

Nota de Aceptación

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



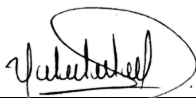
Dr. HERNÁN CAMILO ROCHA
Decano de la Facultad de Ingeniería



Dr. GERARDO MAURICIO SARRIA
Director Carrera Ingeniería Sistemas y Computación.



Dr. HERNÁN DARÍO VARGAS CARDONA
Director(a) Trabajo



ING. VALENTINA CORCHUELO GUZMAN
Jurado 1



Dr. GERARDO MAURICIO SARRIA
Jurado 2



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

Fecha: 20 Julio 2023

Autores: Juan Manuel Cuellar Borrero y Edixon Alirio Salas Medina

Nombre del Proyecto de Grado: Mejoramiento de la resolución espacial en imágenes dMRI mediante la aplicación de arquitecturas de aprendizaje profundo

Director: Dr. Hernan Darío Vargas Cardona

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

Firma de Director(a) del Proyecto de Grado

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

Mejoramiento de la resolución espacial en imágenes dMRI mediante la aplicación de arquitecturas de aprendizaje profundo

Juan Manuel Cuellar Borrero
Edixon Alirio Salas Medina

Director: Dr. Hernán Dario Vargas Cardona

25 de julio de 2023



Santiago de Cali, 25 de julio de 2023.

Señores

Pontificia Universidad Javeriana Cali.

Dr. Gerardo Mauricio Sarria

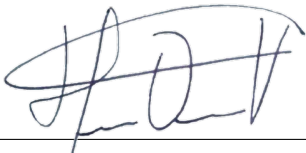
Director Carrera de Ingeniería de Sistemas y Computación.

Cali.

Cordial Saludo.

Por medio de la presente me permito informarles que los estudiantes de Ingeniería de Sistemas y Computación Juan Manuel Cuellar Borrero (cod: 8939771) y Edixon Alirio Salas Medina(cod: 8934749) trabajan bajo mi dirección en el proyecto de grado titulado “Mejoramiento de la resolución espacial en imágenes dMRI mediante la aplicación de arquitecturas de aprendizaje profundo”.

Atentamente,



Dr. Hernán Dario Vargas Cardona

Santiago de Cali, 25 de julio de 2023.

Señores

Pontificia Universidad Javeriana Cali.

Dr. Gerardo Mauricio Sarria

Director Carrera de Ingeniería de Sistemas y Computación.

Cali.

Cordial Saludo.

Nos permitimos presentar a su consideración el proyecto de grado titulado “Mejoramiento de la resolución espacial en imágenes dMRI mediante la aplicación de arquitecturas de aprendizaje profundo” con el fin de cumplir con los requisitos exigidos por la Universidad para llevar a cabo el proyecto de grado y posteriormente optar al título de Ingeniero de Sistemas y Computación.

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

Atentamente,



Juan Manuel Cuellar Borrero
Código: 8939771



Edixon Alirio Salas Medina
Código: 8934749

Resumen

Conforme avanza el tiempo más son los beneficios que trae la tecnología para diversidad de actividades humanas: entretenimiento, turismo, comodidad, y por supuesto la medicina. Dichos avances han logrado crear maneras de poder explorar el interior del cuerpo humano de manera que no sea necesario el uso de procedimientos quirúrgicos.

Las imágenes médicas son un conjunto de modalidades que son utilizadas con fines diagnósticos, por lo que son de suma importancia para la medicina, dentro de dichas modalidades se encuentra la resonancia magnética (MRI, por sus siglas en inglés) la cual utiliza radiación magnética para poder obtener información sobre los tejidos con una visualización en tercera dimensión (3D); dentro de los tipos de MRI, se encuentra la imagen de resonancia magnética por difusión (dMRI), en la cual se somete al paciente a campos magnéticos desde distintos gradientes direccionales generando una excitación en las moléculas de agua que se encuentran dentro de los tejidos del cuerpo, las cuales empiezan a moverse y reorientarse con respecto al campo magnético aplicado, lo que genera señales detectables.

El problema con este tipo de imágenes es su resolución espacial y la relación señal-ruido (SNR) debido principalmente a las limitaciones de hardware en los escáneres usados, pues los protocolos clínicos actuales permiten adquisiciones rápidas, lo que conlleva a una resolución espacial baja del estudio y muchas veces se da que no son lo suficientemente buenos para el diagnóstico clínico.

El acercamiento propuesto en este documento es realizar **Super Resolución** mediante el uso de *Deep Learning* con 3 tipos de redes neuronales: Super-Resolution Convolutional Neural Network (SRCNN), Enhanced Deep Super-Resolution Network (EDSR) y Very Deep Super Resolution Network (VDSR) las cuales son herramientas muy usadas en los sistemas de visión por computadora. Así pues, se entrena la IA, para tener un modelo predictivo que con base en una imagen de baja resolución pueda generar una nueva con mayor resolución espacial.

Palabras Clave: Diffusion magnetic resonance imaging - dMRI, Diffusion tensor imaging - DTI, Resolución espacial, Interpolación, Convolutional Neural Network - CNN, Enhanced Deep Super-Resolution Network - EDSR, Very Deep Super-Resolution Network - VDSR, Regresión, Deep Learning.

Índice general

1. Descripción del Problema	13
1.1. Planteamiento del Problema	13
1.1.1. Formulación	15
1.1.2. Sistematización	15
1.2. Objetivos	16
1.2.1. Objetivo General	16
1.2.2. Objetivos Específicos	16
1.3. Justificación	17
1.3.1. Relevancia	17
1.3.2. Viabilidad	17
1.3.3. Impacto	18
1.4. Delimitaciones y Alcances	19
1.4.1. Entregables	19
2. Desarrollo del Proyecto	21
2.1. Marco de Referencia	21
2.1.1. Áreas Temáticas	21
2.1.2. Marco Teórico	21
2.1.3. Trabajos Relacionados	28
2.2. Metodología	36
2.2.1. Tipo de Estudio	36
2.2.2. Actividades	36
3. Desarrollo Metodológico	39
3.1. Entorno de ejecución	40
3.1.1. Uso de la GPU	40
3.2. Exploración del Conjunto de Datos	40
3.3. Preprocesamiento	41
3.3.1. Utilización de Máscara	43
3.4. Modelo SRCNN	44
3.4.1. Implementación del Modelo	44
3.5. Modelo VDSR	45
3.5.1. Implementación del Modelo	45
3.6. Modelo EDSR	45
3.6.1. Implementación del Modelo	45
3.6.2. Aumento del Conjunto de Datos	46

4. Resultados	49
4.1. Modelo SRCNN	50
4.1.1. Tiempos de Ejecución	51
4.1.2. Métricas de Imágenes	52
4.1.3. Métricas Tensoriales	54
4.2. Modelo VDSR	56
4.2.1. Tiempos de Ejecución	58
4.2.2. Métricas de Imágenes	58
4.2.3. Métricas Tensoriales	61
4.3. Modelo EDSR	63
4.3.1. Tiempos de Ejecución	65
4.3.2. Métricas de Imágenes	65
4.3.3. Métricas Tensoriales	68
4.4. Comparación de los Modelos	70
5. Conclusiones y Trabajos Futuros	75
Bibliografía	77

A mi padre y a mi abuela ... (Juan Manuel Cuellar)

Introducción

En las últimas décadas, la tecnología ha estado avanzando progresivamente, lo que ha logrado un mejor desempeño en la obtención de diagnósticos médicos. Por ejemplo, el poder computacional y el almacenamiento de grandes volúmenes de datos ha dado paso al análisis y manipulación de los mismos y ha impulsado el uso de técnicas como *deep learning* para realizar predicción o clasificación de datos. En general, la tecnología ha impactado positivamente a la medicina y más específicamente al campo de las imágenes médicas, las cuales permiten estudiar el cuerpo humano sin necesidad del uso de métodos invasivos. Algunos métodos como la resonancia magnética, tomografía computarizada, ultrasonidos y rayos X, permiten obtener imágenes del cuerpo humano y ofrecen información anatómica relevante sobre la condición médica actual del paciente, lo que ayuda a realizar diagnósticos para el mismo [1].

La imagen de resonancia magnética por difusión(dMRI) es una forma de imagen por resonancia magnética la cual se enfoca en la medición del movimiento de las moléculas de agua en el tejido al someterse a una radiación magnética, la cual es aplicada desde diferentes gradientes de orientación[2, 3, 4, 5], es muy conocida por su excelente contraste en materia blanca(WM)[2] y su utilidad en la caracterización de tumores y la isquemia cerebral, al igual que en estudios de desordenes neurológicos, procesos de envejecimiento cerebral e infartos cerebrales [6] por lo que es utilizado para estudiar la microestructura y la conectividad estructural del tejido cerebral [7, 8, 9]. Por otro lado, innumerables casos de cáncer se presentan anualmente en el mundo y su control está centrado en la prevención y la detección temprana, por lo que el uso de este tipo de imágenes es fundamental para un correcto tratamiento[10]. Así mismo, este tipo de imágenes tiene muchas más aplicaciones en diferentes situaciones, por ejemplo, la identificación temprana de accidente cerebrovascular isquémico, la diferenciación entre el accidente cerebrovascular agudo del accidente cerebrovascular crónico, clasificación de gliomas y meningiomas difusos, entre otras [11, 12].

Sin embargo, la imagen de resonancia magnética por difusión(dMRI) tiene un problema en cuanto a resolución espacial y la relación señal-ruido (SNR)[2, 13, 4, 5] debido principalmente a las limitaciones de hardware en los escáneres usados, pues, los protocolos clínicos actuales permiten adquisiciones rápidas, lo que conlleva a una resolución espacial baja del estudio[14] y muchas veces se da que no son lo suficientemente buenos para el diagnóstico clínico [9, 15]. Por lo anterior, es necesario lograr una mejor resolución espacial, procesando las imágenes para obtener mejores resultados. La solución propuesta en este documento se basa principalmente en el desarrollo de métodos haciendo uso de *deep learning* para realizar una mejora en la resolución espacial de dichas imágenes.

Descripción del Problema

1.1 Planteamiento del Problema

En la actualidad las imágenes médicas representan una poderosa herramienta utilizada por numerosos hospitales, pues gracias a estas se permite obtener información anatómica relevante sobre la condición médica actual del paciente, dado el hecho de que los datos obtenidos a partir de las mismas tienen el potencial de revelar condiciones no identificables a primera vista. Entre estas imágenes se encuentran las tomografías, radiografías o ultrasonografías [1]

Las imágenes de resonancia magnética por difusión o dMRI, por sus siglas en inglés, tienen el potencial de proveer información supremamente valiosa y relevante para el conocimiento médico, como lo es el proceso de inferir propiedades de tejido de materia blanca (WM)[6, 2] pues además de ser un método no invasivo *in vivo* para medir propiedades del tejido, permite también ver pequeños detalles estructurales del mismo [4]. Este método ha sido usado durante años para detectar patologías intracraneales, principalmente para el diagnóstico precoz del accidente cerebrovascular isquémico, así como para estudios de la salud mental y procesos cognitivos humanos[2, 7, 8, 9]. Además, la interpretación es muy rápida y sencilla a la hora de demostrar alteraciones isquémicas del encéfalo y su utilidad se ha visto confirmada para patologías del sistema nervioso central (SNC), como traumas, epilepsia, demencia y neurotoxicidad. Por otro lado, se utilizan en la realización de mapeos de conexiones en el cerebro, así como modelos de conductividad cerebrales [9]. No obstante, este tipo de imágenes tienen ciertas dificultades técnicas limitantes, como lo es el compromiso entre una resolución espacial muy baja y SNR [9, 13, 14, 4], esto sucede debido a limitaciones en cuanto a los equipos utilizados para la adquisición de las mismas (Hardware) y a protocolos médicos, los cuales definen unas cantidades necesarias y máximas de exposición a radiación (**mSv**) para cada tipo de imagen médica[9, 15, 14]. Además, los tiempos de exploración son limitados, por ejemplo, en el Human Connectome Project (HCP), el tiempo de exploración de dMRI es de alrededor a 60 minutos para un adulto y en el Baby Connectome Project (BCP), el tiempo de adquisición es de máximo 15 minutos[16, 17].

Debido a la ya mencionada baja resolución espacial de las imágenes obtenidas por dMRI, estas pierden valor a la hora de realizar diagnósticos médicos, pues disminuye la precisión de la información obtenida de las mismas [14, 2], lo cual necesariamente limita la versatilidad del personal médico, ya que, se tiene acceso a un menor número de herramientas para poder diagnosticar a los pacientes. Este problema ya ha sido abordado con anterioridad, utilizando métodos de interpolación,

acercamientos por medio de algoritmos de super resolución (SR), incluso modelos probabilísticos [9, 4, 5], sin embargo, los métodos de interpolación generan un error considerable debido a su poca profundidad en términos técnicos, y los modelos probabilísticos usados requieren de conocimientos sumamente avanzados.

De acuerdo a lo anterior, se puede ver que si bien la información provista por *imágenes de resonancia magnética* (MRI y dMRI) es sumamente importante y valiosa, existe un problema constante, su baja resolución espacial, si bien ya existen metodologías capaces de realizar un mejoramiento de la resolución espacial en imágenes dMRI, estos o tienen un margen de error lo suficientemente alto como para realizar un mal diagnóstico médico con respecto a la imagen obtenida, o son sumamente complejos. El uso de redes de aprendizaje profundo ha arrojado buenos resultados para el mejoramiento de resolución en imágenes médicas en distintos campos y problemas realacionados con las imágenes médicas. Es a raíz de este que nace la idea de realiza un estudio de diferentes métodos de *Super Resolución* para evaluar es desempeño de estos aplicandolo al campo de imágenes dMRI.

1.1.1. Formulación

¿Cuáles algoritmos de aprendizaje profundo tienen un mejor desempeño al aplicarse super resolución a estudios dMRI?

1.1.2. Sistematización

- ¿Cuáles modelos de aprendizaje profundo se pueden implementar para generar estudios dMRI con resolución espacial mejorada?
- ¿Cómo ajustar la arquitectura de las redes profundas para adaptarlas a la problemática de estudios dMRI?
- ¿Cuáles métricas numéricas y resultados clínicos se pueden usar para validar los algoritmos implementados?

1.2 Objetivos

1.2.1. Objetivo General

Evaluar el desempeño de metodologías que permita el mejoramiento de la resolución espacial de estudios dMRI basada en técnicas de deep learning y super resolución para apoyar procedimientos clínicos.

1.2.2. Objetivos Específicos

- Implementar 3 modelos de aprendizaje profundo para generar estudios dMRI de alta resolución.
- Adaptar las arquitecturas de las redes de aprendizaje profundo para que puedan recibir como entrada estudios dMRI.
- Evaluar el rendimiento de los métodos propuestos empleando métricas de error en imágenes y validación morfológica.

1.3 Justificación

1.3.1. Relevancia

La importancia del proyecto radica en el hecho de que la información obtenida a raíz de las imágenes dMRI es de suma importancia en ámbito diagnóstico, pues permite inferir propiedades de tejido de materia blanca [2, 9], así como estudiar desordenes neurológicos [6], no obstante, una resolución espacial baja hace que dicha información se vuelva imprecisa, lo que limita al personal médico, pues se genera un nivel de incertidumbre que conlleva a una pérdida de información relevante sobre el estado del paciente [9], haciendo que el médico llegue a hacer suposiciones [4]. Mediante el mejoramiento de la resolución espacial, se está ayudando al médico a que pueda distinguir mejor los detalles en la imagen, facilitando así el diagnóstico clínico, así como otras actividades relacionadas como lo puede ser la segmentación [14, 9].

Adicionalmente, las imágenes derivadas a raíz de la información obtenida de dMRI, como lo es la tractografía, proporciona ventajas como lo es una visualización tridimensional del seguimiento de la trayectoria del nervio trigeminal [18], así como el mapeo de fascículos de materia blanca humana [2].

1.3.2. Viabilidad

Trabajos de investigación previos [14, 9, 5, 4] muestran resultados satisfactorios en cuanto al mejoramiento de la resolución espacial, probando así que efectivamente se puede lograr un mejoramiento visible y relevante. Además, la implementación de metodologías de aprendizaje profundo a problemáticas similares, como lo puede ser la tractografía [13, 19] y la tomografía de coherencia óptica [20] ha demostrado la precisión y la fiabilidad de la aplicación de técnicas como la puede ser la Super-Resolución (SR) o interpolación.

Un acercamiento planteado desde la ingeniería y la computación es por medio de algoritmos de *Deep Learning*, pues en los últimos años, numerosos estudios han demostrado la aplicabilidad de dichas técnicas a diversos tipos de imágenes médicas, como lo puede ser la tomografía de coherencia óptica trabajada por Yamashita en [20], microscopía trabajada por Kwon en [21] o los diversos acercamientos hechos a tractografías expresados en [13, 19]. De igual manera soluciones planteadas desde el *Deep learning* a otras problemáticas relacionadas con imágenes médicas, como lo puede ser la segmentación de tejidos, son estudiadas en [22, 2, 6]. Todos estos estudios han reportado mejoras significantes y relevantes en cuanto a los resultados esperados [5], lo que muestra que el acercamiento desde los algoritmos de aprendizaje profundo es viable y eficaz.

Adicionalmente el estudio realizado por el Doctor Hernan Darío Vargas Cardona en su tesis doctoral "*Probabilistic modeling of tensorial data for enhancing spatial resolution in magnetic resonance imaging*" [9], donde se desarrolla un *framework* probabilístico basado en procesos estocásticos, en el que se mejora la resolución espacial de distintas modalidades de imágenes de

resonancia magnética, entre ellas las dMRI. Tomando en cuenta el hecho de que él es el director del trabajo de grado, se cuenta con una persona no solo altamente capacitada, sino también con un conocimiento profundo del tema que será un guía de bastante utilidad. Adicionalmente el contar con 2 estudiantes pertenecientes al programa de ingeniería de sistemas y ciencias de la computación, con conocimiento en áreas como lo es el diseño de algoritmos, así como computación científica e inteligencia artificial y aprendizaje de máquina, hace que el poder llevar a cabo dicho proyecto sea viable.

1.3.3. Impacto

El proyecto tiene un fuerte impacto en el ámbito social, pues tiene el potencial de proveer información vital para generar diagnósticos más certeros, adicionalmente, debido a que las imágenes dMRI son incomparables con otro tipo de imágenes médicas en cuanto a inferir información anatómica sobre la materia blanca (WM) de manera no invasiva [2], por lo que el contribuir a obtener mayores detalles permite generar un mayor conocimiento médico, así como procedimientos médicos tales como la segmentación de tejidos, y modelos de conductividad cerebrales.

Adicionalmente, el proyecto reduce la inversión necesaria para poder obtener una mejor resolución espacial, puesto que evita la adquisición de sensores más potentes, pues todo el procesamiento realizado es a nivel de software, después de que se haya realizado la adquisición de los datos.

1.4 Delimitaciones y Alcances

- Se implementarán tres modelos de aprendizaje profundo usando los métodos de *Super-Resolution Convolutional Neural Network*(SRCNN), *Enhanced Deep Super-Resolution Network*(EDSR) y *Very Deep Super Resolution Network* (VDSR).
- Se hará uso del conjunto de datos obtenidos mediante un estudio que se ha realizado el Dr. Hernan Dario vargas en su zona creneal para hacer el entrenamiento de los algoritmos.
- El tiempo establecido para la realización del proyecto es de 8 meses, donde se implementarán las metodologías, se obtendrán resultados, se analizarán y se hará un reporte.
- Las imágenes a utilizar para el desarrollo del proyecto pertenecen al área del cerebro.

1.4.1. Entregables

- Documento de trabajo de grado.
- Documento con descripción de las metodologías de *deep learning* aplicadas durante el proyecto, junto con evidencias de los resultados obtenidos con sus respectivos análisis.
- Notebooks con las metodologías implementadas.
- Artículo publicable que evidencie los resultados obtenidos y el proceso realizado.

Desarrollo del Proyecto

2.1 Marco de Referencia

Para el desarrollo de proyecto propuesto se debe tener en cuenta aspectos tanto de Ingeniería como de Ciencias de la computación

2.1.1. Áreas Temáticas

A continuación se presentan las categorías relacionadas con el tema en cuestión de acuerdo a la clasificación propuesta por ACM:

- *Computing methodologies* → *Machine learning* → *Machine learning approaches* → *Neural networks*
- *Applied computing* → *Life and medical sciences* → *Computational biology* → *Imaging*
- *Computing methodologies* → *Computer graphics* → *Image manipulation* → *Image processing*

2.1.2. Marco Teórico

Diffusion magnetic resonance imaging (dMRI): Es una técnica de imagen médica que permite visualizar estructuras internas del cuerpo a través del uso de radiación magnética, su importancia radica en el hecho de que es un método no invasivo in vivo para obtener información acerca de tejidos [1]. dMRI permite detectar y describir el movimiento browniano (difusión) de las partículas de agua a través del uso de tensores de difusión (DT) [9, 2]. La difusión puede entenderse a través de la ecuación $\langle r^2 \rangle \geq 6Dt$ donde $\langle r^2 \rangle$ es la media al cuadrado del desplazamiento de las moléculas, t es el tiempo de difusión y D es la constante de difusión, a raíz de esto se puede estimar el coeficiente de difusión aparente (ADC) el cual refleja la capacidad de difusión específica medida [23].

En el aspecto técnico del dMRI, todos los átomos contienen partículas subatómicas con cargas eléctricas, como lo son los electrones y protones, siendo estos negativos y positivos respectivamente. Estas moléculas al ser sometidas a un fuerte campo magnético son capaces de absorber energía (resonancia), para luego devolverla en forma de ondas de radio frecuencia (relajación). Los átomos de hidrógeno al ser los más abundantes en el tejido orgánico son los que se toman como referencia en las imágenes MRI en general. El hidrógeno posee un electrón y un protón, en el que el electrón

gira alrededor del núcleo del átomo y el protón gira alrededor de su propio eje, a este se le conoce como espín. Al ser sometido a un campo magnético, este se orienta con respecto a dicho campo (magnetización longitudinal), al seguir enviando pulsos electromagnéticos que carguen de energía al átomo se disminuye la magnetización longitudinal y se induce una magnetización transversal, al cesar los pulsos se entra en una etapa de relajación donde se recupera la longitudinal (recuperación T1) y se disminuye la transversal (relajación T2). La energía liberada es captada por una antena, para luego ser interpretada y obtener una imagen [23].

Imágenes de tensor de difusión (DTI): Es una de las modalidades de imagen de resonancia magnética por difusión en el que se utilizan tensores para modelar y la difusión del agua en tejido orgánico, estos tensores se describen matemáticamente a través de una matriz simétrica 3×3 definida positiva la cual está asociada a un voxel. Ahora bien, una grilla tridimensional de tensores de difusión interconectados entre sí se le conoce como campo de tensor de difusión, siendo estos altamente útiles en aplicativos médicos, como lo es la segmentación de tejidos cerebrales, o el diagnóstico de enfermedades neurodegenerativas [9].

El tensor ajusta la variación angular de los valores ADC a un elipsoide como se muestra en la figura 2.1 donde $D_{ij} = D_{ji}$, dicho elipsoide indica el hecho de que existe una mayor difusión media a lo largo del eje mayor del elipsoide [23]. Dadas las propiedades matemáticas del tensor, se puede obtener a raíz de este los valores propios del mismo, siendo estos $\lambda_1, \lambda_2, \lambda_3$ donde $\lambda_1 \geq \lambda_2 \geq \lambda_3$, es decir, λ_1 representa la dirección principal de difusión [24].

