correlación estadística de los pixeles o correlación estructural. Así, la luminancia de la superficie de un objeto observado es el producto de la iluminación y la reflectancia, pero las estructuras de los objetos en una escena son independientes de la iluminación. Por consiguiente, para explorar la información estructural de una imagen, es importante separar la influencia de la iluminación [177]. Entonces, se define la información estructural de una imagen como los atributos que representan la estructura de los objetos de la escena, independientemente de la luminancia y el contraste.

Siguiendo este procedimiento, sean las ventanas de tamaño NxN de una imagen de referencia $\mathbf{x}$ y una imagen a validar $\mathbf{y}$, primero se compara la luminancia de cada señal y, considerando que las señales son discretas, se estima como la intensidad media de los pixeles de la imagen

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{3.5}$$

De la misma forma, se utiliza la desviación estándar como una estimación del contraste de la señal. Una estimación sin sesgo en forma discreta es de la forma

$$\sigma_x = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2} \tag{3.6}$$

Y, de esta manera, la comparación de luminancia se define como

$$l(\mathbf{x}, \mathbf{y}) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{3.7}$$

en donde la constante C_1 se incluye para evitar inestabilidad cuando la suma $\mu_x^2 + \mu_y^2$ es muy cercana a cero. Por lo que se selecciona como $C_1 = (K_1 L)^2$, donde L es el rango dinámico de los valores de los pixeles de las imágenes y $K_1 \ll 1$ es una constante pequeña.

Para la comparación de contraste se utiliza la siguiente expresión

$$c(\mathbf{x}, \mathbf{y}) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{3.8}$$

en donde $C_2 = (K_2L)^2$ y $K_2 \ll 1$. Luego, la comparación de estructura se realiza tras la sustracción de la luminancia y la normalización de la varianza. En concreto, se relacionan los dos vectores unitarios $(\mathbf{x} - \mu_x)/\sigma_x$ y $(\mathbf{y} - \mu_y)/\sigma_y$. El producto interno entre ellas es una medida eficiente para cuantificar la similitud estructural, por lo que se define como

$$s(\mathbf{x}, \mathbf{y}) = \frac{\sigma_x y + C_3}{\sigma_x\sigma_y + C_3} \tag{3.9}$$

y, al igual que en la comparación de luminancia y contraste, se agrega una pequeña constante $C_3 = (K_3L)^2$ con $K_3 \ll 1$. Además, la correlación entre las dos imágenes se obtiene a partir de

$$\sigma_x y = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y) \tag{3.10}$$

Entonces, combinando estas tres comparaciones, se obtiene el índice de similitud estructural entre las señales $\mathbf{x}$ y $\mathbf{y}$

$$SSIM(\mathbf{x}, \mathbf{y}) = l(\mathbf{x}, \mathbf{y})^{\alpha} c(\mathbf{x}, \mathbf{y})^{\beta} s(\mathbf{x}, \mathbf{y})^{\gamma} \tag{3.11}$$

en donde $\alpha > 0$, $\beta > 0$ y $\gamma > 0$ son parámetros utilizados para ajustar la importancia relativa entre estos tres componentes. De forma que, cuando se tiene que $\alpha = \beta = \gamma = 1$ y $C_3 = C_2/2$, se obtiene el siguiente índice de similitud estructural

$$SSIM(\mathbf{x}, \mathbf{y}) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_x y + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{3.12}$$

3.5.3. Proporción máxima de señal a ruido

La Proporción Máxima de Señal a Ruido (PSNR) es una métrica utilizada en el procesamiento de imágenes y compresión de video para evaluar la calidad de una imagen o secuencia de video comprimida en comparación con la versión original sin comprimir. La PSNR se basa en la comparación de la señal original con el ruido introducido durante el proceso de compresión. En casos donde las imágenes contengan objetos fijos y tipos de distorsión típicos de las aplicaciones de comunicaciones visuales, la métrica PSNR puede funcionar en algunos casos incluso mejor que los modelos de calidad objetiva más complejos conocidos. Cuanto mayor sea el valor de PSNR, mayor será la calidad percibida de la imagen o el video comprimido [92]. Se expresa en decibeles (dB) y se

calcula mediante la siguiente expresión

$$PSNR = 20\log_{10}(\frac{MAX(y)}{MSE}) \tag{3.13}$$

en donde $MAX(y)$ es el valor máximo posible de los pixeles de la imagen de referencia, por ejemplo, para imágenes en escala de grises sería $MAX(y) = 255$ cuando se utilizan pixeles de 8 bits; y MSE es el error cuadrático medio, que representa la diferencia promedio al cuadrado entre los valores de los pixeles de ambas imágenes. Además, es importante tener en cuenta que la medida PSNR no siempre se relaciona de manera precisa con la calidad visual percibida, ya que no tiene en cuenta las características específicas de la visión humana, como la sensibilidad al color y al contraste [70]. En resumen, la PSNR es una métrica comúnmente utilizada en el procesamiento de imágenes y compresión de video para cuantificar la calidad de la imagen original en relación con la versión descomprimida o generada artificialmente.

En este capítulo, se ha detallado la metodología y las técnicas utilizadas para el desarrollo de esta investigación, estableciendo un marco sólido sobre el cual se ha construido todo el estudio. Se ha proporcionado una descripción exhaustiva de la selección de la base de información empleada para la optimización de la arquitectura implementada, así como la construcción del modelo de referencia y las técnicas utilizadas para el diseño del modelo, asegurando así la replicabilidad y validez de los resultados obtenidos. Ahora, en el siguiente capítulo, se presentan y analizan los avances que se han realizado en cada etapa metodológica, se validan las hipótesis planteadas y se facilita una discusión sobre los resultados para la generación de una perspectiva más amplia sobre el tema.

Capítulo 4

Implementación y Resultados

En este capítulo se exponen los aspectos relacionados a la implementación de la arquitectura propuesta para la estimación de los mapas del tensor de difusión. Asimismo, se muestran y desarrollan los resultados alcanzados tanto en el procesamiento como en el ajuste del tensor de difusión utilizando FSL. Estos resultados han sido un punto clave en el desempeño de la arquitectura, pues han servido como punto de referencia en la construcción del modelo implementado en el marco de la presente investigación. Asimismo, se realiza una descripción de los parámetros empleados en el modelo final, los cuales han sido finamente ajustados para maximizar la precisión del modelo. Finalmente, se brindan ejemplos visuales que ilustran las capacidades de la arquitectura en la estimación de los mapas del tensor de difusión obtenidos mediante este tipo de tecnología y se analiza la robustez de este ante imágenes de pacientes con patologías.

4.1. Detalles de implementación

Cuando se aborda el problema de trabajar con modelos de redes neuronales profundas de alta complejidad, lo que implica optimizar una vasta cantidad de parámetros, es importante tomar en consideración los recursos disponibles tanto a nivel de software como de hardware con los que se cuenta para la construcción de la arquitectura. Por consiguiente, en este apartado se ofrece una explicación a detalle del procedimiento efectuado para la configuración del entorno de entrenamiento y validación de la arquitectura previamente implementada en el capítulo anterior. En este sentido, se describen las herramientas de software empleadas, incluyendo las bibliotecas y los entornos de trabajo utilizados, junto con las versiones específicas y la configuración establecida. Asimismo, se proporciona una descripción de la infraestructura de hardware utilizada, abarcando tanto el sistema de procesamiento central como los dispositivos de aceleración por hardware utilizados para agilizar el entrenamiento del modelo. Además, se mencionan las estrategias y las técnicas empleadas para mitigar posibles limitaciones de recursos, como la implementación de mecanismos de paralelización. De esta manera, se asegura que el entorno de entrenamiento y validación estén perfectamente configurados y optimizados

para aprovechar al máximo los recursos disponibles y obtener resultados consistentes y confiables en el desarrollo de la presente investigación.

4.1.1. Hardware

Los resultados que se presentan a continuación se han obtenido mediante el uso del servicio de computación en la nube de *Google Colab*, el cual ofrece un entorno Linux altamente eficiente y versátil para llevar a cabo las operaciones de entrenamiento y validación de redes neuronales profundas. En particular, el entorno utilizado para la validación y pruebas está configurado con una capacidad suficiente de recursos, tal como memoria RAM de 32GB y una tarjeta GPU Nvidia Tesla T4 equipada con 16GB de memoria dedicada. Esta combinación de recursos de hardware de vanguardia ha permitido llevar a cabo los cálculos computacionalmente exhaustivos de manera ágil y eficaz, de forma que se maximiza el rendimiento y la precisión alcanzada del modelo. Uno de los aspectos destacados de este servicio es la capacidad de establecer conexiones fluidas, y de forma segura, con repositorios y sistemas de almacenamiento en la nube públicos y privados, lo que resulta especialmente conveniente en el contexto de este trabajo de investigación. Esta ventaja significativa elimina la necesidad de realizar copias adicionales de la base de datos utilizada para el entrenamiento, ya que el entorno del servicio puede acceder directamente a dichos repositorios, lo que brinda una mayor flexibilidad en el flujo de la construcción del modelo.

4.1.2. Software

El modelo se ha implementado utilizando la biblioteca de aprendizaje automático *PyTorch* 2.1.1v, que se basa en la librería *Torch* de *Linux Foundation umbrella*. Sin embargo, es importante destacar que este modelo depende de varias bibliotecas adicionales en *Python* para su correcto funcionamiento. Esta librería ofrece una amplia gama de arquitecturas y capas predefinidas que facilitan la construcción de modelos de mayor complejidad. Además, permite personalizar y ajustar los parámetros de las capas según los requisitos específicos a cada problema, lo que facilita la experimentación y la iteración rápida en el diseño del modelo. A continuación, se enumeran y describen brevemente estas dependencias:

- *numpy* 1.26.0v: Una biblioteca fundamental para el cálculo numérico en *Python*, proporcionando una amplia gama de funciones y operaciones eficientes para el manejo de matrices y arreglos multidimensionales [66].
- *einops* 0.7.0v: Una biblioteca que simplifica y mejora la manipulación de tensores en *PyTorch*. Proporciona funciones útiles para cambiar la forma y las dimensiones de los tensores, lo que facilita la implementación de diversas operaciones en el modelo [142].

- *torchvision.ops* 0.15v: La biblioteca oficial de visión por computadora de *PyTorch*. Este módulo específico proporciona implementaciones de operaciones de bajo nivel que son útiles para el procesamiento de imágenes y la manipulación de tensores [108].
- *dipy* 1.7.0v: Una biblioteca especializada en procesamiento de imágenes médicas y tractografía de difusión. *dipy* ofrece múltiples herramientas y algoritmos avanzados para el análisis de datos de imágenes médicas, incluido el procesamiento de tensores de difusión [55].
- *sklearn* 1.3.0v: Una de las bibliotecas de aprendizaje automático más populares en *Python*. *sklearn* proporciona una amplia variedad de algoritmos y herramientas para tareas de aprendizaje automático, incluyendo clasificación, regresión, agrupamiento y selección de características, entre otros [130].
- *tqdm.notebook* 2.0v: Una biblioteca que permite mostrar barras de progreso interactivas en entornos de *notebook*, lo que resulta especialmente útil para monitorear el progreso de largos ciclos de entrenamiento y validación del modelo [36].

Estas dependencias son esenciales para garantizar el correcto funcionamiento y la eficiencia del modelo implementado, ya que proporcionan las funcionalidades necesarias para diversas operaciones clave en el procesamiento de imágenes, el análisis de datos y el entrenamiento del modelo. Además, estas herramientas proporcionan funciones para la carga, preprocesamiento y gestión de los datos utilizados para el entrenamiento y validación de los resultados obtenidos, lo que facilita el trabajo con grandes volúmenes de información.

4.2. Procesamiento con FSL

Todo el conjunto de imágenes de la base de datos recopilada se ha procesado utilizando la herramienta FSL. Para cada uno de los 65 casos control, se han obtenido los volúmenes de anisotropía fraccional, difusividad media y el mapa de orientaciones de difusión. Mediante el uso de la librería FSL, se ha llevado a cabo un procesamiento exhaustivo en cada muestra de todos los pacientes. Los volúmenes de las direcciones de difusión se han calculado mediante la proyección del mapa de anisotropía fraccional sobre los vectores propios del tensor de difusión. Estos mapas proporcionan información sobre la dirección y la magnitud de la difusión del agua en el tejido cerebral. Esta medida es relevante para identificar cambios microestructurales en la conectividad en estudios de resonancia magnética de difusión y establece un procesamiento previo en un estudio de tractografía [200, 71] y delineamiento de tractos de interés.

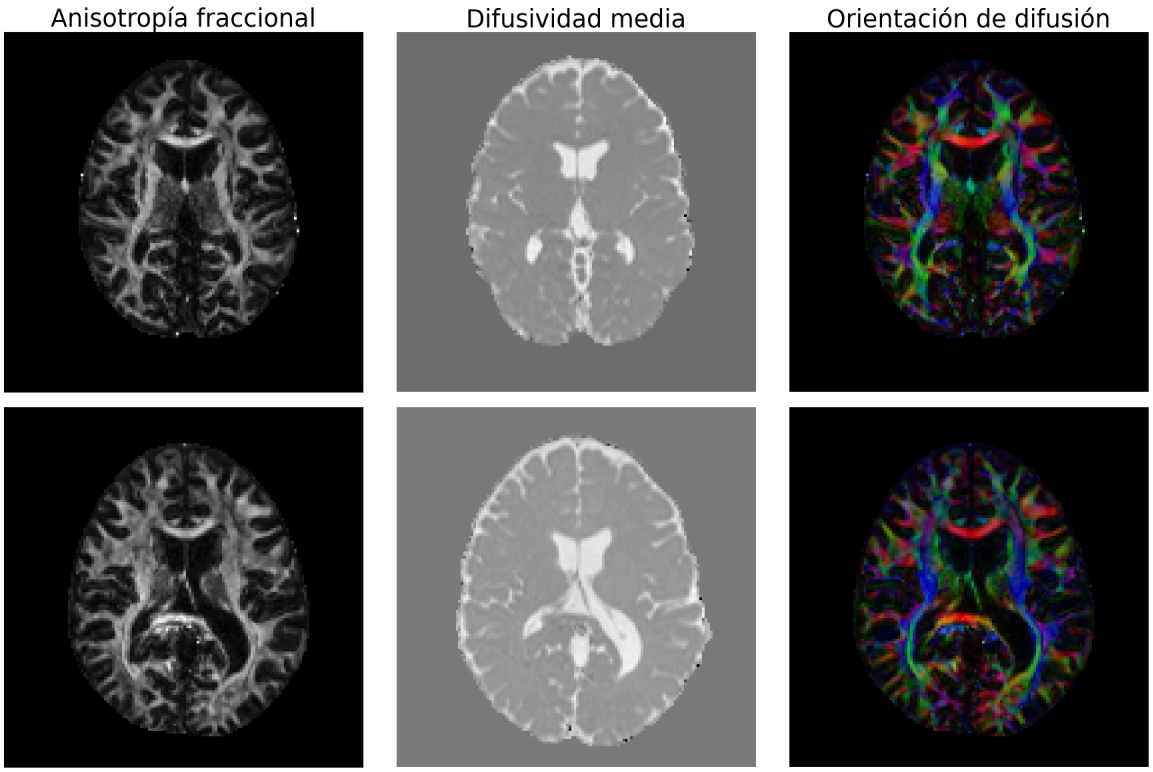

FIGURA 4.1: Mapas de anisotropía fraccional, difusividad media y orientación de difusión obtenidos con FSL para dos pacientes distintos de la base de datos HCP.

Los resultados obtenidos por este procesamiento han servido como referencia para la construcción del modelo que se propone en el presente trabajo de investigación y constituyen un elemento que determina profundamente el desempeño de este. Asimismo, se han adquirido las imágenes de difusividad media para cada caso control, lo que permite evaluar la rapidez con la que el agua se desplaza en diferentes direcciones dentro del tejido cerebral. Esta medida aporta información sobre la integridad y la coherencia estructural de los tejidos cerebrales. Por último, se ha generado el mapa de orientaciones de difusión, el cual muestra la dirección preferencial de la difusión del agua en cada punto del cerebro. Este mapa facilita la visualización y el análisis de la organización de las fibras nerviosas y las conexiones neuronales en el cerebro. Así, mediante el cálculo de los volúmenes de anisotropía fraccional, la difusividad media y la generación de mapas de orientaciones de difusión se extrae información crucial sobre la microestructura y la organización del tejido cerebral en cada uno de los 65 casos control. Estos resultados permiten realizar análisis comparativos y extraer conclusiones concretas en un estudio clínico. En la Figura 4.1 se muestran los mapas de anisotropía fraccional, difusividad media y orientación de difusión para dos pacientes de la base de datos HCP, en donde se

aprecian las diferencias estructurales del tejido de materia blanca en cada paciente.