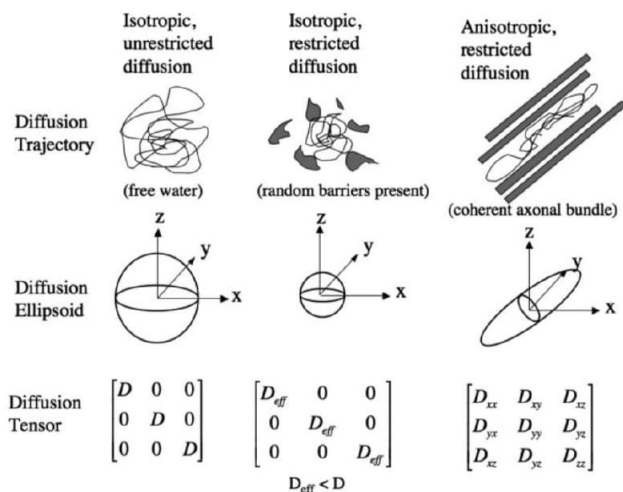


Figura 2.1: Representación gráfica de tensores de difusión. *Imagen tomada de:* [23]

Anisotropía fraccional: El movimiento difuso del agua se ve afectado en gran medida por los obstáculos los cuales tenga alrededor. Cuando las moléculas pueden moverse libremente en todas las direcciones, como lo puede ser en un vaso de agua, se le conoce como difusión isotrópica. En el

cerebro humano existen diversas estructuras y tejidos que impiden dicho movimiento libre, es decir, el coeficiente de difusión depende de la dirección en la que se encuentre, a esto se le conoce como difusión anisotrópica. Ahora bien, dados los 3 ejes principales del tensor de difusión, $\lambda_1, \lambda_2, \lambda_3$, se puede calcular la anisotropía fraccional, la cual mide el grado de direccionalidad intravoxel, mediante la siguiente formula:

$$FA = \frac{\sqrt{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_3 - \lambda_1)^2}}{\sqrt{2}\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}$$

En la imagen 2.2 se puede ver una imagen de anisotropía fraccional a la derecha, y a la izquierda la misma imagen solo que con un código de color dependiendo de la dirección principal del tensor: rojo para una orientación en el eje x , azul para el eje y y verde para el eje z [24, 23].

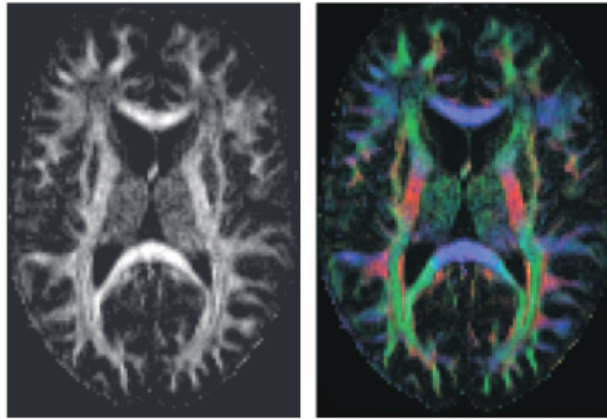


Figura 2.2: Imágenes de anisotropía fraccional. *Imagen tomada de:* [24]

Tractografía: Conociendo las orientaciones de los tractos gracias al modelo mencionado anteriormente, estas orientaciones se pueden unir para lograr una reconstrucción de conexiones cerebrales. Es decir, la tractografía permite visualizar de manera cualitativa la pérdida de continuidad en las fibras, por lo que se puede detectar alteraciones en la microestructura [24, 23].

Machine Learning: Es una disciplina científica del campo de la Inteligencia Artificial que crea sistemas que aprenden automáticamente a través de algoritmos, dando a los ordenadores la capacidad de identificar patrones en datos masivos para hacer predicciones y permitiéndoles realizar tareas específicas de forma autónoma, es decir, sin necesidad de ser programados[25].

Deep learning: Es un conjunto de algoritmos de aprendizaje automático(Machine learning) que modela abstracciones de alto nivel en datos usando arquitecturas computacionales que admiten transformaciones no lineales múltiples e iterativas de datos expresados en forma matricial o tenso-

rial los cuales han permitido una mejor interpretación de las imágenes médicas[26].

Convolutional Neural Network(CNN): Es un tipo de red neuronal artificial profunda que toma una imagen como entrada. Después, calcula y asigna características de los objetos. Este tipo de red es una variación de un perceptrón multicapa y su aplicación es realizada sobre matrices 2D, por lo que son útiles para analizar, adquirir y procesar imágenes. Esta red está compuesta por una capa de entrada, una capa de salida, una Red Neuronal multicapa y por capas convolucionales; cada una tiene una colección de filtros o kernels que sirven para capturar dependencias espaciales y temporales en las imágenes para hacerlas más sencillas de procesar sin perder las características más importantes[27].

Un kernel o filtro es una matriz $n \times m$ la cual se va desplazando por cada pixel de la imagen para hacer la suma de los productos de cada uno de los elementos de ambas matrices y crear una nueva matriz llamada mapa de características, la cual es una imagen más pequeña en altura y anchura pero con mayor profundidad. Dependiendo de los valores en el filtro inicialmente se podrían identificar sombras, bordes, cambios de contraste y superficies entre otras. A medida que se van aplicando los filtros, estos pueden ir generando mapas de características de detalles más grandes, es decir, inicialmente van a obtener líneas, sombras, diferencias de contrastes y después se van a empezar a detectar figuras geométricas para detectar formas más complejas de la imagen. Por ejemplo, si se tiene como entrada un rostro, inicialmente detectará elementos básicos para al final poder detectar un ojo, la nariz o las cejas[28, 29].

Cada capa convolucional cuenta con un conjunto de filtros y cada uno tiene un tamaño. La profundidad de la capa convolucional está dada por el número de filtros y entre más filtros tenga, más profundidad tendrá la imagen resultante. Por otro lado, existe un parámetro llamado stride o paso el cual indica cada cuantos pixeles recorre el filtro en la imagen y entre más grande este valor, más pequeña será la imagen resultante. Después de cada convolución se realiza un proceso de Pooling o agrupamiento el cual sirve para reducir el tamaño de la imagen debido a la cantidad de datos ya que las imágenes resultantes serán la entrada de la siguiente capa convolucional, lo que significa que se reducirán los datos, dejando los elementos más importantes, resolviendo el problema que tienen las redes neuronales ya que el uso de tantos datos es muy pesado computacionalmente. Para el proceso de Pooling hay 2 opciones, Max Pooling y Average Pooling los cuales se basan en seleccionar el elemento más grande del parche y el promedio del parche respectivamente, siendo el parche la submatriz de tamaño el cual es definido para cada caso. Además, la agrupación o pooling ayuda a reducir el número de conexiones y a no sobreajustar el modelo[28, 29].

El proceso se puede ver en la figura 2.3, en donde la capa de entrada le pasa los valores a la primera capa de convolución, para luego realizar pooling, después volver a realizar el proceso de convolución y así sucesivamente hasta terminar las convoluciones y el agrupamiento para ya realizar el proceso de clasificación o regresión con la RN multicapa y arrojar la salida[29].

CNN es muy útil para imágenes por la cantidad de datos y se basa en aprender los filtros, desde los más básicos hasta volverse más abstractos.

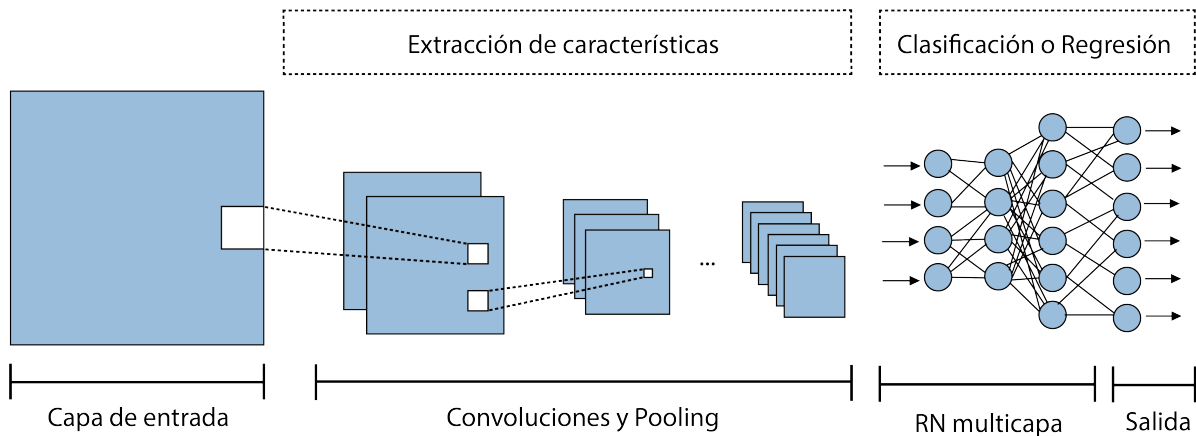


Figura 2.3: Arquitectura CNN. Imagen tomada de: [29]

Enhanced Deep Super Resolution Network (EDSR): Esta red pertenece a la clasificación de redes residuales, este tipo de redes han mostrado una mejora considerable en problemas de *SR*, gracias al hecho de que la imagen de alta resolución I_{HR} está altamente correlacionada con la imagen de baja resolución I_{LR} , el modelo puede aprender el mapa residual que transforma a I_{LR} en I_{HR} [30], adicionalmente estas redes atacan el problema de gradiente desvaneciente (*vanishing gradient problem*) característico de redes neuronales de muy alta profundidad [31, 32], este problema hace que al aumentar la profundidad de la red se obtengan peores resultados.

Esto se logra al utilizar la estructura conocida como *Residual Blocks* los cuales utilizan una estrategia de *Skip-connection*, la cual consiste en añadir la entrada del bloque a la salida del mismo. En el caso de la EDSR los bloques residuales utilizados nacen como una optimización a los bloques usados en la red *Super Resolution Residual Network* (SRResNet) [31]. Los bloques constan de una capa convolucional con una función de activación ReLu seguida de otra capa convolucional, utilizando el método de *skip connection* mencionado anteriormente, esto se puede ver en la figura 2.4 .

La decision de remover las capas de *batch normalization* surge a raiz de que dichas capas normalizan las características, reduciendo el rango de flexibilidad de la red, en el artículo de Lim Bee [31] se muestra de manera experimental que el remover dichas capas da un mejor rendimiento.

Very Deep Super-Resolution Network (VDSR): es un método basado en el aprendizaje profundo para la *SR* de una sola imagen (SISR) que utiliza una red neuronal convolucional muy profunda para mejorar la calidad de las imágenes de baja resolución inspirada en VGG-net utilizada para ImageNet classification [33].

VDSR fue propuesta por el equipo de investigación de Jiwon Kim y está entrenada de extremo

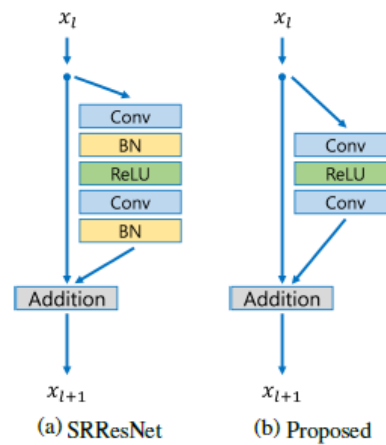


Figura 2.4: Estructura de bloques residuales *imagen tomada de:*
[31]

a extremo para minimizar el error cuadrático medio entre la imagen de alta resolución generada y la imagen real. La idea principal detrás de VDSR es aprender una función no lineal que mapea una imagen de baja resolución a su correspondiente versión de alta resolución.

La red neuronal se entrena utilizando un conjunto de datos de pares de imágenes LR-HR, donde LR representa la imagen de baja resolución y HR representa la imagen original. Asimismo, durante el entrenamiento la red aprende a estimar los detalles finos y las características estructurales que faltan en la imagen LR para producir una versión mejorada HR.[34]

La arquitectura VDSR consta de 20 capas convolucionales profundas, con un gran número de filtros en cada capa como se muestra en la figura 2.5[35].

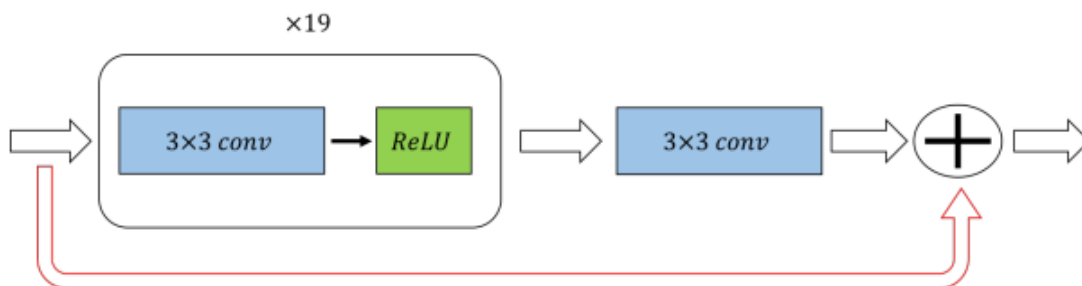


Figura 2.5: Arquitectura de VDSR. *Imagen tomada de:* [35]

En general, VDSR ha demostrado ser uno de los métodos más precisos y efectivos para mejorar la calidad visual y perceptual en imágenes con baja resolución para su fecha de divulgación, lo que la convierte en una herramienta valiosa para una variedad de aplicaciones, incluidas las imágenes médicas.

Resolución espacial: La resolución espacial es una medida del número de píxeles usados para representar la estructura registrada como una imagen digital[36].

Relación señal-ruido: Debido a fluctuaciones en la fuente de energía o en general en el ambiente, aparece ruido de fondo, lo que agrega a la imagen componentes ajenos al objetivo. Una relación señal ruido(S/N) elevada indica una imagen agradable y con buena calidad[36].

Super Resolución: La super resolución SR por sus siglas en inglés, hace referencia al proceso de obtener una imagen de alta resolución I_{HR} a partir de una imagen de baja resolución I_{LR} [30]. Esto puede ser representado por la siguiente fórmula:

$$I_{LR} = d(I_{HR}, \delta)$$

Donde d es una función de degradación responsable de convertir una imagen de alta resolución a una de baja resolución, y δ hace referencia a los parámetros de dicha función como lo pueden ser el factor de escalamiento, el ruido, entre otros [30]. En la práctica lo que se busca es poder encontrar la inversa de la función de degradación d de tal manera de que $g = d^{-1}$, por lo que se tendría una expresión:

$$g(I_{LR}, \theta) = I_E \approx I_{HR}$$

Donde I_E es la imagen estimada, g es la función de super resolución y θ son los parámetros de g . Es importante recalcar que g es una función no-inyectiva lo que implica que hay infinitas posibilidades para I_E donde $d(I_E, \delta) = I_{LR}$.

Cross Validation: Cross Validation o Validación Cruzada es una técnica usada para evitar el sobreajuste y encontrar los valores óptimos de los parámetros. Consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones, definiendo k el cual es el número de particiones del conjunto de entrenamiento y en cada repetición se usa una diferente partición para usarlo como conjunto de validación y el resto se usa como conjunto de entrenamiento[37]. El proceso se puede apreciar en la figura 2.6.

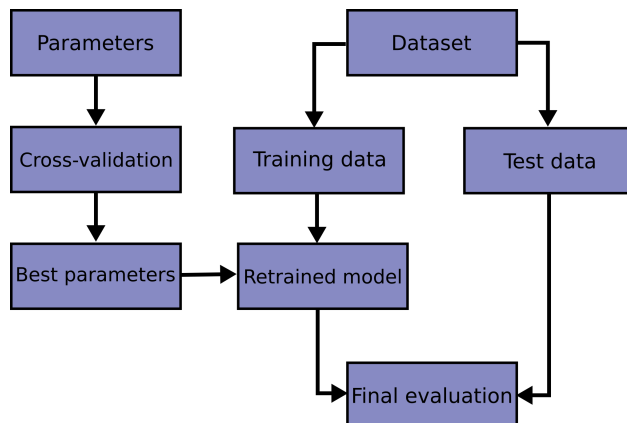


Figura 2.6: Diagrama de flujo de *Cross Validation*. Imagen tomada de:[37]

2.1.3. Trabajos Relacionados

En el artículo **XQ-SR: Joint $x - q$ space super-resolution with application to infant diffusion MRI(2019)** hablan acerca de la resolución espacial baja de las imágenes dMRI y más aún para infantes, ya que su cerebro es un poco más pequeño que el de un adulto y exige una mayor resolución espacial para capturar estructuras sutiles. También hablan de que existen soluciones que usan Super-Resolución, pero la mayoría de los métodos existentes se enfocan en el dominio espacial o el dominio del vector de onda de difusión. Lo que se propone en el artículo es lograr una mejora de la resolución más eficaz, proponiendo un marco para la Super-Resolución conjunta en los dominios espaciales y de vectores de ondas. Además, muestran que el método es capaz de recuperar datos dMRI de alta resolución con una calidad notablemente mejorada[17].

En el artículo **Cost-free resolution enhancement in Convolutional Neural Network for medical image segmentation (2020)** se habla sobre una manera en la que mediante la utilización de CNN se logra aumentar la resolución de imágenes médicas, el procedimiento que realiza es, dada una imagen de entrada en baja resolución, se realiza un desfase de diferentes *sub-voxels*, dichas imágenes se le son pasadas a la red neuronal convolucional, para realizar la segmentación, luego se combinan las imágenes ya procesadas obteniendo una mayor resolución, para una mejor descripción dada una imagen con N dimensiones: $(d_1 \times \dots \times d_N)$, si se quiere aumentar la resolución por un factor k se crean $k - 1$ imágenes nuevas derivadas de la imagen original donde cada imagen nueva tiene un desfase de $+\frac{1}{k}$ con respecto a la imagen anterior. Posteriormente se crea una imagen nueva mediante la intercalación a nivel de *sub-voxel* de las $k - 1$ previamente procesadas. El proceso mencionado anteriormente se ve gráficamente en la figura 2.7 [38].

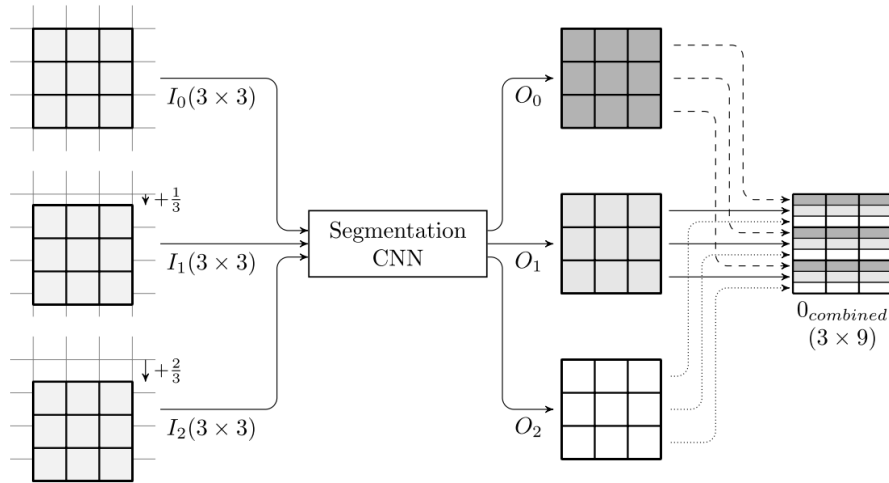


Figura 2.7: Proceso de mejoramiento de resolución. *Imagen tomada de:* [38]

En el artículo **Residual dense network for medical magnetic resonance images super-resolution (2021)** los autores proponen una red residual densa para super resolución a imágenes MRI de rodilla, la estructura de la red se puede apreciar en la figura 2.8, esta cuenta con la particularidad de tener bloques multi-residuales **MRB** los cuales a su vez contienen 3 bloques residuales **RB** donde cada uno tiene 2 capas convolucionales con funciones de activación ReLu, esta red utiliza *skip connections* lo que permite que la información del bloque residual anterior fluya al siguiente sin ningún obstáculo, esto porque entre más se profundiza en la red, las características extraídas por las capas convolucionales se vuelven cada vez más abstractas, por lo que la información de la estructura de la imagen se pierde, debido a esto, las características iniciales de la imagen pueden obtenerse a través de la implementación de aprendizaje residual, permitiendo un mejor flujo de la información de la imagen a través de la red. [39]

En el artículo **Super Resolution Techniques for Medical Image Processing (2015)** se habla acerca de distintas técnicas de Super-Resolución (SR) utilizadas en imágenes médicas, se definen 3 componentes básicos de SR los cuales son *motion compensation*, *interpolation* y *blur-noise removal*. Una de las técnicas es el mejoramiento de rango dinámico de imágenes MRI, en el que una imagen de entrada $I(x, y)$ se descompone en otras 2 imágenes, iluminación $L(x, y)$ y reflectancia $R(x, y)$, tal que $I(x, y) = L(x, y) \times R(x, y)$, luego se hace un *downsampling* de la imagen, a dicha nueva imagen se le hace un proceso en el que se estima el contraste de los píxeles adyacentes para posteriormente realizar interpolación como se muestra en la figura 2.9 [15].

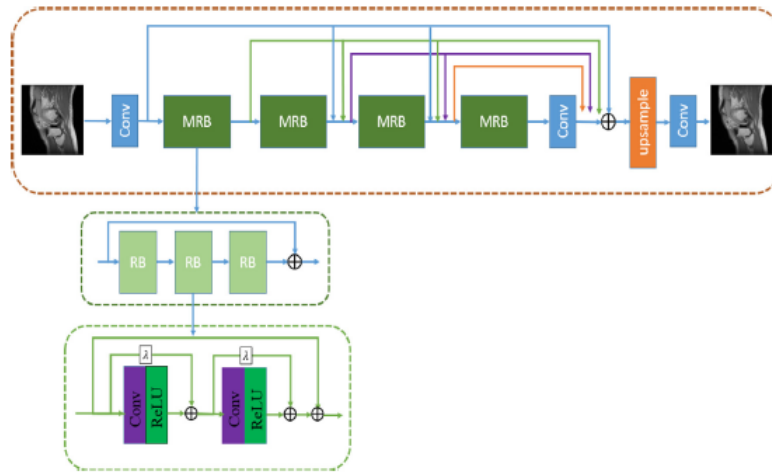


Figura 2.8: Estructura de la Residual dense network *Imagen tomada de:* [39]

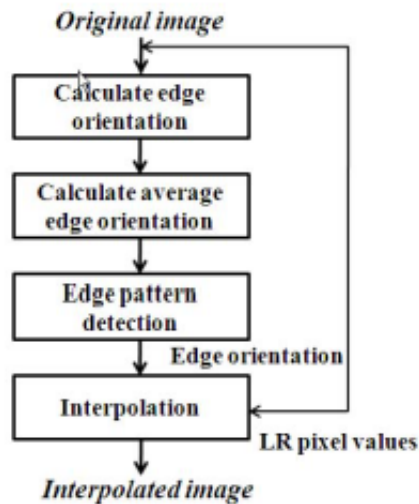


Figura 2.9: Proceso de mejoramiento de resolución y aumento de contraste. *Imagen tomada de:* [15]

En el artículo **DeepRegularizer: Rapid Resolution Enhancement of Tomographic Imaging using Deep Learning (2020)** se realiza un procedimiento para aumentar la resolución de una tomografía óptica de difracción (ODT), mediante el uso de una red neuronal profunda, la cual se entrena con un par de conjuntos de datos, uno de índices refractivos (RI) de tomogramas crudos y otro de RI de tomogramas con resolución mejorada. La red neuronal utilizada se basa en una

U-net CNN combinandola con una búsqueda escalable de arquitectura neuronal (scNAS), el cual se utiliza para optimizar la red. En la red dada una imagen de entrada esta se parte en distintos parches, para que a través de *convolution*, *ReLU*, *global average pooling*, *linear*, *ReLU*, *linear*, *sigmoid*, and *instance norm* se logre expandir dicho parche. Cada neurona toma las salidas de las 2 neuronas previas consecutivas. De esta manera se logra una mejor resolución [19].

En el artículo **Application of Super-Resolution Convolutional Neural Network for Enhancing Image Resolution in Chest CT (2018)** se utiliza una *Super-resolution convolutional neural network* (SRCNN) para mejorar la resolución de imágenes de tomografía computacional del pecho, Para ello dada una imagen de entrada, se le aplican 4 pasos para lograr el mejoramiento de la resolución, en primer una interpolación bicúbica, luego una extracción de subparches, luego un mapeo no-lineal y por último la reconstrucción. En un principio se usa $F_1(Y) = \max(0, w_1 * Y + B_1)$, donde Y es la imagen, w_1 son los filtros y B_1 son los *biases*, en esta capa se utiliza de igual manera ReLu como función de activación y su salida es un mapa de características. En la siguiente capa (mapeo no lineal) se utiliza $F_2(Y) = \max(0, w_2 * F_1(Y) + B_2)$ en la que el mapa anterior se utiliza para mapearlo a otro de características de alta resolución. En la parte de reconstrucción se hace uso de $F_3(Y) = w_3 * F_2(Y) + B_3$ en la que se combinan las características de alta resolución las cuales fueron mapeadas anteriormente a las características originales. Todo este proceso se puede ver reflejado en la figura 2.10 [40].