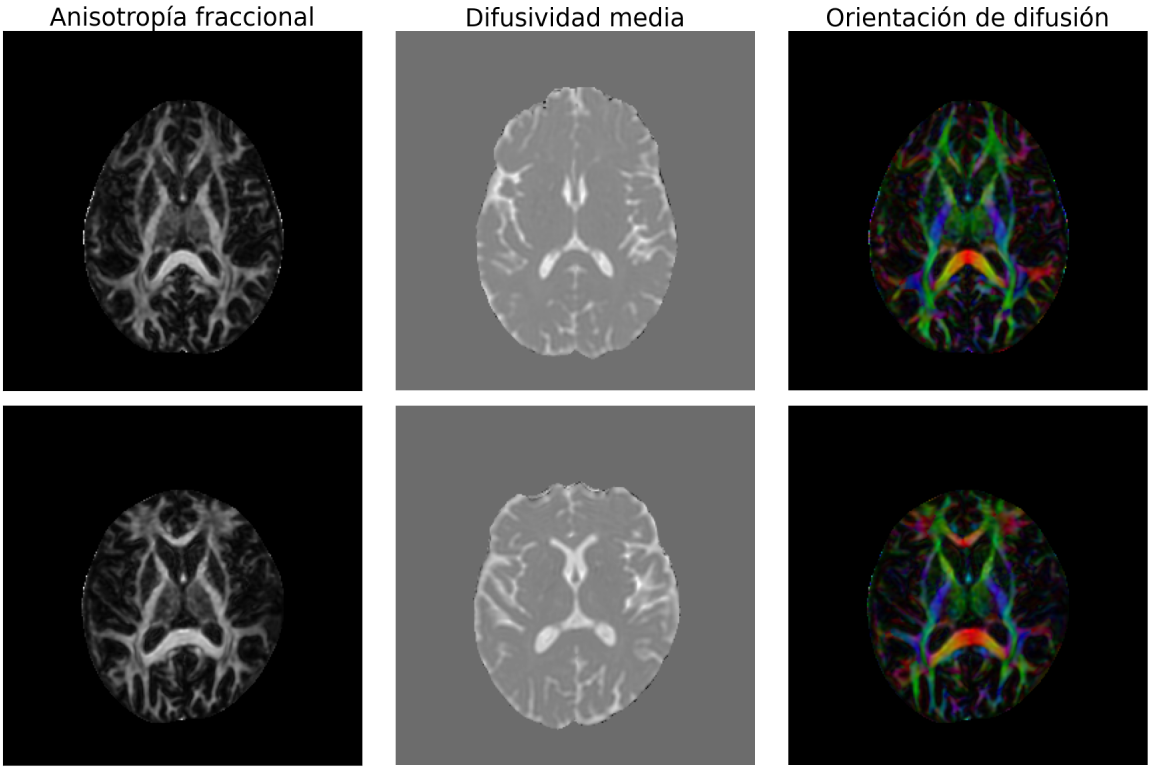

FIGURA 4.2: Mapas de anisotropía fraccional, difusividad media y orientación de difusión obtenidos con FSL para dos pacientes distintos de la base de datos ADNI.

Es importante mencionar que el mapa DO se ha generado a partir de combinar los mapas de FA y los vectores propios del tensor de difusión. Este mapa codifica lo orientación de la difusión en los colores rojo, verde y azul para las direcciones x, y y z, respectivamente, utilizando el marco referencial de la imagen. En la Figura 4.2 se presentan los mapas de FA, MD y la DO obtenidos para dos casos de la base de datos ADNI, en donde se aprecia ligeramente la corrección de anomalías del mapa FA.

4.3. Entrenamiento

Para ajustar los parámetros del modelo se ha empleado un algoritmo de optimización estocástico basado en gradiente y se ha definido la desviación cuadrática media, entre la salida del modelo y la referencia, como función de pérdida a optimizar por este algoritmo. Asimismo, se ha utilizado el 80 % de los datos para el entrenamiento del modelo, el

10 % para validación durante este entrenamiento y 10 % para una evaluación final. En la implementación del modelo y la obtención de los resultados que se muestran a continuación, se ha utilizado el servicio de computación en la nube de *Google Colabs* y, como se ha mencionado anteriormente, permite la posibilidad de comunicar el entorno con un repositorio privado, por lo que es posible monitorear los resultados del entrenamiento y validación en tiempo real.

El modelo implementado se ha entrenado utilizando las 7 señales de difusión a lo largo de los ejes principales de la imagen para cada región, esto supone una reducción de las señales necesarias para la estimación por un factor mayor a 8, y los mapas del tensor de difusión como la salida del sistema. En particular, se ha entrenado un modelo independiente para cada mapa del tensor de difusión deseado y se han generado los pares de entrada y salida para el ajuste de cada modelo. Entonces, el algoritmo de optimización minimiza la pérdida recorriendo iterativamente la red hacia delante y hacia atrás, actualizando los parámetros de la red con gradientes adaptativos. La desviación cuadrática media, entre la salida del modelo y la referencia, como función de pérdida a optimizar es una función definida por la siguiente expresión

$$L(\Theta) = \frac{1}{n}\sum_{t=1}^{n} ||F(x^t;\Theta) - y^t||^2 \tag{4.1}$$

en donde $F(\cdot)$ representa la función del modelo de la red neuronal y Θ denota los parámetros de la red que se ajustan durante el entrenamiento. El modelo se entrena durante 140 épocas con un ritmo de aprendizaje inicial de 0.0002 y se reduce en un factor de 0.9 cada que la función de pérdida no disminuye durante una época. Se ha utilizado un tamaño de lote de entrenamiento igual al número de imágenes contenidas en el volumen de cada paciente. En la Tabla 4.1 se muestran los valores de los hiperparámetros utilizados para el modelo final generando más de 35 millones de parámetros entrenables.

Hiperparámetro	**Valor utilizado para modelo**
in_chans	7
out_chans	1 *(3 para mapa DO)*
img_size	140
embed_dim	64
n_heads	[1,2,4,8]
mlp_ratio	[2,2,4,4]
reduction_ratio	2
depth_prob	0.2
tanh_output	False *(True para mapa MD)*

TABLA 4.1: Valores de hiperparámetros del modelo utilizados durante entrenamiento.

4.3.1. Aumento de datos

El aumento de datos para el entrenamiento de modelos de aprendizaje profundo se ha convertido en una técnica fundamental en la optimización de grandes modelos de redes neuronales. Esta técnica consiste en aumentar el tamaño del conjunto de datos de entrenamiento mediante la creación de nuevas instancias que son variaciones de las originales. Una forma común de hacer esto es aplicar transformaciones simples a las instancias existentes, como rotaciones, recortes, transformaciones espejo, entre otros. Esta técnica es especialmente útil cuando el conjunto de datos original es pequeño o cuando se trata de problemas de clasificación de imágenes o reconocimiento de voz.

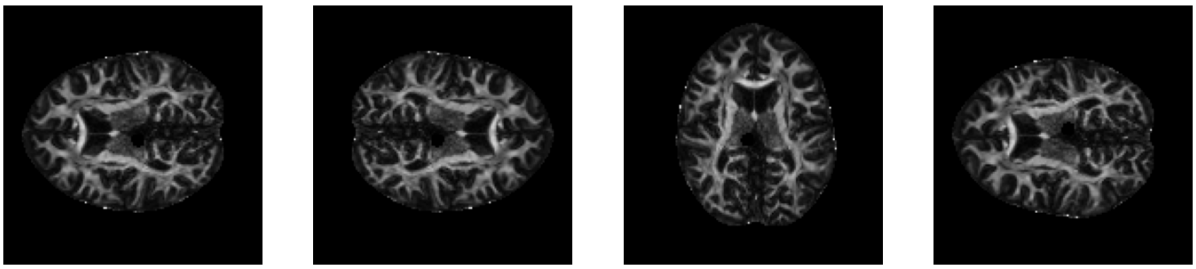

FIGURA 4.3: Aumento de datos utilizado para una imagen FA. De izquierda a derecha: imagen original, transformación espejo vertical, rotación de 90° y transformación espejo horizontal.

En el presente estudio, se utilizan las operaciones de espejo sobre el eje vertical y el eje horizontal y rotaciones aleatorias con ángulos múltiplos de un ángulo recto para evitar pérdida de información en la interpolación durante la reconstrucción de la imagen, como se muestra en la Figura 4.3. Esta técnica además proporciona una forma de evitar el sobre ajuste temprano de la arquitectura durante la optimización de sus parámetros.

4.3.2. Comportamiento de la función de pérdida

En el entrenamiento del modelo se ha calculado el valor de la función de pérdida sobre el conjunto de validación al final de cada época con el objetivo de monitorear el desempeño del sistema sobre conjuntos de datos desconocidos por la red, de forma que sea útil para la selección de los hiperparámetros y facilitar la identificación del momento en el que la arquitectura comienza a sobre ajustarse a los datos de entrenamiento. No obstante, a continuación, se muestra que en todo momento del entrenamiento el modelo se desempeña ligeramente mejor sobre el conjunto de validación. Esto se puede deber a que el cálculo de la función de pérdida sobre el conjunto de validación se realiza al final de cada época, mientras que el valor de la función de pérdida para el conjunto de entrenamiento se obtiene al promediar una ponderación de los valores individuales de cada lote de este conjunto. Aun así, se demuestra la gran capacidad de las redes

neuronales basadas en autoatención de generalizar a partir de datos de entrenamiento. En las gráficas de la Figura 4.4 se muestran la función de pérdida a lo largo de cada época del entrenamiento del modelo para la estimación del mapa FA. La optimización se ha detenido al no obtener una mejora significativa en más de dos épocas consecutivas y esto ha ocurrido aproximadamente en la época 140, como se muestra en el gráfico.

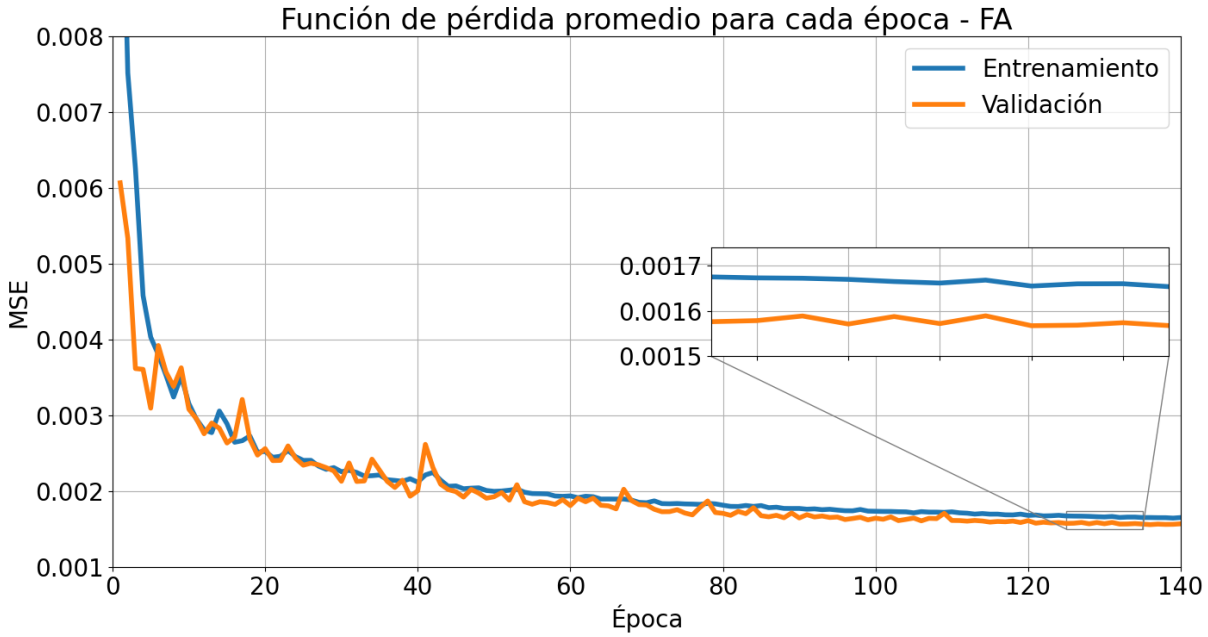

FIGURA 4.4: Función de pérdida durante el entrenamiento del modelo para la estimación del mapa de anisotropía fraccional (FA).

En el caso del entrenamiento del modelo para la estimación del mapa de anisotropía fraccional se observa que, para las primeras épocas, la función de pérdida decae rápidamente, hasta establecerse alrededor de la época 50, a partir de la cual el error cuadrático medio entre la estimación del modelo y la referencia disminuye asintóticamente a un valor aproximado de 0.0016. En cambio, la función de pérdida durante el entrenamiento del sistema para la estimación del mapa de difusividad media presenta cambios abruptos durante las primeras 30 épocas, como se muestra en el gráfico de la Figura 4.5 y, del mismo modo, la función alcanza un estado estable alrededor de un valor de 0.0014. Finalmente, la función de pérdida durante la optimización del sistema para la estimación del mapa de la orientación de difusión ha tenido menos dificultades para ajustar una gran cantidad de parámetros durante las primeras 20 épocas, como se muestra en el gráfico de la Figura 4.6 en términos del error cuadrático medio. Esto se puede deber a que la información de salida contiene más información que en el caso de los mapas anteriores, puesto que las señales de salida contienen 3 canales sobre una dimensión adicional. En los tres casos es evidente que las técnicas de regularización empleadas durante la construcción del modelo han sido efectivas.

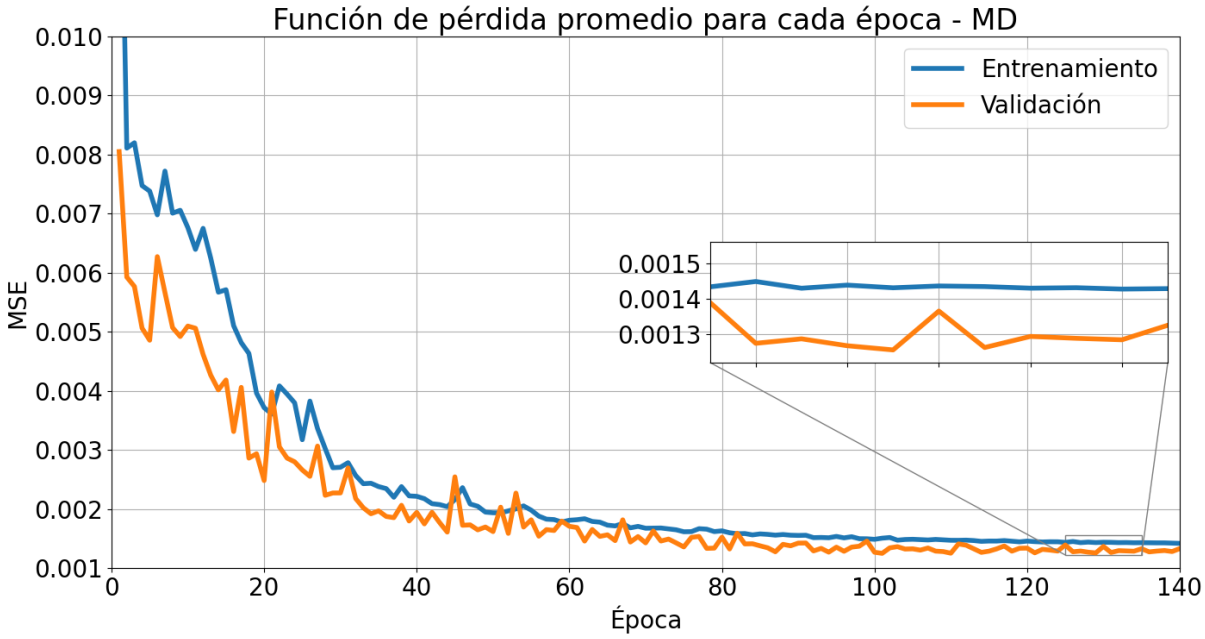

FIGURA 4.5: Función de pérdida durante el entrenamiento del modelo para la estimación del mapa de difusividad media (MD).

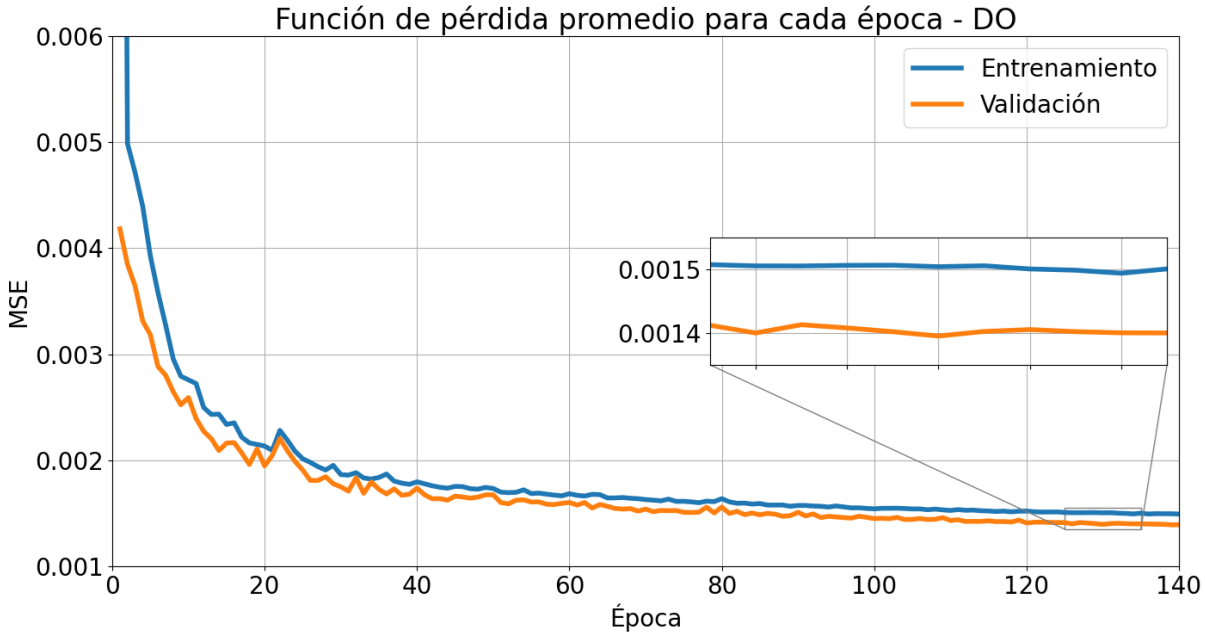

FIGURA 4.6: Función de pérdida durante el entrenamiento del modelo para la estimación del mapa de la orientación de difusión (DO).

4.4. Resultados

Para evaluar el desempeño del modelo propuesto en la estimación de los mapas del tensor de difusión se utilizan las métricas descritas en el capítulo anterior, las cuales son el error cuadrático medio normalizado, el índice de similitud estructural y la proporción máxima de señal a ruido. Se obtienen los valores de cada métrica en cada una de las imágenes del conjunto de prueba y se calcula el promedio de estos resultados. Estos valores promedio se muestran en la Tabla 4.2. Además, se ha incluido otra métrica para evaluar las prestaciones del modelo implementado: el tiempo de procesamiento necesario en CPU para obtener las imágenes de un volumen completo del cerebro de un paciente.

Mapa - Método	NMSE (%)	SSIM	PSNR (db)	Tiempo (min)
FA				
UT-Net	8.5307	**0.8552**	**27.8626**	2.74
U-Net	**6.3676**	0.8383	27.4454	**1.57**
MC (FSL)	–	0.7708	17.2302	18.98
MD				
UT-Net	5.1540	**0.9094**	**27.9955**	3.02
U-Net	**3.8267**	0.8869	26.7074	**1.66**
MC (FSL)	11.0139	0.8789	27.0625	18.98
DO				
UT-Net	9.1360	**0.8252**	**28.7752**	2.45
U-Net	**6.9325**	0.8056	28.2501	**1.40**
MC (FSL)	82.1052	0.7650	20.1209	18.98

TABLA 4.2: Métricas promedio obtenidas para la inferencia del conjunto de evaluación.

Del mismo modo, para evaluar el efecto de utilizar módulos *Transformer* en una arquitectura tipo autoencoder, se ha implementado y entrenado un modelo *U-Net* [143] para la estimación de los mapas del tensor de difusión, el cual se compone únicamente de módulos convolucionales. Este modelo es optimizado utilizando exactamente la misma base de datos que la utilizada para optimizar el modelo basado en *Transformers*, con el fin de evitar algún sesgo en la comparación y se realice bajo las mismas condiciones. Además, se han obtenido estos mapas utilizando el método convencional (*FSL*), pero esta vez con las mismas 7 señales de difusión que utilizan ambos modelos de redes neuronales. En la Tabla 4.2 se muestra que, para los mapas FA, MD y DO el modelo *UT-Net* otorga el mejor desempeño sobre el mismo conjunto de datos respecto a la métrica SSIM y proporcionando un error cuadrático medio normalizado promedio de 8.53 %, 5.15 % y 9.1360 %, respectivamente; y un índice de similitud estructural promedio de 0.85, 0.90 y 0.82, respectivamente. En cuanto al tiempo de procesamiento, el método propuesto

ha sido capaz de estimar el volumen completo del mapa de anisotropía fraccional de un paciente en menos de 3 minutos. Un cambio bastante drástico en comparación al método convencional. En cambio, para el modelo basado en bloques convolucionales, para la estimación de los 3 mapas DTI ha obtenido el mejor desempeño respecto a la métrica NMSE y el tiempo de procesamiento, puesto que la arquitectura *U-Net* realiza el procesamiento de todas las imágenes del conjunto de datos de un paciente en menos de 2 minutos. De esta forma, en los siguientes apartados se presentan algunas instancias de los resultados obtenidos para los tres mapas estimados del tensor de difusión.

4.4.1. Anisotropía fraccional

El mapa de anisotropía fraccional describe la dirección y la intensidad de la difusión de las moléculas de agua en el tejido cerebral. Este estudio es utilizado para visualizar y analizar las fibras nerviosas y la integridad de la sustancia blanca en el cerebro. También, es un mapa necesario en el uso de técnicas de tractografía y es posible estudiar una reconstrucción de la dirección de las conexiones cerebrales.

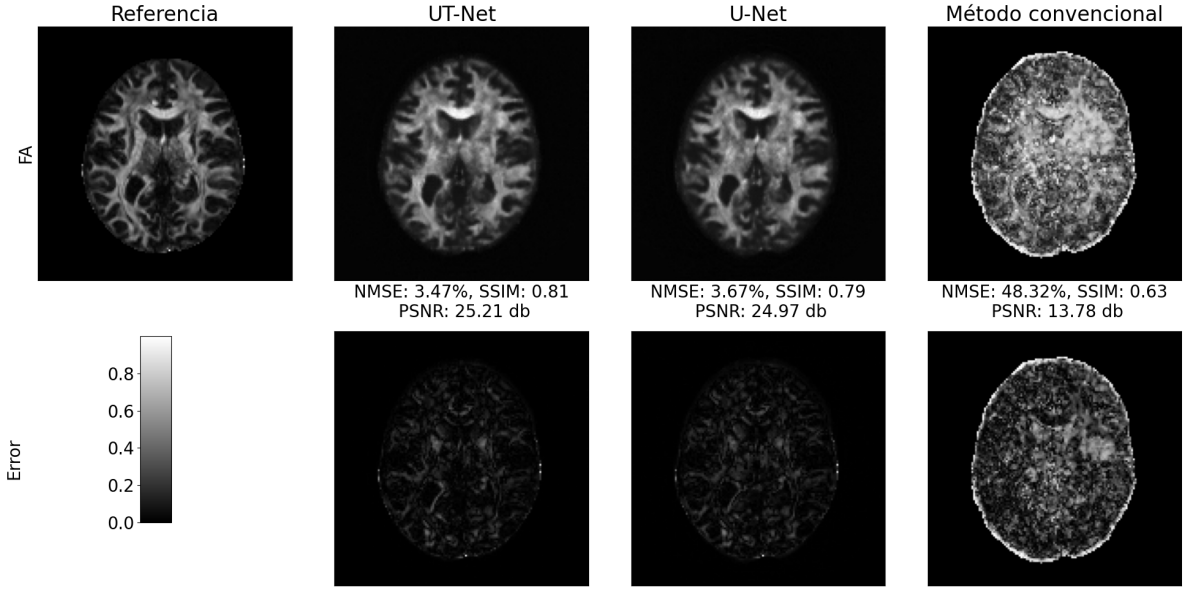

FIGURA 4.7: Comparación de resultados que se han obtenido para un caso del conjunto de evaluación en la estimación del mapa de anisotropía fraccional. El error mostrado es la diferencia absoluta entre la imagen estimada y la imagen de referencia.

Como se ha comentado, se comparan los resultados utilizando distintas metodologías, pero en cada caso se han utilizado las mismas señales de entrada, permitiendo demostrar cuál de los modelos es capaz de extraer mayor información a partir de la misma cantidad

de datos de entrada. En la Figura 4.7 se presenta un ejemplo del mapa de anisotropía fraccional obtenido por el modelo *UT-Net*, el modelo *U-Net* y el método convencional, que se basa en la librería FSL. Además, se muestran los valores obtenidos para las métricas de desempeño y su correspondiente mapa de error absoluto de cada resultado. En este caso, es evidente que el modelo propuesto proporciona el mejor desempeño para la estimación de este mapa del tensor de difusión, con un índice de similitud estructural de 0.81. Además, es importante notar que el método convencional no ha sido capaz de reconstruir las imágenes de anisotropía fraccional por completo a partir de únicamente 7 señales de difusión. Incluso, la etapa de extracción de ruido blanco de este último método ha fallado significativamente.

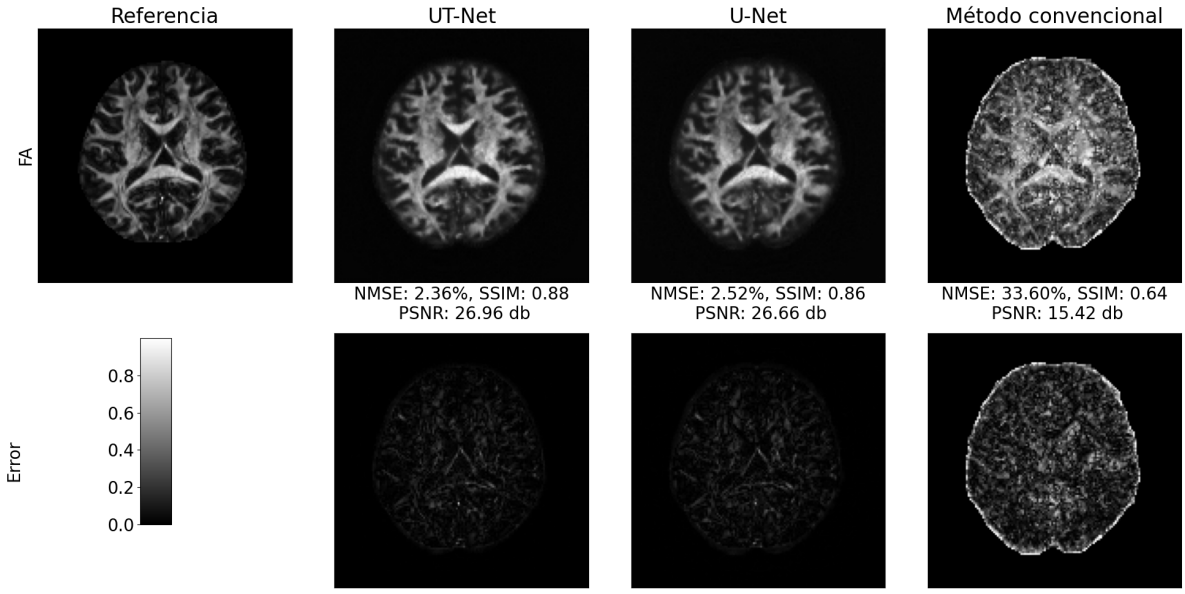

FIGURA 4.8: Comparación de resultados que se han obtenido para un caso del conjunto de evaluación en la estimación del mapa de anisotropía fraccional. El error mostrado es la diferencia absoluta entre la imagen estimada y la imagen de referencia.

En este sentido, el obtener imágenes más precisas del mapa de anisotropía fraccional a partir de menos señales de difusión significa una mejora en análisis de tractografía, permitiendo una visualización más clara y precisa de las conexiones y tractos de fibras nerviosas en el cerebro. De la misma forma, en la Figura 4.8 se presenta otro ejemplo del mapa de anisotropía fraccional estimado por los tres métodos antes mencionados. En este caso, el método basado en bloques *Transformers* proporciona un error cuadrático medio normalizado del 2.36 % y un índice de similitud estructural de 0.88, siendo resultados bastante positivos. De este modo, en la investigación cerebral, tener datos de mayor precisión en un menor tiempo permite obtener conclusiones de mayor solidez, mejorar

la calidad de los estudios y ofrecer diagnósticos más profundos sobre la estructura y funciones cerebrales. Otro aspecto que no se puede dejar pasar por alto, es la propiedad de los modelos basados en redes neuronales profundas a que, implícitamente, han aprendido a extraer las características necesarias para realizar la segmentación del tejido cerebral, así como la extracción del cráneo.

4.4.2. Difusividad media

Mientras que el estudio de anisotropía fraccional se centra en la dirección y la coherencia de la difusión del agua en los tejidos, el mapa de difusividad media se enfoca en el promedio general de la difusión, independientemente de la dirección. Así, este mapa proporciona información sobre la magnitud de la difusión del agua en los tejidos cerebrales. La información que se puede derivar a partir de estos estudios incluye la integridad celular de la materia blanca, en donde un aumento en este valor suele indicar un aumento en la difusividad del agua, lo que podría ser consecuencia de una disminución en las barreras celulares.

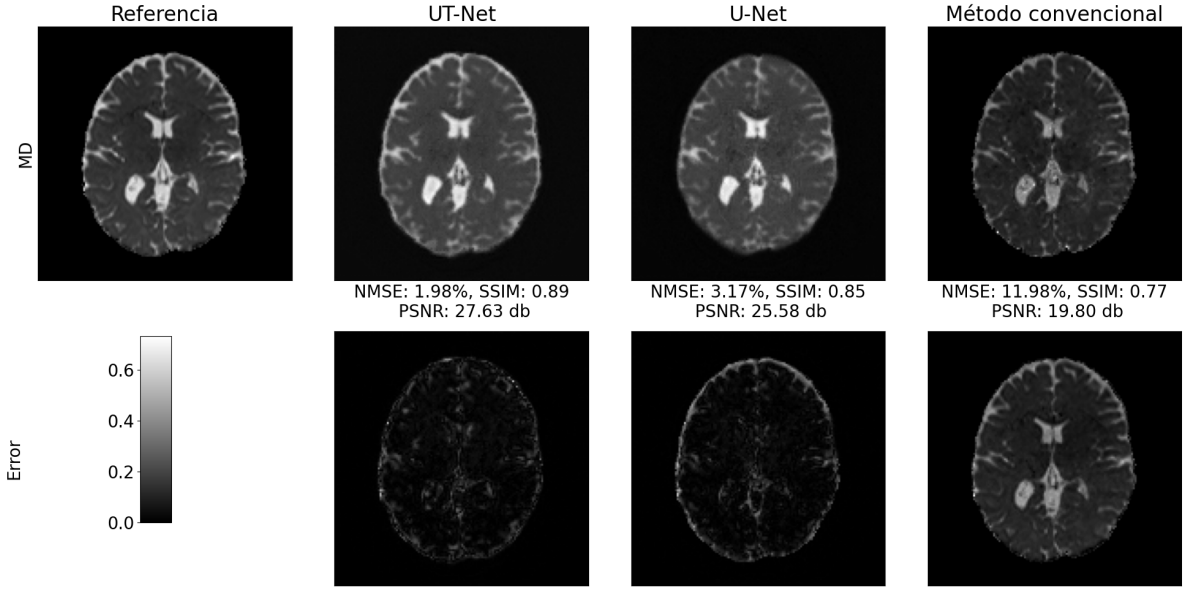

FIGURA 4.9: Comparación de resultados que se han obtenido para un caso del conjunto de evaluación en la estimación del mapa de difusividad media. El error mostrado es la diferencia absoluta entre la imagen estimada y la imagen de referencia.

De la misma manera, se comparan los resultados obtenidos mediante tres métodos distintos y se contrastan de manera cualitativa y cuantitativa. En la Figura 4.9 se muestra un ejemplo del mapa de difusividad media estimado por los tres métodos mencionados

en el apartado anterior. En esta instancia, el modelo basado tanto en módulos convolucionales como *Transformers* ha logrado superar el desempeño de los demás modelos. En la Figura 4.10 se muestra otro ejemplo del mapa de difusividad media calculado mediante los tres métodos y, en este caso, tanto el modelo basado en bloques *Transformers* y el modelo basado únicamente en capas convolucionales han arrojado un desempeño similar, con un índice de similitud estructural de 0.93 y 0.88, respectivamente. Aún más, el método convencional ha arrojado el mejor resultado para la estimación de este mapa, con un índice de similitud estructural de 0.96. Esto último sugiere que la estimación del mapa de difusividad media es una tarea de menor complejidad que la estimación del mapa de anisotropía fraccional y no es necesaria la integración de módulos basados en mecanismos de atención para mejorar la estimación de este mapa. A diferencia de la estimación del mapa de anisotropía fraccional, en este caso el método convencional es lo suficientemente potente para estimar el mapa de difusividad media a partir de solo 7 señales de difusión, lo que sustenta la suposición anterior sobre de la complejidad de este problema.