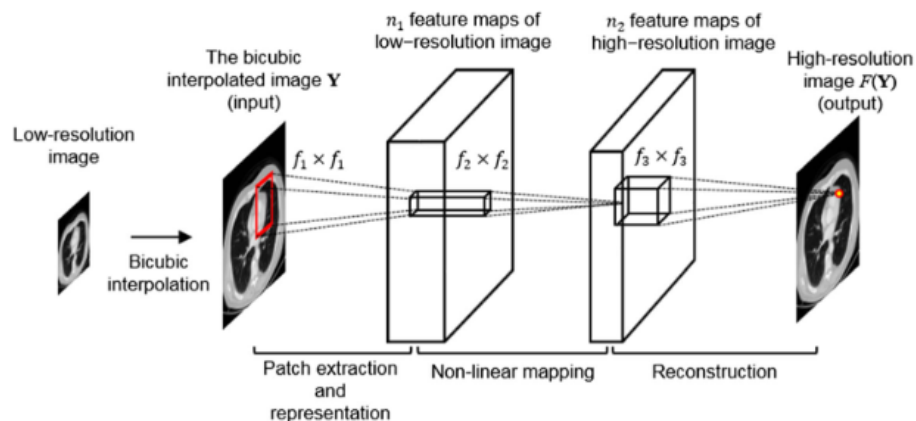


Figura 2.10: Proceso de mejoramiento de resolución por SRCNN. *Imagen tomada de:*[40]

En el artículo *Super-resolution MRI through Deep Learning (2018)* se usa Deep Learning para obtener imágenes de resonancia magnética de mayor resolución (HR) las cuales son muy deseables obteniendo mejoras de la resolución al doble. Para ello, adaptan dos redes neuronales de última generación llamadas GAN-CPCE y GAN-CIRCLE las cuales demostraron una

gran capacidad para producir imágenes de alta resolución(HR)de resonancia magnética a partir de contrapartes de baja resolución(LR), donde $I_{LR}, I_{HR} \in \mathbb{R}^{m \times n}$ son respectivamente imágenes de resonancia magnética LR y HR de tamaño $m \times n$ y su relación puede ser expresada como $I_{LR} = f(I_{HR})$, donde $f: I_{HR} \in \mathbb{R}^{m \times n} \rightarrow I_{LR} \in \mathbb{R}^{m \times n}$ denotando el proceso de muestreo que crea una contraparte LR a partir de una imagen HR. El proceso de generación de imágenes de súper resolución consiste en implementar una función inversa aproximada tal que $g \approx f^{-1}: I_{LR} \approx g(I_{LR})$. GAN-CPCE es una GAN mejorada con la distancia de Wasserstein en combinación con la red neuronal VGG y para este caso cuenta con cuatro capas convolucionales con 32 filtros de 3×3 , tres capas deconvolucionales con 32 filtros de 3×3 y una capa deconvolucionacional con un filtro de 3×3 como se muestra en la figura 2.11. Por otro lado, GAN-CIRCLE incluye dos redes generativas. Cada red generativa consta de una red de extracción de características y una red de reconstrucción. En la red de extracción de características, hay 12 conjuntos de bloques de características de superresolución no lineales compuestos por filtros 3×3 , Leaky ReLU y una capa de abandono como se muestra en la figura 2.12[41].

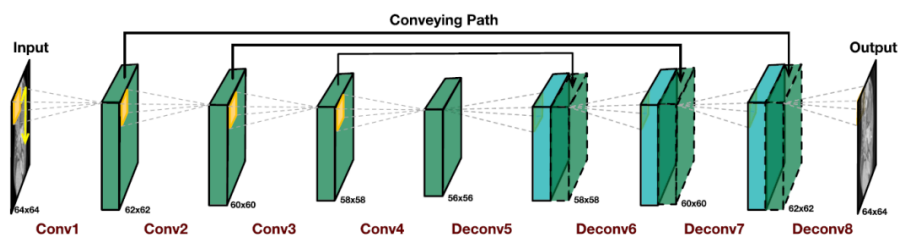


Figura 2.11: Estructura del generador GAN-CPCE. *Imagen tomada de:*[41]

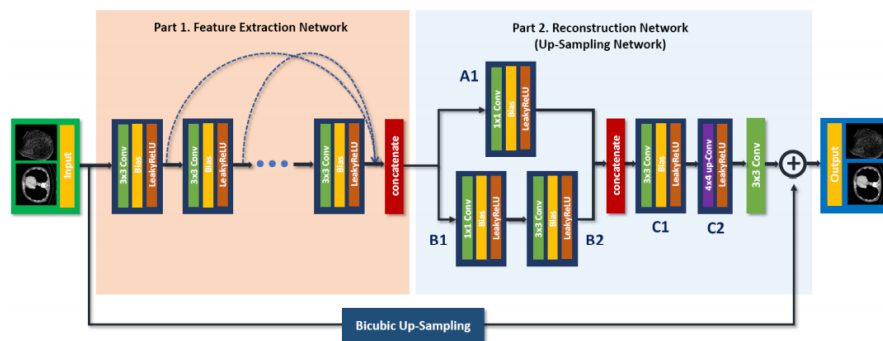


Figura 2.12: Estructura del generador GAN-CIRCLE. *Imagen tomada de:*[41]

En el artículo *Efficient and Accurate MRI Super-Resolution using a Generative Adversarial Network and 3D Multi-Level Densely Connected Network*(2018) proponen un novedoso diseño de red neuronal 3D llamada red de super-resolución de varios niveles densamen-

te conectada (mDCSRN) con *generative adversarial network*(GAN) con entrenamiento guiado. El mDCSRN entrena y hace inferencias rápidamente, y la GAN promueve una salida realista difícilmente distinguible de las imágenes originales de alta resolución(HR). Los autores plantean tres pasos diferentes que se optimizan juntos: extracción de características, aprendizaje múltiple y reconstrucción de imágenes. Durante el entrenamiento, la diferencia entre las imágenes reconstruidas y las imágenes reales se usa para ajustar capa de reconstrucción para restaurar mejores imágenes y para guiar la extracción de características de imágenes precisas. Esa combinación de diferentes componentes hace posible que la red neuronal logre un rendimiento de vanguardia entre otras técnicas de SISR. En la figura 2.13 se puede apreciar la arquitectura propuesta por los autores, donde cuenta con un *DenseBlock*(A) con convoluciones de $3 \times 3 \times 3$ y la red (B, C) mDCSRN-GAN[42].

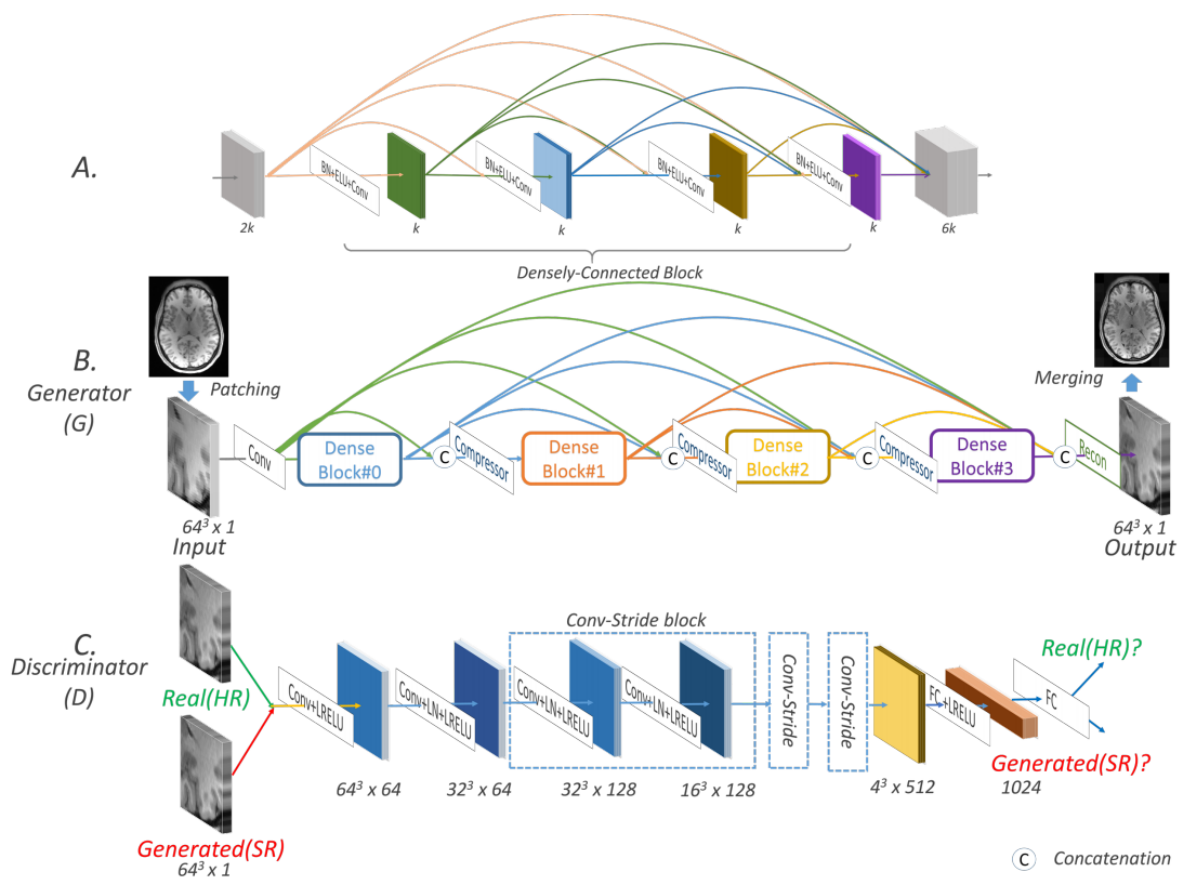


Figura 2.13: Arquitectura de la red mDCSRN. Imagen tomada de:[42]

En el artículo *Super-resolution musculoskeletal MRI using deep learning(2018)* desarrollan una técnica de super-resolución utilizando redes neuronales convolucionales para generar imágenes de Resonancia Magnética de corte fino de la rodilla a partir de cortes de entrada más gruesos y comparar este método con métodos alternativos de interpolación. La red fue denominada

DeepResolve la cual aprende transformaciones basadas en residuos entre imágenes de corte fino de alta resolución e imágenes de corte grueso de baja resolución en las mismas ubicaciones centrales. La red DeepResolve calcula una imagen residual de una imagen de entrada de baja resolución(LR) para generar la imagen de alta resolución(HR) correspondiente como se muestra en la figura 2.15 y consta de 19 capas de bloques de activación no lineales de convolución emparejada, unidad lineal rectificadora (ReLU) y una capa convolucional final sin una activación asociada. La primera y última capa de DeepResolve tiene dimensiones de $32 \times 32 \times 32$, mientras que todas las demás capas tienen dimensiones de $32 \times 32 \times 32 \times 64$ y la capa final no tiene la función de activación. En cuanto a datos, manejaron un conjunto de datos 3D de entrada y lo dividieron en parches isotrópicos de $32 \times 32 \times 32$ píxeles, y todos los parches de todos los pacientes se utilizaron durante el proceso de formación de DeepResolve tal y como se muestra en la figura 2.14 [43].

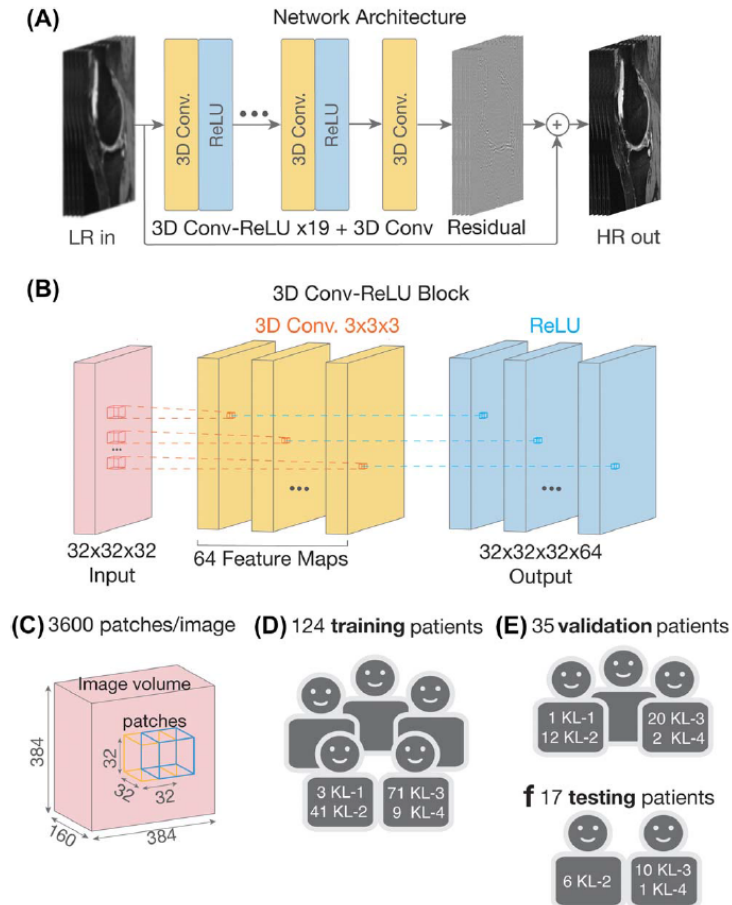


Figura 2.14: Arquitectura DeepResolve, número de pacientes, datos usados y división de datos. Imagen tomada de:[43]

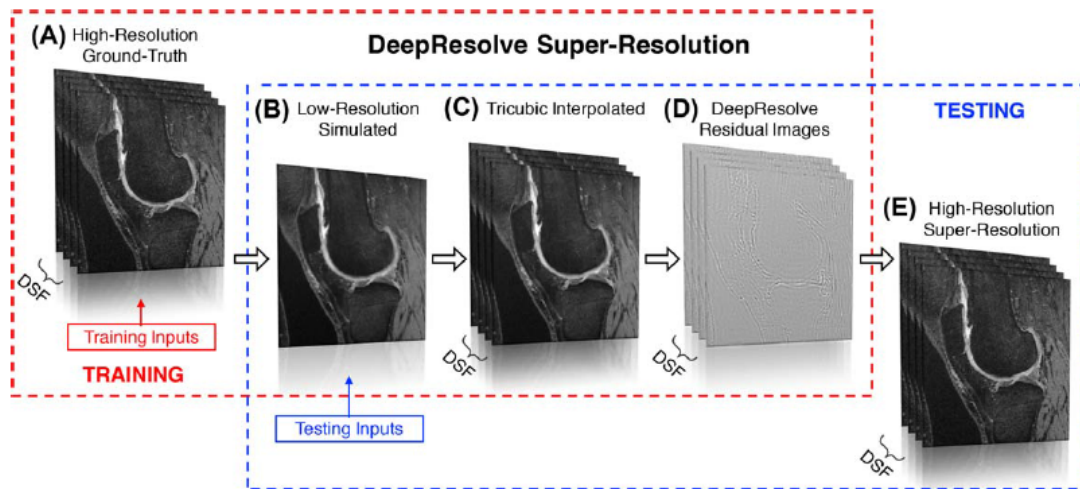


Figura 2.15: Proceso de obtención de imagen HR. *Imagen tomada de:*[43]

En el artículo *Brain MRI super resolution using 3D deep densely connected neural networks(2018)* presentan una nueva arquitectura de red neuronal llamada red de super-resolución densamente conectada en 3D (DCSRN) para restaurar las características de las imágenes de resonancia magnética cerebral estructural, la cual logra un entrenamiento más rápido, es un modelo liviano, tiene menos sobraajuste durante el entrenamiento ya que reducen el número de parámetros y las funciones se reutilizan en gran medida. La red recibe parches sacados de la imagen 3D completa. Luego, se aplica una capa convolucional con un tamaño de filtro de 3×3 y un número de filtro de $2k$ a la imagen de entrada antes del bloque densamente conectado con 4 unidades, cada una de ellas tiene una capa de normalización por lotes y una función de activación de unidades lineales exponenciales (ELU) seguida de una capa convolucional con k filtros. Por último, se utiliza una capa de convolución para proporcionar una salida SR final como se puede ver en la figura 2.16[44].

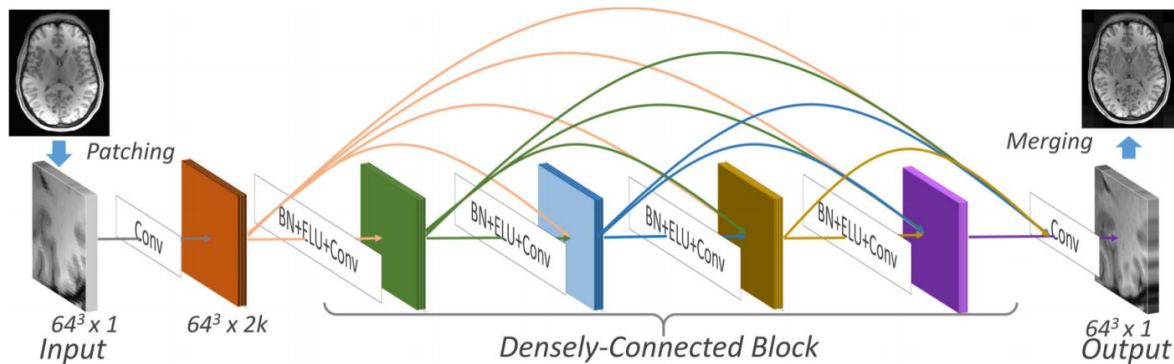


Figura 2.16: Arquitectura DCSRN. Imagen tomada de:[44]

2.2 Metodología

2.2.1. Tipo de Estudio

Este proyecto es de tipo experimental pues se planea poder escoger 3 modelos de *deep learning*, con los cuales se pueda obtener imágenes dMRI con una resolución mejorada, haciendo ajustes a los distintos parámetros e hiperparámetros de cada arquitectura. Reportando los resultados obtenidos, así como las optimizaciones implementadas.

2.2.2. Actividades

Para la solución de este proyecto se proponen las siguientes actividades basadas en cada uno de los objetivos específicos propuestos.

1. **Objetivo Específico 1:** *Implementar 3 modelos de aprendizaje profundo para generar estudios dMRI de alta resolución.* Para este objetivo se pretende realizar el proceso de investigación y desarrollo de las 3 técnicas seleccionadas.

- a) Investigar los 3 modelos(SRCNN, EDSR Y VDSR): En esta actividad se investigará el estado del arte en cuanto a los modelos básicos de estos 3 tipos de redes neuronales y acerca de sus variaciones para encontrar que arquitecturas se ajustan más a nuestra problemática. También, se investigará acerca de los parámetros e hiperparámetros que más influyen en la red, tales como el tamaño del filtro o kernel, la profundidad de la capa convolucional, el stride o paso, el número de neuronas por capa, el número de capas, las funciones de activación, la dimensionalidad de la entrada, las funciones de pérdida, las funciones de *Pooling*, entre otros.

- b) Implementar los 3 modelos(SRCNN, EDSR Y VDSR): En esta actividad se buscará aplicar los conocimientos adquiridos en la actividad anterior para implementar los 3 modelos y probar su funcionamiento
2. **Objetivo Específico 2:** *Ajustar las arquitecturas de las redes de aprendizaje profundo a modo de que estas puedan recibir como entrada estudios dMRI.* En este objetivo se pretende realizar cambios sobre la implementación de cada modelo para poder utilizar el tipo de imágenes que se obtiene de un estudio dMRI, esto debido a que dichas imágenes son distintas a las mas comunes por lo que se debe adaptar la red para su correcto uso.
- a) Investigar a profundidad la interpretación computacional de las imágenes dMRI así como la arquitectura de los 3 modelos implementados(SRCNN, EDSR Y VDSR): Investigar a profundidad el manejo de imágenes dMRI en el ámbito computacional, de igual forma investigar la manera para adaptar los modelos de aprendizaje profundo para que estos puedan utilizarse con imágenes dMRI.
- b) Realizar ajustes en la arquitectura de cada uno de los 3 modelos(SRCNN, EDSR Y VDSR): Esta actividad tiene como finalidad el poder adaptar los modelos, especialmente la capa de entrada (*input*) para que estos puedan tomar el estudio dMRI como insumo.
3. **Objetivo Específico 3:** *Evaluar el rendimiento de los métodos propuestos empleando métricas de error en imágenes y validación morfológica.* Finalmente, se busca evaluar los resultados usando métricas de error en imágenes y validación morfológica. Además, se realizará un análisis comparativo de resultados entre los 3 modelos y se plasmarán en el documento de entrega.
- a) Investigar y seleccionar las métricas de error en imágenes: En esta tarea se investigará acerca de métricas las cuales serán útiles para el análisis de resultados y evaluar cuán parecidos son a los datos reales, como lo son *structural similarity*(SSIM), PSNR, MSE, RMSE, entre otras métricas que se encuentran en librerías de Python como sklearn y scikit image.
- b) Investigar acerca de los métodos para realizar validación morfológica: En esta tarea se investigará acerca de los métodos existentes para validar la preservación de características morfológicas, como lo puede ser la comparación de características de la imagen obtenida a través de la metodología propuesta, con el *ground truth*, esta comparación se puede hacer a través de la segmentación de tejidos.
- c) Evaluar los modelos con los métodos de validación morfológica: Después de haber seleccionado los modelos, se procederá a realizar la validación de cada metodología (*SRCNN, EDSR Y VDSR*), en la que se verificará la preservación de las características morfológicas.
- d) Evaluar los modelos con las métricas seleccionadas: Después de haber seleccionado las métricas, se calcularán los valores de las mismas y a partir de ellas se evaluará por separado los resultados de cada uno de los modelos.

- e) Comparar los resultados obtenidos de los 3 modelos: A partir de las métricas ya calculadas en la anterior actividad, se compararán y clasificarán los 3 modelos buscando mostrar con cual se obtuvieron mejores resultados.
- f) Artículo publicable donde se evidencien los resultados del proyecto: En esta actividad se realizará un artículo en el que se plasmará la metodología, la explicación de las redes implementadas y los resultados obtenidos con el objetivo de publicarlo.
- g) Documentación y presentación del proyecto: En esta actividad se realizará el documento entregable del proyecto y la respectiva documentación junto a los algoritmos creados para la solución del problema y los resultados obtenidos para después ser presentado a los respectivos jueces.

Desarrollo Metodológico

Para abordar el problema de aumento de la resolución de imágenes en una escala de 2, se emplearon tres arquitecturas de redes neuronales: SRCNN (Super Resolution Convolutional Neural Network), VDSR (Very Deep Super Resolution) y EDSR (Enhanced Deep Super Resolution). En esta sección, se describe en detalle la metodología utilizada para implementar y entrenar las 3 redes, además de describir el entorno de ejecución, el uso de la GPU para acelerar las tareas y el preprocesamiento de los datos.

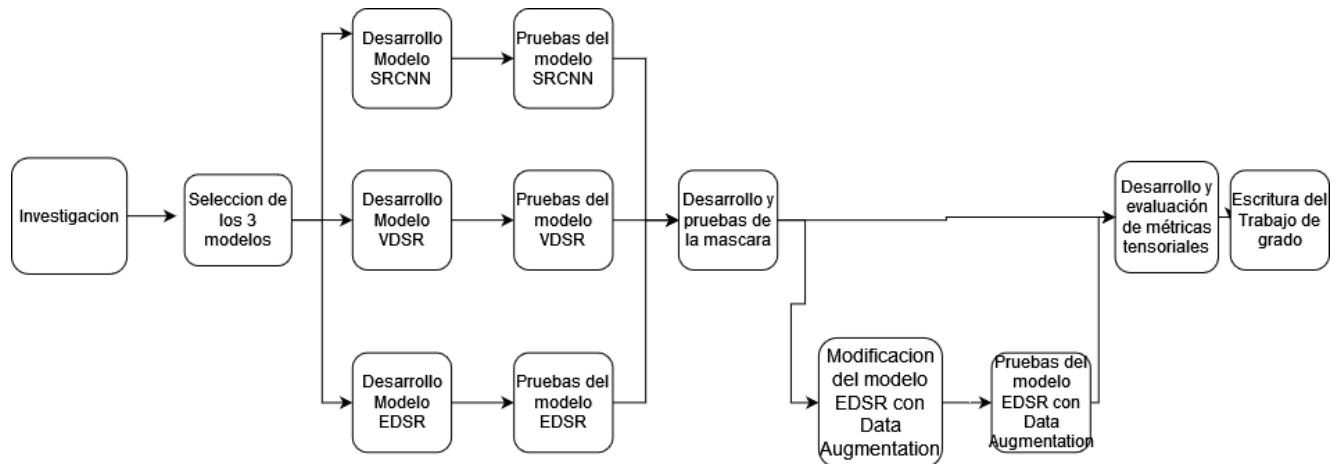


Figura 3.1: Flujo del desarrollo del proyecto

La figura 3.1 muestra los distintos momentos a lo largo del desarrollo del proyecto de grado, se puede apreciar como en primer lugar hubo un proceso de investigación para poder seleccionar los 3 modelos con los cuales se iba a trabajar (SRCNN, VDSR y EDSR), posteriormente se realiza en paralelo el desarrollo y adaptación de los 3 modelos y se realizan pruebas con diferente número de épocas.