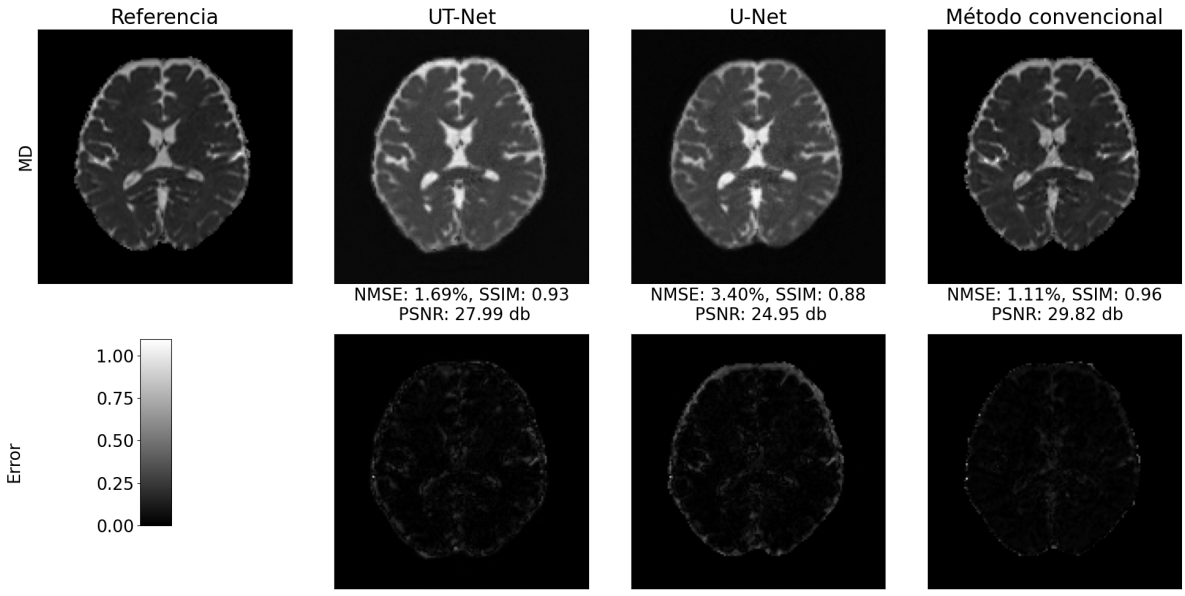

FIGURA 4.10: Comparación de resultados que se han obtenido para un caso del conjunto de evaluación en la estimación del mapa de difusividad media. El error mostrado es la diferencia absoluta entre la imagen estimada y la imagen de referencia.

Con estos ejemplos es posible apreciar que el modelo ha aprendido a modelar los coeficientes del tensor de difusión a partir de menos señales que con las que han sido generadas las imágenes de referencia. Además, si bien solo se ha logrado una ligera mejora en el desempeño para la estimación de este mapa, son importante los resultados

de este estudio para indagar con mayor profundidad el efecto de los mecanismos de atención en arquitecturas profundas para el procesamiento de señales médicas.

4.4.3. Orientación de difusión

El mapa de orientación de difusión no es una métrica específica por sí misma, como lo son la anisotropía fraccional o la difusividad media, puesto que es la combinación del mapa de anisotropía fraccional y los vectores propios del tensor. Sin embargo, este mapa proporciona información de las orientaciones principales de difusión que se derivan del tensor de difusión. Estas orientaciones indican la dirección principal en la que el agua se mueve libremente en una región particular de la imagen. La información que proporciona este mapa incluye las direcciones y alineaciones de las fibras, la visualización de haces de tractos y la identificación de cruces de fibras. Y, como se ha comentado, los colores rojo, verde y azul representan la difusión en los ejes x, y y z respectivamente.

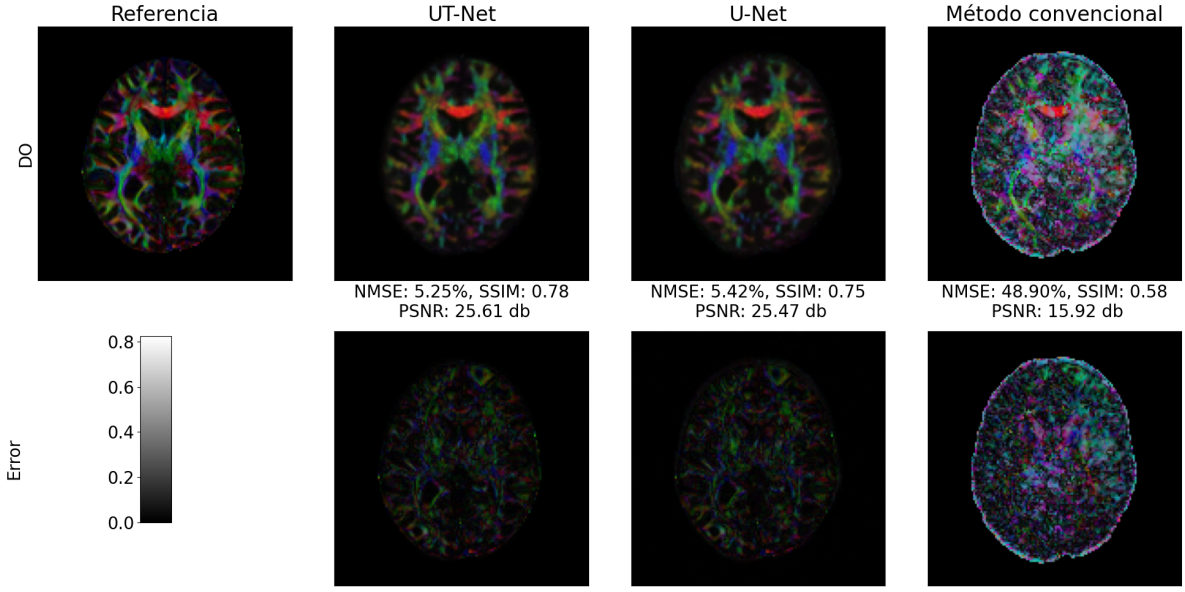

FIGURA 4.11: Comparación de resultados que se han obtenido para un caso del conjunto de evaluación en la estimación del mapa de la orientación de difusión. El error mostrado es la diferencia absoluta entre la imagen estimada y la imagen de referencia.

En concreto, este mapa proporciona información descriptiva sobre la dirección y trayectoria de las fibras nerviosas en el cerebro, lo que permite visualizar y analizar la estructura y conectividad de la sustancia blanca. En la Figura 4.11 se presentan los resultados obtenidos de la estimación del mapa de las orientaciones de difusión mediante el método basado en *Transformers*, el modelo basado en operaciones convolucionales

y el método convencional. Este es un problema similar al cálculo del mapa de anisotropía fraccional y, por tanto, el uso de las ecuaciones convencionales no proporciona una reconstrucción completa de las imágenes. En cambio, el modelo *UT-Net* aporta el mejor desempeño con un índice de similitud estructural de 0.78 y una proporción de señal a ruido de 25.61db. En la Figura 4.12 se muestra otra instancia de los resultados obtenidos en el cálculo de este mapa y, de la misma manera, la estimación obtenida mediante la arquitectura *UT-Net* presenta el mejor resultado con un error cuadrático medio normalizado de 4.38 % y un índice de similitud estructural de 0.81, lo que demuestra que los bloques basados en mecanismos de atención son capaces de extraer una mayor cantidad de información sobre la estructura y regularidad de los tractos neuronales en las imágenes.

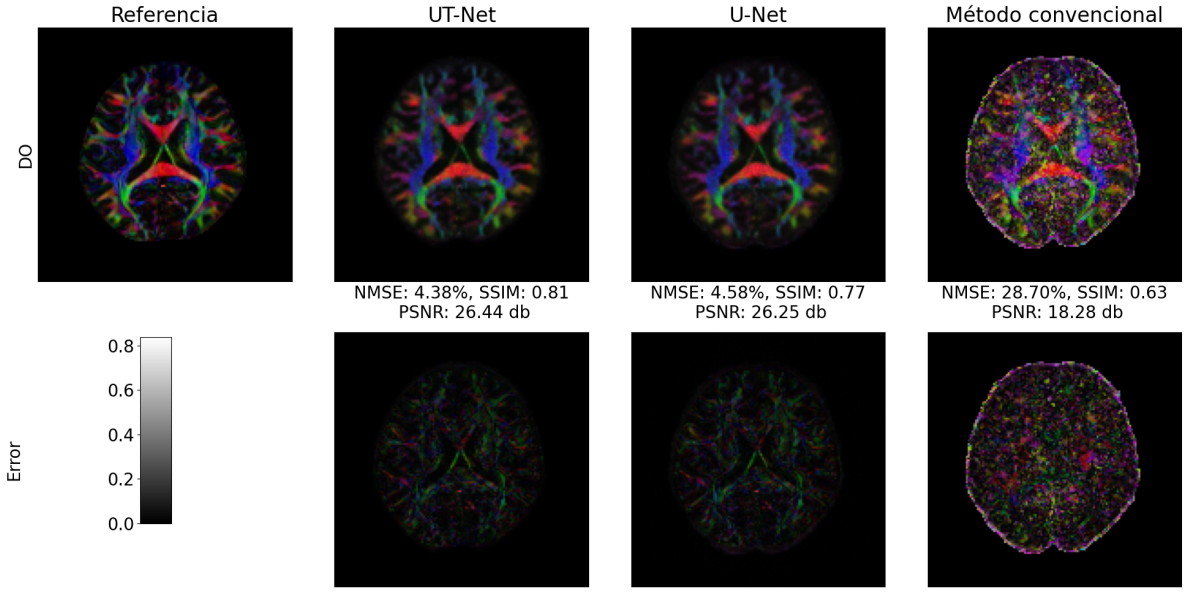

FIGURA 4.12: Comparación de resultados que se han obtenido para un caso del conjunto de evaluación en la estimación del mapa de la orientación de difusión. El error mostrado es la diferencia absoluta entre la imagen estimada y la imagen de referencia.

Como se ha comentado, la integración de los bloques *Transformers* ha permitido obtener mejoras considerables, respecto a una arquitectura con operaciones convolucionales, cuando el problema consiste en modelar elementos altamente estructurados en una imagen. En el contexto del aprendizaje computacional, la tarea de estimación del mapa de orientación de difusión constituye un problema de regresión dentro de un tensor. Así, el incrementar la precisión mientras se reduce el tiempo de procesamiento para obtener este mapa del tensor de difusión permite agilizar el diagnóstico médico y acelerar

análisis posteriores, tal como la tractografía, que se utiliza para rastrear y visualizar las conexiones cerebrales en tres dimensiones.

4.5. Desempeño del modelo en pacientes con patologías o alteraciones cerebrales

Por último, para evaluar la robustez del modelo implementado se han obtenido las estimaciones de los mapas de anisotropía fraccional, difusividad media y orientaciones de difusión en dos grupos de pacientes con patologías o alteraciones cerebrales. Estas imágenes han sido extraídas con un escáner distinto al utilizado por las imágenes del conjunto de muestras de entrenamiento y son datos totalmente no vistos por la red durante el entrenamiento, por lo que resulta una prueba ideal para medir las prestaciones del modelo basado en *Transformers*. Las características de los conjuntos de imágenes utilizados para esta evaluación se describen a continuación.

Sujetos con Deterioro Cognitivo Leve: La población consta de un total de 4 sujetos de origen mexicano diagnosticados por expertos en neurología. Todos los pacientes tienen un *Mini-mental state examination* (MMSE) menor o igual a 24 puntos, un *Clinical Dementia Rating* (CDR) con un índice mayor o igual a 0.5 y su rango de edades es de 79.9 ± 9.2 años. Todos los pacientes dieron su consentimiento escrito de acuerdo con la declaración de Helsinki. Las imágenes para esta población fueron adquiridas en el Centro Nacional de Investigación en Imagenología e Instrumentación Médica (CI3M) de la Universidad Autónoma Metropolitana, unidad Iztapalapa (UAM-I), con el número de aprobación PND_AC_08_16. Los volúmenes cerebrales de tensor de difusión fueron adquiridos con un equipo 3T Phillips, con *b-values* de 0 y 800 s/mm^2, y 32 direcciones de difusión.

Población de niños con afectaciones causadas por la desnutrición: La población consta de un total de 18 pacientes pediátricos, quienes padecieron desnutrición infantil en sus primeros años de vida. La población de estudio tiene un rango de edades de entre 6 a 8 años, y son originarios de México. Las adquisiciones fueron realizadas por la Facultad de Psicología, Facultad de Medicina e Instituto de Neurobiología de la UNAM bajo el protocolo número INB-065 y con el consentimiento informado de acuerdo con la declaración de Helsinki. Las imágenes de tensor de difusión fueron adquiridas considerando *b-values* de 0 y 800 s/mm^2, y 32 direcciones de difusión mediante un equipo 3T Phillips.

Entonces, resulta evidente que las condiciones en la extracción de estas imágenes han sido diferentes a las condiciones de la extracción de las imágenes utilizadas para optimizar el modelo. Con ello, es factible evaluar el desempeño del modelo basado en bloques *Transformers* ante entradas con características que no se han modelado durante el ajuste de los parámetros de la arquitectura. Así, se obtienen los mapas del tensor de difusión utilizando el método convencional con todas las señales de difusión proporcionadas en

cada conjunto de datos, para obtener las imágenes de referencia. Luego, para generar los tensores de entrada, se extraen las 6 señales de difusión en las direcciones sobre los ejes coordenados de la imagen y la imagen sin dirección de difusión. Posteriormente, se estiman los mapas del tensor de difusión utilizando el modelo con módulos *Transformers* y el modelo convolucional, y se comparan contra las imágenes obtenidos mediante el método convencional utilizando todas las señales. En la Figura 4.13 se muestra una instancia de estos mapas y su respectiva imagen de error, para un paciente con deterioro cognitivo leve. El mapa de error muestra la diferencia absoluta entre el tensor de referencia y el tensor obtenido por el modelo que se basa en mecanismos de atención.

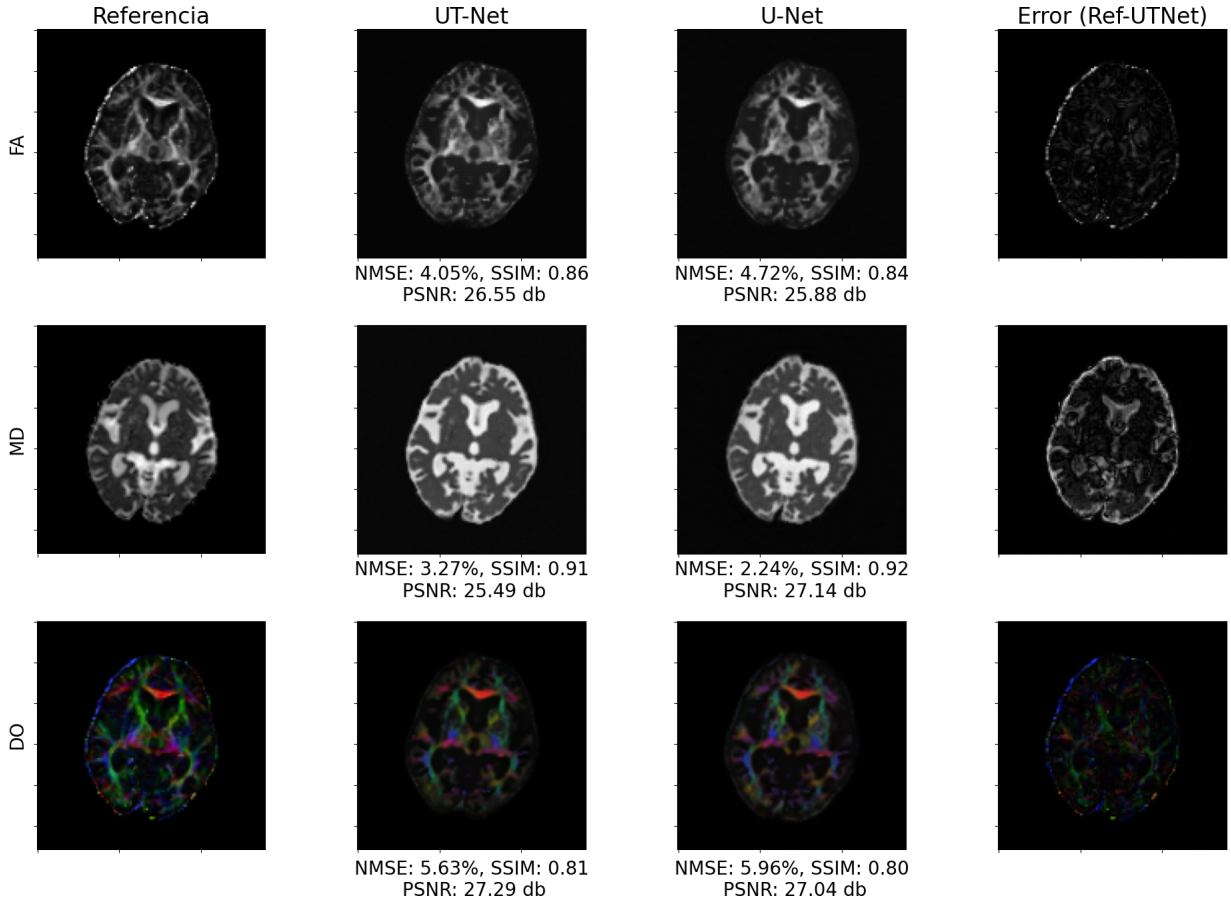

FIGURA 4.13: Comparación de mapas del tensor de difusión obtenidos con el modelo *UT-Net* y la arquitectura *U-Net* para el conjunto de casos con deterioro cognitivo leve.