A raíz de dichas pruebas se nota un desempeño aceptable, sin embargo, no es el esperado, por lo que se toma la decisión de aislar la región de interés (el cerebro) para la evaluación de los modelos. Una vez hecho esto el modelo EDSR sigue sin tener un buen desempeño por lo que se opta por utilizar técnicas de *Data Augmentation*. Los resultados obtenidos con la nueva técnica son mejores y se procede a realizar la evaluación a nivel tensorial de las imágenes generadas para finalmente escribir el trabajo de grado.

3.1 Entorno de ejecución

Para llevar a cabo este trabajo, se utilizó la plataforma Google Colab en su versión Pro. En la versión gratuita, la herramienta dispone de 12.8GB de RAM, 125GB de disco y una GPU de tamaño muy limitado. Sin embargo, la versión Pro ofrece 25.46GB de RAM, 225GB de disco y acceso a GPUs de alto rendimiento como las unidades A100, V100 y T4. Estas GPUs proporcionan diferentes capacidades de procesamiento y memoria, lo que permite a los usuarios trabajar con modelos de Deep Learning más grandes y entrenarlos de manera más rápida y eficiente. Además, Google Colab se integra con Google Drive, lo que facilita el acceso y almacenamiento de datos.

Los resultados obtenidos demostraron que la utilización de Google Colab para la implementación de modelos de Deep Learning es altamente efectiva, permitiendo obtener modelos de alta precisión y un rendimiento significativamente mejorado en comparación con el uso de computadoras personales que tienen características inferiores a los servidores en los cuales se ejecuta Google Colab.

3.1.1. Uso de la GPU

Como se explicó anteriormente, uno de los beneficios de usar Google Colab Pro en comparación con la versión gratuita y hacerlo localmente, es la reducción significativa de los tiempos de ejecución. Se observó que con la versión gratuita, un entrenamiento podía tomar alrededor de 10 horas, mientras que con el uso de Colab Pro, que cuenta con más RAM y una GPU más potente, el tiempo se redujo a 17 minutos. Por esta razón, se eligió pagar y utilizar la versión Pro. Para lograr esta optimización del tiempo, se realizó un cambio en la configuración del notebook en Google Colab. Se seleccionó la opción de acelerador de hardware “GPU” y se eligió la A100 en el tipo de GPU. Además, la opción de especificación del tiempo de ejecución viene por defecto con una configuración de RAM amplia.

3.2 Exploración del Conjunto de Datos

Tal y como se mencionó en la sección 1.4 el conjunto de datos del cual se hizo uso fue un estudio que se realizó el Dr. Hernán Darío Vargas Cardona en su zona craneal, y claramente se tuvo su consentimiento para el uso del mismo. Este conjunto de datos tiene como dimensiones $256 \times 256 \times 33 \times 26$, donde las primeras 2 dimensiones hacen referencia a la resolución de las imágenes, y las siguientes hacen referencia a los cortes y los gradientes, respectivamente. Los datos varían en un rango de 0 a 23995 y pertenecen a un estudio de la zona craneal de un paciente como se puede ver en la figura 3.2.

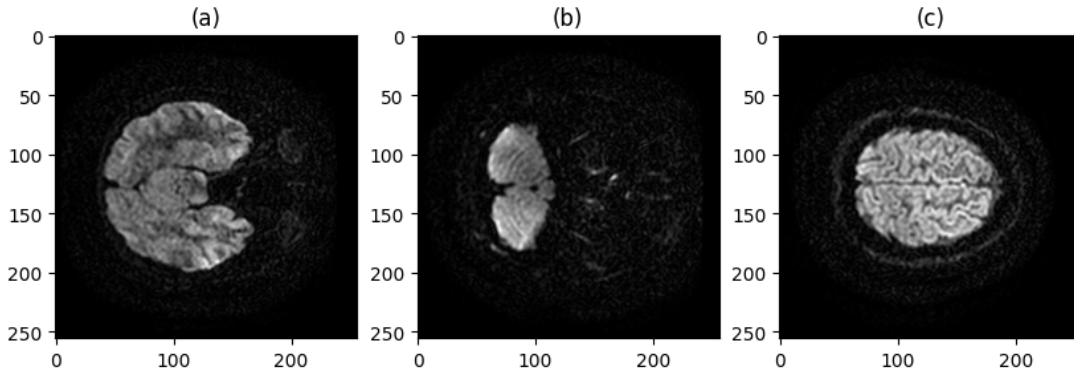


Figura 3.2: Imágenes bidimensionales del conjunto de datos

3.3 Preprocesamiento

En primer lugar, dado el hecho de que se están utilizando redes neuronales con capas convolucionales de 2 dimensiones, es necesario cambiar la dimensionalidad del conjunto de imágenes. Para ello, se utiliza la función ‘reshape’ perteneciente a la librería ‘NumPy’ de Python. Como resultado, se obtiene un arreglo de dimensiones (256 x 256 x 858), es decir, 858 imágenes bidimensionales con 256 píxeles tanto de alto como de ancho.

Posteriormente, se realiza un proceso de normalización del conjunto de imágenes con el fin de reducir el rango con el que los modelos deben trabajar y así obtener mejores resultados. Esto se logra mediante el uso de la fórmula mencionada. De esta forma, los datos que anteriormente variaban en un rango de 0 a 23955 ahora se encuentran en un rango de 0 a 1 para las redes VDSR y SRCNN.

$$X_{normalized} = \frac{X}{X_{max}}$$

Por otro lado, con el objetivo de adaptar la entrada al modelo definido para EDSR, se normalizó el conjunto de datos utilizando la siguiente fórmula, de manera que cada píxel quedara con un valor entre 0 y 255.

$$X_{normalized} = \frac{X}{X_{max}} * 255$$

Hecho esto, se procede a transponer los ejes del conjunto de datos de tal manera que quede de la forma: (858 x 256 x 256). Para lograrlo, se utiliza nuevamente la función *transpose* de *NumPy*. Esta operación se realiza debido a que los modelos reciben los arreglos en esta forma, donde el primer eje representa la cantidad de imágenes y los ejes siguientes hacen referencia a las dimensiones de las imágenes.

Posteriormente, se realiza una copia del conjunto de imágenes que se someterá a un proceso de *downgrading* para obtener imágenes de baja resolución. Para lograr esto, se reduce la resolución eliminando uno de cada dos píxeles, lo que resulta en un conjunto de imágenes de dimensiones: (858 x 128 x 128). Luego, utilizando la función *resize* de *TensorFlow*, se aumenta nuevamente la resolución de las imágenes a 256 x 256 píxeles utilizando una interpolación bicúbica.

Este proceso se realiza debido a que todos los modelos utilizados en el proyecto pertenecen a la categoría de *pre-upscaling*, como se indica en las referencias [20, 30], donde se realiza primero un proceso de *upsampling* antes de utilizar las imágenes en el entrenamiento. Este proceso se puede observar en la figura 3.3.

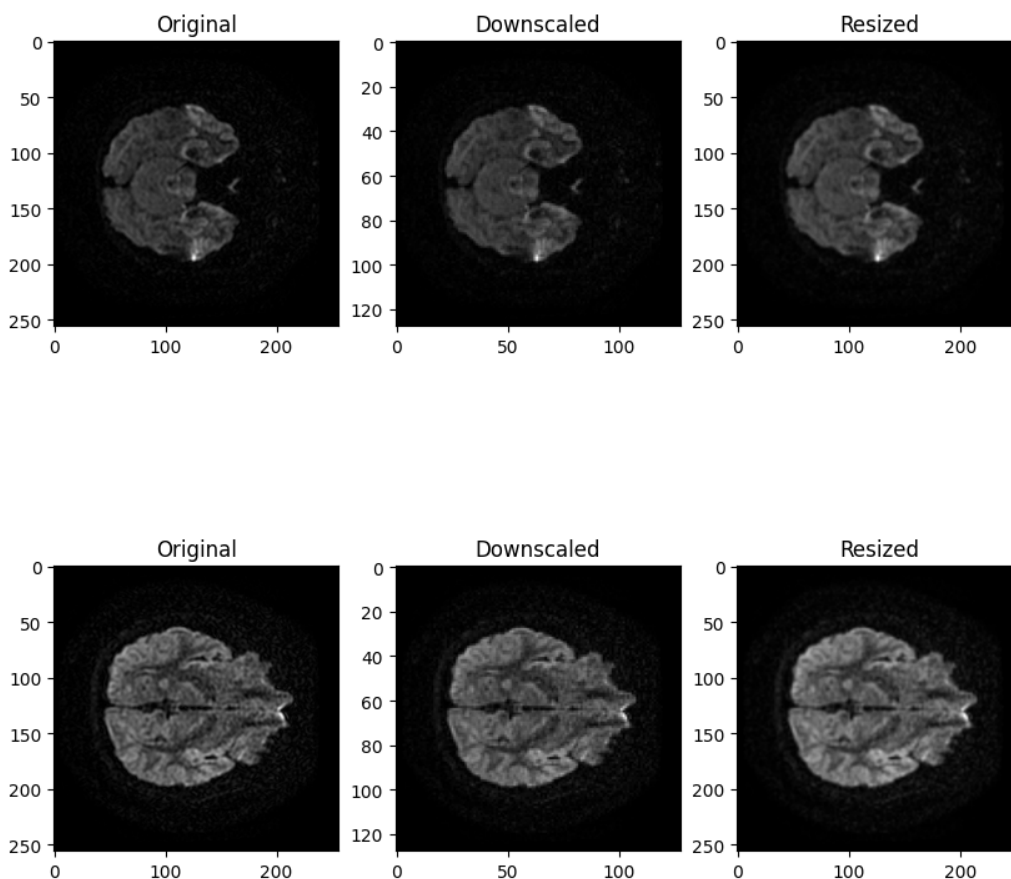


Figura 3.3: Ejemplo de procesamiento de las imágenes

Por último, se realiza la separación en *train* y *test* mediante el uso de la función *train_test_split* de la librería *scikit-learn* dejando un 85 % para training y un 15 % para testing.

3.3.1. Utilización de Máscara

Uno de los desafíos encontrados fue el hecho de la existencia de ruido fuera de la zona craneal, como se puede apreciar en la figura 3.2. Este ruido no solo es irrelevante para el estudio, sino que también genera un mayor error al calcular las métricas de desempeño de los modelos. Por esta razón, se opta por implementar una máscara para las imágenes una vez que han sido predichas por los modelos.

Esta máscara se crea utilizando la librería *OpenCV*, que permite encontrar el contorno de la imagen y recortar todo lo que se encuentra fuera de ese contorno. Inicialmente, se pensó en utilizar un recorte dinámico para cada imagen, de modo que cada una tuviera su propio contorno. Sin embargo, esto no se llevó a cabo debido a la presencia de imágenes como la (b) de la figura 3.2, en la que el contorno dejaba fuera parte del cráneo y se perdía información relevante.

Para solucionar esto, se decidió utilizar una máscara genérica para todas las imágenes, utilizando la imagen mostrada en la figura 3.4 cuyo contorno se adapta mejor al área del cráneo. Esto hace que algunas imágenes contengan más ruido que otras, sin embargo, todas las imágenes experimentan una reducción considerable de ruido, como se puede apreciar en la figura 3.5. Esto permite estudiar el desempeño en la región que es verdaderamente relevante.

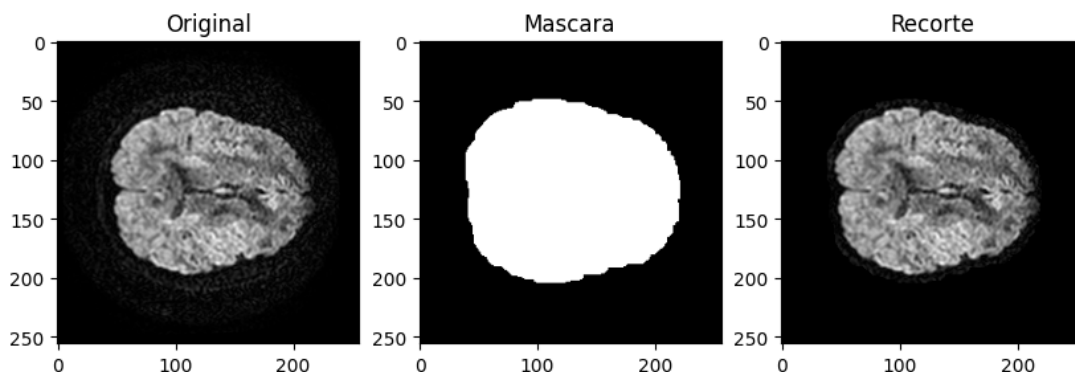


Figura 3.4: Creación de la máscara

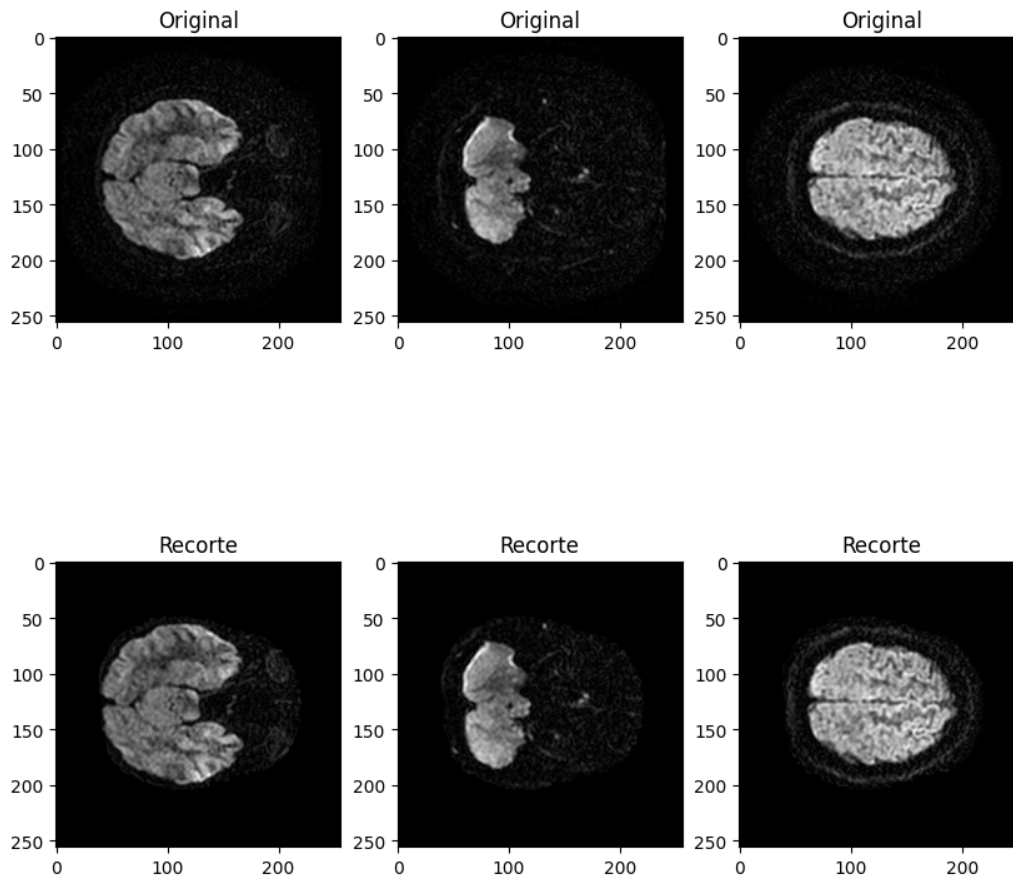


Figura 3.5: Ejemplo de la mascara en diferentes imágenes

3.4 Modelo SRCNN

El modelo de Super Resolution CNN (SRCNN) es un tipo de red neuronal convolucional o *CNN* por sus siglas en ingles, que tiene como proposito realizar procesos de super resolución los cuales se explicaron en 2.1.2.

3.4.1. Implementación del Modelo

Para implementar el modelo de SRCNN primero se investigaron distintas maneras de implementarlo, finalmente basandose en el articulo de Dong [45], una implementación oficial ofrecida como ejemplo en la documentación de Keras y un repositorio de Github¹ que tambien toma el trabajo

¹El link al que se hace referencia: https://github.com/goodboychan/goodboychan.github.io/blob/main/_notebooks/2020-10-13-01-Super-Resolution-CNN.ipynb

de Dong como referencia, se logra implementar la red utilizando el modelo *Sequential* propio de Keras, este modelo como su nombre lo indica, se basa apilar capas una detrás de la otra por lo que sean $c_1, c_2, c_3, \dots, c_{n-1}, c_n$ n capas de una red neuronal, la capa de entrada sería c_1 , y de ahí en adelante la entrada de una capa es la salida de la capa inmediatamente anterior, es decir:

$$\forall i | i > 1 \Rightarrow Output(c_{i-1}) = Input(c_i).$$

Donde c_i hace referencia a la i ésima capa de la red neuronal

Esta red realiza 3 operaciones: Extracción y representación de características, mapeo de características y reconstrucción.

3.5 Modelo VDSR

Como se mencionó previamente en la sección 2.1.2, la VDSR es un enfoque basado en el aprendizaje profundo para la mejora de la resolución de imágenes individuales (SISR). Este método utiliza una red neuronal convolucional muy profunda para mejorar la calidad de las imágenes de baja resolución, y se inspira en la red VGG-net utilizada en la clasificación de ImageNet[33].

3.5.1. Implementación del Modelo

El proceso para utilizar el enfoque VDSR se realizó una búsqueda exhaustiva de implementaciones que utilizaban diferentes bibliotecas para implementarla. Después de investigar y evaluar varias opciones, se encontró una implementación adecuada en el este repositorio de [Github](#)².

Dentro de este repositorio, se identificó el archivo que contiene la definición de la red VDSR. Este archivo es el componente principal necesario para utilizar el modelo VDSR en el proceso de super resolución. Se hizo uso exclusivo de este archivo en particular y se omitieron los demás archivos y recursos proporcionados en el repositorio.

La implementación se basa en la biblioteca Keras, una popular biblioteca de aprendizaje profundo, que facilita el desarrollo y entrenamiento del modelo VDSR.

3.6 Modelo EDSR

La EDSR (Enhanced Deep Super-Resolution) es una red neuronal utilizada para el problema de super resolución (SR) que utiliza bloques residuales con conexiones de salto que son una estructura clave utilizada en las redes neuronales convolucionales para mejorar el rendimiento y la capacidad de aprendizaje de la red[31].

3.6.1. Implementación del Modelo

El proceso de implementación de la EDSR (Enhanced Deep Super Resolution Network) se llevó a cabo utilizando el algoritmo encontrado en la página oficial de Keras el siguiente [enlace](#)³ y se

²El link al que se hace referencia: <https://github.com/Likarian/VDSR-keras/tree/14a6bf94c7d23f197bcf0a18353bdcb4c7e3bd8c>

³El link al que se hace referencia: <https://keras.io/examples/vision/edsr/>

aprovechó todo el contenido disponible en el repositorio para ajustar y aplicar de manera efectiva la red EDSR en el proyecto.

Adicionalmente, se ajustó la red para que tomara como entrada las imágenes utilizadas en el proyecto. En lugar de imágenes de 3 canales en escala de 0 a 255, el proyecto requería imágenes de un solo canal en escala de 0 a 1. Para satisfacer esta necesidad, se crearon funciones específicas en el código para convertir las imágenes de un solo canal a imágenes de 3 canales y viceversa, además se ajustaron los datos para que quedaran entre 0 y 255 ya que así estaba configurada la red.

Se estableció que las imágenes de un solo canal se convertirían a imágenes de 3 canales duplicando los valores de píxeles en los otros 2 canales. Esta duplicación garantiza que las imágenes resultantes sean equivalentes en información visual a las imágenes originales de un solo canal.

3.6.2. Aumento del Conjunto de Datos

Con el objetivo de mejorar la capacidad del modelo para generalizar a nuevas imágenes, esta técnica es fundamental ya que enriquece el conjunto de entrenamiento. Al aplicar recortes aleatorios, volteos horizontales y rotaciones aleatorias, se incrementa la diversidad del conjunto de datos, previniendo el sobreajuste y mejorando el rendimiento del modelo EDSR en la tarea de super resolución de imágenes. Dentro del código, se realizó el aumento del conjunto de datos mediante varias técnicas. Se utilizó la función *randomcrop* para recortar aleatoriamente las imágenes de entrada a un tamaño específico, lo que introduce variaciones en la posición y el tamaño de las imágenes.

Además, se aplicó la función *randomfliplefttright* para realizar volteos horizontales aleatorios en las imágenes. Esto ayuda a aumentar la variabilidad al enfrentar diferentes orientaciones de objetos en las imágenes.

Otra función que se utilizó fue *randomrotate*, que aplica rotaciones aleatorias a las imágenes, lo que permite simular diferentes ángulos de captura y mejorar la robustez del modelo.

Las figuras 3.6 y 3.7 muestran algunos ejemplos de imágenes resultantes después de aplicar estas 3 funciones.

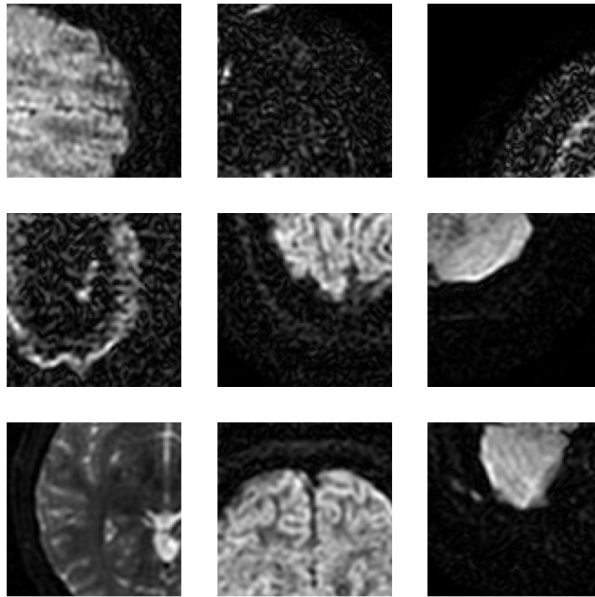


Figura 3.6: Imágenes generadas con rotaciones, cortes y reflejadas de las imágenes de alta resolución

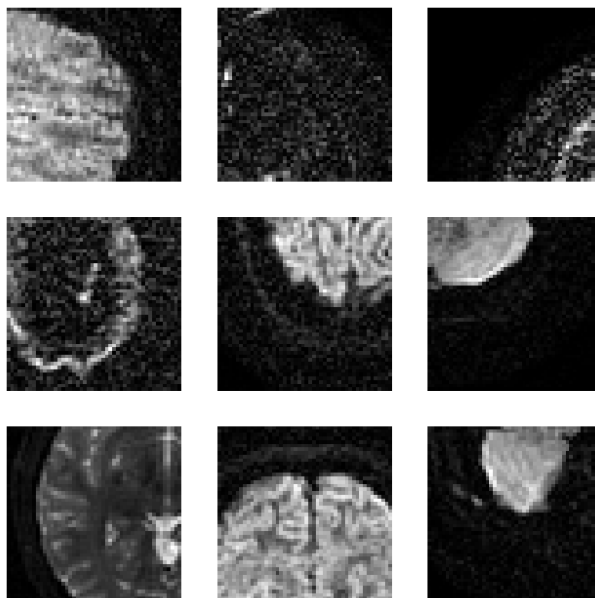


Figura 3.7: Imágenes generadas con rotaciones, cortes y reflejadas apartir de las imágenes de baja resolución

Resultados

Para realizar la evaluación de los modelos se utilizaron 6 métricas distintas las cuales permiten hacer un análisis del desempeño de los modelos, 4 de estas métricas se utilizan para evaluar el desempeño de los modelos a nivel de imágenes y estas son: Structural Similarity Index (Ssim), Peak Signal to Noise Ratio (Psnr), Mean Squared Error (Mse) y Rooted Mean Squared Error (Rmse), las otras 2 métricas se utilizan para evaluar los modelos a nivel tensorial las cuales son Frobenius Norm (Fnorm) y Mean Squared Error de la Fractional Anisotropy (Mse of FA).

En estas fórmulas, m y n son las dimensiones de las imágenes, de igual manera I y L así como T^1 y T^2 hacen referencia a los valores *ground truth* y los valores predichos respectivamente, la diferencia es que en el primero es a nivel de imagen y en el segundo es a nivel tensorial.

$$MSE = \frac{1}{mn} \sum_{i=0}^m \sum_{j=0}^n I_{ij} - L_{ij}$$

$$RMSE = \sqrt{MSE}$$

$$PSNR = 20 \log_{10} \left(\frac{MAX}{MSE} \right)$$

$$FNORM = \sqrt{\text{trace}[(T^1 - T^2)^\top (T^1 - T^2)]}$$

$$SSIM = \frac{(2\mu_I \mu_L + C_1) + (2\sigma_{IL} + C_2)}{(\mu_I^2 + \mu_L^2 + C_1) + (\sigma_I^2 + \sigma_L^2 + C_2)}$$

1. **MSE:** Es una de las metricas más utilizadas en la evaluación de calidad de imágenes, se basa en calcular la intensidad al cuadrado de las diferencias entre las imágenes y luego promediarla. A raíz de esto se busca minimizar este resultado [46].
2. **RMSE:** Es una metrica que indica qué tan dispersos se encuentran los resultados obtenidos de la media. Tal y como con el MSE, esta se busca minimizar [46].
3. **SSIM:** La particularidad de la métrica es que toma en consideración factores biológicos del sistema visual ser humano (luminiscencia, contraste y estructura) para la evaluación de las imágenes. Los valores varían entre -1 y 1, -1 siendo anti-correlación, 0 la no similaridad y 1 similaridad perfecta [46].

4. **PSNR:** Es un métrica que permite calcular el ratio entre el máximo valor posible para la señal, en este caso la imagen, y la distorsion o error de la imagen reconstruida. Esta métrica se busca maximizar pues en caso de tener 2 imágenes iguales esta retorna infinito [46].
5. **FNORM:** Es una norma matricial que trata a una matriz de $n \times m$ como un vector de tamaño nm hayando así la norma vectorial de dicho vector. Con esta métrica se pretende hallar la magnitud de la diferencia a nivel tensorial por lo que se busca minimizar.
6. **MSE of FA:** La métrica se calcula de la misma manera que MSE mostrada anteriormente con la diferencia de que en vez de tomar el valor por pixel de la imagen, se toma el valor que se obtiene por voxel para la evaluación de la anisotropía fraccional. Tal y como con la métrica de MSE esta se busca minimizar.

4.1 Modelo SRCNN

A continuación, se muestran los resultados de los diferentes entrenamientos realizados con el modelo SRCNN. Las figuras 4.1, 4.2 y 4.3 logran evidenciar el hecho de que no se está teniendo un problema de *overfitting* ni *underfitting*, esto gracias a que tanto el *Validation loss* como el *training loss* disminuyen a medida que pasan las épocas, adicionalmente el validation loss no se dispara.

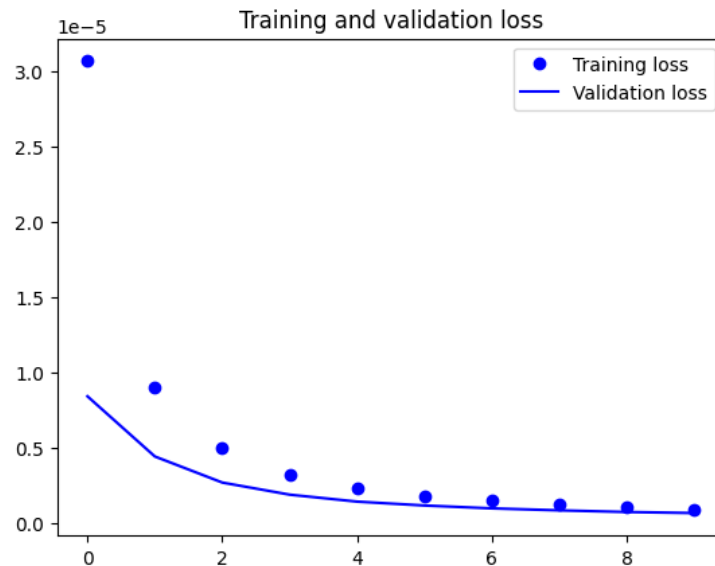


Figura 4.1: épocas vs Training loss y validation loss; épocas = 10

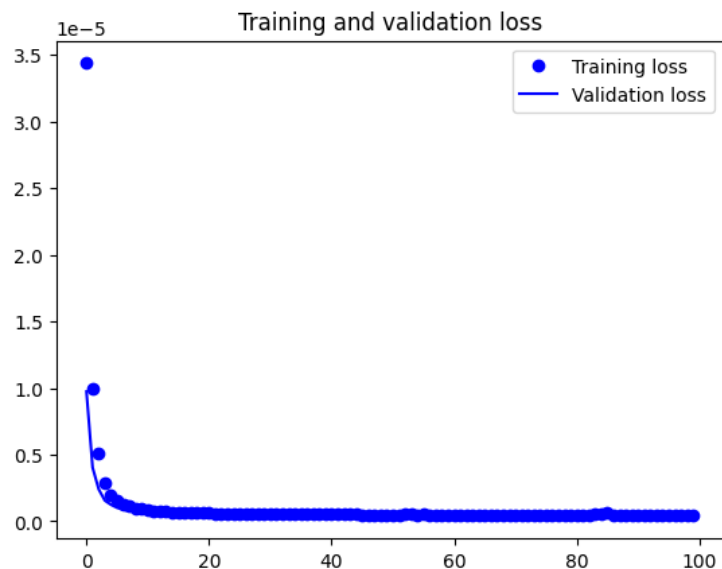


Figura 4.2: épocas vs Training loss y validation loss; épocas = 100

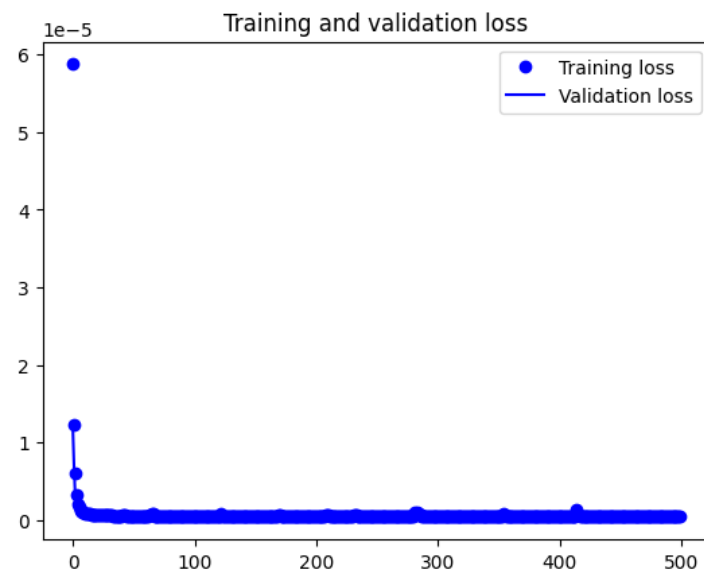


Figura 4.3: épocas vs Training loss y validation loss; épocas = 500

4.1.1. Tiempos de Ejecución

Para la toma de los tiempos de entrenamiento se variaron las cantidades de épocas sobre las que se iba a realizar el entrenamiento, ahora bien, para cada una de dichas cantidades se realizan

5 iteraciones y se toma el tiempo promedio medido en segundos, el cual se ve reflejado en la tabla 4.1, estos entrenamientos se hacen con un *batchsize* de 32.

épocas	Tiempo(s)
10	21.502
100	72.804
500	356.080

Cuadro 4.1: Tiempos de entrenamiento dadas n épocas con *batchsize* 32 modelo SRCNN

4.1.2. Métricas de Imágenes

Utilizando las métricas anteriormente mencionadas se muestran los resultados tanto para las imágenes con recorte y sin recorte para 10, 100 y 500 épocas.

Métricas con 10 épocas		
	Sin recorte	Con recorte
SSIM	$9,50 \times 10^{-1} \pm 1,29 \times 10^{-2}$	$9,73 \times 10^{-1} \pm 0,91 \times 10^{-2}$
MSE	$6,93 \times 10^{-7} \pm 5,99 \times 10^{-7}$	$4,43 \times 10^{-7} \pm 5,57 \times 10^{-7}$
RMSE	$8,00 \times 10^{-4} \pm 2,29 \times 10^{-4}$	$6,33 \times 10^{-4} \pm 2,05 \times 10^{-4}$
PSNR	$1,103 \times 10^2 \pm 2,35$	$1,123 \times 10^2 \pm 2,08$
Métricas con 100 épocas		
	Sin recorte	Con recorte
SSIM	$9,63 \times 10^{-1} \pm 9,83 \times 10^{-3}$	$9,83 \times 10^{-1} \pm 7,90 \times 10^{-3}$
MSE	$4,09 \times 10^{-7} \pm 1,92 \times 10^{-7}$	$2,05 \times 10^{-7} \pm 1,38 \times 10^{-7}$
RMSE	$6,23 \times 10^{-4} \pm 1,42 \times 10^{-4}$	$4,41 \times 10^{-4} \pm 1,04 \times 10^{-4}$
PSNR	$1,124 \times 10^2 \pm 2,19$	$1,154 \times 10^2 \pm 1,76$
Métricas con 500 épocas		
	Sin recorte	Con recorte
SSIM	$9,63 \times 10^{-1} \pm 9,81 \times 10^{-3}$	$9,83 \times 10^{-1} \pm 7,91 \times 10^{-3}$
MSE	$4,04 \times 10^{-7} \pm 1,87 \times 10^{-7}$	$2,02 \times 10^{-7} \pm 1,32 \times 10^{-7}$
RMSE	$6,20 \times 10^{-4} \pm 1,41 \times 10^{-4}$	$4,38 \times 10^{-4} \pm 1,02 \times 10^{-4}$
PSNR	$1,125 \times 10^2 \pm 2,18$	$1,154 \times 10^2 \pm 1,76$

Cuadro 4.2: Comparación de métricas con diferentes épocas y recortes modelo SRCNN

Se puede observar que el realizar un recorte en las imágenes efectivamente hace que se obtengan mejores resultados, esto gracias a que se deja de tomar en cuenta el ruido existente por fuera del cráneo, además, es fácil notar en el cuadro 4.2 que, en este caso, el nivel de mejora que hay de 100 épocas a 500 es muy reducido, para el hecho de que tome 4.9 veces más tiempo el entrenamiento. De acuerdo con los resultados se obtiene un índice de similitud de 0,983 lo que quiere decir que el ojo humano practicamente no nota diferencias entre las imágenes. Esto se respalda por el hecho de tener un MSE de orden 10^{-7} , por lo que error pixel a pixel es muy bajo.

Por otro lado, al observar las figuras 4.4 y 4.5 se puede notar un gran similitud entre la imagen generada por el modelo con el *Ground Truth* lo cual respalda los resultados de las métricas obtenidas.

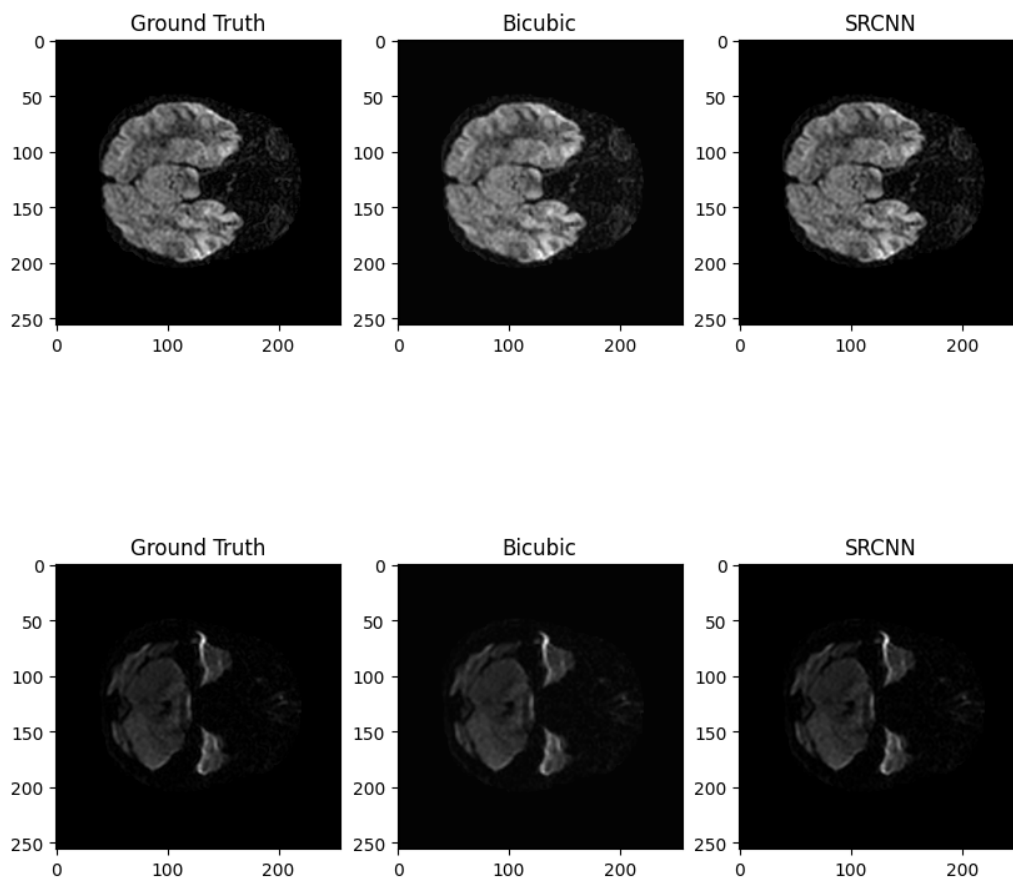


Figura 4.4: Muestra resultados modelo SRCNN

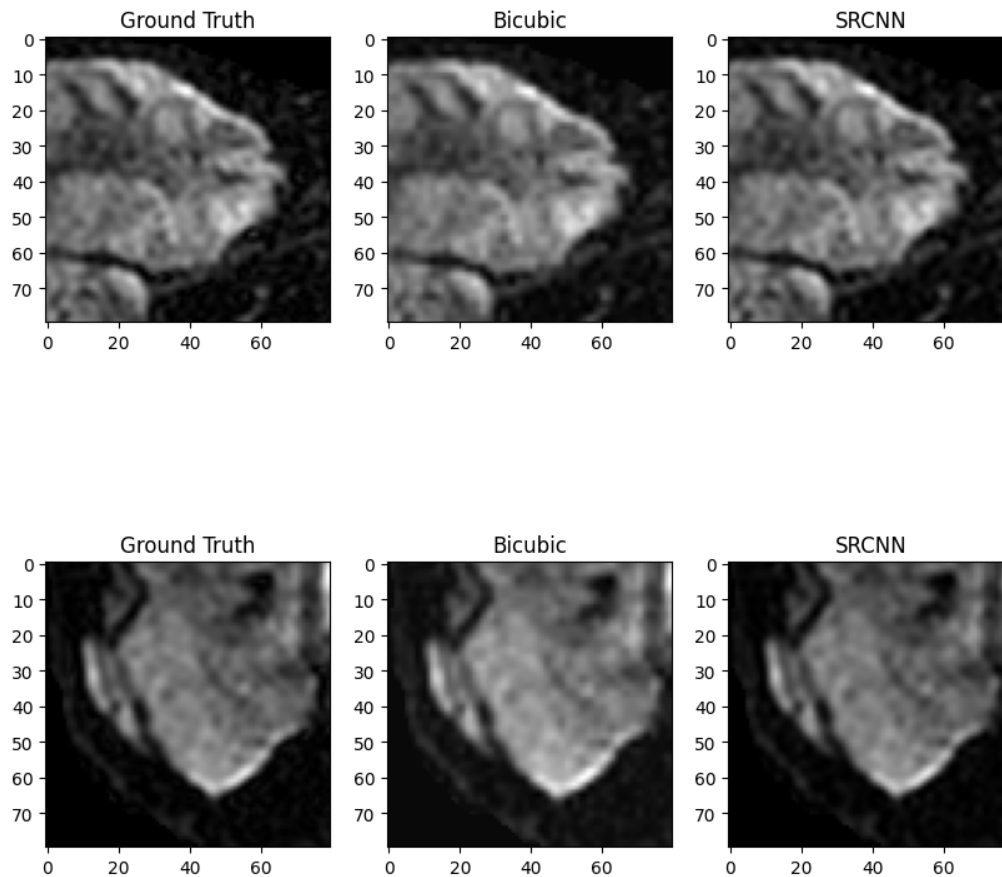


Figura 4.5: Muestra resultados modelo SRCNN Zoom

4.1.3. Métricas Tensoriales

A continuación, se muestra el desempeño obtenido a nivel tensorial para el modelo SRCNN. Tal y como se dijo anteriormente la mejora que hay de 100 a 500 épocas es muy poca, por lo que se decidió realizar el análisis sobre las métricas tensoriales tomando en cuenta el modelo entrenado con 100 épocas, de igual forma, se utiliza la máscara (recorte) debido al hecho de que se obtienen mejores resultados haciendo uso de esta.

Los resultados se resumen en el cuadro 4.3, donde se muestran las métricas FNORM y MSE FA junto con sus respectivas desviaciones estándar. Al analizar los resultados del modelo SRCNN, se puede notar que logra una buena aproximación, evidenciado en bajos errores cuadráticos medios, una FNORM baja y consistencia en estos valores reflejado en desviaciones estándar pequeñas para cada valor correspondiente. Vale la pena resaltar el hecho de que a nivel de FA el modelo tiene un peor rendimiento que a nivel de imagen ya que por un lado el MSE de la imagen es de orden 10^{-7}

mientrasas que en FA es 10^{-2} . Adicionalmente en las figuras 4.6 y 4.7 se puede apreciar la gran similitud de los mapas de anisotropía fraccional.

Métrica	Valor
FNORM	$1,16 \times 10^{-4} \pm 3,20 \times 10^{-4}$
MSE FA	$1,34 \times 10^{-2} \pm 8,08 \times 10^{-3}$

Cuadro 4.3: Resultados de las métricas tensoriales para el modelo SRCNN

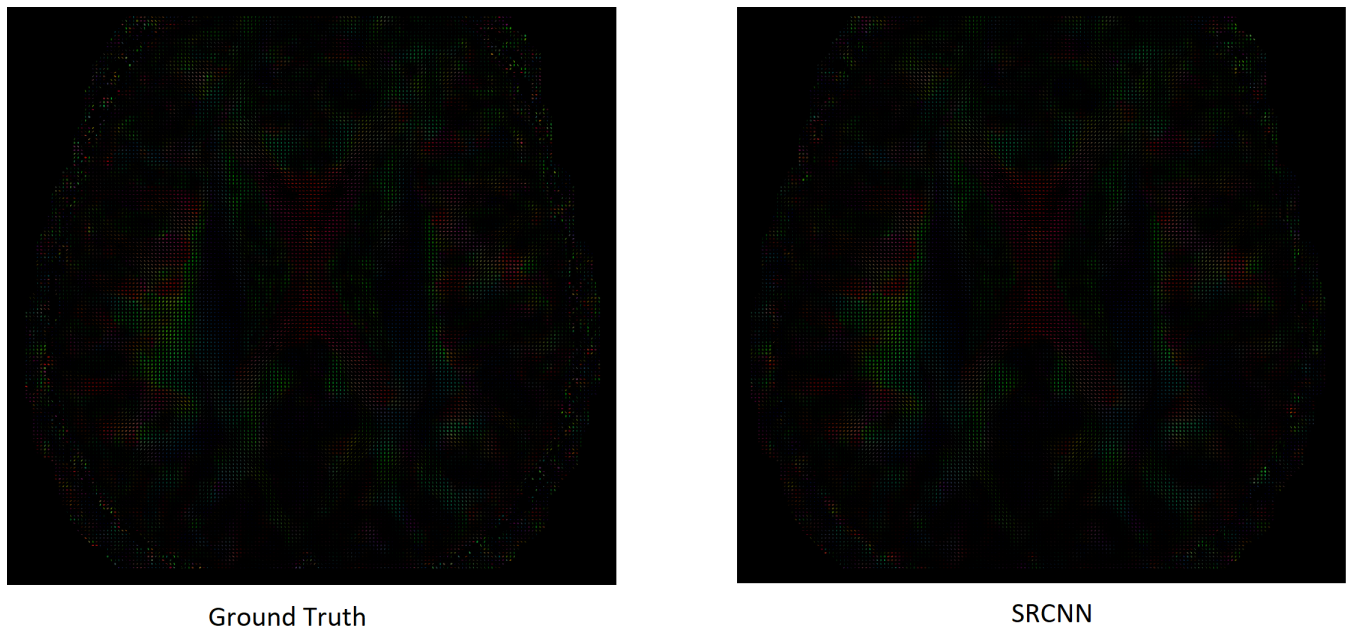


Figura 4.6: Muestra de resultados en Mapa de Anisotropía Fraccional modelo SRCNN

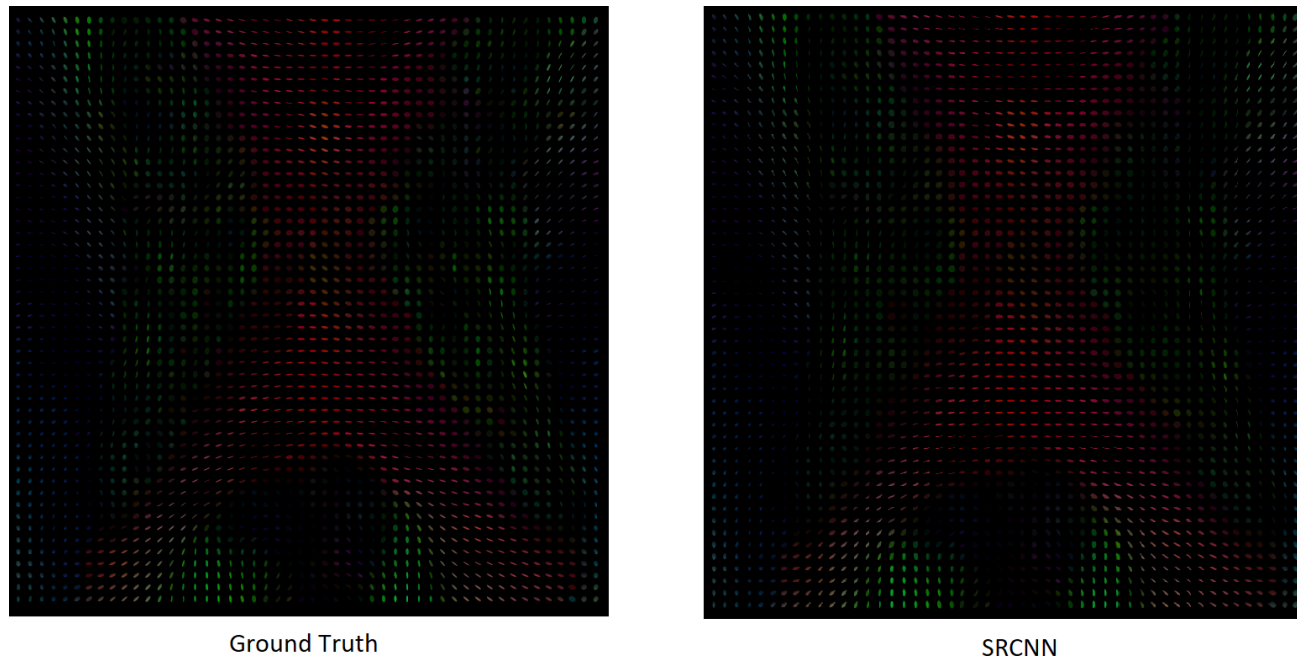


Figura 4.7: Muestra de resultados en Mapa de Anisotropía Fraccional zoom modelo SRCNN

4.2 Modelo VDSR

En esta sección se mostrarán los resultados obtenidos para el modelo VDSR con los diferentes entrenamientos realizados. Las figuras 4.8, 4.9 y 4.10 muestran el training y validation loss a lo largo de las épocas para 10 100 y 500 épocas respectivamente, con un *batchsize* de 32.