En esta instancia, el modelo *UT-Net* ha obtenido el mejor desempeño proporcionando un índice de similitud estructural de 0.86 y un error cuadrático medio normalizado de 4.05 % para la estimación del mapa FA. En cambio, esto no resulta así para la estimación del mapa MD, para el cual el modelo *U-Net* ha dado un mejor desempeño con un

índice de similitud estructural de 0.92. Además, se calculan las métricas de desempeño utilizadas en la presente investigación para evaluar cuantitativamente el rendimiento de ambos modelos. En la Tabla 4.3 se presenta el promedio de cada una de estas métricas para cada uno de los mapas DTI que se han obtenido.

Mapa - Método	NMSE (%)	SSIM	PSNR (db)
FA			
UT-Net	7.2246	0.8375	26.0209
U-Net	**6.9101**	**0.8475**	**26.1548**
MD			
UT-Net	7.1511	0.9160	26.1828
U-Net	**3.3111**	**0.9237**	**27.3424**
DO			
UT-Net	9.4878	**0.8365**	**28.6930**
U-Net	**7.3028**	0.8183	28.1553

TABLA 4.3: Métricas promedio obtenidas para la inferencia del conjunto de casos con deterioro cognitivo leve.

En este caso, para el mapa de anisotropía fraccional el modelo propuesto ha proporcionado un índice de similitud estructural promedio de 0.83, mientras que el modelo convolucional ha obtenido un índice de 0.84, ligeramente mayor. Sin embargo, en la estimación del mapa de las orientaciones de difusión, el modelo propuesto proporciona el mejor desempeño promedio. Por otra parte, se obtienen los mismos estudios para el conjunto de imágenes de pacientes infantes con afectaciones causadas por desnutrición. Un ejemplo de los mapas obtenidos para estos pacientes, y sus correspondientes imágenes de error, se muestran en la Figura 4.14. Las afectaciones causadas por esta patología no son predominantemente locales, de modo que no es posible focalizarlas dentro de la imagen. La figura muestra una comparativa de imágenes cerebrales obtenidas mediante las diferentes técnicas comentadas anteriormente, organizada en tres filas correspondientes a las modalidades FA, MD y DO, y cuatro columnas que representan la imagen de referencia, los resultados de procesamiento con el modelo *UT-Net* y *U-Net*, y el error entre la referencia y *UT-Net*. Cada imagen procesada en las columnas *UT-Net* y *U-Net* viene acompañada de las métricas de desempeño que cuantifican su precisión en relación con la referencia. Estas métricas ofrecen una evaluación cuantitativa de la eficacia de las redes neuronales construidas con bloques basados en mecanismos de atención y operadores convolucionales en comparación con la imagen original, la cual se ha obtenido a partir de todos los volúmenes de difusión para cada paciente. En este ocasión, las estimaciones obtenidas por el modelo *UT-Net* han proporcionado el mejor desempeño respecto al índice de similitud estructural.

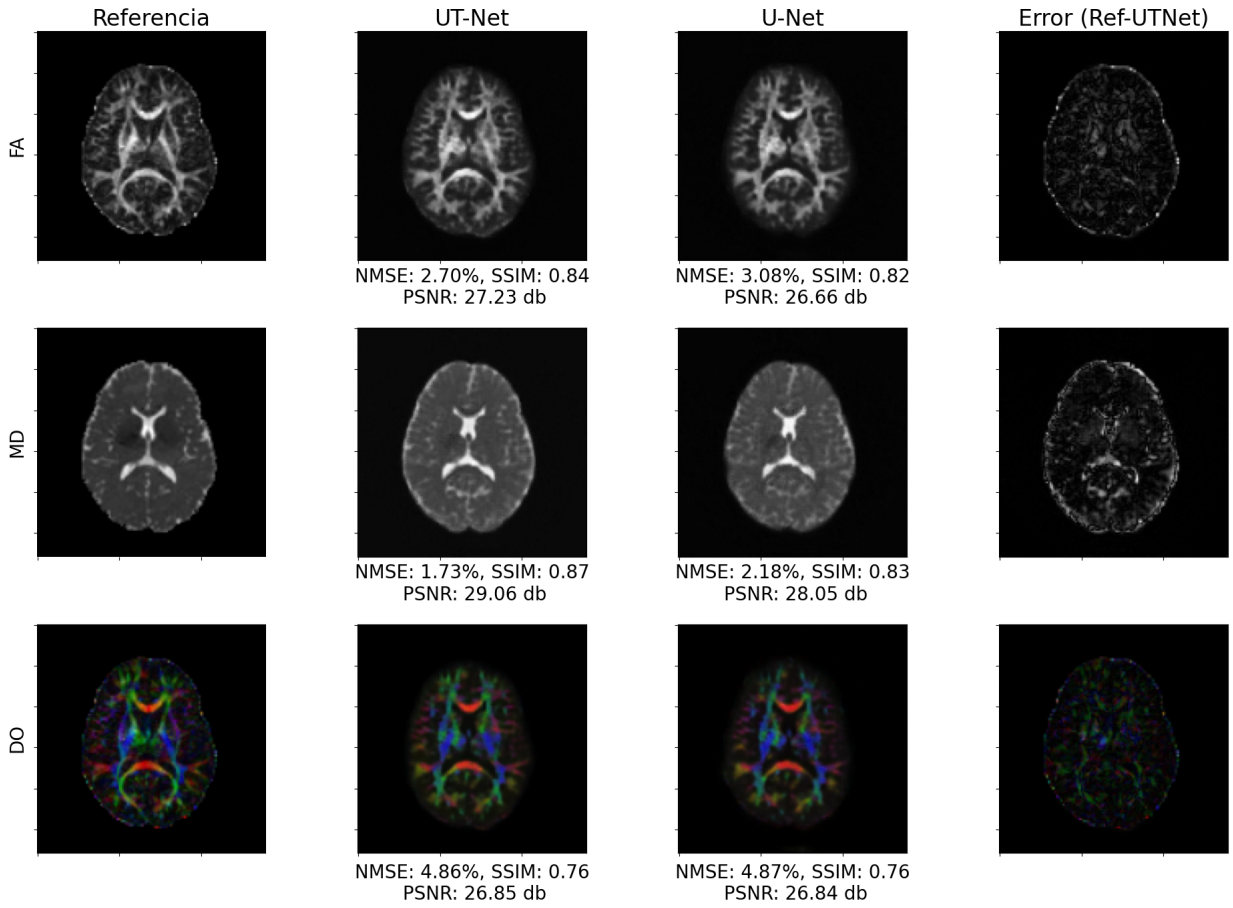

FIGURA 4.14: Comparación de mapas del tensor de difusión obtenidos con el modelo *UT-Net* y la arquitectura *U-Net* para el conjunto de casos de niños con afectaciones causadas por la desnutrición.

De esta forma, las métricas promedio obtenidas para este conjunto de datos se encuentra en la Tabla 4.4. En esta prueba, los mejores valores para la métrica de desempeño NMSE se han obtenido mediante el modelo *U-Net*, proporcionando índices promedio de 5.99 %, 3.89 % y 6.52 % para el mapa de anisotropía fraccional, difusividad media y las orientaciones de difusión, respectivamente. En ambos casos, en la evaluación con el conjunto de imágenes de pacientes con deterioro cognitivo leve y las imágenes de pacientes infantes, se tiene que el modelo basado en únicamente operaciones convolucionales ha obtenido el mejor resultado para el mapa FA y MD, respectivamente. Esto último se puede deber a que, como se ha comentado, el problema de la estimación de estos mapas es de menor complejidad, puesto que la regularidad espacial utiliza un mayor espacio dentro de las imágenes y no se requiere la información de las direcciones principales del tensor de difusión, es una representación de la media del grado y dirección de la difusividad en el tejido cerebral.

Mapa - Método	NMSE (%)	SSIM	PSNR (db)
FA			
UT-Net	6.3256	0.8150	26.2538
U-Net	**5.9995**	**0.8324**	**27.3179**
MD			
UT-Net	4.4130	**0.8933**	30.2078
U-Net	**3.8922**	0.8669	**30.4245**
DO			
UT-Net	10.2101	**0.8082**	28.0701
U-Net	**6.5209**	0.8005	**28.3074**

TABLA 4.4: Métricas promedio obtenidas para la inferencia del conjunto de casos de niños con afectaciones causadas por la desnutrición.

No obstante, es importante resaltar que se ha demostrado que el modelo que integra bloques tipo *Transformer* es capaz de estimar los mapas del tensor de difusión en pacientes con patologías, incluso cuando no ha sido entrenado u optimizado para estos casos en particular y con un protocolo de adquisición de imágenes distinto al utilizado en la construcción de la arquitectura.

4.6. Discusión

En la presente investigación se ha estudiado el efecto de integrar módulos basados en autoatención en una arquitectura de red neuronal profunda para la estimación de tres distintos mapas derivados del tensor de difusión en imágenes cerebrales de resonancia magnética. Específicamente, la microestructura de las fibras de materia blanca forma una complicada red en todo el cerebro con características altamente estructuradas y, debido a esto, el modelar mapas de características a partir de imágenes de resonancia magnética ponderadas por difusión es todo un reto computacional. En este sentido, es importante destacar que la estimación del mapa de anisotropía fraccional, difusividad media y la orientación de difusión es una tarea crucial en el procesamiento de imágenes por tensor de difusión, ya que estos mapas componen una representación visual y cuantitativa de la microestructura del tejido interno de materia blanca, proporcionando información importante sobre la integridad del tejido de las fibras nerviosas y la conectividad del cerebro. Entonces, con el uso de una arquitectura de aprendizaje profundo, que integra bloques *Transformer*, ha sido posible modelar la relación no lineal entre las señales de difusión y el tensor de difusión en imágenes. Esto ha permitido disminuir drásticamente el número de señales necesarias para efectuar la reconstrucción del tensor y el tiempo de cómputo necesario para procesar el volumen completo de un paciente. Esto último abre la posibilidad de integrar esta tecnología en el área clínica para el procesamiento de grandes conjuntos de datos o el procesamiento y visualización en tiempo real. Por tanto, el uso

de redes neuronales profundas basadas en mecanismo de atención para la estimación de estos parámetros tiene varias ventajas en comparación con los métodos tradicionales de procesamiento de DTI, puesto que el modelo ha sido capaz de modelar características complejas y no lineales en la estructura de los mapas a partir de únicamente 6 señales de difusión, aun cuando la base de información utilizada para optimizar el modelo ha sido limitada. Además, una ventaja importante del uso de los módulos *Transformer* es la alta capacidad de paralelización del procesamiento, lo que permite escalar el sistema a un entorno de mayor demanda. Entonces, sería interesante estudiar el efecto del tamaño de la base de datos utilizada para el entrenamiento, considerando que las arquitecturas que incorporan bloques *Transformers* tienden a requerir una mayor cantidad de información para su optimización, en comparación con las redes convolucionales. Además, mientras que el modelo se ha entrenado utilizando las muestras de únicamente pacientes control, se ha comprobado la robustez y el desempeño de este ante la estimación de nuevos conjuntos de datos con características de extracción distintas y en pacientes con patologías o alteraciones cerebrales. En este último caso, los bloques convolucionales resultaron preservar una mayor precisión ante estos cambios. En cualquier situación, estos resultados representan un avance en el uso e implementación de modelos basados en *Transformers* para el procesamiento de imágenes médicas, puesto que se ha demostrado su capacidad de modelar estructuras altamente complejas dentro de una imagen. Esto representa una aportación positiva en la precisión y velocidad en la estimación de estos mapas para la detección temprana de enfermedades cerebrales y el estudio de la conectividad cerebral mediante algoritmos de aprendizaje automático que aprovechan los mecanismos de atención.

Ahora, si bien las estimaciones obtenidas por el modelo utilizado en este trabajo no han superado el desempeño de modelos propuestos recientemente, tal como el modelo *SuperDTI* [96] o el modelo *Transformer-DTI* [85], la arquitectura implementada representa una aportación en el estudio de la integración de modelos basados en autoatención en redes neuronales profundas, cómo su comportamiento difiere a las redes convolucionales y su potencial aplicación en el procesamiento de imágenes médicas. En definitiva, es importante mencionar que la construcción de la arquitectura propuesta ha estado limitada al conjunto de datos utilizado para el entrenamiento de la red. El tamaño de este conjunto de datos ha sido relativamente reducido, en contraste al tamaño de las bases de información utilizadas en los modelos mencionados anteriormente y, sin duda, este parámetro es de relevancia en la precisión y robustez del modelo implementado.

Capítulo 5

Conclusiones y Trabajo Futuro

En este último capítulo se establecen las conclusiones generales del trabajo de investigación y las observaciones realizadas para los resultados que se han obtenido sobre distintos conjuntos de prueba que se han descritos en el capítulo anterior. Se mencionan las aportaciones relevantes realizadas por la misma y se presentan las propuestas de trabajos futuros para continuar con esta línea de investigación.

5.1. Conclusiones

La estimación del tensor de difusión en imágenes de resonancia magnética ha emergido como una herramienta fundamental en el estudio de la compleja microestructura de los tractos en la materia blanca cerebral. Esta técnica, que se basa en la cuantificación y visualización de las direcciones y magnitudes de la difusión del agua en los tejidos cerebrales, proporciona información sobre la integridad de las fibras nerviosas y la conectividad cerebral como ningún otro estudio médico. Así, a medida que la tecnología de resonancia magnética avanza y las demandas clínicas se vuelven más específicas, es necesario que las metodologías de estimación del tensor de difusión sean precisas, rápidas y robustas, permitiendo tanto la detección temprana de afecciones neurológicas como el monitoreo eficiente en la progresión de enfermedades. En este sentido, la integración de módulos basados en mecanismos de atención, en particular los bloques *Transformer*, en redes neuronales profundas para el procesamiento de imágenes DWI ha mostrado ser una estrategia prometedora para la estimación eficiente de los mapas derivados del tensor de difusión, tal como la anisotropía fraccional, la difusividad media o las direcciones de difusión. Entonces, por medio de la presente investigación, se ha demostrado que estos módulos permiten modelar relaciones complejas entre las señales de difusión y el tensor de difusión, optimizando tanto la calidad de las estimaciones como la eficiencia computacional. Además, la red implementada ha aprendido a modelar las relaciones de la etapa de procesamiento previo al ajuste del tensor de difusión, tal como la corrección de corrientes de Eddy y la extracción del tejido no cerebral en la imagen. De este modo, se ha desarrollado un marco metodológico para la construcción de una arquitectura de redes neuronales profundas, tipo *autoencoder* basada en mecanismos

de atención, para la estimación de distintos mapas del tensor de difusión a partir de 6 imágenes con direcciones distintas de difusión y una señal sin dirección de difusión, reduciendo tanto el número de muestras necesarias para ajustar el tensor como el tiempo de cómputo requerido para procesar todas las imágenes de un volumen completo de un cerebro.

De este modo, utilizando la librería FSL se han procesado dos conjuntos de las bases de datos del Proyecto de Conectoma Humano y de la Iniciativa de Neuroimágenes para la Enfermedad de Alzheimer, del cual se han obtenido muestras de un total de 65 pacientes control, y se han calculado los mapas del tensor de difusión de interés en este trabajo de investigación. Es importante tener en cuenta que, previamente al ajuste del tensor para cada caso, se ha efectuado un procedimiento estándar para acondicionar las imágenes ponderadas por difusión en el que se realiza una extracción del ruido blanco de las imágenes, se corrige cualquier movimiento del paciente entre las señales, se corrigen artefactos producidos por corrientes de Eddy y anillos de Gibbs y se extrae el tejido cerebral en las imágenes, removiendo cualquier parte en las imágenes perteneciente al cráneo, ojos y músculos. Además, se han utilizado las señales en las direcciones más cercanas sobre los ejes coordenados de las imágenes, siendo un total de 6 señales de difusión. Sin embargo, es posible realizar una selección de estas señales de forma que los datos de entrada contengan más información respecto a la difusión del agua en el tejido cerebral y que, por tanto, mejore el desempeño del modelo propuesto. No obstante, esto conduce a un algoritmo de mayor complejidad y ha quedado fuera del enfoque de la presente investigación.

Para terminar, la robustez de la arquitectura propuesta se ha validado ante escenario de datos con protocolos de adquisición distintos y ante patologías o alteraciones cerebrales, y se ha demostrado una alta capacidad de adaptabilidad y versatilidad de este método en contextos clínicos. Sin embargo, es esencial considerar que el rendimiento final de la red ha estado condicionado por el tamaño del conjunto de datos de entrenamiento. En el futuro, el uso de conjuntos de datos más extensos podría ser crucial para mejorar la precisión y robustez de modelos similares. Así, aunque la arquitectura propuesta no superó el desempeño de otros métodos presentados en la literatura, este aporta resultados experimentales valiosos sobre el potencial y los desafíos actuales al integrar los mecanismos de atención en redes profundas para aplicaciones en imágenes médicas. Además, las arquitecturas compuestas por *Transformers* tienen la ventaja de ser altamente escalables y eficientes en términos de recursos computacionales. Un hallazgo importante de esta investigación es que, mientras que las arquitecturas basadas en mecanismos de atención pueden requerir bases de datos de mayor tamaño para su completa optimización en comparación con redes convolucionales, el potencial de paralelización y procesamiento en tiempo real que ofrecen es significativo. Finalmente, este estudio resalta la importancia y la necesidad de continuar explorando la integración de módulos basados en *Transformers* en el ámbito del procesamiento de imágenes médicas, dado su potencial para mejorar tanto la calidad como la eficiencia en la estimación de parámetros

vitales para el diagnóstico y monitoreo de enfermedades neurodegenerativas.

5.2. Principales aportaciones

La presente investigación ha realizado varias contribuciones importantes en el campo de la estimación de parámetros de difusión en imágenes de resonancia magnética utilizando redes neuronales basadas en *Transformers*. En primer lugar, se ha propuesto e implementado la arquitectura *UT-Net*, que ha demostrado ser altamente efectiva en la segmentación de imágenes médicas, también lo ha sido en la estimación del mapa de anisotropía fraccional, difusividad media y orientaciones de difusión. Esta arquitectura ha permitido obtener resultados superiores en términos de exactitud y eficiencia de las estimaciones en comparación con el enfoque convencional. Además, se han utilizado métricas de desempeño bien establecidas, como el error cuadrático medio normalizado, el índice de similitud estructural y la proporción máxima de señal a ruido, para validar y cuantificar el rendimiento del modelo propuesto, lo que representa todo un marco de diseño para este tipo de modelos. Por último, se han presentado resultados experimentales del efecto de los mecanismos de atención en arquitecturas tipo *autoencoder*. Estas aportaciones en conjunto representan avances significativos en el campo de la estimación del tensor de difusión y ofrecen nuevas perspectivas y herramientas para la investigación en el estudio de la microestructura cerebral.

5.3. Trabajos futuros

Para terminar, se presentan las propuestas de trabajos futuros para continuar con esta línea de investigación. Principalmente, se ha encontrado que se puede mejorar la arquitectura *UT-Net* implementada, puesto que, aunque la arquitectura ha demostrado un rendimiento prometedor, aún existen oportunidades para su mejora. Se puede estudiar la incorporación de capas adicionales, ajustar los hiperparámetros mediante algoritmos de optimización, como modelos evolutivos, o explorar variantes de *Transformers* para mejorar el modelo y aumentar su capacidad de estimación. De la misma forma, a continuación, se enumeran posibles líneas de investigación que puedan ser inspiradas por este trabajo.

1. Exploración de otros conjuntos de datos: se han utilizado imágenes de referencia generadas mediante FSL para entrenar el modelo. Una propuesta interesante es ampliar la investigación utilizando conjuntos de datos más diversos y de mayor tamaño. Esto permitirá evaluar el rendimiento del modelo en diferentes contextos y aumentar su generalización a distintas poblaciones o condiciones patológicas.

2. Validación clínica: para llevar el enfoque propuesto a la práctica médica, es necesario realizar una validación clínica exhaustiva. Se pueden realizar estudios comparativos con técnicas existentes, evaluar el rendimiento en diferentes escenarios clínicos y explorar la utilidad del modelo en la detección y seguimiento de enfermedades neurodegenerativas u otras afecciones cerebrales.

3. Integración con técnicas de adquisición avanzadas: en lugar de utilizar imágenes de referencia generadas por FSL, se puede explorar la integración directa del modelo propuesto con técnicas de adquisición avanzadas, como la resonancia magnética de difusión de alta angularidad (HARDI) o imágenes por curtosis de difusión (DKI). Esto permitiría una estimación de parámetros de mayor precisión y robustez al utilizar información de adquisición más completa.

4. Aplicación a otros dominios de imagen: este trabajo se ha enfocado en imágenes de resonancia magnética del cerebro, el enfoque basado en *Transformers* podría aplicarse a otros dominios de imagen, como la resonancia magnética corporal, la tomografía computarizada o la microscopía. Investigar la adaptación y la transferencia del modelo a estos dominios podría ampliar su aplicabilidad y abrir nuevas oportunidades de investigación.

Estas propuestas de trabajos futuros permitirán seguir avanzando en el campo de la estimación de parámetros del tensor de difusión utilizando redes neuronales basadas en *Transformers* y explorar nuevas aplicaciones y mejoras en este campo.

Referencias

[1] O. I. Abiodun et al. «State-of-the-art in artificial neural network applications: A survey». En: *Heliyon* 4.11 (nov. de 2018), e00938. DOI: 10.1016/j.heliyon.2018.e00938.

[2] J. Andersson et al. «A comprehensive Gaussian process framework for correcting distortions and movements in diffusion images». En: *20th ISMRM* (ene. de 2012).

[3] R. Ansorge y M. Graves. *Physics and Mathematics of MRI*. Morgan & Claypool Publishers, 2016. ISBN: 9781681740683.

[4] S. Aslani y J. Jacob. «Utilisation of deep learning for COVID-19 diagnosis». En: *Clinical Radiology* 78.2 (feb. de 2023), págs. 150-157. DOI: 10.1016/j.crad.2022.11.006.

[5] Y. Assaf y P. J. Basser. «Composite hindered and restricted model of diffusion (CHARMED) MR imaging of the human brain». En: *NeuroImage* 27.1 (ago. de 2005), págs. 48-58. DOI: 10.1016/j.neuroimage.2005.03.042.

[6] D. Bahdanau, K. Cho e Y. Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2014. DOI: 10.48550/ARXIV.1409.0473.

[7] V. Baliyan et al. «Diffusion weighted imaging: Technique and applications». En: *World Journal of Radiology* 8.9 (2016), pág. 785. DOI: 10.4329/wjr.v8.i9.785.

[8] R. Bammer. «Basic principles of diffusion-weighted imaging». En: *European Journal of Radiology* 45.3 (mar. de 2003), págs. 169-184. DOI: 10.1016/s0720-048x(02)00303-0.

[9] T. Beppu. En: *Journal of Neuro-Oncology* 63.2 (2003), págs. 109-116. DOI: 10.1023/a:1023977520909.

[10] B. A. Berkowitz et al. «QUEST MRI assessment of fetal brain oxidative stress in utero». En: *NeuroImage* 200 (oct. de 2019), págs. 601-606. DOI: 10.1016/j.neuroimage.2019.05.069.

[11] S. Berryman. «Euclid and the Sceptic: A Paper on Vision, Doubt, Geometry, Light and Drunkenness». En: *Phronesis* 43.2 (1998), págs. 176-196. DOI: 10.1163/15685289860511078.

[12] D. L. Bihan y H. Johansen-Berg. «Diffusion MRI at 25: Exploring brain tissue structure and function». En: *NeuroImage* 61.2 (jun. de 2012), págs. 324-341. DOI: 10.1016/j.neuroimage.2011.11.006.

[13] D. L. Bihan et al. «Diffusion tensor imaging: Concepts and applications». En: *Journal of Magnetic Resonance Imaging* 13.4 (2001), págs. 534-546. DOI: 10.1002/jmri.1076.

[14] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2008, pág. 738. ISBN: 9780387310732.

[15] C.M. Bishop. *Pattern Recognition and Machine Learning: All "just the Facts 101"Material*. Information science and statistics. Springer (India) Private Limited, 2013. ISBN: 9788132209065.

[16] K. T. Block, M. Uecker y J. Frahm. «Suppression of MRI Truncation Artifacts Using Total Variation Constrained Data Extrapolation». En: *International Journal of Biomedical Imaging* 2008 (2008), págs. 1-8. DOI: 10.1155/2008/184123.

[17] D. Bonekamp et al. «Radiomic Machine Learning for Characterization of Prostate Lesions with MRI: Comparison to ADC Values». En: *Radiology* 289.1 (oct. de 2018), págs. 128-137. DOI: 10.1148/radiol.2018173064.

[18] E. Bonet-Carne et al. «VERDICT-AMICO: Ultrafast fitting algorithm for non-invasive prostate microstructure characterization». En: *NMR in Biomedicine* 32.1 (oct. de 2018), e4019. DOI: 10.1002/nbm.4019.

[19] K. Borkowski y A. T. Krzyżak. «The generalized Stejskal-Tanner equation for non-uniform magnetic field gradients». En: *Journal of Magnetic Resonance* 296 (nov. de 2018), págs. 23-28. DOI: 10.1016/j.jmr.2018.08.010.

[20] A. Boudet et al. «Limitation of Screening of Different Variants of SARS-CoV-2 by RT-PCR». En: *Diagnostics* 11.7 (jul. de 2021), pág. 1241. DOI: 10.3390/diagnostics11071241.

[21] R. Bourne y E. Panagiotaki. «Limitations and Prospects for Diffusion-Weighted MRI of the Prostate». En: *Diagnostics* 6.2 (mayo de 2016), pág. 21. DOI: 10.3390/diagnostics6020021.

[22] R. M. Bourne et al. «Microscopic diffusion anisotropy in formalin fixed prostate tissue: Preliminary findings». En: *Magnetic Resonance in Medicine* 68.6 (ene. de 2012), págs. 1943-1948. DOI: 10.1002/mrm.24179.

[23] G. Brauwers y F. Frasincar. «A General Survey on Attention Mechanisms in Deep Learning». En: *IEEE Transactions on Knowledge and Data Engineering* (2021), págs. 1-1. DOI: 10.1109/tkde.2021.3126456.

[24] R. W. Brown et al. *Magnetic Resonance Imaging*. Wiley, abr. de 2014. DOI: 10.1002/9781118633953.

[25] T. B. Brown et al. *Language Models are Few-Shot Learners*. 2020. DOI: 10.48550/ARXIV.2005.14165.

[26] H. Cao et al. «Swin-Unet: Unet-Like Pure Transformer for Medical Image Segmentation». En: *Lecture Notes in Computer Science*. Springer Nature Switzerland, 2023, págs. 205-218. DOI: 10.1007/978-3-031-25066-8_9.

[27] N. Carion et al. *End-to-End Object Detection with Transformers*. 2020. DOI: 10.48550/ARXIV.2005.12872.

[28] D. Chen et al. «Video Person Re-identification with Competitive Snippet-Similarity Aggregation and Co-attentive Snippet Embedding». En: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, jun. de 2018. DOI: 10.1109/cvpr.2018.00128.

[29] J. Chen et al. *TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation*. 2021. DOI: 10.48550/ARXIV.2102.04306.

[30] W. Chen et al. «Visualizing diffusion tensor imaging data with merging ellipsoids». En: *2009 IEEE Pacific Visualization Symposium*. IEEE, abr. de 2009. DOI: 10.1109/pacificvis.2009.4906849.

[31] Z. W. Chen et al. «Assessment of breast lesions by the Kaiser score for differential diagnosis on MRI: the added value of ADC and machine learning modeling». En: *European Radiology* 32.10 (jun. de 2022), págs. 6608-6618. DOI: 10.1007/s00330-022-08899-w.

[32] X. Chu et al. «Comparison of brain microstructure alterations on diffusion kurtosis imaging among Alzheimer's disease, mild cognitive impairment, and cognitively normal individuals». En: *Frontiers in Aging Neuroscience* 14 (ago. de 2022). DOI: 10.3389/fnagi.2022.919143.

[33] R. Collobert, K. Kavukcuoglu y C. Farabet. «Torch7: A Matlab-like Environment for Machine Learning». En: *BigLearn, NIPS Workshop*. 2011.

[34] L. Cordero-Grande et al. «Complex diffusion-weighted image estimation via matrix recovery under general noise models». En: *NeuroImage* 200 (oct. de 2019), págs. 391-404. DOI: 10.1016/j.neuroimage.2019.06.039.

[35] B. Cyganek y J. P. Siebert. *An introduction to 3D Computer Vision Techniques and Algorithms*. Vol. 1. 1. John Wiley & Sons, 2009.

[36] C. Da Costa-Luis et al. *tqdm: A fast, Extensible Progress Bar for Python and CLI*. 2023. DOI: 10.5281/ZENODO.8233425.

[37] B. M. Dale, M. A. Brown y R. C. Semelka. *MRI Basic Principles and Applications*. Wiley, ago. de 2015. DOI: 10.1002/9781119013068.

[38] M. Daniluk et al. «Frustratingly Short Attention Spans in Neural Language Modeling». En: *International Conference on Learning Representations*. 2017. URL: https://openreview.net/forum?id=ByIAPUcee.

[39] F. Dell'Acqua y J. D. Tournier. «Modelling white matter with spherical deconvolution: How and why?» En: *NMR in Biomedicine* 32.4 (ago. de 2018). DOI: 10.1002/nbm.3945.

[40] N. Demian et al. «Surgical Navigation for Oral and Maxillofacial Surgery». En: *Oral and Maxillofacial Surgery Clinics of North America* 31.4 (nov. de 2019), págs. 531-538. DOI: 10.1016/j.coms.2019.06.001.

[41] S. C.L. Deoni et al. «Standardized structural magnetic resonance imaging in multicentre studies using quantitative T 1 and T 2 imaging at 1.5 T». En: *NeuroImage* 40.2 (abr. de 2008), págs. 662-671. DOI: 10.1016/j.neuroimage.2007.11.052.

[42] A. Dosovitskiy et al. «An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale». En: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=YicbFdNTTy.

[43] H. Duffau. «Stimulation mapping of white matter tracts to study brain functional connectivity». En: *Nature Reviews Neurology* 11.5 (abr. de 2015), págs. 255-265. DOI: 10.1038/nrneurol.2015.51.

[44] O. Esteban et al. «fMRIPrep: a robust preprocessing pipeline for functional MRI». En: *Nature Methods* 16.1 (dic. de 2018), págs. 111-116. DOI: 10.1038/s41592-018-0235-4.

[45] E. Europa et al. «Neural Connectivity in Syntactic Movement Processing». En: *Frontiers in Human Neuroscience* 13 (feb. de 2019). DOI: 10.3389/fnhum.2019.00027.

[46] C. M. Fan, T. J. Liu y K. H. Liu. «SUNet: Swin Transformer UNet for Image Denoising». En: *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, mayo de 2022. DOI: 10.1109/iscas48785.2022.9937486.

[47] A. F. A. Fernandes et al. «Comparison of data analytics strategies in computer vision systems to predict pig body composition traits from 3D images». En: *Journal of Animal Science* 98.8 (ago. de 2020). DOI: 10.1093/jas/skaa250.

[48] C. R. Figley et al. «Potential Pitfalls of Using Fractional Anisotropy, Axial Diffusivity, and Radial Diffusivity as Biomarkers of Cerebral White Matter Microstructure». En: *Frontiers in Neuroscience* 15 (ene. de 2022). DOI: 10.3389/fnins.2021.799576.

[49] A. Filler. «The History, Development and Impact of Computed Imaging in Neurological Diagnosis and Neurosurgery: CT, MRI, and DTI». En: *Nature Precedings* (mayo de 2009). DOI: 10.1038/npre.2009.3267.1.

[50] A. D. Friederici. «Pathways to language: fiber tracts in the human brain». En: *Trends in Cognitive Sciences* 13.4 (abr. de 2009), págs. 175-181. DOI: 10.1016/j.tics.2009.01.001.

[51] A. F. Gad. *Practical Computer Vision Applications Using Deep Learning with CNNs*. Apress, 2018. DOI: 10.1007/978-1-4842-4167-7.

[52] D. V. Gadasin, A. V. Shvedov e I. A. Kuzin. «Reconstruction of a Three-Dimensional Scene from its Projections in Computer Vision Systems». En: *2021 Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED)*. IEEE, nov. de 2021. DOI: 10.1109/tirved53476.2021.9639161.

[53] Y. Gao, M. Zhou y D. Metaxas. *UTNet: A Hybrid Transformer Architecture for Medical Image Segmentation*. 2021. DOI: 10.48550/ARXIV.2107.00781.

[54] Z. Gao et al. «Global Second-Order Pooling Convolutional Networks». En: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun. de 2019. DOI: 10.1109/cvpr.2019.00314.

[55] E. Garyfallidis et al. «Dipy, a library for the analysis of diffusion MRI data». En: *Frontiers in Neuroinformatics* 8 (feb. de 2014). DOI: 10.3389/fninf.2014.00008.

[56] J. Gehring et al. *Convolutional Sequence to Sequence Learning*. 2017. DOI: 10.48550/ARXIV.1705.03122.