Estos valores se utilizaron, ya que Google Colab realiza una penalización por el uso de sus recursos, y al tener que probar con más algoritmos se decidió no tener tiempos de entrenamiento excesivamente largos. Por otro lado, se notó que el modelo no tenía mejoras sustanciales al pasar de 100 a 500 épocas, por lo que no valía la pena entrenar el modelo con un número mayor, como se puede ver en las figuras y en la cuadro de métricas 4.5. Además, en comparación entre el entrenamiento con 100 épocas y 500 épocas, se consumió más del doble del tiempo.

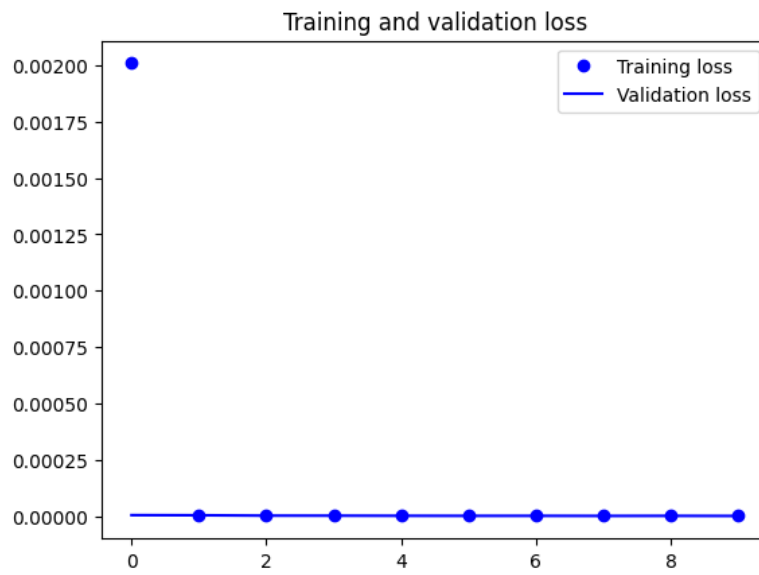


Figura 4.8: épocas vs Training loss y validation loss; épocas = 10

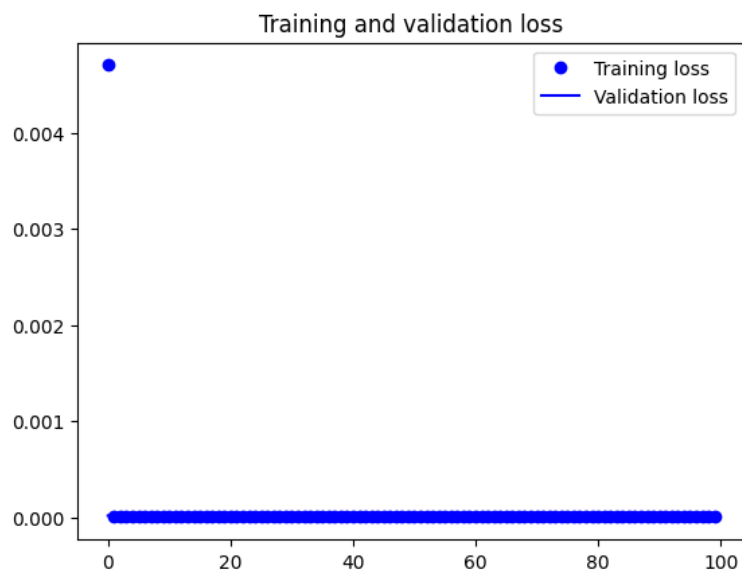


Figura 4.9: épocas vs Training loss y validation loss; épocas = 100

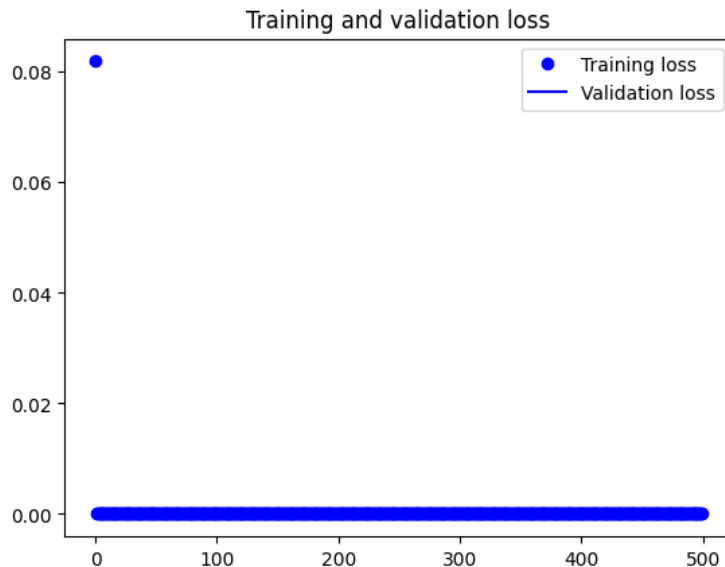


Figura 4.10: épocas vs Training loss y validation loss; épocas = 500

4.2.1. Tiempos de Ejecución

Se evaluaron los tiempos de ejecución del entrenamiento utilizando diferentes cantidades de épocas en un entorno con un tamaño de lote o *batchsize* de 32. Los resultados se presentan en el cuadro 4.4. Los tiempos de ejecución se midieron en segundos y muestran la duración promedio necesaria para completar cada conjunto de épocas.

épocas	Tiempo(s)
10	130.351
100	1039.032
500	2316.302

Cuadro 4.4: Tiempos de entrenamiento dadas n épocas con *batchsize* 32 modelo VDSR

4.2.2. Métricas de Imágenes

En esta sección, se presentan las métricas obtenidas durante el entrenamiento y evaluación de este modelo. Como se mencionó, se variaron la cantidad de épocas. Los resultados se muestran en la Cuadro 4.5 y se comparan dos escenarios: uno sin recorte y otro con recorte.

Se presenta tanto el valor promedio como la desviación estándar de cada métrica para proporcionar una medida de consistencia en los resultados. Se observa que a medida que aumenta el número de épocas, las métricas de evaluación mejoran tanto en el caso sin recorte como en el caso

con recorte y se nota un aumento significativo en la SSIM y el PSNR, la mejora relativamente pequeña que hay en PSNR es debido a que es una métrica logarítmica por lo que este es considerado una mejora sobresaliente, además, la métrica de SSIM en 0,985 da a entender que el ojo humano difícilmente puede notar diferencias en las imágenes de *ground truth* y la generada por el modelo. Asimismo, se observa una disminución en el MSE y el RMSE, lo que indica una reducción en el error de reconstrucción. También, como se puede ver en el mismo cuadro de métricas, al recortar las imágenes con la máscara de recorte y eliminar el ruido, se obtuvo una mejora significativa en las métricas.

Estos resultados se ven respaldados por la figuras 4.11 y 4.12 donde se puede apreciar la gran similitud que existe con el resultado esperado, *Ground Truth* y los resultados obtenidos por el modelo VDSR.

Métricas con 10 épocas		
	Sin recorte	Con recorte
SSIM	$8,81 \times 10^{-1} \pm 3,75 \times 10^{-2}$	$9,41 \times 10^{-1} \pm 1,42 \times 10^{-2}$
MSE	$2,21 \times 10^{-6} \pm 2,78 \times 10^{-6}$	$1,76 \times 10^{-6} \pm 2,67 \times 10^{-6}$
RMSE	$1,41 \times 10^{-3} \pm 4,85 \times 10^{-4}$	$1,24 \times 10^{-3} \pm 4,74 \times 10^{-4}$
PSNR	$1,06 \times 10^2 \pm 2,52$	$1,07 \times 10^2 \pm 2,56$
Métricas con 100 épocas		
	Sin recorte	Con recorte
SSIM	$9,63 \times 10^{-1} \pm 1,01 \times 10^{-2}$	$9,84 \times 10^{-1} \pm 8,25 \times 10^{-3}$
MSE	$4,40 \times 10^{-7} \pm 2,25 \times 10^{-7}$	$2,12 \times 10^{-7} \pm 1,66 \times 10^{-7}$
RMSE	$6,46 \times 10^{-4} \pm 1,52 \times 10^{-4}$	$4,46 \times 10^{-4} \pm 1,12 \times 10^{-4}$
PSNR	$1,12 \times 10^2 \pm 2,20$	$1,15 \times 10^2 \pm 1,77$
Métricas con 500 épocas		
	Sin recorte	Con recorte
SSIM	$9,65 \times 10^{-1} \pm 9,48 \times 10^{-3}$	$9,85 \times 10^{-1} \pm 8,15 \times 10^{-3}$
MSE	$3,90 \times 10^{-7} \pm 1,70 \times 10^{-7}$	$1,73 \times 10^{-7} \pm 1,06 \times 10^{-7}$
RMSE	$6,10 \times 10^{-4} \pm 1,36 \times 10^{-4}$	$4,05 \times 10^{-4} \pm 9,34 \times 10^{-5}$
PSNR	$1,13 \times 10^2 \pm 2,17$	$1,16 \times 10^2 \pm 1,77$

Cuadro 4.5: Comparación de métricas con diferentes épocas y recortes modelo VDSR

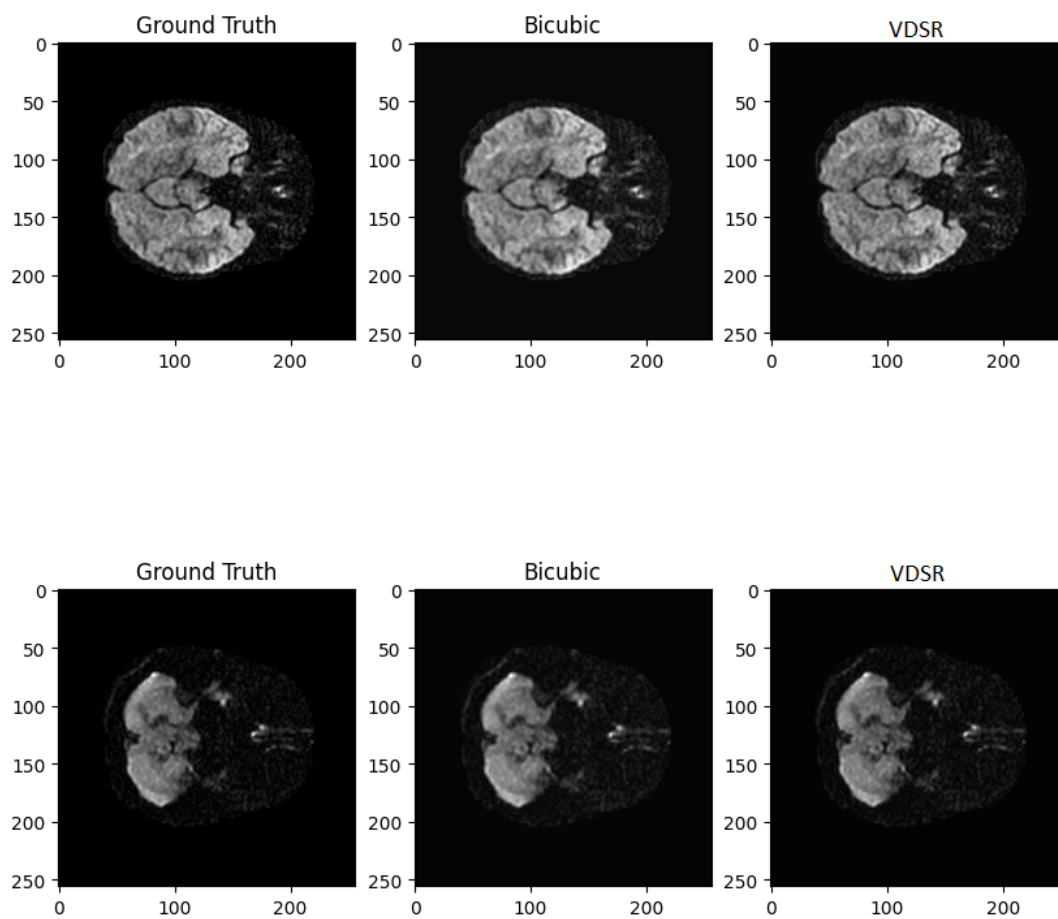


Figura 4.11: Muestra resultados modelo VDSR

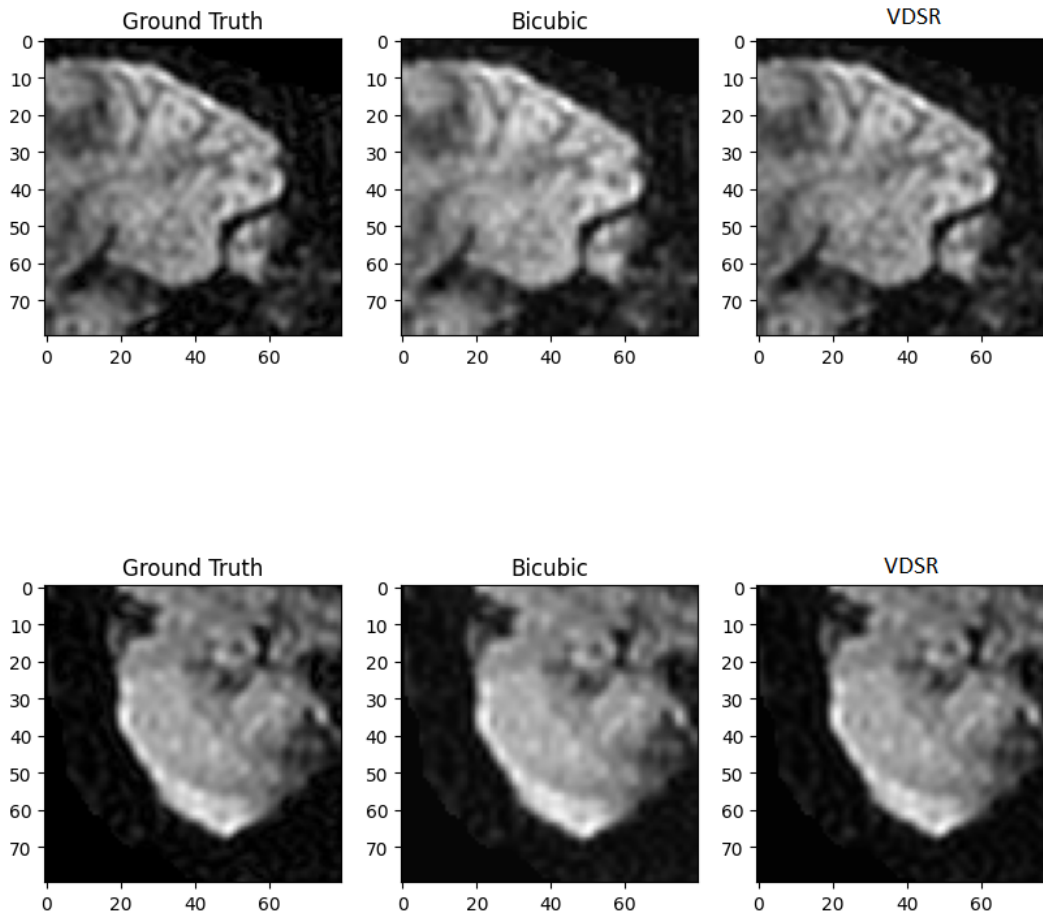


Figura 4.12: Muestra resultados modelo VDSR Zoom

4.2.3. Métricas Tensoriales

Se presentan los resultados obtenidos a partir de las métricas tensoriales ya mencionadas anteriormente. Los resultados se pueden ver en la Cuadro 4.6, donde se muestra la métrica FNORM y el MSE de FA junto con sus respectivas desviaciones estándar.

Al analizar los resultados, se puede ver que el modelo VDSR tiene un desempeño sumamente similar al modelo SRCNN, logrando también una buena aproximación que sigue siendo mejor a nivel de imagen que a nivel tensor. Esto se evidencia en el bajo valor de la FNORM y el error cuadrático medio. Esto también se puede observar en las figuras 4.13 y 4.14 donde se logra evidenciar que las diferencias son mínimas pero perceptibles al menos en los mapas de anisotropía fraccional.

Métrica	Valor
FNORM	$1,10 \times 10^{-4} \pm 3,06 \times 10^{-4}$
MSE FA	$1,28 \times 10^{-2} \pm 7,95 \times 10^{-3}$

Cuadro 4.6: Resultados de las métricas tensoriales para el modelo VDSR

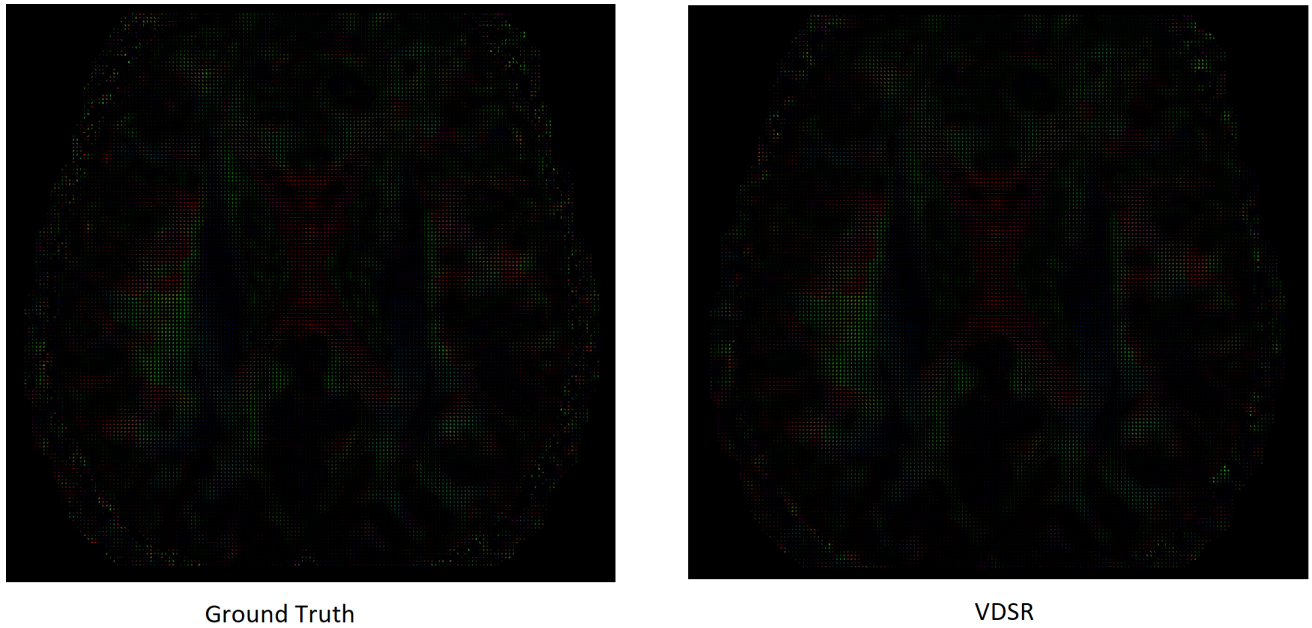


Figura 4.13: Muestra de resultados en Mapa de Anisotropía Fraccional modelo VDSR

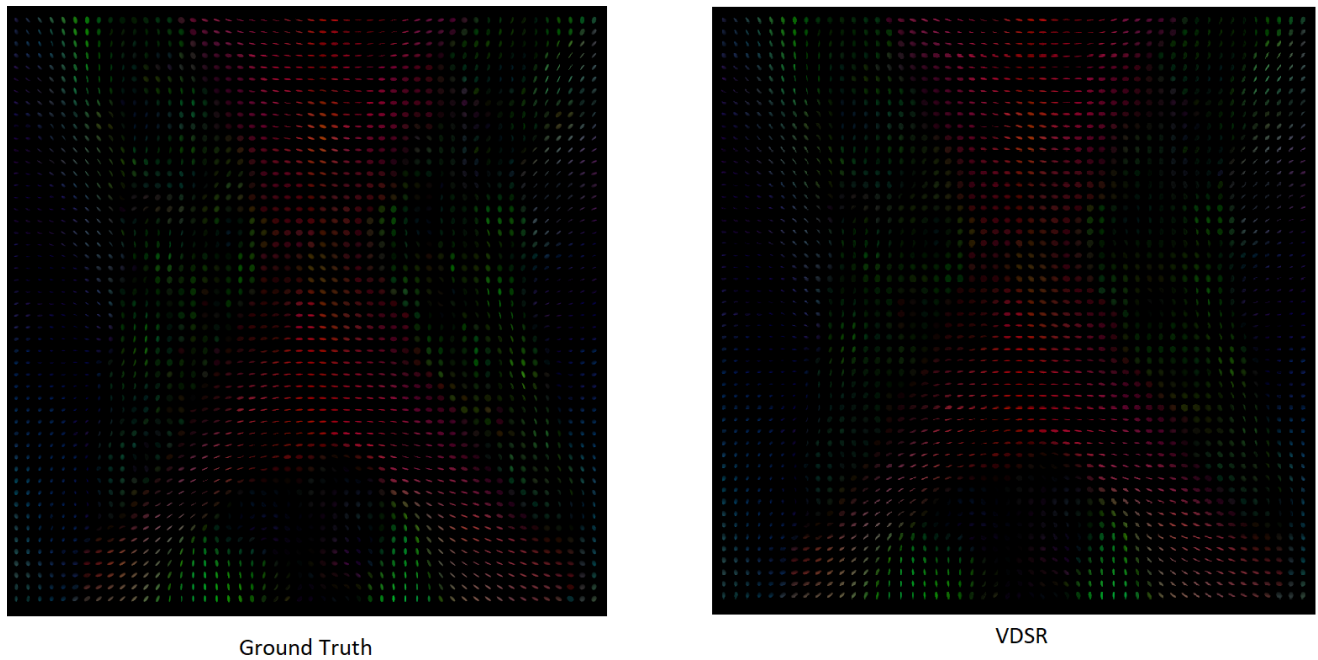


Figura 4.14: Muestra de resultados en Mapa de Anisotropía Fraccional zoom modelo VDSR

4.3 Modelo EDSR

Para el *batchsize*, inicialmente se utilizaron 3 valores (8, 16, 32), pero los resultados en las métricas eran muy similares. Entonces, para aprovechar la GPU y obtener tiempos más cortos, se fijó el batch size en 32.

En cuanto a las épocas, se hicieron varias pruebas con valores en el rango (0, 4000] y se encontró que con un número de épocas entre 2100 y 3000 comenzaba a haber sobreajuste (*overfitting*). Inicialmente, se probó con 1000 épocas y se obtuvieron buenos resultados. Luego, se probó con 2000 épocas y los resultados mejoraron un poco. Sin embargo, cuando se probó con 4000 épocas, los resultados empeoraron. Al analizar la gráfica, se notó que el modelo estaba sobre ajustado. Por lo tanto, se decidió que la mejor opción era dejar el valor de las épocas en 2000.