[57] M. F. Glasser et al. «The minimal preprocessing pipelines for the Human Connectome Project». En: *NeuroImage* 80 (oct. de 2013), págs. 105-124. DOI: 10.1016/j.neuroimage.2013.04.127.

[58] A. Goel y U. Bashir. *Diffusion-weighted imaging*. Feb. de 2012. DOI: 10.53347/rid-16718.

[59] M. Gokhale, S. K. Mohanty y A. Ojha. «GeneViT: Gene Vision Transformer with Improved DeepInsight for cancer classification». En: *Computers in Biology and Medicine* 155 (mar. de 2023), pág. 106643. DOI: 10.1016/j.compbiomed.2023.106643.

[60] I. Goodfellow, Y. Bengio y A. Courville. *Deep Learning*. deeplearningbook.org. MIT Press, 2016.

[61] D. N. Greve y B. Fischl. «Accurate and robust brain image alignment using boundary-based registration». En: *NeuroImage* 48.1 (oct. de 2009), págs. 63-72. DOI: 10.1016/j.neuroimage.2009.06.060.

[62] M. H. Guo et al. «Attention mechanisms in computer vision: A survey». En: *Computational Visual Media* 8.3 (mar. de 2022), págs. 331-368. DOI: 10.1007/s41095-022-0271-y.

[63] P. Hagmann et al. «Understanding Diffusion MR Imaging Techniques: From Scalar Diffusion-weighted Imaging to Diffusion Tensor Imaging and Beyond». En: *RadioGraphics* 26.suppl_1 (oct. de 2006), S205-S223. DOI: 10.1148/rg.26si065510.

[64] P. Hamou. «Vision, Color, and Method in Newton's Opticks». En: *Newton and Empiricism*. Oxford University Press, jun. de 2014, págs. 66-94. DOI: 10.1093/acprof:oso/9780199337095.003.0004.

[65] J. T. Hancock y T. M. Khoshgoftaar. «Survey on categorical data for neural networks». En: *Journal of Big Data* 7.1 (abr. de 2020). DOI: 10.1186/s40537-020-00305-w.

[66] C. R. Harris et al. «Array programming with NumPy». En: *Nature* 585.7825 (sep. de 2020), págs. 357-362. DOI: 10.1038/s41586-020-2649-2. URL: 10.1038/s41586-020-2649-2.

[67] N. Hata. «Surgical Navigation Technology». En: *Intraoperative Imaging and Image-Guided Therapy*. Springer New York, nov. de 2013, págs. 249-257. DOI: 10.1007/978-1-4614-7657-3_17.

[68] J. He et al. «A unified global tractography framework for automatic visual pathway reconstruction». En: *NMR in Biomedicine* (feb. de 2023). DOI: 10.1002/nbm.4904.

[69] K. He et al. *Deep Residual Learning for Image Recognition*. 2015. DOI: 10.48550/ARXIV.1512.03385.

[70] A. Horé y D. Ziou. «Is there a relationship between peak-signal-to-noise ratio and structural similarity index measure?» En: *IET Image Processing* 7.1 (feb. de 2013), págs. 12-24. DOI: 10.1049/iet-ipr.2012.0489.

[71] S.M.H. Hosseini et al. «CTtrack: A CNN+ Transformer-based framework for fiber orientation estimation & tractography». En: *Neuroscience Informatics* 2.4 (dic. de 2022), pág. 100099. DOI: 10.1016/j.neuri.2022.100099.

[72] J. Hu, L. Shen y G. Sun. «Squeeze-and-Excitation Networks». En: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, jun. de 2018. DOI: 10.1109/cvpr.2018.00745.

[73] J. Hu et al. «Voronoi-Based Multi-Robot Autonomous Exploration in Unknown Environments via Deep Reinforcement Learning». En: *IEEE Transactions on Vehicular Technology* 69.12 (dic. de 2020), págs. 14413-14423. DOI: 10.1109/tvt.2020.3034800.

[74] G. Huang et al. *Deep Networks with Stochastic Depth*. 2016. DOI: 10.48550/ARXIV.1603.09382.

[75] M. Hussain, D. Koundal y J. Manhas. «Deep learning-based diagnosis of disc degenerative diseases using MRI: A comprehensive review». En: *Computers and Electrical Engineering* 105 (ene. de 2023), pág. 108524. DOI: 10.1016/j.compeleceng.2022.108524.

[76] M. Jaderberg et al. *Spatial Transformer Networks*. 2015. DOI: 10.48550/ARXIV.1506.02025.

[77] G. James. *An Introduction To Statistical Learning With Applications In R*. Springer-Verlag New York Inc., 2013. ISBN: 9781461471370.

[78] M. Jenkinson et al. «FSL». En: *NeuroImage* 62.2 (ago. de 2012), págs. 782-790. DOI: 10.1016/j.neuroimage.2011.09.015.

[79] J. H. Jensen et al. «Diffusional kurtosis imaging: The quantification of non-gaussian water diffusion by means of magnetic resonance imaging». En: *Magnetic Resonance in Medicine* 53.6 (2005), págs. 1432-1440. DOI: 10.1002/mrm.20508.

[80] B. Jeurissen et al. «Multi-tissue constrained spherical deconvolution for improved analysis of multi-shell diffusion MRI data». En: *NeuroImage* 103 (dic. de 2014), págs. 411-426. DOI: 10.1016/j.neuroimage.2014.07.061.

[81] T. K. Jhamb, V. Rejathalal y V.K. Govindan. «A Review on Image Reconstruction through MRI k-Space Data». En: *International Journal of Image, Graphics and Signal Processing* 7.7 (jun. de 2015), págs. 42-59. DOI: 10.5815/ijigsp.2015.07.06.

[82] D. K. Jones, T. R. Knösche y R. Turner. «White matter integrity, fiber count, and other fallacies: The do's and don'ts of diffusion MRI». En: *NeuroImage* 73 (jun. de 2013), págs. 239-254. DOI: 10.1016/j.neuroimage.2012.06.081.

[83] D. Jurafsky y J. H. Martin. *Speech and language processing. an introduction to natural language processing, computational linguistics, and speech recognition*. Draft, 2020.

[84] K. Kantarci. «Fractional Anisotropy of the Fornix and Hippocampal Atrophy in Alzheimer's Disease». En: *Frontiers in Aging Neuroscience* 6 (nov. de 2014). DOI: 10.3389/fnagi.2014.00316.

[85] D. Karimi y A. Gholipour. «Diffusion tensor estimation with transformer neural networks». En: *Artificial Intelligence in Medicine* 130 (ago. de 2022), pág. 102330. DOI: 10.1016/j.artmed.2022.102330.

[86] P. G. Kele. «Diffusion weighted imaging in the liver». En: *World Journal of Gastroenterology* 16.13 (2010), pág. 1567. DOI: 10.3748/wjg.v16.i13.1567.

[87] E. Kellner et al. «Gibbs-ringing artifact removal based on local subvoxel-shifts». En: *Magnetic Resonance in Medicine* 76.5 (nov. de 2015), págs. 1574-1581. DOI: 10.1002/mrm.26054.

[88] S. Khan et al. «Transformers in Vision: A Survey». En: *ACM Computing Surveys* 54.10s (ene. de 2022), págs. 1-41. DOI: 10.1145/3505244.

[89] Y. Kim et al. «Structured attention networks». En: *International Conference on Learning Representations* (2017).

[90] S. Kiranyaz et al. «1D convolutional neural networks and applications: A survey». En: *Mechanical Systems and Signal Processing* 151 (abr. de 2021), pág. 107398. DOI: 10.1016/j.ymssp.2020.107398.

[91] P. Kochunov et al. «Fractional anisotropy of water diffusion in cerebral white matter across the lifespan». En: *Neurobiology of Aging* 33.1 (ene. de 2012), págs. 9-20. DOI: 10.1016/j.neurobiolaging.2010.01.014.

[92] J. Korhonen y J. You. «Peak signal-to-noise ratio revisited: Is simple beautiful?» En: *2012 Fourth International Workshop on Quality of Multimedia Experience*. IEEE, jul. de 2012. DOI: 10.1109/qomex.2012.6263880.

[93] L. Kumaralingam et al. «Segmentation of Whole-Brain Tractography: A Deep Learning Algorithm Based on 3D Raw Curve Points». En: *Lecture Notes in Computer Science*. Springer Nature Switzerland, 2022, págs. 185-195. DOI: 10.1007/978-3-031-16431-6_18.

[94] G. Larose et al. «High intraoperative accuracy and low complication rate of computer-assisted navigation of the glenoid in total shoulder arthroplasty». En: *Journal of Shoulder and Elbow Surgery* (ene. de 2023). DOI: 10.1016/j.jse.2022.12.021.

[95] J. H. Legarreta et al. «Generative Sampling in Bundle Tractography using Autoencoders (GESTA)». En: *Medical Image Analysis* 85 (abr. de 2023), pág. 102761. DOI: 10.1016/j.media.2023.102761.

[96] H. Li, Z. Liang y C. Zhang y col. «SuperDTI: Ultrafast DTI and fiber tractography with deep learning». En: *International Society for Magnetic Resonance in Medicine* (jul. de 2021).

[97] K. Li et al. «Fractional anisotropy alterations in individuals born preterm: a diffusion tensor imaging meta-analysis». En: *Developmental Medicine & Child Neurology* 57.4 (oct. de 2014), págs. 328-338. DOI: 10.1111/dmcn.12618.

[98] Y. Li et al. «Exploring Plain Vision Transformer Backbones for Object Detection». En: *Lecture Notes in Computer Science*. Springer Nature Switzerland, 2022, págs. 280-296. DOI: 10.1007/978-3-031-20077-9_17.

[99] Z. Li et al. «A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects». En: *IEEE Transactions on Neural Networks and Learning Systems* 33.12 (dic. de 2022), págs. 6999-7019. DOI: 10.1109/tnnls.2021.3084827.

[100] S. Lifshits, A. Tamir e Y. Assaf. «Combinatorial fiber-tracking of the human brain». En: *NeuroImage* 48.3 (nov. de 2009), págs. 532-540. DOI: 10.1016/j.neuroimage.2009.05.086.

[101] Z. Lin et al. *A Structured Self-attentive Sentence Embedding*. 2017. DOI: 10.48550/ARXIV.1703.03130.

[102] H. Liu et al. «Diffusion kurtosis imaging and diffusion tensor imaging parameters applied to white matter and gray matter of patients with anti-N-methyl-D-aspartate receptor encephalitis». En: *Frontiers in Neuroscience* 16 (nov. de 2022). DOI: 10.3389/fnins.2022.1030230.

[103] T. Liu, E. Siegel y D. Shen. «Deep Learning and Medical Image Analysis for COVID-19 Diagnosis and Prediction». En: *Annual Review of Biomedical Engineering* 24.1 (jun. de 2022), págs. 179-201. DOI: 10.1146/annurev-bioeng-110220-012203.

[104] Z. Liu et al. «Quality control of diffusion weighted images». En: *Medical Imaging 2010: Advanced PACS-based Imaging Informatics and Therapeutic Applications*. Ed. por Brent J. Liu y William W. Boonn. SPIE, mar. de 2010. DOI: 10.1117/12.844748.

[105] Z. Liu et al. *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows*. 2021. arXiv: 2103.14030 [cs.CV].

[106] T. Luong, H. Pham y C. D. Manning. «Effective Approaches to Attention-based Neural Machine Translation». En: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2015. DOI: 10.18653/v1/d15-1166.

[107] M. Maaz et al. «EdgeNeXt: Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications». En: *Lecture Notes in Computer Science*. Springer Nature Switzerland, 2023, págs. 3-20. DOI: 10.1007/978-3-031-25082-8_1.

[108] TorchVision maintainers y contributors. *TorchVision: PyTorch's Computer Vision library*. https://github.com/pytorch/vision. 2016.

[109] L. Mannelli et al. «Advances in Diffusion-Weighted Imaging». En: *Radiologic Clinics of North America* 53.3 (mayo de 2015), págs. 569-581. DOI: 10.1016/j.rcl.2015.01.002.

[110] F. A. Mansouri et al. «Managing competing goals — a key role for the frontopolar cortex». En: *Nature Reviews Neuroscience* 18.11 (sep. de 2017), págs. 645-657. DOI: 10.1038/nrn.2017.111.

[111] D. S. Marcus et al. «Informatics and Data Mining Tools and Strategies for the Human Connectome Project». En: *Frontiers in Neuroinformatics* 5 (2011). DOI: 10.3389/fninf.2011.00004.

[112] D. G. McLaren et al. «A population-average MRI-based atlas collection of the rhesus macaque». En: *NeuroImage* 45.1 (mar. de 2009), págs. 52-59. DOI: 10.1016/j.neuroimage.2008.10.058.

[113] U. Mezger, C. Jendrewski y M. Bartels. «Navigation in surgery». En: *Langenbeck's Archives of Surgery* 398.4 (feb. de 2013), págs. 501-514. DOI: 10.1007/s00423-013-1059-4.

[114] V. Mnih et al. *Recurrent Models of Visual Attention*. 2014. DOI: 10.48550/ARXIV.1406.6247.

[115] D. Moratal et al. «k-Space tutorial: an MRI educational tool for a better understanding of k-space». En: *Biomedical Imaging and Intervention Journal* 4.1 (ene. de 2008). DOI: 10.2349/biij.4.1.e15.

[116] S. Mori y J. Zhang. «Principles of Diffusion Tensor Imaging and Its Applications to Basic Neuroscience Research». En: *Neuron* 51.5 (sep. de 2006), págs. 527-539. DOI: 10.1016/j.neuron.2006.08.012.

[117] A. Murphy y F. Gaillard. *MRI sequences (overview)*. Jun. de 2015. DOI: 10.53347/rid-37346.

[118] P. Mörters e Y. Peres. *Brownian Motion*. Cambridge University Press, ene. de 2001. DOI: 10.1017/cbo9780511750489.

[119] Y. Nazarathy y H. Klok. *Statistics with Julia. Fundamentals for Data Science, Machine Learning and Artificial Intelligence*. Springer International Publishing AG, 2021. ISBN: 9783030709006.

[120] I. Nelkenbaum et al. «Automatic Segmentation of White Matter Tracts Using Multiple Brain MRI Sequences». En: *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*. IEEE. 2020, págs. 368-371.

[121] I. E. Nielsen et al. «Robust Explainability: A tutorial on gradient-based attribution methods for deep neural networks». En: *IEEE Signal Processing Magazine* 39.4 (jul. de 2022), págs. 73-84. DOI: 10.1109/msp.2022.3142719.

[122] T. Nowotny, J. P. Turner y J. C. Knight. *Loss shaping enhances exact gradient learning with EventProp in Spiking Neural Networks*. 2022. DOI: 10.48550/ARXIV.2212.01232.

[123] I. Oguz et al. «DTIPrep: quality control of diffusion-weighted images». En: *Frontiers in Neuroinformatics* 8 (2014). DOI: 10.3389/fninf.2014.00004.

[124] D. A. Borges Oliveira et al. «A review of deep learning algorithms for computer vision systems in livestock». En: *Livestock Science* 253 (nov. de 2021), pág. 104700. DOI: 10.1016/j.livsci.2021.104700.

[125] K. O'Shea y R. Nash. *An Introduction to Convolutional Neural Networks*. 2015. DOI: 10.48550/ARXIV.1511.08458.

[126] O. Ozyurt et al. «Integration of arterial spin labeling into stereotactic radiosurgery planning of cerebral arteriovenous malformations». En: *Journal of Magnetic Resonance Imaging* 46.6 (mar. de 2017), págs. 1718-1727. DOI: 10.1002/jmri.25690.

[127] S. C. Partridge et al. «Diffusion-weighted breast MRI: Clinical applications and emerging techniques». En: *Journal of Magnetic Resonance Imaging* 45.2 (sep. de 2016), págs. 337-355. DOI: 10.1002/jmri.25479.

[128] A. Parvaiz et al. «Vision Transformers in medical computer vision—A contemplative retrospection». En: *Engineering Applications of Artificial Intelligence* 122 (jun. de 2023), pág. 106126. DOI: 10.1016/j.engappai.2023.106126.

[129] A. Paszke et al. «PyTorch: An Imperative Style, High-Performance Deep Learning Library». En: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, págs. 8024-8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[130] F. Pedregosa et al. «Scikit-learn: Machine Learning in Python». En: *Journal of Machine Learning Research* 12 (2011), págs. 2825-2830.

[131] J. Perez-Gonzalez et al. «Mild cognitive impairment classification using combined structural and diffusion imaging biomarkers». En: *Physics in Medicine & Biology* 66.15 (jul. de 2021), pág. 155010. DOI: 10.1088/1361-6560/ac0e77.

[132] A. Pfefferbaum, E. Adalsteinsson y E. V. Sullivan. «Replicability of diffusion tensor imaging measurements of fractional anisotropy and trace in brain». En: *Journal of Magnetic Resonance Imaging* 18.4 (sep. de 2003), págs. 427-433. DOI: 10.1002/jmri.10377.