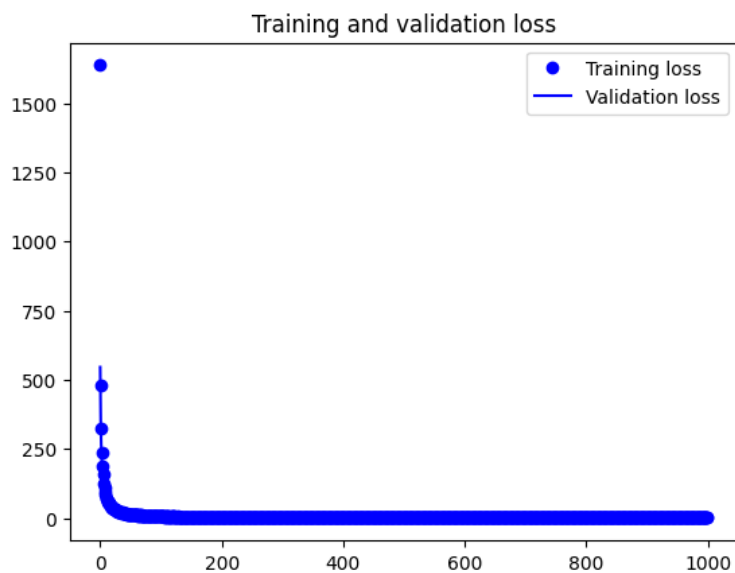


Figura 4.15: épocas vs Training loss y validation loss; épocas = 1000

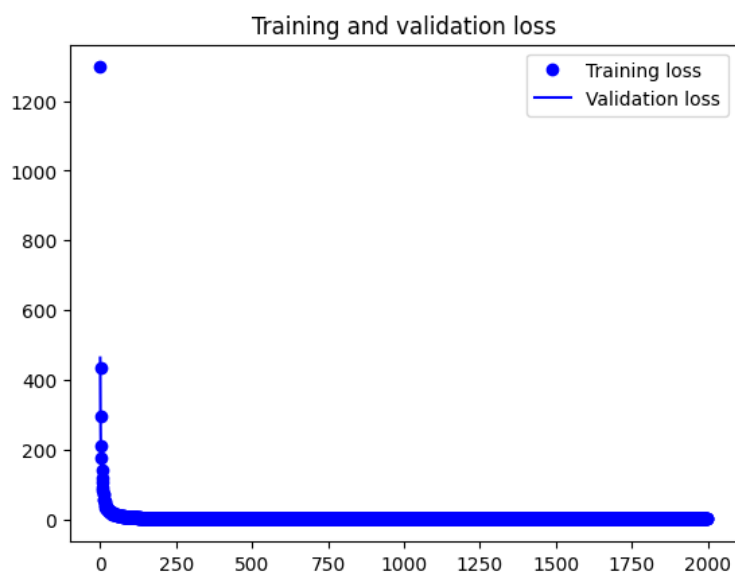


Figura 4.16: épocas vs Training loss y validation loss; épocas = 2000

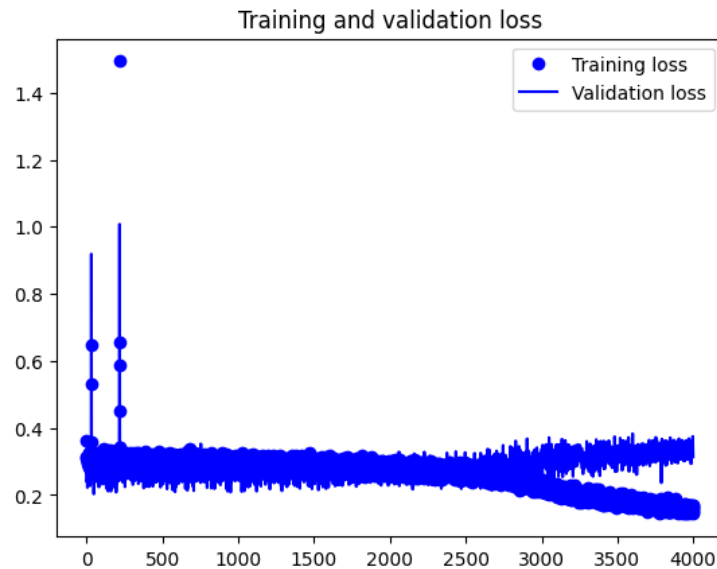


Figura 4.17: épocas vs Training loss y validation loss; épocas = 4000

4.3.1. Tiempos de Ejecución

Se evaluaron los tiempos de ejecución del entrenamiento utilizando diferentes cantidades de épocas con un *batchsize* de 32. Los resultados se presentan en el cuadro 4.7. Los tiempos de ejecución se midieron en segundos y muestran la duración promedio necesaria para completar cada conjunto de épocas.

épocas	Tiempo(s)
1000	491.168
2000	1007.573
4000	2053.907

Cuadro 4.7: Tiempos de entrenamiento dadas n épocas con *batchsize* 32 modelo EDSR

4.3.2. Métricas de Imágenes

A continuación, se presentan las métricas obtenidas durante el entrenamiento y la evaluación de la EDSR. Se realizaron experimentos utilizando diferentes cantidades de épocas para analizar el impacto en el rendimiento del modelo. Los resultados se muestran en el cuadro 4.8 y se comparan dos escenarios: uno sin recorte y otro con recorte. Se utilizaron valores de épocas de 1000 y 2000. Se presenta tanto el valor promedio como la desviación estándar de cada métrica para proporcionar una medida de consistencia en los resultados. Se observa que a medida que aumenta el número de épocas, las métricas de evaluación mejoran tanto en el caso sin recorte como en el caso con recorte. Al obtener un SSIM de 0,799 antes de recorte, se puede decir con certeza que el ojo humano

podría detectar diferencias en las imágenes, las cuales se hacen menos evidentes una vez se aplica la máscara obteniendo un valor de 0,927, a pesar de esto, el valor de MSE es considerable, dando a entender que el ojo humano no puede llegar a evaluar la imagen a nivel de detalle de pixel a pixel.

Por otro lado, gráficamente se notó cierta similitud entre la imagen generada por la EDSR y la Ground Truth, además de una mejora en cuanto a nivel de detalle comparada con la generada por el algoritmo bicúbico como se puede ver en las imágenes 4.18 y 4.19.

Métricas con 1000 épocas		
	Sin recorte	Con recorte
SSIM	$7,96 \times 10^{-1} \pm 5,16 \times 10^{-2}$	$9,26 \times 10^{-1} \pm 3,01 \times 10^{-2}$
MSE	$2,40 \times 10^{-1} \pm 2,08 \times 10^{-1}$	$1,83 \times 10^{-1} \pm 1,96 \times 10^{-1}$
RMSE	$4,69 \times 10^{-1} \pm 1,43 \times 10^{-1}$	$4,06 \times 10^{-1} \pm 1,37 \times 10^{-1}$
PSNR	$5,50 \times 10 \pm 2,35$	$5,63 \times 10 \pm 2,31$
Métricas con 2000 épocas		
	Sin recorte	Con recorte
SSIM	$7,99 \times 10^{-1} \pm 5,10 \times 10^{-2}$	$9,27 \times 10^{-1} \pm 2,94 \times 10^{-2}$
MSE	$2,35 \times 10^{-1} \pm 2,07 \times 10^{-1}$	$1,79 \times 10^{-1} \pm 1,85 \times 10^{-1}$
RMSE	$4,65 \times 10^{-1} \pm 1,43 \times 10^{-1}$	$4,23 \times 10^{-1} \pm 1,36 \times 10^{-1}$
PSNR	$5,51 \times 10 \pm 2,33$	$5,62 \times 10 \pm 2,30$

Cuadro 4.8: Comparación de métricas con diferentes épocas y recortes del modelo EDSR

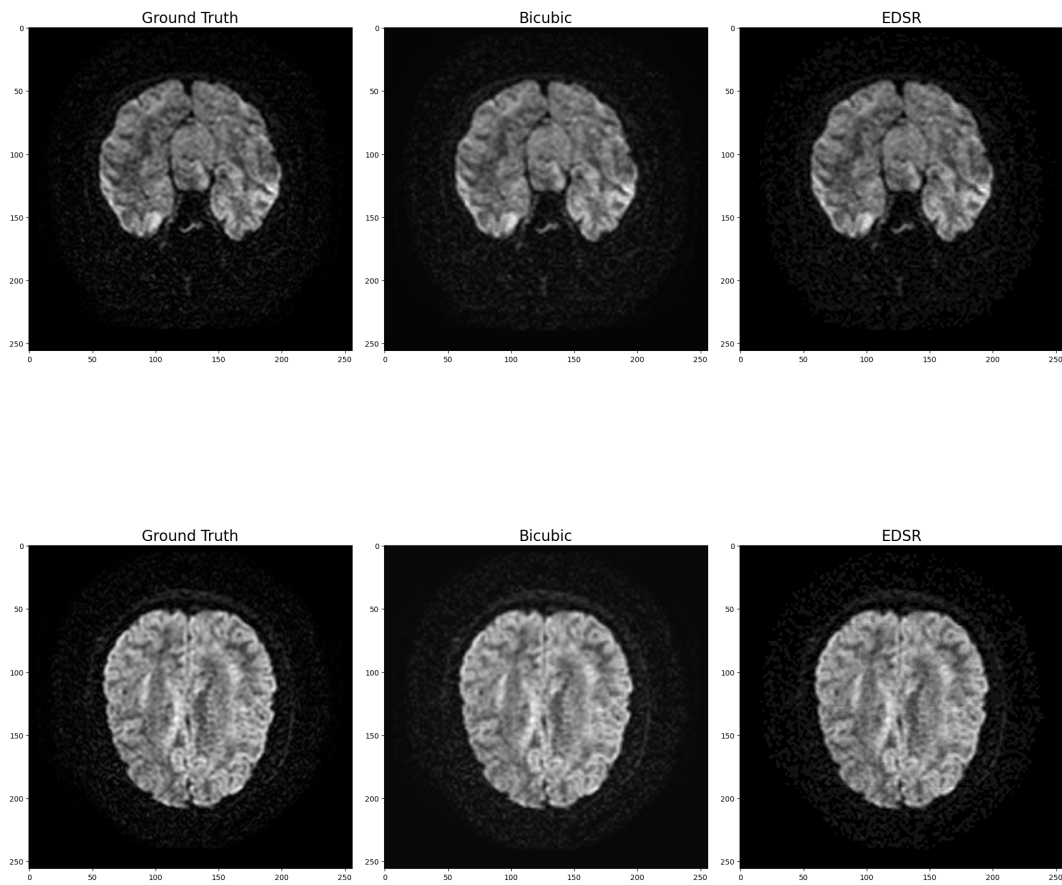


Figura 4.18: Muestra resultados modelo EDSR

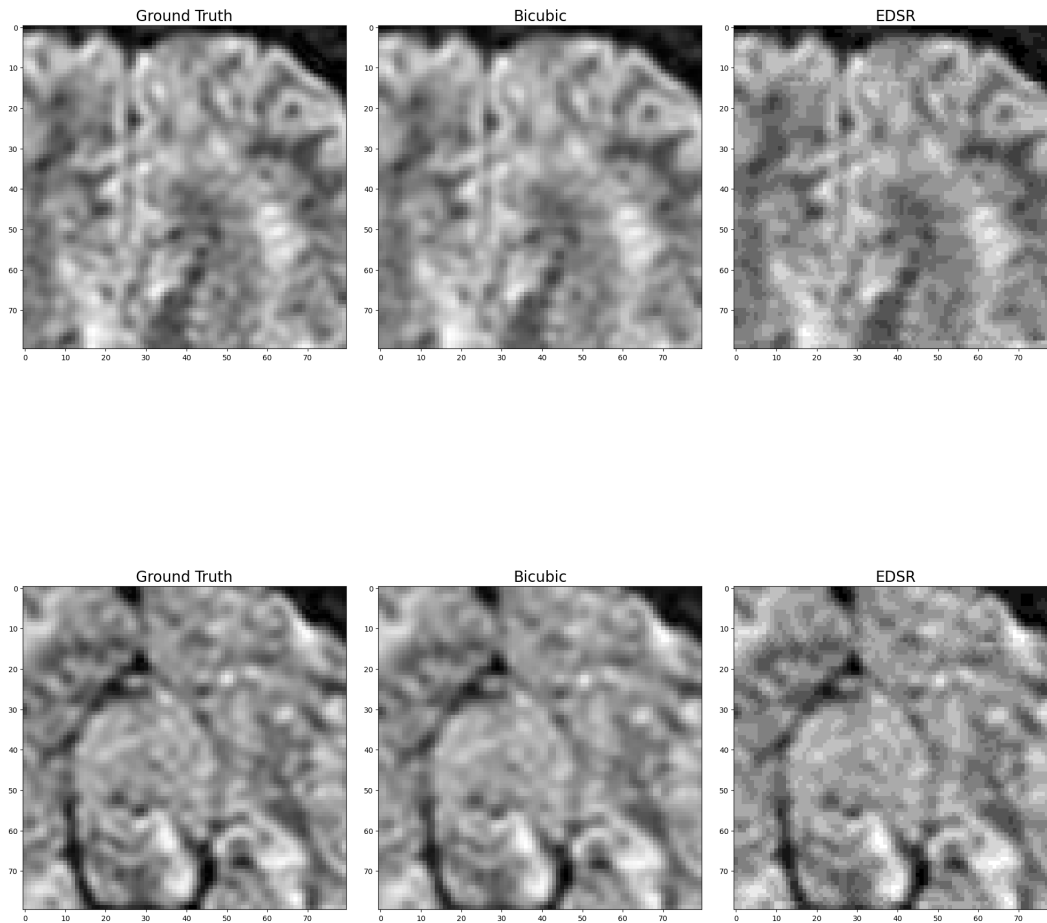


Figura 4.19: Muestra resultados modelo EDSR Zoom

4.3.3. Métricas Tensoriales

En el Cuadro 4.9 se pueden ver los resultados obtenidos con este modelo, tiene contenida la métrica FNORM y el MSE de FA junto con sus respectivas desviaciones estándar.

Al analizar los resultados, tal y como se esperaba a raíz de lo visto en el cuadro 4.8, se puede ver que el modelo EDSR no logra una buena aproximación a nivel de tensor ni de FA, obteniendo valores notablemente mayores a los demás modelos. Este desempeño a nivel tensorial era de esperarse debido al hecho de que existe un procedo de ajuste (*fit*) de los tensores a las imágenes por lo que, si no se tiene un buen desempeño a nivel de imágenes no podrá tenerse a nivel tensorial.

Métrica	Valor
FNORM	$7,90 \times 10^{-2} \pm 5,71 \times 10^{-1}$
MSE FA	$1,91 \times 10^{-1} \pm 4,29 \times 10^{-2}$

Cuadro 4.9: Resultados de las métricas tensoriales para el modelo EDSR

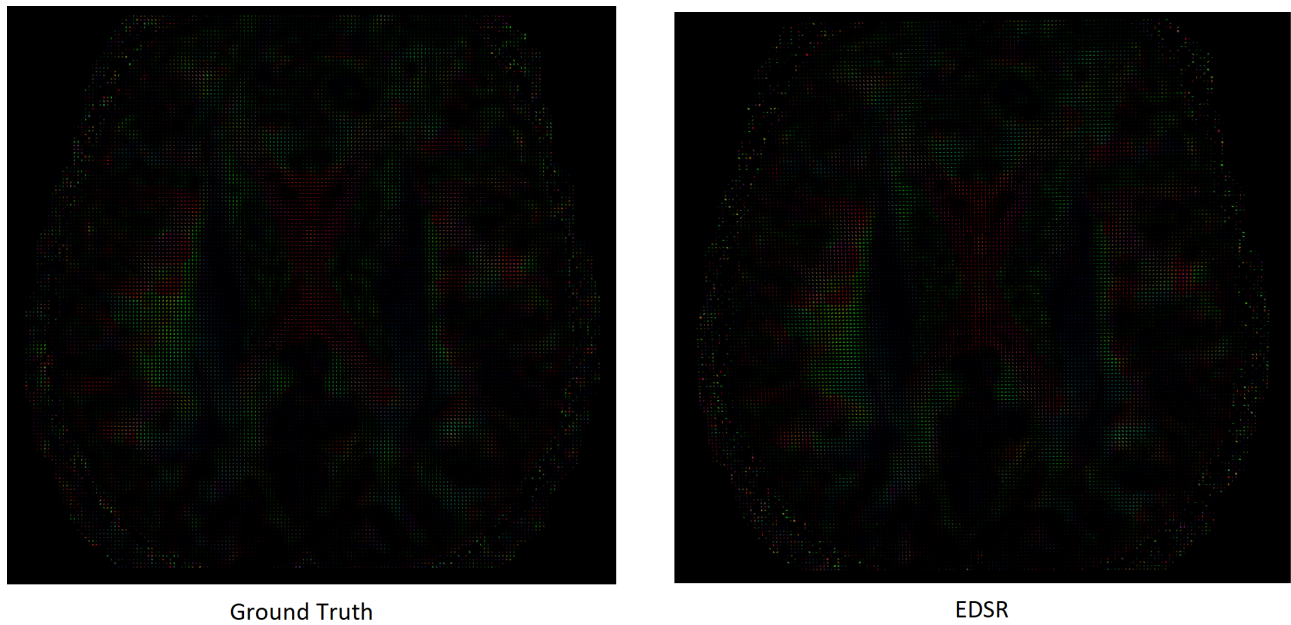


Figura 4.20: Muestra de resultados en Mapa de Anisotropía Fraccional modelo EDSR

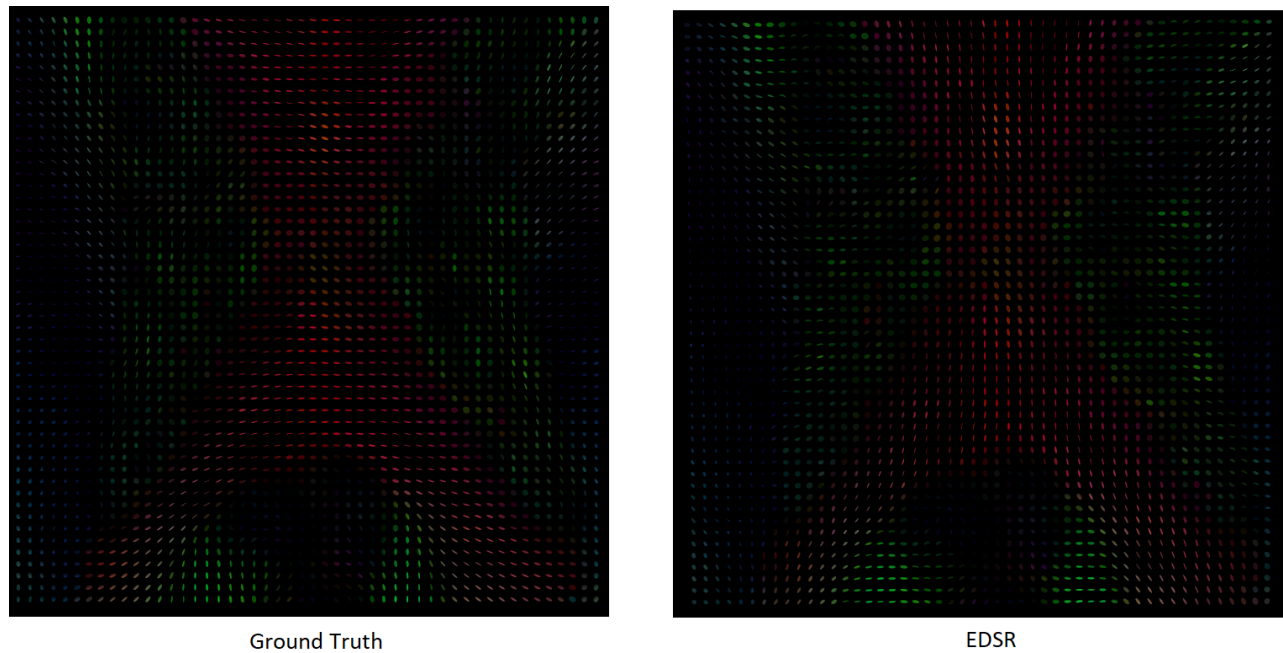


Figura 4.21: Muestra de resultados en Mapa de Anisotropía Fraccional zoom modelo EDSR

4.4 Comparación de los Modelos

En la tarea de Super Resolución de imágenes, los modelos SRCNN, VDSR y EDSR son ampliamente utilizados y han demostrado su efectividad en aumentar la resolución de imágenes de baja calidad. Analizando las métricas obtenidas en sus mejores épocas, se puede realizar una comparación entre estos modelos, entendiendo como mejores épocas aquella cantidad de épocas en las que se notó una mejora sustancial con respecto a la cantidad de épocas anterior sin recurrir a un aumento sustancial en el tiempo de entrenamiento. Debido a esto, se tomaron el número de épocas 100, 500 y 2000 para la SRCNN, VDSR y EDSR respectivamente, todas ellas haciendo uso de la máscara mencionada en la sección 3.4.

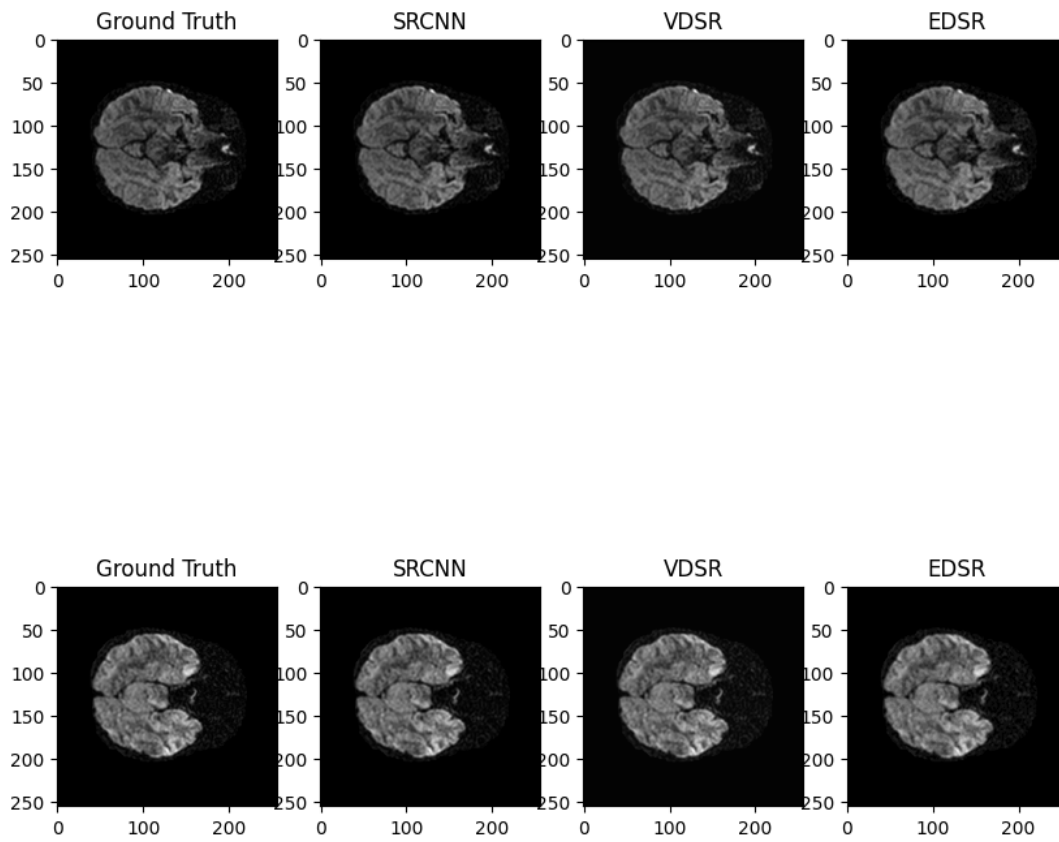


Figura 4.22: Comparación de resultados de todos los modelos

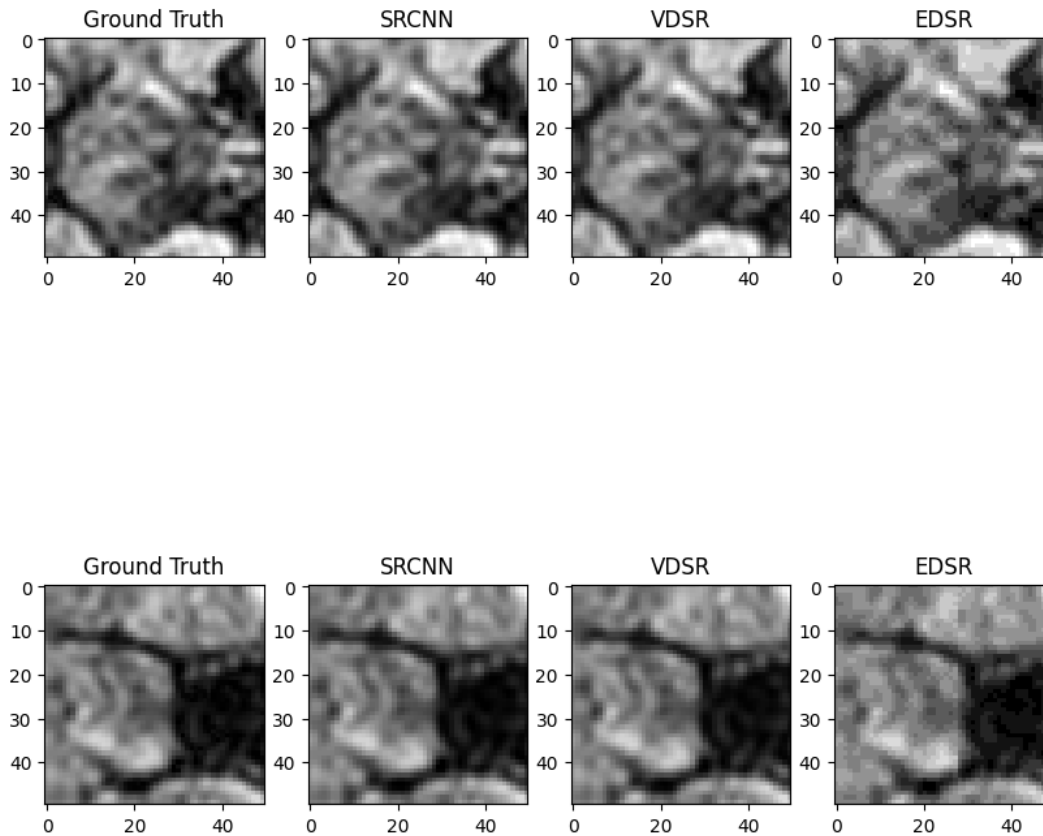


Figura 4.23: Comparación de resultados de todos los modelos zoom

En términos de calidad de imagen, tanto SRCNN como VDSR obtuvieron resultados recalca- bles que se puede apreciar en el Cuadro 4.10 así como en las figuras 4.22 y 4.23. El modelo SRCNN logró un coeficiente SSIM de 0,983, lo que indica una alta similitud estructural con la imagen de referencia. Además, el MSE obtenido por SRCNN fue de $2,05 \times 10^{-7}$, lo que demuestra una baja pérdida de información en comparación con la imagen original. Por su parte, VDSR logró un coeficiente SSIM ligeramente superior de 0,985 y un MSE de $1,73 \times 10^{-7}$, lo que sugiere una mayor fidelidad en la reproducción de los detalles y una menor presencia de errores. Comparando estos 2 modelos se puede decir que con certeza que, el humano promedio no podría notar a simple vista una diferencia evidente en la reproducción de las imágenes, esto debido a los altos SSIM obtenidos por ambos modelos, esto cambia al evaluarlos a nivel computacional o meramente numérico, pues el modelo VDSR lleva la delantera obteniendo $0,32 \times 10^{-7}$ menos de error medio cuadrático.