[133] P. Pietruski et al. «Replacing cutting guides with an augmented reality-based navigation system: A feasibility study in the maxillofacial region». En: *The International Journal of Medical Robotics and Computer Assisted Surgery* (feb. de 2023). DOI: 10.1002/rcs.2499.

[134] D. B. Plewes y W. Kucharczyk. «Physics of MRI: A primer». En: *Journal of Magnetic Resonance Imaging* 35.5 (abr. de 2012), págs. 1038-1054. DOI: 10.1002/jmri.23642.

[135] S. J. Powers. *MRI Physics: Tech to Tech Explanations*. John Wiley & Sons, 2021.

[136] Z. Qu y L. Zhang. «Research on Image Segmentation Based on the Improved Otsu Algorithm». En: *2010 Second International Conference on Intelligent Human-Machine Systems and Cybernetics*. IEEE, ago. de 2010. DOI: 10.1109/ihmsc.2010.157.

[137] S. Raj et al. «Comparative Evaluation of Diffusion Kurtosis Imaging and Diffusion Tensor Imaging in Detecting Cerebral Microstructural Changes in Alzheimer Disease». En: *Academic Radiology* 29 (mar. de 2022), S63-S70. DOI: 10.1016/j.acra.2021.01.018.

[138] G. B. Raja. «Early detection of breast cancer using efficient image processing algorithms and prediagnostic techniques: A detailed approach». En: *Cognitive Systems and Signal Processing in Image Processing*. Elsevier, 2022, págs. 223-251. DOI: 10.1016/b978-0-12-824410-4.00009-x.

[139] H. Ramchoun et al. «Multilayer Perceptron: Architecture Optimization and Training». En: *International Journal of Interactive Multimedia and Artificial Intelligence* 4.1 (2016), pág. 26. DOI: 10.9781/ijimai.2016.415.

[140] P. Ramos-Giraldo et al. «Drought Stress Detection Using Low-Cost Computer Vision Systems and Machine Learning Techniques». En: *IT Professional* 22.3 (mayo de 2020), págs. 27-29. DOI: 10.1109/mitp.2020.2986103.

[141] J. L. Dalboni da Rocha et al. «Fractional Anisotropy changes in Parahippocampal Cingulum due to Alzheimer's Disease». En: *Scientific Reports* 10.1 (feb. de 2020). DOI: 10.1038/s41598-020-59327-2.

[142] A. Rogozhnikov. «Einops: Clear and Reliable Tensor Manipulations with Einstein-like Notation». En: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=oapKSVM2bcj.

[143] O. Ronneberger, P.Fischer y T. Brox. «U-Net: Convolutional Networks for Biomedical Image Segmentation». En: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Vol. 9351. LNCS. (available on arXiv:1505.04597 [cs.CV]). Springer, 2015, págs. 234-241. URL: http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a.

[144] F. Rosenblatt. «The perceptron: A probabilistic model for information storage and organization in the brain.» En: *Psychological Review* 65.6 (1958), págs. 386-408. DOI: 10.1037/h0042519.

[145] N.V. Ruiter et al. «AUTOMATIC IMAGE MATCHING FOR BREAST CANCER DIAGNOSTICS BY A 3D DEFORMATION MODEL OF THE MAMMA». En: *Biomedizinische Technik/Biomedical Engineering* 47.s1b (2002), págs. 644-647. DOI: 10.1515/bmte.2002.47.s1b.644.

[146] S. J. Russell. *Inteligencia artificial un enfoque moderno. un enfoque moderno*. Pearson Prentice Hall, 2004, pág. 1212. ISBN: 9788420540030.

[147] A. Sadikov et al. *Generalized Diffusion MRI Denoising and Super-Resolution using Swin Transformers*. 2023. arXiv: 2303.05686 [eess.IV].

[148] D. R. Sarvamangala y R. V. Kulkarni. «Convolutional neural networks in medical image understanding: a survey». En: *Evolutionary Intelligence* 15.1 (ene. de 2021), págs. 1-22. DOI: 10.1007/s12065-020-00540-3.

[149] J. Schlemper et al. «Attention gated networks: Learning to leverage salient regions in medical images». En: *Medical Image Analysis* 53 (abr. de 2019), págs. 197-207. DOI: 10.1016/j.media.2019.01.012.

[150] R. Schurr et al. «Tractography optimization using quantitative T1 mapping in the human optic radiation». En: *NeuroImage* 181 (nov. de 2018), págs. 645-658. DOI: 10.1016/j.neuroimage.2018.06.060.

[151] S. D. Serai. «Basics of magnetic resonance imaging and quantitative parameters T1, T2, T2*, T1rho and diffusion-weighted imaging». En: *Pediatric Radiology* 52.2 (abr. de 2021), págs. 217-227. DOI: 10.1007/s00247-021-05042-7.

[152] H. C. Shao et al. «Real-time liver tumor localization via combined surface imaging and a single x-ray projection». En: *Physics in Medicine & Biology* 68.6 (mar. de 2023), pág. 065002. DOI: 10.1088/1361-6560/acb889.

[153] Z. Shen et al. *Efficient Attention: Attention with Linear Complexities*. 2020. arXiv: 1812.01243 [cs.CV].

[154] R. Shimofusa et al. «Diffusion-Weighted Imaging of Prostate Cancer». En: *Journal of Computer Assisted Tomography* 29.2 (mar. de 2005), págs. 149-153. DOI: 10.1097/01.rct.0000156396.13522.f2.

[155] H. Singh. *Statistics for Machine Learning : Implement Statistical methods used in Machine Learning using Python*. 1.ª ed. bpb online, ene. de 2021.

[156] D. Soydaner. «Attention mechanism in neural networks: where it comes and where it goes». En: *Neural Computing and Applications* 34.16 (mayo de 2022), págs. 13371 -13385. DOI: 10.1007/s00521-022-07366-3.

[157] P. Sprawls. *Magnetic Resonance Imaging. Principles, Methods, and Techniques*. Medical Physics Publishing Corporation, pág. 173. ISBN: 9780944838976.

[158] E. O. Stejskal y J. E. Tanner. «Spin Diffusion Measurements: Spin Echoes in the Presence of a Time-Dependent Field Gradient». En: *The Journal of Chemical Physics* 42.1 (ene. de 1965), págs. 288-292. DOI: 10.1063/1.1695690.

[159] S. Suganyadevi, V. Seethalakshmi y K. Balasamy. «A review on deep learning in medical image analysis». En: *International Journal of Multimedia Information Retrieval* 11.1 (sep. de 2021), págs. 19-38. DOI: 10.1007/s13735-021-00218-1.

[160] E. V. Sullivan y A. Pfefferbaum. «Diffusion tensor imaging and aging». En: *Neuroscience & Biobehavioral Reviews* 30.6 (ene. de 2006), págs. 749-761. DOI: 10.1016/j.neubiorev.2006.06.002.

[161] H. Suzuki et al. «Metabolic Tumour Volume as a Predictor of Survival for Sinonasal Tract Squamous Cell Carcinoma». En: *Diagnostics* 12.1 (ene. de 2022), pág. 146. DOI: 10.3390/diagnostics12010146.

[162] J. Tang, C. Deng y G. B. Huang. «Extreme Learning Machine for Multilayer Perceptron». En: *IEEE Transactions on Neural Networks and Learning Systems* 27.4 (abr. de 2016), págs. 809-821. DOI: 10.1109/tnnls.2015.2424995.

[163] C. M. W. Tax y M. Bastiani y col. «What's new and what's next in diffusion MRI preprocessing». En: *NeuroImage* 249.118830 (2022).

[164] J-D. Tournier. «Diffusion MRI in the brain – Theory and concepts». En: *Progress in Nuclear Magnetic Resonance Spectroscopy* 112-113 (2019), págs. 1-16.

[165] J. D. Tournier, S. Mori y A. Leemans. «Diffusion tensor imaging and beyond». En: *Magnetic Resonance in Medicine* 65.6 (abr. de 2011), págs. 1532-1556. DOI: 10.1002/mrm.22924.

[166] J. D. Tournier et al. «MRtrix3: A fast, flexible and open software framework for medical image processing and visualisation». En: *NeuroImage* 202 (nov. de 2019), pág. 116137. DOI: 10.1016/j.neuroimage.2019.116137.

[167] N. Tsiknakis et al. «Deep learning for diabetic retinopathy detection and classification based on fundus images: A review». En: *Computers in Biology and Medicine* 135 (ago. de 2021), pág. 104599. DOI: 10.1016/j.compbiomed.2021.104599.

[168] L. Tunstall, L. von Werra y T. Wolf. *Natural Language Processing with Transformers Building Language Applications with Hugging Face. Building Language Applications with Hugging Face*. O'Reilly Media, Incorporated, 2022. ISBN: 9781098103248.

[169] A. Vaswani et al. «Attention is All You Need». En: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, 6000–6010. ISBN: 9781510860964.

[170] J. Veraart, E. Fieremans y D. S. Novikov. «Diffusion MRI noise mapping using random matrix theory». En: *Magnetic Resonance in Medicine* 76.5 (nov. de 2015), págs. 1582-1593. DOI: 10.1002/mrm.26059.

[171] J. Veraart et al. «Denoising of diffusion MRI using random matrix theory». En: *NeuroImage* 142 (nov. de 2016), págs. 394-406. DOI: 10.1016/j.neuroimage.2016.08.016.

[172] A. Vilanova et al. «An Introduction to Visualization of Diffusion Tensor Imaging and Its Applications». En: *Mathematics and Visualization*. Springer Berlin Heidelberg, 2006, págs. 121-153. DOI: 10.1007/3-540-31272-2_7.

[173] M. Wahl et al. «Microstructural correlations of white matter tracts in the human brain». En: *NeuroImage* 51.2 (jun. de 2010), págs. 531-541. DOI: 10.1016/j.neuroimage.2010.02.072.

[174] Q. Wang et al. «ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks». En: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun. de 2020. DOI: 10.1109/cvpr42600.2020.01155.

[175] W. Wang et al. *CrossFormer++: A Versatile Vision Transformer Hinging on Cross-scale Attention*. 2023. arXiv: 2303.06908 [cs.CV].

[176] Y. Wang et al. «Estimating Brain Connectivity With Varying-Length Time Lags Using a Recurrent Neural Network». En: *IEEE Transactions on Biomedical Engineering* 65.9 (sep. de 2018), págs. 1953-1963. DOI: 10.1109/tbme.2018.2842769.

[177] Z. Wang et al. «Image Quality Assessment: From Error Visibility to Structural Similarity». En: *IEEE Transactions on Image Processing* 13.4 (abr. de 2004), págs. 600-612. DOI: 10.1109/tip.2003.819861.

[178] M. Waqas et al. «Effectiveness of improved Fourier-Fick laws in a stratified non-Newtonian fluid with variable fluid characteristics». En: *International Journal of Numerical Methods for Heat & Fluid Flow* 29.6 (jun. de 2019), págs. 2128-2145. DOI: 10.1108/hff-12-2018-0716.

[179] J. Wasserthal et al. «Combined tract segmentation and orientation mapping for bundle-specific tractography». En: *Medical Image Analysis* 58.101559 (sep. de 2019).

[180] N. Weiskopf et al. «Advances in MRI-based computational neuroanatomy». En: *Current Opinion in Neurology* 28.4 (ago. de 2015), págs. 313-322. DOI: 10.1097/wco.0000000000000222.

[181] R. Woodhams et al. «Diffusion-weighted Imaging of the Breast: Principles and Clinical Applications». En: *RadioGraphics* 31.4 (jul. de 2011), págs. 1059-1084. DOI: 10.1148/rg.314105160.

[182] M. W. Woolrich et al. «Bayesian analysis of neuroimaging data in FSL». En: *NeuroImage* 45.1 (mar. de 2009), S173-S186. DOI: 10.1016/j.neuroimage.2008.10.055.

[183] Z. Xia et al. «Vision Transformer With Deformable Attention». En: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Jun. de 2022, págs. 4794-4803.

[184] E. Xie et al. *SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers*. 2021. DOI: 10.48550/ARXIV.2105.15203.

[185] S. Xu et al. «Jointly Attentive Spatial-Temporal Pooling Networks for Video-based Person Re-Identification». En: (ago. de 2017).

[186] T. Xue et al. «Superficial white matter analysis: An efficient point-cloud-based deep learning framework with supervised contrastive learning for consistent tractography parcellation across populations and dMRI acquisitions». En: *Medical Image Analysis* 85 (abr. de 2023), pág. 102759. DOI: 10.1016/j.media.2023.102759.

[187] T. Xue et al. «Supwma: Consistent and Efficient Tractography Parcellation of Superficial White Matter with Deep Learning». En: *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*. IEEE, mar. de 2022. DOI: 10.1109/isbi52829.2022.9761541.

[188] S. Xun et al. «Generative adversarial networks in medical image segmentation: A review». En: *Computers in Biology and Medicine* 140 (ene. de 2022), pág. 105063. DOI: 10.1016/j.compbiomed.2021.105063.

[189] B. Yang et al. «Classification of Medical Image Notes for Image Labeling by Using MinBERT». En: *Tsinghua Science and Technology* 28.4 (ago. de 2023), págs. 613-627. DOI: 10.26599/tst.2022.9010012.

[190] T. E. Yankeelov et al. «Integration of quantitative DCE-MRI and ADC mapping to monitor treatment response in human breast cancer: initial results». En: *Magnetic Resonance Imaging* 25.1 (ene. de 2007), págs. 1-13. DOI: 10.1016/j.mri.2006.09.006.

[191] F. C. Yeh et al. «Deterministic Diffusion Fiber Tracking Improved by Quantitative Anisotropy». En: *PLoS ONE* 8.11 (nov. de 2013). Ed. por Wang Zhan, e80713. DOI: 10.1371/journal.pone.0080713.

[192] X. X. Yin et al. «U-Net-Based Medical Image Segmentation». En: *Journal of Healthcare Engineering* 2022 (abr. de 2022). Ed. por Hangjun Che, págs. 1-16. DOI: 10.1155/2022/4189781.

[193] P. N. E. Young et al. «Imaging biomarkers in neurodegeneration: current and future practices». En: *Alzheimer's Research & Therapy* 12.1 (abr. de 2020). DOI: 10.1186/s13195-020-00612-7.

[194] T. Yousaf, G. Dervenoulas y M. Politis. «Advances in MRI Methodology». En: *International Review of Neurobiology*. Elsevier, 2018, págs. 31-76. DOI: 10.1016/bs.irn.2018.08.008.

[195] F. Yuan, Z. Zhang y Z. Fang. «An effective CNN and Transformer complementary network for medical image segmentation». En: *Pattern Recognition* 136 (abr. de 2023), pág. 109228. DOI: 10.1016/j.patcog.2022.109228.

[196] Y. Yuan et al. *OCNet: Object Context Network for Scene Parsing*. 2018. DOI: 10.48550/ARXIV.1809.00916.

[197] F. Zenke y T. P. Vogels. «The Remarkable Robustness of Surrogate Gradient Learning for Instilling Complex Function in Spiking Neural Networks». En: *Neural Computation* 33.4 (2021), págs. 899-925. DOI: 10.1162/neco_a_01367.

[198] H. Zhang et al. «Context Encoding for Semantic Segmentation». En: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, jun. de 2018. DOI: 10.1109/cvpr.2018.00747.

[199] R. Zhang et al. «SCAN: Self-and-Collaborative Attention Network for Video Person Re-Identification». En: *IEEE Transactions on Image Processing* 28.10 (oct. de 2019), págs. 4870-4882. DOI: 10.1109/tip.2019.2911488.

[200] Y. Zhang y A. J. Furst. «Brainstem Diffusion Tensor Tractography and Clinical Applications in Pain». En: *Frontiers in Pain Research* 3 (mar. de 2022). DOI: 10.3389/fpain.2022.840328.

[201] Y. Zhang et al. «Multifunctional ferromagnetic fiber robots for navigation, sensing, and treatment in minimally invasive surgery». En: (ene. de 2023). DOI: 10.1101/2023.01.27.525973.

[202] H. Zhao et al. «Deep learning based diagnosis of Parkinson's Disease using diffusion magnetic resonance imaging». En: *Brain Imaging and Behavior* 16.4 (mar. de 2022), págs. 1749-1760. DOI: 10.1007/s11682-022-00631-y.

[203] T. Zheng et al. «A microstructure estimation Transformer inspired by sparse representation for diffusion MRI». En: *Medical Image Analysis* 86 (mayo de 2023), pág. 102788. DOI: 10.1016/j.media.2023.102788.

[204] D. X. Zhou. «Theory of deep convolutional neural networks: Downsampling». En: *Neural Networks* 124 (abr. de 2020), págs. 319-327. DOI: 10.1016/j.neunet.2020.01.018.

[205] T. Zhou et al. «Volumetric memory network for interactive medical image segmentation». En: *Medical Image Analysis* 83 (ene. de 2023), pág. 102599. DOI: 10.1016/j.media.2022.102599.