El modelo EDSR no alcanzó los mismos niveles de calidad que SRCNN y VDSR en esta comparación. Aunque obtuvo un coeficiente SSIM bastante alto de $9,27 \times 10^{-1}$, lo que indica que para el ojo humano la reproducción de la imagen podría considerarse adecuada, el MSE fue significativa-

mente más alto, alcanzando $1,79 \times 10^{-1}$, lo que muestra la incapacidad del ojo humano evaluar un nivel de detalle a mucha más profundidad, en este caso una comparación directa de pixel a pixel. Esto da a entender que existe una considerable pérdida de información además de una imagen más ruidosa, lo cual se ve respaldado por la métrica de PSNR de 56,2.

Al considerar las métricas tensoriales, se observa que tanto SRCNN como VDSR obtuvieron resultados similares, con valores de FNORM de $1,16 \times 10^{-4}$ y $1,10 \times 10^{-4}$, respectivamente. Estos valores indican una buena conservación de los tensores en la imagen predicha, vale la pena resaltar el hecho de que, si bien la VDSR es ligeramente mejor, esta toma más de 4 veces la cantidad de tiempo que demora entrenar la red SRCNN como se puede observar en la tabla 4.12. Por otro lado, el modelo EDSR obtuvo un valor de FNORM de $7,90 \times 10^{-2}$, lo que sugiere una capacidad inferior de preservar los tensores en la imagen mejorada, esto se puede ver claramente en la figura 4.24.

En general, 2 de los modelos obtuvieron resultados notoriamente buenos, siendo estos SRCNN y VDSR, no solo en el apartado de imágenes sino también a nivel tensorial, demostrando así que estas arquitecturas las cuales son altamente utilizadas en super resolución, en otras áreas diferentes a la médica, tienen el potencial de explorarse aún más a fondo en tareas relacionadas con las imágenes médicas. Por otro lado, la red EDSR no presenta el mismo desempeño, pudiendo esto significar que se debería explorar más técnicas de preprocesamiento o de entrenamiento para alcanzar mejores resultados.

Métrica	SRCNN	VDSR	EDSR
SSIM	$9,83 \times 10^{-1}$	$9,85 \times 10^{-1}$	$9,27 \times 10^{-1}$
MSE	$2,05 \times 10^{-7}$	$1,73 \times 10^{-7}$	$1,79 \times 10^{-1}$
RMSE	$4,38 \times 10^{-4}$	$4,05 \times 10^{-4}$	$4,23 \times 10^{-1}$
PSNR	$1,154 \times 10^2$	$1,16 \times 10^2$	$0,562 \times 10^2$

Cuadro 4.10: Comparación de métricas de imágenes para los modelos SRCNN, VDSR y EDSR.

MÉTRICA	SRCNN	VDSR	EDSR
FNORM	$1,16 \times 10^{-4}$	$1,10 \times 10^{-4}$	$7,90 \times 10^{-2}$
MSE FA	$1,34 \times 10^{-2}$	$1,28 \times 10^{-2}$	$1,91 \times 10^{-1}$

Cuadro 4.11: Comparación de métricas tensoriales para los modelos SRCNN, VDSR y EDSR.

SRCNN	VDSR	EDSR
72,804	2316,302	1007,573

Cuadro 4.12: Comparación de tiempos de entrenamiento de los modelos SRCNN, VDSR y EDSR.

Los notebooks se pueden encontrar en el siguiente repositorio de github: [Tesis de Grado](#)¹

¹<https://github.com/Juanma1909/Bachelor-Thesis>

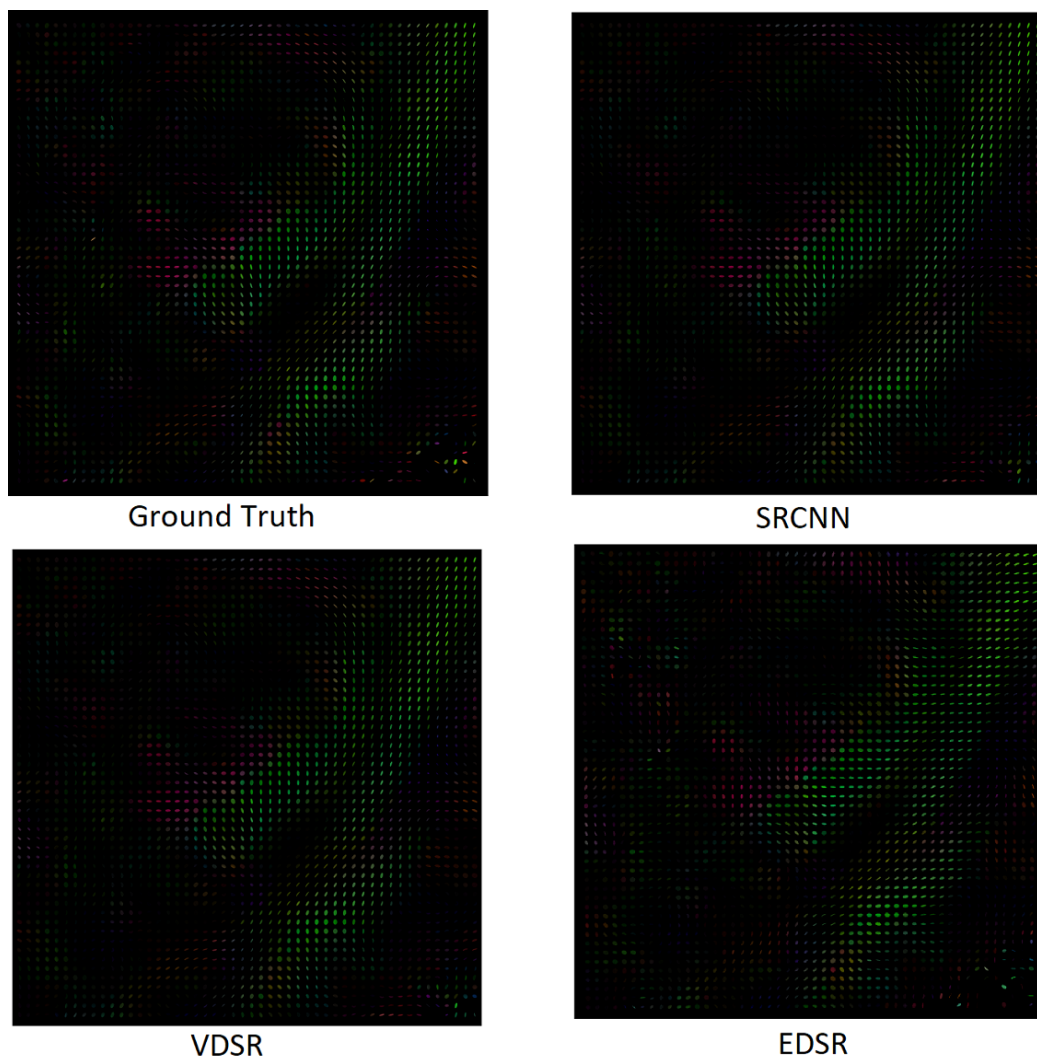


Figura 4.24: Comparación de resultados mapas de anisotropía fraccional de todos los modelos

Conclusiones y Trabajos Futuros

Este proyecto logró abordar un problema altamente estudiado, como lo es la super resolución de imágenes médicas, como se puede ver en [9, 17, 38], aplicando modelos que, si bien son muy utilizados en el campo de super resolución, no necesariamente en el ámbito de las imágenes médicas, especialmente con imágenes dMRI, logrando así estudiar su desempeño.

En primer lugar, se evaluaron y compararon tres modelos bastante utilizados en este campo: SRCNN, VDSR y EDSR. A raíz de los resultados obtenidos, se puede concluir que tanto SRCNN como VDSR demostraron ser modelos altamente efectivos en la mejora de la resolución de las imágenes dMRI, obteniendo resultados muy similares tanto a nivel de imagen como a nivel tensorial, además de esto, se logró mostrar empíricamente el hecho de que el utilizar una máscara en las imágenes para aislar la zona de interés, en este caso el cráneo, mejora notablemente las métricas de desempeño de los modelos evaluados, esto gracias a que ya no se toma en cuenta gran parte del ruido generado.

Por otro lado, el modelo EDSR no obtuvo resultados destacables, teniendo un desempeño claramente más pobre en comparación con los otros modelos estudiados, como se muestra en la sección 4.4. Las métricas de MSE, RMSE, FNROM y MSE FA obtuvieron valores mucho más altos en comparación con los otros 2 modelos, lo cual indica una calidad más baja de los resultados, con valores más dispersos y diferentes a los reales. Gracias a estos resultados se puede ver cómo el ojo humano no es capaz de detectar anomalías a mayor detalle, pixel a pixel, pues su SSIM no es necesariamente malo, sin embargo, las demás métricas mostraban un desempeño mucho más pobre a comparación de los otros modelos.

También se notó en este modelo que genera imágenes con un error mucho mayor en el área fuera del cráneo. Esto se puede apreciar debido a que las métricas mejoran considerablemente en comparación con los demás modelos cuando se le aplica una máscara, como se puede observar en la tabla 4.8.

Durante el trabajo se logró adquirir conocimiento en las áreas de *Deep Learning*, logrando modificar los modelos de tal manera de que estos pudiesen ser entrenados con el conjunto de imágenes dMRI, y del procesamiento de imágenes, como se evidenció en el apartado de 3.3, donde se manipuló el conjunto de datos de diversas maneras incluso llegando a crear una máscara capaz de aislar el cráneo en las imágenes.

En definitiva, los modelos SRCNN y VDSR se destacaron como opciones muy efectivas en el mejoramiento de la resolución espacial en imágenes dMRI, logrando una alta similitud estructural y una baja pérdida de información, además, la SRCNN mostró una alta eficiencia temporal con respecto a las otras 2 redes, pero el modelo EDSR no alcanzó los mismos niveles de calidad ya que no obtuvo resultados tan notables y podría ser objeto de futuras investigaciones y mejoras.

Este proyecto puede tomarse como una base para realizar un estudio más exhaustivo acerca de los diferentes modelos de *Deep Learning* para Super resolución ya existentes, tomando en cuenta una mayor variedad de modelos como los presentados por Bashir en [30], donde se pueden ver modelos de *post-upsampling*, *progressive-upsampling*, entre otros. De igual manera se sugiere usar técnicas de entrenamiento para la EDSR, entrenarla con un conjunto de datos más grande o explorar otras arquitecturas de aprendizaje profundo para mejorar aún más la resolución espacial en imágenes dMRI, así como un posible proceso de optimización de hiperparametros de las redes mediante el uso de la API de Keras llamada *KerasTuner*, la cual permite definir un modelo de tal forma que sus hiperparametros sean un espacio de búsqueda, para que a la hora de entrenar se realice una exploración de dicho espacio en diferentes *trials*.

Bibliografía

- [1] World Health Organization, “Diagnostic imaging - imaging modalities.” https://www.who.int/diagnostic_imaging/en/, May 2021.
- [2] Q. Tian, B. Bilgic, Q. Fan, C. Liao, C. Ngamsombat, Y. Hu, T. Witzel, K. Setsompop, J. R. Polimeni, and S. Y. Huang, “DeepDTI: High-fidelity six-direction diffusion tensor imaging using deep learning,” *NeuroImage*, vol. 219, no. January, p. 117017, 2020.
- [3] A. Hart, B. Smith, S. Smith, E. Sales, J. Hernandez-Camargo, Y. M. Garcia, F. Zhan, L. Griswold, B. Dunkelberger, M. R. Schwob, S. Chaudhry, J. Zhan, L. Gewali, and P. Oh, “Resolving intravoxel white matter structures in the human brain using regularized regression and clustering,” *Journal of Big Data*, vol. 6, no. 1, 2019.
- [4] F. Yang, Y. M. Zhu, J. H. Luo, M. Robini, J. Liu, and P. Croisille, “A comparative study of different level interpolations for improving spatial resolution in diffusion tensor imaging,” *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 4, pp. 1317–1327, 2014.
- [5] Y. H. Wang, J. Qiao, J. B. Li, P. Fu, S. C. Chu, and J. F. Roddick, “Sparse representation-based MRI super-resolution reconstruction,” *Measurement: Journal of the International Measurement Confederation*, vol. 47, no. 1, pp. 946–953, 2014.
- [6] H. K. Aggarwal, M. P. Mani, and M. Jacob, “MoDL-MUSSELS: Model-Based Deep Learning for Multishot Sensitivity-Encoded Diffusion MRI,” *arXiv*, vol. 39, no. 4, pp. 1268–1277, 2018.
- [7] M. Jiang, Z. Wen, L. Long, C. W. Wong, N. Ye, C. Zee, and B. T. Chen, “Assessing cerebral white matter microstructure in children with congenital sensorineural hearing loss: A tract-based spatial statistics study,” *Frontiers in Neuroscience*, vol. 13, no. JUN, pp. 1–11, 2019.
- [8] C. Eichner, M. Paquette, T. Mildner, T. Schlumm, K. Pléh, L. Samuni, C. Crockford, R. M. Wittig, C. Jäger, H. E. Möller, A. D. Friederici, and A. Anwender, “Increased sensitivity and signal-to-noise ratio in diffusion-weighted MRI using multi-echo acquisitions,” *NeuroImage*, vol. 221, no. July, 2020.
- [9] H. D. Vargas, “Probabilistic modeling of tensorial data for enhancing spatial resolution in magnetic resonance imaging,” 2018.
- [10] A. Graña, “Brief historical evolution of cancer,” vol. 5, no. 1, pp. 26–31, 2015.
- [11] F. D. Tsitsopoulos and P. P. Tsitsopoulos, “Handbook of Neurosurgery (ebook), 7th Edition, by Mark S. Greenberg,” *Acta Neurochirurgica*, vol. 156, no. 10, pp. 2019–2019, 2014.
- [12] W. Moritani, T. Ekholm, S., *Diffusion-Weighted MR imaging of the brain*. 1377.

- [13] P. Poulin, D. Jörgens, P. M. Jodoin, and M. Descoteaux, “Tractography and machine learning: Current state and open challenges,” *Magnetic Resonance Imaging*, vol. 64, no. April, pp. 37–48, 2019.
- [14] T. B. Dyrby, H. Lundell, M. W. Burke, N. L. Reislev, O. B. Paulson, M. Ptito, and H. R. Siebner, “Interpolation of diffusion weighted imaging datasets,” *NeuroImage*, vol. 103, pp. 202–213, 2014.
- [15] J. S. Isaac and R. Kulkarni, “Super resolution techniques for medical image processing,” *Proceedings - International Conference on Technologies for Sustainable Development, ICTSD 2015*, 2015.
- [16] D. C. Van Essen, K. Ugurbil, E. Auerbach, D. Barch, T. E. Behrens, R. Bucholz, A. Chang, L. Chen, M. Corbetta, S. W. Curtiss, S. Della Penna, D. Feinberg, M. F. Glasser, N. Harel, A. C. Heath, L. Larson-Prior, D. Marcus, G. Michalareas, S. Moeller, R. Oostenveld, S. E. Petersen, F. Prior, B. L. Schlaggar, S. M. Smith, A. Z. Snyder, J. Xu, and E. Yacoub, “The Human Connectome Project: A data acquisition perspective,” *NeuroImage*, vol. 62, no. 4, pp. 2222–2231, 2012.
- [17] G. Chen, B. Dong, Y. Zhang, W. Lin, D. Shen, and P. T. Yap, “XQ-SR: Joint x-q space super-resolution with application to infant diffusion MRI,” *Medical Image Analysis*, vol. 57, pp. 44–55, 2019.
- [18] F. Zhang, G. Xie, L. Leung, M. A. Mooney, L. Epprecht, I. Norton, Y. Rathi, R. Kikinis, O. Al-Mefty, N. Makris, A. J. Golby, and L. J. O’Donnell, “Creation of a novel trigeminal tractography atlas for automated trigeminal nerve identification,” *NeuroImage*, vol. 220, no. April, p. 117063, 2020.
- [19] D. H. Ryu, D. Ryu, Y. S. Baek, H. Cho, G. Kim, Y. S. Kim, Y. Lee, Y. Kim, J. C. Ye, H. S. Min, and Y. K. Park, “DeepRegularizer: Rapid resolution enhancement of tomographic imaging using deep learning,” *arXiv*, 2020.
- [20] K. Yamashita and K. Markov, *Medical image enhancement using super resolution methods*, vol. 12141 LNCS. Springer International Publishing, 2020.
- [21] K. C. Kwon, K. H. Kwon, M. U. Erdenebat, Y. L. Piao, Y. T. Lim, M. Y. Kim, and N. Kim, “Resolution-Enhancement for an Integral Imaging Microscopy Using Deep Learning,” *IEEE Photonics Journal*, vol. 11, no. 1, pp. 1–12, 2019.
- [22] F. Zhang, A. Breger, K. I. K. Cho, L. Ning, C. F. Westin, L. J. O’Donnell, and O. Pasternak, “Deep learning based segmentation of brain tissue from diffusion MRI,” *NeuroImage*, vol. 233, no. July 2020, p. 117934, 2021.
- [23] A. Mar, E. A. Tutor, R. Estepa, and A. Co, “Protocolo para analizar diffusion tensor imaging (dti) del cerebro humano,” 2016.

- [24] P. Hagmann, L. Jonasson, P. Maeder, J. P. Thiran, J. V. Wedeen, and R. Meuli, “Understanding diffusion MR imaging techniques: From scalar diffusion-weighted imaging to diffusion tensor imaging and beyond,” *Radiographics*, vol. 26, no. SPEC. ISS., pp. 205–224, 2006.
- [25] Iberdrola, “Descubre los principales beneficios del machine learning.” <https://www.iberdrola.com/innovacion/machine-learning-aprendizaje-automatico>, 2021.
- [26] Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
- [27] A. Glassner, “Deep Learning, From Basics to Practice,” vol. 2, p. 914, 2018.
- [28] “Deep learning architectures.” <https://developer.ibm.com/es/technologies/deep-learning/articles/cc-machine-learning-deep-learning-architectures>, 2021.
- [29] “¿cómo funcionan las convolucional neural networks? visión por ordenador.” <https://www.aprendemachinelarning.com/como-funcionan-las-convolucional-neural-networks-vision-por-ordenador/>, 2021.
- [30] S. M. A. Bashir, Y. Wang, M. Khan, and Y. Niu, “A comprehensive review of deep learning-based single image super-resolution,” *PeerJ Computer Science*, vol. 7, pp. 1–56, 2021.
- [31] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, “Enhanced Deep Residual Networks for Single Image Super-Resolution,” *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2017-July, pp. 1132–1140, 2017.
- [32] Y. Tai, J. Yang, and X. Liu, “Image Super-Resolution via Deep Recursive Residual Network,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 2790–2798, 2017.
- [33] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2015.
- [34] J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” 2016.
- [35] W. Yang, X. Zhang, Y. Tian, W. Wang, J.-H. Xue, and Q. Liao, “Deep learning for single image super-resolution: A brief review,” *IEEE Transactions on Multimedia*, vol. 21, pp. 3106–3121, dec 2019.
- [36] F. D. I. Médicas and F. D. E. Imágenes, “Fundamentos de Imágenes Médicas.,” pp. 1–12, 2003.
- [37] “3.1. cross-validation: evaluating estimator performance — scikit-learn 0.24.2 documentation2021.” https://scikit-learn.org/stable/modules/cross_validation.html, 2021.

-
- [38] O. J. Pellicer-valero, “Cost-free resolution enhancement in Convolutional Neural Networks for medical image segmentation,” no. October, pp. 2–4, 2020.
- [39] D. Zhu and D. Qiu, “Residual dense network for medical magnetic resonance images super-resolution,” *Computer Methods and Programs in Biomedicine*, vol. 209, p. 106330, 2021.
- [40] K. Umehara, J. Ota, and T. Ishida, “Application of Super-Resolution Convolutional Neural Network for Enhancing Image Resolution in Chest CT,” *Journal of Digital Imaging*, vol. 31, no. 4, pp. 441–450, 2018.
- [41] Q. Lyu, C. You, H. Shan, and G. Wang, “Super-resolution MRI through Deep Learning,” 2018.
- [42] Y. Chen, F. Shi, A. G. Christodoulou, Y. Xie, Z. Zhou, and D. Li, “Efficient and accurate MRI super-resolution using a generative adversarial network and 3D multi-level densely connected network,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11070 LNCS, pp. 91–99, 2018.
- [43] A. S. Chaudhari, Z. Fang, F. Kogan, J. Wood, K. J. Stevens, E. K. Gibbons, J. H. Lee, G. E. Gold, and B. A. Hargreaves, “Super-resolution musculoskeletal mri using deep learning,” *Magnetic Resonance in Medicine*, vol. 80, pp. 2139–2154, 11 2018.
- [44] Y. Chen, Y. Xie, Z. Zhou, F. Shi, A. G. Christodoulou, and D. Li, “Brain mri super resolution using 3d deep densely connected neural networks,” in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 739–742, 2018.
- [45] C. Dong, C. C. Loy, K. He, and X. Tang, “Image Super-Resolution Using Deep Convolutional Networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2016.
- [46] U. Sara, M. Akter, and M. S. Uddin, “Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study,” *Journal of Computer and Communications*, vol. 07, no. 03, pp. 8–18, 2019.